

.NET  
Powershell  
C/C++  
...

Automatic  
code  
generation

I/O  
C++

TwinCAT

TwinCAT

The Windows Control and Automation Technology

Version 3

BECKHOFF

BECKHOFF

Version 3

Windows Control and Automation Technology

7

Effizientes Engineering mit TwinCAT 3

# Automation Interface – die offene Schnittstelle zur automatischen Codegenerierung

Das Automation Interface der Automatisierungssoftware TwinCAT 3 ermöglicht die komplette Fernsteuerung des TwinCAT-Engineering-Systems und die automatische Erstellung von Programmen und Konfigurationen. So kann, abhängig vom Automatisierungsgrad, die manuelle Bearbeitung von Steuerungsprojekten, und die damit verbundenen Fehler, verringert bzw. vermieden werden. Durch die Automatisierung der Projekterstellung wird die Qualität der Engineering-Ergebnisse gesteigert – bei gleichzeitiger Zeit- und Kostenersparnis.



Die Automatisierung von Maschinen und Anlagen wird immer komplexer und zeitaufwendiger. Proportional hierzu wächst der Aufwand für das Engineering, was sich unter anderem in erhöhten Kosten für die Entwicklung der Konfiguration und die Programmierung der SPS widerspiegelt. Diese können vermindert werden, wenn entweder die komplette Software oder Teile daraus automatisiert erstellt werden. Darüber hinaus lassen sich durch eine automatisierte Codegenerierung Fehler in der Konfiguration reduzieren. Das TwinCAT Automation Interface setzt genau an diesem Punkt an: Es bietet eine Programmierschnittstelle, mit deren Hilfe das komplette TwinCAT-System mit nahezu allen Offline- und Online-Funktionen ferngesteuert werden kann. Die fehleranfällige und (personeil) teure Erzeugung von I/O-Konfigurationen und Programmen ist so einfach durchführbar.

## Das TwinCAT Automation Interface

Schon unter TwinCAT 2 stehen seit vielen Jahren diverse Schnittstellen zu verschiedenen Werkzeugen zur Verfügung, welche das Engineering vereinfachen. Das TwinCAT Automation Interface ist eine dieser offengelegten Schnittstellen,

welche an das Component Object Model (COM) von Microsoft angelehnt ist und – neben den Standard Microsoft .NET-Programmiersprachen – auch moderne Skriptsprachen, wie die Windows Powershell, unterstützt. Hierdurch ergeben sich neue Anwendungsbereiche und Einsatzszenarien für das TwinCAT Automation Interface.

Das TwinCAT Automation Interface besteht aus einer Vielzahl von Programmerroutinen, welche dem Anwender in Form von Klassen und Methoden zur Verfügung stehen. Diese ermöglichen unter anderem, TwinCAT-Projekte aus einer Sourcecode-Datenbank, wie dem Team Foundation Server, zu laden. Die Projekte können anschließend modifiziert und auf dem Laufzeitsystem aktiviert werden. Die Navigation durch ein Projekt erfolgt intuitiv, basierend auf der bereits aus der TwinCAT-Entwicklungsumgebung bekannten Baumstruktur, was die Einarbeitung in das Automation Interface wesentlich erleichtert.

Innerhalb eines Projektes können Konfigurationsknoten angefügt, gelöscht und modifiziert werden. Hierfür stehen – unabhängig von der Art des Knotens (PLC,

I/O oder Motion) – entsprechende Routinen zur Verfügung. Die Modifikation von Parametern an einem Knoten ist einfach, da nahezu alle Einstellungen in einem generischen XML-Format vorliegen. Über die im Automation Interface vorhandenen Methoden wird die Konfiguration eines Knotens geladen, über Standard-XML-Mechanismen, wie sie z. B. durch das .NET-Framework geliefert werden, verändert und wieder in der TwinCAT-Konfiguration gespeichert. I/O-Knoten können als Parameter z. B. Hardwareadressen oder Baudraten beinhalten, während ein PLC-Knoten (z. B. ein Funktionsbaustein) den Sourcecode des Knotens in PLCopen-XML in der entsprechenden Sprache der IEC 61131-3 liefert. Auch hier kann der Sourcecode geladen, modifiziert und wieder gespeichert werden.

Alle Elemente der TwinCAT-Konfiguration werden über das Automation Interface feingranular nach außen zur Verfügung gestellt. Hierdurch ergibt sich für die Kundenapplikation ein hohes Maß an Flexibilität, da das Automation Interface, im Gegensatz zur schrittweisen Erstellung der Konfiguration „Knoten-für-Knoten“, auch eine Konfigurationserstellung auf Basis von Templates ermöglicht. Die Applikation könnte sich hierbei aus einem Template-Pool, z. B. einem Source-Control-System, bedienen und so sukzessive die Konfiguration aufbauen. Templates lassen sich, angefangen von kompletten Projekten, bis hinunter auf Knoten-Ebene definieren; d. h. ein einmal konfiguriertes EtherCAT-Gerät aus dem I/O-Bereich ließe sich in einer Template-Datei abspeichern und später über das Automation Interface importieren und zu neuen Konfigurationen hinzufügen.

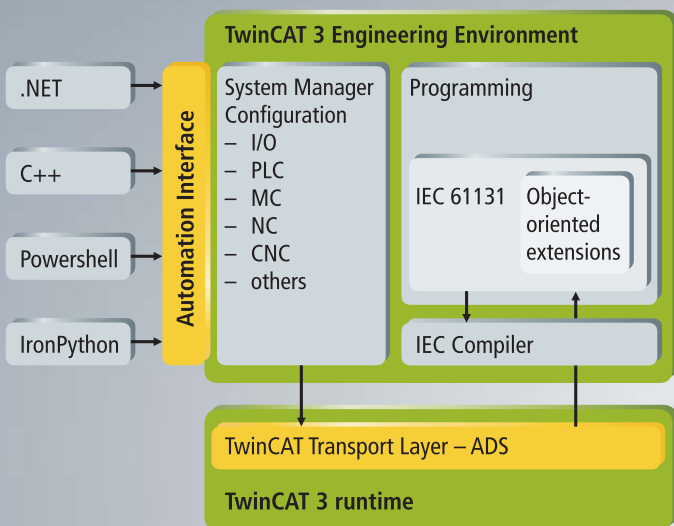
Die Erstellung einer Konfiguration basiert, gerade bei I/O-Konfigurationen, unter anderem auf Hardware-Adressen, welche natürlich nur dann vorhanden sind,

wenn ein Zugriff auf das entsprechende I/O-Gerät existiert. Das Automation Interface bietet die Möglichkeit, eine Konfiguration offline, also ohne angeschlossene I/O-Geräte, vorzubereiten und später – bei Vorhandensein der I/Os – die notwendigen Parameter nachzuschleifen. Alternativ kann eine Konfiguration natürlich auch mit angeschlossenen I/O-Geräten über die aus TwinCAT bekannte „Scan“-Funktionalität erfolgen.

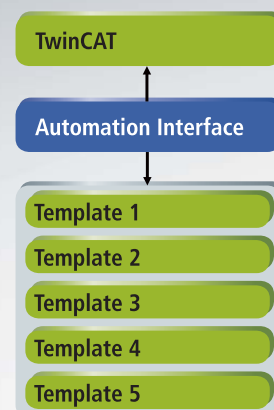
**Anwendungen des TwinCAT Automation Interfaces**

Mit Hilfe des TwinCAT Automation Interfaces können auf generischem und flexiblem Wege komplette Projekte generiert oder verändert werden. Die sich hieraus für den Kunden ergebenden Anwendungsmöglichkeiten sind überaus vielfältig: Sie können von einer einfachen Anwendung zum automatisierten Hinzufügen von Versionsinformationen zu SPS-Bausteinen bis hin zu komplexen, benutzerdefinierten Engineering-Tools reichen, welche die reguläre TwinCAT-Konfigurationsumgebung nur noch im Hintergrund verwenden. Aber auch offiziell verfügbare TwinCAT-Supplement/Function-Produkte bedienen sich des Automation Interfaces: So zum Beispiel das TwinCAT ECAD Import Tool, welches intern das Automation Interface verwendet, um aus einer, aus einem ECAD-Tool kommenden, XML-Datei ein Projekt zu generieren.

Im Beckhoff Information System finden Anwender eine detaillierte Beschreibung des Automation Interfaces, viele Lösungsbeispiele und Best-Practice-Artikel. Unter anderem wird hier eine Beispiel-Anwendung als Download zur Verfügung gestellt, welche dem Kunden einen Denkansatz für eigene Applikationen bietet und somit erste Erfolgserlebnisse im Umgang mit dem Automation Interface verspricht. Die Anwendung stellt auf einer grafischen Oberfläche, welche mit-



Architektur des Automation Interfaces in TwinCAT 3



Verwendung von Templates zur Konfigurationserstellung

tels WPF in C# geschrieben wurde, eine bestimmte Anzahl unterschiedlicher TwinCAT-Konfigurationen zur Verfügung. Diese lassen sich in einer XML-Datei detailliert spezifizieren. Auf Knopfdruck wird dann im Hintergrund automatisch die entsprechende Konfiguration erstellt und bei Bedarf aktiviert. Der Zeitaufwand hierfür beträgt wenige Sekunden – je nach Umfang der zu erstellenden Konfiguration. Eine solche Anwendung bietet ein sehr hohes Optimierungspotenzial für Situationen, in denen an einer Maschine mehrmals täglich die Konfigurationen gewechselt werden müssen, zum Beispiel weil Werkzeuge mit anderen I/O-Systemen aktiviert werden sollen. Ein Maschinenbediener kann dann ohne TwinCAT-Know-how auf einfache Art und Weise die unterschiedlichen Konfigurationen bei Bedarf aktivieren.

Eine weitere Demo-Anwendung demonstriert die mögliche Nutzung des Automation Interfaces aus einem Visual Studio® Add-in heraus. Die Anwendung lässt sich bei einem geöffneten TwinCAT-Projekt starten und fügt automatisch Versionsinformationen (Autor, Version) zu allen PLC-Bausteinen des Projektes in Form von später auslesbaren Konstanten hinzu. Eine solche Anwendung unterstreicht einmal mehr die Einsatzmöglichkeiten und die Flexibilität bei der Integration von Applikationen in die Visual-Studio®-Entwicklungsumgebung.



Autor: Josef Papenfort,  
Produktmanager TwinCAT,  
Beckhoff

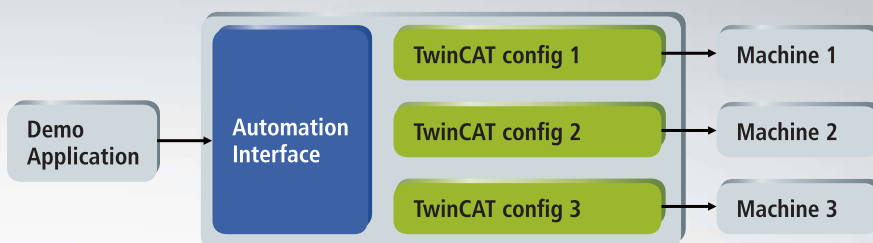


Autor: Sven Goldstein,  
Produktmanager TwinCAT,  
Connectivity & Embedded Systems,  
Beckhoff

weitere Infos unter:

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

<http://infosys.beckhoff.com>



Aufbau der Beispiel-Anwendung