

TwinCAT MTP for cyber-physical modularity in the process industry

# Automatic code generation enables efficient module engineering



With TwinCAT MTP, automation software from Beckhoff meets the requirements for increasing plant modularity in the process industry.

The Module Type Package (MTP) provides a modern solution to modularize a plant in the domain of process industries. To achieve this, MTP defines an interface for the simple orchestration and connection of modules to a higher-level control system. With TwinCAT MTP, Beckhoff enables MTP project planning and automatic code generation. This solution opens up an integrated and efficient way to develop modules in the universal TwinCAT engineering environment.

Rigid automation systems that operate unchanged for many years are increasingly becoming obsolete in the process industries; in fact, individuality and flexibility in production are getting higher attention. Through cyber-physical modularization, process plants can be realized with little effort and flexibly adapted. In this way, companies can effectively overcome modern challenges such as the increasing fluctuation of commodities markets. This requires the shortest possible time-to-market through accelerated development cycles as well as the profitable and individualized production of even small batches.

Manufacturers of machines and production modules usually have to meet project-related customer requirements for connection to a higher-level production system. The use of a module type in different projects therefore entails increased integration or development work. At the same time, this means significant expenditures in time and money for plant development, which usually results in rigid automation systems. Future-proof plants in process technology, on the other hand, require significantly more flexible architectures.

### **Modularity promotes reusability**

A suitable solution is the modular design of production plants and the resulting ability to reuse individual modules. The entire process is broken down into individual sub-processes and mapped by corresponding modules. With a separate controller per module a complete modularization of the plant can be reached. The connection of the individual modules to a higher-level control system (e.g. a DCS) enables the orchestration and therefore the mapping of the entire process.

As a result, the development work shifts from plant engineering to module engineering, enabling flexible modification of the plant with little effort, depending on current requirements. This way, additional modules can easily be added or already existing ones can be rearranged or removed. Complete reprogramming is not required, as most of the logic is implemented in the individual modules. The higher-level controller only takes care of their orchestration. It is therefore called the Process Orchestration Layer (POL).

In practice, the concept is implemented by a vendor-independent standard for the description of process modules: the Module Type Package defined in the

VDI/VDE/NAMUR 2658 standard. The MTP file contains all the information necessary to integrate a process module into a modular plant. This includes the functionalities in the form of services, communication and an HMI template. The interfaces defined in the MTP standard enable plug-and-produce behavior, eliminating the overhead of reusing a module. In this way, the modules are developed only once and can then be integrated into different plants independent of the used controller and POL.

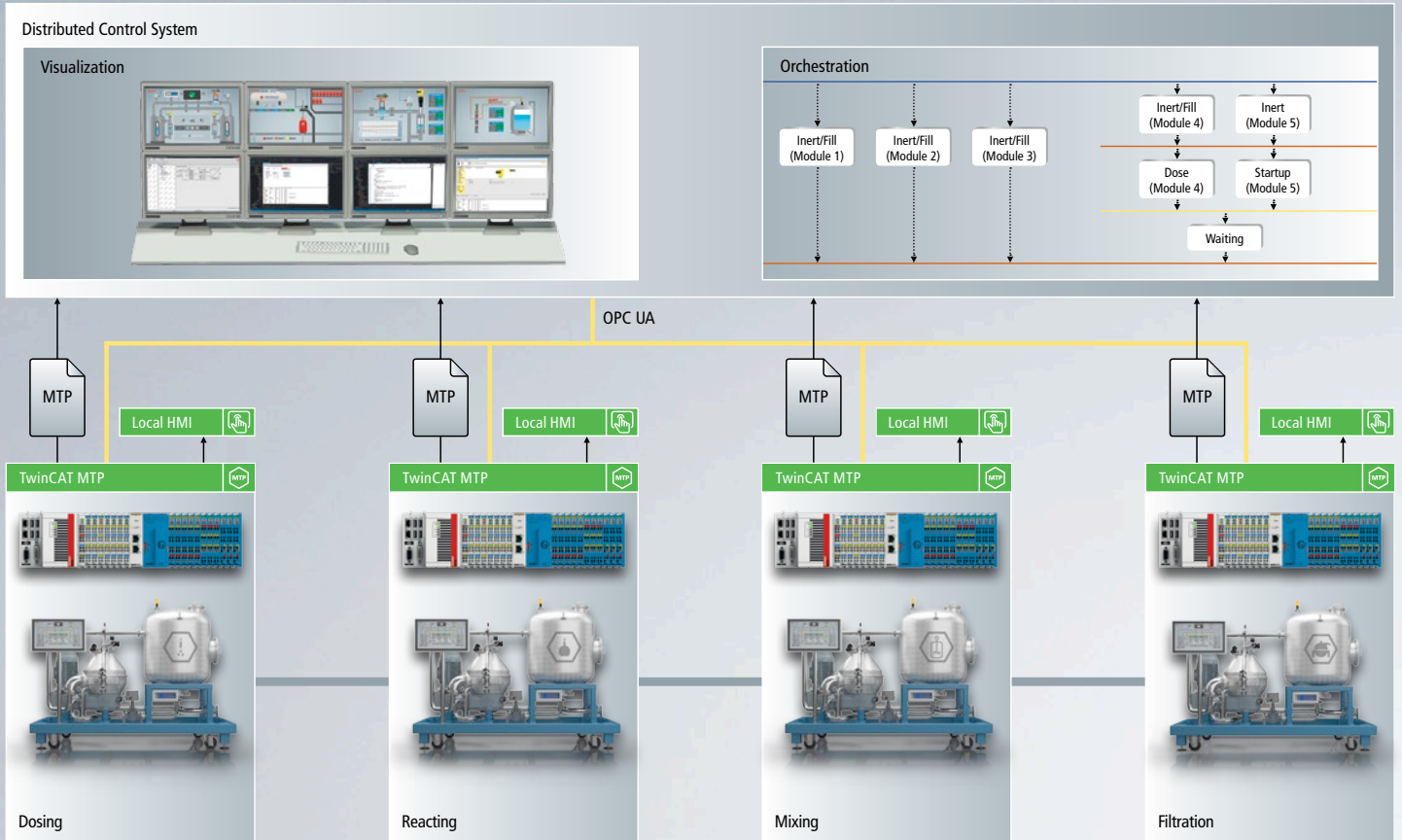
### **The Module Type Package**

The MTP concept is based on intelligent modules with an own controller and describes an interface between the module and a higher-level POL. The MTP itself is a module description in the form of an archive file that is exported from the module engineering and imported into the higher-level control system. The file contains all information necessary for the POL to establish a communication with the modules via OPC UA. So far, the following aspects have been modeled in the MTP description of a module:

- Services: A service-based architecture encapsulates the process functionality of the module; for example, reaction, filtering or dosing. The services of the different modules are orchestrated in the POL.
- Service relationships: To prevent the individual services from interfering with one another, dependencies can be defined. For example, the reaction service will be prevented from starting when dosing is currently taking place.
- HMI template: The P&ID (Piping and Instrumentation Diagram) of the module is stored as a structural description in the MTP, so that the POL can automatically generate the actual user interface for the system from this information with a uniform look and feel. The process values are integrated into this generated user interface via the OPC UA protocol.

### **MTP integration in TwinCAT software**

With TwinCAT MTP, the engineering of MTP-capable modules for process plants integrates directly into the familiar engineering environment. It offers all options, from the definition of the module and the import/export of an MTP to automatic code generation. The goal in developing this new TwinCAT function was to minimize the required standard expertise and enable the module developer to focus on the actual control logic.



### Structure of a modularized process engineering plant

The MTP concept is integrated into TwinCAT by the products TwinCAT MTP Runtime and TwinCAT MTP Engineering. The TwinCAT Engineering Environment includes a project management into which the TwinCAT MTP Engineering integrates itself as its own project type. On the runtime side, TwinCAT MTP includes an IEC 61131-3 library, which is used by the generated source code to implement a standard-compliant interface. The function blocks described in this library ensure their MTP-compliant representation via the OPC UA interface. TwinCAT OPC UA, which is used for this purpose, thus enables data access from the POL according to the standard definition.

### MTP workflow with TwinCAT MTP

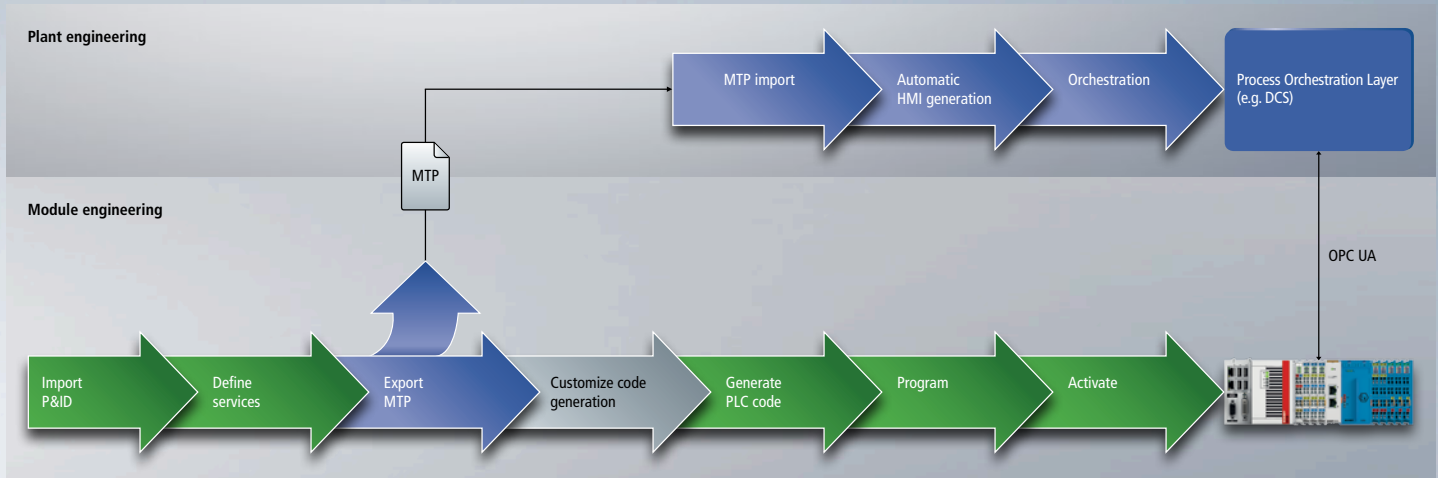
A module manufacturer can start by importing an existing MTP or external data. This is especially valuable when other planning tools are used in previous steps, for example, in order to define the P&ID of a module. In the project management, the first step is to design the module by defining the services, HMI objects and additional information. The complete MTP can subsequently be exported and later used by a POL to control the module. As a next step the mentioned PLC template is generated. The programming effort for the module

developer is minimized among other things by automatically implemented services and service dependencies. Thus, the developer only has to implement the process-specific functionalities in the service states.

The modular approach of TwinCAT Engineering and the integration of TwinCAT MTP provides a smooth transition between module definition and controller programming. This is especially evident in the flexibility of code generation, which is ensured by adaptable plug-ins. These plug-ins can be loaded into the TwinCAT XCAD interface and, in addition to pure PLC code, e.g. complete I/O configurations can be automatically generated.

### Conclusions

The Module Type Package meets the requirements of modern process technology and offers both the module manufacturer and the plant manufacturer the possibility of cost-effective module and plant engineering. This is ensured in particular by the standardized interfaces of the MTP, so that modules no longer have to meet project-specific communication requirements. Instead, the higher-level automation system can import the interface description, directly display a visualization and orchestrate the stored services.



The module engineering workflow with TwinCAT MTP

## Workflow in detail

The MTP support in TwinCAT is characterized by an efficient workflow:

- 1. Import P&ID:** With TwinCAT MTP, an MTP can be imported and used as an interface to a P&ID editor. Alternatively, the TwinCAT MTP Automation Interface can be used to connect proprietary data sources. The interface provides an API for the module manufacturer in order to gain programmatic access to the MTP project. Existing data from P&ID editors can thus be used, even if they do not offer an MTP export.
- 2. Define services:** The services as well as other module aspects such as service dependencies can be defined conveniently.
- 3. Export MTP:** The MTP can be exported directly from TwinCAT and then used by a POL to control the module.
- 4. Individualized code generation:** Optionally, the TwinCAT XCAD interface offers possibilities to individualize the code generation and to further process the generated information in a targeted manner.
- 5. Generate PLC code:** In the next step, TwinCAT MTP generates a PLC template. This is based on the TwinCAT MTP library and includes comprehensive logic via which the pre-configuration of all standard requirements takes place.
- 6. Programming:** The pre-configured code is finally completed by the state programming of the previously defined services.
- 7. Activate:** When activating the configuration, TwinCAT automatically ensures that the OPC UA communication can be established by the POL as described in the MTP.



Dr. Henning Mersch (see left), Product Manager TwinCAT, and Laurids Wetzel, Business Management Process Industry, both Beckhoff Automation

In TwinCAT MTP, Beckhoff offers an integrated and easy-to-use solution for creating a valid MTP. The appropriate, automatically generated PLC template can be completed with little effort in the engineering environment of TwinCAT to form a standard-compliant module controller.

Published in atp magazin 11-12/2020, Vulkan-Verlag, [www.vulkan-verlag.de](http://www.vulkan-verlag.de)

More information:

[www.beckhoff.com/twincat-mtp](http://www.beckhoff.com/twincat-mtp)