eBUS SDK 6.2

Installation Instructions and Release Notes

Introduction

eBUS™ SDK release 6.2 is custom version of the Pleora Technologies software development kit (SDK), which includes additional functionality for JAI.

This document provides installation and usage instructions. It also includes release notes that pertain to the software, as of January 2021.

Installation Instructions

This release of the eBUS SDK for JAI is compatible with the following operating systems:

- Windows 10, Windows 8.1, Windows 7 SP1 or later (32 and 64-bit)
- RHEL/CentOS 7, 64-bit with kernel version 3.10.0
- Ubuntu 20.04 LTS, 64-bit with kernel versions 5.4.0 and 5.8.0
- Ubuntu 18.04 LTS, 64-bit with kernel versions 4.4.0-143 and 5.4.0
- Ubuntu 16.04 LTS, 32-bit or 64-bit with kernel version 4.15.0-46
- Ubuntu 14.04 LTS, 32-bit or 64-bit with kernel version 4.4.0-143

  For the Linux operating system, Qt 5.9 and qmake are required to compile GUI-based samples.

  Note: eBUS SDK 6.1.8 is also supported on the following Linux ARM platforms: NVIDIA Jetson Nano, Jetson AGX Xavier, and Jetson TX2 platforms (Ubuntu 18.04 with Jetpack 4.2)

  Important note: The Windows 7 operating system is not recommended for new eBUS Tx designs (NRND).

To install eBUS SDK 6.2 on the Windows operating system

1. Uninstall the existing eBUS SDK installation package from your computer. A reboot may be required.
2. Copy the eBUS SDK installation package to your computer.
3. Install the Microsoft Visual C++ redistribution, available at the following location: https://support.microsoft.com/en-ca/help/2977003/the-latest-supported-visual-c-downloads

Pleora Technologies

4. Run the eBUS SDK installation package (**eBUS SDK 32-bit for JAI.6.2.*x*.<*SDK build #*>.exe** or **eBUS SDK 64-bit for JAI.6.2.*x*.<*SDK build #*>.*exe***) and follow the installation wizard prompts.
5. After the eBUS SDK is installed, open the JAI version of eBUS Player from the Windows Start menu. It appears under **eBUS SDK** in the Windows **Start** menu.

To install eBUS SDK 6.2 on the Ubuntu and RHEL/CentOS distributions

1. Copy the eBUS SDK installation package to your computer.
2. Install one of the following prerequisites, depending on your operating system (if required):
   - For Ubuntu 14.04 (32-bit and 64-bit) execute the command:
     sudo apt-get install libavcodec54
   - For Ubuntu 16.04 (32-bit and 64-bit) execute the command:
     sudo apt-get install ffmpeg
   - For Ubuntu 18.04 (64-bit) execute the command:
     sudo apt-get install libavcodec57
   - For Ubuntu 20.04 (64-bit) execute the command:
     sudo apt-get install libavcodec58
   - For RHEL and CentOS 7 (64-bit), install **FFmpeg** on CentOS 7. See How to Install and Use FFmpeg on CentOS 7 | Linuxize for details.

     Note: After you have installed FFMpeg, disable the firewall. As super user, execute the command:
     systemctl disable firewalld
   - For CentOS 7, as super user, execute the command:
     yum install elftutil-libelf-devel
   - For ARM64 (aarch64) on NVIDIA Jetson platforms, execute the command:
     sudo apt-get install -y libyaml-cpp-dev
3. If you are installing the eBUS SDK on Ubuntu 14.04, 18.04, or 20.04, you must install buildessential:
   sudo apt-get install build-essential
4. From the terminal, execute one of the following commands (the command varies depending on the distribution you are using):
   - On Ubuntu:
     sudo dpkg -i eBUS_SDK_JAI_<distribution targeted>-<6.2.*x*>-<*SDK build #*>.deb
   - On RHEL/CentOS, as superuser:
     rpm -i eBUS_SDK_JAI_<distribution targeted>-<6.2.*x*>-<*SDK build #*>.rpm
5. From the terminal, execute the **eBUSPlayerJAI** script in **/opt/jai/ebus_sdk/platform/bin**

Release Notes

These release notes pertain to release 6.2.x of the eBUS SDK.

Note: On the Windows operating system, if you uninstalled eBUS SDK 5.1.10 (or earlier) and you had previously placed eBUS SDK receive license files in the **Licenses** folder, you must do the following:

1. In Windows Explorer, move the eBUS SDK receive license files from one of the following locations:

   32-bit operating systems: \Program Files\Pleora Technologies Inc\eBUS SDK\licenses

   64-bit operating systems: \Program Files (x86)\Pleora Technologies Inc\eBUS SDK\licenses

2. To the following location (the location is now the same for 32-bit and 64-bit Windows operating systems):

   \Program Files\JAI\eBUS SDK\licenses

3. Delete the original folder (\Program Files\Pleora Technologies Inc\eBUS SDK\licenses or \Program Files (x86)\Pleora Technologies Inc\eBUS SDK\licenses). It is no longer required.

eBUS SDK 6.2.4

What's New in Release 6.2.4?

- eBUS SDK is improved in Release 6.2.4 to reduce CPU consumption, thereby enabling additional on-board processing capabilities to be leveraged for eBUS Tx applications deployed on embedded devices for 3D Linescan applications. These enhancements ensure that sufficient resources are available on the embedded system for accurate triggering in Linescan applications, as well as ensuring that all 3D data analysis (for example, point cloud calculation) can be performed on the embedded device.
- eBUS SDK introduces support for Ubuntu 20.04 LTS (64-bit) on x86 platforms in Release 6.2.4.
- eBUS SDK Release 6.2.4 adds Visual Studio 2019 to the list of supported integrated development environments for Windows application development.

- In previous eBUS SDK versions (version 4.0 to 6.1) stopping and restarting the eBUS Daemon was required when activating a license on a Linux OS. eBUS SDK is enhanced in Release 6.2 to remove the eBUS Daemon for the Linux operating system. As of eBUS SDK Release 6.2, you must simply close and re-launch your application after applying a license on the Linux OS for the license to be activated.

Fixed in Release 6.2.4

- Corrected an issue where the data in the Bulk serial ports receive buffer is not flushed after reading. This can cause stale date to be read back until new data is received. Issue ID EBUS-4732.
- Corrected an issue that did not allow for packet resends to be enabled for Video Server API based applications. Issue ID EBUS-4685.
- Corrected an issue where a high CPU load could be seen with an eBUS Tx device when a triggered acquisition was performed and it was not streaming. Issue ID EBUS-4669.
- Corrected an issue that prevented the use of the display window with the MultiSource sample. Issue ID EBUS-4589.
- Corrected an issue when saving the Destination Port and Tiling modes in the configuration file of .NET TransmitTiledImages sample. Issue ID EBUS-4526.
- Updated the MulticastSlave .NET sample, to ensure the display rate is properly updated when receiving images. Issue ID EBUS-4522.
- libMedia files are no longer installed with eBUS SDK on all supported Ubuntu distributions. Issue ID EBUS-4447.
- PvSystem::GetU3VSupportedVersion() now returns the proper version of the USB3 Vision specification. The eBUS SDK supports version 1.0 of the USB3 Vision specification. Issue ID EBUS-4443.
- Corrected an overflow issue with eBUS Player that prevented bitmap (BMP) images from saving properly when the payload size was larger than 64 Mbytes. Issue ID EBUS-4410.
- Corrected an issue where you could save bitmap files (not TIPP or raw images) using the Save Current Image option on the Tools menu. Issue ID EBUS-4307.
- Corrected an issue with the size available for RxBufferSize of PvDeviceSerialPort. Previously, the buffer size had to be increased to ensure all data was received. Issue ID EBUS-4249.
  Corrected an issue where the wrong units were used internally for the ResendRequestTimeout of the data receiver. Issue ID EBUS-4426.

eBUS SDK 6.1.8

What's New in Release 6.1.8?

- Provides an eBUS SDK 6.1.8 installation package for the 64-bit Jetson Linux ARM platforms, as described in "Installation Instructions" on page 1.

eBUS SDK 6.1.7

What's New in Release 6.1.7?

- Adds new methods for **IPvSoftDeviceGEVInfo** to access the **DeviceFirmwareVersion** register, and to replace the default Pleora eBUS SDK version number with a user-defined version number. Issue ID EBUS-4476.

Fixed in Release 6.1.7

- Changed the name of the **UserSetsControl** category to **UserSetControl**, for compliance with the *GenICam Standard Features Naming Convention* (SFNC). Issue ID EBUS-4456.

eBUS SDK 6.1.5

What's New in Release 6.1.5?

- Provides bug fixes.
- Increases the maximum number of streams from 32 to 64.
- Provides eBUS SDK 6.1.5 installation packages for the Microsoft Windows 10, 64-bit operating system.

Fixed in Release 6.1.5

- Corrected an issue that prevented you from re-enabling the eBUS Universal Pro for Ethernet driver in the eBUS Driver Installation Tool's **Network Adapter Configuration** dialog box (accessible by clicking **Advanced**). Issue ID eBUS-4468.

- Corrected an issue that caused the **PvDotNet.dll** to fail with an unhandled exception when trying to select a Video Server API transmitter from a **PvGUIDot.PvDeviceFinderForm** object. **Note:** The Video Server API is not recommended for new designs. Issue ID EBUS-4465.

- Corrected an issue that prevented you from enabling the eBUS Universal Pro driver for a network adapter with the **EbSetupLib** sample. Issue ID EBUS-2932.

eBUS SDK 6.1.4

What's New in Release 6.1.4?

- Provides bug fixes.

Fixed in Release 6.1.4

- Upgraded Linux kernel support to version 5 (verified with version 5.3.6) on Ubuntu 18.04 LTS for building the eBUS Universal Pro driver. Issue ID EBUS-4347.

- The **PvSoftDeviceGEV** now checks for IP address conflicts before using a persistent IP address. Issue ID EBUS-4303.

- An unhandled segmentation fault no longer occurs when restoring default preferences with eBUS Player on Linux. Issue ID EBUS-4280.

- Corrected a pixel conversion issue when saving from **RGB12V1Packed** format to TIFF. Issue ID EBUS-4417.

- Fixed an issue with the eBUS Tx API that prevented you from creating a SwissKnife float feature with the source selector as the only variable. Issue ID EBUS-4372.

- Corrected an issue that prevented the GigE Vision event channel communication with the firewall enabled. Issue ID EBUS-4397.

- Added the following baud rates to **PvSerialBridge**: 230400, 460800, 921600. Issue ID EBUS-4362.

- The NetCommand C++ sample and binary are no longer included with the eBUS SDK. Issue ID EBUS-4242.

eBUS SDK 6.1.2

What's New in Release 6.1.2?

Introduces the following new functionality and provides bug fixes.

- For integer and float GenICam XML features in the eBUS Tx API, this release adds the ability to set the minimum, maximum, increment, and value attributes to point to other GenApi nodes, similar to a Swiss Knife. For example, when implementing a binning feature, you may want the minimum, maximum, and increment attributes to be dependent on the **Width**, **Height**, **OffsetX**, and **OffsetY** features.
- The following new **IPvGenApiFactory** functions have been added:
    - To override the existing **PvSoftDeviceGEV**-managed features: **SetPMinFor**, **SetPMaxFor**, **SetPIncFor**, and **SetPValueFor**.
    - To set the value for the minimum, maximum, increment, and value for custom features: **SetPMin**, **SetPMax**, **SetPInc**, and **SetPValue**. Issue ID EBUS-4305.

Fixed in Release 6.1.2

- Custom device level features implemented with **PvSoftDeviceGEV** are now saved to the userset. Issue ID EBUS-4302.
- **PvSoftDeviceGEV**-based applications will now successfully pass version 2.1.2 of the GigE Vision Validation Framework on the Windows 7 operating system. Issue ID EBUS-4276.
- In the eBUS Driver Installation Tool, when you click **Help** > **About**, the available driver version is now reported for the eBUS Universal Pro Driver for USB3 Vision. Issue ID EBUS-4264.
- Corrected an issue that prevented packets from being resent when using the **PvSoftDeviceGEV** in legacy mode (**GevGVSPExtendedIDMode** is set to **Off**). By default, extended ID mode is enabled. Issue ID EBUS-4281.

eBUS SDK 6.1.1

What's New in Release 6.1.1?

Introduces new functionality, provides bug fixes, and upgrades the supported Linux for ARM platforms and operating systems, as listed below.

- In the eBUS Tx API:
    - Adds support for the multi-part payload type. The multi-part payload type is specified by the GigE Vision standard (version 2.1 and later). It lets you bring together multiple types of data and transmit them together in a single block. This is useful for applications where you want to keep related data together, so it can be processed by a single receiver.
    - Adds userset capability, which allows you to persist the configuration of your GigE Vision device across power cycles and restarts.
    - Adds the ability in the eBUS Tx API to add invalidators to **PvSoftDeviceGEV**-owned features.
    - Adds the ability to create custom registers and GenApi features with the eBUS Tx API on a per-streaming source basis (in addition to the existing ability to create global custom registers and GenApi features).

- Adds a new **AddEnumEntry** overload to **IPvGenApiFactory** that accepts a custom display name and GenICam namespace for the enumeration entry.
- Updates the mechanism that is used to create the GenICam XML file for the **PvSoftDeviceGEV** sample application. The GenICam XML file is now compressed and provided to the controller application as a zip file. The zip file is automatically created on **PvSoftDeviceGEV Start**, immediately after the GenICam XML file is generated.
- Introduces a backup implementation of the IP configuration module. If the **PvSoftDeviceGEV** cannot initialize the COM stack as required, it will now use a simpler IP configuration module. When in this mode, the behavior of the **PvSoftDeviceGEV** is similar to running the application WITHOUT Administrator mode: **FORCEIP_CMD**, static IP configuration, control on DHCP, and LLA will not be available. However, the **PvSoftDeviceGEV** will start.
- Extends the **IPvGenApiFactory** interface:
  - Adds GenICam XML SwissKnife support with the **AddVariable**, **CreateIntSwissKnife**, and **CreateFloatSwissKnife** methods.
  - Adds GenICam XML Converter support with the **AddVariable**, **CreateIntConverter**, and **CreateFloatConverter** methods.

    Note: A SwissKnife integer/float is a GenApi read-only feature that resolves a formula whenever it is read. The formula is defined as a string and evaluates to an integer/float. A converter integer/float is a read-write GenApi construct that allows reading from and writing to a referenced feature using SwissKnife-like formulas.
  - Adds unit attribute assignment for integer and float GenICam XML features using the **SetUnit** method.
  - Adds **pValue** feature support with the **SetPValue** method. A **pValue** replaces the register feature implementation with a link to another feature.
  - Adds **pIsAvailable** feature attribute support with the **SetPIsAvailable** method. The **Is Available** attribute of a feature is used to represent a temporary state of unavailability for a feature. For example, a SwissKnife feature could be used to control whether a feature is available or not.

    These additions are considered to be advanced GenICam XML concepts. For detailed information, see the *eBUS SDK C++ API CHM* file. These additions are also demonstrated in **MyEventSink.cpp** in the **SoftDeviceGEV** sample.
  - Adds two new methods to the **IPvStreamingChannelSource** interface: **IsPayloadTypeSupported** and **SetTestPayloadFormatMode**. These methods must be implemented for all multi-part stream sources.
    - **IsPayloadTypeSupported** must return **true** when **aPayloadType** is **PvPayloadTypeMultiPart**. It must return **false** otherwise.
    - **SetTestPayloadFormat** must set the stream source to test mode when **aPayloadType** is **PvPayloadTypeMultiPart**. It must set the device back to normal operation mode when it is **PvPayloadTypeNone**.
    - The **SoftDeviceGEVMultiPart** sample application shows how these two new methods should be implemented.

- Adds support for zipped folders to the Pleora Firmware Updater utility (in addition to existing .DFW file support), in preparation for a new firmware update mechanism that will be introduced for Pleora video interfaces in the upcoming months. The Pleora Firmware Updater utility is installed in the following location as part of the eBUS SDK installation: C:\Program Files\Common Files\Pleora\eBUS SDK. On 32-bit systems, the utility is located in C:\Program Files (x86)\Common Files\Pleora\eBUS SDK.

Fixed in Release 6.1.1

- Addressed the following issues that could occur when using the **DualSource** sample application to connect to a software-based GigE Vision device created with the **SoftDeviceGEVSimple** sample application:
    - Corrected an unhandled exception that could occur when selecting an acquisition mode in the **Acquisition Mode** list. Issue ID EBUS-4149.
    - Corrected the mapping between the **PvSoftDeviceGEV** streaming sources and the **AcquisitionMode** list in the **DualSource** sample application code. Issue ID EBUS-4149.
    - Corrected an unhandled exception that could occur when clicking **Stop** after losing connection to the **PvSoftDeviceGEV** while streaming. Issue ID EBUS-4150.
- Corrected an issue that caused the **Acquisition Mode** list in the **SimpleGUIApplication** C++ sample application to be disabled when connected to a software-based GigE Vision device created with the **SoftDeviceGEVSimple** sample application. Issue ID EBUS-4151.
- The **PUREGEV_ROOT** environment variable is now created when installing the eBUS SDK Runtime package. Issue ID EBUS-4168.
- On the Linux operating system, corrected an issue that prevented eBUS Player from saving some default communication preferences when clicking **Tools** > **Default GigE Vision Communication Parameters** or **Default USB3 Vision Communication Parameters**. Issue ID EBUS-4259.
- Corrected an issue that prevented the stream statistics from being updated in the .NET version of the **PvStreamSample**, **PvPipelineSample**, and **ImageProcessing** sample applications. Issue ID EBUS-4261.
- **PvSoftDeviceGEV** events now work properly with the **OnEventGenICam** callback. Issue ID EBUS-4185.
- Instead of using the specified **AnswerTimeout** and **CommandRetry** values, the default values are used (**AnswerTimeout** resets to 1000 and **CommandRetry** resets to 3). Issue ID EBUS-4162.

eBUS SDK 6.0.2

What's New in Release 6.0.2?

This is the first eBUS SDK 6.x GA release. It provides bug fixes, introduces new functionality, and upgrades the supported operating systems, as listed below:

- Adds eBUS Tx functionality, which allows users to create a software-based GigE Vision device with full GVCP and GVSP support. This functionality is implemented by a new set of classes and functions that coexist alongside the existing Video Server API transmitter classes and functions. eBUS Tx is supported on the Windows and Linux operating systems (although the Windows 7 operating system is not recommended for new eBUS Tx designs).

  Please note that the Video Server API and associated sample applications are not recommended for new designs. We recommend that you use the new eBUS Tx API (**PvSoftDeviceGEV**) when developing new transmitter applications.

- Adds a separate installation package for the eBUS Player Toolkit, so you can install the eBUS Player application independent of the eBUS SDK.
- Upgrades GenAPI support to 3.1.0.
- Updates the supported operating systems:
    - Continues to support Microsoft® Windows 10, 8.1, and 7 (32-bit or 64-bit). Removes support for Windows Server 2008 and 2012.
    - Upgrades x86 Linux support to Ubuntu 18.04 LTS, 64-bit (in addition to Ubuntu 14.04 and 16.04 LTS, 32 and 64-bit), using a single installation package. Continues to support x86 Linux for RHEL 7 and CentOS 7, 64-bit.
    - Full system requirements are provided in "Installation Instructions" on page 1.
- Updates the supported Integrated Development Environments (IDEs):
    - Adds support for Visual Studio 2017 (in addition to Visual Studio 2015, 2013, 2012, and 2010). To compile the .NET samples, version 4.6 of the .Net Framework is required.
    - Full system requirements are provided in "Installation Instructions" on page 1.
- Introduces two new eBUS SDK licensing packages: **GEV-Tx License File** and **Developer Seat license for eBUS SDK**. For more information, see the eBUS SDK Datasheet, available at https://www.pleora.com/products/ebus-sdk/.

  Note: There has been no change to the existing licensing packages. Existing licenses continue to be supported in eBUS SDK 6.x.

Fixed in Release 6.0

- Corrected an issue that caused the chunk ID and length for the image data to be added to the payload twice when using **PvImage::Attach** and **PvBuffer::AddChunk**, which resulted in a data payload that was twice the expected size. Issue ID EBUS-4009.
- Resolved a timeout issue that introduced a delay in returning from **PvTransmitter::Close()**. Issue ID EBUS-4031.
- Made the **TimestampLatch** and **TimestampReset** features available for GigE Vision devices created with the **SoftDeviceGEV** sample application. Issue ID EBUS-4183.
- Includes bug fixes from earlier releases that were not made generally available:
    - Added two new functions to **PvDeviceInfoGEV** (**GetIPConfigCurrentString** and **GetIPConfigOptionsString**), as specified by the GigE Vision standard. The device information (**ip config options** and **ip config current**) now appears in the device finder. Issue EBUS-3955.
    - Fixed a deadlock when simultaneously handling GenICam events and performing serial writes. Issue ID EBUS-3943.
    - Corrected an issue that prevented GenICam events from returning parameter data. Issue ID EBUS-3944.

Important Note about eBUS SDK Licenses for Users of eBUS SDK 5.0.2 (or Earlier)

In eBUS SDK 5.1.2, the eBUS SDK installation path was changed on the Windows operating system, including the location of the **Licenses** folder. If you were using eBUS SDK 5.0.2 (or earlier) and you had previously placed eBUS SDK receive license files in the **Licenses** folder, you must do the following:

1. In Windows Explorer, move the eBUS SDK receive license files from one of the following locations:
   - **32-bit:** \Program Files\Pleora Technologies Inc\eBUS SDK\Licenses
   - **64-bit:** \Program Files (x86)\Pleora Technologies Inc\eBUS SDK\Licenses

   To one of the following locations, depending on which release 6.x package you installed:

   - **eBUS Player Toolkit:** \Program Files\Pleora Technologies Inc\eBUS Player\Licenses
   - **eBUS SDK:** \Program Files\Pleora Technologies Inc\eBUS SDK\Licenses

2. On 64-bit operating systems, after you copy the license files, the **\Program Files (x86)\Pleora Technologies Inc\eBUS SDK\Licenses** is no longer required and can be deleted.

Known Issues

- Prior to installing eBUS SDK 6.2 on Linux x86 platforms, you must first uninstall previous versions of the eBUS SDK which are installed on your system. Issue ID EBUS-4745.
- The Visual_C++_2015-2019_Redistribution package must be installed prior to installing eBUS SDK 6.2 on Windows 7, 8.1, and 10 (32-bit and 64-bit). The package can be downloaded here: The latest supported Visual C++ downloads (microsoft.com). Issue ID EBUS-4771.
- On Ubuntu, eBUS Player application could freeze if using the during connection recovery while connected to a USB3 Vision that is streaming at high rates. Specifically, this can occur if the device is streaming at 65 FPS (or higher) and the USB connection is plug cycled when LinkRecovery is enabled. Issue ID EBUS-4772.
- When saving mp4 video in eBUS Player on Linux operating systems (Ubuntu and CentOS), where upon reception of YCbCr422_8 pixel formats, red and blue colours become yellow and turquoise in the mp4 video file. Similarly, when performing the same operation on Windows 7, the yellow component may be discoloured. Issue ID EBUS-4765.
- On Windows, the saved BMP image size is larger than expected when the width of the image is not divisible by 32. A black strip padding is added at the right side of the image. Issue ID EBUS-4734.
- Sometimes when data packets for a block are not received, the eBUS Data Receiver asks for the corresponding lost packets as expected however, in some cases the requested packet ID is for a packet ID which does not exist as part of the block. For example, if the packet trailer ID of block is 36, the data receiver requests packet ID 37 which is unexpected. Issue ID EBUS-4315.
- When a **PvSoftDeviceGEV**-based application fails to resend packet data, nothing is sent (whereas the GigE Vision standard requires that the **PvSoftDeviceGEV** send a streaming data packet with a header that includes the status). Issue ID EBUS-4311.
- In the **GenICamParameters** .NET sample application, an unhandled exception occurs when selecting a feature that is “not available”. Issue ID EBUS-4298.
- The eBUS Player histogram feature is not working properly with color images. Issue ID EBUS-4299.
- When accessing the C++ Code Samples and .NET Code Samples pages (index.html) using the Microsoft Edge browser, clicking the links to the sample applications will not open the corresponding folder. To work around

this Microsoft known issue, right-click the link, click **Copy link**, paste it into the address bar of the Edge browser, and then press **ENTER**. Issue ID EBUS-4135.

- On the Windows 7 operating system, a message will appear when installing the eBUS SDK that indicates that the publisher of the eBUS Universal Pro driver cannot be verified. Choose **Install this driver software anyway** to proceed with the driver installation. Issue ID EBUS-4188.
- For some devices, the list of options available in the **Mode** list on the main page of eBUS Player may vary from those available in the **Acquisition Mode** > **AcquisitionMode** list in the eBUS Player **Device Control** dialog box. To access all acquisition modes, use the eBUS Player **Device Control** dialog box. Issue ID EBUS-3720.
- On the Windows operating system, the following actions can cause the operating system to stop unexpectedly, with an error message displayed on a blue screen:
    - Changing the NIC parameters while streaming video with eBUS Player, the eBUS SDK sample applications, or applications created with the eBUS SDK. Issue ID EBUS-3891.
    - Disabling the eBUS Universal Pro for Ethernet driver while eBUS Player, the eBUS SDK sample applications, or applications created with the eBUS SDK are connected to a GigE Vision device. Issue ID EBUS-3867.
- The **TransmitChunkData** sample application does not display the correct number of transmitted images. The value that is displayed is lower than the actual value. Issue ID EBUS-4330.
- When using the **SoftDeviceGEV** sample application, the **SampleString** and **SampleBoolean** features are not automatically updated when you select a different **EnumEntry** in the **SampleEnum** list. Issue ID EBUS-4341.
- **PvDevice::GetAccessType** returns a non-OK status when using **PvDevice::Connect** with **PvAccessExclusive** mode. Issue ID EBUS-4398.
- Attempting to connect to a USB3 Vision device with a non-empty **UserDefinedName** using **PvDeviceU3V** will fail. Instead, you should use **PvDevice**. Issue ID EBUS-4442.
- **PvSoftDeviceGEV::Start(const PvString &aMACAddress)** returns **true** if the **aMACAddress** is an IP address (instead of returning **false**). Issue ID EBUS-4394.
- The GenICam parameter **GevSCPD** is not fully implemented for eBUS Tx. Issue ID EBUS-4366.

System Requirements

Supported Operating Systems

For the Windows operating system:

- Microsoft Windows 10, 32-bit or 64-bit
- Microsoft Windows 8.1, 32-bit or 64-bit
- Microsoft Windows 7 with Service Pack 1 (or later), 32-bit or 64-bit

Important note: The Windows 7 operating system is not recommended for new eBUS Tx designs.

For the Linux platform:

- Red Hat Enterprise Linux 7, 64-bit
- CentOS 7, 64-bit with kernel version 3.zxz

- Ubuntu 20.04 LTS, 64-bit with kernel versions 5.4.0 and 5.8.0
- Ubuntu 18.04 LTS, 64-bit with kernel versions 4.4.0-143 and 5.4.0
- Ubuntu 16.04 LTS, 32-bit or 64-bit with kernel version 4.15.0-46
- Ubuntu 14.04 LTS, 32-bit or 64-bit with kernel version 4.4.0-143

Note: The eBUS SDK is supported on the following Linux ARM platforms: NVIDIA Jetson Nano, Jetson AGX Xavier, Jetson TX2, and Jetson TX2i platforms (Ubuntu 18.04 with Jetpack 4.2)

Required Hardware

The following hardware is required:

- Gigabit Ethernet network card or USB 3.0 host controller.

Pleora has validated and supports the following USB 3.0 chipsets:

- NEC Electronics/Renesas Electronics chipset
- Intel® Ivy Bridge chipset

Supported Development Environments

The following development environments are supported:

For the Windows operating system:

- Visual Studio 2019, 2017, 2015, 2013, 2012, and 2010

For the Linux operating system, Qt and qmake are required to compile GUI-based samples:

- For Ubuntu 20.04 Desktop: Qt 5.12.8
- For Ubuntu 18.04 Desktop: Qt 5.9.5
- For Ubuntu 16.04 Desktop: Qt 5.5.1
- For Ubuntu 14.04 Desktop: Qt 5.2.1
- For Ubuntu for ARM: Qt 5.9.5
- For RHEL/CentOS7: Qt 5.9.2

GenApi, GenCP, GigE Vision, and USB3 Vision Support

The following table lists the supported GenApi, GenCP, GigE Vision, and USB3 Vision versions.

Table 1: GenApi, GenCP, GigE Vision, and USB3 Vision Support

Component	Supported version
GenApi (GenICam)	Version 3.1.0
GenCP	Version 1.0
GigE Vision	Version 2.1 (and earlier)
USB3 Vision	Version 1.0

For More Information

When you install the eBUS SDK, you can access documentation that describes the interfaces, classes, and functions that are available. This documentation is only available when you install the eBUS SDK (not the eBUS Player Toolkit).

Table 2: eBUS SDK Documentation

Operating system	eBUS SDK documentation location
Windows	Windows Start menu > eBUS Or: C:\Program Files\JAI\eBUS SDK\Documentation
Linux	<*installation_directory*>/share/doc/sdk/index.html

Additional documentation is available on the Pleora Technologies Support Center (http://www.pleora.com/support-center), such as:

- *eBUS Player User Guide*, available for Windows and Linux
- *eBUS SDK API Quick Start Guides*, available for C++, .NET, and Linux
- *Getting Started with eBUS Tx*
- *eBUS SDK 3.x to 4.x Migration Guide*
- *Vision SDK to eBUS SDK Migration Guide*
- *eBUS SDK Licensing Overview Knowledge Base Article*

Copyright Information

Document Version

EX008-017-0023 Version 1.0, January 2021