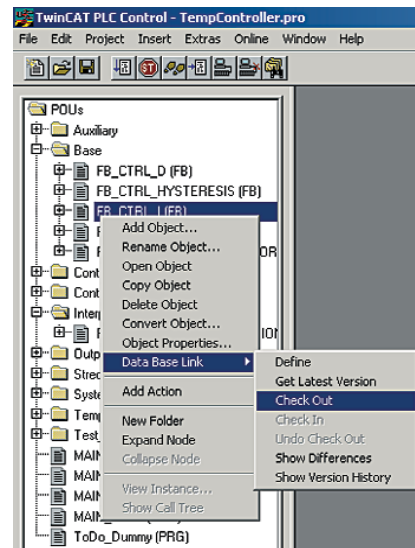


# TwinCAT Engineering Interface Server (ENI)



Source code management  
via Visual Source Safe  
or MKS Source Integrity

→ When several programmers work on a PLC project, the problem was how to integrate the work of the different programmers in a single project. Up to now, this was only possible by comparing, importing and copying. The TwinCAT Engineering Interface (ENI) enables tasks to be co-ordinated via a source code management system.

For the coordination of different PLC projects, the TwinCAT Engineering Interface Server is installed on a central server within the network. It provides a "shell" for a variety of source code management systems. Under this shell, Microsoft Visual Source Safe or MKS Source Integrity may be operated. A driver for CVS source code management is in preparation; drivers for others tools can be developed by the user.

In addition to versioning PLC objects such as blocks, data structures and variable lists, the TwinCAT Engineering Interface enables several programmers to work on the same project (although obviously not on the same object). Another option would be to use the documented interfaces for creating an application that enters or retrieves objects into/from the database. Complete projects can also be created within the database without manual intervention. Application generators can save the PLC programmer a lot of work during the creation of a PLC project for a new machine including blocks, data structures and lists. In combination with a little "glue logic", commissioning can be achieved very quickly.

## Versioning of projects and objects

The main feature of the TwinCAT Engineering Interface is versioning of projects and objects. How often have two programmers modified the same block simultaneously? How often have changes been lost due to lack of coordination? How often has the question been asked why a particular change was made in a particular object? Who did it and when? And of course: What was changed? All these questions not only arise for PLC projects, but also for all software developments. Appropriate tools for versioning of software pieces have therefore been used for many years. For Unix, SCCS, RCS and CVS have prevailed. Some of these tools have also been ported to the Windows world. Developers using Microsoft tools will be familiar with Microsoft Visual Source Safe. For distributed developments, MKS Source Integrity and Rational Clear Case have prevailed.

The TwinCAT Engineering Interface represents a bridge between classic software development and PLC software development. The Engineering Interface is used for TwinCAT PLC Control in the same way as Visual Source Safe or Rational Clear

Case is used for the associated C++ or VB development tools, without suffering the familiar disadvantages of source code databases. Communication between PLC Control and the server is based solely on the transmission of an XML file via http port 80. This means that no firewalls have to be specially configured, and no domain limits have to be adhered to. Distributed development, perhaps with out-sourced PLC code development, is conceivable.

## Working in a familiar environment

The PLC programmer only sees his familiar PLC Control. While there are a few more settings and icons, the programmer doesn't have to worry about the database. He is not even aware which database is being used. The Engineering Interface completely screens the PLC programmer from the source code database, while he can utilize its benefits. Objects with different version numbers written in any of the available programming languages can be compared, and a particular project state can be labelled. The labelled project state can be reconstructed at any time, irrespective of the version numbers of the individual objects. The object tree at a glance shows which objects are subject to source code management, which objects have been checked out, and which objects are locked, i.e. checked out by other users.

The server component obviously has to be configured accordingly. To this end, TwinCAT ENI contains configuration, management and diagnostics programs. The communication route and the associated database have to be selected, and the connected users have to be configured with passwords and individual rights. A diagnostic tool offers the option of displaying which user is carrying out what tasks in what objects, independent of the database used.

The familiar standard software development tools for source code management are now also being used for PLC software. TwinCAT PLC Control, the normal programmer tool, includes a connection to the TwinCAT ENI. This shell, which always is the same as far as PLC Control is concerned, includes a connection to various familiar source code management programs. All familiar features of these programs are supported.