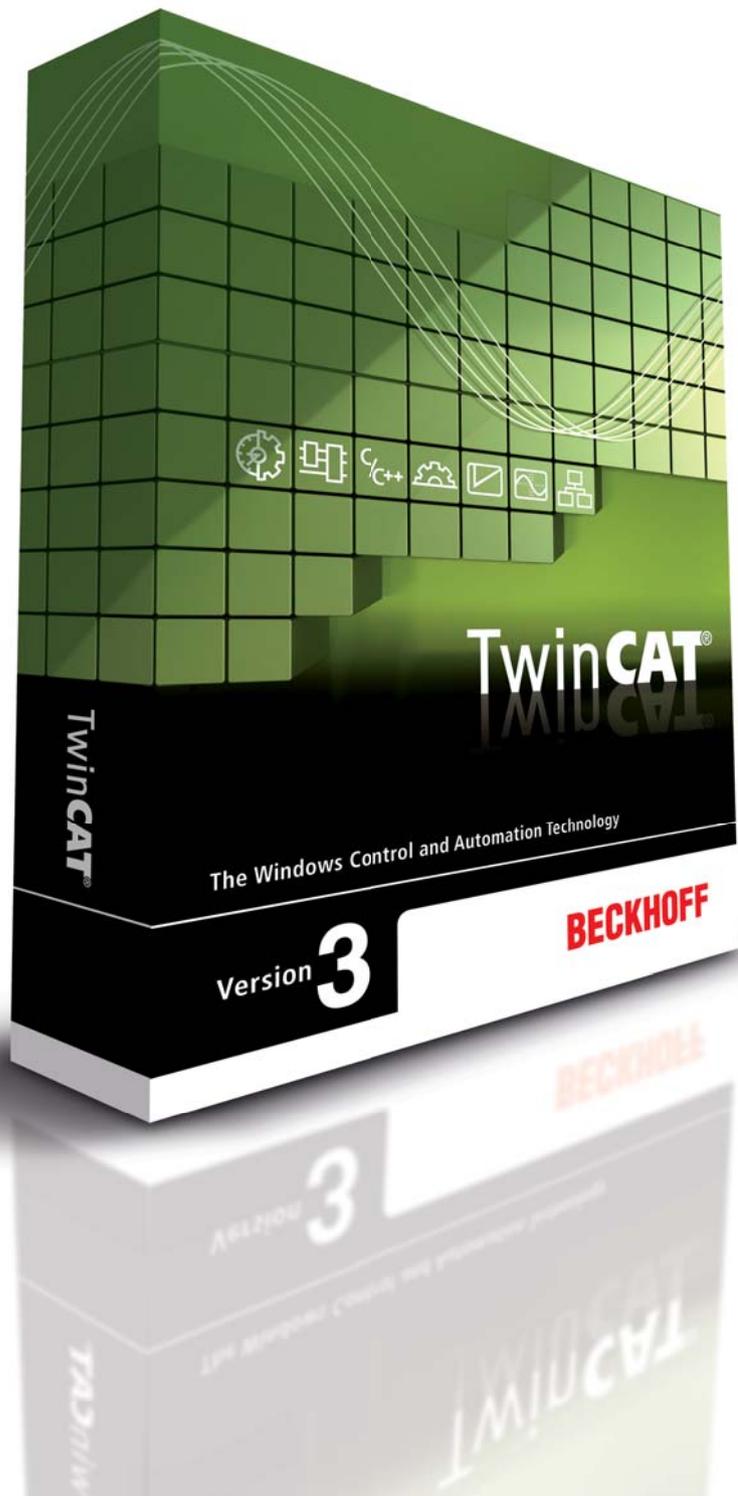


Interview on the release of TwinCAT 3

# Control programming utilizes benefits of the PLC and IT worlds

Modern software programming goes beyond Ladder Diagram. The software development process has to take this into account: The new TwinCAT 3 software generation from Beckhoff, which comprises a standard IT environment, engineering with modules and run-time on multi-core processors, will open up new opportunities for controls programmers. On the occasion of the release of TwinCAT 3 at the end of the year 2011, Ramon Barth and Dr. Dirk Jansen, who are Director for Software Development at Beckhoff and played a key role in the new software architecture, talk about the beta phase and the next stages on the roadmap.



Integrated development environments are becoming increasingly important in automation engineering. They tend to become the key criterion on which users base their system decisions. Increased engineering efficiency also plays an important role. TwinCAT, the software system from Beckhoff, transforms almost any compatible PC into a real-time controller with multi-PLC system, NC axis control, programming environment and operating station. It enables the embedding of an IEC 61131-3 software PLC, software NC and software CNC in Windows operating systems.

"With TwinCAT 3, the newest major update of our automation software, we have created a software platform for the automation technology of the next decade," said Ramon Barth, Director of System, HMI and Real-time Software Development at Beckhoff. "TwinCAT 3 enables optimal utilization of modern PC architectures. The development did not focus on the portal concept, but on a fully open system with a comprehensive product range," said Barth. The new software version offers advanced PLC programming with object-oriented extensions, C++ and Matlab®/Simulink® integration – all in a modular run-time that can be used by a whole range of devices from small ARM chips up to advanced multi-core X86 processors.

TwinCAT 3 supports real-time C++ based on the market-leading compiler from Visual Studio®. "The integration of the TwinCAT 3 development tools in a standard environment enables users to access all achievements of the IT world," said Ramon Barth. "Programs can be modular. Other benefits include code reusability and object-oriented mechanisms. In addition, the Matlab®/Simulink® algorithms run directly in the real-time environment."

## Suitable for PLC and high-level language programmers

According to Ramon Barth, TwinCAT 3 is ideal for two user groups: One group includes PLC programmers who feel at home in the IEC 61131-3 compliant environment. The other group is made up of users who want to use C++ to solve their tasks. "TwinCAT 3 can do



Dr. Dirk Janssen, Manager Software Development System, CNC and I/O at Beckhoff



Ramon Barth, Manager Software Development System, HMI and Real-Time at Beckhoff

both: object-oriented programming with inheritance mechanisms and the Ladder Diagram programming,” said Ramon Barth. The integration in Visual Studio® facilitates access to further tools. “Visual Studio® provides a whole range of advanced programming tools,” explained Dirk Janssen, Director of System, CNC and I/O Software Development. Ramon Barth lists UML integration and version control as examples. “Automation projects can now be planned in a manner that is standard in the IT sector, namely in line with the software development process,” Barth said. This is state of the art and is taught at universities. “Visual Studio® is a standard that everyone adheres to. Alternatives are only available in the open source world, but not with the same versatility, distribution and performance.” Dirk Janssen adds: “We give our users the same tools that we use for programming.” Users can now create the complete automation project in Visual Studio® and subsequently integrate it in TwinCAT 3.

### Code reusability

TwinCAT 3 offers software developers an infrastructure for generating reusable software modules. “The new TwinCAT version 3 demonstrates this through its own modularization,” said Ramon Barth. The principle is based on the exchange of interfaces, their description and the support of techniques for object-oriented programming (OOP). The interfaces guarantee compatibility with future developments. “TwinCAT provides the frame and parts of the implementation, in which users can embed their modules,” said Ramon Barth, reflecting on the fact that today’s projects are often realized by young engineers. “They want to introduce new ideas and demand modern tools. The younger programmers are familiar with the new technology and want to use it. Others might want to grow slowly into this new world, for example by starting with simple features such as inheritance.” “Nevertheless, existing PLC programmers have also been considered very specifically,” said Dirk Janssen, “since standard IEC 61131-3

programming continues to be fully available in TwinCAT 3. The object-oriented extensions in the PLC are seamlessly integrated into the existing IEC 61131-3 programming language, so that no user is forced to learn and use new programming paradigms.” In addition, the software is scalable so that “slimmed-down” versions are available. Thanks to object-oriented programming, the code can be easily reused, so that many program components can be created automatically,” said Dirk Janssen. While this was already possible in the previous version of TwinCAT, it required a higher-level compiler layer and was cumbersome for subsequent modifications. Ramon Barth provides an example for the practical benefits of code reusability: “A drilling unit that is used in different versions should be addressed as uniformly as possible by the controller. The user therefore defines the common features in a basic function block from which the various versions are derived, so that only the differences have to be newly developed.” Dirk Janssen stresses that TwinCAT 3 also simplifies commissioning further, since the modules can be fully tested in advance. “Nowadays commissioning engineers often only work with the adaption layer,” he said. “Locally, only smaller changes are required that are easy to implement. The object-oriented extensions offer convenient options for mapping the increased modularization of machines and their units in the control program.”

### Multi-core support

“The multi-core support of TwinCAT 3 enables the performance capability of the cores to be fully utilized,” said Dirk Janssen. “This has significant benefits since the machine program can be simply allocated to different cores.” “We already successfully tested this kind of allocation with 16 cores,” said Ramon Barth. “The hyperthreading function enables formation of virtual cores which are only marginally slower than genuine cores. On a Xeon server system with two physical cores and each with 6 cores 24 virtual cores for real-time can easily be allocated.”



"We have carried out benchmarks with all processors and the results have shown that TwinCAT's multi-core support enables performance enhancements in line with the number of CPU cores," said Dirk Janssen. "This is due to the fact that the control program is immediately available from memory without delay."

"With a single core, there are many disruptions that affect the processing power," said Ramon Barth: "High-priority tasks are processed first. In a multi-core system the performance losses caused by the synchronization of the cores are compensated by the elimination of changes in task context." The two software specialists agree: "The more complex the application, the greater the benefits."

"Modularization and multi-core technology complement each other ideally," said Dirk Janssen. "Because multi-core applications enable the modules to run independently without affecting each other."

"This saves processing power," said Ramon Barth. "Another advantage is that modularization makes the testing of programs easier."

### Reducing complexity

"The beta-phase of TwinCAT 3 was used intensively for the further improvement of the software," said Dirk Janssen. "To this end, we have worked with different users." Ramon Barth added: "Representatives of conventional automation technology were keen on object-oriented extensions and implemented their modularization ideas accordingly. We also received a number of suggestions for Matlab®/Simulink® support, for example, relating to the exchange of Matlab® elements." Simple handling and installation have proven to be additional benefits, said Ramon Barth: "For example, one tester who used an application with complex sensors had written his own hardware driver and was very satisfied." The performance capability of modern PC systems with multi-core support comes to the fore in complex sensor pre-processing tasks, since it eliminates the need for special and expensive external measuring systems. "A new experience for us were users who rely solely on C/C++ as an automation language without using PLCs," said Ramon Barth. "We also received positive feedback for the debugging options we transferred from the PLC world."

Ramon Barth stresses that TwinCAT 3 reduces complexity: "Only three to four mouse clicks are required for distributing the individual tasks to different cores." Users simply have to implement their code. The software itself deals with the complete infrastructure and the com-

munication via ADS. "This also applies to the integration of Matlab®/Simulink®," said Dirk Janssen. "It works just as it does with C++."

"No special target module is required for creating a TwinCAT system module with Matlab® – it is the PC itself," said Ramon Barth. "In the past, readily defined controllers were used that were no longer controllable," said Dirk Janssen. "Now individual controllers can be created and integrated."

TwinCAT 3 will be released at the end of the year 2011. Delivery of TwinCAT 3 is expected for early 2012.

[www.beckhoff.com/TwinCAT3](http://www.beckhoff.com/TwinCAT3)

The Author: Ronald Heinze, Published in ETZ Elektrotechnik + Automation 55, 2011, VDE Verlag, [www.vde-verlag.de](http://www.vde-verlag.de)



## Issues encountered in practice

Ramon Barth and Dirk Janssen, responsible for software development at Beckhoff, respond to issues encountered in practice.

**ETZ: To what extent is TwinCAT 3 open for other controller manufacturers?**

**Ramon Barth:** We recognize the desire of smaller controller manufacturers to run their code in TwinCAT. It enables them to utilize the wide range of options which our software environment offers, including close integration to the PC world and utilization of EtherCAT communication. Technically, this is no problem at all. This not only applies to controller manufacturers. In many cases machine manufacturers have also “cast” their expertise in the C programming language and initialized the possibility of utilizing it in a high-performance and open environment.

**ETZ: How can the source code be protected?**

**Ramon Barth:** Our TwinCAT licensing model required the target system to be identified. As an additional option the user code can be encrypted, so that it only runs on a particular target system or a protected system. Incidentally, the increasing significance of protected source codes does not exactly speak in favor of open source models.

**ETZ: What are your plans for the strategic further development of TwinCAT 3?**

**Ramon Barth:** The further development will focus on additional support for the development and utilization of predefined software modules. In TwinCAT 3 we introduced an integrated description format for software modules that will be extended and complemented with suitable tools.

**Dirk Janssen:** We plan to release an extended version of TwinCAT 3 at the end of 2012. Visualization, the development of which we supported with high-performance data interfaces in the past, will play an important role. Next year we will also launch a module with an integrated PC-based safety PLC. In addition, an authentication feature will be integrated in the communication with controllers and between controllers as an additional security level to provide even better protection from attacks.

**Ramon Barth:** The run-time system will become 64-bit-capable. In the past this only applied to the development environment, since there was little demand in the run-time sphere.

**ETZ: What progress have you made with your plan to offer software modules from third-party suppliers for TwinCAT?**

**Dirk Janssen:** From a technical point of view this has already been completed. The required interfaces and description formats are available and match the ones we use in-house. The roadmap still contains the issue of commercialization. We remain determined to include software modules from third-party suppliers in our product portfolio.

**ETZ: Do you envisage Eplan support in TwinCAT?**

**Dirk Janssen:** With TwinCAT 2 it is already possible to transfer variables from the Eplan world via supplements. It goes without saying that this is also possible in the new TwinCAT version 3. We regard this as state of the art. E-CAD systems can be integrated by means of an import tool.

**ETZ: Can old TwinCAT projects be maintained with version 3?**

**Dirk Janssen:** It is possible to run both versions on the same computer, so that service technicians can select the tools that match the respective run-time. We assume that users will manipulate their target systems with the version they used to create the programs. This ensures that only pertinent modifications are made. There is, however, an option to import existing TwinCAT projects in TwinCAT 3, so that they can run unchanged or extended in a TwinCAT 3 run-time.

**ETZ: What do the license models for TwinCAT 3 look like?**

**Dirk Janssen:** Licenses continue to apply to the run-time systems, rather than the programming environments. For the run-time systems we have introduced a fairer license model that is much more granular. Basic run-time components (real-time, I/O, PLC run-time,...) are regarded as modules. User modules can be integrated in the license system and utilize the Beckhoff infrastructure. Licensing is based on the target system.

**Ramon Barth:** By the way, our modules are not tied to a particular version.

**ETZ: Do you still offer free trials?**

**Dirk Janssen:** Of course. All TwinCAT modules support time-limited test operation. There are no functional limitations.