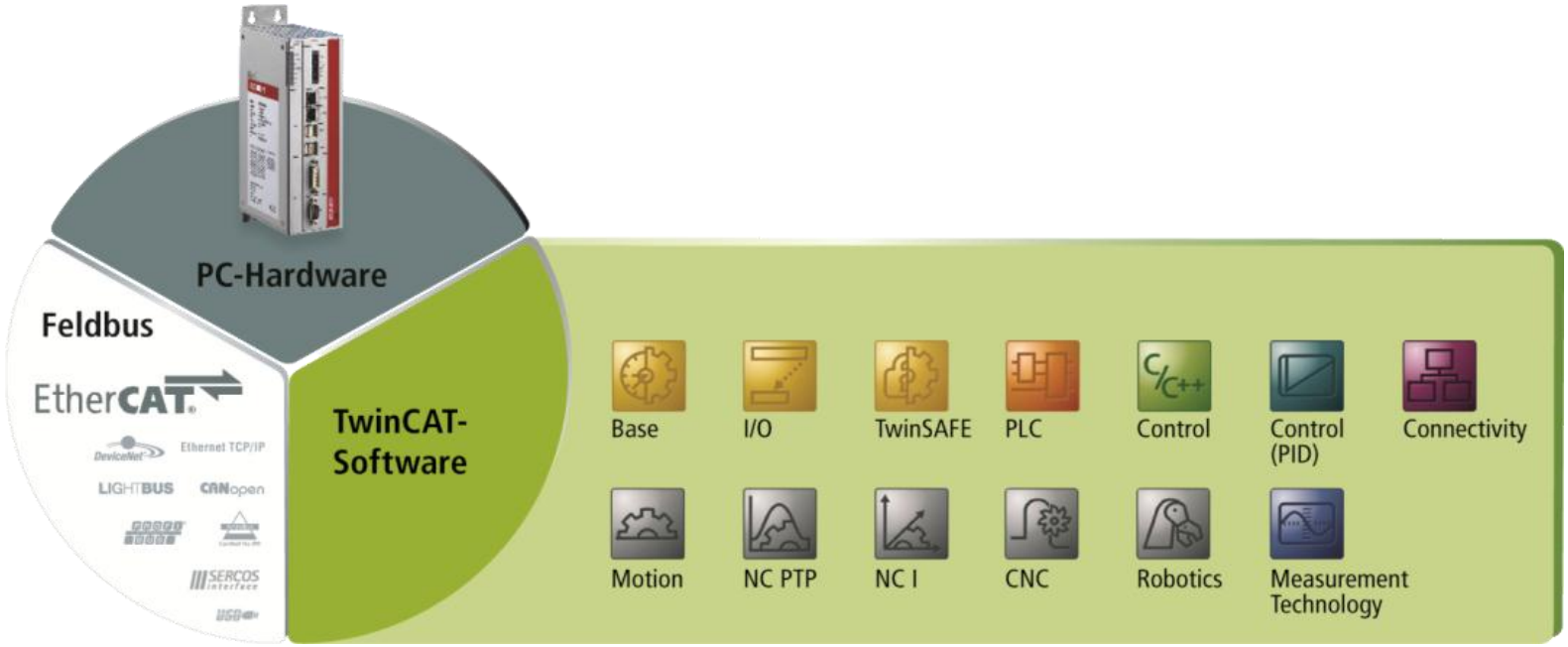
Agenda

- Motivation
- eXtended Automation (XA)
  - Architecture (XAA)
  - Engineering (XAE)
    - System Manager, PLC, Motion Control, C/C++ programming, Simulink
    - C#/ .NET programming
  - Runtime
    - Modules
    - Multi-Core and 64bit OS
- Roadmap
- Summary

Beckhoff: PC-based Automation

Beckhoff PC-based Control: Setting new standards in Automation!



Beckhoff PC-based Control – Advantages

- Integration of PLC, Motion and HMI into **one** software on **one** CPU:
  - Minimized Hardware
  - Faster cycle times due to no hardware interfaces
  - Reduced interface complexity
  - Better Diagnosis

- PC Control offers an „open” control system
  - Abstraction as a principle
  - Functions in software & independent from hardware

- Automation and IT world share the same benefits:
  - Performance increase
  - Cost decrease

PC-based Control - Milestones

1986

PC Control
PC-compatible machine control

1988

S1000
PLC/NC on PC

1989

Lightbus

1993

S2000
PLC/NC/CNC on PC

1995

Bus Terminal
universal field-bus module

1996

TwinCAT
IEC 61131 PLC real-time under Windows NT

2003

EtherCAT
real-time Ethernet Fieldbus

2008

XFC
eXtreme Fast Control Technology

PC-based Control - Milestones

1986

PC Control
PC-compatible machine control

1988

S1000
PLC/NC on PC

1989

Lightbus

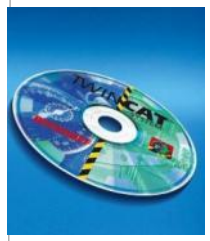
1993

S2000
PLC/NC/CNC on PC

1995

Bus Terminal
universal field-bus module

1996



TwinCAT
IEC 61131 PLC real-time under Windows NT

2003

EtherCAT
real-time Ethernet Fieldbus

2008

XFC
eXtreme Fast Control Technology

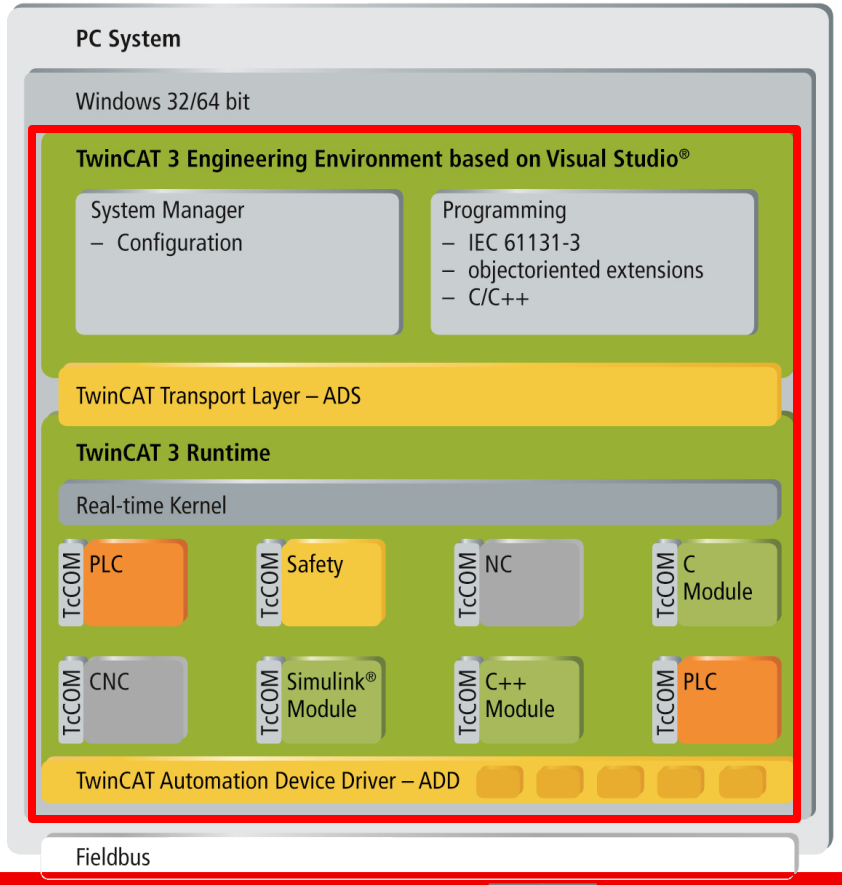
2010

TwinCAT3

TwinCAT 3 | eXtended Automation Architecture

eXtended Automation Technology (XAT)

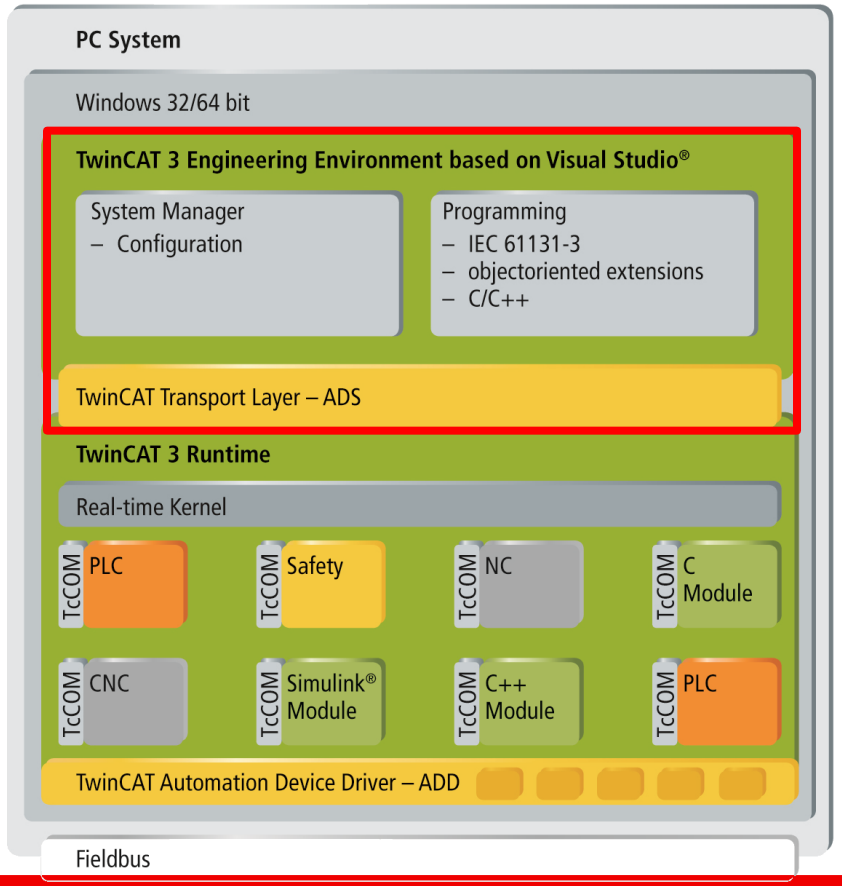
- More than standard automation



TwinCAT 3 | eXtended Automation Architecture

eXtended Automation Engineering (XAE)

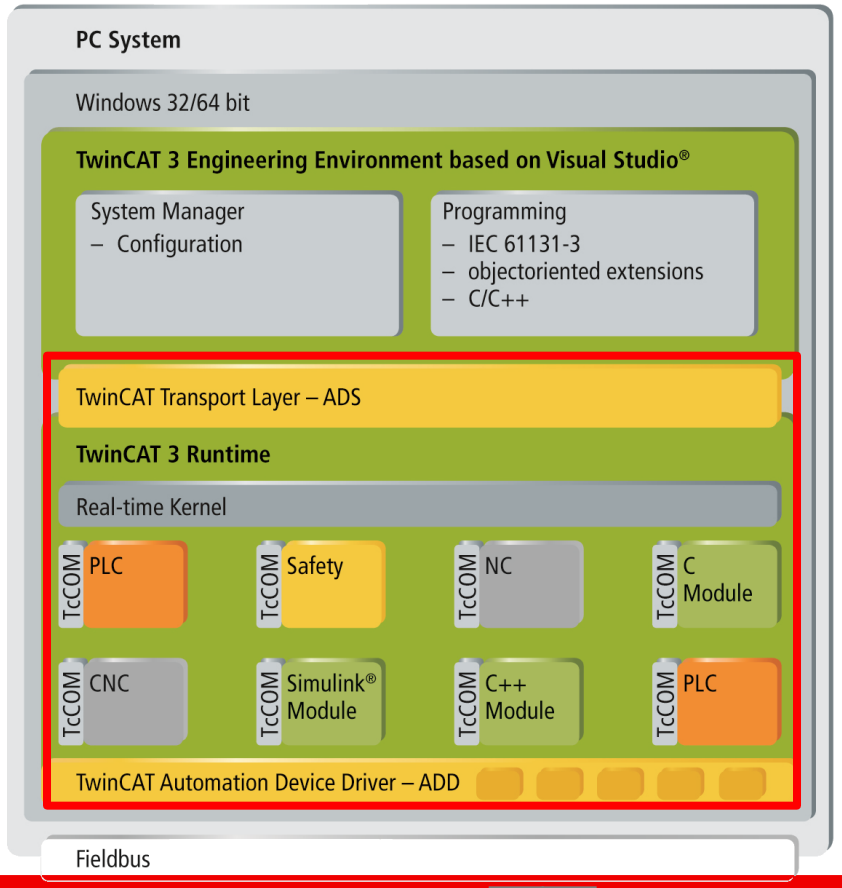
- Integration of Tools in **one** known framework: VS 2010



TwinCAT 3 | eXtended Automation Architecture

eXtended Automation Runtime (XAR)

- Realtime execution of modules written in different languages



TwinCAT 3 | eXtended Automation Engineering

eXtended Automation Engineering (XAE)

- TwinCAT 3 – extended , modular Engineering Tool
- **One** programming environment, **one** project file, **one** debug environment
- Integrated TwinCAT System Manager
- Programming according to IEC 61131-3 3rd Edition (including **NEW** object orientation extensions)
- Usage of C and C++ for real time programming
- Link to Matlab®/Simulink®
- Runs all TwinCAT 2 PLC projects without change *or*
- Migration of TwinCAT 2 projects (conversion)
- Based on Microsoft Visual Studio ® 2010

eXtended Automation Engineering (XAE)

Deliverable Engineering Products:

- **TwinCAT 3 Standard:**
  - Based on Microsoft Visual Studio Shell
  - Integrated System Manager
  - Integrated IEC 61131-3 (3rd. Edition) programming (including object oriented extensions)
  - Integrated Safety PLC

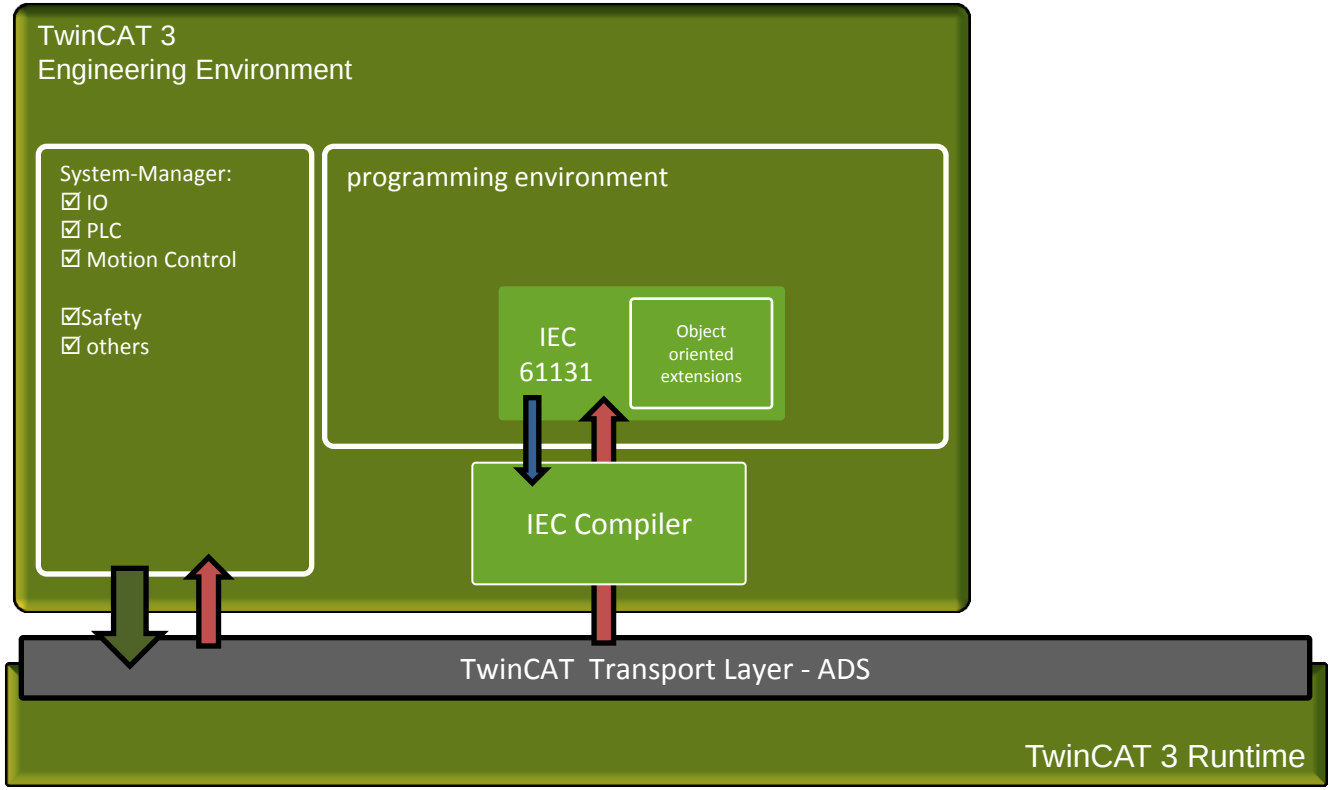
- **TwinCAT 3 Integrated**:
  - Integration into the Microsoft Visual Studio
  - Integrated System Manager
  - Integrated IEC 61131-3
  - Integrated Safety PLC
  - C and C++ programming
  - Link to Matlab Simulink
  - C# and .NET programming for (none real time) applications in the same environment
  - Option for further links to third party Software-tools

S

eXtended Automation Engineering (XAE)

TwinCAT 3 Standard

- For PLC Programmers
- For users of modules (e.g. with C/C++ or Matlab/Simulink generated)
- Configuring, setting up and diagnosis of system/fieldbus/motion
- Debugging of PLC application

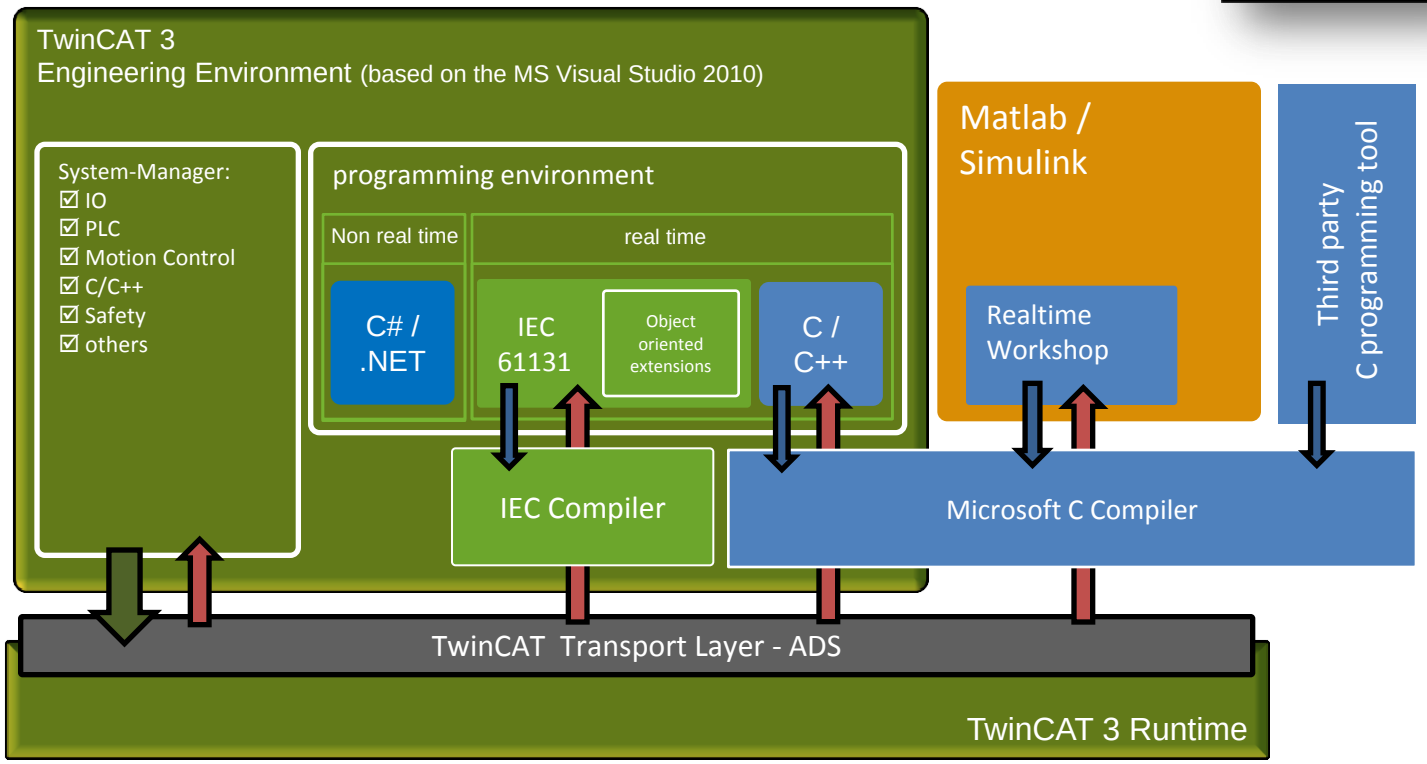


eXtended Automation Engineering (XAE)

TwinCAT 3 Integrated

- For PLC and C/C++ Programmers
- Configuring, setting up and diagnosis
- Module generation (C/C++ or Matlab/Simulink)
- Debugging PLC, C/C++, Matlab/Simulink

Microsoft Visual Studio PARTNER



caption:
- code generation
- debugging
- upload of TC modules

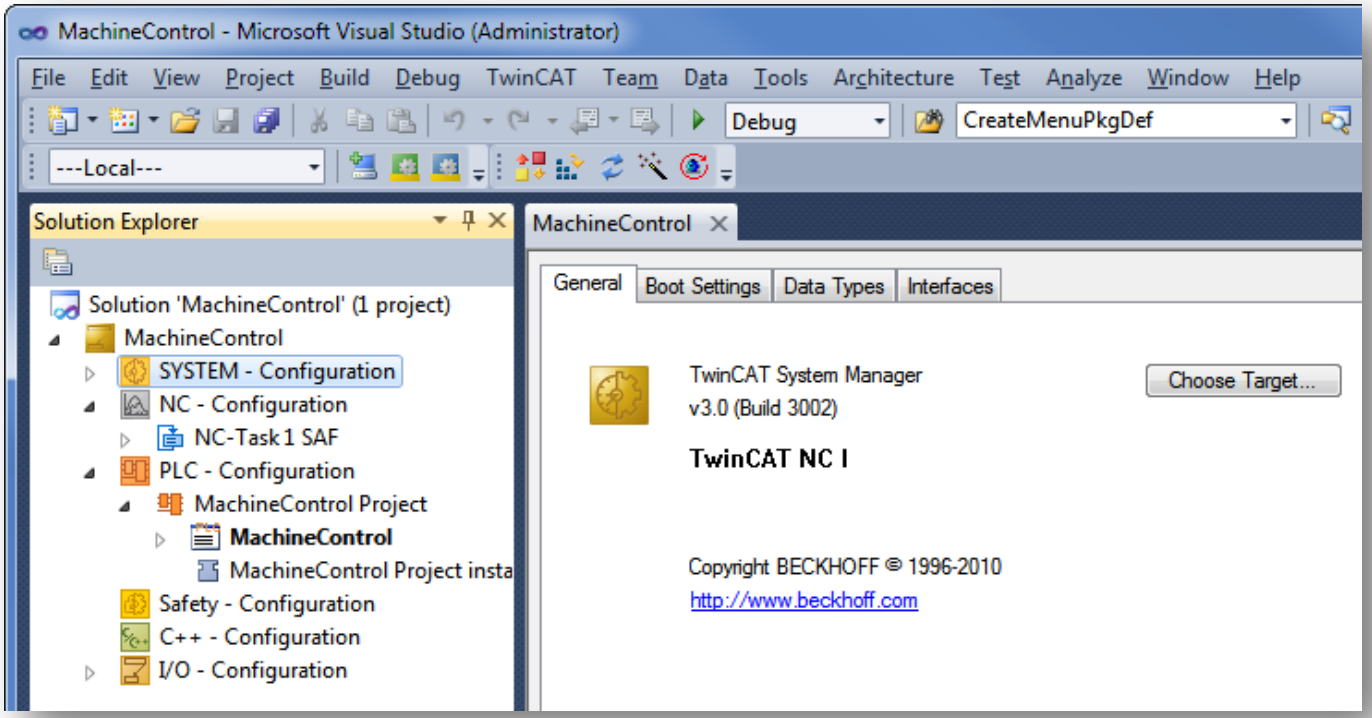
Workbench Integration

TwinCAT 3 framework = Microsoft Visual Studio 2010

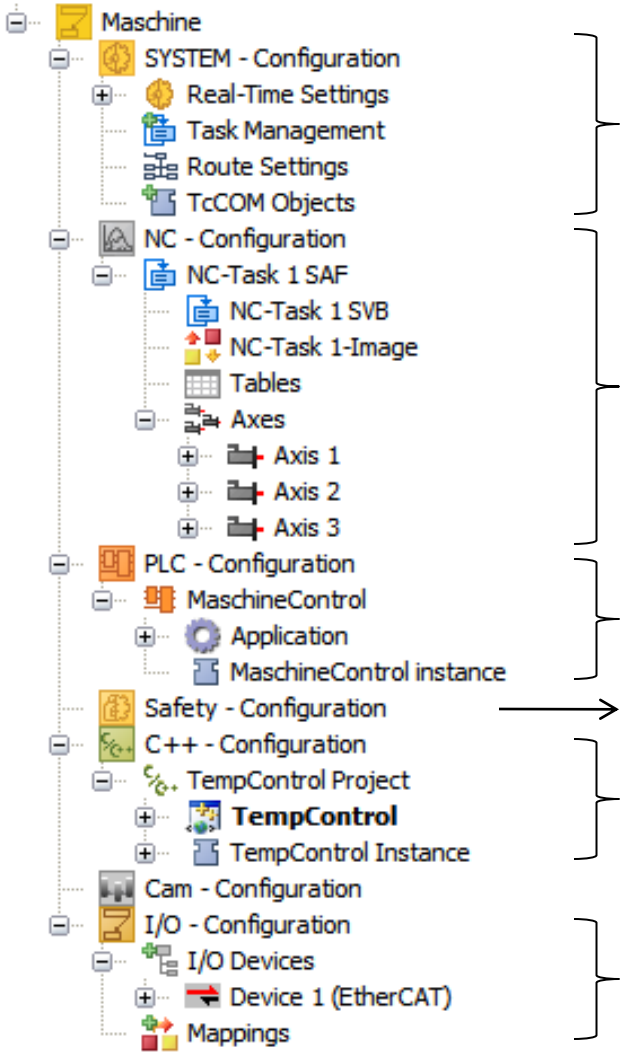
- Usage of the most common programming environment
- Extendable via PlugIns
- Link to common source control software
- Usage of C und C++ for programming automation devices
- Usage of .NET languages for none real time applications (e.g. HMI)
- Improved help system

TwinCAT I/O – Integrated System Manager

- Programming, configuration und diagnoses in one tool
  → Continuous Engineering since 1996
- Uniform task management
- Parameterization of TwinCAT modules
- Creation and administration of mappings between the process images
- Simulation of I/O's and axis



TwinCAT I/O – Integrated System Manager



System configuration

Motion control configuration

PLC configuration

Safety PLC configuration

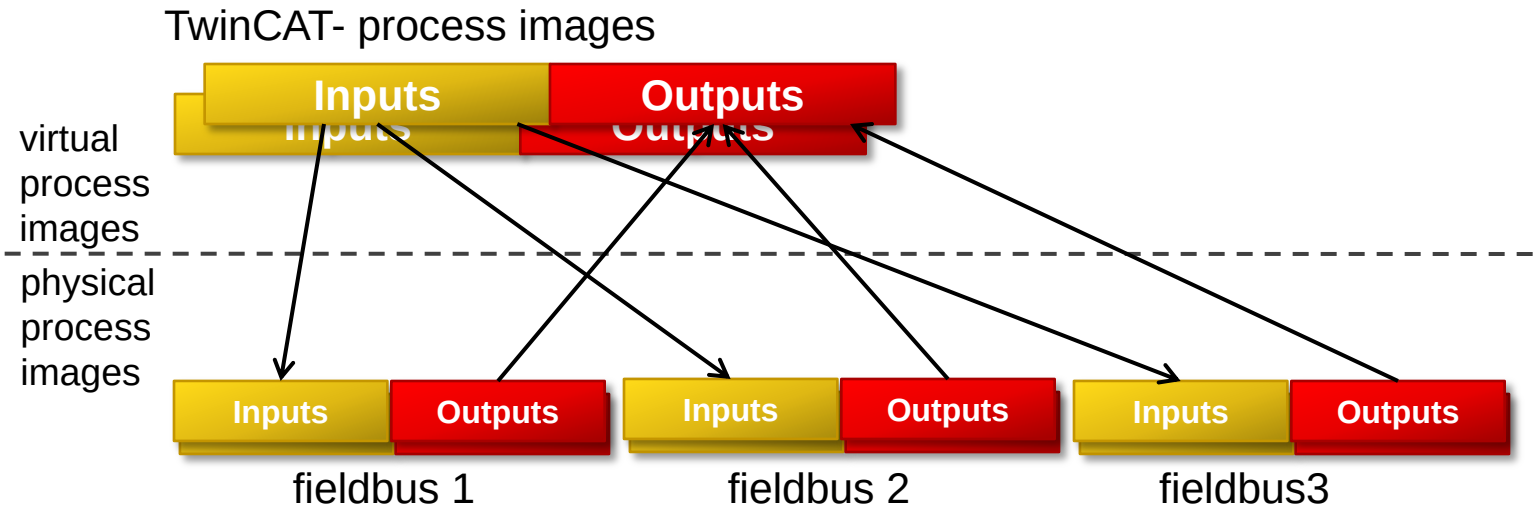
C++ module configuration

I/O configuration

TwinCAT I/O – mapping of process images

- Open for all common field busses
- Support of all PC hardware interfaces
- Easy commissioning and diagnosis
- Assignment of logical and physically process image
  → Changes of the bus system do not require a change of the PLC code

DeviceNet, CANopen, Modbus, RS232 / RS485, Ethernet TCP/IP, LIGHTBUS, USB, SERCOS interface, INTERBUS, PROFIBUS, ControlNet, EtherCAT



J

TwinCAT 3 PLC

Multiple PLC projects:
- Number of possible tasks: 65.000
- Number of PLC projects: only limited by memory

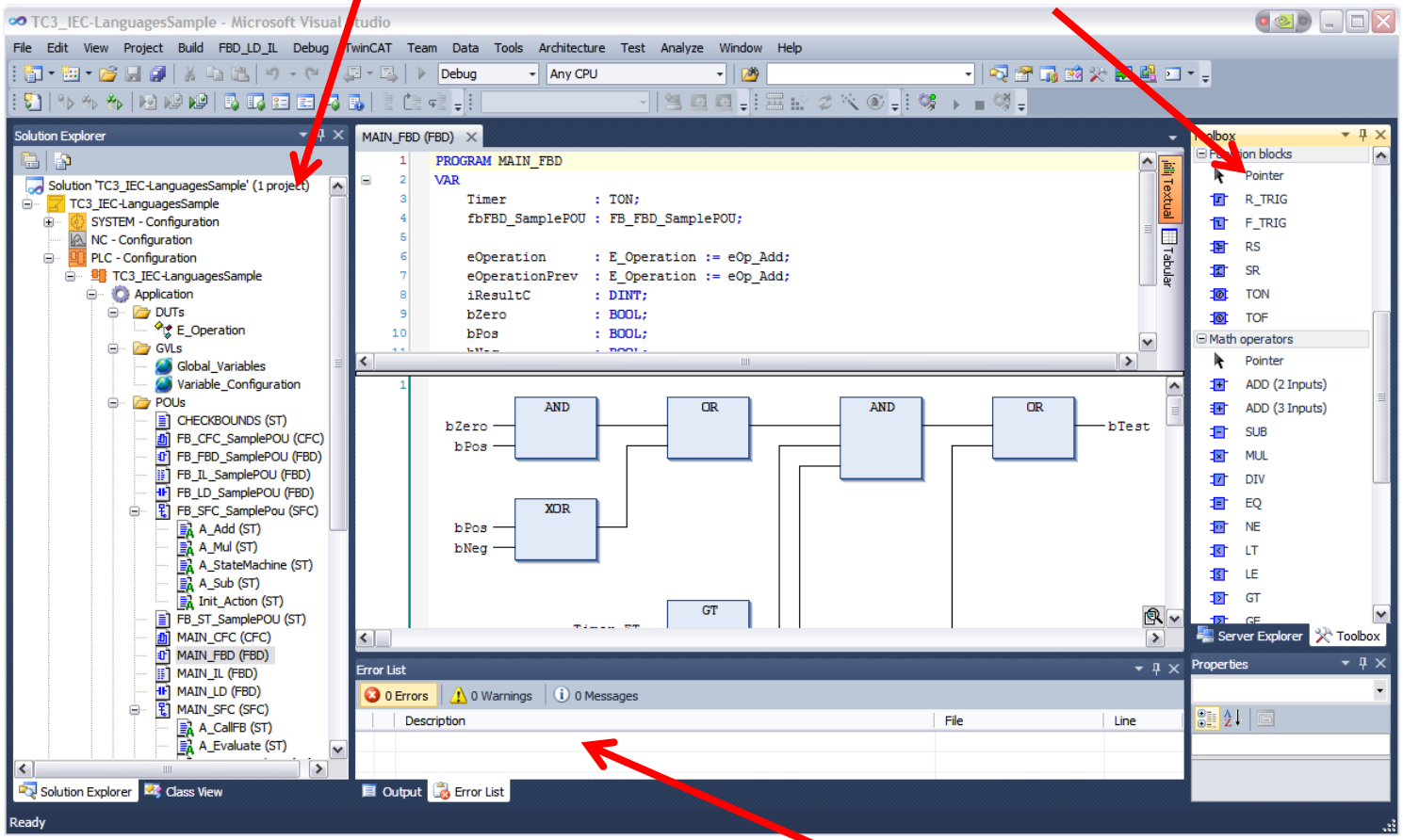
Programming:
- Languages of the IEC 61131-3 (IL, ST, FBD, LD, SFC) + CFC
- Usage of the object oriented extensions of the 3rd. Edition of the IEC 61131
- Multiple import and export interfaces
- No direct addressing necessary

Commissioning/ maintenance
- Source code up- und download
- Online change
- Full debugging functionality (breakpoints, monitoring, flow control,…)

TwinCAT 3 PLC

Shared tree structure for hard- and software

Programming language dependent toolbox



Shared output window for all languages

TwinCAT 3 PLC – New editor-properties – LD/FBD/IL

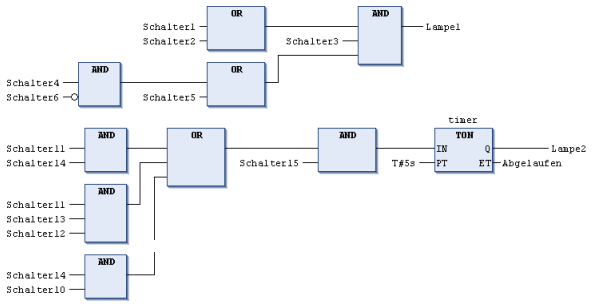
Only **one** editor for all 3 programming languages!

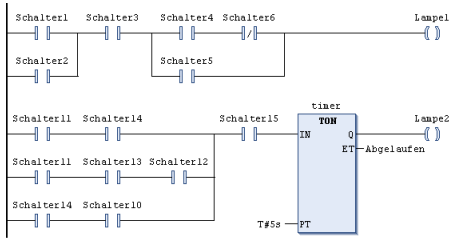
➔ Switchover without compile

- Also in Online mode
- Settings e.g. for symbol comments take effect on all views
- Mixed networks

- LD
  - Contact networks at all inputs
  - Coils at all outputs
  - Multiple function blocks in one network

- IL now chart-oriented



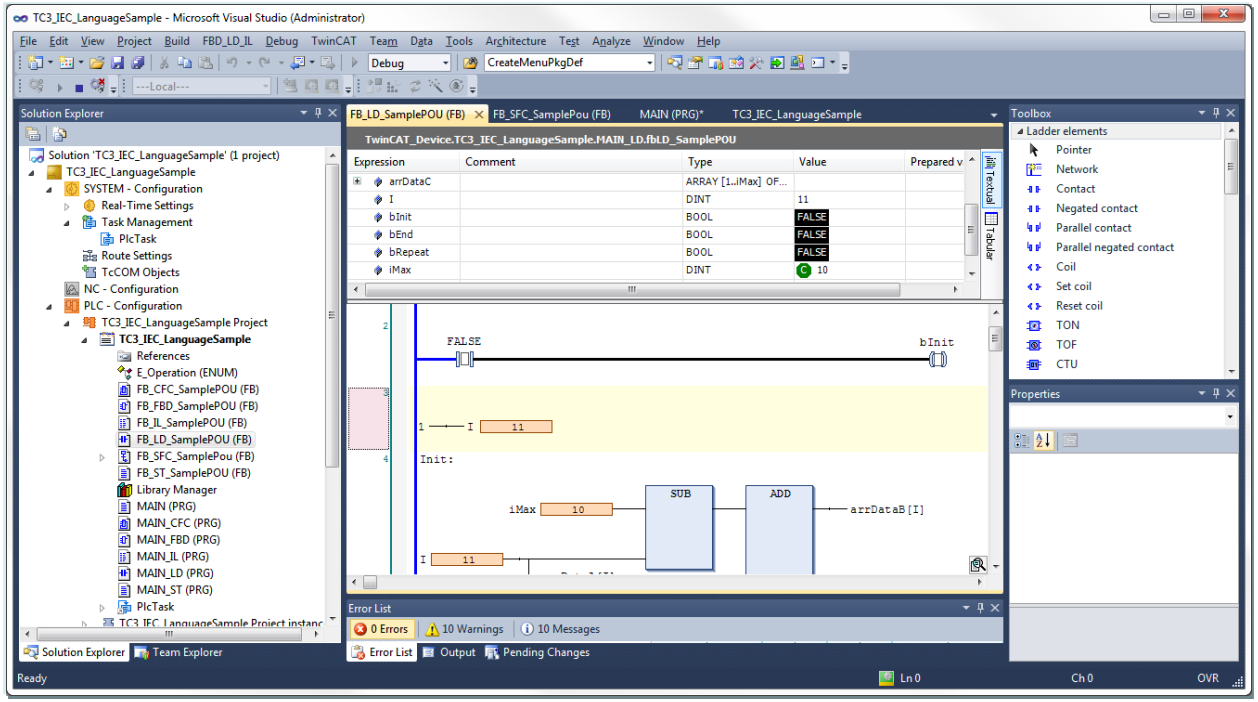



```
LD      Schalter1
OR      Schalter2
AND     Schalter3
AND(    Schalter4
ANDN    Schalter6
OR      Schalter5
)
ST      Lampe1
LD      Schalter11
AND     Schalter14
OR(     Schalter11
AND     Schalter13
AND     Schalter12
)
OR(     Schalter14
AND     Schalter10
)
AND     Schalter15
ST      timer.IN
CAL     timer(
        PT:= T#5s,
        ET=> Abgelaufen)
LD      timer.Q
ST      Lampe2
```

TwinCAT 3 PLC – New properties

Debugging

- Inline-Monitoring (especially for ST)
- Force much more improved
- Flexible handling of breakpoints (i.e. conditional Breakpoints)



TwinCAT 3 PLC – Object Oriented Extensions to IEC61131

Benefits of the object oriented extensions

- Increased readability of the code by encapsulation algorithms into methods
  → Increased maintainability

- Modularization, structuring of the code
  → Increased reusability

- Abstract programming by using interfaces
  → Increased extensibility and adaptability

- Construction of inheritance hierarchies
  → Increased extensibility and adaptability

Consistent usage of the object oriented extensions enables:

- Increased software quality
- Decreased time for programming and maintenance

Object orientation with the 3rd edition of IEC61131-3

Six new keywords:

- `METHOD` : Action on FB with own variables
- `PROPERTY`: POE-Pair for Set/Get of attribute
- `THIS` : in Method/Property for the current FB-instance
- `EXTENDS` : Inheritance between FBs
- `INTERFACE` : defines abstract objecttype (FB without implementation)
- `IMPLEMENTS` in the FB: instances over named interface callable

Call syntax for methods:

- *Objekt*`.`*Methodenname*`(`*...*`)`

Special methods:

- `FB_Init`, `FB_Exit`, `FB_Reinit`

Object orientation with the 3rd edition of IEC61131-3

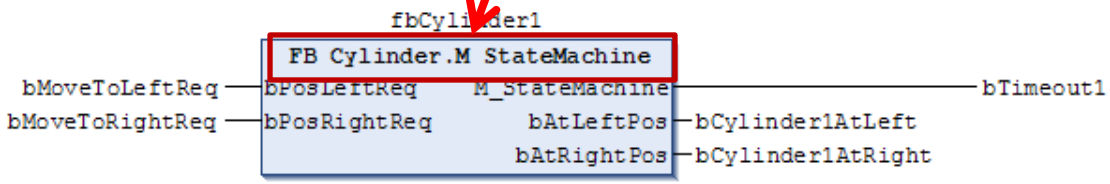
Language properties	2nd Edition IEC 61131-3	3rd Edition IEC 61131-3	C++	Java	C#
Multi language support	+	+	-	-	-
OOP/procedural mixed	-	+	+	-	-
Classes	~ (FB)	+	+	+	+
Methods	~ (Aktionen)	+	+	+	+
Interfaces	-	+	-	+	+
Partial abstract classes	-	-	+	+	+
Polymorphy	-	+	+/-	+	+
Reference semantics	-	+ (Interfaces)	-	+	+
Constructor / Destructor	-	+	+	+	+
Properties	-	+	-	-	+
Visibility	~ (Variables)	~ (Variablen)	+	+	+
Dyn. Memory („new“)	-	- (in TC3)	+	+	+

TwinCAT 3 PLC – Object Oriented Extensions to IEC61131

Object oriented extensions of the IEC 61131-3 3rd edition:

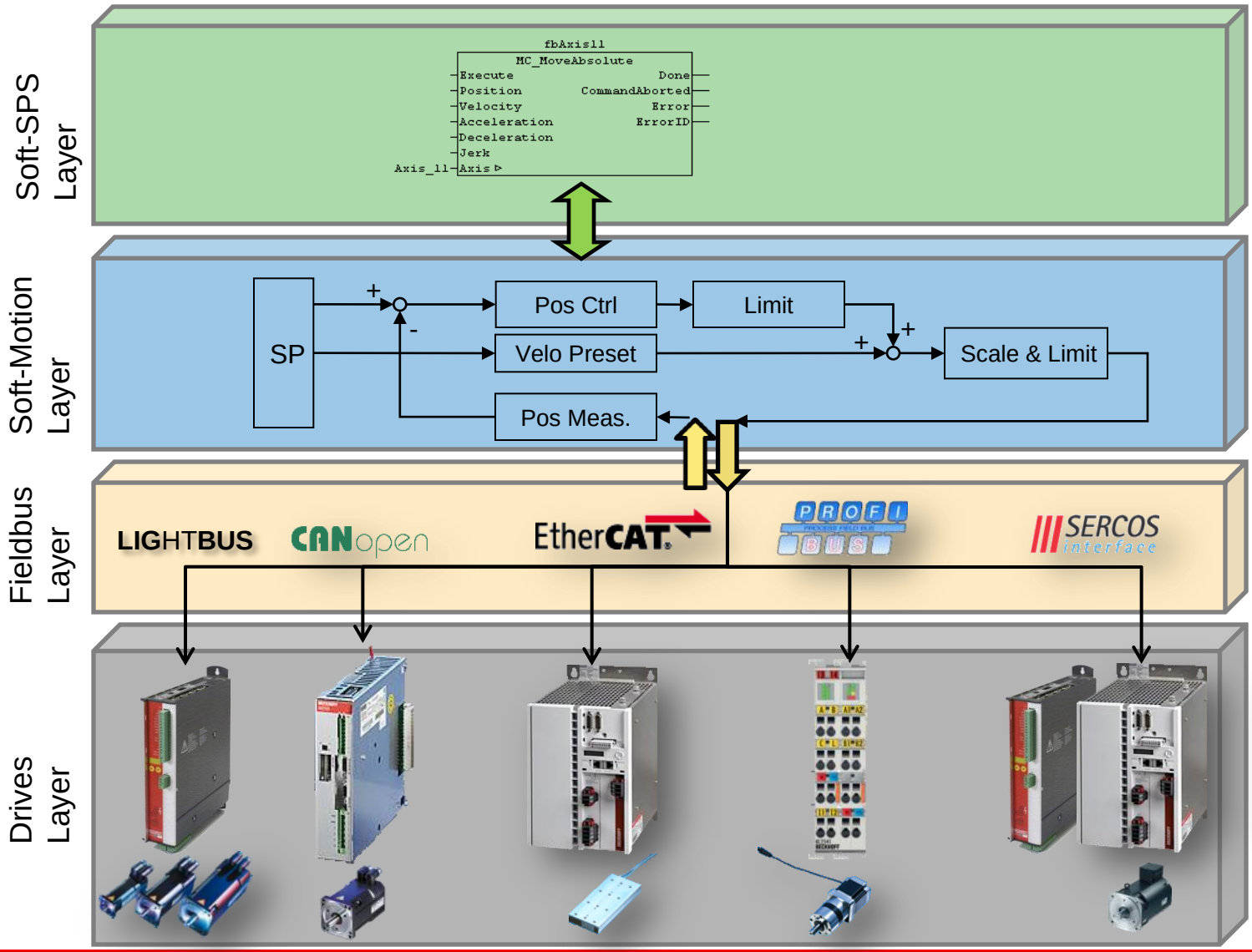
- Concept of the function blocks was extended by
  - Classes
  - Interfaces
  - Methods
  - Inheritance
  - Properties
  - Key words THIS, SUPER

```
FUNCTION_BLOCK AC_Meldung EXTENDS TWINCAT_MODULE_BASE_LIBRARY.BaseModule IMPLEMENTS IAC_MELDUNG, ITCCHILD
```



- Usage of the extensions
  - Is possible in all IEC languages
  - Independent from the used hardware
  - **Not mandatory**

TwinCAT 3 MC – Abstraction layer



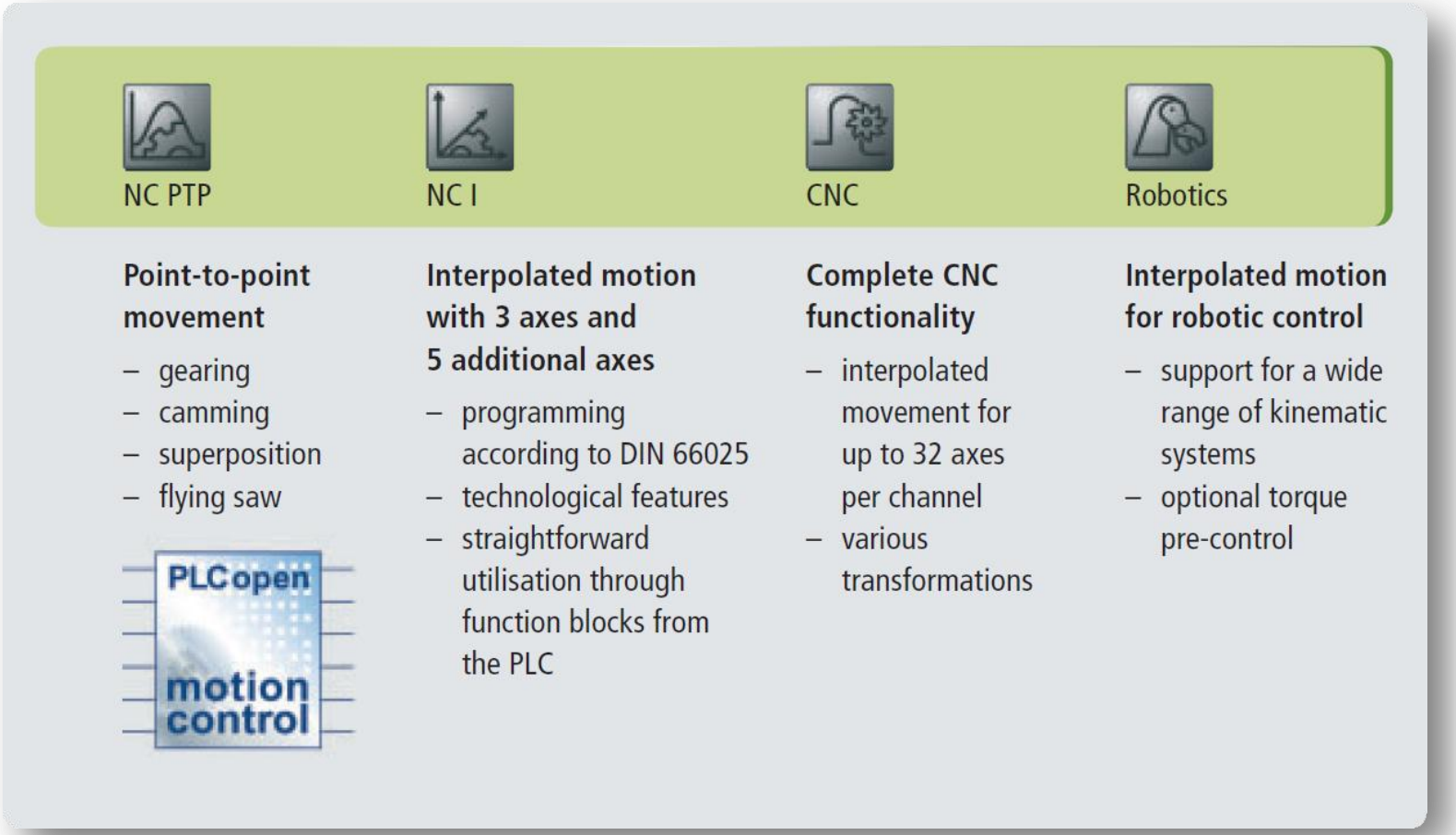
PLC
C/C++

NC PTP
NC I
CNC

IO

TwinCAT 3 MC – from PTP to Robotic Control

Functionality



S

C/C++ Programming languages

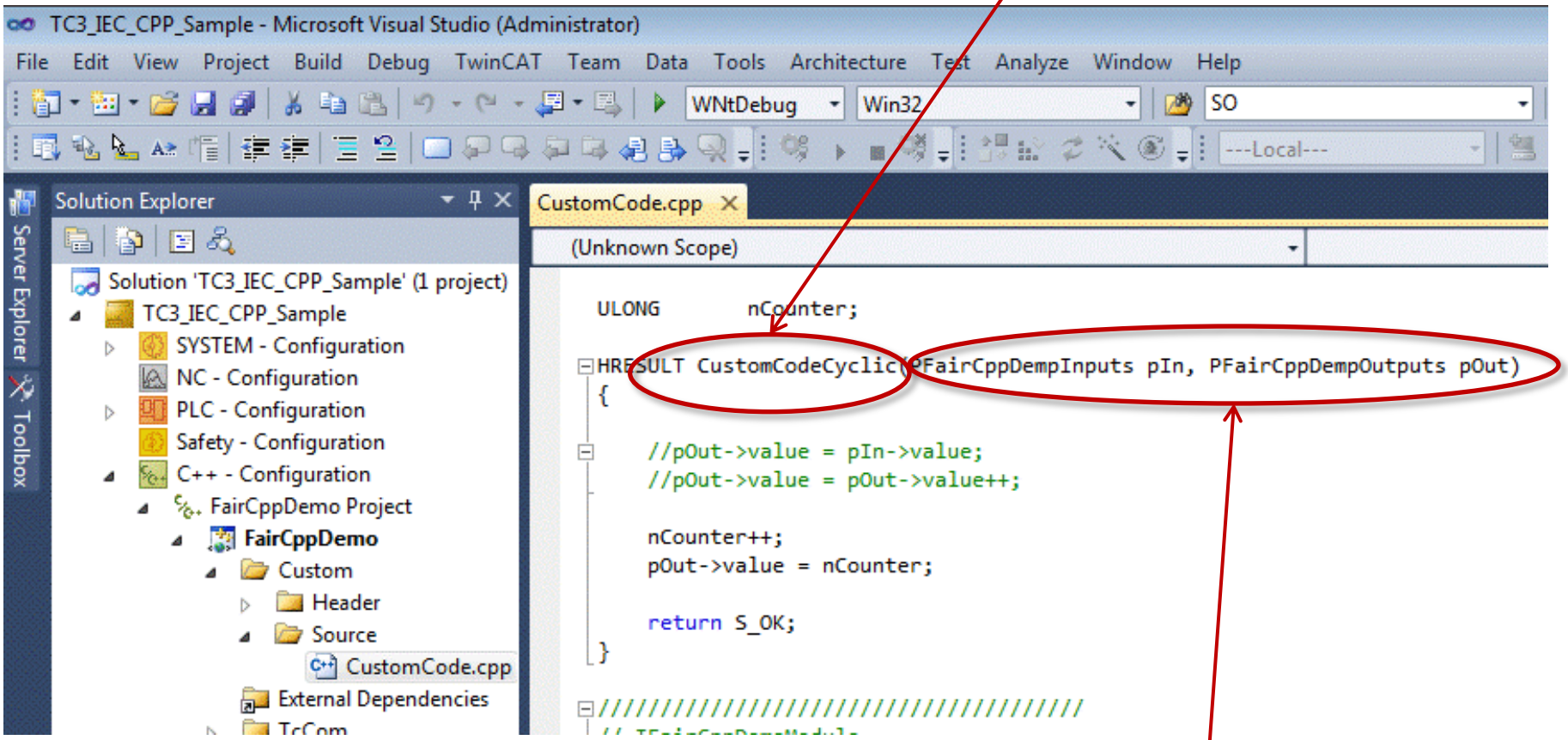
- Reuse existing C/C++ code
- Cooperation of C/C++ and PLC code
- Real time applications for all platforms (CE, XP…)
- Opens new areas – not standard PLC users
- Well known programming language
- Standardized (C: ISO/IEC 9899 TC3, C++: IEC 14882)
- Generation of Automation Device Drivers (ADD)
  Enables to implement own drivers (e.g. fieldbus drivers)
- Beckhoff SDK delivers functional range of (analog to PLC-Libraries)
  - ADS
  - Motion
  - File IO
  - ….

Application areas

- image processing
- robotics
- measurement technology
- …

C/C++ Programming languages

Method CustomCodeCyclic: – is called cyclically

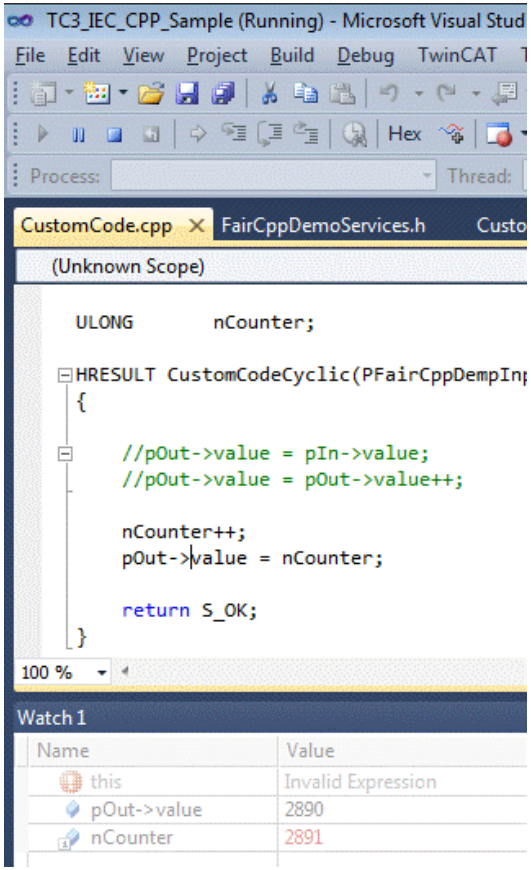


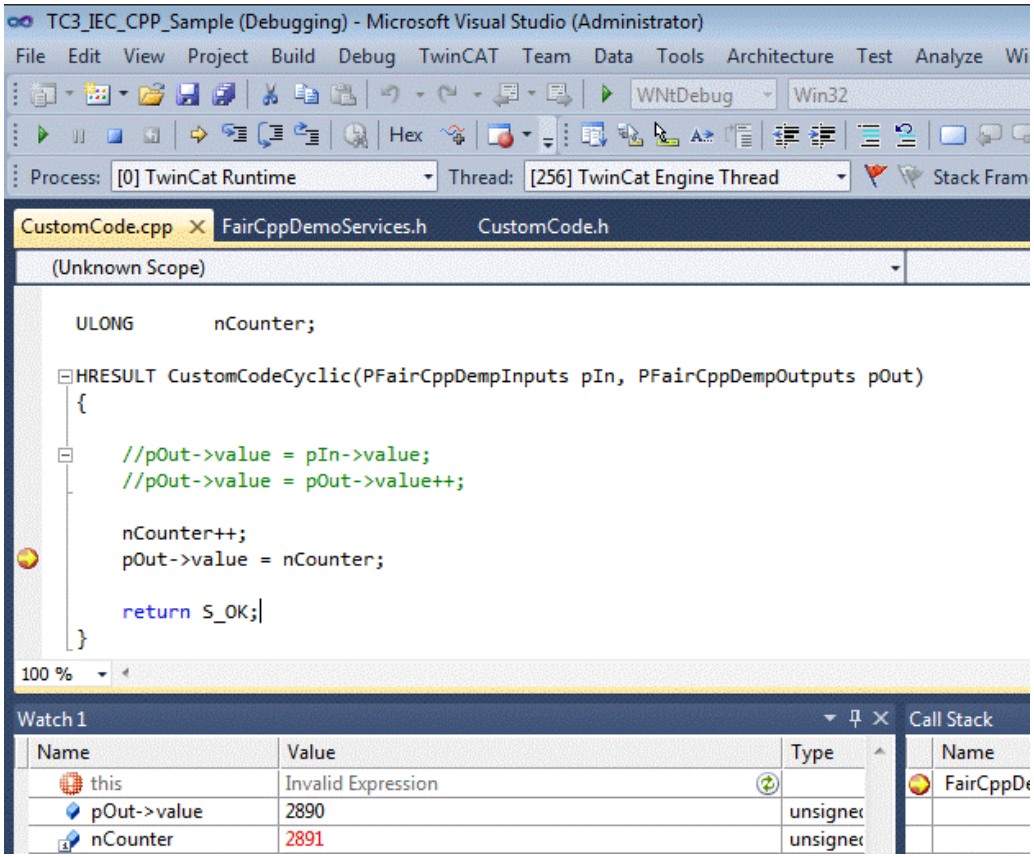
Pointer to logical input/output image

C/C++ Programming languages

VS2010 Standard debugger:

- Monitoring / Modification of variables onyl with breakpoint

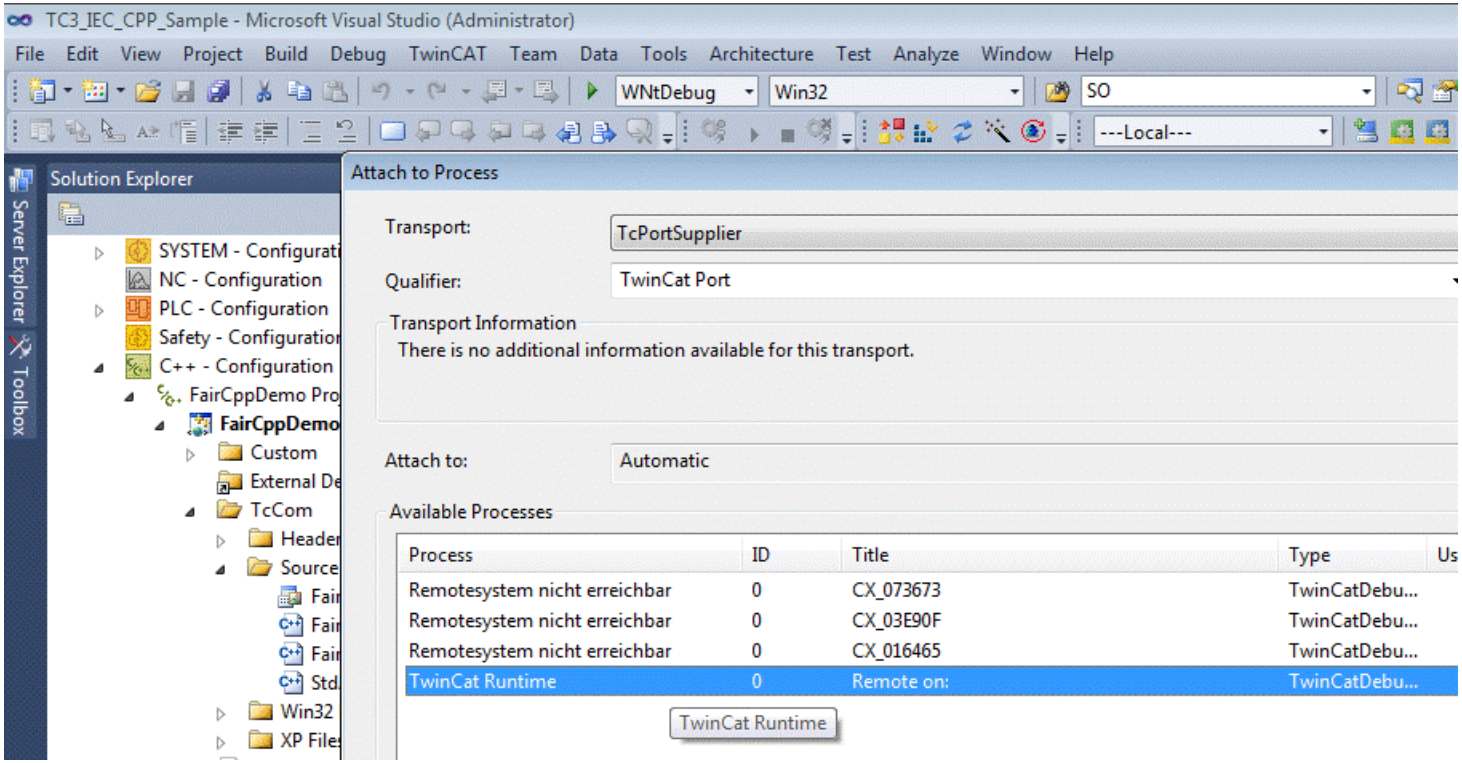




C/C++ Programming Languages

VS2010 Beckhoff Debugger

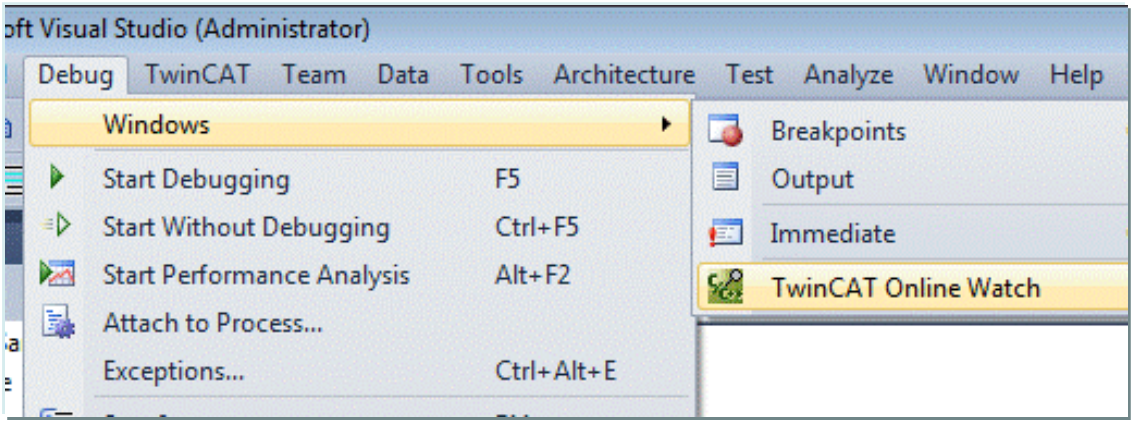
- Connection to C++-Target (Selection Target system)

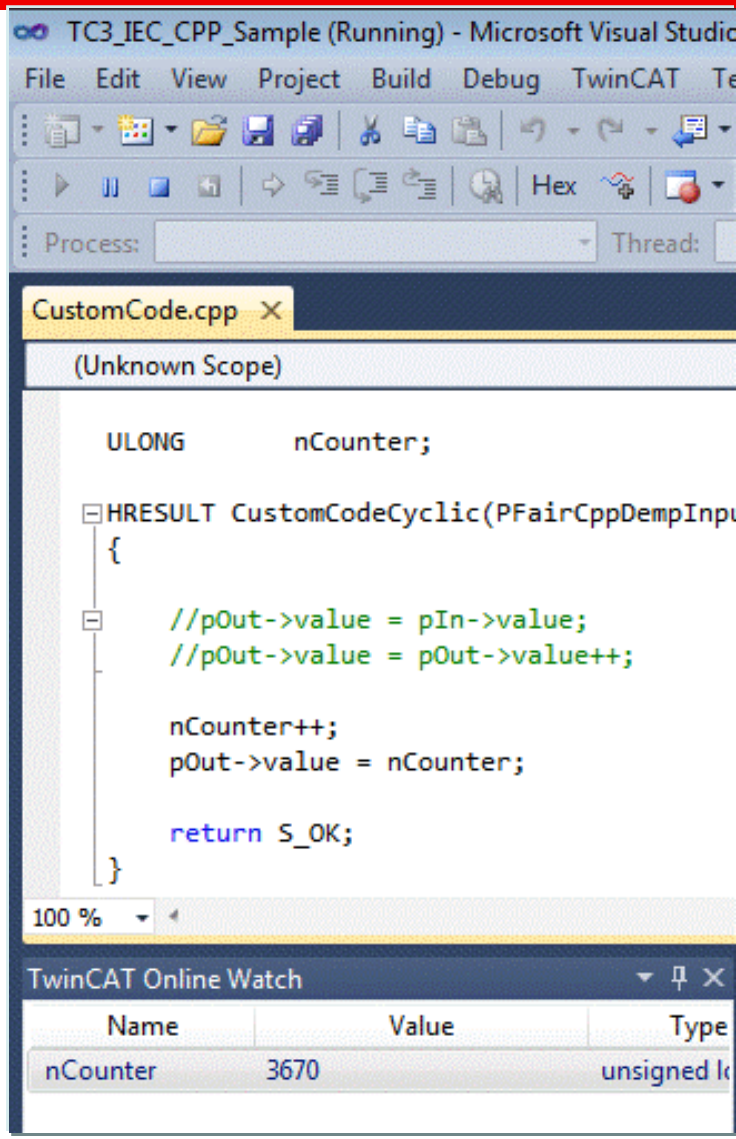


C/C++ Programming Languages

VS2010 Beckhoff Debugger

- Monitoring / Modification of variables without breakpoint
  (analog to PLC without breakpoint)



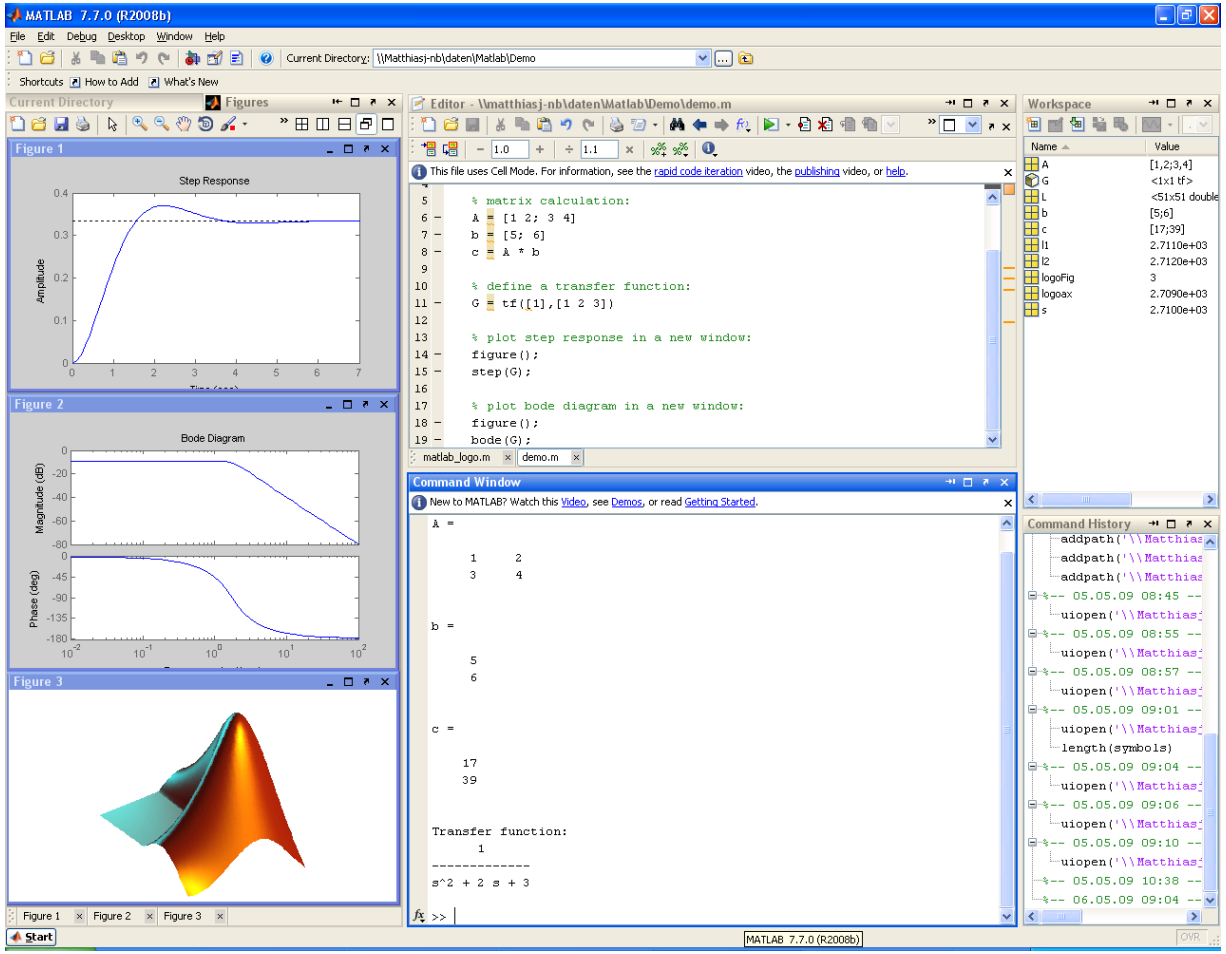


J

What is Matlab?

Matlab

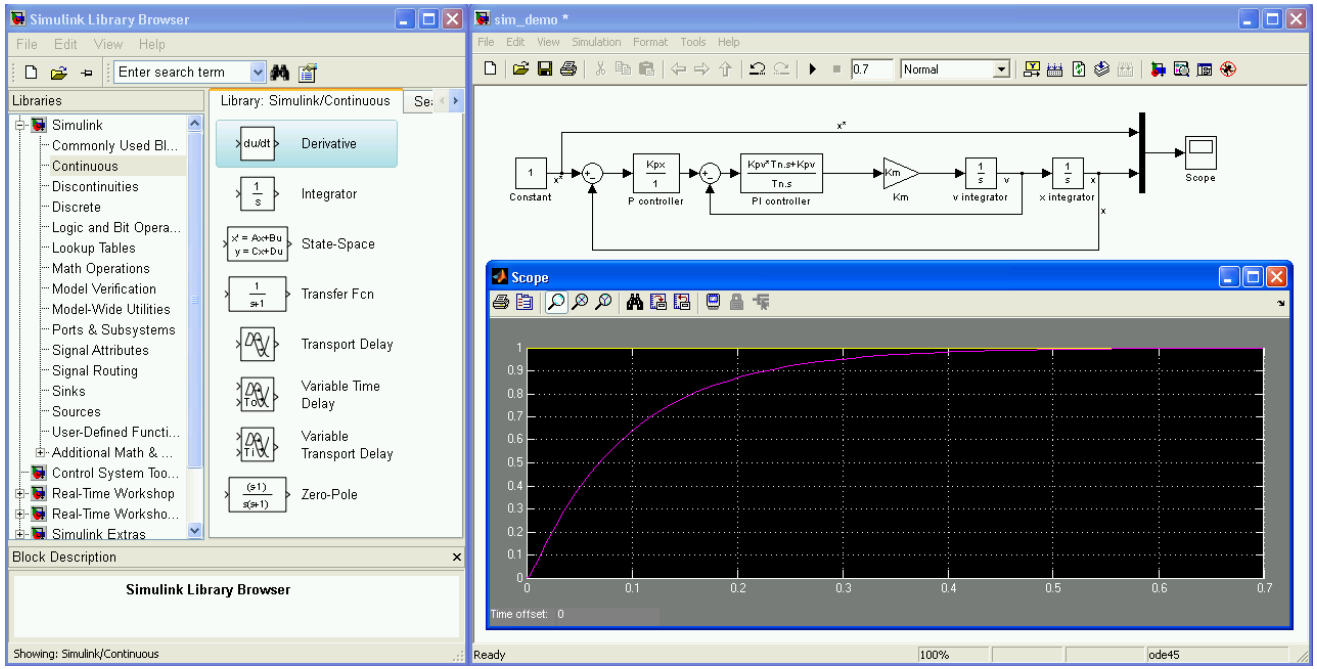
- Matrix-Operations
- Easy programmability
- Graphical data preparation
- Many special functions for a wide field of application
- Very common in the scientific/ university environment



What is Simulink?

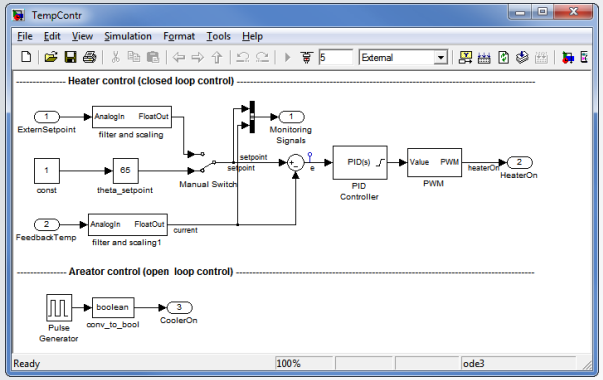
Simulink

- Simulation of dynamic systems
- Graphical programming, C-Code can be integrated optionally
- Control loop optimization for complex systems by means of simulations
- Multiple toolboxes accelerate the programming for special applications and different users (electricians, machinebuilders,…)



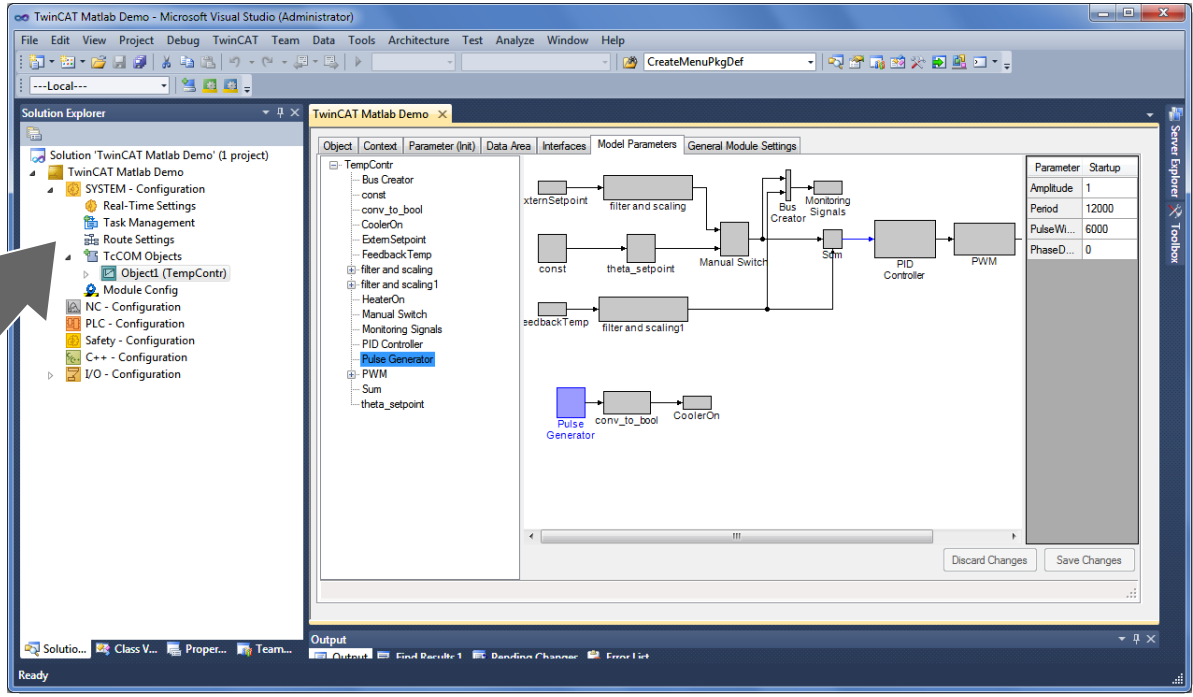
Configuration of the TcCOM-Module in TwinCAT

Simulink-Model



TcCOM-Modul

- exampleW32.dll für CE
- example.sys für NT/XP
- example.tmc



.NET Programming Languages (e.g. C#)

- Well known programming languages
- Standardized C# (ISO/IEC 23270)
- Creates intermediate code (Common Intermediate Language – CIL)

- Benefits:
  - Efficient engineering with higher abstraction level
  - Widely accepted
  - „garbage collection“ takes care on memory
  - Could now be handled as part of one integrated solution

- Restrictions:
  - Garbage collector is not suitable for real time applications

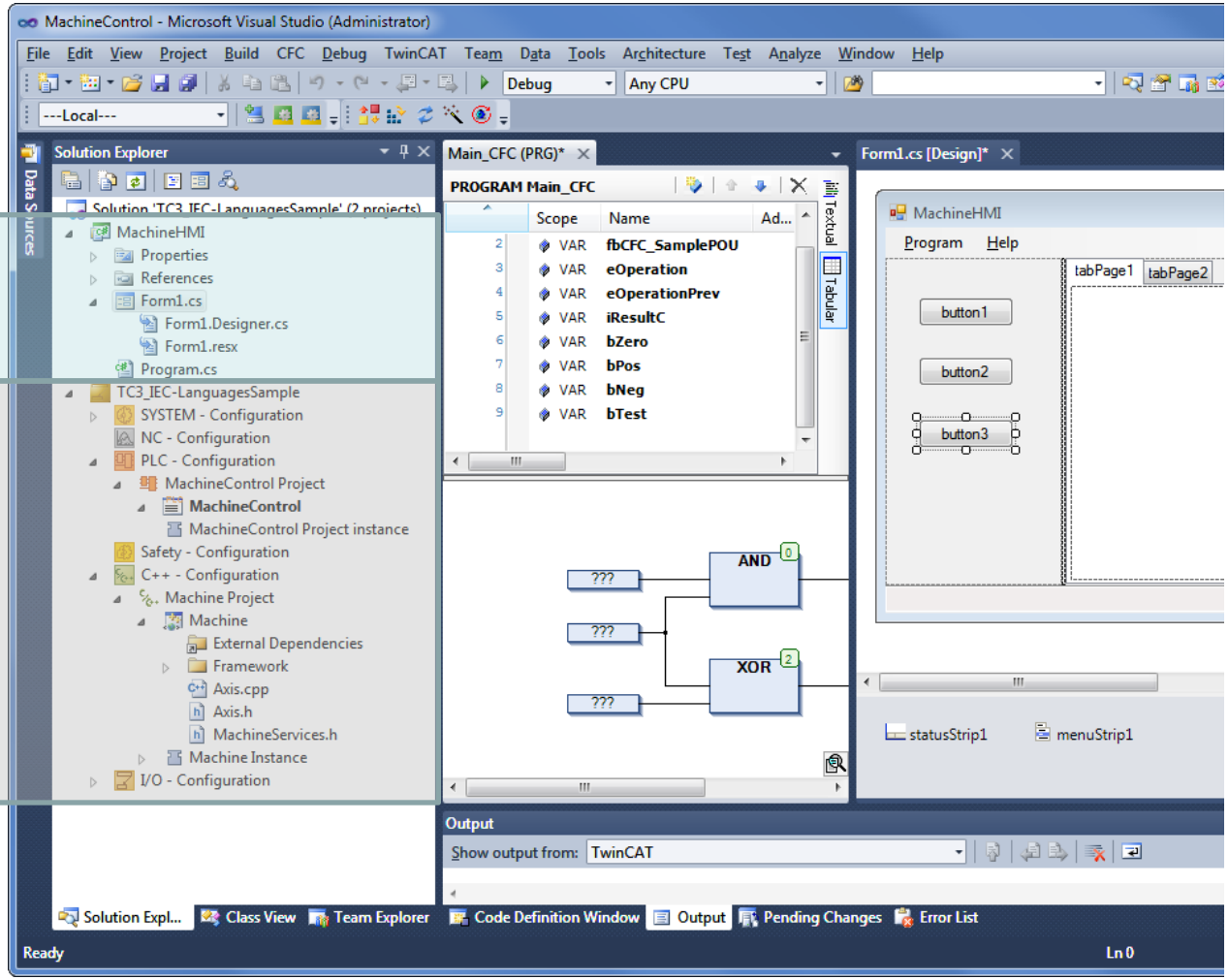
C#/ .NET programming

HMI / Windows processes

PLC module

C++ module

Real time Code



Silverlight: Evolution in design of the user interface

The Evolution of User Interface Design

1970s
First Era:
Command Line

Type

1990s
Second Era:
Graphical User Interfaces

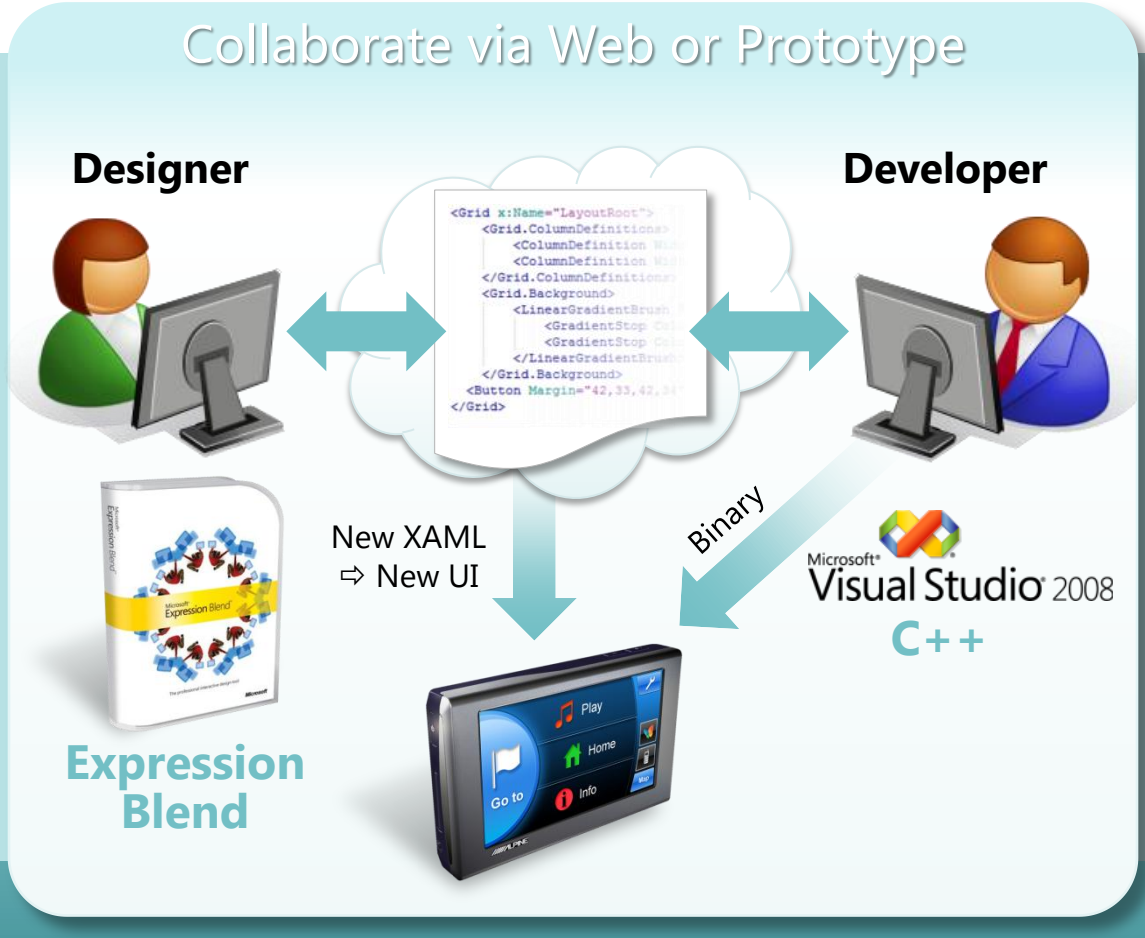
Point-and-click

Now
Third Era:
User Experiences (UX)

Engaging, touch sensitive graphical environments with video and sound

Consumers buy based on your product's UX, not specs. The next-generation cell phones and media players changed the playing field.

Silverlight For Windows Embedded Development Environment



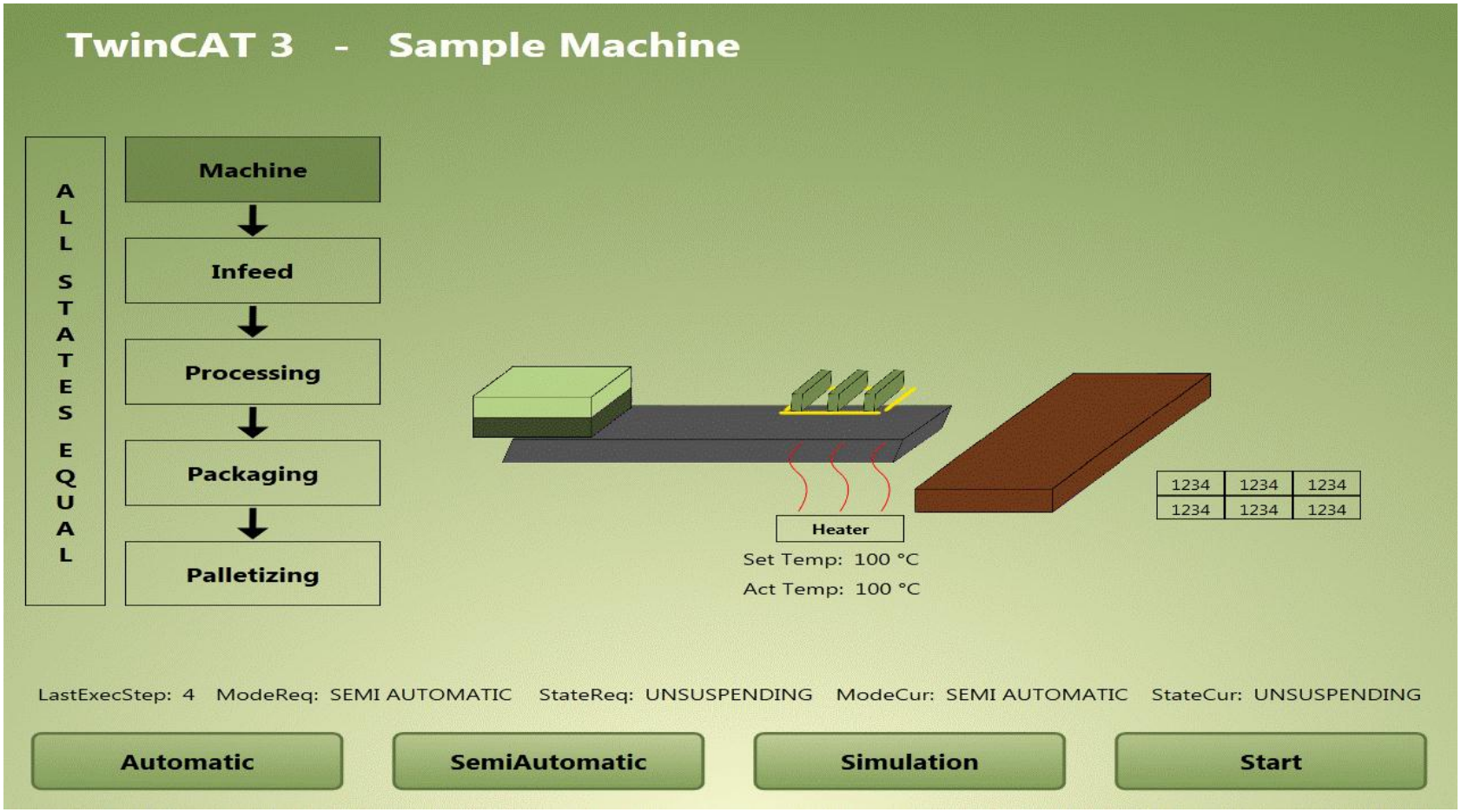
The SWE environment is a C++ XAML user interface framework for Windows CE, based on Silverlight

Designers focus on design tools such as Expression Blend

Developers focus on tools such as Platform Builder and Visual Studio

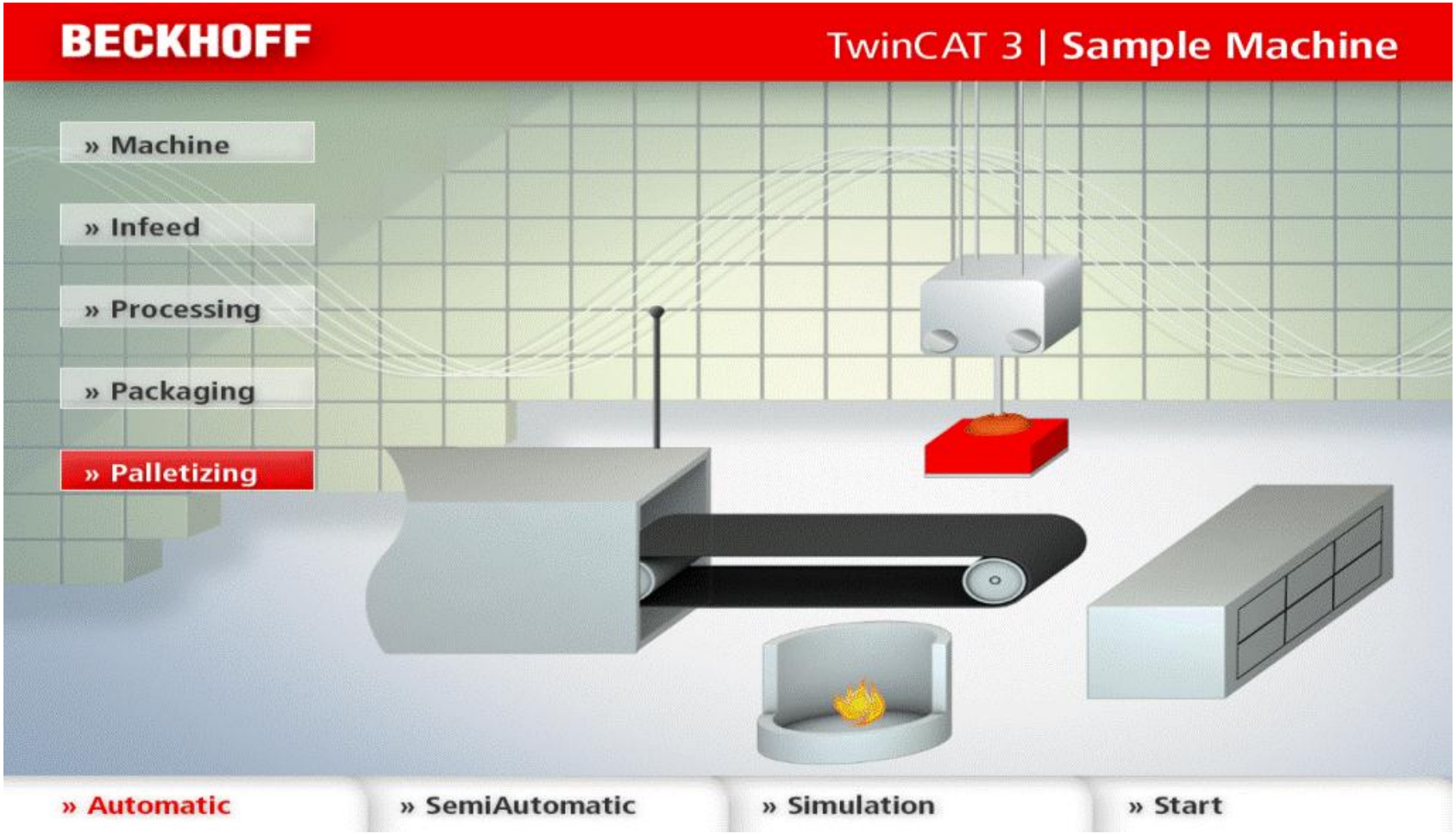
C#/ .NET programming

Silverlight: The new UI experience !



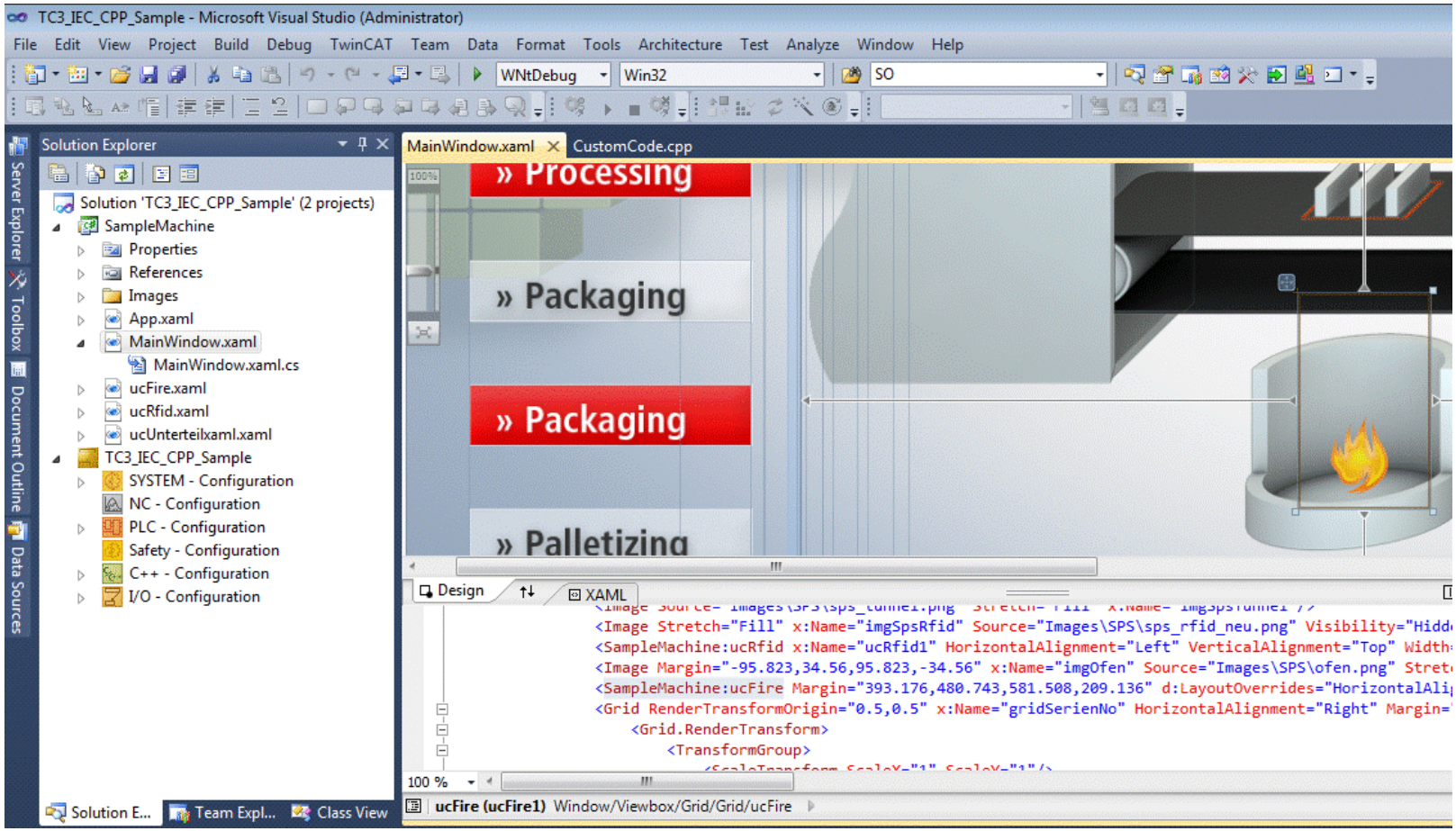
C#/ .NET programming

Silverlight: The new UI experience ! Second try !



C#/ .NET programming

Microsoft Silverlight:

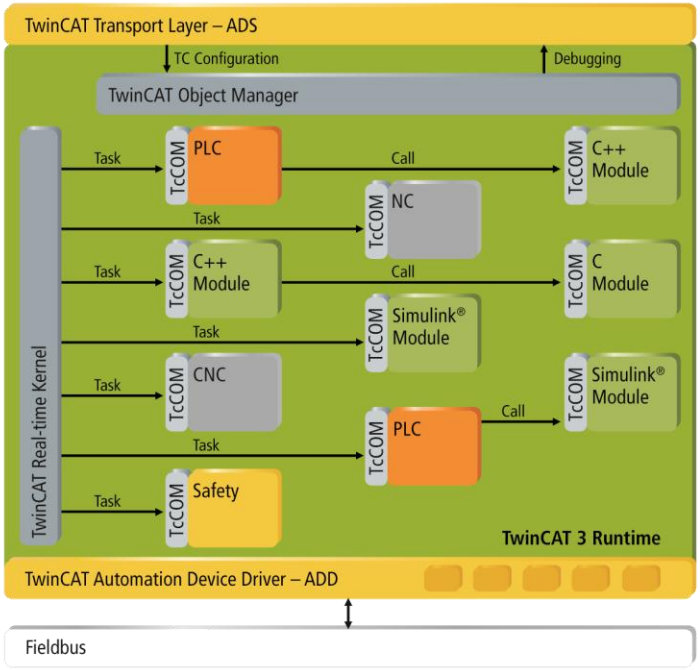


J

eXtended Automation Runtime (XAR)

Modular runtime

- **Dynamic** environment for the execution and administration of TwinCAT 3 modules
- Administration of runtime modules (by TwinCAT Object Manager)
- defined Interfaces (e.g. TwinCAT Component Object Model –TcCOM) and behavioral model



caption:

↔ ADS-communication

↔ communication

eXtended Automation Runtime (XAR)

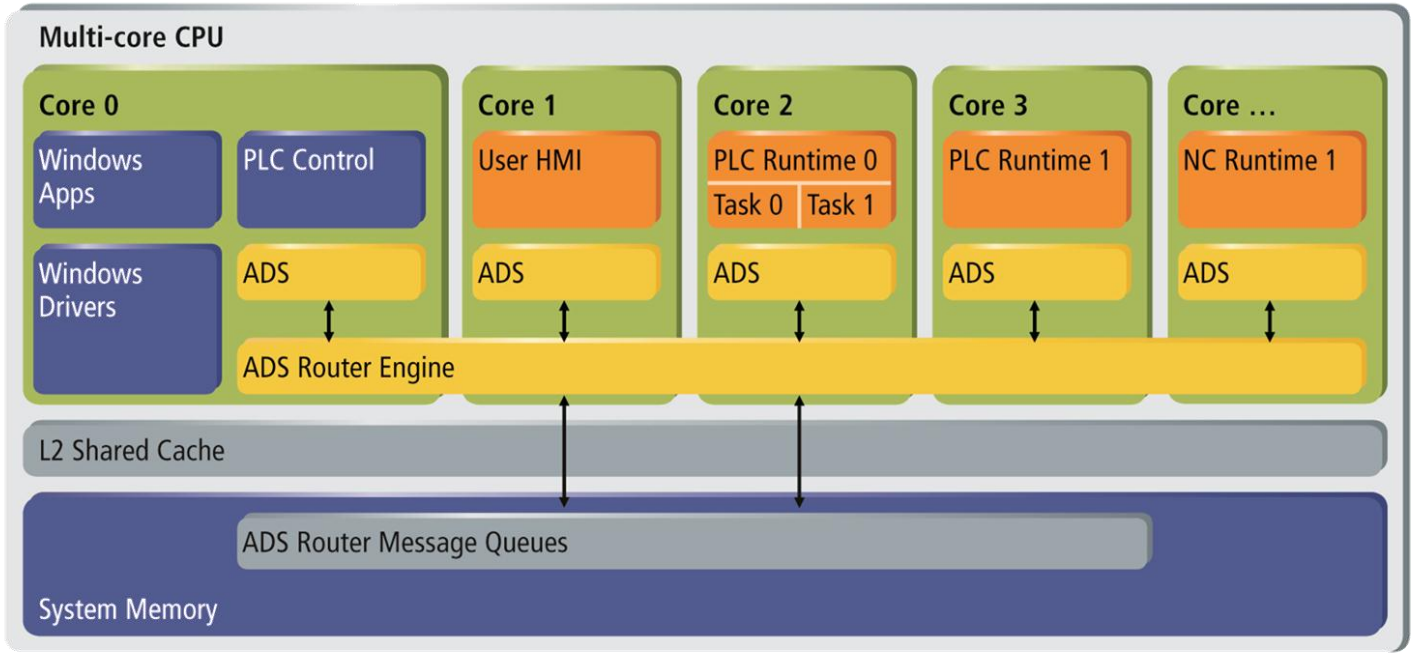
Modular runtime Interface

- Separation of encapsulated functionality into modules
- Extension of the base system by own drivers (Automation Device Drivers – ADD) e.g. fieldbus drivers
- Scalability: modules can contain simple functions, complex algorithms and real time tasks or complete projects
- Reusability of modules
- Cooperation of modules written in
  - IEC 61131-3
  - C/C++
  - Matlab generated modules

eXtended Automation Runtime (XAR)

Support of multi-core systems

- Distribution of projects to cores (e.g. PLC, NC, Motion Control and HMI run on different cores)
- Scalable base time for each core

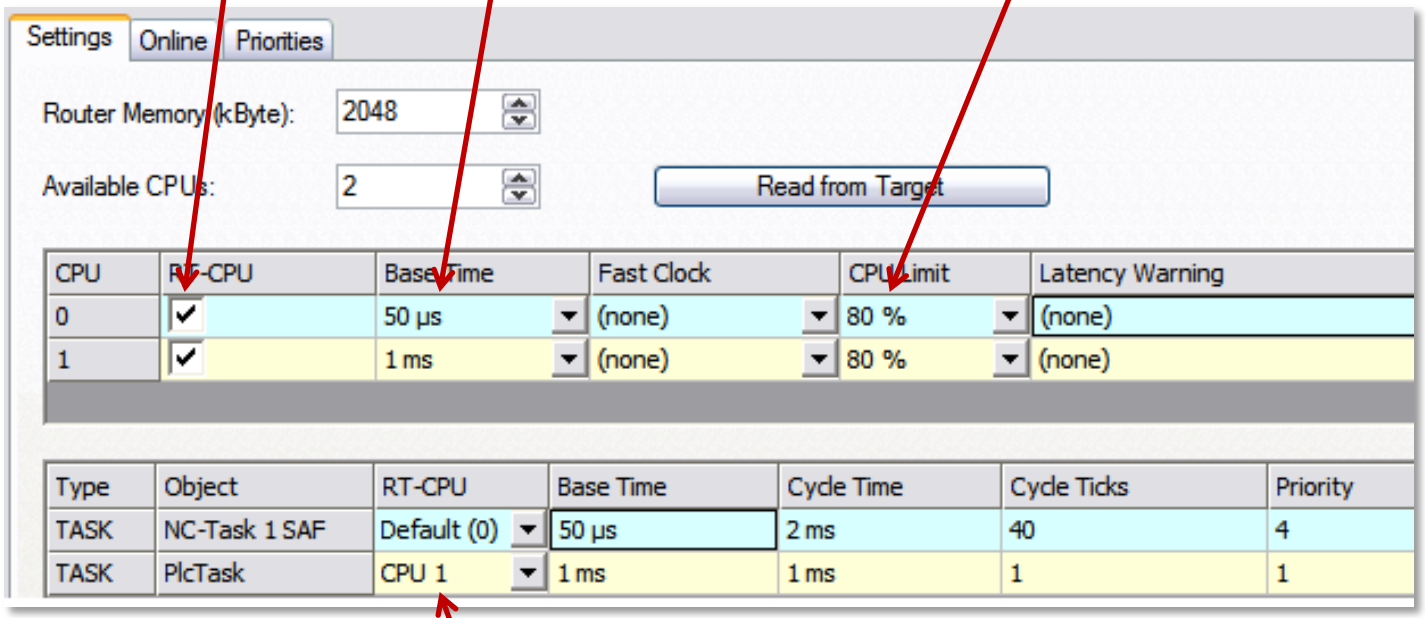


eXtended Automation Runtime (XAR)

Enables cores for real time usage

Defines the base time for a core

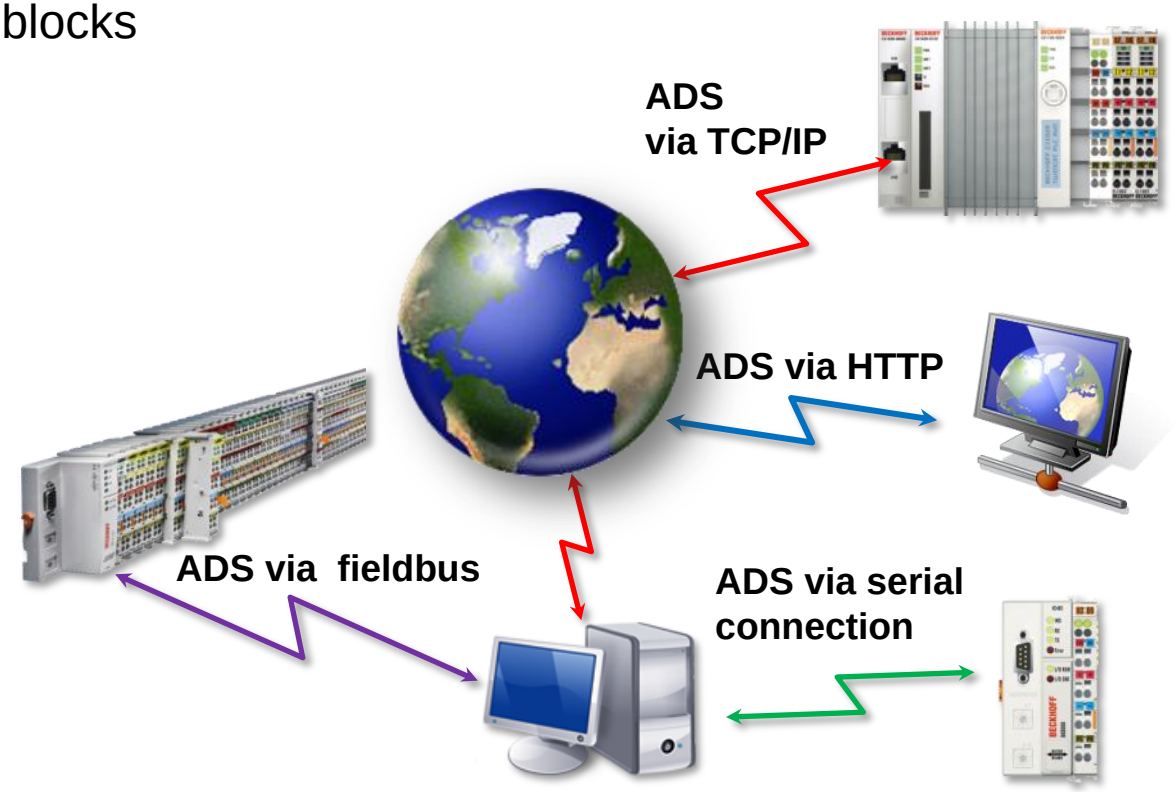
Defines the CPU-usage limit



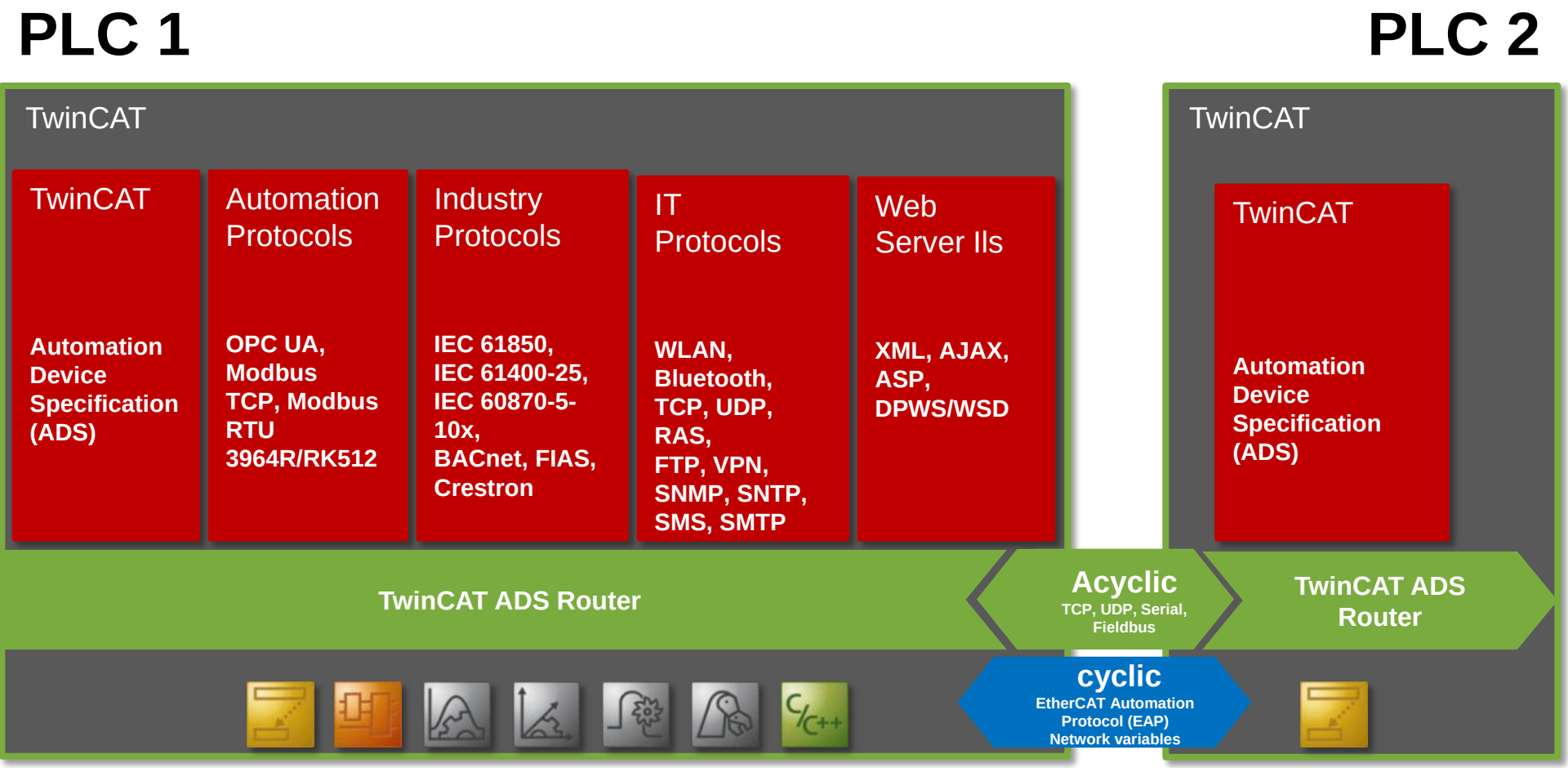
Assignment of a task to a CPU

Connectivity | ADS (Automation Device Specification)

- Consistent, vertical, horizontal
- Cyclical/event-driven
- Data exchange and/or commands
- Open protocol with example code
- Available for major Windows platforms
- Access from PLC via function blocks
- Routable via: local/network
- Components free of charge:
- OCX/DLL/.NET/ Script/Webservice



Connectivity – different solutions, all based on ADS

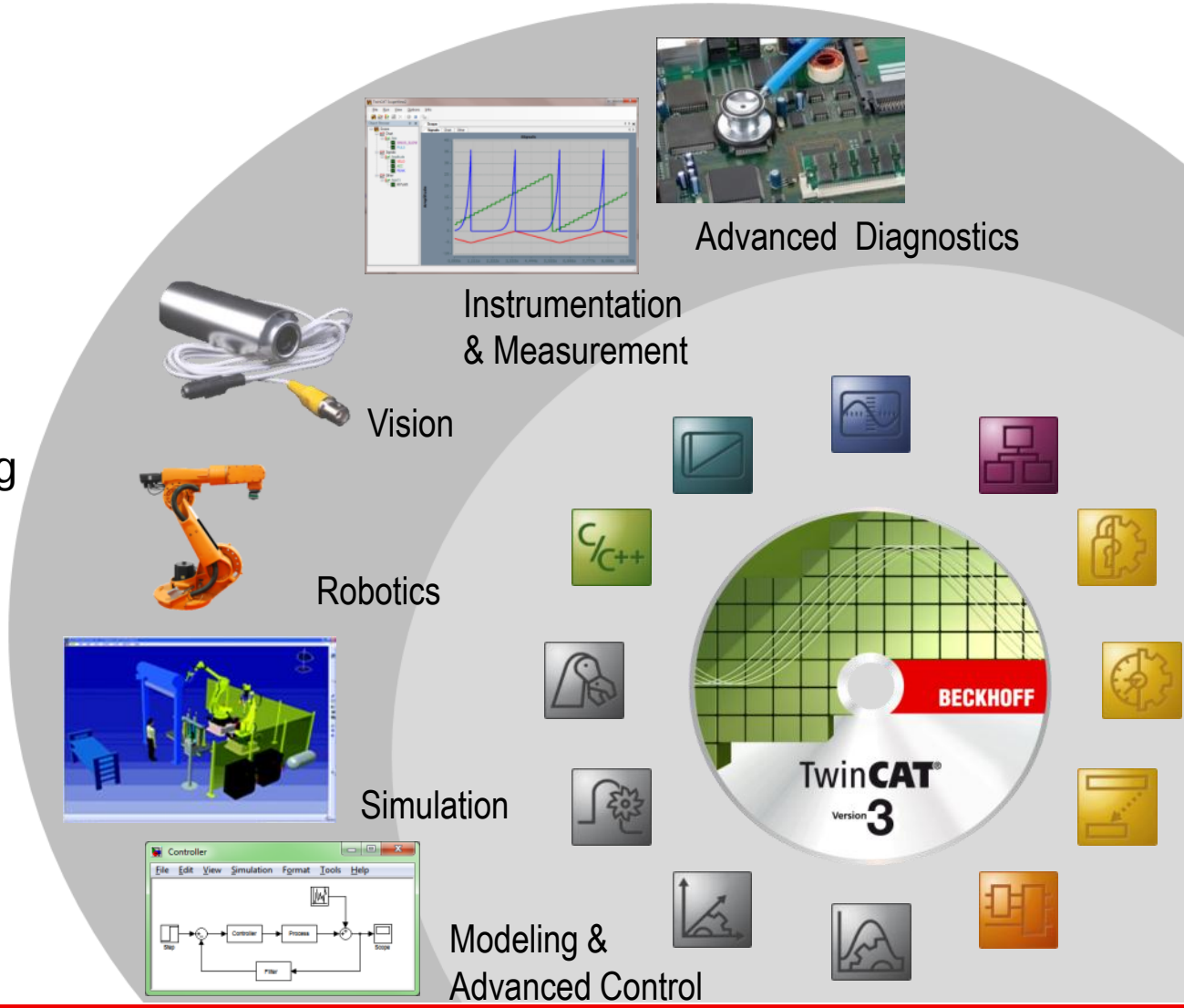


J

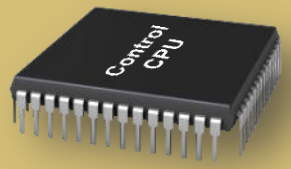
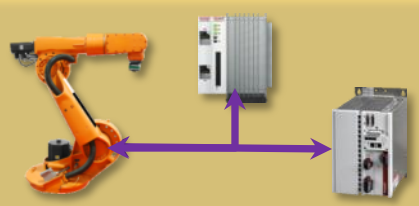
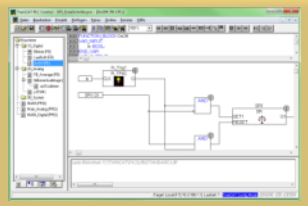
Open and modular platform

TwinCAT extends Automation into new fields of application

- Instrumentation & Measurement
- Advanced control
- Rapid control prototyping
- Simulation/real-time
- Data Analysis
- Test bench Automation
- Scientific Automation



Convergence of Technologies

	Automation Technology (AT)		Information Technology (IT)		PC- based Control
1986		+		→	
1996	Special RTOS	+	Windows Embedded	→	
2003		+		→	
2010		+	Microsoft Visual Studio PARTNER	→	

TwinCAT | The Windows Control and Automation Technology

IPC

I/O

Ethernet TCP/IP

RS232
RS485

DeviceNet

USB

PROFIBUS

INTERBUS

Modbus

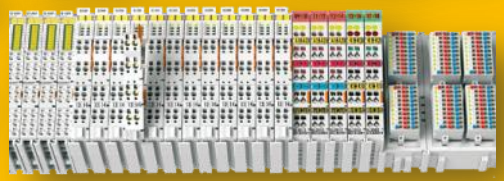
LIGHTBUS

CANopen

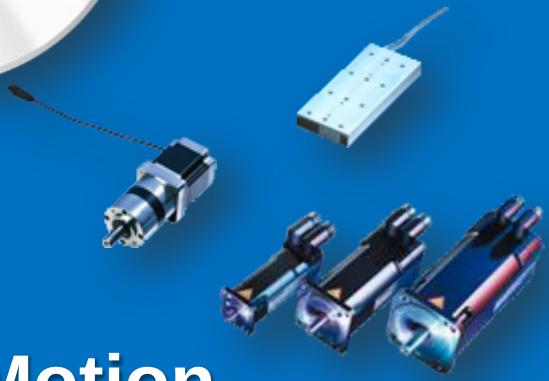
ControlNet

EtherCAT

SERCOS interface



Application



Motion

Thank you!