Sapera LT™ 8.60

Getting Started Manual
for USB3 Vision Cameras

sensors | cameras | frame grabbers | processors | **software** | vision solutions

P/N: OC-SAPM-GSUSB

www.teledynedalsa.com

TELEDYNE DALSA
Everywhere**you**look™

NOTICE

Printed on October 16, 2020

Document Number: OC-SAPM-GSUSB
Printed in Canada

About This Manual

This manual exists in Windows Help, and Adobe Acrobat® (PDF) formats (printed manuals are available as special orders). The Help and PDF formats make full use of hypertext cross-references. The Teledyne DALSA home page on the Internet, located at http://www.teledynedalsa.com/imaging, contains documents, software updates, demos, errata, utilities, and more.

About Teledyne DALSA

Teledyne DALSA, a business unit of Teledyne Digital Imaging Inc., is an international high-performance semiconductor and electronics company that designs, develops, manufactures, and markets digital imaging products and solutions, in addition to providing wafer foundry services.

Teledyne Digital Imaging offers the widest range of machine vision components in the world. From industry-leading image sensors through powerful and sophisticated cameras, frame grabbers, vision processors and software to easy-to-use vision appliances and custom vision modules.

Contents

Sapera LT with USB3 Vision Cameras

USB3 Vision cameras rely on GenICam™ to describe the features supported by the camera, providing a user-friendly common interface to control camera functionality across platforms. These devices use an XML file, stored on the device, to describe the camera features, which are standardized. These features are then exposed through the Sapera API.

All devices that are GenICam compliant use the same feature names (established by the Standard Feature Naming Convention (SNFC)) to control the camera, though manufacturers can use camera specific features that are not part of this standard.

For USB cameras, the Sapera LT USB3 Vision driver is employed. Teledyne Lumenera USB-3 Vision compliant cameras provide a quick and easy means of capturing high quality images on any USB 3.0 equipped desktop, laptop or embedded computer. Because they are USB-based, there is no need for a frame grabber. Instead, a single cable provides full command control and data transfer.

USB 3 cameras are powered via the USB 3 computer port. Cameras also have an external interface header for hardware input, output signals and optional camera power.

Supported Industry Standards

USB3™ VISION	The USB3 Vision® interface is based on the standard USB 3.x interface found on all current PCs and many embedded systems. Cameras and other devices utilizing this standard are compatible with a wide range of products from many vendors. The standard is currently in version 1.1. For more information see: https://www.visiononline.org/

Introduction to Sapera LT

Sapera™ LT is a software API for controlling image acquisition devices such as USB3 Vision cameras. Sapera LT libraries support Teledyne DALSA cameras and frame grabbers as well as hundreds of 3rd party camera models across all common interface formats like USB3 Vision®, GigE Vision®, Camera Link®, as well as emerging new image acquisition standards such as CLHS.

⚠ If your application requires image processing or GPU optimization, Sapera Essential, a full-featured image processing library, is available as a separate software package. For more information see www.teledynedalsa.com/imaging/products/software/.

This manual introduces the Sapera LT API and is designed to help programmer's with installation, and quickly perform hardware setup and validation.

Sapera LT Licensing

Sapera LT is available free of charge, both SDK and runtime versions, when used with Teledyne DALSA frame grabber or camera products.

The Sapera LT APIs

Sapera LT includes everything you need to acquire and display images, using one of its 3 application programming interfaces (API):

- Sapera LT++ classes (based on C++ language)
- Sapera LT .NET classes (based on .NET languages)
- Sapera LT Standard API (based on C language)

It is targeted at developers that have their own image processing libraries and want to interface those libraries to a Sapera LT compatible device. Sapera LT includes tools such as CamExpert to speed up application development.

Hardware independent classes allow one application to control different Teledyne DALSA devices through the same API. It also guarantees seamless migration to any future Teledyne DALSA hardware product supported by Sapera LT.

Sapera Tools and Utilities

The Sapera LT SDK and runtime installations include the following a set of tools and utilities:

- Sapera Explorer: provides quick access to all tools, demos, examples and source code
- CamExpert: acquisition and device configuration, including image display
- Sapera Monitor: real-time event viewing of applications.
- Sapera Log Viewer: error and other message log for applications and hardware
- Sapera Configuration Utility: memory resource configuration and multi-threading optimization

Sapera LT Manual Descriptions

Sapera LT is supported by the following manuals in PDF, and compiled HTML help formats.

API/Topic	Title	Description
General	**Sapera LT Getting Started Manual for Frame Grabbers**	Provides a general overview of the Sapera LT APIs with frame grabbers, possible hardware configurations and a quick overview of Sapera programming.
General	**Sapera LT Getting Started Manual for GigE Vision Cameras and 3D Sensors**	Provides a general overview of the Sapera LT APIs with GigE Vision cameras and 3D sensors, a quick overview of Sapera programming as well as troubleshooting guidelines.
General	**Sapera LT Getting Started Manual for USB3 Vision Cameras**	Provides a general overview of the Sapera LT APIs with USB3 Vision cameras, a quick overview of Sapera programming as well as troubleshooting guidelines.
General	**Sapera LT User's Manual**	Introduces Sapera LT ++ API and Sapera .NET API programming procedures, including sample code for typical operations in C++, C# and VB .NET.
GigE Vision	**Network Imaging Package for Sapera LT Optimization Guide**	Network setup and optimization guide for GigE Vision cameras.
.NET	**Sapera LT .NET Programmer's Manual**	Provides a complete reference of the Sapera .NET Framework for Visual Studio. Sapera .NET reflects the underlying low-level Sapera LT architecture.
C++	**Sapera LT ++ Programmer's Manual**	Provides a complete reference of all the Sapera LT ++ classes. Sapera LT ++ is based on the C++ language.
C++	**Sapera LT GUI Classes Reference Manual**	Describes the C++ GUI (graphical user interface) helper classes used to create common application dialogs. These classes are used in the Sapera demo programs and are provided to help users with applications that require a GUI.
C++	**Sapera LT Legacy Classes Reference Manual**	Describes the obsolete C++ classes that continue to be supported but have been replaced or retired.
C	**Sapera LT Acquisition Parameters Reference Manual**	Describes the Sapera LT low level acquisition parameters and capabilities (based on the C language).
C	**Sapera LT Basic Modules Reference Manual**	Lists in detail the Sapera LT low-level module functions as well as data definitions, file formats, and macros (based on the C language).

Supported Operating Systems

- Windows 10 (64-bit version)

Supported Sapera LT Development Environments

- C/C++ and .NET language compilers, for both 32-bit and 64-bit development:
    - Microsoft Visual Studio 2010
    - Microsoft Visual Studio 2012
    - Microsoft Visual Studio 2013
    - Microsoft Visual Studio 2015
    - Microsoft Visual Studio 2017

Installation Types

When installing Sapera LT, you are provided with a choice of different setup types, each with its own dedicated installation program.

The full SDK (software development kit) installation provides access to all available Sapera LT functions, tools, and utilities, such as CamExpert, for 32 or 64-bit application development, and the GigE Vision module (required when using GigE cameras) or USB3 Vision Interface (required when using USB3 Vision cameras).

Installation Type	Notes
Sapera LT Full SDK	Full installation of the software development kit, including all tools and utilities. Installation options allow you to install components for frame grabbers, for GigE Vision cameras (includes the Sapera Network Imaging Package), 3D sensors or USB3 Vision cameras.
Sapera LT Runtime (32 or 64-bit)	Runtime installation for application deployment, including all tools and utilities, and optionally the GigE Vision module (if required). The USB3 Vision interface is available in 64-bit version only.
Sapera LT Runtime WoW64 (32-bit application for 64-bit OS)	Runtime installation for application deployment, including all tools and utilities, and optionally the GigE Vision module (if required).
CamExpert	Installation of CamExpert only (includes GigE Vision module and USB3 Vision interface).
CamExpert WoW (32-bit application for 64-bit OS)	Installation of CamExpert only (includes GigE Vision module).
Sapera LT Camera SDK	Installation of Sapera LT Camera SDK for Gen CP CameraLink cameras. Feature control only.
Sapera LT Camera SDK Runtime	Runtime installation for application deployment of Sapera LT Camera SDK for GenCP CameraLink cameras. Feature control only.

These executable files are available for download directly from the Teledyne DALSA website:

http://teledynedalsa.com/imaging/support/downloads/sdks/

⚠	Note: The Sapera LT Camera SDK is also available for use when using Teledyne DALSA Camera Link cameras with third party frame grabbers only. It is provided as part of the Teledyne DALSA camera installation.

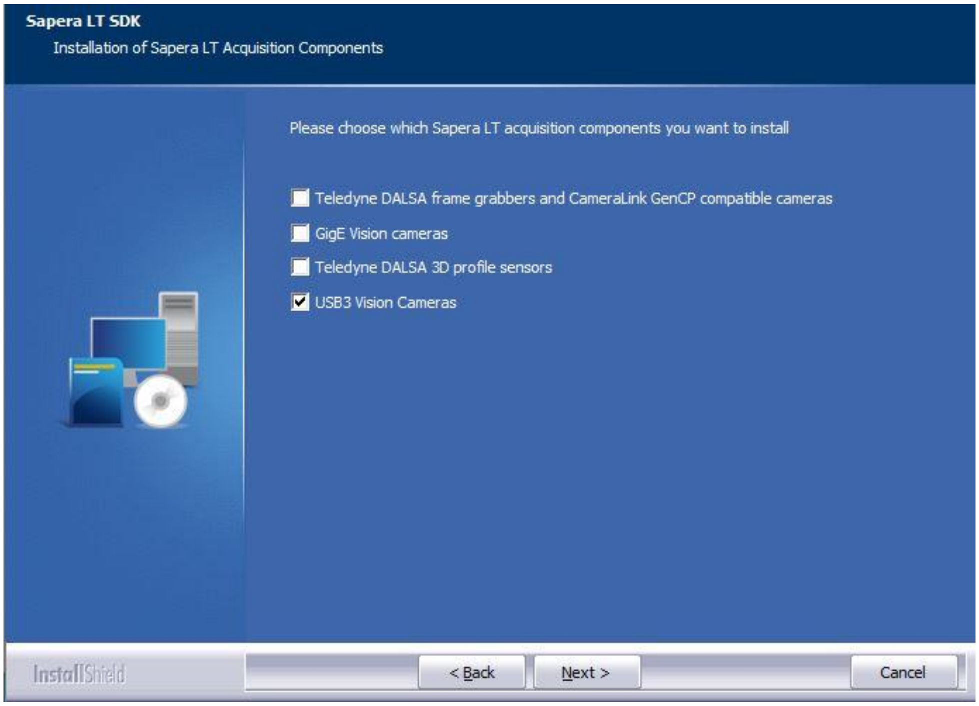
Installing Sapera LT for use with USB3 Vision

To install the Sapera LT USB3 Vision driver, it is recommended that the installation be started locally instead of from a network location.

Warning: When installing Sapera LT for USB 3 Vision cameras, no cameras can be connected to the host computer.

The Sapera LT USB3 Vision Interface is only available under Windows 10 64-bit.

During the installation process, you are prompted to choose the Sapera LT acquisition components to install. Select the **USB3 Vision Cameras** options. This will install the Sapera LT USB3 Vision Interface on your system.



Sapera LT SDK

Installation of Sapera LT Acquisition Components

Please choose which Sapera LT acquisition components you want to install

☐ Teledyne DALSA frame grabbers and CameraLink GenCP compatible cameras

☐ GigE Vision cameras

☐ Teledyne DALSA 3D profile sensors

☑ USB3 Vision Cameras

Upgrading Previous Versions of Sapera LT

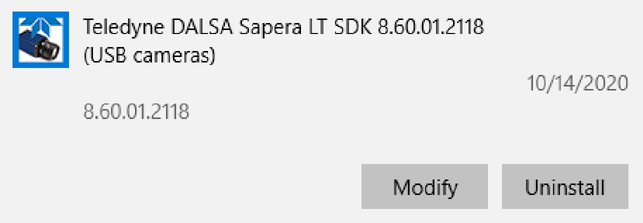
Sapera LT 8.51 and Lower

⚠	Warning: When upgrading from a previous version of Sapera LT (before 8.60), Sapera LT must be uninstalled first.

⚠	Note: The installation option is persistent. When upgrading to a newer version of Sapera LT (8.60 or higher), the previously chosen options (frame grabbers/GigE Vision Cameras/3D Sensors/USB3 Vison Cameras) are used. If you want to change options you must uninstall Sapera LT first.

To verify the current installation option, check the Sapera LT SDK entry in the **Windows Settings > Apps and features** section.

For example, if the **USB3 Vision Cameras** option was selected, it is indicated as follows:



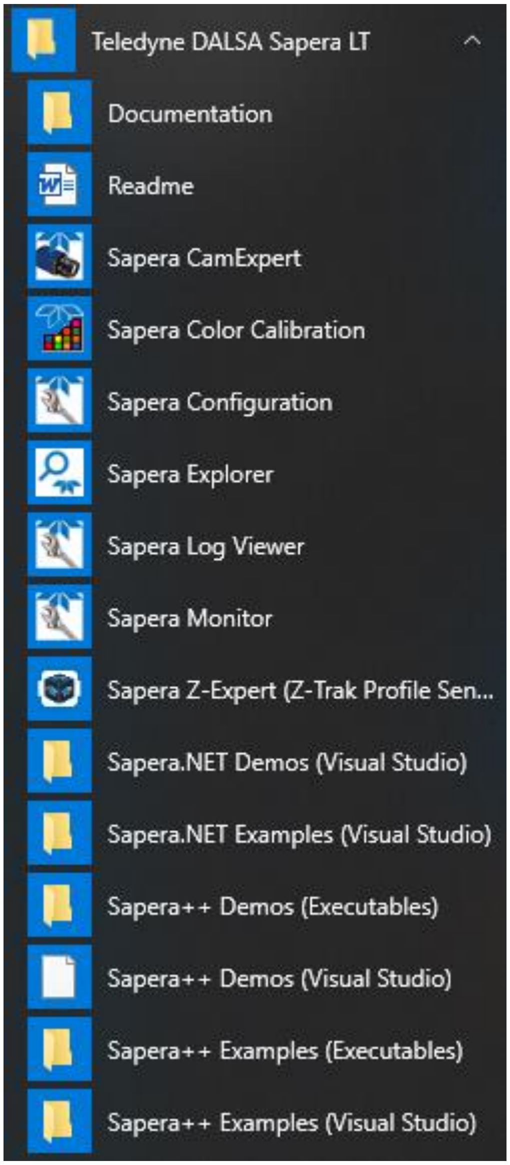
Teledyne DALSA Sapera LT SDK 8.60.01.2118 (USB cameras)

10/14/2020

8.60.01.2118

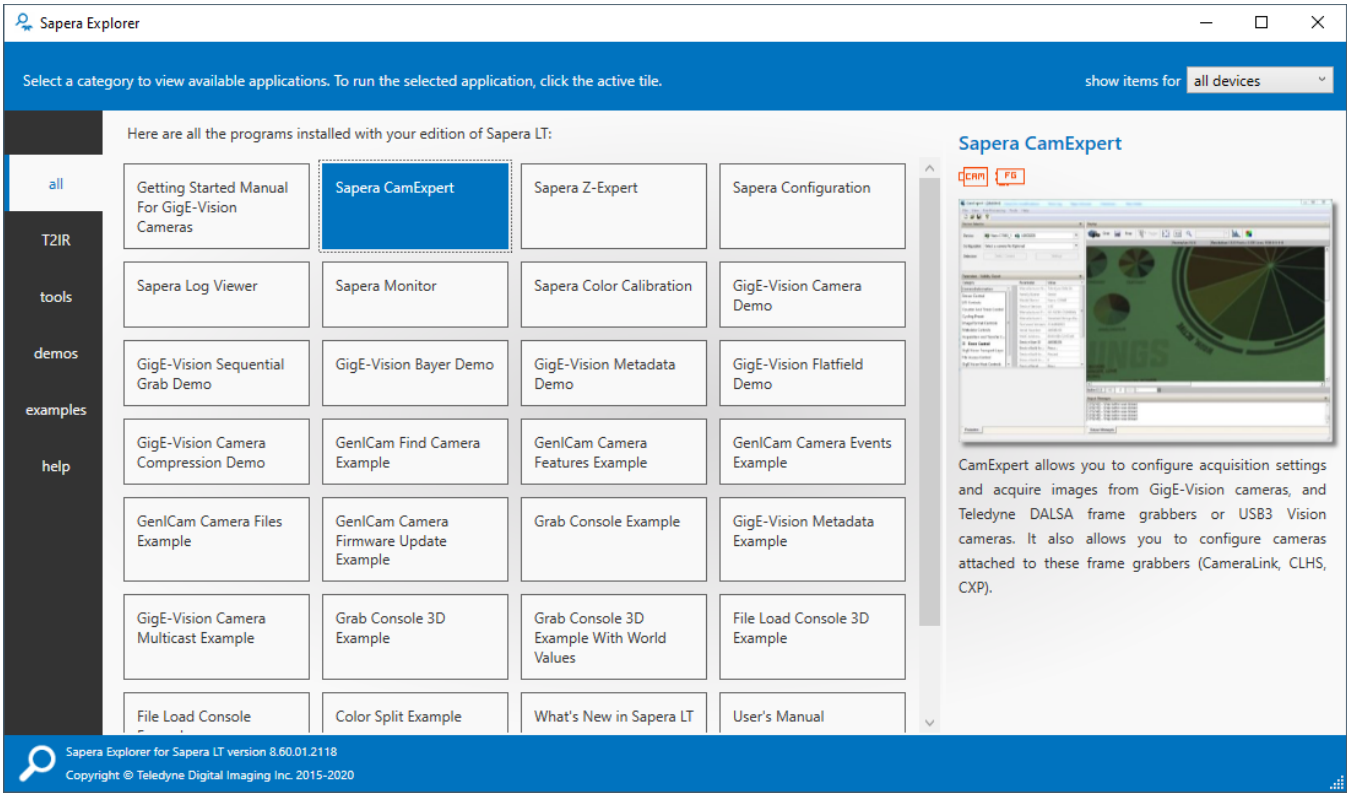
Start Menu Shortcuts

For Windows 10, Start menu shortcuts for Sapera LT are available under Teledyne DALSA Sapera LT.


Teledyne DALSA Sapera LT
Documentation
Readme
Sapera CamExpert
Sapera Color Calibration
Sapera Configuration
Sapera Explorer
Sapera Log Viewer
Sapera Monitor
Sapera Z-Expert (Z-Trak Profile Sen...
Sapera.NET Demos (Visual Studio)
Sapera.NET Examples (Visual Studio)
Sapera++ Demos (Executables)
Sapera++ Demos (Visual Studio)
Sapera++ Examples (Executables)
Sapera++ Examples (Visual Studio)

Sapera Explorer

The Sapera Explorer application provides quick access to the Sapera LT tools documentation, demos, examples and source code, grouped by category. Select a category to view available applications. To run the selected application, click the active tile.



Sapera Explorer

Select a category to view available applications. To run the selected application, click the active tile.

show items for: all devices

Here are all the programs installed with your edition of Sapera LT:

Getting Started Manual For GigE-Vision Cameras	Sapera CamExpert	Sapera Z-Expert	Sapera Configuration
Sapera Log Viewer	Sapera Monitor	Sapera Color Calibration	GigE-Vision Camera Demo
GigE-Vision Sequential Grab Demo	GigE-Vision Bayer Demo	GigE-Vision Metadata Demo	GigE-Vision Flatfield Demo
GigE-Vision Camera Compression Demo	GenICam Find Camera Example	GenICam Camera Features Example	GenICam Camera Events Example
GenICam Camera Files Example	GenICam Camera Firmware Update Example	Grab Console Example	GigE-Vision Metadata Example
GigE-Vision Camera Multicast Example	Grab Console 3D Example	Grab Console 3D Example With World Values	File Load Console 3D Example
File Load Console	Color Split Example	What's New in Sapera LT	User's Manual

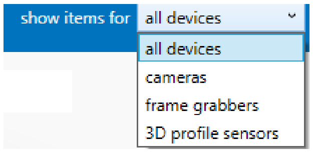
Sapera CamExpert

CAM FG

CamExpert allows you to configure acquisition settings and acquire images from GigE-Vision cameras, and Teledyne DALSA frame grabbers or USB3 Vision cameras. It also allows you to configure cameras attached to these frame grabbers (CameraLink, CLHS, CXP).

Sapera Explorer for Sapera LT version 8.60.01.2118
Copyright © Teledyne Digital Imaging Inc. 2015-2020

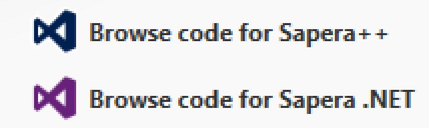
Items can be filtered using the **show items for** drop-down list:



show items for: all devices

- all devices
- cameras
- frame grabbers
- 3D profile sensors

For Demos and Examples, click **Browse code for Sapera ++** or **Browse code for Sapera .NET** to open the source code directory.



Browse code for Sapera++

Browse code for Sapera .NET

The CAM, FG and 3D icons indicate whether the demo or example is supported by cameras, frame grabbers or 3D sensors.

CAM FG 3D

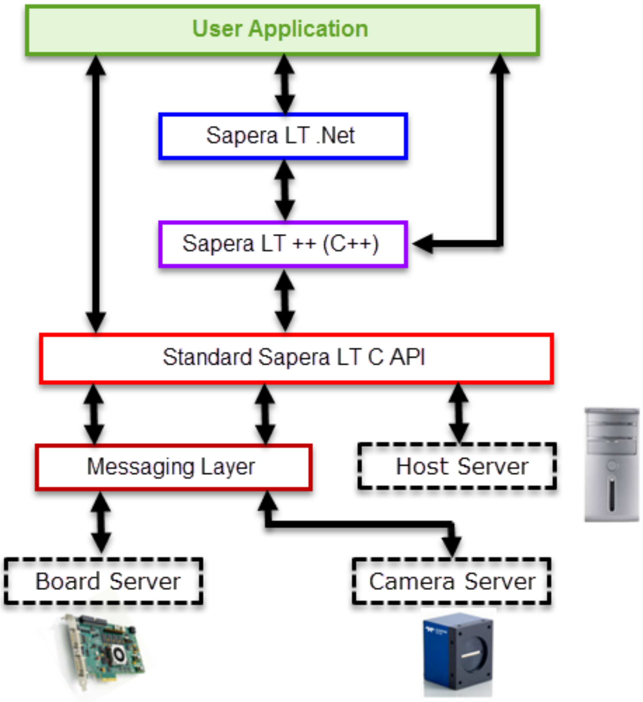
Note: GigE Vision demos and examples are compatible with USB3 Vision cameras since they use the GenICam protocol.

Sapera LT Architecture Overview

The following section describes application architecture, related terms, and illustrates Sapera LT's library architecture.

Application Architecture

Whichever API is used (Sapera LT ++, Sapera LT .NET, or Standard C), the Sapera LT modular architecture allows applications to be distributed on different Sapera LT servers. Each server can run either on the host computer or on a Teledyne DALSA device. Sapera LT calls are routed to different servers via the Sapera LT messaging layer in a fashion completely independent of the underlying hardware.



What is a server?

A Sapera Server is an abstract representation of a physical device like a frame grabber, a camera or 3D sensor, or a desktop PC. In general, a Teledyne DALSA board is a server. Some processing boards, however, may contain several servers; this is true when using multi-processor boards.

A server allows Sapera applications to interact with the server's resources. The server name consists of the product model name and an index. For example, the Teledyne DALSA Luminera C2020 camera has a server name "Lt-C2020_1". If more than one camera of the same type is available, the index differentiates the cameras.

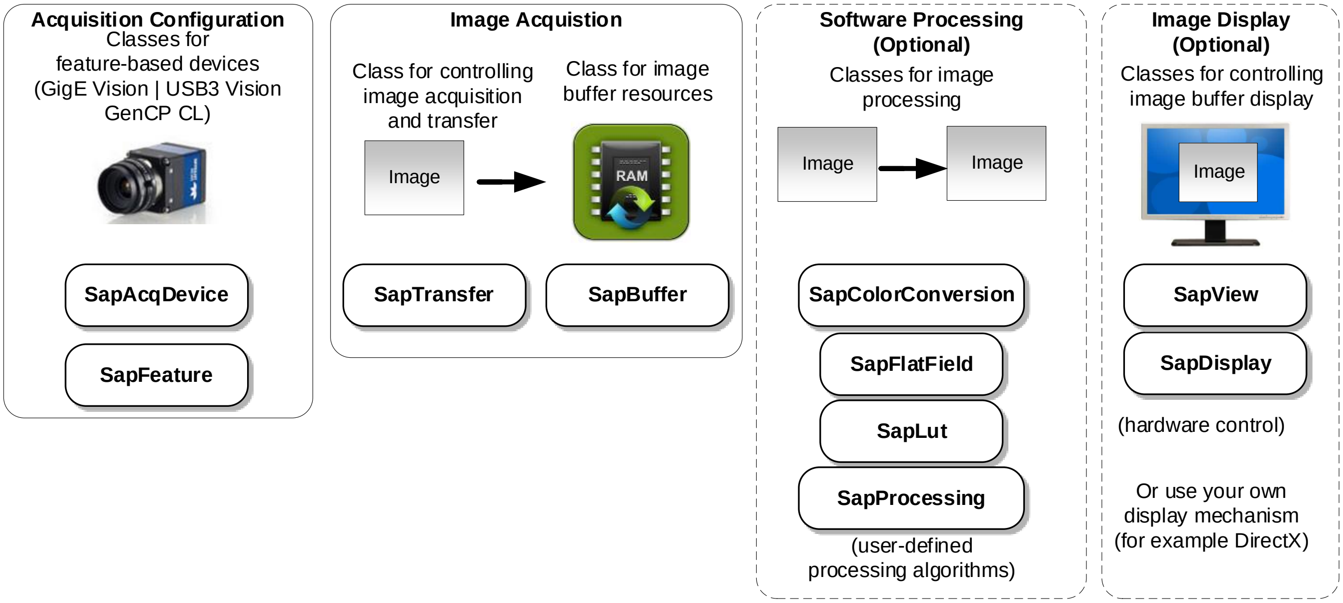
Library Architecture

The typical machine vision application requires configuration of acquisition resources, image capture and transfer to memory buffers. These image buffers can then be processed or displayed, analyzed, with results determining subsequent processes. Events can also be monitored to trigger appropriate responses. The Sapera LT library architecture is organized around these basic machine vision functional blocks.

Vision applications developed with Sapera LT are typically programmed in either the Sapera LT++ API (Application Programming Interface) or the Sapera LT .NET API. In general, both APIs use similar classes and naming conventions.

Sapera LT Standard C API is available for programmers who prefer working with the underlying Sapera LT C layer or who are maintaining legacy code. It provides access most of the same functionality as the higher level, object-oriented programming C++ and .NET APIs (for example, it does not support flat field calibration and software correction). For more information refer to the Sapera LT Basic Modules Reference Manual and the Sapera LT Acquisition Parameters Reference Manual.

The following block diagram, while not exhaustive of all the classes available in Sapera LT, illustrates the major functional blocks with the corresponding classes.


Acquisition Configuration
Classes for feature-based devices (GigE Vision | USB3 Vision GenCP CL)
SapAcqDevice
SapFeature
Image Acquistion
Class for controlling image acquisition and transfer
Class for image buffer resources
Image
RAM
SapTransfer
SapBuffer
Software Processing (Optional)
Classes for image processing
Image
Image
SapColorConversion
SapFlatField
SapLut
SapProcessing
(user-defined processing algorithms)
Image Display (Optional)
Classes for controlling image buffer display
Image
SapView
SapDisplay
(hardware control)
Or use your own display mechanism (for example DirectX)

For the complete reference to the Sapera LT APIs refer to the **Sapera LT ++ Programmer's Manual** or **Sapera LT.NET Programmer's Manual**.

In addition, the **Sapera LT User's Manual** provides explanations and multiple code snippets, in both C++ and .NET languages, for typical application operations.

⚠	It is always recommended to use the source code provided with the demos and examples as both a learning tool and a starting point for your applications. For a complete list and description of the demos and examples included with Sapera LT see Demos and Examples.

Configuration Files

Most USB3 Vision cameras support saving a user-defined set of parameter settings that can be loaded instead of the factory defaults on power-up.

If the camera does not support file access, Sapera LT applications can load an acquisition configuration file before acquiring images from a camera. The camera configuration file has the extension **.ccf**.

Use CamExpert to generate a *.ccf* file for cameras with parameters as required by your imaging project.

The SapAcqDevice constructor, for GeniCam-compliant cameras, have prototypes that use *.ccf* files. For cameras, if no *.ccf* file is available, the camera default parameters are used.

USB3 Vision XML Files for Host and Device Controls

The USB3 Vision XML device description file is retrieved from devices such as the Teledyne DALSA USB3 Vision products, allowing USB3 Vision Compliant applications to retrieve the camera capabilities. The Host Control XML file is provided with the USB3 Vision Interface.

USB3 Vision Host Controls XML File

The USB3 Vision Host Controls are independent of any USB3 Vision device used with the Teledyne DALSA driver. The default parameter values are specified by an XML file installed by the Teledyne DALSA USB3 Vision Interface.

At application run time, the Host controls XML contents are merged with the USB3 Vision device XML to define the total feature set of the vision system. See USB Vision Host Controls for details about these Host controls.

- File location: \Teledyne DALSA\USB Vision\hostfeatures*xx*.xml.
- User can change parameter default values if required.
- User should respect the specified minimum and maximum parameter range values to avoid problems.

Automatic Retrieval of Device XML Files

Device XML feature files are automatically retrieved from the device when first connected. By convention, the XML files will have unique names such that new versions for the device are easily identified.

- When connecting a device for the first time, its XML parameter file is copied to the host system (for example, in the <Install>\ProgramData\Teledyne DALSA\GenICam\download directory).
- When connecting the same device again, its XML file is not retrieved unless the file name is different than the one previously transferred to the host system.
- The Teledyne DALSA XML file naming convention for USB3 Vision cameras includes the camera firmware version, for easy identification.

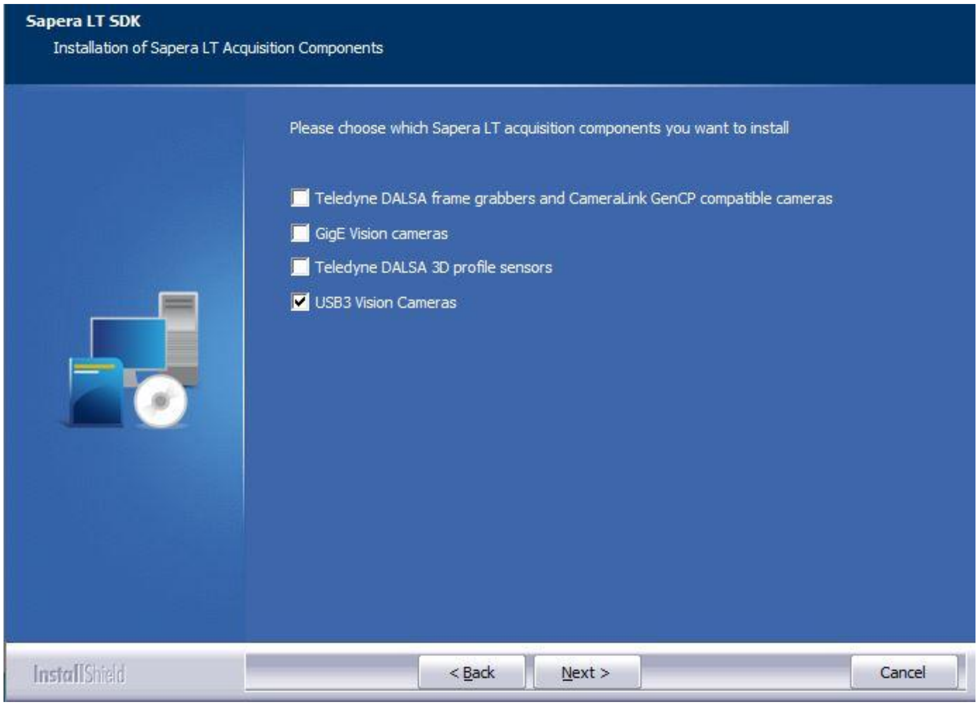
Sapera Camera Demo and Example Code

The Sapera LT GigE Vision demos and examples are compatible with USB3 Vision since they use the GenICam standard. However, certain demo and example features may not be supported by all USB3 Vision devices.

Program	**Teledyne DALSA•Sapera LT•Demos•Cameras•GigE Vision Camera Demo**
Program file	\…\...\Sapera\Demos\Classes\vc\GigeCameraDemo\Release\GigeCameraDemo.exe
Workspace	\…\...\Sapera\Demos\Classes\vc\SapDemos_2010.sln \…\...\Sapera\Demos\Classes\vc\SapDemos_2012.sln \…\...\Sapera\Demos\Classes\vc\SapDemos_2013.sln \…\...\Sapera\Demos\Classes\vc\SapDemos_2015.sln \…\...\Sapera\Demos\Classes\vc\SapDemos_2017.sln
Description	This program demonstrates the basic acquisition functions included in the Sapera library. The program allows you to acquire images, either in continuous or in one-shot mode. The program code may be extracted for use within your own application.
Remarks	The executable provided in the Sapera LT installation for this demo is built using Visual C++ 2013. It is based on Sapera C++ classes. See the Sapera User’s and Reference manuals for more information.

Quick Start Guide

For Teledyne DALSA USB3 Vision cameras, during installation, choose the option to install Sapera LT for USB3 Vision cameras.



Sapera LT SDK

Installation of Sapera LT Acquisition Components

Please choose which Sapera LT acquisition components you want to install

□ Teledyne DALSA frame grabbers and CameraLink GenCP compatible cameras

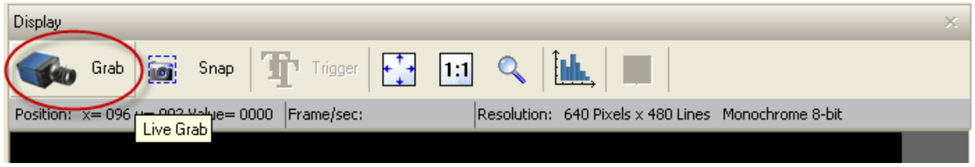
□ GigE Vision cameras

□ Teledyne DALSA 3D profile sensors

☑ USB3 Vision Cameras

Connecting a Teledyne Lumenera USB3 Vision Camera

To configure a Teledyne Lumenera USB3 Vision Camera:
For USB3 Vision cameras, start the CamExpert application, select the camera to configure, and modify the camera parameter settings as required, and test the image acquisition by clicking the **Grab** button.



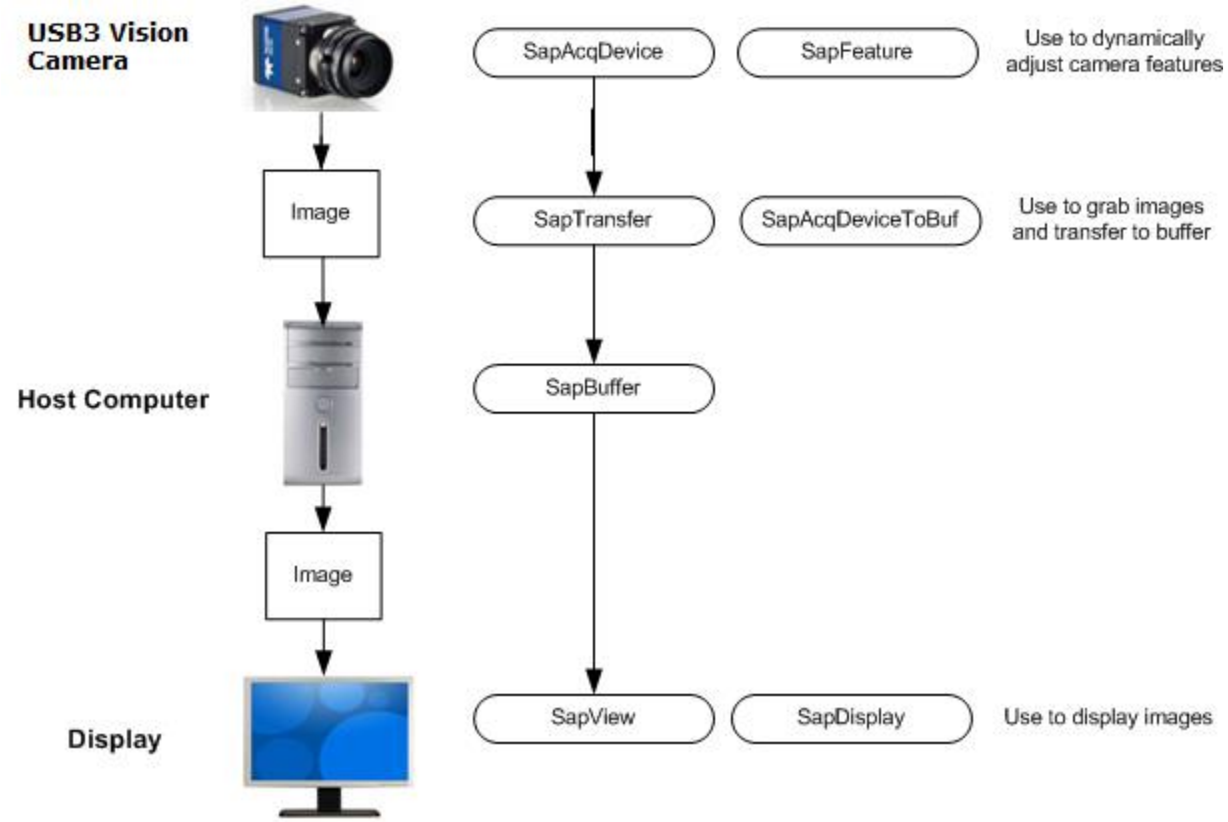
Position: x= 096 y= 002 Value= 0000 | Frame/sec: | Resolution: 640 Pixels x 480 Lines Monochrome 8-bit

Note: if a properly powered and connected camera is not found, another USB3 Vision driver may be present and active on the system; see the No Device Detected in CamExpert troubleshooting section.

Using Sapera LT with a Teledyne Lumenera USB3 Vision Device

Sapera LT Classes for USB3 Vision Device Applications

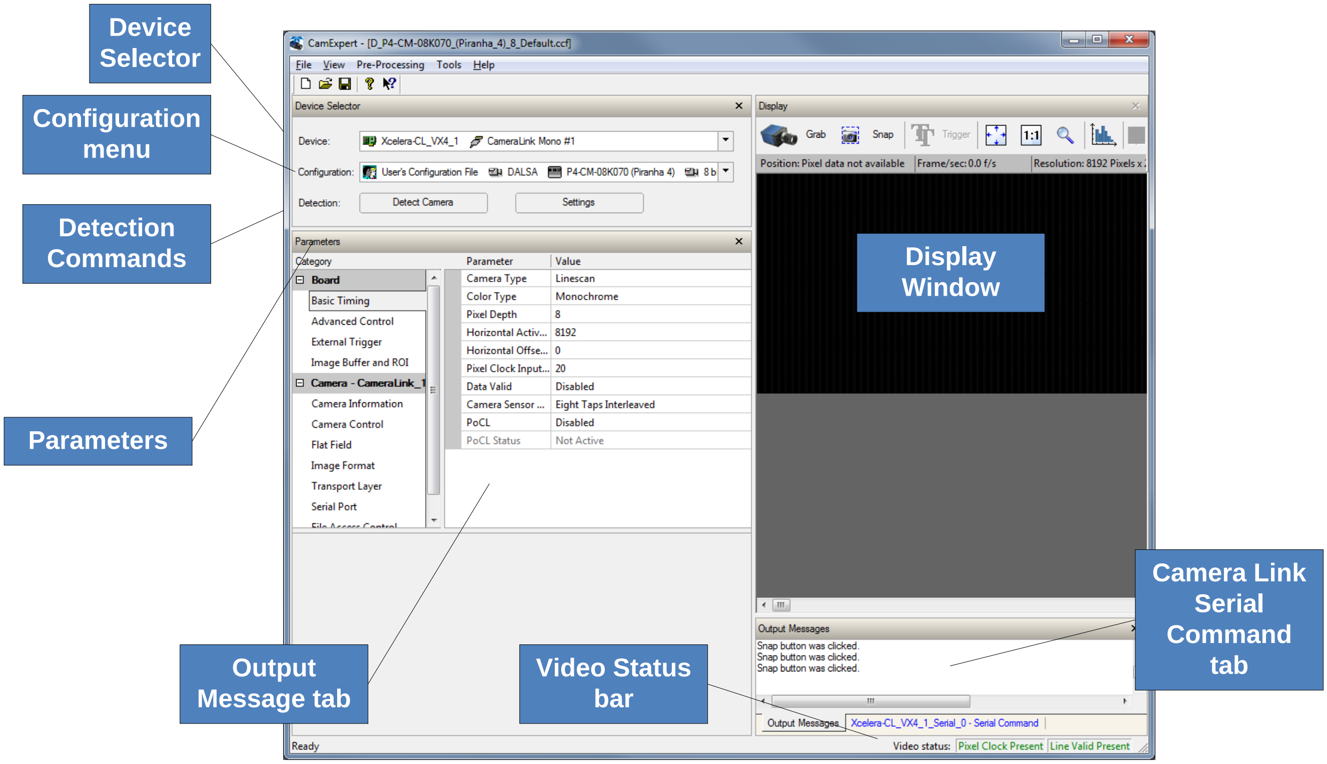
The following diagram represents a typical application flow showing the Sapera LT object types associated with each component or stage.



Using the CamExpert Tool

The CamExpert application uses various panels to simplify choosing and configuring camera files or acquisition parameters for the installed devices. Features include:

- Supports all Teledyne DALSA hardware currently supported by Sapera LT.
- Creates and modifies camera configuration (*.ccf* files).
- Supports Teledyne Lumenera USB3 Vision cameras, such as the Lt series, by presenting the camera Features controls as defined by the camera’s XML file.
- Groups acquisition parameters into related categories for easier access to any specific parameter.
- Intelligent editing of video timings through a locking mechanism that allows explicit modification of some values and automatic recalculation of the remaining ones.
- Live acquisition display window which allows immediate verification of timing or control parameters without the need to run a separate acquisition program



Device: Xcelera-CL_VX4_1 CameraLink Mono #1

Configuration: User's Configuration File DALSA P4-CM-08K070 (Piranha 4) 8 b

Detection:

Position: Pixel data not available | Frame/sec: 0.0 f/s | Resolution: 8192 Pixels x

Parameter	Value
Camera Type	Linescan
Color Type	Monochrome
Pixel Depth	8
Horizontal Activ...	8192
Horizontal Offse...	0
Pixel Clock Input...	20
Data Valid	Disabled
Camera Sensor ...	Eight Taps Interleaved
PoCL	Disabled
PoCL Status	Not Active

```
Snap button was clicked.
Snap button was clicked.
Snap button was clicked.
```

Video status: Pixel Clock Present Line Valid Present

- **Device Selector**: The Device menu allows you to view and select from any installed Sapera acquisition device. After a device is selected, CamExpert only presents acquisition parameters applicable to that device.
The Configuration menu allows selecting any camera file that is included with the Sapera installation. Only camera files supported by the selected acquisition device are displayed. When there is more than one acquisition server, such as monochrome and RGB, selecting an inappropriate camera file will produce a message prompting you to select the correct acquisition server.
Parameters panel: Allows viewing or changing all acquisition parameters supported by the acquisition device. CamExpert displays parameters only if those parameters are supported by the installed device. This avoids confusion by eliminating parameter choices when they do not apply to the hardware in use.

- **Display panel**: Provides a live or single frame acquisition display. Frame buffer parameters are shown in an information bar above the image window. The Display pane includes CamExpert control buttons. These are:

Grab Freeze	**Acquisition control button:** Click once to start live grab, click again to stop.
Snap	**Single frame grab:** Click to acquire one frame from device.
T Trigger	**Software trigger button:** With the I/O control parameters set to Trigger Enabled / Software Trigger type, click to send a single software trigger command.
✥ 1:1 🔍	**CamExpert display controls:** (these do not modify the frame buffer data) Stretch image to fit, set image display to original size, or zoom the image to any size and ratio.
📊	**Histogram / Profile tool:** Select to view a histogram or line/column profile during live acquisition.

- **Output Messages Panel**: Displays messages from CamExpert or the device driver.
- **Video Signal Status bar**: Located on the lower right of the CamExpert window, color coded camera signal status information is displayed. These are in green for valid signals detected, and in red for missing or incorrect signals. Video status items may differ with different devices.

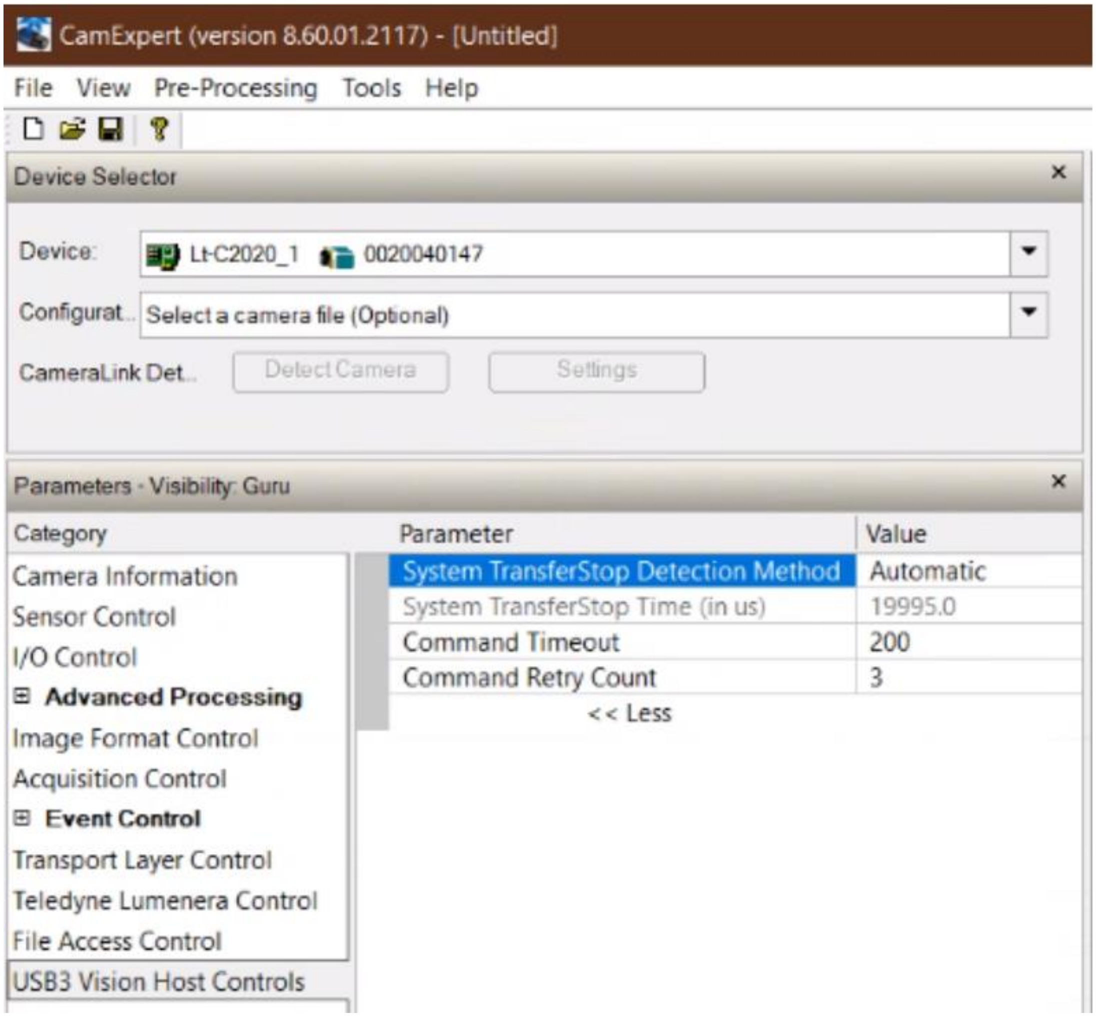
Additional Information

See either the corresponding device user’s manual or search within this manual for limitations applicable to specific Teledyne DALSA hardware.

USB Vision Host Controls

The USB Vision Host controls as shown by CamExpert, groups parameters used to configure the host computer system USB3 Vision features, which are used for connected camera networking management. None of these features are stored in the connected camera – they remain as settings to the host system control software.

Parameters in gray are read only, either always or due to another parameter being disabled. Parameters in black are user set in CamExpert or programmable via an imaging application.



Device: Lt-C2020_1 0020040147

Configurat... Select a camera file (Optional)

Parameter	Value
System TransferStop Detection Method	Automatic
System TransferStop Time (in us)	19995.0
Command Timeout	200
Command Retry Count	3

USB3 Vision Host Control Feature Descriptions

Display Name	Feature	Description	View
System TransferStop Detection Method	systemTransferStopDetectionMethod	Specify if the systemTransferStopdetectionMethod feature is based on the USB3 Vision driver or is controlled by the User, based on the SystemTransferStopTime feature. SystemTransferStopTime defines if a TransferStop is truly completed, if no data arrives from the device after the timer count is the last DATA trailer.	Guru
	Automatic	*The stream is considered stopped when it is idle for more than the current exposure time.*	
	Manual	*The transfer is considered stopped when it is idle for more than the feature systemTransferStopTime.*	
System Transfer Stop Time	systemTransferStopTime	When the feature systemTransferStopDetectionMethod is set to Manual, this is used to set the time a transfer can be inactive before been considered stopped. This time is only used by the CorXferStop and CorXferWait functions.	Guru
Command Timeout	CommandAcknowledgeTimeout	Specifies the time the host system controller will wait for a command acknowledgment from the connected USB3 Vision device. Minimum and maximum values are dependent on the connected device (as defined in its XML file).	Beginner
Command Retry Count	CommandRetryCount	Specifies the number of retries for a command sent to a device.	Beginner

Sapera LT Utilities

Sapera LT includes the following utilities that can be used to monitor Sapera LT hardware and software events, error messages, as well as network configuration and diagnostics:

Sapera Monitor	Monitors user selected events generated by a Sapera LT application.
Sapera Log Viewer	Displays error and other messages generated by Sapera LT applications and Teledyne DALSA hardware.
Recover Camera Utility	Command line utility to recover incorrectly configured cameras using the MAC address. Can be used if the Network Configuration Tool recovery fails.
U3V Device Manager	Displays available USB3 Vision cameras and drivers. Allows setting the active USB3 Vision driver to use.

Sapera Monitor

As part of the Trigger-to-Image-Reliability (T2IR) framework, the Sapera Monitor tool allows users to view the acquisition and transfer events generated by an acquisition device in real-time. This is very useful since one can use the Sapera Monitor tool to debug application s and identify problems without having to code event handlers.

The key advantage to Sapera Monitor is that it can run concurrently with CamExpert or your own application. This can be useful for debugging applications and identifying problems without having to code event handlers.

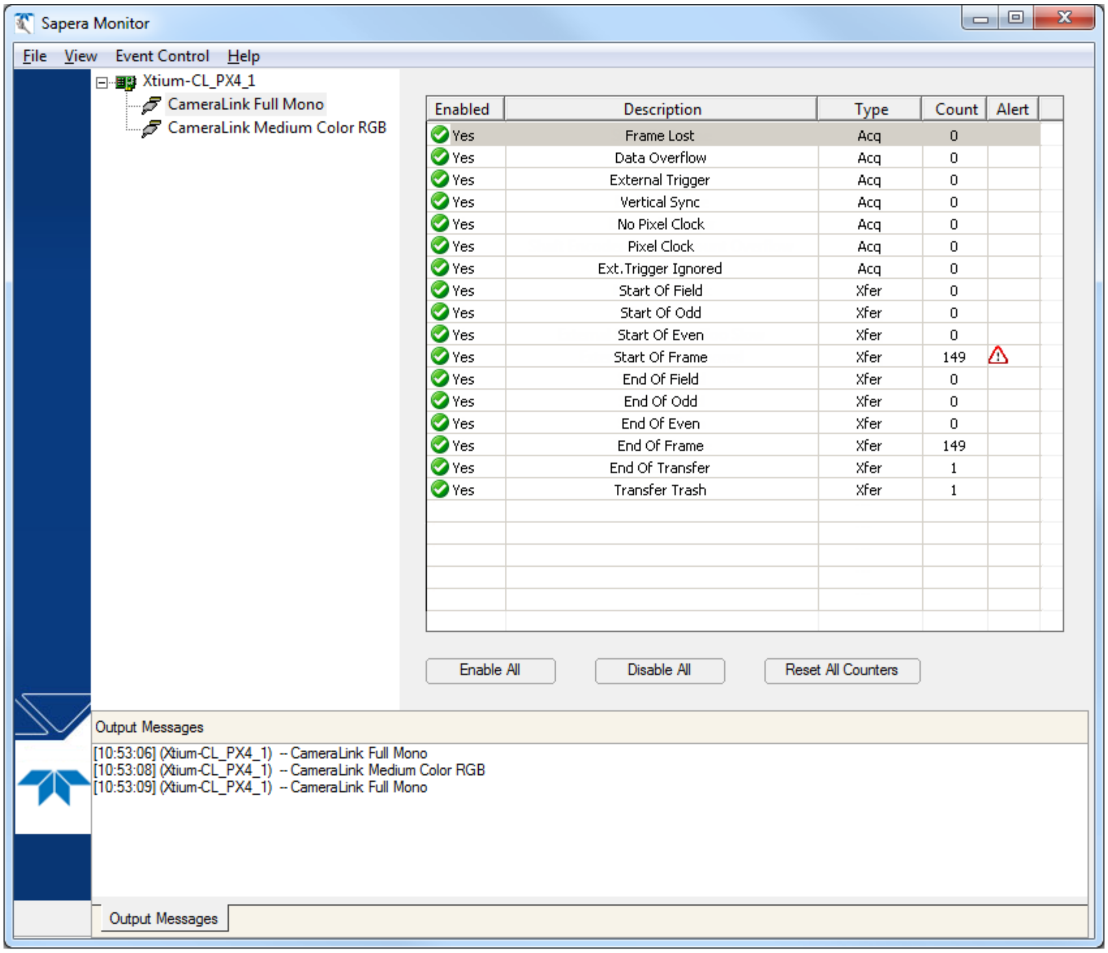
To launch the Sapera Monitor use the **Start•All Programs•Teledyne DALSA•Sapera LT•Tools•Sapera Monitor** menu shortcut.

⚠	Note: older driver versions of Teledyne DALSA devices may not support Sapera Monitor. Check the Teledyne DALSA website for updated drivers for your device that support Sapera Monitor. In addition, when using Teledyne DALSA USB3 Vision devices, you must start a Sapera application, such as CamExpert, that uses the device you want to monitor, before launching Sapera Monitor.

The Sapera Monitor Window

The Sapera Monitor Window is divided into three panes:

- Device directory tree: displays the available acquisition devices to monitor
- Event table: displays the available events to monitor for the selected device
- Output Messages pane: displays the messages generated by the selected monitored events.



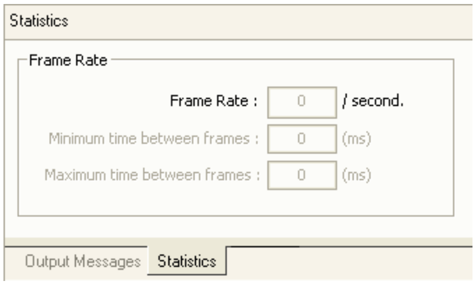
Enabled	Description	Type	Count	Alert
✓ Yes	Frame Lost	Acq	0	
✓ Yes	Data Overflow	Acq	0	
✓ Yes	External Trigger	Acq	0	
✓ Yes	Vertical Sync	Acq	0	
✓ Yes	No Pixel Clock	Acq	0	
✓ Yes	Pixel Clock	Acq	0	
✓ Yes	Ext.Trigger Ignored	Acq	0	
✓ Yes	Start Of Field	Xfer	0	
✓ Yes	Start Of Odd	Xfer	0	
✓ Yes	Start Of Even	Xfer	0	
✓ Yes	Start Of Frame	Xfer	149	⚠
✓ Yes	End Of Field	Xfer	0	
✓ Yes	End Of Odd	Xfer	0	
✓ Yes	End Of Even	Xfer	0	
✓ Yes	End Of Frame	Xfer	149	
✓ Yes	End Of Transfer	Xfer	1	
✓ Yes	Transfer Trash	Xfer	1	

Output Messages

```
[10:53:06] (Xtium-CL_PX4_1) -- CameraLink Full Mono
[10:53:08] (Xtium-CL_PX4_1) -- CameraLink Medium Color RGB
[10:53:09] (Xtium-CL_PX4_1) -- CameraLink Full Mono
```

Statistic Tab

Clicking on the Statistic tab displays various real-time acquisition statistics, such as the Frame Rate.



Frame Rate : 0 / second.

Minimum time between frames : 0 (ms)

Maximum time between frames : 0 (ms)

⚠	Note that different devices can support different statistics and not all devices support all statistics. In addition, these real-time acquisition statistics are not included in generated reports. Thus, depending on the selected device, the Statistics tab may not be available.

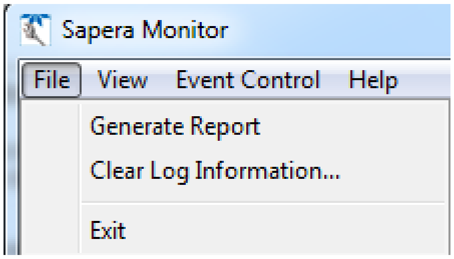
Sapera Monitor Menu Commands

The Sapera Monitor menu provides access to File, View, and Event Control commands.

File Menu Commands

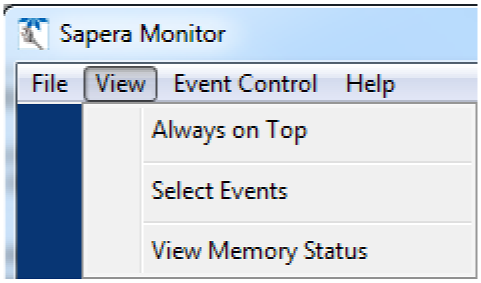
The **File•Generate Report** command generates a text file report that includes all event settings and messages included in the current Output Messages pane.

The **File•Clear Log Information** command clears the current Output Messages pane.

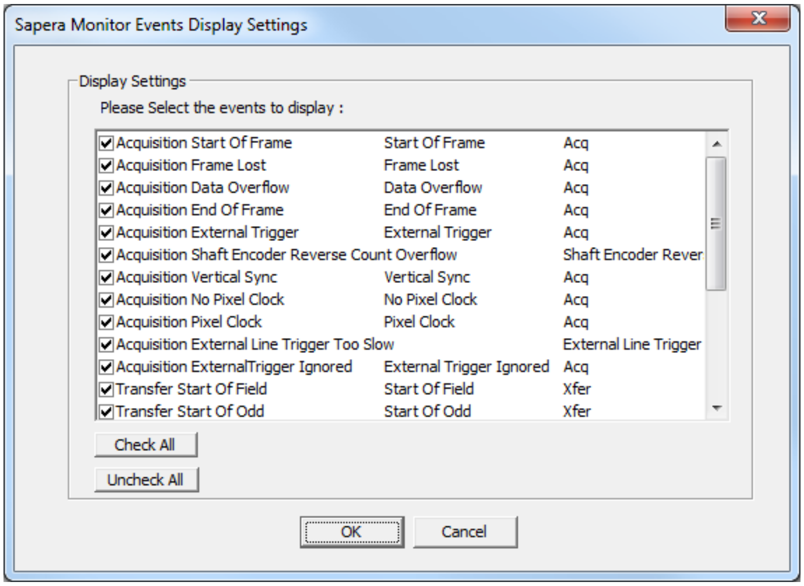


View Menu Commands

The **View•Always on Top** command displays the Sapera Monitor on top of any other windows that may be visible on the desktop.



The **View•Select Events** command opens the Sapera Monitor Events Display Settings dialog which allows you to specify the events to display in the Event table.



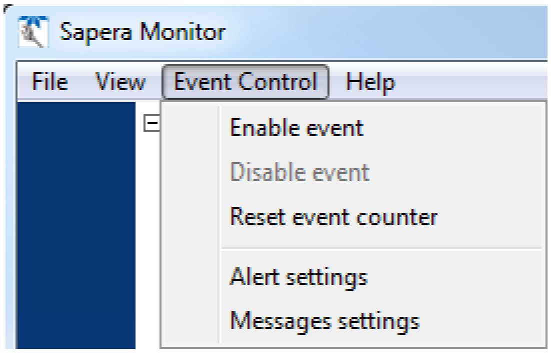
Please Select the events to display :

☑ Acquisition Start Of Frame	Start Of Frame	Acq
☑ Acquisition Frame Lost	Frame Lost	Acq
☑ Acquisition Data Overflow	Data Overflow	Acq
☑ Acquisition End Of Frame	End Of Frame	Acq
☑ Acquisition External Trigger	External Trigger	Acq
☑ Acquisition Shaft Encoder Reverse Count Overflow		Shaft Encoder Rever
☑ Acquisition Vertical Sync	Vertical Sync	Acq
☑ Acquisition No Pixel Clock	No Pixel Clock	Acq
☑ Acquisition Pixel Clock	Pixel Clock	Acq
☑ Acquisition External Line Trigger Too Slow		External Line Trigger
☑ Acquisition ExternalTrigger Ignored	External Trigger Ignored	Acq
☑ Transfer Start Of Field	Start Of Field	Xfer
☑ Transfer Start Of Odd	Start Of Odd	Xfer

Event Control Menu Commands

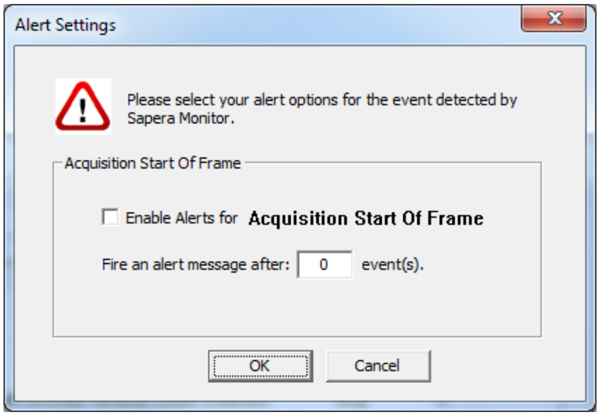
The **Event Control•Enable** and **Disable** commands enable or disable the currently selected event in Event table.

The **Event Control•Reset event count** command returns the event counter to zero for the currently selected event in Event table.



Alert Settings

You can specify the alert options for each available event using the Alert Settings command, available through the Event Control menu or by right-clicking on the selected event. The Alert Settings dialog allows you to enable or disable alerts for the event and to specify the number of events required to generate an alert.

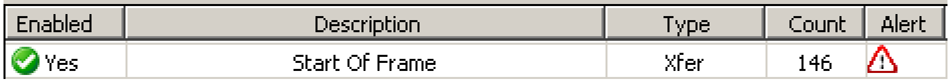


Please select your alert options for the event detected by Sapera Monitor.

☐ Enable Alerts for **Acquisition Start Of Frame**

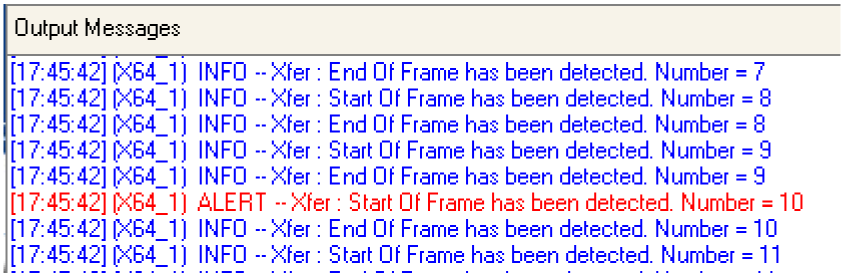
Fire an alert message after: 0 event(s).

When an alert is generated, the Alert icon is displayed in the Alert column of the event.



Enabled	Description	Type	Count	Alert
✓ Yes	Start Of Frame	Xfer	146	⚠

An alert message, in red, also appears in the Output Messages Settings pane.

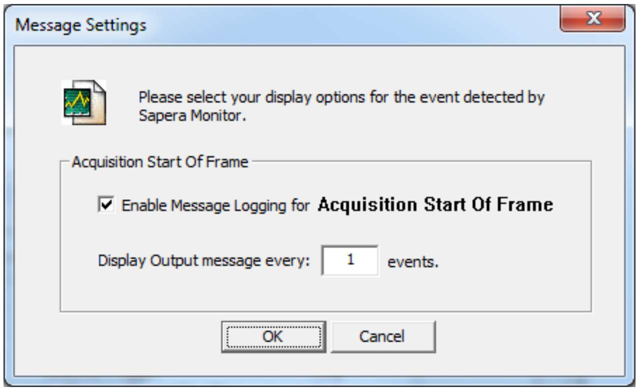


Output Messages

```
[17:45:42] (X64_1) INFO -- Xfer : End Of Frame has been detected. Number = 7
[17:45:42] (X64_1) INFO -- Xfer : Start Of Frame has been detected. Number = 8
[17:45:42] (X64_1) INFO -- Xfer : End Of Frame has been detected. Number = 8
[17:45:42] (X64_1) INFO -- Xfer : Start Of Frame has been detected. Number = 9
[17:45:42] (X64_1) INFO -- Xfer : End Of Frame has been detected. Number = 9
[17:45:42] (X64_1) ALERT -- Xfer : Start Of Frame has been detected. Number = 10
[17:45:42] (X64_1) INFO -- Xfer : End Of Frame has been detected. Number = 10
[17:45:42] (X64_1) INFO -- Xfer : Start Of Frame has been detected. Number = 11
```

Message Settings

You can specify the events to enable message logging and the number of events required to generate a log message using the Message Settings command, available through the Event Control menu or by right-clicking on the selected event. The log messages appear in the Output Message pane.



Message Settings

Please select your display options for the event detected by Sapera Monitor.

Acquisition Start Of Frame

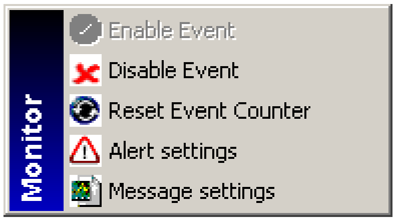
☑ Enable Message Logging for **Acquisition Start Of Frame**

Display Output message every: 1 events.

Using Sapera Monitor

To use the Sapera Monitor tool to monitor a device:

- Run a Sapera application, such as CamExpert, that uses the device.
- Launch the Sapera Monitor application
- In the Sapera Monitor Device directory tree, select the device.
- In the Sapera Monitor Event table, select the events to monitor. Double-clicking on the row containing the event toggles the Enabled status between ✓ Yes and ✗ No .
  Right-clicking on the event opens a context menu which provides easy access to commands to reset the event counter and modify the event Alert settings or Message settings.



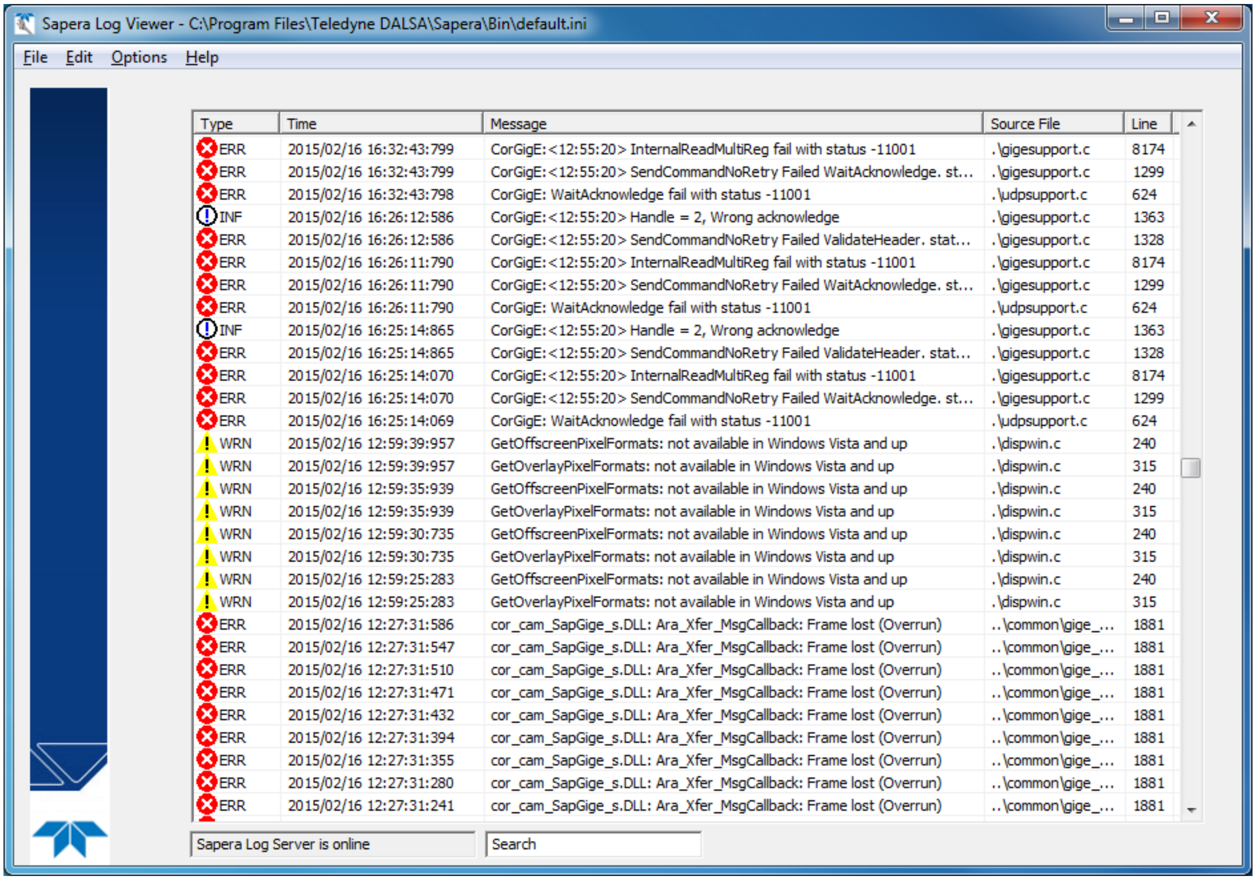
These commands are also available through the Event Control Menu.
Alternatively, you can use the Enable All and Disable All buttons to quickly clear or select events.

Sapera Log Viewer

The Sapera Log Viewer utility program included with Sapera LT provides an easy way to view error and other types of messages generated by Sapera LT applications and Teledyne DALSA hardware, such as cameras and frame grabbers. Typically, the Sapera Log Viewer application is used by technical support to troubleshoot software and hardware problems.

During development it is recommended to start the Sapera Log Viewer before your application and then let it run so it can be referred to any time a detailed error description is required. However, errors are also stored by a low-level service (running in the background), even if the utility is not running. Therefore, it is possible to run it only when a problem occurs with your application.

Refer to the utility's online help for more information on using the Sapera Log Viewer.


Sapera Log Viewer - C:\Program Files\Teledyne DALSA\Sapera\Bin\default.ini
File
Edit
Options
Help

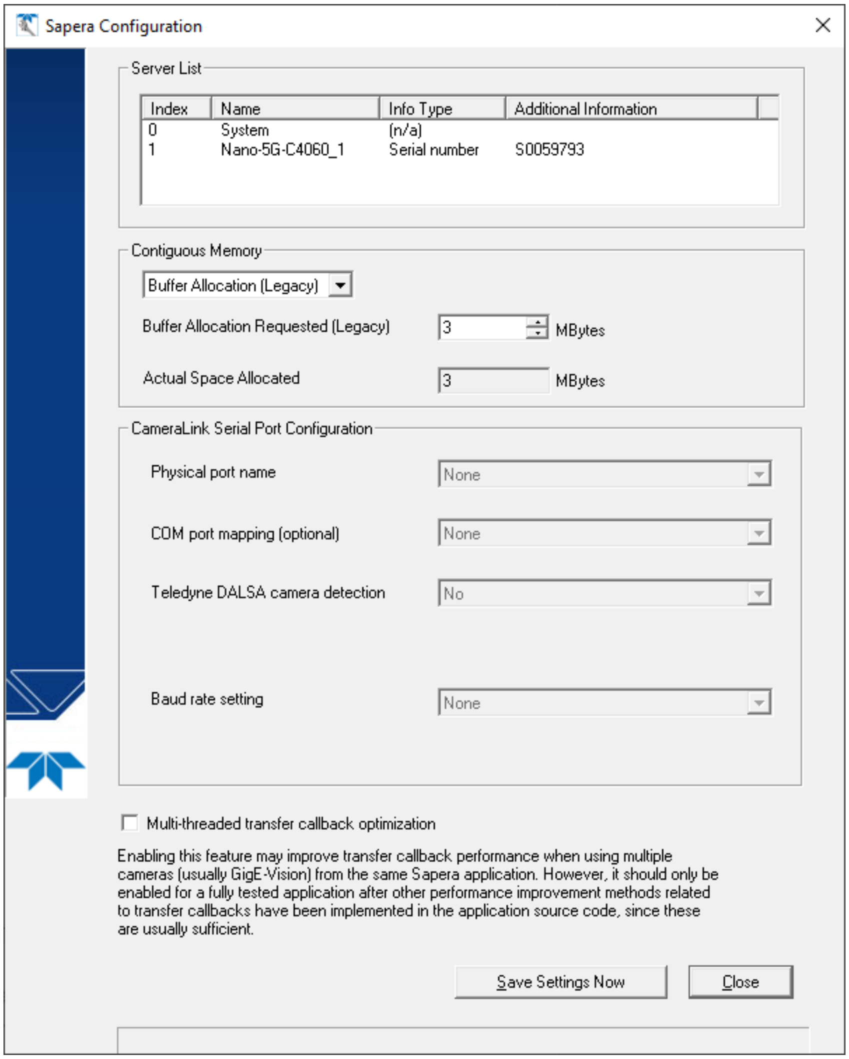
Type	Time	Message	Source File	Line
ERR	2015/02/16 16:32:43:799	CorGigE:<12:55:20> InternalReadMultiReg fail with status -11001	.\gigesupport.c	8174
ERR	2015/02/16 16:32:43:799	CorGigE:<12:55:20> SendCommandNoRetry Failed WaitAcknowledge. st...	.\gigesupport.c	1299
ERR	2015/02/16 16:32:43:798	CorGigE: WaitAcknowledge fail with status -11001	.\udpsupport.c	624
INF	2015/02/16 16:26:12:586	CorGigE:<12:55:20> Handle = 2, Wrong acknowledge	.\gigesupport.c	1363
ERR	2015/02/16 16:26:12:586	CorGigE:<12:55:20> SendCommandNoRetry Failed ValidateHeader. stat...	.\gigesupport.c	1328
ERR	2015/02/16 16:26:11:790	CorGigE:<12:55:20> InternalReadMultiReg fail with status -11001	.\gigesupport.c	8174
ERR	2015/02/16 16:26:11:790	CorGigE:<12:55:20> SendCommandNoRetry Failed WaitAcknowledge. st...	.\gigesupport.c	1299
ERR	2015/02/16 16:26:11:790	CorGigE: WaitAcknowledge fail with status -11001	.\udpsupport.c	624
INF	2015/02/16 16:25:14:865	CorGigE:<12:55:20> Handle = 2, Wrong acknowledge	.\gigesupport.c	1363
ERR	2015/02/16 16:25:14:865	CorGigE:<12:55:20> SendCommandNoRetry Failed ValidateHeader. stat...	.\gigesupport.c	1328
ERR	2015/02/16 16:25:14:070	CorGigE:<12:55:20> InternalReadMultiReg fail with status -11001	.\gigesupport.c	8174
ERR	2015/02/16 16:25:14:070	CorGigE:<12:55:20> SendCommandNoRetry Failed WaitAcknowledge. st...	.\gigesupport.c	1299
ERR	2015/02/16 16:25:14:069	CorGigE: WaitAcknowledge fail with status -11001	.\udpsupport.c	624
WRN	2015/02/16 12:59:39:957	GetOffscreenPixelFormats: not available in Windows Vista and up	.\dispwin.c	240
WRN	2015/02/16 12:59:39:957	GetOverlayPixelFormats: not available in Windows Vista and up	.\dispwin.c	315
WRN	2015/02/16 12:59:35:939	GetOffscreenPixelFormats: not available in Windows Vista and up	.\dispwin.c	240
WRN	2015/02/16 12:59:35:939	GetOverlayPixelFormats: not available in Windows Vista and up	.\dispwin.c	315
WRN	2015/02/16 12:59:30:735	GetOffscreenPixelFormats: not available in Windows Vista and up	.\dispwin.c	240
WRN	2015/02/16 12:59:30:735	GetOverlayPixelFormats: not available in Windows Vista and up	.\dispwin.c	315
WRN	2015/02/16 12:59:25:283	GetOffscreenPixelFormats: not available in Windows Vista and up	.\dispwin.c	240
WRN	2015/02/16 12:59:25:283	GetOverlayPixelFormats: not available in Windows Vista and up	.\dispwin.c	315
ERR	2015/02/16 12:27:31:586	cor_cam_SapGige_s.DLL: Ara_Xfer_MsgCallback: Frame lost (Overrun)	..\common\gige_...	1881
ERR	2015/02/16 12:27:31:547	cor_cam_SapGige_s.DLL: Ara_Xfer_MsgCallback: Frame lost (Overrun)	..\common\gige_...	1881
ERR	2015/02/16 12:27:31:510	cor_cam_SapGige_s.DLL: Ara_Xfer_MsgCallback: Frame lost (Overrun)	..\common\gige_...	1881
ERR	2015/02/16 12:27:31:471	cor_cam_SapGige_s.DLL: Ara_Xfer_MsgCallback: Frame lost (Overrun)	..\common\gige_...	1881
ERR	2015/02/16 12:27:31:432	cor_cam_SapGige_s.DLL: Ara_Xfer_MsgCallback: Frame lost (Overrun)	..\common\gige_...	1881
ERR	2015/02/16 12:27:31:394	cor_cam_SapGige_s.DLL: Ara_Xfer_MsgCallback: Frame lost (Overrun)	..\common\gige_...	1881
ERR	2015/02/16 12:27:31:355	cor_cam_SapGige_s.DLL: Ara_Xfer_MsgCallback: Frame lost (Overrun)	..\common\gige_...	1881
ERR	2015/02/16 12:27:31:280	cor_cam_SapGige_s.DLL: Ara_Xfer_MsgCallback: Frame lost (Overrun)	..\common\gige_...	1881
ERR	2015/02/16 12:27:31:241	cor_cam_SapGige_s.DLL: Ara_Xfer_MsgCallback: Frame lost (Overrun)	..\common\gige_...	1881

Sapera Log Server is online

Search

Sapera Configuration Utility

The **Sapera Configuration** program (**SapConf.exe)** allows you to see all the Sapera LT-compatible devices present within your system, together with their respective serial numbers. It can also adjust the amount of contiguous memory to be allocated at boot-time. After activating this program, it displays all the servers related to the installed devices as shown in the figure below (64-bit version shown).



Sapera Configuration

Server List

Index	Name	Info Type	Additional Information
0	System	(n/a)	
1	Nano-5G-C4060_1	Serial number	S0059793

Contiguous Memory

Buffer Allocation (Legacy)

Buffer Allocation Requested (Legacy): 3 MBytes

Actual Space Allocated: 3 MBytes

CameraLink Serial Port Configuration

Physical port name: None

COM port mapping (optional): None

Teledyne DALSA camera detection: No

Baud rate setting: None

□ Multi-threaded transfer callback optimization

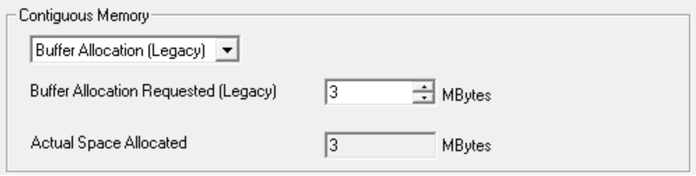
Enabling this feature may improve transfer callback performance when using multiple cameras (usually GigE-Vision) from the same Sapera application. However, it should only be enabled for a fully tested application after other performance improvement methods related to transfer callbacks have been implemented in the application source code, since these are usually sufficient.

- The **System** entry represents the system server. It corresponds to the host machine (your computer) and is the only server that should always be present. The other servers correspond to the devices present within the system.

Configuring Contiguous Memory

The **Contiguous Memory** section lets you specify the total amount of contiguous memory to be reserved for allocating **buffers** and **messages**. This memory is used by frame grabbers to allocate DMA tables; however, a certain amount of contiguous memory is required for Sapera LT buffer descriptors and 1MB for every 3000 buffers should be allocated.

Contiguous memory is reserved at boot time for the allocation of dynamic resources used for frame buffer management such as scatter-gather list plus other kernel needs. Adjust this value higher if your application generates any out-of-memory error while allocating host frame buffers or when connecting the buffers via a transfer object.



Contiguous Memory

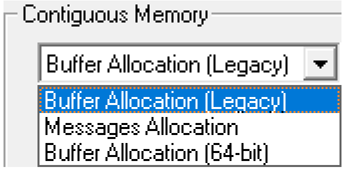
Buffer Allocation (Legacy)

Buffer Allocation Requested (Legacy) 3 MBytes

Actual Space Allocated 3 MBytes

Note: All Sapera LT demos and examples **do not use contiguous memory** other than the 1MB per 3000 buffers required for buffer descriptors. Therefore, you should not modify these settings unless your application requires contiguous memory.

The drop-down list specifies the memory type to allocate.



Contiguous Memory

Buffer Allocation (Legacy)

Buffer Allocation (Legacy): Contiguous memory is allocated in the 1st 4GB of host memory. In practice, not all 4GB is available since it is also used by other hardware resources; the actual amount available can range from 2GB to 3.5GB. RAM amounts greater than 4GB are unused since there is no address space to map it to. The amount required is determined by the number of buffers only (DMA tables do not apply for USB3 Vision cameras). 1MB for every 3000 buffers should be allocated.

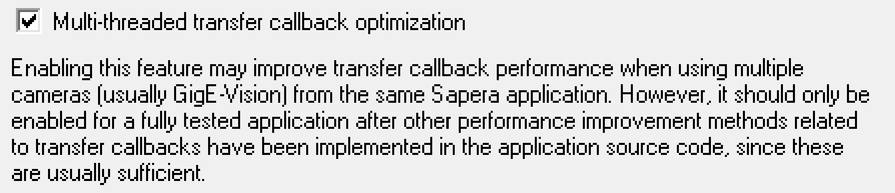
Buffer Allocation (64-bit): This does not apply to USB3 Vision cameras; used for frame grabber DMA.

Messages Allocation: This does not apply to USB3 Vision cameras

The **Requested** value displays what was requested.

The **Allocated** value displays the amount of contiguous memory that was allocated successfully.

Multi-Threaded Transfer Callback Optimization



☑ Multi-threaded transfer callback optimization

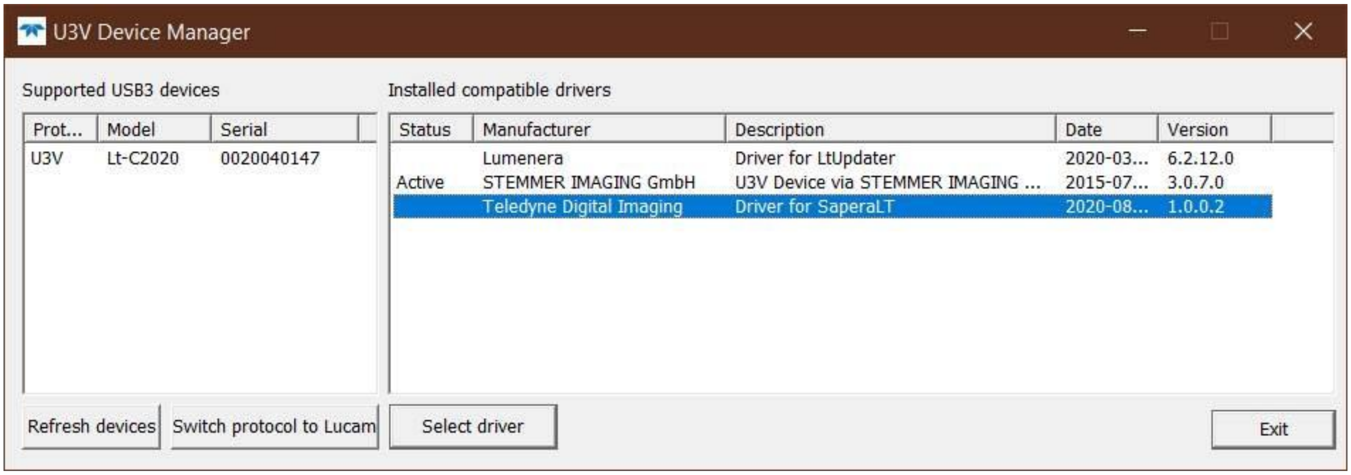
Enabling this feature may improve transfer callback performance when using multiple cameras (usually GigE-Vision) from the same Sapera application. However, it should only be enabled for a fully tested application after other performance improvement methods related to transfer callbacks have been implemented in the application source code, since these are usually sufficient.

Usage Notes When Writing Sapera Applications

- Always disable this option (the default) while developing and thoroughly testing the application, especially making sure that appropriate robustness standards are met.
- If the application does not meet performance requirements, all the known performance improvements which can be implemented in application code must be tried (for example, limiting operations as much as possible in the transfer callback function).
- If performance requirements are still not met, and there is only one camera per running instance of the application, then still leave disabled since it provides no performance benefit.
- Only consider enabling if performance requirements are not met with multiple cameras in the same running instance of the application.
- If enabling does not improve performance, then disable it.
- If enabling improves performance, the application must be once again thoroughly tested to prove that it still meets the same robustness requirements as before.

U3V Device Manager

The U3V Device Manager allows you to view all connected USB3 Vision cameras on the system and select the required USB3 Vision driver to use. It also allows you to switch the transport layer protocol to the Lucam protocol if required.



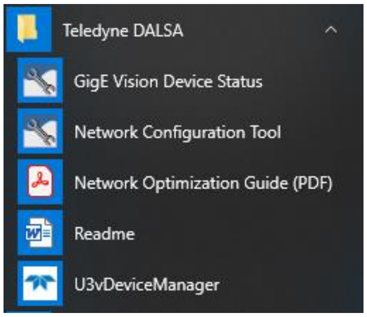
Supported USB3 devices

Prot...	Model	Serial
U3V	Lt-C2020	0020040147

Installed compatible drivers

Status	Manufacturer	Description	Date	Version
	Lumenera	Driver for LtUpdater	2020-03...	6.2.12.0
Active	STEMMER IMAGING GmbH	U3V Device via STEMMER IMAGING ...	2015-07...	3.0.7.0
	Teledyne Digital Imaging	Driver for SaperaLT	2020-08...	1.0.0.2

The U3V Device Manager is available through the Windows Start menu under Teledyne DALSA.


Teledyne DALSA
GigE Vision Device Status
Network Configuration Tool
Network Optimization Guide (PDF)
Readme
U3vDeviceManager

The application executable is located in the Teledyne DALSA\USB Interface\Bin directory.

Demos and Examples

Several generic demos and examples are available for both Sapera ++ and Sapera .NET. Complete source code is provided for projects in Microsoft Visual Studio 2010 to 2017.

Source code for Sapera LT ++ based demos and examples can now be compiled as Unicode instead of ANSI. Project files provided by Sapera for Visual Studio 2012 now support both character sets. However, project files for earlier versions of Visual Studio still support ANSI only. Project files for Visual Studio 2013/2015/2017 support Unicode only.

If your application requires a user interface, Sapera LT includes the GUI classes used by many of the demos to create commonly used dialog boxes. The GUI classes include a set of Microsoft® Foundation Classes (MFC) based dialog boxes designed to implement some of the most commonly used tasks for Sapera LT applications, such as loading an acquisition configuration file. They, however, do not constitute an official API. Rather, they are provided 'as is' with source code so that you may modify them at your discretion. For more information on these classes refer to the Sapera LT GUI Class Reference manual.

Certain device driver installations provide other demos and examples that demonstrate the specific usages and capabilities of the device. Refer to a specific device user's manual for further details.

Demo Source Code

Several demo programs are available with Sapera. They are more complete applications than the supplied examples. There are demos that cover Sapera LT ++ and Sapera .NET.

The demos main purpose is to provide the user with a starting application that can be modified in order to become the user's end application.

The Sapera LT ++ and Sapera LT .NET demo source code for the supported compilers are found in the following directory:

- `Sapera\Demos`

Projects are also provided to allow you to recompile all the demos in a batch, together with the Sapera LT ++ GUI Classes.

Example Source Code

Several example programs are available within Sapera. They are essentially basic applications demonstrating simple tasks like grabbing an image and loading an image file from the disk.

The main purpose of the examples is to provide the user with code samples that can be easily extracted and integrated into an application. Examples cover both Sapera LT ++ and Sapera .NET.

The Sapera LT ++ and .NET example source code for the supported compilers are found in the following directory:

- `Sapera\Examples`

Projects are also provided to allow you to recompile all the examples in a batch.

Demos and Examples for USB3 Vision Devices

The following demo programs and corresponding source code are available:

Demo Name	Description
GigEBayerDemo	This program demonstrates the Bayer conversion functionality included in Sapera LT with a USB3 Vision camera. It allows you to acquire images either in continuous or in one-shot mode, while adjusting the acquisition parameters. It includes interactive control of Bayer conversion parameters. You may optionally apply Bayer filtering to acquired images. The minimum requirement to run this demo is a Sapera-compatible USB Vision color camera that output Bayer format. Note, the SapBayer class has been deprecated and replaced by the SapColorConversion class.
GigECameraDemo	This program demonstrates how to acquire images from a USB3 Vision camera. The demo either loads a configuration file (previously generated by CamExpert) or uses the camera defaults. The minimum requirement to run this demo is a Sapera-compatible USB3 Vision camera. The supplied executable is built using Sapera LT ++ plus the MFC library under Visual Studio 2013.
GigESeqGrabDemo	This program demonstrates how to grab a sequence of images from a USB3 Vision camera into memory and then display them. The program allows you to record several images and save AVI files, plus load those AVI files for playback. Each image is stored in its own buffer and can be reviewed. A small number of images are allocated by default but can be increased using the buffer options inside the demo. The minimum requirement to run this demo is a Sapera-compatible USB3 Vision camera. The supplied executable is built using Sapera LT ++ plus the MFC library under Visual Studio 2013.

The following compiled console examples and source code are available:

Example Name	Description
CameraEvents	Shows how to list all the available events with SapAcqDevice. Using the registering and unregistering callback mechanism, it shows also how to track when a specific event occurs.
CameraFeatures	Shows how to enumerate available features on a camera. It also shows how to retrieve feature specific information (for example, access mode), and how to change feature values.
CameraFiles	Shows how to upload/download files for USB3 Vision cameras that support file access such as firmware upload and LUTs.
CameraFirmwareUpdate	Shows how to update firmware for USB3 Vision cameras that support file access, allowing automatic firmware updates at the application level. (GenCP and CLHS cameras are also supported by this example.)
FindCamera	Shows how to list all detected cameras when more than one camera is present, listing them by username, serial number, model name or server name. By uncommenting a part of code, you will be able to change the user defined name of the camera.
GrabCPP	Grab Console. Shows how to grab an image from a selected USB3 Vision camera into a Sapera buffer and then display it. The buffer is created according to the camera settings. This example is named Grab in .NET.

Generic Sapera LT Examples

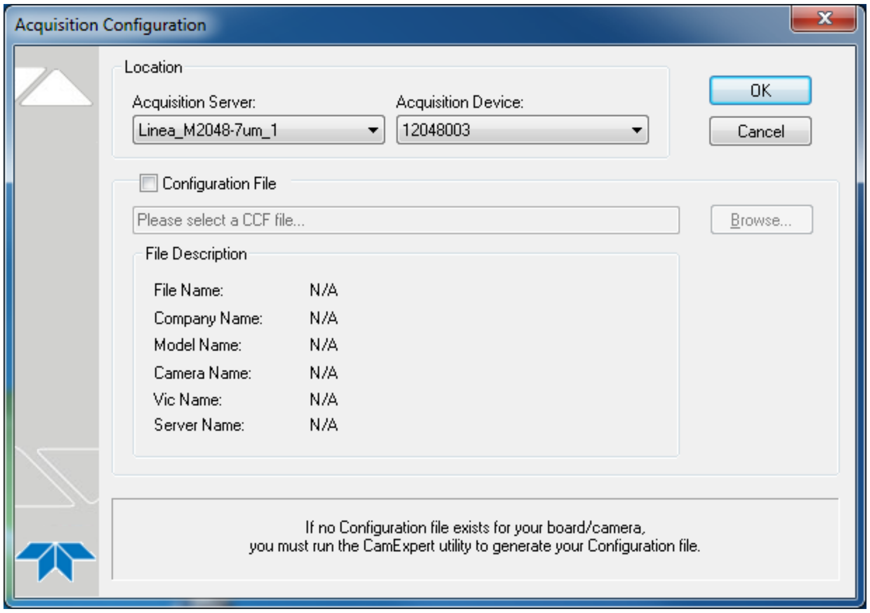
The generic Sapera LT example do not require an acquisition device. The following compiled console examples and source code is available:

Example Name	Description
ColorSplit	Shows how to split and merge color images into single monochrome components. An RGB image is loaded, split into three monochrome components, then a simple processing is applied to the three components before they are merged back to RGB as output.
FileLoadCPP	Shows how to load an image file from the disk into a Sapera buffer and then display it. The buffer is created according to the image file properties. One of several images (monochrome, RGB, or YUV) can be selected for loading. This example is named FileLoad in .NET.

Acquiring with GigE Vision / USB3 Vision Camera Demo

The Sapera LT GigE Vision / USB3 Vision Camera Demo program allows you to grab and display a live image in a host buffer. This demo is a good starting point to verify that your camera and frame grabber are properly installed.

The following dialog box appears when starting Grab Demo:



Location

Acquisition Server: Linea_M2048-7um_1

Acquisition Device: 12048003

□ Configuration File

Please select a CCF file...

File Description

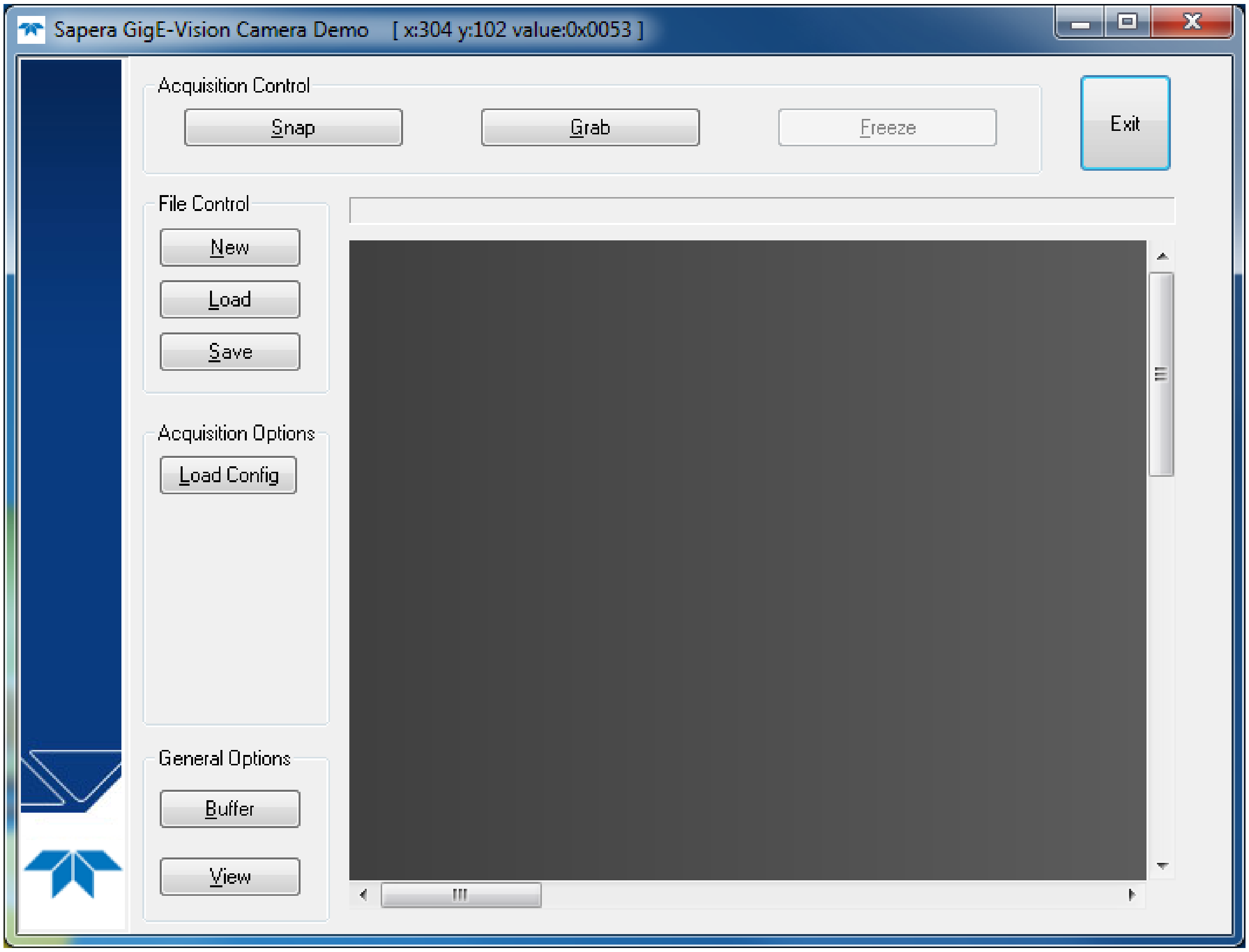
File Name: N/A
Company Name: N/A
Model Name: N/A
Camera Name: N/A
Vic Name: N/A
Server Name: N/A

If no Configuration file exists for your board/camera,
you must run the CamExpert utility to generate your Configuration file.

You must select the **Acquisition Server and the Acquisition Device**. The first one corresponds to the device you want to grab from; the second represents the acquisition device on this board (some devices may have more than one).

You can then select an acquisition configuration file (CCF File) compatible with your camera from the list of available files or use the camera default values without loading a .ccf file. CamExpert must be used to generate CCF files (for example, external trigger, cropping window, and so forth).

Click OK to start the demo.

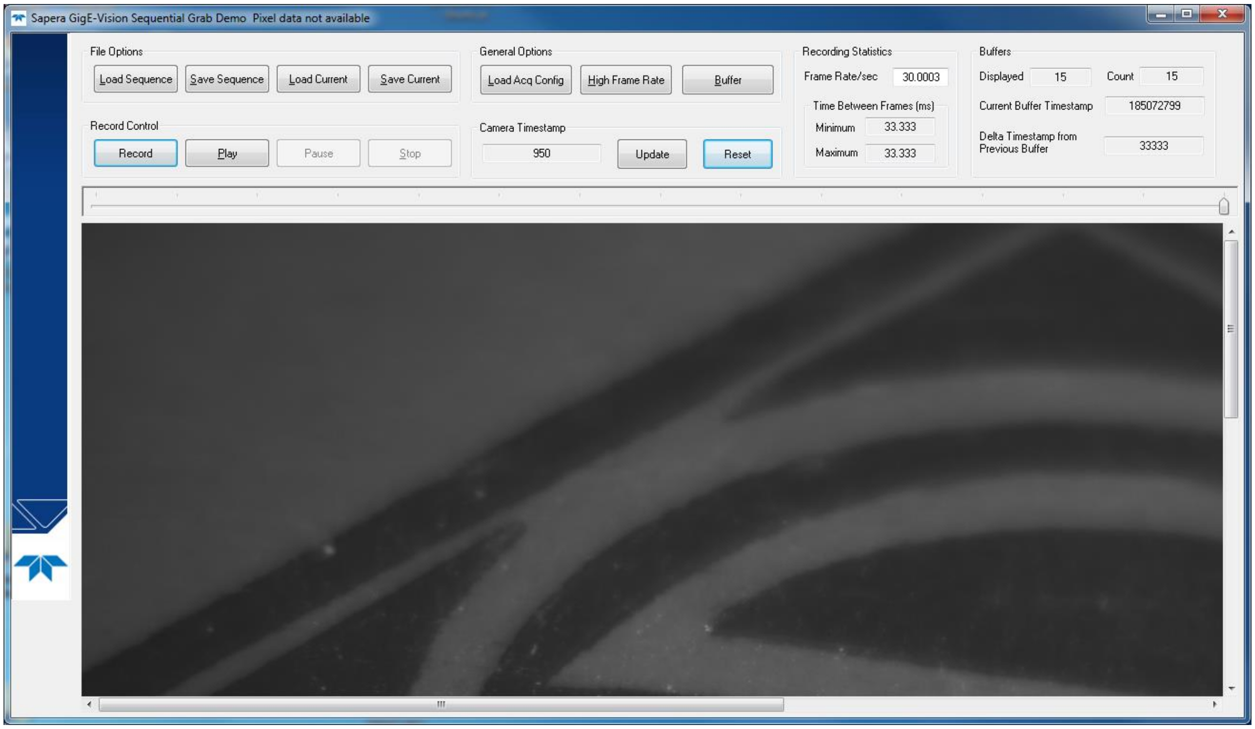


By using GigE Vision / USB3 Vision Camera Demo you can now:

- Control the acquisition using the **Snap**, **Grab**, and **Freeze** buttons.
- Load/save images from/to disks using the **Load** and **Save** buttons.
- Reload the CCF file using the **Load Config** button (this overwrites all the parameters modified in step 3).
- The **Buffer** button allows you to change the number of buffers used for internal cycling and the type of buffer used (contiguous, scatter-gather, off-screen, or overlay).
- The **View** button allows you to adjust the scaling and the select the bit range to display when grabbing images with formats greater than 8-bits.

Using the GigE / USB3 Vision Sequential Grab Demo

The sequential grab demo allows you to grab a sequence of images and save them as a .AVI file.



Sapera GigE-Vision Sequential Grab Demo Pixel data not available

Recording Statistics

Frame Rate/sec: 30.0003

Time Between Frames (ms)

Minimum: 33.333

Maximum: 33.333

Camera Timestamp: 950

Buffers

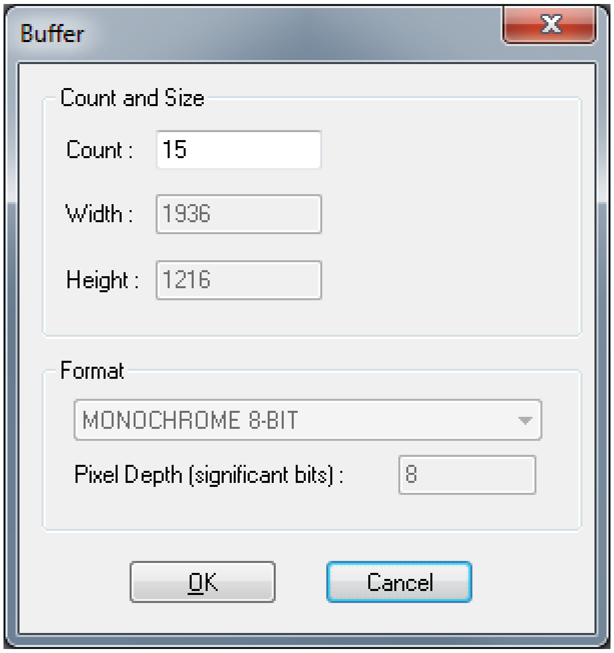
Displayed: 15

Count: 15

Current Buffer Timestamp: 185072799

Delta Timestamp from Previous Buffer: 33333

The user can set the number of images in the sequence. For high frame rate applications, the number of frames that generate transfer events can be increased to reduce the CPU overhead.



Buffer

Count and Size

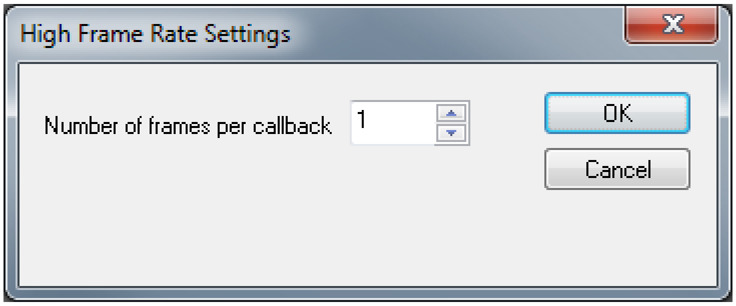
Count : 15

Width : 1936

Height : 1216

Format: MONOCHROME 8-BIT

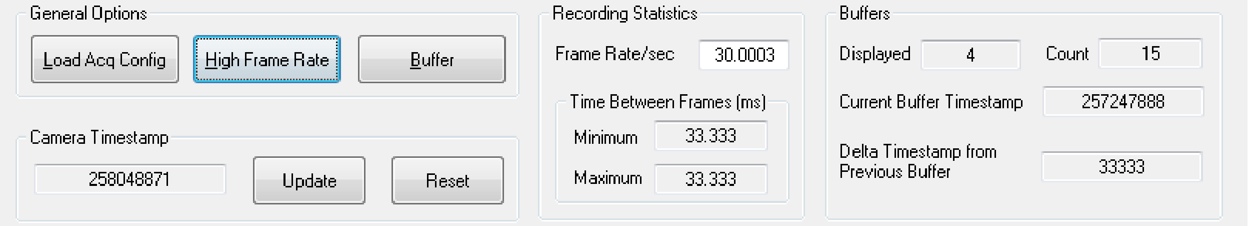
Pixel Depth (significant bits) : 8



High Frame Rate Settings

Number of frames per callback: 1

The demo also displays the timestamps generated by the camera and saved in the image buffer. These timestamps can be analyzed to ensure that no images are lost. In addition, clicking the camera timestamp Reset button provides an easy way to calculate the round trip required for a function to be sent to the camera and the response received, providing a means to evaluate the network speed.



Camera Timestamp: 258048871

Frame Rate/sec: 30.0003

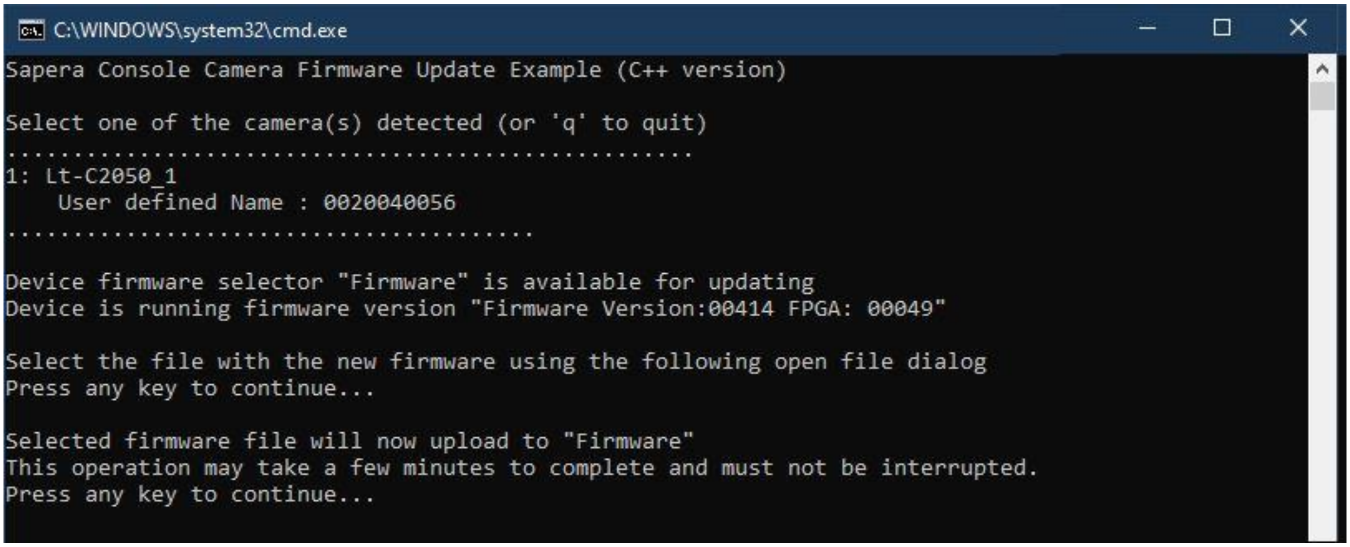
Time Between Frames (ms)
Minimum: 33.333
Maximum: 33.333

Displayed: 4
Count: 15
Current Buffer Timestamp: 257247888
Delta Timestamp from Previous Buffer: 33333

Note: The timestamp timebase is device-dependent. For example, if a device only supports nanosecond timebase, features in milliseconds may not function.

Camera Firmware Update Example

For Teledyne DALSA GigE cameras that support GenICam file access, the camera firmware update example demonstrates how to implement camera firmware updates and validation at the application level. This can be useful to ensure that cameras are equipped with the required firmware version certified for your system. The firmware version on the camera can be read (using standard SNFC features), validated and updated as required by your application.



```
Sapera Console Camera Firmware Update Example (C++ version)

Select one of the camera(s) detected (or 'q' to quit)
....................................................
1: Lt-C2050_1
    User defined Name : 0020040056
........................................

Device firmware selector "Firmware" is available for updating
Device is running firmware version "Firmware Version:00414 FPGA: 00049"

Select the file with the new firmware using the following open file dialog
Press any key to continue...

Selected firmware file will now upload to "Firmware"
This operation may take a few minutes to complete and must not be interrupted.
Press any key to continue...
```

Note, all Teledyne Lumenera USB3 Vision cameras are designed with failsafe measures for the firmware update process. If, for any reason, the firmware update is interrupted, the camera will always revert to the previously installed firmware.

Troubleshooting

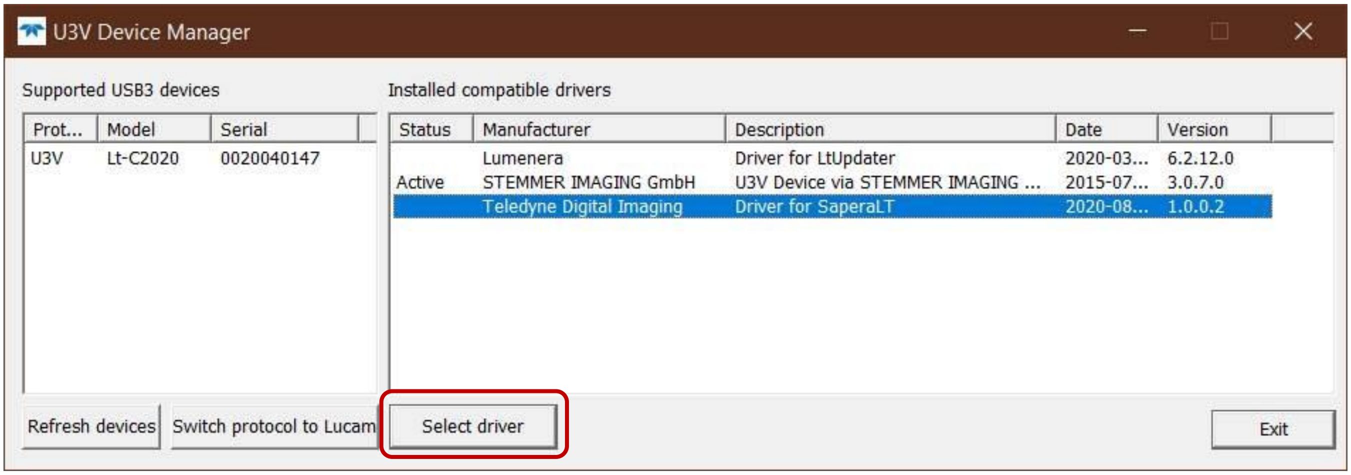
For more troubleshooting information refer to the camera documentation.

No Device Detected in CamExpert

If a properly connected and powered Teledyne Lumenera USB3 Vision camera is not detected in CamExpert, it is generally due to the presence of another USB3 Vision driver that is active on the system.

To set the Sapera LT USB3 Vision driver as the active driver use the U3V Device Manager:

- Select the entry for Teledyne Digital Imaging and click **Select Driver.**



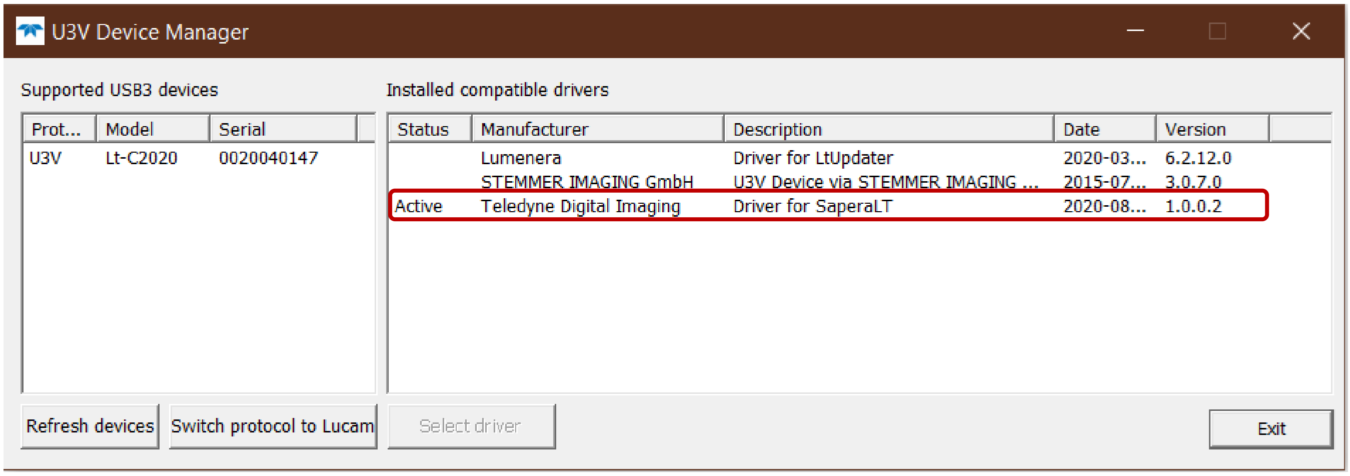
Supported USB3 devices

Prot...	Model	Serial
U3V	Lt-C2020	0020040147

Installed compatible drivers

Status	Manufacturer	Description	Date	Version
	Lumenera	Driver for LtUpdater	2020-03...	6.2.12.0
Active	STEMMER IMAGING GmbH	U3V Device via STEMMER IMAGING ...	2015-07...	3.0.7.0
	Teledyne Digital Imaging	Driver for SaperaLT	2020-08...	1.0.0.2

The Status will now show the Teledyne Digital Imaging driver as Active.



Supported USB3 devices

Prot...	Model	Serial
U3V	Lt-C2020	0020040147

Installed compatible drivers

Status	Manufacturer	Description	Date	Version
	Lumenera	Driver for LtUpdater	2020-03...	6.2.12.0
	STEMMER IMAGING GmbH	U3V Device via STEMMER IMAGING ...	2015-07...	3.0.7.0
Active	Teledyne Digital Imaging	Driver for SaperaLT	2020-08...	1.0.0.2

USB3 Vision cameras will now be accessibly in CamExpert.

Appendix A: File Locations

The table below lists the different file groups and locations under the Teledyne DALSA installation directory:

Directory	**Contents**
USB Interface	Sapera USB3 Vision Interface
USB Interface\Bin	U3V Device Manager utility program
Sapera	Sapera LT Readme and version history documents
	Third-party software licenses (for example, GenICam)
Sapera\Bin	Utility programs
Sapera\CamExpert	CamExpert frame-grabber and camera configuration utility
Sapera\CamFiles	Camera configuration files for frame grabbers
Sapera\Classes	Sapera LT ++ header files (Basic and GUI Classes) Sapera LT ++ source code (GUI Classes only)
Sapera\Components\NET	.NET classes
Sapera\Demos	Source code for GUI-based demo applications
Sapera\Demos\Binaries	Executable files for GUI-based demo applications
Sapera\Examples	Source code for console-based demo applications
Sapera\Examples\Binaries	Executable files for console-based demo applications
Sapera\Help	On-line documentation (Compiled HTML and PDF formats)
Sapera\Help\VisualStudio	Integrated C++ and .NET help for Visual Studio 2010, 2012, 2013, 2015, and 2017
	Images files used by demos and examples
Sapera\Include	Header files for C libraries
Sapera\Lib	Import libraries for Microsoft and Borland (Embarcadero) compilers
Windows\system32 directory	Dynamic Link Libraries (DLLs)
Windows\system32\drivers directory	Device drivers

Appendix B: Supported Image Formats

The table below lists the image formats currently supported by the Teledyne DALSA USB3 Vision Interface.

PFNC Format	Sapera Data Format
Mono8 BayerGR8 BayerRG8 BayerGB8 BayerBG8	SapFormat.Mono8
Mono10 BayerGR10 BayerRG10 BayerGB10 BayerBG10 Mono12 BayerGR12 BayerRG12 BayerGB12 BayerBG12 Mono16	SapFormat.Mono16

Contact Information

The following sections provide sales and technical support contact information.

Sales Information

Visit our web site:	www.teledynedalsa.com
Email:	mailto:info@teledynedalsa.com

Technical Support

Submit any support question or request via our web site:

Technical support form via our web page:	
Support requests for imaging product installations	http://www.teledynedalsa.com/en/support/options/
Support requests for imaging applications	
Camera support information	
Product literature and driver updates	

When encountering hardware or software problems, please have the following documents included in your support request:

- The Sapera Log Viewer .txt file

⚠	Note, the Sapera Log Viewer and PCI Diagnostic tools are available from the Windows start menu shortcut **Start•All Programs•Teledyne DALSA•Sapera LT**.