NVIDIA DRIVE OS 6.0.5.0 Linux

Release Notes

DA-08634-005
NVIDIA CONFIDENTIAL

| November 2022
| Prepared and Provided Under NDA

Table of Contents

Introduction

The NVIDIA DRIVE™ OS 6.0 Linux Release Notes are for NVIDIA DRIVE™ AGX Orin Development Kits.

Note: **This DRIVE OS and DriveWorks release may only be used for test and development.**

NVIDIA DRIVE™ OS is the reference operating system and associated software stack, which includes CUDA, TensorRT, NvMedia, NvStreams, and Developer Tools, designed specifically for developing and deploying autonomous applications on DRIVE AGX-based hardware.

DRIVE OS Development Kits

NVIDIA DRIVE™ OS Software Development Kit (SDK) is used to develop DRIVE OS applications for deployment on NVIDIA DRIVE AGX™ based hardware platforms.

NVIDIA DRIVE™ OS Platform Development Kit (PDK) is used to adapt NVIDIA DRIVE OS to run on custom hardware based on NVIDIA Automotive SoC (i.e., Orin).

DRIVE OS Base Operating Systems

DRIVE OS Linux “Standard”

DRIVE OS Linux “Standard” is a reference platform based on Ubuntu 20.04 Linux which is intended for prototyping and development of autonomous vehicle platforms. DRIVE OS Linux is production ready but does not go through the same safety assessment as DRIVE OS QNX for Safety.

The NVIDIA® DRIVE® OS 6.0 TensorRT™ 8.4.12 release includes a TensorRT Standard+Proxy package. The Standard+Proxy package for NVIDIA DRIVE OS users of TensorRT, which is available on all platforms except QNX safety, contains the builder, standard runtime, proxy runtime, consistency checker, parsers, Python bindings, sample code, standard and safety headers, and documentation. The builder can create engines suitable for the standard runtime and DLA. This release includes safety headers and the capability to build standard engines restricted to the scope of operations that will be supported by the safety and proxy runtimes in this and future NVIDIA DRIVE OS 6.0 releases.

Release Highlights

Key Features in this Release

For a complete list of new features and enhancements in this release, see New Features and Enhancements.

- Enhanced PKCS #11 security support improvements
- Mechanism introduced to provide controlled accessibility to prevent Personal Identifiable Information (PII) leakage
- Improved mechanisms to configure and identify software versions in the bootchains

Deprecations in this Release

The following items are deprecated in this release:

<table>
<tr><th>Summary</th><th>Module</th><th>Impact</th></tr>
<tr><td>Starting in 6.0.5, TA985SA Orin SKU is deprecated.</td><td>Boards</td><td>-</td></tr>
<tr><td>Deprecation of copytarget-setup-rootfs YAML and renaming of oem-config filestamps.

The copytarget-setup-rootfs YAML content is being merged into copytarget-configs as YAML contained configuration files specific to Ubuntu RFS packages.

The stamp files controlling the behavior of oem-config have been renamed (to reflect meaningful filenames) as provided in the table below.
<table>
<tr><th>Controls what oem-config prompt?</th><th>Old Stamp File Name</th><th>New Stamp File Name</th></tr>
<tr><td>EULA</td><td>/etc/nvidia/driveos_fs_eula_accepted.stamp</td><td>/etc/nvidia/oem-config/driveos_eula_accepted</td></tr>
<tr><td>Primar</td><td>/etc/nvidia/primary_usersetup_completed</td><td>/etc/nvidia/oem-</td></tr>
</table>
</td><td>Copytarget</td><td>If you directly use copytarget-setup-rootfs YAML to copy files, please switch to using copytarget-configs YAML as it contains entries previously in copytarget-setup-rootfs YAML.

For controlling oem-config using stamp files, please use the new stamp filenames from the table, as oem-config shall not recognize the old stamp filenames.</td></tr>
</table>

y user account setup	.stamp	config/primary_user_completed		
Skip or launch oem-config	/etc/nvidia/skip-oem-config.stamp	/etc/nvidia/oem-config/oem_config_completed		
Support for Vulkan SC extension VK_NV_external_sci_sync has been deprecated. This has been replaced by VK_NV_external_sci_sync2.			Graphics	Transition to the new VK_NV_external_sci_sync2 extension for 6.0.5.0. Support for the deprecated extension may be removed as soon as 6.0.6.0.
The legacy feature of private UDP VLAN 200 between Guest VM and MCU will be deprecated. You are expected to expose MCU API of DRIVE OS at Guest VM using a communication medium of the customer's choice. The UDP VLAN 200 between Guest VM and MCU will continue to be used as a reference implementation for verification of IST and DRIVE Update features but the same will have to replaced with a communication medium of your choice with the required safety/security measures.			Standard	You must configure the communication medium based on your choice and provide MCU_APIs at CCPLEX for IST_CLIENT and DRIVE Update to communicate with MCU with the required safety/security measures.

Planned Upcoming Changes

The following sections describe planned, upcoming changes.

Summary	Module	Impact
In the DRIVE OS 6.0.6.0 release, transition to K5.15 (from K5.10) starts. The transition will occur as follows: 6.0.6.0: K5.10 is the default kernel, K5.15 kernel is provided and is fully functional, but will have minimal testing. 6.0.7.0: K5.15 is the default kernel and is fully tested, K5.10 will have minimal testing 6.0.8.0: K5.15 is the default	Linux Kernel/BSP	You are encouraged to start integration of the new kernel in 6.0.6.0 for a smooth transition in 6.0.7.0.

kernel, K5.10 is removed.		
Passthrough interface access is enabled only for single virtual partition for Linux Standard AV PCT variant.	Standard	Tools, utilities using passthrough interface access may not work. You must selectively enable passthrough interface access for required virtual partition following customer documentation.
API changes are planned for selection of encoder preset configuration.	Standard	Update to the new encoder preset APIs starting in 6.0.6.0.
In 6.0.6.0 release, NVIDIA is planning to deprecate two fields from the public SIPL header here: https://tegra-sw-opengrok.nvidia.com/source/xref/stage-main_automotive/camera/fusa/sipl/include/NvSIPLClient.hpp#84 These two member variables will be removed: timeBase, captureGlobalTimeStamp The captureGlobalTimeStamp and the associated timeBase are not necessary anymore as they are replaced by frameCaptureTSC already exposed by the same struct.	Standard	Stop using the deprecated fields in 6.0.6.0 and use frameCaptureTSC instead for profiling.
NVIDIA DRIVE OS 6.0.6.0 will contain a consistency checker that includes all expected checks.	TensorRT	This release of the consistency checker performs most but not all possible checks to ensure that engines can be run in the safety runtime without invoking undefined or nondeterministic behavior. Operations within the safety scope are checked, tensor sizes and formats are checked, and inputs to each layer are analyzed to ensure no uninitialized values are read from memory. Some tactics require specialized kernels and internal data structures. Most, but not all, of these internal data structures are validated in this release. Kernels that do not have consistency checker support include: • Kernels that are used only in the first layer when optimizing three channel image input convolutions.

		○ sm80_xmma_fprop_image_first_layer_f16f16_f32_f16_nhwckrsc_nhwc_hmma ○ sm80_xmma_fprop_image_first_layer_f32f16f16_f32_f16_nchwkrsc_nhwc_hmma • Kernels used for deconvolution when certain restrictions on parameters are met. ○ sm80_xmma_deconv_implicit_gemm_interleaved_indexed_i8i8_i8i32_f32_nchw_vect_c_32kcrs_vect_n_32_nchw_vect_c_32 (strided) sm80_xmma_deconv_implicit_gemm_interleaved_indexed_i8f32_i8i32_f32_nchw_vect_c_32kcrs_vect_n_32_nchw_vect_c_32 (strided)

New Features and Enhancements

This release includes support for these new features and enhancements.

New Features for DRIVE OS

DRIVE OS Core Elimination

The intent of DRIVE OS core elimination is to allow an application VM to run on all cores, in part by tightly bounding the asynchronous work done by higher-priority servers on any cores. The DRIVE OS facilitates Guest VM(s) to fully utilize all CPU Cores in the system, and delegate CPU cycles to the services and server VMs as needed.

Core priority assignment for servers and HVRTOS processes, Dynamic HRO, ISR, timer on single LCPU2, load balancing, and performance improvement will be supported in the upcoming releases.

Splash Screen

DRIVE OS displays a splash screen graphic image via supported display interfaces when the DRIVE AGX system is powered on.

Support for Orin DRAM ECC Page Retirement

When the system is in DRAM Page Retirement mode, the DRIVE OS authenticates and encrypts the read/write operation of bad page information.

Camera Software New ISP Recoverable Error Notification

As part of ISP recoverable error handling design, a new ISP error notification SIPL::NOTIF_ERROR_ISP_PROCESSING_FAILURE_RECOVERABLE is defined.

The new notification indicates that an ISP recoverable error has happened in the last ISP processed frame for the specified pipeline. The following behaviors are expected:

- Sipl pipeline will continue streaming when ISP recoverable error happens. The notification is returned to client. The design in Camera SW leaves the decision to the client whether to take immediate action (exit from app) or continue.
- The recoverable error is reported from ISP falcon, the error is expected to be recovered by next frame. If this type of error happens continuously for several frames, the client should take safety actions.
- The problem ISP output frame can be discarded on the consumer side by checking engine status after ISP frame done fence expired.

Original camera software treats all ISP errors as fatal error, which requires pipeline stop at any ISP error. To improve error handling, SIPL camera pipeline can be recovered if ISP recoverable error (not fatal) happens intermittently during streaming .

You must update your applications to use the newly added ISP recoverable error notification.

Graceful Platform Shutdown

The DRIVE OS supports graceful platform shutdown triggered by MCU with a customer-configurable timeout wait for the platform shutdown to be completed before cutting off the power rails (unless timeout happens).

FSKP Fuse Key Encryption (FEK)

When the user of DRIVE OS enables Factory Secure Key Provisioning, and when DRIVE OS receives a request to program the FSKP Blob via <DOS::BOOT_IF#FSKP> with an FSKP key Index, the DRIVE OS performs an AES GCM decryption of the received blob using the FSKP key selected by the key index then program the decrypted fuses.

CUDLA Fences Support Timestamps

The DRIVE OS provides time stamping information associated with start and end of the processing tasks, on different hardware engines.

Library for Measuring DLA Resource Utilization

The DRIVE OS provides a mechanism to obtain the engine utilization of DRIVE hardware DLA at run-time.

The DRIVE OS provides a library for profiling the resource utilization of DLA to help debug and optimize the applications.

The DRIVE OS provides a mechanism to measure the memory footprint of the DLA loadable.

Layer-by-layer reporting with public interface and a public tool that can show the overall utilization of dla core as a percentage is planned for upcoming DRIVE OS release.

Graphics

- Support for Vulkan® SC 1.0
  - Vulkan SC is a low-level, deterministic, robust API that is based on Vulkan 1.2. This API enables state-of-the-art GPU-accelerated graphics and computation that can be deployed in safety-critical systems and that are certified to meet industry functional safety standards. Refer to https://www.khronos.org/vulkansc/ for more information.
  - Vulkan SC can also be invaluable for real-time, non safety critical embedded applications. Vulkan SC is designed to increase determinism, provide predictable execution times, and reduce the application size by shifting the preparation of the run-time application environment offline or to application initialization. This process includes the offline compilation of graphics pipelines that define how the GPU processes data and static memory allocation. These options both enable detailed GPU control that can be rigorously specified and tested. For more details see https://www.khronos.org/blog/vulkan-sc-overview.
  - Support for Vulkan SC 1.0.10 was added in 6.0.4.0, including support for VK_NV_external_sci_sync and VK_NV_external_memory_sci_buf.
  - Support for Vulkan SC 1.0.11 was added in 6.0.5.0, including support for the new VK_NV_external_sci_sync2 extension. VK_NV_external_sci_sync is now deprecated.
  - Vulkan SC loader, validation layers, and samples are supported for non-safety targets.
- Support for Vulkan® 1.3 (including the Roadmap 2022 Profile).
  - See the Vulkan 1.3 Announcement.

PKCS#11 Implementation Supports Multiple Tokens

Starting in DRIVE OS 6.0.5, NVIDIA PKCS#11 implementation supports multiple tokens, unlike previous DRIVE OS 6.0 releases where a single token instance (a single persistent storage area (ID 2) and a single set of hardware (CCPLEX)) was supported. For more information, refer to the Understanding Security section of the *NVIDIA DRIVE OS 6.0 Linux PDK Developer Guide*.

PKCS#11: Secure Persistent Client Object Storage with Synchronous I/O

When a user application invokes <DRIVEOS:PKCS11#TodoWriteObjectSynchronous> and DRIVE OS is in de-init Mode, the DRIVE OS programs the buffered Critical Security Parameters (CSPs) in Secure NOR in a single synchronous call.

Session Secret Key Derivation Using CKM_ECDH1_DERIVE with Curve25519, Curve448, secp256r1

DRIVE OS supports the following PKCS #11 cryptographic mechanisms [PKCS11-CURR-v3.0] with the listed PKCS#11 functions [PKCS11-BASE-v3.0], using the Hardware Security Offload Engine via <DRIVEOS:PKCS11#TODO> :

PKCS #11 Functions	PKCS #11 Mechanisms	Key Sizes	Standards to be compliant with
C_DeriveKey	CKM_ECDH1_DERIVE	256 bits	[NIST SP800-56A], using elliptic curve secp256r1 [SEC2-V2].
			[RFC 7748], using elliptic curve Curve25519 with CKK_EC_MONTGOMERY keys.

Support from Closed-Box, Production-Ready, FSI Binary

The DRIVE OS provides the following flavors of closed-box FSI binary for the system integrator without requiring the need to procure development AutoSAR tools/licenses.

1. Production-Ready binary to use in production
2. Development/debug binary to test error injection.

Exclusive Chip Identification (ECID)

The DRIVE OS provides System on Chip (SoC) unique Exclusive Chip Identification (ECID) with a length of 128 bits. The DRIVE OS restricts read access of the Exclusive Chip Identification (ECID) to the intended users. The DRIVE OS also provides an Exclusive Chip Identification (ECID) that remains Immutable following the initial programming.

Read-Only Access to Partitions in Inactive Bootchain

The DRIVE OS provides a configuration mechanism to read metadata (hashes, versions, partition names) from inactive-chain for a set of statically configured predefined non-system user-designated partitions.

Versioning Info to Bootchains at Partition Level for DRIVE Update to Identify the Current Version

The DRIVE OS provides a mechanism in DRIVE Update to extract the current version information included in the active and inactive boot chains and its partitions.

For each of the boot-chain it supports, the DRIVE OS shall include version information for the entire boot-chain as well as each of the partitions it comprises of.

Note:

The API shall provide 2 version related capabilities:

1. a read-only access to the version of each partition
2. ability to compare any two versions for a particular partition (i.e. whether a given version is greater, lesser, or equal to current partition version).

The version needs to be a part of each signed storage partition, rather than partition table, so that it is updated when partition content is updated.

TKE Watchdog Timer

The DRIVE OS provides TKE driver with interfaces to start, stop, kick-in the TKE_TOP watchdog timer for CCPLEX. When the configurable timeout of TKE_TOP watchdog driver expires, the DRIVE OS shall trigger only HSM error signal for Error ID in the ErrorID spec, via FSI for propagating to the MCU, without triggering CPU interrupt or SOC reset.

VIC Fences Timestamps Support

NvSciSync fences that are signaled from VIC supports timestamps are obtainable via NvSciSyncFenceGetTimestamp().

NvEnc/ OFA Fences Timestamp Support

The DRIVE OS provides time stamping information associated with start and end of the processing tasks, on different hardware engines.

Layer Wise Profiling Tool for DLA Engines

The DRIVE OS provides a tool to generate layer wise profiling information for the DRIVE hardware DLA engine.

VLAN Support for GOS VM Networking Interface

The DRIVE OS provides configurability to support up to 20 VLANs in Guest VM.

PTP Bridge to RGMII [PTP Bridge support in Orin (MGBE↔EQOS)]

The DRIVE OS provides PTP client/server functionality for Orin CCPLEX, compatible with AVNU Auto CDS profile specification 1.6, in conjunction with on-board ethernet switch in DRIVE OS Standard and Extended Safety Debug Overlay Builds.

Note:- For all known use-cases, PTP client functionality is needed on MGBE Ethernet Interface where as PTP server functionality on EQoS interface.

For all supported PTP variants, the DRIVE OS provides a configuration mechanism to enable synchronization of time values, as maintained in hardware timestamps of EQoS(1G) and MGBE(10G) Ethernet Controllers.

Note:- This synchronization is supposed to happen on external primary clock reference SYNC/Follow up packets receive frequency.

AUTOSAR EthTsyn PTP on Orin

The DRIVE OS provides PTP client/server functionality for Orin CCPLEX compatible with AutoSAR EthTSync profile specifications for AutoSAR 4.3.1, specifically supporting following features:-

1. No BMCA
2. No Announce/Signaling messages
3. Fixed path delay, no Pdelay protocol support.
4. VLAN tagged PTP frames
5. AutoSAR specific sub-TLVs attached to Follow_Up packets (including CRC for some fields of IEEE part of Follow_Up packet)
6. Configurable sync period, down to minimum 0.5 sec

Note:- For all known use-cases, PTP client functionality is needed on MGBE Ethernet Interface where as PTP server functionality on EQoS interface.

Interface for AVB/TSN Configuration

The DRIVE OS exposes driver interface to configure necessary parameters for AVB/TSN(IEEE 802.1Qav, Qbv, Qbu) HW configuration.

In Field Pub Key Revokation

Where the customer programs more than a single Root Public Key, and when the below criteria is true, the DRIVE OS revokes the OEM Authentication Public Key(s) by burning the corresponding revocation fuse bit:

- BootROM detects a firmware signature is verified with a Root Public Key higher than the default fused Root Public Key;

- The deployed firmware issues a request to burn the corresponding fuse bit via <DRIVE_OS::BOOT_IF#Revoke_RootKey>

Prioritized Access to Storage Partitions

Note: **This feature only applies to NDAS profiles, not Ecosystem profiles.**

NVIDIA is delivering prioritized access to storage partitions. Customers adding custom storage partitions must modify storage partition priorities by referring to below documentation as applicable. For more information, refer to the Partition Priority field of virtual_storage_ivc_ch partition attribute table in the Partition Attributes Table chapter of the *NVIDIA DRIVE OS 6.0 Developer Guide*.

VPU Utilization

Two file nodes are added to support calculating statistics on VPU utilization for VPU 0 and VPU 1:

- /sys/kernel/debug/pva0/stats_enabled: this is a boolean and accepts Y/N for enable/disable. Write Y to it to enable stats capture.
- /sys/kernel/debug/pva0/vpu_stats: this returns two integer values separated by new line. The integer values can be divided by 100 to get percentage utilization.

PVA users can calculate statistics on VPU utilization for VPU 0 and VPU 1 using the added file nodes.

FSI CCPLEX

- Added run, deinit_prepare and deinit callback registration with nvdvms for NvFsiCom.
- NvFsiComWaitForEvent() return values updated. It can additionally return -EINTR and -ETIMEDOUT.

New Features for DriveWorks 5.8

This release includes support for these new features and enhancements.

Installation and Getting Started

- DriveWorks 5.8 is installed with DRIVE OS 6.0.5.0. No separate installation of DriveWorks libraries are needed.
- Please refer to the Getting Started section of the DriveWorks SDK Reference Documentation for information about how to verify the installation and get started developing with DriveWorks.
- DriveWorks samples and data are not installed on the target OOBE RFS for DRIVE Linux, as they would occupy too much space. Refer to the Getting Started section of the DriveWorks SDK Reference Documentation for information about building and running samples on Orin.

Note: The existing VehicleIO structures are being deprecated and will be replaced with new structures and supporting functions in a future release. Please see the VehicleIO module section in the DriveWorks SDK Reference for more information on the new structures and functions.

Note: STM's dla_simple sample source code is still present even though STM does not support DLA scheduling anymore.

New Features for TensorRT 8.4.12

TensorRT Standard Build

The TensorRT 8.4 release includes changes to the standard builder and runtime that appear in TensorRT for DRIVE OS 6.0. For more information, refer to the TensorRT 8.4.1 Release Notes.

Relaxed Engine Version Check

This release supports relaxed engine version checks on safety runtime and consistency checker for safe engines that meet certain requirements. Engines that meet the requirements might be qualified as being minimally compatible. Refer to the *NVIDIA DRIVE OS 6.0 Safety Developer Guide* for more information.

Documentation Changes

The TensorRT 8.4.12 documentation has been updated accordingly:

- The *NVIDIA DRIVE OS 6.0 TensorRT 8.4.12 Developer Guide* is based on the enterprise TensorRT 8.4.x release. We have modified the TensorRT 8.4.x Developer Guide documentation for DRIVE OS 6.0.5 accuracy. The TensorRT safety content has been removed.
- The TensorRT safety content is in the *NVIDIA DRIVE OS 6.0 Safety Developer Guide*. Refer to this PDF for all TensorRT safety specific documentation.

DLA Support for `IShuffleLayer`

TensorRT 8.4.12 now supports offloading `IShuffleLayer` to DLA. Refer to *Working with DLA* in the *NVIDIA DRIVE OS 6.0 TensorRT 8.4.12 Developer Guide* for details on the restrictions for running `IShuffleLayer` on DLA.

Fixed Issues

The following DRIVE OS issues from the previous release are resolved in this release:

Feature	Module	Description
3445088	MCU Firmware	IST specific VRS12 threshold values are not available and thus default value used during IST mode
3469587	BPMP	AXI_CBB clock is running at 408MHz frequency, the POR value for use case is 204MHz
3478510	Multimedia	In DRIVE OS 5.2 all NvMedia engine APIs were based on NvMediaImage. But that is being removed in 6.0. In 6.0 there's a new set of APIs that are based on NvSciBuf. These APIs are incompatible with the old 5.2 APIs and customers will need to port their applications to use the new API. The new APIs are in the mentioned include/nvmedia_6x folder and the old APIs will be removed in a later DRIVE OS 6.0.x.0 release.
3480301	Bootloader	Device failed to boot (hung in QB) after PKC (RSA 3K) fusing.
3436673	Bootloader	P3663 ES was not flashing/booting after FSKP changes. This issue has been resolved.
200778180 3416653	DriveWorks	The CGF Demo is not functional on the host or target. DriveWorks Graph UI Tool is not included in the packages; it can be provided upon request.
3719823	Image and Point Cloud Processing	Image Processing is not supported.
200777663	SAL	LRAW recording works with an additional sensor parameter “encoder instance=0”. h264/5 recording is unaffected.
200778225	SAL	Camera Server Client crashes.
200778085 3409980	SAL	Video Exporter Tool fails.
3777686	Image and Point Cloud Processing	Recorder app with 4 cam recording crashes Segmentation fault : NvMapMemAllocInternalTagged: 1074810371 error 12 NvMapMemHandleAlloc: error 0 (nvidia.com)
3795934	STM	STM and SSM samples binary not generating after compilation and cross compilation. sample_image_pyramid_pva is also missing

3746011	CGF	Sample_cgf_camera_interprocess failing with error: Failed with NvSciError_BadParameter(256) in src/dwcgf/channel/impl/ChannelNvSciStream_new.hpp:466
3790584	CGF	CGF Demo tool shows inconsistent behavior. Generates random exit codes. 207, 6, 0 etc.
200776374	General	Every GUI based sample dumps error on console: Failed to list sessions: Unit dbus-org.freedesktop.login1.service is masked.
3750024	AURIX	Occasionally SDKManager Flashing step will fail due to garbled control messages between the host and target device.
3741367	Bootburn	flash_bsp_images.py fails with -x option.
3712840	Connectivity	With tegrareset or aurix reset, the Guest VM log in the serial console stops at: `pl:[I]: jumping to kernel at 0x80200000 (virtual 0x80000000)`
3698885	Video	The system mLockTimeout will occur during the Decoding process for all those frames where Decoding time > mLockTimeout. For High Bitrate streams (mostly of 8K resolution), which do not decode in real time, the Decode Time may exceed the NVDEC mLocktimeout set. Hence certain streams that can't decode in real time due to very High Bitrate, may fail during Decoding on (AV +Q) and (AV + L) platforms .
CUDA	CUDA	During ongoing testing NVIDIA identified that due to a rounding algorithm error in a very small number of corner cases (less than 0.0000005% of tested combinations) 64-bit floating point division results can differ from the IEEE754 standard by 1 least-significant bit. Floating point operations have many sources of error accumulation and most algorithms will not have encountered this discrepancy. NVIDIA recommends that all developers requiring strict IEEE754 compliance update to CUDA Toolkit 11.7 Update 1 or newer. The affected algorithm is present in both offline compilation as well as just-in-time (JIT) compilation. As JIT compilation is handled by the driver, NVIDIA recommends updating to driver version 515.48.08 or newer for full IEEE754 compliance when required and when using JIT. This issue was identified late in the development cycle and will be addressed in the next DRIVE OS software release.
3614812	DriveWorks: SAL	LRAW format recording is not working with sample_camera.
3754693	DriveWorks: Image and Point Cloud Processing	Sample_calibration_stereo, camera_seek, video_rectifier dumps error on exit: "Bus Error", functionality wise working fine
3776375	DriveWorks	Sample_video_rectifierLDC fails to record the screen, crash with segmentation fault
3776370	DriveWorks	Sample_image_capture fails to capture the screen, DW_NOT_AVAILABLE: dwFrameCapture_initialize is not available for serializeGL is truein current version
3776381	DriveWorks	Sample_video_rectifier recorded mp4 video fails to playback:

		Driveworks exception thrown: DW_FILE_INVALID: ContainerMp4: error loading header
3755898	DriveWorks	Playback of H264 video is corrupted which is extracted using extractLRawPreview tool.
3741019	DriveWorks	Sample_video_rectifier crash on exit: DW_INVALID_ARGUMENT: SyncNvMedia2D::fillNvSciSyncAttrs: NvMedia2DUnregisterNvSciSyncObj failed.
3404259	DriveWorks: DNN Framework	Sample_dnn_plugin with custom digits fails to detect drawn digit on output window. Always show Detected digit: 1
3744318	DriveWorks: Calibration	Luminar H3 lidar self-calibration using DW sample sample_calibration_lidar is not working
3773202	DriveWorks: General	Replayer tool dumps segmentation fault (core dumped) at the end of exit. Functionality is working properly

NVIDIA Software Security Updates

This release of NVIDIA DRIVE OS 6.0 Linux includes updates that address the following issue[s]:

CVE ID	NVIDIA Issue Number	Description
Not Assigned	3101503	NVIDIA DRIVE OS Security Services contains a vulnerability in PKCS#11 Key Store, where an attacker with physical access to the platform can cause the degradation in cryptographic security strength for customer applications using AES-CTR or AES-CMAC, which may lead to information disclosure, tampering of data, or denial of service.
CVE-2022-23960	3681680	Computer systems with microprocessors utilizing speculative execution and indirect branch prediction, contain a cache speculation vulnerability known as Branch History Injection (BHI) or Spectre-BHB, where an attacker with local user access who performs a complex Spectre v2 style attack but using the shared branch history, may be able to infer sensitive information causing information disclosure.
Not Assigned	3711119	NVIDIA distributions of Drive 6.0 and Orin contain a vulnerability in hardware initialization, where failure to enable stack canary randomization, may allow a local attacker to compromise Integrity and Confidentiality, and cause Denial of Service.
Not Assigned	3776894	NVIDIA Tegra kernel driver for Linux contains a vulnerability in NVIDIA camera, where failure to check input from an untrusted source, may allow an attacker

		with normal user privileges to cause limited Denial of Service.
CVE-2018-25032	3790367	NVIDIA distributions of Linux contains a vulnerability in zlib, where zlib before 1.2.12 allows memory corruption when deflating (i.e., when compressing) if the input has many distant matches.
CVE-2020-24977	3794223	NVIDIA Tegra kernel driver or Windows/Linux GPU Display Driver contains a vulnerability in component i.e., NVHost, NVMAP, NVIDIA camera or the kernel mode layer (nvlddmkm.sys) handler for DxgkDdiEscape where CWE description (see below text), which may lead to impact from the impact field.
CVE-2022-25315	3794223	NVIDIA Tegra kernel driver or GPU Display Driver for Windows and/or Linux contains a vulnerability in component i.e., NVHost, NVMAP, NVIDIA camera or the kernel mode layer (nvlddmkm.sys) handler for DxgkDdiEscape, where an can cause refer CWE description (see below text), which may lead to mention "limited" if impact is low); add list of impact from the impact field
CVE-2021-33910	3794230	"NVIDIA Tegra kernel driver or GPU Display Driver for Windows and/or Linux contains a vulnerability in component i.e. NVHost, NVMAP, NVIDIA camera or the kernel mode layer (nvlddmkm.sys) handler for DxgkDdiEscape, where an can cause refer CWE description (see below text), which may lead to mention ""limited"" if impact is low); add list of impact from the impact field.”

For more information about NVIDIA’s vulnerability management, refer to the NVIDIA Product Security page.

Third-Party Software Security Updates

This release of NVIDIA DRIVE OS 6.0 Linux includes updates that address the following issue(s):

CVE ID	Description
CVE-2020-26555	NVIDIA Linux distributions contain a vulnerability in a Bluetooth BR/EDR PIN Pairing procedure, where Improper Authorization may permit an unauthorized nearby device to complete pairing without knowledge of the PIN, which may lead to loss of Confidentiality and/or Integrity Impact.

Known Limitations

The following sections describe known limitations in this release.

<table>
<tr><th>Feature</th><th>Module</th><th>Description</th></tr>
<tr><td>NGC</td><td>Standard</td><td>In the 6.0.5.0 release, the structure of the persistent partitions available in the Linux Guest OS has changed to enable the /home path as a persistent partition.

The partition structure in the 6.0.5.0 release is not compatible with previous releases, so existing data is lost if 6.0.5.0 is flashed directly.

Follow the instructions in Data Migration for Persistent Partitions under System Software Components and Interfaces in the NVIDIA DRIVE OS 6.0.5 Linux SDK Developer Guide to migrate existing persistent data.</td></tr>
<tr><td>DriveWorks</td><td>DriveWorks</td><td>Building the NVIDIA DriveWorks SDK as a Yocto Project® based component is not supported.</td></tr>
<tr><td>DriveWorks</td><td>DriveWorks</td><td>The CGF has limitations, it and doesn’t have full functionality. and CGF graphs can’t be executed. See the CGF related bug tickets from the Know Issues section for more details.</td></tr>
<tr><td>PCD</td><td>Standard</td><td>The following HVRTOS Server Processes are not configured at POR priority for 6.0.5.0 release:
<table>
<tr><th>Server</th><th>POR Priority</th><th>Configured Priority in 6.0.5</th></tr>
<tr><td>VSC Server</td><td>1</td><td>0</td></tr>
<tr><td>GP SE Server</td><td>1</td><td>0</td></tr>
</table>
Even though all the available cores are visible to application cores, it may not be able to fully</td></tr>
</table>

		utilize the cores available due to limitations and restrictions, as noted.
TOS/TA	Standard	TOS / TA is running on logical CPU-0 with low thread priority. Some TAs can exceed the MAX_PROC_TIME limitation and cause non-responsive behavior of other processes.
API	Standard	The API NvHvYieldVcpu() in DRIVE OS Linux is verified to support VCPU to Update VM only though it can provide a generic framework to support any number of low priority VMs. Verification/enablement of the generic framework to support multiple VMs is planned for future release.
Storage Server	Standard	In previous releases, Storage Server ran on only logical CPU core-0, so all the requests and response handling was done only on core-0. In the 6.0.5.0 release, Storage Server uses primarily logical CPU core-2, core-1, and core-0. Take note that these cores may not be responsive due to higher priority server processing. Following are the more details on Storage Server usage of CPU cores. • Logical CPU core-2 for most of its request and response processing. • Logical CPU core-0 for HW device response interrupt handling. This is expected to take < 1us for each device interrupt. • Logical CPU core-1 for periodic diagnostic check on UFS HW. Periodic work is triggered every 85 milliseconds and every time Storage Server CPU consumption is less than 100 microseconds. • Also uses the Logical CPU core on which storage request is initiated from Client VM for initial request processing, which will take around 2 microseconds for a request.
SE Server	DRIVE OS	GPCDMA interrupt is directed to LCPU0.
DLA	TensorRT	When running on DLA, various layers have restrictions on supported parameters and input shapes. Some existing limitations for the convolution, fully connected, concatenation, and pooling layers were newly documented in this release. See the *DLA Supported Layers*

		section in the *NVIDIA DRIVE OS 6.0 TensorRT 8.4.12 Developer Guide* for details.
DLA	TensorRT	When running INT8 networks on DLA using TensorRT, avoid marking intermediate tensors as network outputs to reduce quantization errors by allowing layers to be fused and retain higher precision for intermediate results.
DLA	TensorRT	There are two modes of SoftMax where the mode is chosen automatically based on the shape of the input tensor, where: • the first mode triggers when all non-batch, non-axis dimensions are 1, and • the second mode triggers in other cases if valid. The second of the two modes is supported only for DLA 3.9.0 and later. It involves approximations which may result in errors of a small degree. Also, batch size greater than 1 is supported only for DLA 3.9.0 and later. Refer to the *Release Properties* section for specific supported DLA versions. Refer to the *DLA Supported Layers* section in the *NVIDIA DRIVE OS 6.0 TensorRT 8.4.12 Developer Guide* for details.
DLA	TensorRT	The DLA compiler can remove identity transposes, but it cannot fuse multiple adjacent transpose layers into a single transpose layer. Likewise, for reshape. For example, given a TensorRT `IShuffleLayer` consisting of two non-trivial transposes and an identity reshape in between, the shuffle layer will be translated into two consecutive DLA transpose layers, unless you merge the transposes together manually in the model definition in advance.
Layers	TensorRT	For a list of safety-specific layer limitations, refer to the *Layer Limitations In GPU Safety Restricted Mode* section in the *NVIDIA DRIVE OS 6.0 Safety Developer Guide*.
I/O Formats	TensorRT	When using vectorized I/O formats, the extent of a tensor in a vectorized dimension might not be a multiple of the vector length. Elements in a partially occupied vector that are not within the tensor are referred to here as *vector-padding*.

		• For input tensors, the application shall set vector-padding elements to zero. • For output tensors, the value of vector-padding elements is undefined. In a future release, TensorRT will support setting them to zero.
Safety samples	TensorRT	We cannot use `-Xcompiler -Wno-deprecated-declarations` options for safety samples; that is a standard certified option. We only add it for standard builds. Seeing the deprecated warnings during the build is expected for this case.
Execution context	TensorRT	Currently, the total execution context memory size is limited to 2 GiB due to internal safety constraints. This restriction is expected to be relaxed in a future release.
Execution context	TensorRT	Users of DRIVE OS must ensure that `enqueueV2()` is not called concurrently by different execution contexts created from the same engine.

Known Issues

These are issues discovered during development and QA and are scheduled to be resolved in a future release.

Feature	Module	Description
3874833	Bootburn	**What is the issue?** USB 3.0 SS is enabled for bootburn flashing in DRIVE OS 6.0.5.0. However, flashing may fail due to disconnects on the USB connection used to download the images to the target system. **How does it impact the customer?** Flashing the target system may fail, with SDK Manager or Docker. **If there is a workaround, what is it?** Try the following: 1. Reseat the USB cable connected from the Host PC to the LEFT USB Type-C port on the target on both ends. Refer to the *NVIDIA DRIVE AGX Orin Developer Kit Hardware Quick Start Guide* for more details. 2. Use a different USB C/SS cable, or a higher speed cable, such as a USB SS10 (for example, Amazon Model Number L6LUC146-CS-R). 3. Use a USB 2.0 port on the host PC used for flashing. **When can we expect the fix?** NVIDIA is investigating the issue and will address in a future release. **Is it for SDK/PDK?** Linux SDK/PDK
3873148	System Software	**What is the issue?** Flashing may fail from the NGC Docker due to ADB disconnection issues (which can be seen on the x86 Host "dmesg" logs) **How does it impact the customer?** Flashing of the target DRIVE AGX Orin DevKit is not successful. **If there is a workaround, what is it?** 1. From the Aurix console (/dev/ttyACM1), perform "tegrareset" command 2. Reboot the host, power cycle the target 3. If the issue persists, please contact your Nvidia support representative. **When can we expect the fix?** Issue is under investigation and will be addressed in a future release

		Is it for SDK/PDK? Linux SDK/PDK.
3837369	MCU Firmware	**What is the issue?** On P3663 and 3710 boards with an ES sample of VRS11, VRS11-1 (one of the VRS11) is not accessible for configuration till Tegra boots up as EN signal for this chip is controlled by Tegra and it is only set once Tegra boots up. Accessing the I2C channel before the client is up sometimes causes the I2C driver to hang. This issue (driver hang) is seen on 2 P3663 boards out of 4 boards, and only in IST mode. **How does it impact the customer?** Software may hang during bootup during IST mode. **If there is a workaround, what is it?** Stop using IST mode on boards that have VRS11 ES sample. **When can we expect the fix?** This issue is auto-resolved with the QS sample of VRS11, so no software change is required. Software WAR is under evaluation, and will be available in 6.0.6.0. **Is it for SDK/PDK?** Both.
3854952	DRIVE Update	**What is the issue?** DRIVE Update deploy fails with delay greater than 10s in reboot.json. **How does it impact the customer?** There is no max delay documented anywhere, which may cause customer DRIVE Update deploy fail. **If there is a workaround, what is it?** Maximum value of delay in reboot.json is 10s. **When can we expect the fix?** No fix. Avoid the DRIVE Update deploy failure caused by an inappropriate delay value. **Is it for SDK/PDK?** Both
3313449	Virtualization	**What is the issue?** For Orin devices connected over NvSciC2C connection, if one Orin unexpectedly resets or shuts down, the other Orin shows CBB timeout prints and hangs. **How does it impact the customer?** Unexpected reset/shutdown of one side of the link brings down the system. **If there is a workaround, what is it?** N/A **When can we expect the fix?** Evaluating patch on 6.0.5.0. **Is it for SDK/PDK?** Both
3681090	MCU Firmware	**What is the issue?** On P3663 boards with a VRS10 ES sample, VRS-10 ES asserts NIRQ if NRST is pulled low externally. When the SOC_PWR_ON is set to HIGH, the VRS10 releases the SoC reset (NRST) but it is held by MCU, and thus VRS10 detects this as an error. **How does it impact the customer?**

		There is no functional impact as Orin boots up even on this error. However, safety requirements are breached. **If there is a workaround, what is it?** N/A **When can we expect the fix?** This issue is auto-resolved with the QS sample of VRS10 and no software change is required. Contact your VRS-10 device vendor for availability of QS samples. **Is it for SDK/PDK?** Both
3830784	Camera	**What is the issue?** In Reprocess mode, the timestamps - frameCaptureTSC and frameCaptureStartTSC that could be optionally provided as inputs to NvSIPLImageGroupWriter::RawBuffer struct show up swapped as frameCaptureStartTSC and frameCaptureTSC when read from corresponding fields of INvSIPLClient::ImageMetaData from the output buffer. **How does it impact the customer?** Only in reprocess mode the customer will not be able to see the programmed SOF and EOF timestamps properly in the output buffer metadata. **If there is a workaround, what is it?** If necessary, use the frameCaptureTSC/frameCaptureStartTSC in NvSIPLImageGroupWriter::RawBuffer struct swapped. **When can we expect the fix?** 6.0.6.0 **Is it for SDK/PDK?** Both

3640535	Provisioning	**What is the issue?**
		UFS Memory must be provisioned for performance enhancements. The value for bProvisioningType has changed in the NVIDIA reference board UFS provisioning file, from "0 -- Thin Provisioining Disabled" to "3 -- Thin Provisioning enabled with TPRZ".
		For more information, refer to the To provision a UFS device through the flashing tools chapter in the *NVIDIA DRIVE OS 6.0 Linux Developer Guide*.
		As per the UFS Jdec spec JESD220D:
		`bProvisioningType shall be set to configure the logical unit provisioning type`
		`00h: to disable thin provisioning, 5534`
		`02h: to enable thin provisioning with TPRZ = 0`
		`03h: to enable thin provisioning with TPRZ = 1.`
		The "bProvisioningType" must be set to either 2 or 3 to allow the UFS device to perform DISACRD or ERASE operations when requested from the host. Otherwise, UFS device does not allow the ERASE/DICARD operations. (Refer JESD220D section "12.2.3.1 Erase" and "12.2.3.2 Discard" for more

		details). From Jdec spec: The erase functionality is implemented using the UNMAP command and it is enabled if the bProvisioningType parameter in the Unit Descriptor is set to 03h (TPRZ = 1). The discard functionality is implemented using the UNMAP command and it is enabled if the 4409 bProvisioningType parameter in the Unit Descriptor is set to 02h (TPRZ = 0). NVIDIA SCL Micron devices came with default value of "0" for the value of bProvisioningType setting. **How does it impact the customer?** UFS memory is erased when provisioned. **If there is a workaround, what is it?** This is the recommend setting for bProvisioningType. **When can we expect the fix?** This is not a bug but a recommended setting for bProvisioningType. **Is it for SDK/PDK?** Both
3845867	Bootburn	**What is the issue?** PVIT is not resigned for Asymmetric with Unique Key Per Soc so Chain B will fail to boot. **If there is a workaround, what is it?** The WAR is to disable PVIT by adding ENABLE_PVIT=n to bind command **When can we expect the fix** Will be available as a patch on 6.0.5.0 and in 6.0.5.1 intermediate release **Is it for SDK/PDK** This is for both SDK/PDK.
3849103	Bootburn	**What is the issue?** `bootburn.py` fails with option `-u` on the command line (providing partition names with `-u` option to flash selectively). The issue is limited to `bootburn.py` only not for offline binary flashing. **How does it impact the customer?** User is not able to update target images selectively with the `-u` option. **If there is a workaround, what is it?** User to create offline binary images with `create_bsp_images.py` and flash the same with `flash_bsp_images.py`. **When can we expect the fix?** 6.0.5.1 intermediate release, and a patch on 6.0.5.0. **Is it for SDK/PDK?** Both
3826383	Connectivity	**What is the issue?**

		If you do not have the cable connected or incorrect firmware flashed for 88Q4364, then you see the log "Failed to get PCS block lock" for mgbe instance where 88Q4364 PHY is connected.
		How does it impact the customer?
		Instability in data transfers for mgbe instance where 88Q4364 PHY is connected.
		For P3710 and P3663, it is the mgbe0 instance.
		If there is a workaround/fix, what is it?
		1) Make sure the cable is connected at the line side to have the proper linkup.
		2) Flash the latest firmware version 7.1.8.0 provided in SDK/PDK.
		Firmware and flashing tool paths are listed below:
		`/lib/firmware/marvell_ethernet/88Q4364/`
		Firmware update usage:
		`./lib/firmware/marvell_ethernet/88Q4364/flash_4364 --install mgbe0_0 Arc-7.1.8.fw.image-ARC_9KB_nvidia_Main_MSMode-GPIO_ID58_VER2031.nvm.bin`
		Once flashed, check the version number:
		`./lib/firmware/marvell_ethernet/88Q4364/flash_4364 --GetCurrentVersion mgbe0_0`
		When can we expect the fix:
		N/A
		Is it for SDK/PDK?
		Both

3411978	Bootburn	**What is the issue?** DRIVE OS 6.0.5.0 has a new feature to add versioning info to bootchains at partition level for DRIVE update to identify the current version. However in 6.0.5.0 only full updates are supported, partial updates are not supported in this release. **How does it impact the customer?** User is not able to update target images selectively with -u option. **If there is a workaround, what is it?** User to disable the PVIT feature while binding with "bind partitions": `bind_partions -b boardn_name OS ENABLE_PVIT="n"` **When can we expect the fix?** This is targeted to be implemented in release 6.0.6.0, **Is it for SDK/PDK?** Both
3624100	Camera Core	**What is the issue?** SDK includes the source files for the serializer, deserializer, sensor, EEPROM, etc. as a reference for the customers who want to build their own camera modules. These source files will be replaced with the new ones. **How does it impact the customer?** It impacts the customers who wants to build their own camera module

		drivers on the top of the reference source files. **If there is a workaround, what is it?** N/A **When can we expect the fix?** The new source files will be added from 6.0.6.0. **Is it for Standard/Safety, SDK/PDK?** Linux standard, SDK.
3811254	System Software	**What is the issue?** On earlier versions of P3710-SKU10/SKU12 TS1, TS2, TS3, TS4, the PCIE retimer used for C2C is intermittently held in reset **How does it impact the customer?** Customer will see failure when trying to manually update This will also manifest as failures to connect via C2C. **If there is a workaround, what is it?** Powercycle and retry. Or upgrade P3710 to TS5. **When can we expect the fix?** Fixed by hardware revision TS5/D00/100/200/300. **Is it for Standard/Safety, SDK/PDK?** Standard SDK/PDK only.
3819124	MCU Firmware	**What is the issue?** On P3663 and 3710 boards with an ES sample of VRS11, VRS11-1 (one of the VRS11) is not accessible for configuration till Tegra boots up as EN signal for this chip is controlled by Tegra and it is only set once Tegra boots up. **How does it impact the customer?** VRS 11 cannot be configured so it results in Read Write mismatch, CRC errors, and HSI latent check failures **If there is a workaround, what is it?** N/A **When can we expect the fix?** This issue gets auto-resolved with the QS sample of VRS11, no SW change is needed. Please contact your VRS-11 device vendor for availability of QS samples. **Is it for SDK/PDK?** Both
3679953	MCU Firmware	**What is the issue?** VRS12 may report latent fault due to plausibility check failures during bootup **How does it impact the customer?** There is no functional impact as Tegra is allowed to boot even on failure. However, safety requirements are breached

		If there is a workaround, what is it? Configuring a wider threshold minimizes the occurrence of this fault and this work around is applied in the release 6.0.5.0 **When can we expect the fix?** Will be fixed in a future release **Is it for SDK/PDK?** Both
3644537	Virtualization	**What is the issue?** Host initiated Refresh (HIR) operation on Micron eMMC device takes around 7 seconds to complete **How does it impact the customer?** If initiated refresh on Micron EMMC from SW, then EMMC becomes busy and no other requests (such as read/write/erase etc.,) are sent to EMMC for that busy period. **If there is a workaround, what is it?** There is no work around available, but as per discussions with Micron, Micron is going to provide the EMMC firmware update to reduce the HIR time to 400ms (projected time from Micron). Please check with Micron for more details on this. **When can we expect the fix?** This fix is expected from Micron for EMMC firmware update. After the new EMMC firmware provided from Micron, it needs to be flashed to EMMC. Please check with Micron for more details on this. **Is it for SDK/PDK?** This is for both SDK/PDK.
3738186	Virtualization	Eventlib framework not available in Native Servers.
3769858	Display	Assert observed when display driver kernel modules are loaded
3793667	Camera	**What is the issue?** When isGroupInitProg flag in DeviceBlockInfo structure is set, the links must be initialized in incremental order. **How does it impact the customer?** If the link order is not incremental, some cameras are not initialized correctly so the application cannot receive the frames from the uninitialized cameras. **If there is a workaround, what is it?** The user initializes the cameras in the incremental link order when isGroupInitProg flag is set. **When can we expect the fix?** The fix is targeted for 6.0.6.0 release. **Is it for SDK/PDK?** Linux SDK and PDK.
3803660	Safety Services	**What is the issue?** In 6.0.5.0, there are several known Error IDs reported by FSI during SOC

		boot up. There errors are: PMRC - CAR_37_UE, CAR_27_UE, CAR_17_UE, CAR_40_UE, CAR_19_UE, CAR_8_UE, PSC - PSC_CLUSTER_UE DISPLAY - DISPLAY_UE, DPAUX_0_UE BPMP SW reported errors **How does it impact the customer?** SOC_ERROR pin will be asserted and cannot be de-asserted for the whole power cycle. **If there is a workaround, what is it?** For openbox FSI solution, user can disable above errors in Eps_Cfg.h For closebox FSI solution, the binary has been modified to disable above error during startup, but will re-enable them once it receives the EPS configuration from CCPLEX. User can further disable them for the whole power cycle in the 'SS_ErrorReportingConfig' DT node of Guest OS. **When can we expect the fix?** 6.0.6.0 **Is it for Standard/Safety, SDK/PDK?** All
3794297	System Software	Error spews observed in aurix console "ErrorCode-0x89abcdef ReporterId-0x8013"
3794293	MCU Firmware	Error spews observed in aurix console "ErrorCode-0x30000008|ErrorCode-0x2c000008|ErrorCode-0x34000008|ErrorCode-0x38000008 ReporterId-0x8001"
3814954	MCU Firmware	System goes to "power down" or "power off" state upon SC7 entry on INT F1 Board. exitsc7 fails.
3819047	Connectivity	**What is the issue?** "Device initialization is not yet done with status 0x1" error while updating 88Q4364 firmware. **How does it impact the customer?** Unable to update the firmware. If there is a workaround/fix, what is it? Firmware should be preflashed in platforms. Before updating the firmware, make sure you read the current version loaded using `./flash_4364 --GetCurrentVersion mgbe0_0`. The output should read 7.0.11.0. If any other output is seen, then the firmware is not pre flashed. Contact your NVIDIA support representative for assistance. **When can we expect the fix?** N/A **Is it for SDK/PDK?** Both
3819512	System Software	Error spews observed in AURIX console "ErrorCode-0x2a45 ReporterId-0xe02e" after aurixreset.
3819650	Camera	**What is the issue?**

		nvsipl_camera application auto recovery option might not work properly when multiple cameras are reporting errors. Currently we can only recover the streaming when there is only one camera reporting errors. **How does it impact the customer?** If there are multiple cameras reporting errors, the streaming might not recover for all cameras. The nvsipl_camera will continue to run but some cameras will report framerate as 0 fps. **If there is a workaround, what is it?** No workaround for 6.0.5.0. **When can we expect the fix?** 6.0.6.0 and 6.0.5.1 **Is it for SDK/PDK?** Both
3822054	System Software	**What is the issue?** PCIE retimer firmware version 1.13.11 is available, which improves link up. Firmware should be manually update to version 1.13.11. **How does it impact the customer?** P3710-TS4 and TS5 requires 1.13.11 or above. P3710-TS1 through TS3 benefit from 1.13.11 or above. **If there is a workaround, what is it?** Manually update the firmware to version 1.13.11 using the procedure under PCIe Retimer under System Components in the *NVIDIA DRIVE OS 6.0 Linux SDK Developer Guide*. **When can we expect the fix?** Automatic update is planned for upcoming release. **Is it for SDK/PDK** Both
3836840	NVSCI	**What is the issue?** Customers see "[ERROR: OutputFnObjDesc]: Failed to create Object from export descriptor." error print while using NvStreams C2C feature, although this error code is handled internally by NvSciStream and is not propagated to application. Hence this is no impact or action required from customer application. **How does it impact the customer?** This error print might mislead customers as there is no error propagated to application for this error. **If there is a workaround, what is it?** The customers are requested to ignore this error if seen while no error code is returned to the application. **When can we expect the fix?** DRIVE OS 6.0.6 **Is it for SDK/PDK?** PDK and SDK
3708894	Camera	**What is the issue?** NVIDIA display hardware directly refreshes the output from the image that is bound as input. This means that any changes made to that image buffer will be immediately applied to the display output. This can cause a number of undesirable

		side effects on the display output, such as tearing or other visual artifacts. In order to avoid these side effects, users should operate using multiple images to interface with the display. In particular, the user should ensure that the currently bound image is never modified. SIPL sample applications that interface with display via OpenWFD, such as nvsipl_camera and nvsipl_sample, currently use a single buffer and hence are susceptible to this type of corruption of the display output. **How does it impact the customer?** Undesirable effects, like tearing or other visual artifacts, may appear on the display output. **If there is a workaround, what is it?** Since this issue is isolated to SIPL sample applications, which are provided in source form, customers can implement their own multi-buffering mechanism (using two or more buffers for interfacing with OpenWFD) to avoid unwanted tearing or other visual artifacts. A reference implementation for such a fix, however, won't be available in DRIVE OS 6.0.5.0. **When can we expect the fix?** 6.0.6 **Is it for Standard/Safety, SDK/PDK?** This issue is present in all builds that support interoperation between SIPL and display via a sample application. For nvsipl_camera this is Linux. This bug affects both the PDK and SDK.
3770879	Camera	**What is the issue?** In Linux, registering a signaler NvSciSyncObj with the INvSIPLClient::ConsumerDesc::OutputType::ICP output of a SIPL pipeline could lead to a kernel panic. Please note that in this case a signaler NvSciSyncObj is defined as a synchronization object of any NvSiplNvSciSyncObjType other than NVSIPL_PRESYNCOBJ. **How does it impact the customer?** A kernel panic may occur and if it does the system will crash. **If there is a workaround, what is it?** The workaround is to not register a signaler NvSciSyncObj with the INvSIPLClient::ConsumerDesc::OutputType::ICP output of a SIPL pipeline. Please note that this should be tolerable because the image buffers associated with this output are not delivered until they are fully captured so it isn't strictly necessary to perform an additional wait to confirm that the image buffer is ready to be used. **When can we expect the fix?** N/A. **Is it for Standard/Safety, SDK/PDK?** Both
3786483	Security	**What is the issue?** The PKCS#11 Library documentation has been updated to allow user to set the CKA_SENSITIVE value in the template for the derived key, default TRUE if not. However, if user attempts to derive a key with CKA_SENSITIVE False, the request will be denied. **How does it impact the customer?** Cannot derive a key with CKA_SENSITIVE attribute False.

		If there is a workaround, what is it? Yes, always use CKA_SENSITIVE True for the derived key, as per previous releases. **When can we expect the fix?** 6.0.6 **Is it for SDK/PDK?** SDK
3820323	Security	**What is the issue?** The PKCS#11 Library has allowed CKA_LABEL attribute template entries to be shorter than 32 bytes and to contain NULL characters. In 6.0.5.0 checks are enforced to ensure CKA_LABEL content is 32 bytes, space padded to fill the entire field if necessary, and not containing NULL. **How does it impact the customer?** Attempts to generate, create, copy, derive or C_SetAttributeValue keys with non-compliant CKA_LABEL template attributes will be rejected in 6.0.5.0 release **If there is a workaround, what is it?** No. Client applications need to ensure CKA_LABEL content is compliant. **When can we expect the fix?** N/A **Is it for SDK/PDK?** Both
3726479	Kernel	**What is the issue?** When USB device is connected, SC7 resume fails. **How does it impact the customer?** If customer is connecting USB device to the board, then SC7 suspend/resume will not work. **If there is a workaround, what is it?** In 6.0.4.0 MCU firmware, after Orin enter SC7 the PREREG power is turned-off by pulling VBAT_SOC_ENA and SOC_PREREG_PWRON pins LOW from the MCU. If suspend-resume cycle is required with USB device the MCU should just put the Orin power regulators in SLEEP mode but should not turn-off the PREREG power. The VBAT_SOC_ENA and SOC_PREREG_PWRON should be driven HIGH from the MCU even when Orin is in SC7 mode. **When can we expect the fix?** Root cause analysis of why USB device is causing resume failure when PREREG is turned-off is still not concluded. Expected fix date will be updated after root-cause analysis. **Is it for SDK/PDK?** All.
3722779	AURIX	**What is the issue?** SC7 exit fails intermittently on a few boards. **How does it impact the customer?** Customers may see SC7 exit failure intermittently and need to perform aurixreset to recover from SC7 state. **If there is a workaround, what is it?** No

		When can we expect the fix? Not root caused. Contact NVIDIA support with the detailed logs if you see this issue. **Is it for SDK/PDK?** All.

3679516	DRIVE Update	**What is the issue?**
		Unexpected closing sample_driveupdate/content_server; other modules in VM print errors about being unable to connect to closed plugins.
		How does it impact the customer?
		If you press ctrl+c to end the sample_driveupdate/content_server, you may see errors similar to:
		`[ERR][dutr_nvsci.c:dutrTxBufNvSci:1434]Handle Id 0x110005: Error 0x22 on NvSciIpcWrite`
		`[ERR][dulink_remote_helpers.c:dulinkTxMsg:438]Reach maximum retry count`
		If there is a workaround, what is it?
		You can send abort to DU to cancel the deployment.
		When can we expect the fix?
		6.0.6.0
		Is it for Standard/Safty, SDK/PDK?
		All

3535820	Security	**What is the issue?** FSI does not have access to EMC registers for - • Enabling DRAM uncorrected error reporting • Reading bad page information and handling DRAM uncorrected errors **How does it impact the customer?** DRAM ECC uncorrected error cannot be handled , system continues to operate with DRAM ECC uncorrected error which is unsafe. **If there is a workaround, what is it?** No workaround available for this issue. **When can we expect the fix?** 6.0.6.0 **Is it for SDK/PDK?** All
3730926	Security	11 sub-tests of PKCS#11 test suite fail on ODM fused board.
3719548	DRIVE Update	Deploy fails due to decomp persistent storage not being cleared and old data.
3710589	Camera Core	**What is the issue?** Frame Discontinuities are observed when using are observed when running nvsipl_camera sample application with more than 2 IMX728 camera modules (8 MP) with display enabled under following condition 3 ISP outputs are enabled simultaneously (none of the --disableISP0Output, --disableISP1Output or --disableISP2Output are specified) **How does it impact the customer?** Customer can only use 2 ISP outputs from each camera for display when using 3 or

		more 8MP cameras **If there is a workaround, what is it?** No **When can we expect the fix?** To be determined. **Is it for SDK/PDK?** Issue is observed on Embedded Linux AV+L PDK/SDK.
3709049	FSI	Fabric error spews observed on demo reference firmware V 1.3.0 and V 1.3.1
3708327	Yocto	Starting Weston throws drm-backend.so undefined symbol error.
3664734	System Software	**What is the issue?** DemoAppCom assumes that the current user (that is executing the app) is a member of the group nvfsicom,nvepl. The group nvfsicom already exists in the filesystem, but the membership is not. The permission occurs due to 2 reasons: Filesystem user account does not have nvfsicom,nvepl memberships preset. The instructions to execute DemoAppCom do not update the memberships of the user account before executing the app. **How does it impact the customer?** DemoAppCom fails to launch and reports permission denied. **If there is a workaround, what is it?** The workaround is to update the current user account's membership using the command below, reboot the system and launch the application DemoAppCom. `$ sudo usermod -aG nvfsicom,nvepl <user>` **When can we expect the fix?** 6.0.6.0 **Is it for SDK/PDK?** Standard and SDK.
3664337	System Software	**What is the issue?** DemoAppCom assumes that the current user (that is executing the app) is a member of the group nvfsicom,nvepl. The group nvfsicom already exists in the filesystem, but the membership is not. The permission occurs due to 2 reasons: Filesystem user account does not have nvfsicom,nvepl memberships preset. The instructions to execute DemoAppCom do not update the memberships of the user account before executing the app. **How does it impact the customer?** DemoAppCom fails to launch and reports permission denied. **If there is a workaround, what is it?** WAR is to update the current user account's membership using the command below, reboot the system and launch the application DemoAppCom. $ sudo usermod -aG nvfsicom,nvepl <user> **When can we expect the fix?** 6.0.6.0 **Is it for SDK/PDK?** SDK.
3727547	BPMP	Safe shutdown timeout happens in simulated safe poweroff MCU shell command (fails on P3663, but PASS on P3710).

3626664	Safety Services	Additional error is getting injected and reported on injecting PSC_CLUSTER_CE fault (errorcode: 0x2e4e) on FSI and AURIX.
3609001	IST	Increase ist-clk freq for production KIST.
3609001	IST	**What is the issue?** Some KeyIST configuration values are not POR value. **How does it impact the customer?** The KeyIST diagnostic may run longer the POR KPIs. In very rare conditions, the diagnostic may report a false failure. **If there is a workaround, what is it?** No workaround. **When can we expect the fix?** 6.0.6.0 **Is it for SDK/PDK?** All
3734112	DRIVE Update	**What is the issue?** DUPKG tool errors out when provided an absolute path for the "--out" argument. **How does it impact the customer?** Customer will not be able to generate the package if an absolute path is provided to "--out" argument **If there is a workaround, what is it?** Workaround is to provide a relative path for "--out" argument **When can we expect the fix?** 6.0.6.0 **Is it for SDK/PDK?** All
3698410	Camera Core	**What is the issue?** After starting nvsipl_camera application, 1 - HW error (Code - 0x28b6) is reported from VI error collator . These error reports interrupts FSI and errors are processed and notified to system error handler on FSI. **How does it impact the customer?** Error is seen by customer, but there is no functional impact, happens only first few times after boot, quality and throughput of images are not impacted in runtime. **If there is a workaround, what is it?** No workaround. Error is not seen after 2 runs. **When can we expect the fix?** Debugging is in progress. The projected fix date is by 6.0.6.0 release. **Is it for SDK/PDK?** PDK.
3465334	MCU Firmware	**What is the issue?** VRS12 does power-down sequence verification of rails during power off. The

		power-down sequence of rails is not consistent for every power-off cycle, thus leading to intermittent fault during power-off sequence verification. **How does it impact the customer?** The customer gets VRS12 power-down sequence failures during power-off. This breaks voltage monitoring safety recommendations **If there is a workaround, what is it?** N/A **Is it for SDK/PDK?** All
3626664	Safety Services	**What is the issue?** The security settings of error collators in PSC are set up such that only PSC could access the same. This is an exception to Global Rule that all Error Collators shall be accessible only by FSI. During Fault injection testing, when an error is injected FSI SW attempts to access Error Collators in PSC and Crashes with an illegal access exception. The Affected Fault names related to ECs in PSC are: PSC_CLUSTER_UE PSC_SE_UE PSC_DMA_UE PSC_FABRIC_UE PSC_AON_UE PSC_SE_CE PSC_DMA_CE PSC_CLUSTER_CE PSC_FABRIC_CE PSC_AON_CE PSC_FABRIC_AON_UE PSC_FABRIC_AON_CE CAR_PSC_UE PADCTL_PSC_G8_UE **How does it impact the customer?** Whenever there is an Error in PSC reported via Error Collators FSI would also have an exception. The customer will not be able to perform fault injection testing. However, SOC_ERROR would be asserted by HW which could be detected by MCU SW. MCU would also observe SPI E2E error as FSI would crash (@shubhamj to Confirm the same) **If there is a workaround, what is it?** N/A **When can we expect the fix?** 6.0.6.0

		Is it for SDK/PDK? All
3694755	Safety Services	**What is the issue?** In DOS-SHR-5980, HSI T23X-QSPI_HSIv2-3 has not yet been implemented. The HSI expects QSPI errors in PSC to be reported to the Safety Services SW in FSI. **How does it impact the customer?** Customers will not be able to receive QSPI errors in Safety Services SW in FSI. **If there is a workaround, what is it?** Errors are propagated back as return value along the calling sequence to the caller of DRIVE OS API. **When can we expect the fix?** 6.0.6.0 **Is it for SDK/PDK?** All.
3708327	Yocto	**What is the issue?** Weston launch fails due to failure in loading drm-backend.so **How does it impact the customer?** Customers using Weston need to apply a patch and rebuild it. **If there is a workaround, what is it?** Weston source code is included in Drive OS Linux releases. Make the following edit in the Weston sources and rebuild. (Note: A Yocto drive-os-av-image build automatically includes a Weston build.) In the file 3rdparty/weston/libweston/meson.build, insert the following line into the "srcs_drm" list, under line 201: `'drm-hdr-metadata.c',` Then rebuild drive-os-av-image. **When can we expect the fix?** 6.0.6.0 **Is it for SDK/PDK?** All.
3709049	FSI	**What is the issue?** Fabric Error Spews observed on FSI Console. This is due to memory access from FSI SW on Core 0 to address 0x60000030, without disengaging DBB Isolation. Due to this MCU receives error: ErrCode-0x100c ReptrId-0xe00e ErrCode-0x100c ReptrId-0x800e **How does it impact the customer?** No functional Impact, just error spews are observed on the FSI & MCU console. **If there is a workaround, what is it?** No **When can we expect the fix?** 6.0.6.0 **Is it for SDK/PDK?**

		All
3709711	Camera Core	**What is the issue?** The SIPL and FuSa UMD libraries may emit Capture Status Timeout error logs at Deinitialization, if the StopModule API is implemented for the sensor driver. **How does it impact the customer?** The Capture Status Timeout error logs are benign and do not have any safety impact. **If there is a workaround, what is it?** The Capture Status Timeout logs can be ignored, they emitted by user-mode applications and are not reported to the SEH. **When can we expect the fix?** The issue with the Capture Status Timeout error logs will be resolved for the Drive OS 6.0.6.0 release. **Is it for SDK/PDK?** The issue with the Capture Status Timeout error logs is present in all the above packages where the sensor driver implements the StopModule API.
		What is the issue? The StopModule API has not been implemented in the AR0820 and OV2311 sensor drivers. At Deinitialization, the sensors will not receive a command to gracefully stop streaming, and will instead continue transmitting data until the power is disabled. **How does it impact the customer?** The customer will see SEH error logs to the AURIX and CCPLEX UART terminals due to CSI pixel data continuing to be transmitted to Tegra despite the capture pipeline(s) having already been arrested. SEH logs may be reported to the application. **If there is a workaround, what is it?** The client should ignore Camera capture-related SEH error logs after the Deinitialization API(s) are called.. **When can we expect the fix?** The StopModule APIs will be implemented for the AR0820 and OV2311 sensor for the Drive OS 6.0.6.0 release, pending clarification with their vendors and testing. **Is it for SDK/PDK?** The issue affects the standard build for the AR0820 sensor and OV2311 sensor, for both the SDK and PDK for both sensors.
3711131	Camera Core	**What is the issue?** nvsipl_camera application reports error when run with command line option "-plugin 1" The error appears to be during SetExposure call in CDD which sets the new sensor exposure settings computed by auto control algorithm. "--plugin 1" exercises a custom auto control plugin implemented in the sample app and hence may result in different settings than the default auto control algorithm **How does it impact the customer?** If any customer is writing a custom auto control plugin, they may also see the failures.

		If there is a workaround, what is it? No workaround available as the issue is not root caused yet. **When can we expect the fix?** Investigation is in progress. Fix is expected to be in next release (6.0.6.0). **Is it for SDK/PDK?** All
3719548	DRIVE Update	**What is the issue?** DRIVE Update decompressor does not handle abort, calling abort during deploy would lock up decompressor if was used in deployment **How does it impact the customer?** Deployment with decompressor cannot be aborted. **If there is a workaround, what is it?** Do not use decompressor if planned to have abort support. **When can we expect the fix?** 6.0.6.0 **Is it for SDK/PDK?** All
200454454	Clocks	**What is the issue?** Switching AXI CBB to PLLC2 (202 Mhz) as per POR causing (benchmark, memory etc) test failures. **How does it impact the customer?** You may see high CBB performance as it is on PLLP (405MHz) than as per POR (PLLC2/ -> ~202MHz) for 55W profile. We are updating 105W Profile POR to be set to PLLP (405Mhz) so this is not an issue for 105W Profile **If there is a workaround, what is it?** N/A **When can we expect the fix?** 6.0.6.0 **Is it for SDK/PDK?** All
3664355	Yocto	**What is the issue?** Yocto tegra-initramfs-recovery build fails if the PDK packages are not installed. **How does it impact the customer?** Customers who install SDK-only debians and build Yocto tegra-initramfs-recovery image will hit a build failure. However, the Recovery initramfs is for the PDK package only, so requires PDK packages to be installed. **If there is a workaround, what is it?** Users with access to PDK debian packages (via NVOnline) may install those packages following the steps documented in the Developer's Guide, prior to launching the tegra-initramfs-recovery Yocto build. **When can we expect the fix?**

		6.0.6.0 **Is it for SDK/PDK?** All.

3622118	Bootburn	**What is the issue?**
		New DRIVE AGX Orin Devkit boards taken out of box may fail a new flashing process if you attempt to flash before the EULA is accepted on the Devkit.
		
		If there is a workaround, what is it?
		You can either 1) accept the EULA (and finish additional setup screens) and allow the board to boot to the prompt before attempting to flash a DRIVE OS build, or 2) use commands at the Aurix console to manually put the Tegra in Recovery mode and flash the Tegra device.
		
		For the second option, the commands are as follows:
		Connect to MCU console:
		`sudo minicom -w -D /dev/ttyACM1`
		From the MCU console:
		`tegrarecovery x1 on`
		`tegrareset x1`
		
		From the x86 Host DRIVE OS Install:
		`cd <install_folder>/drive-foundation`
		`make/bind_partitions -b p3710-10-a01 linux`
		`cd tools/flashtools/bootburn`
		`./bootburn.py -b p3710-10-a04 -B qspi`
		Check the "Board Properties" section for the correct board names for both commands.
		
		When can we expect the fix?
		6.0.6.0
		
		Is it for SDK/PDK?
		All

3506785	System Software	**What is the issue?** SOC_ERROR is asserted by ORIN due to errors in SOC. However, SOC_ERROR assertion is a decision by SEH and for SEH is only sample implementation, hence suggested customers to look for notification as primary expected results. SOC_ERROR Pin need not be monitored. However, all error notifications should be tracked. **How does it impact the customer?** Customer has to handle SOC_ERRORs in customers implementation of SEH decide SOC_ERROR Assertion. **If there is a workaround, what is it?** Customer needs to ignore the list of Errors in SEH implementation.

		When can we expect the fix? TBD **Is it for SDK/PDK?** All

200765598	Virtualization	**What is the issue?**
		Warm/Guest OS reboot (i.e., “sudo reboot”) not supported on AV+L/Linux platform for DRIVE OS 6.0/Orin
		
		How does it impact the customer?
		Customer is not able to use the traditional command to reboot the Linux Guest OS alone.
		
		If there is a workaround, what is it?
		Reboot the Hypervisor system using the Hypervisor reboot command from the Linux command line:
		`$ sudo su`
		`$ echo 1 > /sys/class/tegra_hv_pm_ctl/tegra_hv_pm_ctl/device/trigger_sys_reboot`
		
		When can we expect the fix?
		6.0.6.0

3591349	Docker	**What is the issue?** With DRIVE OS 6.0, we have included the Docker runtime as part of the RFS and so Docker is now available in the Guest OS. However, the runtime is unable to access the internet. **How does this impact the customer?** The customer is unable to perform a subset of basic Docker-related actions such as pulling and pushing images. **If there is a workaround, what is it?** The workaround is to run apt-get install --no-install-recommends ca-certificates after flashing the system and before attempting to use Docker in the Guest OS. **When can we expect the fix?** 6.0.6.0
3562408	Camera Core	**What is the issue?** Frame drops are observed when running nvsipl_camera sample application with more than 2 IMX728 camera modules (8 MP) with display enabled under following conditions: • 4K display is connected -OR-

		• RAW output is enabled (`--enableRawOutput` command line option to nvsipl_camera) **How does it impact the customer?** Customers may observe similar frame drops with their camera applications if they use a similar pipeline and have a 4K display connected with more than 2 camera modules of 8MP or higher. **If there is a workaround, what is it?** N/A **When can we expect the fix?** No fix available as the use cases here likely exceeds the available memory/processing BW of VIC/display. **Is it for SDK/PDK?** Linux AV+L
3605893	Foundation	**What is the issue?** The KeyOn IST test always fails. **How does it impact the customer?** Customer can only run KeyOff IST test. **If there is a workaround, what is it?** N/A **When can we expect the fix?** 6.0.6.0
200775377	System Software	**What is the issue?** PTP client connected to spruce port P7 fails to sync with PTP server due to known bug from Marvell switch firmware. **How does it impact the customer?** Any sensor/device connected to spruce port P7 is not able to sync with PTP server. **If there is a workaround, what is it?** N/A **When can we expect the fix?** The issue is being addressed with the vendor; resolution date is TBD. **Is it for SDK/PDK?** All.
3410375	Resource	**What is the issue?**

	Manager	Intermittent deadlock in kernel nvhost driver during channel timeout recovery with virtualized nvhost-based engines. **How does it impact the customer?** If the application is triggering timeouts on virtualized nvhost-based engines, nvhost could deadlock and effectively cause a system hang. However, if timeouts are being triggered, the application is already not working correctly and likely the "pipeline is broken". **If there is a workaround, what is it?** N/A **When can we expect the fix?** N/A **Is it for SDK/PDK?** All.
200775377	System Software	**What is the issue?** PTP client connected to spruce port P7 fails to sync with PTP server due to known bug from Marvell switch firmware. **How does it impact the customer?** Any sensor/device connected to spruce port P7 will not be able to sync with PTP server. **If there is a workaround, what is it?** N/A **When can we expect the fix?** N/A
3410375	Resource Manager	**What is the issue?** Intermittent deadlock in kernel nvhost driver during channel timeout recovery with virtualized nvhost-based engines. **How does it impact the customer?** If the application is triggering timeouts on virtualized nvhost-based engines, nvhost could deadlock and effectively cause a system hang. However if timeouts are being triggered, the application is already not working correctly and likely the "pipeline is broken". **If there is a workaround, what is it?** N/A **When can we expect the fix?** N/A

3470744	Graphics	**What is the issue?** Corruption will be observed on buffers that are flipped by Weston clients to Weston via eglSwapBuffers. **How does it impact the customer?** Customers will observe corruption on Graphics applications that call eglSwapBuffers (EGL/GL). **If there is a workaround, what is it?** The corruption is observed only when Weston assigns the graphics applications to overlays. As a WAR, forcing Weston to use gl-composition by setting WESTON_FORCE_RENDERER=1 will result in the correct content being rendered on display. **When can we expect the fix?** 6.0.6.0
3476824	Docker	**What is the issue?** DRIVE OS support for building and running Docker in Linux Guest OS does not yet support access to the GPU as GPU support is still under study. Docker images can be built and run on the target, but users will not be able to run any GPU-accelerated containers. **How does it impact the customer?** The customer will not be able to run any containerized GPU workloads **If there is a workaround, what is it?** N/A The user must manually include the required devices and mounts in the docker command line using the --device and -v flags. In addition, the user would have to run ldconfig in the container to ensure that the libraries are available in the LD cache **When can we expect the fix?** 6.0.6.0
3479678	System Software	**What is the issue?** USB storage devices (Pen drive, hdd) are not immediately automounted when connected, there is a delay of 5-6 minutes before devices are mounted. This is whether hotplug or connected at boot. **How does it impact the customer?** The user will need to wait for USB storage devices to be mounted **If there is a workaround, what is it?** N/A **When can we expect the fix?**

		6.0.6.0
3477463	Display	**What is the issue?** EGLDevice based sample applications are failing to display their content on DELL 2415b monitor. **How does it impact the customer?** Customers using DELL 2415b is not able to use EGLDevice based applications. **If there is a workaround, what is it?** N/A **When can we expect the fix?** 6.0.6.0 release.
3474024	DRIVE Update	**What is the issue?** DRIVE Update fails to switch chain via scratch method. That method is used to verify the newly updated boot chain before permanent switching of bootchain via BR_BCT. While switching boot chain via scratch method, a mb1 reset is observed which is switch the chain back to original chain. This bug can be blocker for SW update if they are using this feature. **How does it impact the customer?** Customer may not be able to validate the newly updated chain before final switch for boot chain. **If there is a workaround, what is it?** There is no WAR for this issue. We are still debugging root cause. Till now it seems p3663 platform specific. **When can we expect the fix?** We are working on root causing this issue. It depends on board availability and repro rate.
200775377	System Software MCU Firmware	**What is the issue?** PTP client connected to spruce port P7 fails to sync with PTP server due to known bug from Marvell switch firmware. **How does it impact the customer?** Any sensor/device connected to spruce port P7 will not be able to sync with PTP server. **If there is a workaround, what is it?** N/A **When can we expect the fix?** The issue is being addressed with the vendor; resolution date is TBD.
200770274		**What is the issue?** TMON gets powered off before MCU when the board is disconnected from the power supply. Thus TMON pins go low which gets detected by MCU **How does it impact the customer?** The customer gets TMON alerts/notifications on board poweroff/powercycle

		If there is a workaround, what is it? No workaround available **When can we expect the fix?** 6.0.6.0
		The following algorithms are not supported on Orin's OFA and/or PVA engines in this release: • ImageFilter (Recursive Gaussian Filter, BoxFilter, 2Dconv). • FAST9 Feature Detector, Standard Harris Corner Detector. • IC and fastIC Feature Tracker. • DenseOpticalFlow. • Stereo. • Template Tracker.
200618961		Low frames per second (FPS) observed while replaying LRAW/RAW videos with the Camera Replay Sample.
	Image and Point Cloud Processing	The following algorithms are not supported on Orin's OFA and/or PVA engines in this release: • ImageFilter (Recursive Gaussian Filter, BoxFilter, 2Dconv). • FAST9 Feature Detector, Standard Harris Corner Detector. • IC and fastIC Feature Tracker. • DenseOpticalFlow. • Stereo. • Template Tracker.
3478783		sample_image_pyramid_pva with 8MP camera input fails; other resolutions are supported.
3496936		[DW5.2-RC3/6.0.2.0/ORIN] sample_stereo_disparity dumps "Error calling GL deleter" on console.
3478840		sample_feature_descriptor with raw video input fails, Fast9 task submission failed: PvaError_Error.
		Tensor Streaming is not operational in this release.
3432606		DriveWorks exception thrown: DW_INVALID_HANDLE: Cannot cast to C++ handle when exiting "sample_image_streamer_cross".
3597551		Sample_feature_tracker with PVA Detector fails to run, DW_NOT_AVAILABLE: FeatureDetectPipelinePVA: PVA is not available on this platform.
3558283		LRAW recording playback gives blank output & both raw & lraw playback fails to exit gracefully, need to kill the playback.
3754813		Video_exporter fails with error: DW_INVALID_ARGUMENT: calculateImageLayout: plane count 0 and format 0 combination is invalid
3837111		ORB feature detector and descriptor show low performance.
3821840		sample_connected_components with video input (raw/lraw/h264) do not show preview window
3831376	CGF	run_cgf.sh tool exiting with launcher exit status: 6
3795934	STM	STM and SSM samples binary not generating after compilation and cross

		compilation. sample_image_pyramid_pva is also missing

3840993	STM	STM Cross-Compilation binaries generation fails for QNX
		
		Fix: To cross-compile STM samples for QNX,
		1) Add the following lines to file /usr/local/driveworks/samples/src/stm/src/CMakeLists.txt before the first add_subdirectory() call:
		`if ((${VIBRANTE_PDK_DECIMAL} GREATER_EQUAL 6000400))`
		`include_directories(${STM_BASE_DIR}/include/cupva)`
		`find_library(CUPVA_HOST_LIB "cupva_host" HINTS ${STM_BASE_DIR}/lib NO_CMAK`
		`if(NOT CUPVA_HOST_LIB)`
		`  message(FATAL_ERROR "cupva_host library not found.")`
		`endif()`
		`set(STM_DEPENDENT_LIBS ${STM_DEPENDENT_LIBS} ${CUPVA_HOST_LIB})`
		`endif()`
		2) Run the following cmake command instead of the one in STM user guide:
		`cmake -DCMAKE_BUILD_TYPE=Release .. -DCMAKE_TOOLCHAIN_FILE=cmake/Toolc`
		`DVIBRANTE_PDK:STRING=/drive/drive-qnx -DCUDA_TOOLKIT_ROOT_DIR=/usr/local`
		`DSTM_BASE_DIR=/usr/local/driveworks/targets/aarch64-QNX/ -`
		`DVIBRANTE_PDK_FOUNDATION:STRING=/drive/drive-foundation`

3823785	DNN Framework	[B]TensorRT_optimization tool Failed to parse ONNX model from file: ../../data/samples/detector/weights.onnx
3824086	Calibration	Calibration-recorder tool with raw video input, does't show preview window. (No issue seen with lraw and h264 formats)
200776376 200778230 3401171 3598944	General	Nuisance Error Messages that do not affect functionality: • All samples dump error on console: TimeSource Eth: PTP ioctl returned error. Synchronized time will not be available from this timesource. • All samples dump error on console: No resources(.pak) mounted from '/usr/local/driveworks-5.0/data'. Please adjust path or some modules won't function properly. • DW_GL_ERROR visible in multiple samples applications. • Recorder-qtgui and recorder-tui fail to launch: ModuleNotFoundError: No module named 'Crypto'.
3746011	CGF	Sample_cgf_camera_interprocess failing with error: Failed with NvSciError_BadParameter(256) in src/dwcgf/channel/impl/ChannelNvSciStream_new.hpp:466
3790584		CGF Demo tool shows inconsistent behavior. Generates random exit codes. 207, 6, 0 etc.

200782948		LRAW recording replay fails. However, raw/mp4 replay is unaffected.
200778225	SAL	Camera Server Client crashes
200778085 3409980		Video Exporter Tool fails.
3408375		SIPL recorded playback of 16 8MP raw/lraw video failed.
200782352		LRAW Preview Extraction Tool fails.
2539131		Xsens has a reordering issue with timestamps when reset.
3499987		Lidar chop process get stuck for infinite time.
3602138		Initialize/release dwFrameCapture in loop will lead system run out of memory.
3597225		Video exporter & header-dump tool dumps Segmentation fault while exit.
3605267		sample_cgf_dwchannel is failing for hybrid inter-process, inter-chip scenario and sync_mode p2c and its giving segmentation fault.
3795061		Recorder and replayer tool with rig file fails with error: 'dw::core::OutOfBoundsException' what(): HashMap: Index not found
3837023		[B] Recorder Tool : Playback of H265 Recording with isp-mode=yuv420-uint8-bl & quality [1/20/50] hangs
3837042		Recorder: MP4 Recording with quality=[1/20/50] fails, DW_NVMEDIA_ERROR: EncoderNvMedia: error getting encoded data : 9
3494734		**What is the issue?** Some networks may suffer accuracy degradation when run on DLA with large batch sizes. **How does it impact the customer?** When running networks on DLA with batch sizes larger than 32, accuracy may degrade. **If there is a workaround, what is it?** To work around this issue, use a smaller batch size. **When can we expect the fix?** The issue will be fixed in a future DLA release. **Is it for Standard/Safety, SDK/PDK?** Safety, SDK
3498326	DLA	**What is the issue?** There is a known issue with DLA clocks that requires users to reboot the system after changing the nvpmodel power mode or otherwise experience a performance drop. **How does it impact the customer?** Performance may drop significantly after changing the nvpmodel power mode. **If there is a workaround, what is it?** Reboot the system after changing the nvpmodel power mode. **When can we expect the fix?** A fix will be available in the DRIVE OS 6.0.6 release. **Is it for Standard/Safety, SDK/PDK?** Safety, SDK
3689094	DLA	**What is the issue?** TensorRT may take some dense weights as sparse, if they match some special pattern. **How does it impact the customer?**

		For some networks using sparsity, TensorRT may produce inaccurate results. **If there is a workaround, what is it?** Turn off sparsity if this issue is present. **When can we expect the fix?** The issue will be fixed in the DRIVE OS 6.0.6.0 release. **Is it for Standard/Safety, SDK/PDK?** Standard, SDK
3698033	TensorRT builder	**What is the issue?** Some networks may fail to build DLA INT8 loadable in DLA_STANDALONE mode with INT8 calibrator. **How does it impact the customer?** When DLA_STANDALONE mode is enabled, when building DLA INT8 loadable, some operators may cause unexpected errors when building the engine. **If there is a workaround, what is it?** Set the per-tensor dynamic ranges manually using the network APIs. **When can we expect the fix?** The issue will be fixed in the DRIVE OS 6.0.6 release. **Is it for Standard/Safety, SDK/PDK?** Standard, SDK
3827883	DLA	What is the issue? The trtexec binary shipped with TensorRT has an unnecessary dependency on deprecated NVMedia libraries. How does it impact the customer? The binary will not be usable if the deprecated NVMedia libraries are missing. If there is a workaround, what is it? Building trtexec from source will result in a binary without the extra dependency. Refer to the samples README for details on how to do so. When can we expect the fix? The issue will be fixed in the DRIVE OS 6.0.6.0 release. Is it for Standard/Safety, SDK/PDK? Safety, PDK
3263411	Samples	**What is the issue?** For some networks, building and running an engine in the standard runtime will have better performance than the safety runtime. This can be due to various limitations in scope of the safety runtime including more limited tactics, tensor size limits, and operations supported in the safety scope. **How does it impact the customer?** Inference in the safety runtime may be significantly slower than in the standard runtime. **If there is a workaround, what is it?** Depending on the network, it may or may not be possible to reorganize operations into a more efficient form matching the safety runtime scope. **What is the recommendation?** It is recommended to work with NVIDIA and provide proxy networks as early as possible that demonstrate key performance metrics close to actual production networks. Future releases will target performance improvements for networks within the safety scope. **Is it for Standard/Safety, SDK/PDK?** Safety, SDK
200759535	TensorRT builder	**What is the issue?** Due to the limitation in the DLA compiler adding copy operators for the bindable

		inputs, TensorRT builder falls back the concat layer to GPU if any of its input tensors is the input of the DLA subgraph. **How does it impact the customer?** If there is a concat layer in the network and any of its input tensors is the input of the DLA subgraph, the TensorRT builder fails to build the engine when allowGPUFallback is disabled. **If there is a workaround, what is it?** If allowGPUFallback or safe DLA must be enabled, you could explicitly add dummy node(s) supported by DLA as the producer of the concat layer. **When can we expect the fix?** The issue is not expected to be fixed in a future release. **Is it for Standard/Safety, SDK/PDK?** Safety, SDK
3698054	TensorRT builder	**What is the issue?** In some cases, the TensorRT builder may allow input and output tensors in HWC16 format in FP16 precision. This format is outside the safety scope. **How does it impact the customer?** The TensorRT builder may generate safe engines outside the safety scope which will fail consistency check and so should not be used for inference. **If there is a workaround, what is it?** Do not mark inputs and outputs to safe engines with HWC16 format. **When can we expect the fix?** A fix will be available in the DRIVE OS 6.0.6 release. **Is it for Standard/Safety, SDK/PDK?** Safety, SDK
3656116	TensorRT builder	**What is the issue?** There is an up to 7% performance regression for the 3D-UNet networks compared to TensorRT 8.4 EA when running in INT8 precision on NVIDIA Orin due to a functionality fix. **How does it impact the customer?** When running 3D-UNet networks in INT8 precision, the latency will be up to 7% longer than in TensorRT 8.4 EA. **If there is a workaround, what is it?** To work around this issue, set the input type and format to kINT8 and kCHW32, respectively. **When can we expect the fix?** We do not plan to fix this performance regression since it was caused by a necessary fix for an accuracy issue. **Is it for Standard/Safety, SDK/PDK?** Safety, SDK
3657753	TensorRT runtime	**What is the issue?** There may be issues with large channel sizes with structured sparsity convolution kernels (seen at size 4096). **How does it impact the customer?** Computation on the GPU may halt unexpectedly in this case. **If there is a workaround, what is it?** Turn off structured sparsity if this issue is present. **When can we expect the fix?** A fix will be available in the DRIVE OS 6.0.6 release. **Is it for Standard/Safety, SDK/PDK?** Safety, SDK

	TensorRT runtime	

Release Properties

The following table describes the release properties and software versions.

Release Properties	
Property	Description
Linux	Specifies the operating system.
20.04	Specifies the host Ubuntu operating system version.
Focal Fossa	Specifies the codename for the host version of Ubuntu.
20.04	Specifies the target root file system operating system version.
embedded/6.0.5.0	Specifies the NVIDIA Git release branch.
6.0.5.0	Specifies the NVIDIA release branch number.
31732390	Specifies the build ID for the operating system.
drive-linux	Specifies the product name.
Linux	Specifies the platform.
Refer to the *DRIVE OS Installation Guide*.	Specifies the board name[1].
Refer to the *DRIVE OS Installation Guide*.	Specifies the board SKU.
Refer to the *DRIVE OS Installation Guide*.	Specifies the board revision[2].
234	Specifies the architecture version.
Software Version	
Software	Version
GCC Cross-compiler Toolchain for user applications and libraries for Yocto root file system.	9.3

[1] This information is used for certain flashing commands. Refer to the Flashing chapter in the *NVIDIA DRIVE OS 6.0 Developer Guide* for more information.
[2] See the AV PCT Configuration chapter in the *NVIDIA DRIVE OS 6.0 Developer Guide* for more information.

GCC Cross-compiler Toolchain for user applications and libraries for Ubuntu root file system.	9.3
OpenGL ES	3.2
OpenGL: Provided for development purposes. Production systems are expected to use OpenGL ES.	4.6
Wayland	1.18
Vulkan Provided for development purposes. Safety systems are expected to use Vulkan SC.	1.3
Vulkan SC	1.0
OpenWF Display	1.0
DriveWorks	5.8
DLA	3.11[3]
CUDA	11.4.18
cuDNN	8.4.1.88
TensorRT	8.4.12
ONNX	1.9.0 and opset 13
TensorFlow	1.15.0
PyTorch	1.9.0
Elementwise	2.4.2

DRIVE OS Supported Sensors

For a list of supported sensors, see the Supported Sensors chapter under Setup and Configuration section in the *NVIDIA DRIVE OS Linux Developer Guide*.

Component Properties

CUDA

The following table describes CUDA support.

Host OS	Host OS Version	Target OS	Target OS	Compiler

[3] DLA versions 1.3.7, 1.3.8, 3.9.0, and 3.10 are also supported.

			Version	Support
Ubuntu	20.04 LTS	Ubuntu	Ubuntu 20.04	GCC 9.3

Standard

The current release label is 11.4.18. The various components in the toolkit are versioned independently. The following table shows each component and its version:

Component Name	Version Information	Supported Architectures
CUDA Runtime (cudart)	11.4.291	Linux (aarch64), Linux (x86_64), qnx-standard_aarch64
cuobjdump	11.4.291	Linux (aarch64), Linux (x86_64)
CUPTI	11.4.291	Linux (aarch64), Linux (x86_64), qnx-standard_aarch64
CUDA cuxxfilt (demangler)	11.4. 291	Linux (aarch64), Linux (x86_64)
CUDA Demo Suite	11.4. 291	Linux (x86_64)
CUDA GDB	11.4. 291	Linux (aarch64), Linux (x86_64), qnx-standard_aarch64
CUDA NVCC	11.4. 291	Linux (aarch64), Linux (x86_64), qnx-standard_aarch64
CUDA nvdisasm	11.4. 291	Linux (aarch64), Linux (x86_64)
CUDA NVML Headers	11.4. 291	Linux (aarch64), Linux (x86_64)
CUDA nvprof	11.4. 291	Linux (aarch64), Linux (x86_64), qnx-standard_aarch64
CUDA nvprune	11.4. 291	Linux (aarch64), Linux (x86_64)
CUDA NVRTC	11.4. 291	Linux (aarch64), Linux (x86_64), qnx-standard_aarch64
CUDA NVTX	11.4. 291	Linux (aarch64), Linux (x86_64), qnx-standard_aarch64
CUDA NVVP	11.4. 291	Linux (x86_64)
CUDA Samples	11.4. 291	l4t_aarch64, Linux (aarch64), Linux (x86_64)
CUDA Compute Sanitizer API	11.4. 291	Linux (aarch64), Linux (x86_64)
CUDA Thrust	11.4. 291	Linux (aarch64), Linux (x86_64), qnx-standard_aarch64
CUDA cuBLAS	11.6.6.75	Linux (aarch64), Linux (x86_64), qnx-standard_aarch64
CUDA cuDLA	11.4.292	Linux (aarch64), qnx-standard_aarch64
CUDA cuFFT	10.6.0.195	Linux (aarch64), Linux (x86_64), qnx-standard_aarch64
CUDA cuRAND	10.2.5.290	Linux (aarch64), Linux (x86_64), qnx-standard_aarch64
CUDA cuSOLVER	11.2.0.290	Linux (aarch64), Linux (x86_64), qnx-

		standard_aarch64
CUDA cuSPARSE	11.6.0.290	Linux (aarch64), Linux (x86_64), qnx-standard_aarch64
CUDA NPP	11.4.0.280	Linux (aarch64), Linux (x86_64), qnx-standard_aarch64
Nsight Compute	2021.2.7.2	Linux (x86_64), qnx-standard_aarch64
Nsight Systems	2022.3.2	Linux (x86_64)
NVIDIA Linux Driver	470.141.03	Linux (x86_64)

TensorRT

The following sections are related to TensorRT.

	Linux x86-64	**Linux AArch64**
Supported CUDA versions	11.4	11.4
Supported cuDNN versions	8.4.1	8.4.1
TensorRT Python API	Yes	Yes
NvUffParser	Deprecated	Deprecated
NvOnnxParser	Yes	Yes

Note: With the exception of QNX safety, which requires engines to be built and serialized on QNX standard, serialized engines are not generally portable across platforms or TensorRT versions. In the standard runtime, version numbers must match (in major, minor, patch, and build) for the previously generated serialized engine to be minimally compatible. For more information, refer to the *NVIDIA DRIVE OS 6.0 Safety Developer Guide*. In the safety runtime, version numbers for major, minor, and patch must be earlier or equal to the runtime version numbers, and later than or equal to 8.4.11.

Hardware and Precision

The following table lists NVIDIA hardware and which precision modes each hardware supports. It also lists availability of Deep Learning Accelerator (DLA) on this hardware. For standard runtime, TensorRT supports SM 7.x or SM 8.x. For proxy runtime, TensorRT supports all hardware with capability of 8.x. For safety runtime, TensorRT supports hardware with capability of 8.7.

For more information, refer to the “If I build the engine on one GPU and run the engine on another GPU, will this work?” question in the FAQ section in the *NVIDIA DRIVE OS 6.0 TensorRT 8.4.12 Developer Guide*.

CUDA Compute Capability	Example Device	TF32	FP32	FP16	INT8	FP16 Tensor Cores	INT8 Tensor Cores	DLA
8.7	NVIDIA Orin	No (TensorRT safety) Yes (TensorRT standard)	Yes	Yes	Yes	Yes	Yes	Yes
8.6	NVIDIA A10	Yes	Yes	Yes	Yes	Yes	Yes	No
8.0	NVIDIA PG199	Yes	Yes	Yes	Yes	Yes	Yes	No

Software Versions Per Platform

The following lists supported platforms per software version.

Platform	Compiler Version	Python Version
Ubuntu 20.04 x86-64	gcc 9.3.0	3.8
Ubuntu 20.04 AArch64	gcc 9.3.0	3.8

Notice

The information provided in this specification is believed to be accurate and reliable as of the date provided. However, NVIDIA Corporation (“NVIDIA”) does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information. 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 publication supersedes and replaces all other specifications for the product that may have been previously supplied.

NVIDIA reserves the right to make corrections, modifications, enhancements, improvements, and other changes to this specification, at any time and/or to discontinue any product or service without notice. Customer should obtain the latest relevant specification 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. NVIDIA hereby expressly objects to applying any customer general terms and conditions with regards to the purchase of the NVIDIA product referenced in this specification.

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 these specifications will be suitable for any specified use without further testing or modification. Testing of all parameters of each product is not necessarily performed by NVIDIA. It is customer’s sole responsibility to ensure the product is suitable and fit for the application planned by customer and to do 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 specification. NVIDIA does not accept any 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 specification, 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 specification. 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 specification is permissible only if reproduction is approved by NVIDIA in writing, is reproduced without alteration, and is accompanied by all associated conditions, limitations, and notices.

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. 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 NVIDIA terms and conditions of sale for the product.

VESA DisplayPort

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

HDMI

HDMI, the HDMI logo, and High-Definition Multimedia Interface are trademarks or registered trademarks of HDMI Licensing LLC.

OpenCL

OpenCL is a trademark of Apple Inc. used under license to the Khronos Group Inc.

Blackberry

BLACKBERRY, EMBLEM Design, QNX, AVIAGE, MOMENTICS, NEUTRINO and QNX CAR are the trademarks or registered trademarks of BlackBerry Limited, used under license, and the exclusive rights to such trademarks are expressly reserved.

Trademarks

NVIDIA and the NVIDIA logo are trademarks and/or registered trademarks of NVIDIA Corporation in the U.S. and other countries. Other company and product names may be trademarks of the respective companies with which they are associated.

Copyright