Baumer

Passion for Sensors

Release Notes
Baumer GAPI SDK

Contents

1 Release Notes Baumer GAPI SDK v2.12

1.1 General

- Information
  - Baumer GAPI package no longer include the CameraExplorer directly. It will be shipped in a separate package which can be downloaded from our website. But for convenience it is still included in the Baumer GAPI installer for Windows® users.
  - The structure of the Windows® SDK is changed to be more unified to the other platforms. Furthermore the SDK is splitted into language specific packages.
  - Rivermax support added for Windows® 64bit.
  - Support for color polarisation sensors added.
    - Support for Linux® 64bit version on x86 platform.
    - Support for Linux® ARM® 64bit (aarch64) platform.
    - Support for Linux® ARM® 32bit (armhf) platform.
    - Support for Linux® 32bit version on x86 platform on request only.

1.2 Consumer

- Enhancements
  - GenICam™ GenAPI reference implementation updated to version 3.3.
  - Simplyfication of the C++ API for access to list objects. Now the iterator classes allow direct access to the BGAPI objects. Index based access introduced. (see C++ API table for details and SDK examples).
  - C++ API behavior changes

C++ Function	new behavior	previous behavior
NodeMap::iterator:: operator->()	Returns directly a Node pointer. The previous access via 'first'/'second.second' doesn't work anymore.	Returns a _pairnm struct which includes an Id and a _pairn struct, which includes the Node pointer via 'first'/'second.second' elements.
NodeMap::iterator:: operator*()	Returns directly a Node pointer. The previous access via 'first'/'second.second' doesn't work anymore.	Returns a _pairnm struct which includes an Id and a _pairn struct, which includes the Node pointer via 'first'/'second.second' elements.
SystemList::iterator:: operator->()	Returns directly a System pointer. The previous access via 'first'/'second' still works.	Returns a _pairs struct which includes an Id and the System pointer via 'first'/'second' elements.
SystemList::iterator:: operator*()	Returns directly a System pointer. The previous access via 'first'/'second' still works.	Returns a _pairs struct which includes an Id and the System pointer via 'first'/'second' elements.
InterfaceList::iterator:: operator->()	Returns directly a Interface pointer. The previous access via 'first'/'second' still works.	Returns a _pairi struct which includes an Id and the Interface pointer via 'first'/'second' elements.

InterfaceList::iterator:: operator*()	Returns directly a Interface pointer. The previous access via 'first'/'second' still works.	Returns a _pairi struct which includes an Id and the Interface pointer via 'first'/'second' elements.
DeviceList::iterator:: operator->()	Returns directly a Device pointer. The previous access via 'first'/'second' still works.	Returns a _paird struct which includes an Id and the Device pointer via 'first'/'second' elements.
DeviceList::iterator:: operator*()	Returns directly a Device pointer. The previous access via 'first'/'second' still works.	Returns a _paird struct which includes an Id and the Device pointer via 'first'/'second' elements.
DataStreamList::iterator:: operator->()	Returns directly a DataStream pointer. The previous access via 'first'/'second' still works.	Returns a _pairs struct which includes an Id and the DataStream pointer via 'first'/'second' elements.
DataStreamList::iterator:: operator*()	Returns directly a DataStream pointer. The previous access via 'first'/'second' still works.	Returns a _pairs struct which includes an Id and the DataStream pointer via 'first'/'second' elements.
BufferList::iterator:: operator->()	Returns directly a Buffer pointer. The previous access via 'first'/'second' still works.	Returns a _pairb struct which includes an Id and the Buffer pointer via 'first'/'second' elements.
BufferList::iterator:: operator*()	Returns directly a Buffer pointer. The previous access via 'first'/'second' still works.	Returns a _pairb struct which includes an Id and the Buffer pointer via 'first'/'second' elements.

- C++ API changes

Added C++ Functions	**Description**
Device::SetSerialPort()	Use this function to made a virtual connection from the camera uart port to a com port on the host pc. Please refer to 505_SerialPort example.
Polarizer::IsPolarized()	Used to check if the Device has a color or mono polarisation sensor.
Polarizer::EnableSingleFormat()	Used to enable only one format from the Polarizer::Formats enumeration.
Polarizer::GetByPredefinedImage()	Use the polarizer with a already defined image.
Polarizer::IsFormatAvailable()	Used to query supported Polarizer::Formats.
Image::GetLength()	This function delivers the image data size.
SystemList::operator[bo_uint64]()	Allows index based access to the list objects.
InterfaceList::operator[bo_uint64]()	Allows index based access to the list objects.
DeviceList::operator[bo_uint64]()	Allows index based access to the list objects.
DataStreamList::operator[bo_uint64]()	Allows index based access to the list objects.

BufferList::operator[bo_uint64]()	Allows index based access to the list objects.
NodeMap::operator[bo_uint64]()	Allows index based access to the list objects.

- C API changes

Added C Functions	**Description**
BGAPI2_Device_SetSerialPort()	Use this function to made a virtual connection from the camera uart port to a com port on the host pc.
BGAPI2_Polarizer_IsPolarized()	Used to check if the Device has a color or mono polarisation sensor.
BGAPI2_Polarizer_EnableSingleFormat()	Used to enable only one format from the Polarizer::Formats enumeration.
BGAPI2_Polarizer_GetByPredefinedImage()	Use the polarizer with a already defined image.
BGAPI2_Polarizer_IsFormatAvailable()	Used to query supported Polarizer::Formats.
BGAPI2_Image_GetLength()	This function delivers the image data size.

- C# API changes

Added C# Functions / Properties	**Description**
Device.SetSerialPort()	Use this function to made a virtual connection from the camera uart port to a com port on the host pc.
Polarizer.IsPolarized()	Used to check if the Device has a color or mono polarisation sensor.
Polarizer.EnableSingleFormat()	Used to enable only one format from the Polarizer::Formats enumeration.
Polarizer.GetByPredefinedImage()	Use the polarizer with a already defined image.
Polarizer.IsFormatAvailable()	Used to query supported Polarizer::Formats.
Image.GetLength()	This function delivers the image data size.

- Bugfix
  - C++ Interface, declaration of comparsion operators of BGAPI2::String class changed to const parameters.
  - C# Interface, solves a problem with function ImageProcessor.CreateBitmap when working with Mono8 images.
  - Solve a problem when register pnp events multiple times.

1.3 GigE Interface

- Enhancements
  - Added the new driver model 'Rivermax' to support the NVIDIA Rivermax.
  - Ressource optimizations in resend algorithm.
- Bugfix

- Solves a streaming abort that occured when using a small number of image buffers.

1.4 USB3 Interface

- No changes

1.5 SDK Examples

- All C++ SDK examples adjusted to the new list access.
- New C++ example 505_SerialPort added. Please refer to the readme.md for details.

1.6 Image Processor

- No changes

1.7 IPConfigTool

- The Tool switched to a command line interface. The GUI version is still available in the CameraExplorer package.

1.8 FilterDriverManager

- No changes

1.9 Setup

- Enhancements
  - SDK packages for Windows® are more unified in terms of packet structure.

1.10 Doc

- ProgrammersGuide Chapter 4 updated.

2 Release Notes Baumer GAPI SDK v2.11

2.1 General

- Information
  - Baumer GAPI package no longer include the CameraExplorer directly. It will be shipped in a separate package which can be downloaded from our website. But for convenience it is still included in the Baumer GAPI installer for Windows users.
    - Support for Linux® 64bit version on x86 platform.
    - Support for Linux® ARM® 64bit (aarch64) platform.
    - Support for Linux® ARM® 32bit (armhf) platform.
    - Support for Linux® 32bit version on x86 platform on request only.

2.2 Consumer

- Enhancements
  - Support for JPEG compressed image streaming introduced.
  - Support for Autobrightness introduced (C++ only).
  - C++ API behavior changes

C++ Function	new behavior	previous behavior
INode::GetName()	The function throws NotInitializedException exception, if the corresponding node map is not yet created (module not opened) or could not be created (is optional).	The function throws InvalidParameterException in this case.
SystemList::Refresh()	Search strategy changed. Baumer producers appear first in the system list.	Alphabetical order.
DataStreamEventControl::RegisterNewBufferEventHandler()	Multiple calls to this function deregister the actual handler and register the new one.	Multiple calls raise exceptions.

  - C++ API changes

Added C++ Functions	Description
InterfaceEventControl::GetEventMode()	Return the current selected event mode of the InterfaceEventControl.
DeviceEventControl::GetEventMode()	Return the current selected event mode of the DeviceEventControl.
DataStreamEventControl::GetEventMode()	Return the current selected event mode of the DataStreamEventControl.
Image::Init(Buffer*)	Convinience function to initialize a bgapi image with a bgapi buffer. Mainly used to decompress JPEG buffers.
Buffer::GetImageLength()	This function returns the length of the image in bytes starting at image offset. Use this function if the Buffer object contains image data and if the image length is variable.

DeviceEvent::GetEventMemPtr()	Direct access to the event memory.
DeviceEvent::GetEventMemSize()	Direct access to the event memory. Size of event memory.
Buffer(bool, void*)	New Buffer constructor introduced to allocate internal memory with the current paylaod size of the camera.
Device::WriteMemory()	Function to write camera register.
Device::ReadMemory()	Function to read from camera register.
Device::GetColorMatrix	Read the current used color matrix from the camera.
Device::SetColorMatrix	Set a color matrix to the camera, used for color calculation.
ImageProcessor::GetColorMatrix	Read the current used color matrix of the image processor object.
ImageProcessor::SetColorMatrix	Set a color matrix to the image processor object, used for color calculation.
Image::Scale	Scale an Image object according to the scaling factors.
Device::EnableChunkAuto(bool)	Allows the auto brightness features to enable necessary image information, transmitted together with the image as chunk blocks on the data stream.
BGAPI2::BrightnessAuto	New class introduced to control the auto brightness functionality. Direct access to this class is via public member Device::brightness_auto.
Removed C++ Functions	**Description**
EventControl::GetEventMode()	Function moved into the derived classes (InterfaceEventControl, DeviceEventControl, DataStreamEventControl)

- C API behavior changes

C Function	new behavior	previous behavior
BGAPI2_xxx_GetNode()	These functions throw NotInitializedException exception, if the corresponding node map is not yet created (module not opened) or could not be created (is optional).	These functions throw InvalidParameterException in this case.
BGAPI2_UpdateSystemList()	Search strategy changed. Baumer producers appear first in the system list.	Alphabetical order.
BGAPI2_DataStream_RegisterNewBufferEventHandler()	Multiple calls to this function deregister the actual handler and register the new one.	Multiple calls raise exceptions.

-

- C API changes

Added C Functions	Description
BGAPI2_Image_InitFromBuffer()	Convinience function to initialize a bgapi image with a bgapi buffer. Mainly used to decompress JPEG buffers.
BGAPI2_Buffer_GetImageLength	This property returns the length of the image in bytes starting at image offset. Use this function if the Buffer object contains image data and if the image length is variable.
BGAPI2_DeviceEvent_GetEventMemPtr()	Direct access to the event memory.
BGAPI2_DeviceEvent_GetEventMemSize()	Direct access to the event memory. Size of event memory.
BGAPI2_CreateBufferWithOptimizedSize()	New Buffer constructor introduced to allocate internal memory with the current paylaod size of the camera.
BGAPI2_Device_WriteMemory()	Function to write camera register.
BGAPI2_Device_ReadMemory()	Function to read from camera register.
BGAPI2_Image_Scale()	Scale an Image object according to the scaling factors.

- C# API behavior changes

C# Function	new behavior	previous behavior
SystemList.Refresh()	Search strategy changed. Baumer producers appear first in the system list.	Alphabetical order.
DataStreamEventControl. RegisterNewBuffer-EventHandler()	Multiple calls to this function deregister the actual handler and register the new one.	Multiple calls raise exceptions.

- C# API changes

Added C# Functions / Properties	Description
Image.Init(BGAPI.Buffer)	Convinience function to initialize a bgapi image with a bgapi buffer. Mainly used to decompress JPEG buffers.
Buffer.ImageLength	This property returns the length of the image in bytes starting at image offset. Use this function if the Buffer object contains image data and if the image length is variable.
DeviceEvent.EventMemPtr	Direct access to the event memory.
DeviceEvent.EventMemSize	Direct access to the event memory. Size of event memory.
Buffer(bool, object)	New Buffer constructor introduced to allocate internal memory with the current paylaod size of the camera.

Device.WriteMemory()	Function to write camera register.
Device.ReadMemory()	Function to read from camera register.
Image.Scale()	Scale an Image object according to the scaling factors.

- Bugfix
  - Solved a crash in function Interface::GetParent().
  - Solved a problem with multiple switching the event mode.
  - Default event mode of all event types is now set to polling.

2.3 GigE Interface

- Enhancements
  - Statistic of the control communication for GEV cameras introduced (see category GVCPStatistic in the device XML file), which helps analyzing connection problems.
  - Camera specific request ID implemented (only internal note)

2.4 USB3 Interface

- Enhancements
  - Stability by receiving asynchronous camera messages improved.

2.5 SDK Examples

- Enhancements
  - New SDK Examples 021_BrightnessAuto and 022_JPEGCapture added.
  - SDK project file creation now use directly the cmake generators offered from cmake tool. This allows the user to create the SDK Examples with new Visual Studio versions (e.g. Visual Studio 2019)
  - MultiThread Example more information added, debugging with GigE cameras improved by disabling the heartbeat.

2.6 Image Processor

- No changes

2.7 IPConfigTool

- No changes

2.8 FilterDriverManager

- No changes

2.9 Setup

- No changes

2.10 Doc

- No changes

3 Release Notes Baumer GAPI SDK v2.10.1

3.1 General

- Information
  - Small issues in CameraExplorer and USB Producer solved.

3.2 USB3 Interface

- Bugfix
  - Solves a problem which leads under some rarely circumstanced to a deadlock of the connected USB3 camera.

3.3 CameraExplorer

- Bugfix
  - Solves the problem that the color transformation settings of the CameraExplorer are locked, when the auto features of the connected cameras are enabled.

4 Release Notes Baumer GAPI SDK v2.10

4.1 General

- Information
  - Baumer GAPI packages and setup now strictly seperated in 32bit and 64bit packages.
  - Baumer GAPI now supports distribution independent packages for Linux. The packages are shipped in rpm, deb and tgz formats.
    - Support for Linux® 64bit version on x86 platform.
    - Support for Linux® ARM® 64bit (aarch64) platform.
    - Support for Linux® ARM® 32bit (armhf) platform.
    - Support for Linux® 32bit version on x86 platform on request only.
  - Linking of MSVC runtime libraries for Windows® changed to dynamic linking. The needed runtime DLL's are shipped with this version, to avoid installing the redistributable runtime package.

4.2 Consumer

- Enhancements
  - GenICam™ GenAPI reference implementation updated to version 3.2.
  - The Baumer GAPI provides now additionally a C interface. With the C interface it is possible to use Baumer GAPI with compilers which doesn't support the C++ Interface. It comes with a new API header file bgapi2_genicam.h and a new set of SDK examples for an easy start.
  - Polarizer class improved.
    - Selection of separate polarized images is possible (Pol0, Pol45, Pol90, Pol135).
    - Selection of the image with lowest reflections is possible.
    - Selection of the image with highest reflections is possible.
  - API changes

Description	**C++ Added** functions/definitions	**C# Added** functions/definitions
Added a new buffer chunk function (renamed see also removed section).	Buffer::GetChunkNodeList	Buffer.ChunkNodeList
Added a new class to support polarized sensors.	class Polarizer	class Polarizer
Added a new function which returns the display name of a camera event.	DeviceEvent::GetDisplayName	DeviceEvent.DisplayName
Added a new function which returns a image timestamp obtained when receiving the image on the host.	Buffer::GetHostTimeStamp	Buffer.HostTimestamp
Added a new function to enable the ThreadID in the trace messages.	Trace::ActivateOutputOption-ThreadId	Trace.ActivateOutputOption-ThreadId

Added a new function to abort the stacked mode.	Device::CancelStack	Device.CancelStack
Added a new function to change the camera remote configuration file (XML).	Device::SetRemoteConfiguration-File	Device.RemoteConfiguration-File
Added a new function to create empty images to be used with polarizer.	ImageProcessor::CreateImage	ImageProcessor.CreateImage
Description	**C++ Removed** functions/definitions	**C# Removed** functions/definitions
Removed the buffer chunk tree function (renamed see also added section).	Buffer::GetChunkNodeTree	Buffer.ChunkNodeTree

4.3 Image Processor

- Enhancements
  - The Image Processor now provides new features to retrieve pixel format characteristics.

4.4 GigE Interface

- Enhancements
  - New interface feature (GevInterfaceLinkSpeed) added to retrieve the NIC link speed.
  - New system feature (ActionCommandID) added to transmit a user programmable ID for an action command.

4.5 USB3 Interface

- Enhancements
  - Speed up the image reception under Linux® by using DMA transfer and by optimizing the data path. DMA transfer is only available when using internal allocated image buffers.

4.6 CameraExplorer

- Enhancements
  - The CameraExplorer stops and starts the image acquisition automatically when the user wants to change stream related features (e.g. Pixelformat, ROI).
  - Optimized the display of camera event information in the InfoView for incoming events.

4.7 IPConfigTool

- Enhancements
  - ForceIP support added in the Linux version.

4.8 FilterDriverManager

- No changes

4.9 Setup

- Enhancements

- Baumer GAPI Windows® setup is now separated in 32bit and 64bit versions.

4.10 Examples

- Enhancements
  - SDK examples available for the new introduced C interface.
  - MultiCamera example improved.

4.11 Doc

- No changes

5 Release Notes Baumer GAPI SDK v2.9.3

5.1 General

- Information
  - All Linux® distributions were up to date at the time of creation version 2.9.3.
- Bugfix
  - Solves a camera connection problem while changing the system time.

5.2 Consumer

- No changes

5.3 Image Processor

- No changes

5.4 GigE Interface

- Enhancements
  - Added a measurement to determine the packet round trip.

5.5 USB3 Interface

- No changes

5.6 CameraExplorer

- No changes

5.7 IPConfigTool

- No changes

5.8 FilterDriverManager

- No changes

5.9 Setup

- No changes

5.10 Examples

- No changes

5.11 Doc

- No changes

6 Release Notes Baumer GAPI SDK v2.9.2

6.1 General

- Information
  - All Linux® distributions were up to date at the time of creation version 2.9.2.

6.2 Consumer

- Enhancements
  - Shading correction improved by adding support for bayer formats and packed formats.
- Bugfix
  - C#-Interface: Solved a problem with reading from and writing to the 'value' property of the Node class.

6.3 Image Processor

- Enhancements
  - The Image Processor now provides performance optimized calculation of polarisation data. Therefore a new concept of single part and multi part images were introduced. It is also possible to correct the zero degree orientation of the camera.

6.4 GigE Interface

- Enhancements
  - Support for payload type Multi-part added.
- Bugfix
  - Correction of XML filename of interface and system modules.
  - Fixed a problem related to the status codes GEV_STATUS_PACKET_REMOVED_FROM_MEMORY and GEV_STATUS_PACKET_AND_PREV_REMOVED_FROM_MEMORY.
  - Fixed a problem with creation of req_id (request ID) in a multi camera system.

6.5 USB3 Interface

- Bugfix
  - Correction of XML filename of interface and system modules.

6.6 CameraExplorer

- Enhancements
  - New basic control added to work with the new polarisation cameras.

6.7 IPConfigTool

- No changes

6.8 FilterDriverManager

- Enhancements

- New command line options available: The filter driver manager is now able to remove unused driver versions from the windows driver cache, uninstall old filter driver versions 2.x, driver update including restore network adapter binding and improved selective choice of network adapter binding.

6.9 Setup

- Bugfix
  - Solved a signing problem and a start up problem of the setup for Windows 7 without service pack.

6.10 Examples

- Enhancements
  - New example introduced which shows how to start developing in a multi threaded and multi camera system.
  - New examples introduced which shows how to work with a polarisation camera with single and multi part images.

6.11 Doc

- No changes

7 Release Notes Baumer GAPI SDK v2.9.1

7.1 General

- Information
  - All Linux® distributions were up to date at the time of creation version 2.9.1.
    - Added Linux® distributions
      - Debian® 8.11
      - Debian® 9.5
      - Ubuntu® 18.04 (64 bit)
      - openSUSE® 15.0
    - Removed Linux® distributions
      - Debian® 8.10
      - Debian® 9.3
      - Ubuntu® 14.04
      - SUSE® 13.2

7.2 Consumer

- Enhancements
  - The API now supports sharpening and denoising.
  - The API now supports shading correction.
  - The API now offers the possibility to use a color transformation matrix stored on the camera for demosaicing performed on host side.
  - The API now supports multi cast for GigE Vision cameras.
  - ReleaseNotes 2.9.0 was subsequently changed, because behavior of function Node::IsDone() has also chagned. The missing description was added there.
  - API behavior changes

C++/C# Function	**new behavior**	**previous behavior**
SystemList:: CreateInstanceFromPath()/ SystemList. CreateInstanceFromPath()	Find GenTL producers only in the specified path.	Find GenTL producers in the specified path and in the application- (library-) path.

7.3 Image Processor

- Enhancements
  - The Image Processor now provides new features for the user to control sharpening and denoising.
  - The Image Processor now provides new features to configure a color transformation matrix.

7.4 GigE Interface

- Enhancements
  - The GigE Producer now supports data transmission to a group of receivers simultaneously (multi cast).
  - IP address assignment improved for cameras which needs more time after reboot (used with Force IP).
  - Force IP is now available under Linux® , but only with known MAC address.
- Bugfix

- The GigE Producer now find devices connected to virtual interfaces for Linux® .

7.5 USB3 Interface

- Enhancements
  - Speed optimization for Linux ® by using one transfer for the whole payload data.

7.6 CameraExplorer

- Enhancements
  - New 'View' Setting introduced to control the loading strategy of GenTL producers. Load local GenTL producer is the default setting.
  - Timestamp value of GEV cameras with PTP support are formatted as date time combination otherwise it is formatted as time span.
  - The Baumer 5x5 demosaicing method becomes the new default behaviour.
  - The CameraExplorer provides a GUI to control sharpening and denoising features.
- Bugfix
  - Fixed a problem with frame rate calculation.

7.7 IPConfigTool

- No changes

7.8 FilterDriverManager

- Information
  - New filter driver version 3.1.1.0 added. It solves a bug regarding multi cast.

7.9 Setup

- No changes

7.10 Examples

- Information
- Enhancements
  - New multicast example added.
  - New sharpening example added.
  - Two new examples added to show the use of shading correction.
  - New example added to show the demosaicing with the use of a color transformation matrix stored on the camera.

7.11 Doc

- No changes

8 Release Notes Baumer GAPI SDK v2.9.0

8.1 General

- Information
  - All Linux® distributions were up to date at the time of creation version 2.9.0.
    - Added Linux® distributions
      - Debian® 8.10
      - Debian® 9.3
      - Fedora® 26
      - Fedora® 27
      - openSUSE® Leap 42.3
    - Removed Linux® distributions
      - Debian® 7.11
      - Debian® 8.9
      - Fedora® 23
      - Fedora® 24
      - SUSE® Leap 42.2
  - Support for new Baumer 10 GigE camera models.
  - Support for new Baumer U3V CX camera models.
  - The Baumer GAPI licence agreement can be found in the subdirectory \Docs\License and also in the About dialog of the Camera Explorer.

8.2 Consumer

- Information
  - C++-Interface - The include path has changed from Components\Dev\C++\Inc to module specific path: Components\Dev\C++\Inc\bgapi2_genicam
- Enhancements
  - The API now is based on the GenICam™ GenAPI reference implementation. Please refer to the following API tables.
  - The API now allows directly creating instances of ImageProcessor objects, to allow image transformation simultaneously. Please refer table API changes.
  - GetDeviceEvent and GetPnPEvent now can be aborted by functions CancelGetDeviceEvent and CancelGetPnPEvent.
  - Baumer GAPI is now using a Strong-Name Signing for the .NET assembly.
  - API behavior changes

C++/C# Function	**new behavior**	**previous behavior**
Node::GetLocked()/ Node.IsLocked	Returns true if the XML tag <Locked> delivers true or if the node has read only (RO) access.	Returns true if the XML tag <Locked> delivers true.
Node::SetInt()	Throws an exception, if the division '(new value - minimum) / increment' has a remainder greater than 0 (to be GenICam compliant).	Returns the next valid value, if the division '(new value - minimum) / increment' has a remainder greater than 0.

Node::SetValue()	Throws an exception for integer nodes, if the division '(new value - minimum) / increment' has a remainder greater than 0 (to be GenICam compliant).	Returns the next valid value for integer nodes, if the division '(new value - minimum) / increment' has a remainder greater than 0.
Node::HasUnit()/ Node.HasUnit	Does no longer throw an NotAvailableException. The function now returns just with false.	Throws an NotAvailableException for nodes which interface type is not 'IFloat' and not 'IInteger'.
Node::GetIntMin()/ -	No exception for other interface types than 'IInteger' and 'IFloat'.	Throws an exception for other interface types than 'IInteger' and 'IFloat'.
Node::GetIntMax()/ -	No exception for other interface types than 'IInteger' and 'IFloat'.	Throws an exception for other interface types than 'IInteger' and 'IFloat'.
Node::GetDouble()/ -	Works for interface types 'IInteger' and 'IFloat'.	Works for interface types 'IInteger', 'IFloat', 'IEnumeration' and 'IBoolean'.
Node::SetDouble()/ -	Works for interface types 'IInteger' and 'IFloat'.	Works for interface types 'IInteger', 'IFloat', 'IEnumeration' and 'IBoolean'.
Node::GetDoubleMin()/ -	Works for interface types 'IFloat' and 'IInteger'	Works for interface type 'IFloat'.
Node::GetDoubleMax()/ -	Works for interface types 'IFloat' and 'IInteger'	Works for interface type 'IFloat'.
Node::GetDoubleInc()/ -	Works for interface types 'IFloat' and 'IInteger'	Works for interface type 'IFloat'.
Node::GetValue()/ -	Return a simple string representation for integer nodes.	Returns string according to its representation, (e.g. IPV4Address, MACAddress).
Node::IsDone()/ Node.IsDone()	Standard conform behavior implemented. Returns true if the self-clearing bit was reset by the device.	Returns true, if the execute command was successful acknowlged.
DeviceEventControl:: CancelGetDeviceEvent()/ DeviceEventControl. CancelGetDeviceEvent()	Is now implemented.	Throws an exception.
InterfaceEventControl:: CancelGetPnPEvent()/ InterfaceEventControl. CancelGetPnPEvent()	Is now implemented.	Throws an exception.

- API changes

Description	**C++ Added** functions/definitions	**C# Added** functions/definitions
Added a new function to provide an easy way to query read access to a node.	Node::IsReadable()	Node.IsReadable

Added a new function to provide an easy way to query write access to a node.	Node::IsWriteable()	Node.IsWriteable
Added a new function to query if an buffer object contains any chunk information.	Buffer::GetContainsChunk()	Buffer::GetContainsChunk()
Added a public constructor and destructor of ImageProcessor class.	ImageProcessor() ~ImageProcessor()	ImageProcessor() ~ImageProcessor()
Added the new function which allows reusing an image object.	Image::Init()	Image.Init()
Added the new function to transform an image object.	Already present.	ImageProcessor. CreateTransformedImage()
Added the new function to transform an image object into a buffer. Until now this function was located in Image class.	ImageProcessor:: TransformImageToBuffer()	-
Added the new function to create a bitmap object. Until now this function was located in Image class.	-	ImageProcessor.CreateBitmap()
Description	**C++ Removed** functions/definitions	**C# Removed** functions/definitions
Create and Release functions which handles the ImageProcessor object are removed.	ImageProcessor:: GetInstance() and ImageProcessor:: ReleaseInstance()	ImageProcessor.Instance
Use function CreateTransformedImage of class ImageProcessor instead.	ImageProcessor:: TransformImage()	Image.TransformImage()
Use function CreateBitmap of class ImageProcessor instead.	-	Image.CreateBitmap()
Use function TransformImageToBuffer of class ImageProcessor instead.	Image::TransformImageToBuffer()	-

Useless functions removed.	EventControl:: RegisterCustomEvent()	EventControl. RegisterCustomEvent()
Useless functions removed.	EventControl:: UnregisterCustomEvent()	EventControl. UnregisterCustomEvent()
Description	**C++ Deprecated** functions/definitions	**C# Deprecated** functions/definitions
Use GetCurrentAccessMode, IsReadable and IsWriteable instead.	Node::GetImposedAccessMode()	Node.ImposedAccessMode
Use GetCurrentAccessMode, IsReadable and IsWriteable instead.	Node::GetLocked()	Node.IsLocked
Use Node::GetValue() instead.	Node::GetString()	-
Use Node::SetValue() instead.	Node::SetString()	-

- Bugfix
  - Node::GetIntMin() and Node::GetIntMax() is now correctly working for interface type 'IFloat'.
  - Node::GetDouble() is now correctly working for interface type 'IInteger'.
  - Solved some API deadlocks in multi-threading environments (e.g. Device::Close).
  - Fixed a problem with function RegisterNewBufferEventHandler. This function fails sometime with an exception.

8.3 Image Processor

- Enhancements
  - New image initialisation function introduced, which now allows reusing existing image objects. This in turn considerable decreases the time of image management.
  - The Image Processor now supports pixel transformation from Mono12 into Mono16.
- Bugfix
  - Fixed a problem with converting the following pixel format: R8, G8 and B8.
  - Fixed a problem with demosaicing and ReverseY.

8.4 GigE Interface

- Enhancements
  - Speed optimization of the buffer management when using many buffer objects.
  - New filter driver version 3.1 introduced. It brings speed optimization and better performance for 10GigE devices.
  - Reworked the descriptions in GigE configuration file bsysgige.xml.
  - The GigE producer will no longer start with the optional GEV feature Primary Application Switchover (PAS). The user has to turn it on by himself if he wants to use it. Therefore the Device Feature 'AllowDeviceSwitchOver' is no longer needed and the new Interface Feature 'GevApplicationSwitchoverKey' was added.
- Bugfix
  - Fixed a problem under non Windows® platforms, that, under some circumstances, cameras were detected twice by the plug and play mechanism.

- The triggering of error events is prevented for synchronous function calls (e.g. for GCGetInfo).
- Fixed a crash in GCInitLib.
- Correct handling of stings which uses the whole register space (GEV 2.0 requirement). So now it's possible to use the full register length of e.g. DeviceUserID.
- Fixed a problem when more than one network adapters using the same subnet.
- Fixed a problem when using GEV feature Primary Application Switchover with the old TXG camera series (GEVCCPOverride).
- Fixed a stability problem with the resend algorithm, if the GEV device sends invalid BlockID's.
- Fixed a problem with invoke the maximum allowed packet size of a GEV device.
- Fixed a problem when catching events with infinite timeout. The function returned with timeout error.

8.5 USB3 Interface

- Enhancements
  - Speed optimization of the buffer management when using many buffer objects.
  - Solved a installation problem of the USB driver when using the original Windows® 7 version.
  - The USB3 Vision™ producer supports open device by a USB port ID. Therefore the following system features were introduced: USBPortSelector, USBPortID and USBPortLocationPath, needed to list available port ID's. Furthermore the following local device features USBPortID and USB3VisionGUID were introduced needed to get the current information.
- Bugfix
  - The triggering of error events is prevented for synchronous function calls (e.g. for GCGetInfo).
  - The stream info STREAM_INFO_PAYLOAD_SIZE returns now the correct error code GC_ERR_NOT_IMPLEMENTED.
  - GenTL requirement implemented: IFOpenDevice is now working without calling IFUpdateDeviceList before.
  - Fixed a problem when catching events with infinite timeout. The function returned with timeout error.
  - Fixed a problem with XML file access when reopening the camera.
  - Fixed a deadlock situation by active streaming. Affected environment was the ODROID ARM® board under Linux® Ubuntu® 16.04.

8.6 CameraExplorer

- Enhancements
  - Improved buffer management and image handling for higher performance and reduced CPU load.
  - New feature Frame Capture introduced. Capture camera frames and create a video files.
  - The setting 'Save and restore basic features' was renamed to 'Save and restore image acquisition settings'. This setting is now additionally as global setting available.
  - The global setting 'Recieve image buffers as fast as possible' is removed. The image acquisition is always running with the highest possible speed.
- Bugfix
  - The CameraExplorer now reverts it's chunk settings, when closing it. This behaviour is optional and can be disabled by the new global setting 'Save and restore chunk control settings'.
  - Fixed a problem, that the Camera Explorer doesn't start with camera settings from the last session.
  - Fixed a problem when starting 3rd party GenTL producer.

8.7 IPConfigTool

- No changes

8.8 FilterDriverManager

- Information
  The Filter Driver for Windows® 10 works now with Secure Boot.
- Enhancements
  - Solved a installation problem when using the original Windows® 7 version.

8.9 Setup

- Bugfix
  - Solved a problem with uninstalling the SDK examples.

8.10 Examples

- Information
  - Generated Eclipse projects works only if the used Eclipse CDT (C/C++ Development Tooling) version is less than 9.4.0 or greater or equal than 9.4.2.
- Enhancements
  - For Integer nodes with representation the function Node::GetValue() is used instead of Node::GetInt(), which delivers a prepared string according to its representation.
- Bugfix
  - Fixed a program abort in 007_Chunk example, if a chunk block node is not accessible.

8.11 Doc

- No changes

9 Release Notes Baumer GAPI SDK v2.8

9.1 General

- Information
  - All Linux® distributions were up to date at the time of creation version 2.8.
    - Added Linux® distributions
      - Debian® 8.9
    - Removed Linux® distributions
      - Debian® 8.6
  - ARM® -based platform
    - Added ARM® board
      - NVIDIA® Jetson TX2

9.2 Consumer

- No changes

9.3 Image Processor

- Enhancements
  - Working with packed pixel formats accelerated.

9.4 GigE Interface

- No changes

9.5 USB3 Interface

- Information
  - The Baumer USB3 Vision™ driver for Windows® 10 works now with Secure Boot.
- Enhancements
  - Compliant to GenTL 1.5 specification.

9.6 CameraExplorer

- Enhancements
  - Speed up the AutoExposure function.
  - CameraExplorer generates a warning, if a newer USB3 Vision driver, than the installed one, is available.
- Bugfix
  - AutoWhitebalance 'Once' corrected.

9.7 IPConfigTool

- No changes

9.8 FilterDriverManager

- No changes

9.9 Setup

- No changes

9.10 Examples

- No changes

9.11 Doc

- Information
  - API documentation format changed. The file GAPI_SDK_HELP.chm is replaced by ProgrammersGuide_Baumer_SDK_Reference.pdf (in folder Docs\Programmers_Guide) to make it also available for non Windows® packages.

10 Release Notes Baumer GAPI SDK v2.7

10.1 General

- Information
  - All Linux® distributions were up to date at the time of creation version 2.7.
    - Added Linux® distributions
      - SUSE® Leap 42.1
      - SUSE® Leap 42.2
    - Removed Linux® distributions
      - openSUSE® 13.1
  - ARM® -based platform
    - Added ARM® board
      - NVIDIA® Jetson TX1
    - Removed ARM® board
      - CompuLab CM-FX6

10.2 Consumer

- Bugfix
  - C# Interface
    - Solved a blocking problem of function RegisterPnPEvent.

10.3 Image Processor

- No changes

10.4 GigE Interface

- Information
  - Static Link Aggregation is now working again under Windows® 10 with the latest version 1607 and the latest Intel® Ethernet Network Drivers version 22.01.
- Enhancements
  - Compliant to GenTL 1.5 specification.
  - Compliant to GEV 2.0 specification.

10.5 USB3 Interface

- Enhancements
  - USB3 Vision™ support added for all delivered Linux® distributions and ARM® -based platforms
    - except Debian® 7
    - known stability issue, when using Plug and Play
  - Windows® : Stability and performance decisively improved regarding image acquisition mostly when reconfiguring and restarting the data stream.

10.6 CameraExplorer

- Enhancements
  - The CameraExplorer restores the Basic View settings from the last session. With the new option ‘Save and restore basic features’ this behavior can be disabled.

  - New GUI element added to show a description of the selected feature from the feature tree. The GUI element was realized as a tab which is located at the bottom of the info view.
  - Features with access status ‘Not Available’ will be shown as grayed out in the feature tree and their values will set to ‘Not Available’.
  - Added a new search box in the feature tree which helps finding a specific feature.
- Bugfix
  - Crash fixed with USB3 Vision™ camera when clicking on Rescan

10.7 IPConfigTool

- No changes

10.8 FilterDriverManager

- No changes

10.9 Setup

- No changes

10.10 Examples

- No changes

10.11 Doc

- No changes

11 Release Notes Baumer GAPI SDK v2.6

11.1 General

- Information
  - All Linux® distributions were up to date at the time of creation version 2.6.
    - Added Linux® distributions
      - Debian® 7.11, Debian® 8.6
      - Fedora® 23, Fedora® 24
      - openSUSE® 13.2 (only x86_64)
      - Ubuntu® 16.04
    - Removed Linux® distributions
      - Debian® 7.9, Debian® 8.2, Fedora® 21, Fedora® 22, Ubuntu® 12.04

11.2 Consumer

- Bugfix
  - Solved a thread problem with CancelGetFilledBuffer in multi camera systems, when using GetFilledBuffer with a very long or infinite timeout.
  - C++ Interface
    - Solved a crash under Linux® , if the environment variable $HOME is not set.
  - C# Interface
    - The function SystemList.CreateInstanceFromPath now works as expected.
    - Problems with RegisterDeviceEvent solved, when there was a prior call to UnregisterDeviceEvent.

11.3 Image Processor

- Enhancements
  - New source pixel formats for pixel transformation added: Mono12Packed to RGB formats, R[8,10,12], G[8,10,12], B[8,10,12] to RGB and Mono formats
  - New pixel transformation from source pixel format Mono12 into Mono16 added

11.4 GigE Interface GiGE VISION

- Enhancements
  - New packet based filter driver with version 3.0 added. It will be installed on all supported Windows® versions and replaces the old 2.x versions. Please note this filter driver is based on NDIS6.0, which is not supported by WinXP.
  - The previous filter driver (version 2.x) is no longer part of Baumer GAPI SDK v2.6 but is still supported. The support will removed definitely after one year.
  - Better support for DualLink camera series regarding resend strategy.
  - New values added to the buffer statistic to enable improved system analysis.
  - Supplementary change of Baumer GAPI SDK v2.5 release notes regarding filter driver versions.
  - DriverManager supports unpacking driver files into a specified folder.

11.5 USB3 Interface USB3 VISION

- Information
  - We strongly recommend, using the latest USB host controller driver in order to avoid system crashes when using the USB3 Vision™ producer. Please use at least the following driver versions:

  - Intel® USB 3.0 eXtensible Host Controller Driver for Intel® 8/9/100 Series and C220/C610 Chipset Family Version **4.0.6.60** from 01-Aug-2016
  - Intel® USB 3.0 eXtensible Host Controller Driver for Intel® 7 Series/C216 Chipset Family Version **1.0.10.255** from 03-Feb-2014
  - Renesas/Nec uPD720201/720202 USB 3.0 Drivers Version **3.0.23.0** WHQL 27-Feb-2014
- Enhancements
  - Buffer handling with USB host controller improved to increase system stability
  - Plug and Play improved, regarding speed and stability
- Bugfix
  - Solved an issued with lost frame counter and incomplete frame counter.
  - Invalid access to Buffer nodes corrected.

11.6 CameraExplorer

- Information (for Windows® user)
  - The CameraExplorer now depends on the Microsoft® Visual C++ Redistributable Package for Visual Studio® 2015. This package will automatically installed by the Baumer GAPI setup. When starting the CameraExplorer without running the Baumer GAPI setup first, you have to install the package separately.
- Enhancements
  - Added histogram and line profile diagrams.
  - ROI management introduced for a better feature usability
  - White Balance functionality added to basic view.
  - Stability improved, when loading an identical producer from different paths.
- Bugfix
  - Crash fixed when undocking several camera views

11.7 IPConfigTool

- Information (for Windows® user)
  - The IPConfigTool now depends on the Microsoft® Visual C++ Redistributable Package for Visual Studio® 2015. This package will automatically installed by the Baumer GAPI setup. When starting the IPConfigTool without running the Baumer GAPI setup first, you have to install the package separately.

11.8 FilterDriverManager

- Information (for Windows® user)
  - The Filter Driver Manager now depends on the Microsoft® Visual C++ Redistributable Package for Visual Studio® 2015. This package will automatically installed by the Baumer GAPI setup. When starting the Filter Driver Manager without running the Baumer GAPI setup first, please install the package separately.

11.9 Setup

- Information
  - Installs the Microsoft® Visual C++ Redistributable Package for Visual Studio® 2015.

11.10 Examples

- No changes

11.11 Doc

- No changes

12 Release Notes Baumer GAPI SDK v2.5

12.1 General

- Information
  - Support for the new Baumer U3V CX and EX camera series
  - Support for the new Baumer GEV CX and EX camera series

12.2 Consumer

- Enhancements
  - Support for firmware update added to API. The firmware update functionality is implemented as an Add-on and needs a separate module, which is not shipped with the Baumer SDK. Please contact Baumer to get the Add-on.
  - API changes

Description	**C++ Added** functions/definitions	**C# Added** funcions/definitions
This function delivers true if the update mode is available.	Device:: IsUpdateModeAvailable()	Device. IsUpdateModeAvailable
This function delivers true if the update mode is active.	Device:: IsUpdateModeActive()	Device.IsUpdateModeActive
This function enables or disables the update mode.	Device::SetUpdateMode(bool bActive, String pcCustomKey)	Device.SetUpdateMode(bool bEnable, string sCustomKey)
This function allows the access to a requested update feature of the remote device.	Device:: GetUpdateNode(String name)	Functionality is already available, use index operator of Device.UpdateNodeTree or Device.UpdateNodeList.
This function provides the update features of the remote device as tree structure.	Device::GetUpdateNodeTree()	Device.UpdateNodeTree
This function provides the update features of the remote device as unstructured list.	Device::GetUpdateNodeList()	Device.UpdateNodeList
This function delivers the GenICam™ XML File of the update features.	Device:: GetUpdateConfigurationFile()	Device. UpdateConfigurationFile
This function is used to read from a register node.	Functionality is already available.	Node.get(ref byte[] arrayBytes, ulong uiLength)
This function is used to write to a register node.	Functionality is already available.	Node.set(byte[] arrayBytes)
Gets the length of the Node object. Only valid for register nodes.	Functionality is already available.	Node.Length
Gets the address of the Node object. Only valid for register nodes.	Functionality is already available.	Node.Address

- Bugfix
  - Access to buffer information after revoking the buffer from buffer list is now possible when using external allocated buffers.

12.3 Image Processor

- Enhancements
  - Pixel transformation from BayerXX10p and BayerXX12p into RGB is now possible.

12.4 GigE Interface

- Information
  - New filter driver version 2.4 for Windows® 10 added. This filter driver is based on NDIS6. For older Windows® versions the existing filter driver version 2.3.1 will still be installed.

12.5 USB3 Interface

- Information
  - The Baumer U3V camera series VLU and MXU allow changing of payload size dependent features (like 'Width' 'Height', 'PixelFormat', . . .) while streaming is active. This behavior may lead to break image acquisition with Baumer GAPI SDK v2.5, when the image buffer is too small to capture the complete image. To avoid this issue, stop the acquisition first to change these parameters.
- Enhancements
  - Supports the new Baumer U3V camera series VCXU and VEXU
  - Different U3V devices can be used in different processes at the same time. E.g., suppose two U3V devices, 'A' and 'B', enumerated by the USB3 Vision™ producer. Now it is possible to use U3V device 'A' in one application and U3V device 'B' in another one.

12.6 CameraExplorer

- Enhancements
  - Optimization regarding image buffers. The CameraExplorer got a new option to work with the current payload size used by the camera. This reduces the memory usage and allows the use of more image buffers compared to older versions.

12.7 IPConfigTool

- No changes

12.8 FilterDriverManager

- No changes

12.9 Setup

- No changes

12.10 Examples

- Enhancements
  - New SDK example 201_DeviceUpdate which demonstrates the firmware update procedure.

12.11 Doc

- No changes

13 Release Notes Baumer GAPI SDK v2.4

13.1 General

- Information
  - All Linux® distributions were up to date at the time of creation version 2.4.
    - Added Linux® distributions Debian® 7.9, Debian® 8.2, Fedora® 22
    - Removed Linux® distributions Debian® 7.8, Debian® 8.0, Fedora® 20
      - test 4 ebene
  - Support for ARM® -based platforms added

13.2 Consumer

- Enhancements
  - Support Windows® 10 for USB3 Vision™
  - Support Windows® 10 for GigE Vision® – NDIS 6 based Filter driver for Windows® 10 added
  - API behavior changes
    - GenICam™ nodes of type 'String' are now supported by functions INode::GetNodeList and INode::GetNodeTree
    - Implemented the mechanism behind the mandatory SFNC feature TLParamsLocked. For backward compatibility with existing Baumer GAPI implementations the Consumer provides the Device feature DisableTLParamsLocked which bypasses this mechanism.
    - Confusing behavior of function SystemList::Refresh clarified. The function throws an exception, if the initialization of at least one GenTL producer fails but the list was filled with valid producer objects. The new version solved this behavior in the following way: The Refresh function doesn't throw an exception when it enumerates an invalid GenTL producer. This producer will now add to the system list. The exception is thrown when the user accessing it directly.
    - Caching type changed from WriteAround to WriteThrough for all node types except nodes with IFloat interface (see function Node::GetInterface to determine the node type and table API changes).
    - Behavior of C++ function Node::GetDoubleInc and C# property Node.Inc changed to be standard conform. The function throws an exception, if the underlying GenICam™ XML description file doesn't contain the ¡Inc¿ element. To avoid the exception use C++ function Node::HasInc (property Node.HasInc for C# respectively).
  - API changes

Description	**C++ Added** functions/definitions	**C# Added** functions/definitions
New parameter added to the function StartStacking to control whether all accesses to a register will be transferred to the device or only the last one.	Device::StartStacking(bo_bool bReplaceMode)	Device.StartStacking(bool bReplaceMode)
Clarify behavior of EVENTMODE_DISABLE and renamed it to EVENTMODE_UNREGISTERED	EVENTMODE_DISABLE renamed to EVENTMODE_UNREGISTERED	EventMode.DISABLE renamed to EventMode.UNREGISTERED

Renamed RGB planar pixel formats to be PFNC compliant.	”RGB8Planar” renamed to ”RGB8_Planar”	”RGB8Planar” renamed to ”RGB8_Planar”
	”RGB10Planar” renamed to ”RGB10_Planar”	”RGB10Planar” renamed to ”RGB10_Planar”
	”RGB12Planar” renamed to ”RGB12_Planar”	”RGB12Planar” renamed to ”RGB12_Planar”
	”RGB16Planar” renamed to ”RGB16_Planar”	”RGB16Planar” renamed to ”RGB16_Planar”
		C# Added
Find functions added – useful for faster access		SystemList.find
		InterfaceList.find
		DeviceList.find
		DataStreamList.find
		BufferList.find
	C++ Removed functions	C# Removed functions
Useless functions removed.	SystemList::erase	SystemList.Remove
	InterfaceList::erase	
	DeviceList::erase	
	DataStreamList::erase	
	BufferList::erase	

  - C# Interface
    - Solved a problem with removing initialized objects (refer to clear() functions of BGAPI2 list classes)

- Bugfix
  - A Buffer with faulty chunk data doesn’t deliver a ChunkNodeMap. But the buffer remains in this error state even if the chunk data is correct within the next image.
  - Solved a problem which blocks the image callback, e.g. caused by a corrupted chunk block
  - Solved a problem under Linux® platform when loading a GenTL producer from the GenICam™ path.
  - C++ Interface
    - Solved a problem with assignment operator.
  - C# Interface
    - Solved a problem, which signals DeviceEvents multiple times, when reopen a device.
    - Access to RemoteNodeList after Device.Close throws now a BGAPI2.Exceptions.NotInitializedException instead of System.AccessViolationException.
    - Solve a problem when calling DataStream::RegisterNewBufferEvent with EventMode.POLLING and EventMode.UNREGISTERED

13.3 Image Processor

- Bugfix
  - Transform image into RGB8_Planar when using the default Demosaicing method Bilinear 3x3 is now possible. Renamed RGB planar pixel formats to be PFNC compliant (see table API changes)

13.4 GigE Interface

- Bugfix
  - Solved a problem with deactivating the filter driver for a selected network adapter.

13.5 USB3 Interface

- No changes

13.6 CameraExplorer

- Enhancements
  - Supports loading and saving user sets for LXG cameras series
  - Improved displaying of images in Baumer Raw Format (.brw)
  - Supports the image format JPEG2000 for saving images to file

13.7 IPConfigTool

- No changes

13.8 FilterDriverManager

- No changes

13.9 Setup

- Enhancements
  - Reorganized Startup Menu. Changed to flat style with the new folder name Baumer GAPI-SDK.

13.10 Examples

- Enhancements
  - renamed the EventMode DISABLE (see Consumer) in the following examples: 006_DeviceEventMode_Polling, 010_PnPEventMode_Polling, 011_ImageCaptureMode_Handler , 012_DeviceEventMode_Handler, 012_DeviceEventMode_Handler

13.11 Doc

- Enhancements
  - InstallationGuide_Linux.pdf
  - ProgrammersGuide.pdf

14 Release Notes Baumer GAPI SDK v2.3 SP5

14.1 General

- Information
  - All Linux® distributions were up to date at the time of creation SP5.

14.2 Consumer

- Enhancements
  - C# interface
    - Documentation file bgapi2_genicam_dotnet.XML added
    - Additional Buffer constructor added, which allows the programmer to use his own memory
    - Problem solved, function CreateBitmap produces only gray images if the source image has RGB pixel format
- Bugfix
  - Crash fixed: BufferList deletes user allocated Buffer objects
  - Crash with multi-thread application fixed
  - Device::Close closes the corresponding data stream object

14.3 Image Processor

- Enhancements
  - Supports image transformation in RGB Planar formats higher than 8 bits
- Bugfix
  - The DemosaicingMethod Bilinear3x3 doesn't reach the maximum gray scale value

14.4 GigE Interface

- Enhancements
  - Deactivation of GEV Control Channel lock is now possible if device connection is lost (see feature UseControlChannelLocking)
- Bugfix
  - Resend algorithm improved: The Resend algorithm signals a buffer under specific conditions as incomplete. This behavior occurs only, if the previous signaled buffer was also incomplete.
  - GigE Producer under WindowsXP now, find new cameras series, like LXG
  - Improved handling with long heartbeat timeouts.
  - Buffer states improved
  - Action command returned with unjustified timeout.

14.5 USB3 Interface

- Bugfix
  - Buffer states improved
  - PnP behavior adjusted: programmer has to call Device::Close after a remove event

14.6 CameraExplorer

- Bugfix

- RGB value in status bar was not refreshed in live view
- Correct displaying GigE interfaces when using Asian OS languages
- Correct refreshing the received buffer counter in the info view.
- Shortcuts doesn't work after switching to full screen mode and back

14.7 IPConfigTool 

- Enhancements
  - Speed up setting persistent IP in multi camera systems

14.8 FilterDriverManager

- No changes

14.9 Setup

- No changes

14.10 Examples

- Enhancements
  - Supports creating project files for Visual Studio 2013 (see new cmake installer)
  - Example 005_PixelTransformation extended: now it shows the using of several demosaicing methods (see image processor feature 'DemosaicingMethod').

14.11 Doc

- Enhancements
  - InstallationGuide_Linux.pdf
    - Document folder corrected (see chapter 4.2.3)
  - ProgrammersGuide.pdf
    - Hint to C# documentation file bgapi2_genicam_dotnet.xml added (see chapter 4.4.3.1)
    - Description added: How to use user allocated buffer (see chapter 5.1.10)

15 Release Notes Baumer GAPI SDK v2.3 SP4

15.1 General

- Information
  - All Linux® distributions were up to date at the time of creation SP4.
  - Linux® distribution Debian® version 8.0 added
  - Linux® distribution Fedora® 21 added
  - Linux® distribution Debian® version 7.7 changed to Debian® version 7.8

15.2 Consumer

- Enhancements
  - Stacked write now available in C# interface.
  - Stacked mode allows writing the same feature more than once.
  - All SFNC 1.5 feature names available in bgapi2_featurenames.h.
- Bugfix
  - GetImagePresent returns false instead of an exception, if image chunk is not present.
  - GetFilledBuffer lost images under certain error situation (faulty chunk format caused by transmission).
  - CreateInstanceFromPath function works now under Linux®.

15.3 Image Processor

- Enhancements
  - Revised color fidelity: New demosaicing algorithms added. New feature added to switch between these algorithms: DemosaicingMethod.
  - New features added to the node interface which allows the modification of active color matrix: ColorTransformationValueSelector and ColorTransformationValue
  - See the documentation ProgrammersGuide_BaumerFeatures.pdf.

15.4 GigE Interface

- Enhancements
  - The limit of interfaces (about 10 NIC's) was repealed regarding MTU detection.
- Bugfix
  - Deadlock solved, when closing an interface under Linux®.

15.5 USB3 Interface

- Enhancements
  - o Added standalone installer for USB driver (msi file) for Windows.
- Bugfix
  - Crash fixed, if camera sends an invalid size of USB descriptor.
  - Crash fixed, if camera sends corrupted chunk data.
  - Reopen device doesn't work if system close was called without calling device close before.
  - Bugfix Pending Acknowledge.

15.6 CameraExplorer

- Bugfix
  - Camera frame rate in status bar will not reach zero if acquisition was stopped.

15.7 IPConfigTool

- Enhancements
  - IPConfigTool now available for Linux® (without Force IP).
- Bugfix
  - Invalid input of IP address leads to inadvertent programming.

15.8 FilterDriverManager

- No changes

15.9 Setup

- No changes

15.10 Examples

- No changes

15.11 Doc

- No changes

16 Release Notes Baumer GAPI SDK v2.3 SP3

16.1 General

- Information
  - Linux® distribution Debian® version 7.6 changed to Debian® version 7.7
  - Linux® distribution Ubuntu® 13.04 removed

16.2 Consumer

- No changes

16.3 Image Processor

- No changes

16.4 GigE Interface

- Bugfix
  - Deadlock fixed, when closing the GigE producer

16.5 USB3 Interface

- No changes

16.6 CameraExplorer

- Enhancements
  - Improved GUI performance with LXG camera series in Dual Link

16.7 IPConfigTool

- No changes

16.8 FilterDriverManager

- No changes

16.9 Setup

- No changes

16.10 Examples

- No changes

16.11 Doc

- No changes

17 Release Notes Baumer GAPI SDK v2.3 SP2

17.1 General

- Information
  - Linux® distribution Debian® version 7.5 changed to Debian® version 7.6

17.2 Consumer

- Enhancements
  - Stability improved
  - Device open error solved, if an interface object was opened before

17.3 Image Processor

- No changes

17.4 GigE Interface

- Enhancements
  - o Interface name for GigE adapters under Linux OS corrected
- Bugfix
  - Open device allows a second application to switchover the control, stream and message channel
  - Add Device feature AllowDeviceSwitchOver
  - Add Device feature DeviceStreamChannelPacketSize
  - Rename Device feature GevGVCPHeartbeatDisable into HeartbeatThreadDisable
  - Speed up the camera discovery process
  - Feature description of feature DevicePresent improved
  - Access to features of node list improved after device reopen

17.5 USB3 Interface

- No changes

17.6 CameraExplorer

- Enhancements
  - Correct displaying IP addresses and MAC addresses in the feature tree
  - Info view representation changed
- Bugfix
  - Crash when close the CameraExplorer fixed

17.7 IPConfigTool

- Information
  - Added support for LXG camera series

17.8 FilterDriverManager

- No changes

17.9 Setup

- No changes

17.10 Examples

- Information
  - Added support for LXG camera series

17.11 Doc

- Enhancements
  - ProgrammersGuide_BaumerFeatures.pdf
    - Added description for the new features DeviceStreamChannelPacketSize, AllowDeviceSwitchOver,
    - Description for feature HeartbeatThreadDisable adjusted
    - Description for DevicePresent adjusted
  - bexplorer_help.pdf
    - Info view description changed

18 Release Notes Baumer GAPI SDK v2.3 SP1

18.1 General

- No information

18.2 Consumer

- No changes

18.3 Image Processor

- Enhancements
  - Changes for RGB and BGR pixel formats.

18.4 GigE Interface

- Information
  - New filter driver version 2.3.1.0 with improved resend mechanism, Driver is furthermore backward compatible to former BGAPI versions, like 2.3, 2.2, 2.1. or 1.7.1
- Enhancements
  - add Device feature DevicePresent
  - add system feature GevFilterDrvVersion
  - add system features for ActionCommand
  - ActionCommandWrite
  - ActionDeviceKey
  - ActionGevDeviceIP
  - ActionGevDeviceIPAdd
  - ActionGevDeviceIPRemove
  - ActionGevDeviceIPSelector
  - ActionGroupKey
  - ActionGroupMask
  - ActionInterfaceIP
  - ActionInterfaceIPAdd
  - ActionInterfaceIPRemove
  - ActionInterfaceIPSelector
  - ActionGevDeviceAckWait
  - Support multi IP addresses on one interface
- Bugfix
  - bug fix feature force ip during open remote device
  - Heartbeat handling much more robust
  - Improve pnp behavior

18.5 USB3 Interface

- Bugfix
  - Improved error handling for transmission errors avoids dead lock.
  - added feature DevicePresent

18.6 CameraExplorer 

- Enhancements
  - improved presentation of info view

18.7 IPConfigTool 

- No changes

18.8 FilterDriverManager

- No changes

18.9 Setup

- Information
- Enhancements
- Bugfix
- No changes

18.10 Examples

- No changes

18.11 Doc

- Enhancements
  - ProgrammersGuide_BaumerFeatures
    - added 2.1.1 Category: ActionControl

19 Release Notes Baumer GAPI SDK v2.3

19.1 General

- Enhancements
  - Supports additional Linux® distributions: Ubuntu® 14.04, Fedora® 20 and OpenSUSE® 13

19.2 Consumer

- Information
  - The function DataStream::GetBufferByIndex is deprecated and will be deleted in the next version. Please use instead the BufferList class.
- Enhancements
  - Event handler added. This offers another way to retrieve images, device events and PnP events via call back functions in C++ and delegates in C# for easy implementation.

new EventMode EVENTMODE_EVENT_HANDLER added
new function pointer / delegate PnPEventHandler added
function RegisterPnPEventHandler is implemented
new function pointer / delegate DeviceEventHandler added
function registerDeviceEventHanlder is implemented
new function pointer / delegate NewBufferEventHandler added
function RegisterNewBufferEventHandler is implemented

  - Improved trace mechanism. The trace data were written into pieces of 100 MB file size. Every piece is stored as a compressed file. In maximum 400 pieces are allowed, this corresponds to a maximum trace size of 40 GB. Common system information, like OS version, memory usage, module versions added to trace output.
  - Buffer class provides new features to get the minimum and the maximum of the counter returned by function GetFrameID. The following features can be accessed via the Node-Interface:

FrameIDMax
FrameIDMin

19.3 Image Processor

- Enhancements
  - Supports the following new pixel formats R12p, G12p, B12p, BGR12p, R8, G8, B8

19.4 GigE Interface

- Information
  - New filter driver for new LXG cameras designed. Customer they are using filter driver, have to update to the new version. The new filter driver is backward compatible to older BGAPI versions, like 2.2, 2.1. or 1.7.1
- Enhancements
  - GenTL conformity to V1.3
  - Supports the new LXG camera series. The GigE interface now supports the Extended Chunk Data Payload Type specified in GEV 1.2, which is used by the LXG camera series.

- Plug and Play mechanism further improved.

19.5 USB3 Interface

- Information
  - Restrict the allowed operating systems to Windows® 7 or greater.
- Enhancements
  - GenTL conformity to V1.3

19.6 CameraExplorer

- Enhancements
  - Simplified handling of Basic View
  - Auto Exposure available in Basic View
  - Feature view added to source tab. This allows the user to control the System and Interface features without opening a camera first.
  - Image capture view revised.
  - searching for cameras accelerated

19.7 IPConfigTool

- No changes

19.8 FilterDriverManager

- No changes

19.9 Setup

- No changes

19.10 Examples

- Information
  - The SDK Examples are divided into common and interface specific examples.
- Enhancements
  - new examples PnPEventMode_Polling, ImageCaptureMode_Handler, DeviceEventMode_Handler and PnPEventMode_Handler
  - example ImagePolling_TriggerModeOff renamed to ImageCaptureMpde_Polling
  - example GEV_HeartbeatTimeout renamed to HeartBeatTimeout
  - example GEV_ForceIP renamed to ForceIP

19.11 Doc

- Information
  - Added some new documents to facilitate the use of Baumer GAPI:

GenICam_Standard_Features_Naming_Convention V1.5.1, V2.0 and V2.1
GenICam_GenTL_SFNC V1.0
Users_Guide for our camera series HXG, MXG, MXU, SXG, TXG, VisiLine and VisiLine_IP
Product flyer for our digital industrial cameras.

- Enhancements
  - New chapter: recommended working materials added.
  - New diagram added, which displays the communication model used by Baumer GAPI.
  - New chapters added to point 5. Application development with Baumer GAPI:

5.2 Information about System, Inferface, Device and DataStream
5.3 Parametrization of the camera

  - Representation of code snippets for C++ and C# improved. The code snippets for C++ and C# are under each other and are highlighted (C++ snippets yellow, C# snippets green)
  - Color coding of main classes improved.

20 Release Notes Baumer GAPI SDK v2.2 SP1

20.1 General

- No Information

20.2 Consumer

- No changes

20.3 Image Processor

- No changes

20.4 GigE Interface

- Bugfix
  - Camera discovery under Windows® XP corrected.
  - Configuration of resend parameter for dual GigE corrected.

20.5 USB3 Interface

- Bugfix
  - Improved error handling for transmission errors avoids dead lock.

20.6 CameraExplorer

- No changes

20.7 IPConfigTool

- No changes

20.8 FilterDriverManager

- No changes

20.9 Setup

- No changes

20.10 Examples

- No changes

20.11 Doc

- No changes

21 Release Notes Baumer GAPI SDK v2.2

21.1 General

- Enhancements
  - Supports versions the following Linux® distributions: Debian® 7.1, Ubuntu® 12.04, Ubuntu® 12.10, Ubuntu® 13.04 and Ubuntu® 13.10
  - Stability improved.

memory leaks fixed
plug'n play in multi camera systems stabilized

21.2 Consumer

- Enhancements
  - Some refresh functions of the list classes are changed. All Refresh functions of the list classes uses a timeout value as parameter. For some main classes this parameter is useless, therefore this parameter was removed for the functions SystemList::Refresh and DataStreamList::Refresh.
  - Handling with main class objects improved. Now it is possible to get access from a main class object to it's superordinate main class. E.g. when using a Device object there was no way to get access to the Interface object on which the device is connected. To achieve this the main classes provides the new function getParent:

Buffer::GetParent
DataStream::GetParent
Device::GetParent
Interface::GetParent

New functions added to check if a main class object is opened:

System::IsOpen
Interface::IsOpen
DataStream::IsOpen

  - DeviceEvent, Image and ImageProcessor inherits from node interface. The classes DeviceEvent, Image and ImageProcessor inherits from the node interface (class INode). For the class DeviceEvent it is possible to access device event features specified in the GenICam™ XML file of the device. For the Image and ImageProcessor classes it is possible to extend the functionality without changing the class definition.
  - Node interface extended. The class NodeMap provides now the iterator access like the list classes and two new find functions for fast node searching. The following new functions were declared:

class BGAPI2_DECL iterator
iterator begin();
iterator end();
iterator find(const String& _keyval);
bo_uint64 size();
Node* operator[](const String& val);
bo_bool GetNodePresent(String name);

The class Node defines a new function to get an increment of floating values:

Node::GetDoubleInc

- Device class extended. The Device class provides the new function GetSerialNumber to get the serial number of the device. This function can also be used for identifying the device without opening.
- Image class extended. New histogram function added to specify a ROI.
- Some functions renamed:

DataStream function renamed: Buffer::GetBufferID to Buffer::GetBufferByIndex
System function renamed System::GetName to System::GetFileName

- Error handling improved. Two new exception classes were added. The ObjectInvalidException is thrown if an BGAPI2 object is not valid or is no longer valid. The LowLevelException is thrown, if a GenTL call to a GenTL producer failes. Some unused exception classes were removed: InvalidIdException, IOException, SysVarMissingException.
- Preparation using event handler. The using of event handler is only prepared, but not implemented yet. Therefore the following functions are not implemented and throw always the NotImplementedException:

InterfaceEventControl::CancelGetPnPEvent
InterfaceEventControl::RegisterPnPEventHandler
DeviceEventControl::CancelGetDeviceEvent
DeviceEventControl::RegisterDeviceEventHandler
DataStreamEventControl::RegisterNewBufferEventHandler

21.3 Image Processor

- No changes

21.4 GigE Interface

- No changes

21.5 USB3 Interface

- Information
  - Supports USB3 Vision™ devices for Windows®

21.6 CameraExplorer

- No changes

21.7 IPConfigTool

- No changes

21.8 FilterDriverManager

- No changes

21.9 Setup

- No changes

21.10 Examples

- No changes

21.11 Doc

- No changes