

EASY, UNRESTRICTED, STANDARD-COMPLIANT

Software development as it should be



With GenICam as standardized camera interface and vision protocols like USB3 and GigE Vision, helpful components for developing machine vision applications are available.

But those who think that development tools of manufacturers are no longer necessary, don't know IDS peak yet.

Who develops machine vision applications today, can hardly avoid the generic interface standard GenICam of the EMVA (European Machine Vision Association). Today, standardized use of cameras from different manufacturers simplifies many integration processes and launches of vision applications while making camera hardware easily replaceable for the user. And also for manufacturers of industrial cameras such as IDS Imaging Development Systems GmbH it is obvious that data interfaces such as USB3 Vision and GigE Vision are indispensable if cameras should succeed on the market. As a result, they work automatically with most GenICam-compatible applications without the need to install manufacturer-specific driver software. Because image processing applications such as HALCON from MVTec already come with compatible Vision Transport Layers. Is there any space available or is there a need for development packages from the manufacturer? Isn't GenICam already the optimal interface for developing standards-compliant machine vision applications?

Optimal camera interface?

By migrating functionality to cameras, standardization has brought a fundamental and necessary breakthrough. The result was a very versatile software basis that can be used for communication with all vision cameras. All modules and functions are absolutely camera-independent and universally designed. Even the smallest details and parameters can be individually addressed and adjusted, making almost countless combinations possible. This makes the syntax of the used standard language very flexible indeed, but anything but easy and user-friendly. Even for simple program sequences, numerous lines of code and parameter settings are required in correct combination, which can quickly make processes error-prone and confusing. The strict separation and different handling of the camera functionality via GenAPI and the pure image buffer path via the transport layer is structurally comprehensible, but does not exactly make application development simpler and clearer.

The programming API of a product is a user interface and thus a tool that is measured by how well it can be operated. And only positive user experiences contribute to the success of a product. As interface it represents the camera itself and its possibilities for developers. Important attributes that describe a positive user experience are: easy, intuitive, flexible, transparent, future-proof and supportive. Despite the many technically outstanding solutions of the GenICam standard, the camera manufacturers cannot offer the user sufficient support and thus also no better user experience simply by the features of the cameras.

”

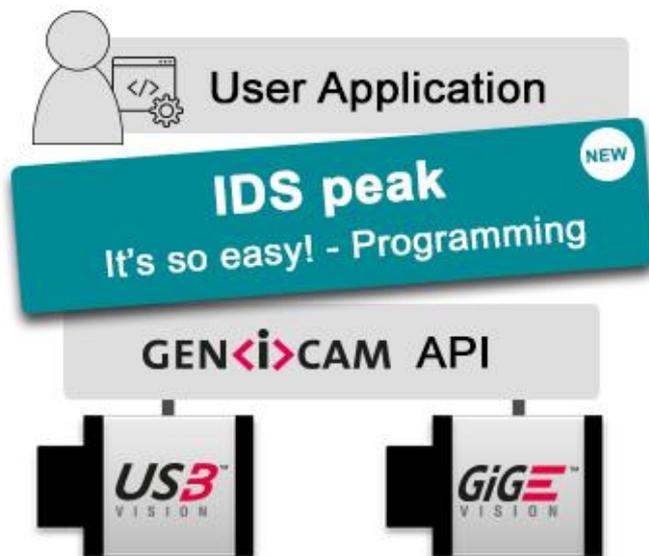
"There are many good reasons for camera manufacturers to offer their users complete software support," says Maike Strecker, Product Manager at IDS.

The best of several worlds

IDS has identified the advantages and disadvantages of GenICam and its own IDS Software Suite based on customer experience and would like to support users with IDS peak in several respects. With GenICam as the standardized software basis, the new SDK still gives users the freedom to choose which cameras they want to use without having to forego the advantages of a modern development interface. For an optimal user experience, the platform-independent IDS peak SDK provides all necessary components from source code samples to transport layer, so that customers can start developing their own applications directly. As a positive side effect, IDS peak users can rely on full support for all application components on IDS as a unique point of contact.

IDS peak simplifies the standard

IDS peak provides all necessary libraries and software interfaces for application developers and thus forms a complete SDK (Software Development Kit). In addition, the newly developed, object-oriented programming interface is much



easier to use than GenICam without limiting or bypassing its functionality.

Communication with the camera hardware takes place exclusively via the standardized interface and the Vision Transport Layer, whereby IDS peak remains absolutely hardware-independent and standard-compliant. Replaceability and compatibility with all USB3 Vision and GigE Vision compliant industrial cameras is preserved. There are no manufacturer-specific restrictions and even mixed operation of cameras from different manufacturers or with different interfaces is possible without any problems.

”

"Our easy to understand programming interface is a user-friendly alternative to direct access via GenTL and GenAPI. In addition, special convenience classes reduce the amount of code required and thus the programming effort," says Maike Strecker.

IDS peak is based on the module hierarchy defined by the GenTL standard. The modules are accessible via descriptors, which represent something like business cards of the modules. This allows you to request specific information from the modules before they are available and open. While the functionalities of the modules in lower layers only become visible and usable through so-called parent-child relationships when the corresponding modules are opened and available. This concept reduces errors when working with the classes and thus also the development time of an application.

An example of this is accessing the camera functions. Only via an already opened camera it is possible to access its "nodemap". Only then can detailed queries and special settings for this camera be made.

```
// get nodemap of an open camera device
auto remoteNodeMap = camera->RemoteDevice()->NodeMaps().at(0);
// get DeviceModelName from cameras nodemap
auto modelName = remoteNodeMap->FindNode<StringNode>("DeviceModelName")->Value();
```

Code Block 1: Reading the device name via the camera's nodemap

Convenience classes and functions, which are not available in the standard, simplify the handling of recurring processes, such as device and memory handling. Using these module classes reduces the necessary application code and thus improves the readability and stability of the camera application.

The DeviceManager module is the easy way to locate, list and open connected cameras and to manage the available transport layers and interfaces. With a simple update command, the module scans the entire system and a complete device tree is created or a specific camera is opened with just a few lines of code. Four lines of code are sufficient to open a connected camera using the IDS peak DeviceManager module.

```
// create a device manager object
auto deviceManager = std::make_shared<peak::DeviceManager>();
deviceManager->Update();
// get vector of device descriptors
auto devices() = deviceManager->Devices();
// select and open the first camera
auto device = deviceManager->Devices().at(0)->OpenDevice();
```

Code Block 2: Open camera with IDS peak DeviceManager module

IDS peak is constantly being expanded with such modules. A simplifying camera class and an acquisition manager for configuring the image acquisition are already planned. Some other well-known methods such as the use of shared pointers, which automatically release unused memory resources, or module-dependent thread security make IDS peak a modern object-oriented programming interface that offers programmers a lot of support and feedback at work.

However, IDS peak is only host software that remains completely camera-independent thanks to the GenICam software base and has no connection to camera models or firmware versions. Application developers are thus still free to choose whether they want to take advantage of the convenience, support and simplification of IDS peak. The camera features of every vision camera can thus also be used with any other GenICam-compliant application.

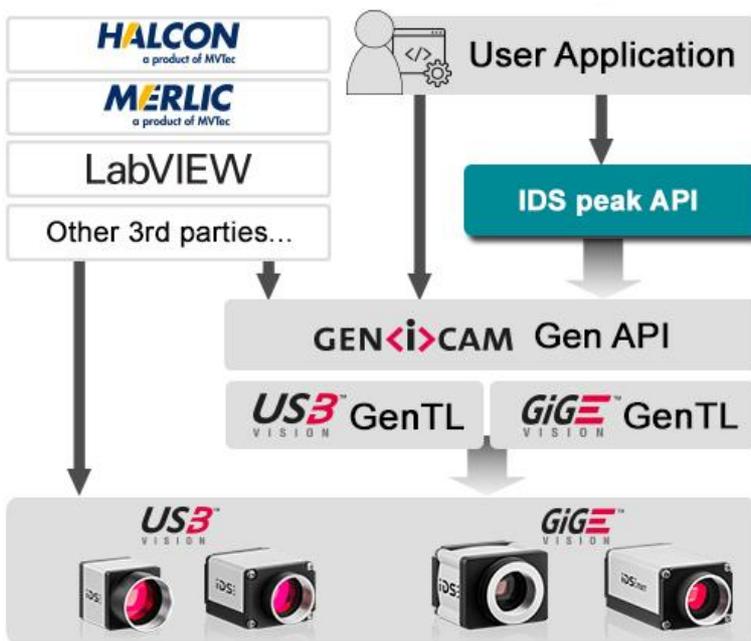


Figure1: Users can access IDS Vision cameras with their own applications via IDS peak, directly via GenICam or via a standards-compliant third-party application.

Application scaling and feature extensions

But IDS has even more in mind with the new SDK. Also the new app-based industrial cameras of the IDS NXT rio & rome families get a vision conform firmware and are therefore programmable with IDS peak. And it goes on. Application processes developed and tested with the IDS peak programming API on a host PC can be easily exported as an IDS Vision App with the new SDK. Self-programmed image processing processes can thus be loaded and executed directly in an IDS NXT camera like an app on a smartphone.

Using the camera's trigger and IO interfaces, machine controls, for example, can be directly notified of results and process decisions. An additional desktop PC is no longer necessary depending on the application.

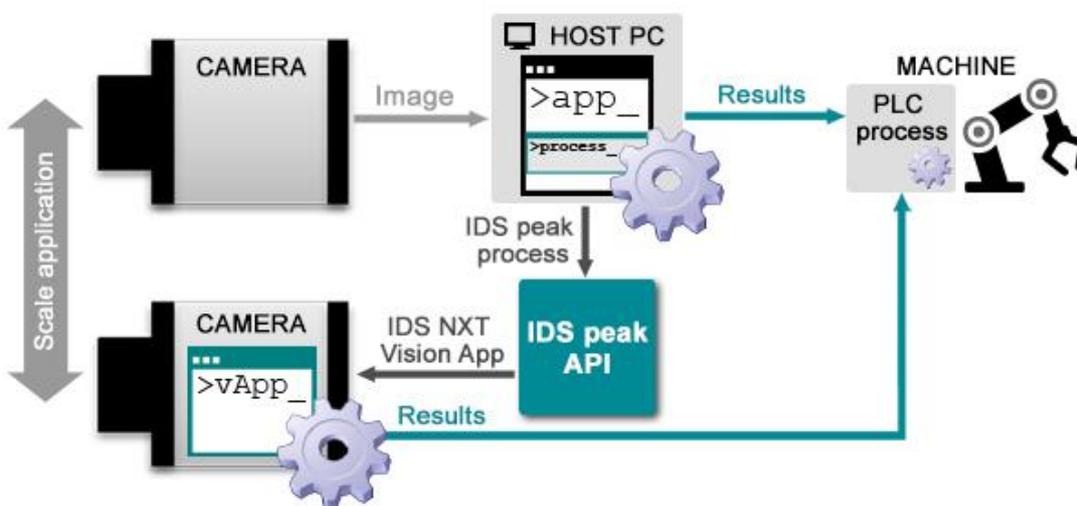


Figure2: Vision App Development with IDS peak

"Smart GenICam" also makes the input and output parameters of Vision Apps available via the camera's standardized XML description file for the customer application and any vision-compliant third-party application. This enables IDS customers to extend or change the feature set of their IDS NXT cameras with IDS peak at runtime. No need to wait for firmware upgrades from the manufacturer.

Software developers don't have to learn a new programming language for creating Vision Apps. With IDS peak you can program desktop applications as well as vision apps for intelligent cameras of the IDS NXT family. Embedded development and application scaling has never been easier. And all from one source.

New concept, new software, known simple

IDS also provides full software support for its USB3 Vision and GigE Vision industrial cameras with IDS peak. Customers of the established IDS Software Suite can now also benefit from easy handling and unlimited manufacturer support in the Vision World. As extension of the GenICam standard, IDS peak is not only a standards-compliant SDK with easy and intuitive programming API, but also independent software for operation, configuration and programming of all vision-compliant cameras.

As firmware basis for the new IDS NXT industrial cameras, it also serves as basis for the vision app development and becomes a universal tool with which embedded vision applications can be realized without much effort. With IDS peak, the camera manufacturer once again unites all its products under one software suite. This makes it again very easy for users.

Author

Dipl. - Ing. Heiko Seitz
Technical Writer

IDS Imaging Development Systems GmbH
Dimbacher Str. 6-8
74182 Obersulm
Germany

T: +49 7134 96196-0

E: h.seitz@ids-imaging.de

W: www.ids-imaging.com

© 2019 IDS Imaging Development Systems GmbH

Short Teaser:

IDS peak is a convenient alternative to GenICam for application development without having to forego standardization.

Newsletter text:

Who wants to develop machine vision applications today, can hardly avoid the generic interface GenICam of EMVA. Today, standardized use of cameras from different manufacturers simplifies many integration processes and launches of vision applications while making camera hardware easily replaceable for the user. Is there any space available or is there a need for development packages from the manufacturer? Isn't GenICam already the optimal development interface?

Read what makes IDS peak a convenient alternative to direct access via GenTL and GenAPI without sacrificing the benefits of standardization.