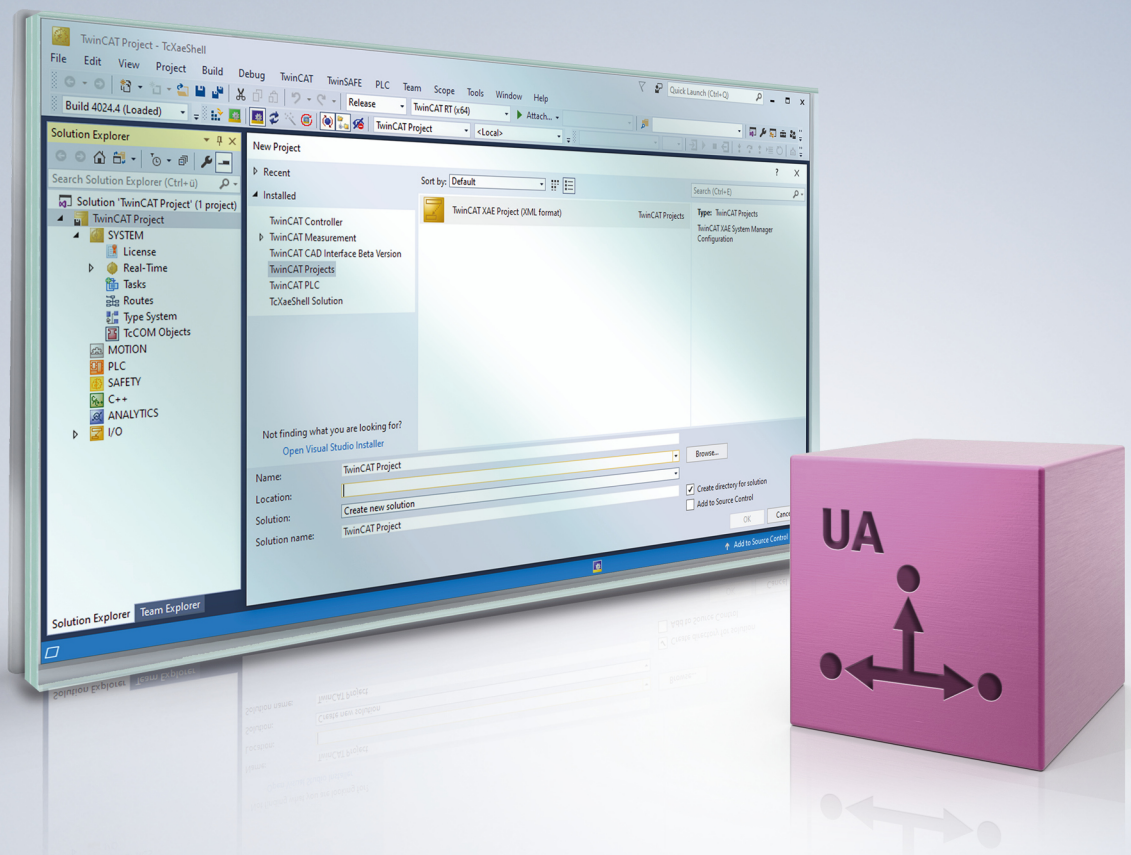
BECKHOFF New Automation Technology

Manual | EN

TF6100

TwinCAT 3 | OPC UA Configurator



2025-03-17 | Version: 1.1.0

Table of contents

1 Foreword 5
1.1 Notes on the documentation 5
1.2 For your safety 6
1.3 Notes on information security 7
1.4 Documentation issue status 7
2 Overview 8
3 Installation 10
3.1 System requirements 10
3.2 Installation 10
4 Technical introduction 14
4.1 Quick start 14
4.2 Application directories 20
4.3 Visual Studio 21
4.3.1 Overview 21
4.3.2 Creating a new project 21
4.3.3 Connecting to a server 22
4.3.4 Performing the server initialization 24
4.3.5 Adding ADS devices 25
4.3.6 Reading and writing the configuration 27
4.3.7 Importing and exporting configuration files 29
4.3.8 Configuring historical access 30
4.3.9 Configuring Alarms and Conditions 31
4.3.10 Configuring alarm texts 34
4.3.11 Configuring endpoints 36
4.3.12 Trust relationship for certificates 36
4.3.13 Configuring security settings 37
4.3.14 Restarting the server 45
4.3.15 Logging 45
4.4 Standalone 47
4.4.1 Overview 47
4.4.2 Connecting to a server 47
4.4.3 Performing the server initialization 48
4.4.4 Adding ADS devices 48
4.4.5 Reading and writing the configuration 50
4.4.6 Configuring historical access 50
4.4.7 Configuring Alarms and Conditions 52
4.4.8 Configuring alarm texts 53
4.4.9 Configuring endpoints 56
4.4.10 Trust relationship for certificates 57
4.4.11 Configuring security settings 57
4.4.12 Restarting the server 60
4.4.13 Logging 60
5 Appendix 62

5.1 ADS Return Codes 62
5.2 Support and Service 66

1 Foreword

1.1 Notes on the documentation

This description is intended exclusively for trained specialists in control and automation technology who are familiar with the applicable national standards.
The documentation and the following notes and explanations must be complied with when installing and commissioning the components.
The trained specialists must always use the current valid documentation.

The trained specialists must ensure that the application and use of the products described is in line with all safety requirements, including all relevant laws, regulations, guidelines, and standards.

Disclaimer

The documentation has been compiled with care. The products described are, however, constantly under development.
We reserve the right to revise and change the documentation at any time and without notice.
Claims to modify products that have already been supplied may not be made on the basis of the data, diagrams, and descriptions in this documentation.

Trademarks

Beckhoff®, TwinCAT®, TwinCAT/BSD®, TC/BSD®, EtherCAT®, EtherCAT G®, EtherCAT G10®, EtherCAT P®, Safety over EtherCAT®, TwinSAFE®, XFC®, XTS®, and XPlanar® are registered and licensed trademarks of Beckhoff Automation GmbH.
If third parties make use of the designations or trademarks contained in this publication for their own purposes, this could infringe upon the rights of the owners of the said designations.



EtherCAT® is a registered trademark and patented technology, licensed by Beckhoff Automation GmbH, Germany

Copyright

© Beckhoff Automation GmbH & Co. KG, Germany.
The distribution and reproduction of this document, as well as the use and communication of its contents without express authorization, are prohibited.
Offenders will be held liable for the payment of damages. All rights reserved in the event that a patent, utility model, or design are registered.

Third-party trademarks

Trademarks of third parties may be used in this documentation. You can find the trademark notices here: https://www.beckhoff.com/trademarks.

1.2 For your safety

Safety regulations

Read the following explanations for your safety.
Always observe and follow product-specific safety instructions, which you may find at the appropriate places in this document.

Exclusion of liability

All the components are supplied in particular hardware and software configurations which are appropriate for the application. Modifications to hardware or software configurations other than those described in the documentation are not permitted, and nullify the liability of Beckhoff Automation GmbH & Co. KG.

Personnel qualification

This description is only intended for trained specialists in control, automation, and drive technology who are familiar with the applicable national standards.

Signal words

The signal words used in the documentation are classified below. In order to prevent injury and damage to persons and property, read and follow the safety and warning notices.

Personal injury warnings

⚠ DANGER
Hazard with high risk of death or serious injury.

⚠ WARNING
Hazard with medium risk of death or serious injury.

⚠ CAUTION
There is a low-risk hazard that could result in medium or minor injury.

Warning of damage to property or environment

NOTICE
The environment, equipment, or data may be damaged.

Information on handling the product

This information includes, for example:
recommendations for action, assistance or further information on the product.

1.3 Notes on information security

The products of Beckhoff Automation GmbH & Co. KG (Beckhoff), insofar as they can be accessed online, are equipped with security functions that support the secure operation of plants, systems, machines and networks. Despite the security functions, the creation, implementation and constant updating of a holistic security concept for the operation are necessary to protect the respective plant, system, machine and networks against cyber threats. The products sold by Beckhoff are only part of the overall security concept. The customer is responsible for preventing unauthorized access by third parties to its equipment, systems, machines and networks. The latter should be connected to the corporate network or the Internet only if appropriate protective measures have been set up.

In addition, the recommendations from Beckhoff regarding appropriate protective measures should be observed. Further information regarding information security and industrial security can be found in our https://www.beckhoff.com/secguide.

Beckhoff products and solutions undergo continuous further development. This also applies to security functions. In light of this continuous further development, Beckhoff expressly recommends that the products are kept up to date at all times and that updates are installed for the products once they have been made available. Using outdated or unsupported product versions can increase the risk of cyber threats.

To stay informed about information security for Beckhoff products, subscribe to the RSS feed at https://www.beckhoff.com/secinfo.

1.4 Documentation issue status

Version	Change
1.1.x	**New:** Installation [▶ 10]: TwinCAT 2 Setup

2 Overview

OPC Unified **A**rchitecture (OPC UA) is the next generation of the familiar OPC standard. This is a globally standardized communication protocol via which machine data can be exchanged irrespective of the manufacturer and platform. OPC UA already integrates common security standards directly in the protocol. Another major advantage of OPC UA over the conventional OPC standard is its independence from the COM/DCOM system.

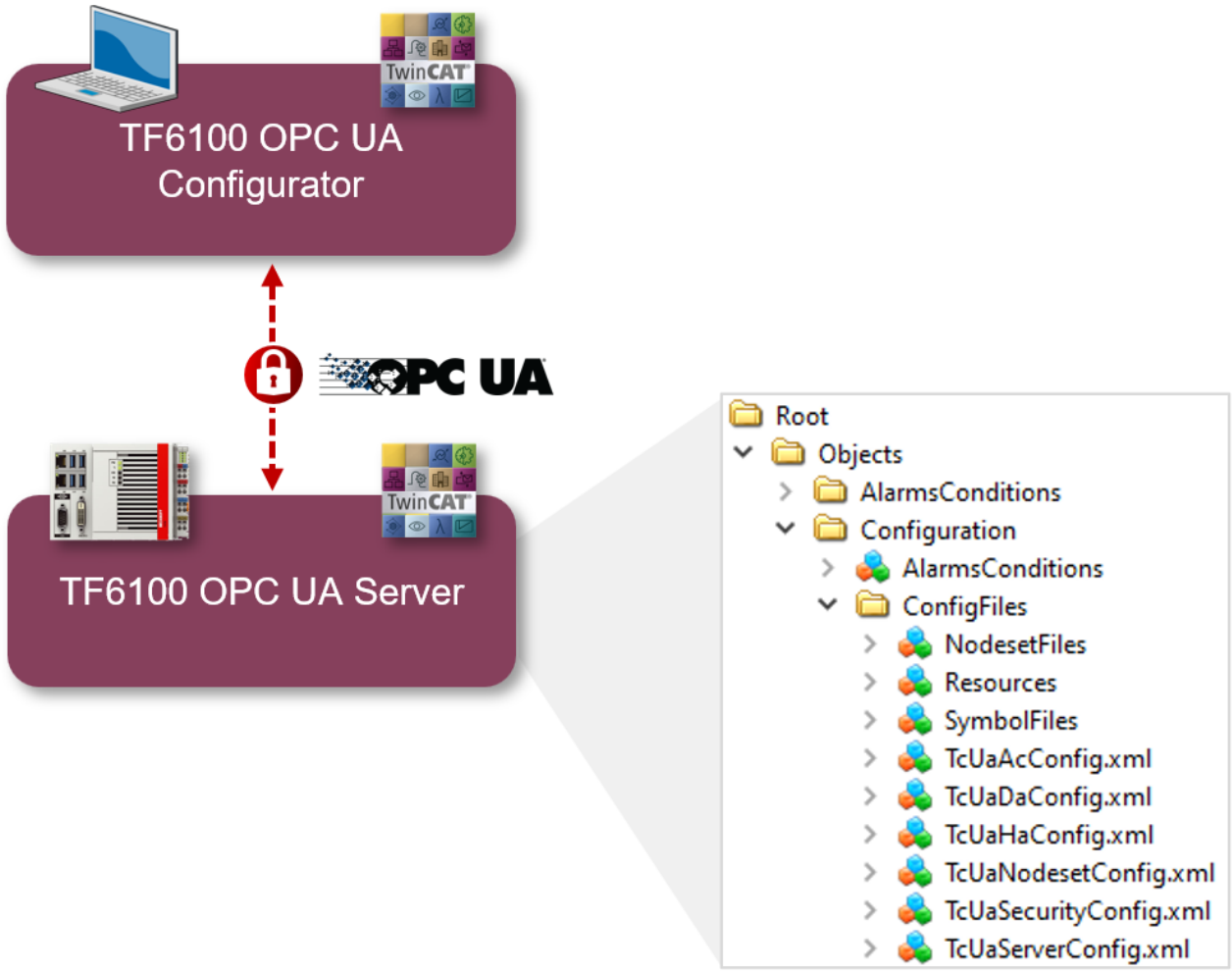


Detailed information on OPC UA can be found on the web pages of the OPC Foundation.

The TwinCAT 3 Function TF6100 OPC UA consists of several software components that enable data exchange with TwinCAT based on OPC UA.
The following table provides an overview of the individual product components.

Software component	Description
TwinCAT OPC UA Server	Provides an OPC UA Server interface so that UA clients can access the TwinCAT runtime.
TwinCAT OPC UA Client	Provides OPC UA Client functionality to enable communication with other OPC UA Servers based on PLCopen-standardized function blocks and an easy-to-configure I/O device.
TwinCAT OPC UA Configurator	Graphical user interface for configuring the TwinCAT OPC UA Server.
TwinCAT OPC UA Sample Client	Graphical sample implementation of an OPC UA Client in order to carry out a first connection test with the TwinCAT OPC UA Server.
TwinCAT OPC UA Gateway	Wrapper technology that provides both an OPC COM DA Server interface and OPC UA Server aggregation capabilities.

This documentation describes the TwinCAT OPC UA Configurator, which is an engineering software component that provides a graphical user interface for configuring the TwinCAT OPC UA Server. The TwinCAT OPC UA Configurator is delivered in two variants: an interface integrated into Visual Studio (or the TwinCAT XAE Shell) and a standalone tool. Both variants have the characteristic that you establish an OPC UA communication connection to a TwinCAT OPC UA Server and configure the server via this connection. The basis for this is the so-called configuration namespace of the TwinCAT OPC UA Server, which provides all relevant configuration files for authenticated users via OPC UA.



For a quick introduction to the product, we recommend our chapters Installation [▶ 10] and Quick Start [▶ 14]. Please also note the system requirements [▶ 10] for this product.

3 Installation

3.1 System requirements

The following system requirements apply for the installation and operation of this product. A distinction must be made between the Standalone Configurator and Visual Studio Configurator.

Visual Studio Configurator

Technical data	Description
Operating system	Windows 10 Windows Server 2022
Target platforms	PC architecture (x86, x64)
.NET Framework	4.7.2
TwinCAT installation level	TwinCAT 3 XAE
Required TwinCAT license	---

Standalone Configurator

Technical data	Description
Operating system	Windows 10 Windows Server 2022
Target platforms	PC architecture (x86, x64)
.NET Framework	4.7.2
TwinCAT installation level	TwinCAT 2 CP, PLC, NC-PTP TwinCAT 3 XAE, XAR, ADS
Required TwinCAT license	---

3.2 Installation

Depending on the TwinCAT version and operating system used, this TwinCAT 3 Function can be installed in different ways, which are described in more detail below.

NOTICE

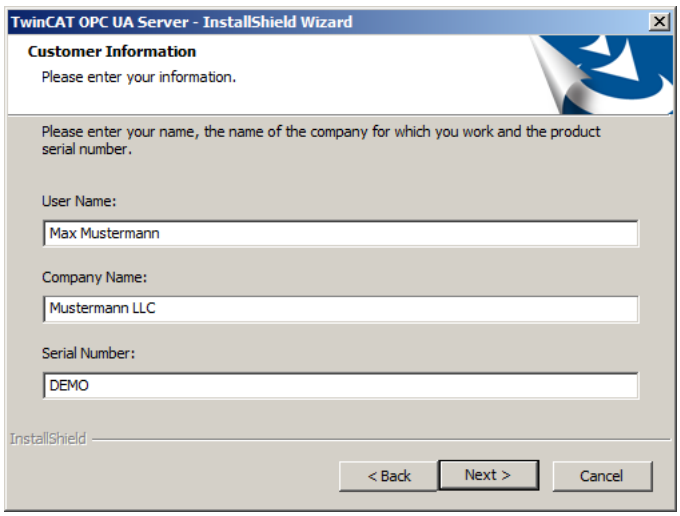
Update installation

An update installation always uninstalls the previous installation. Please make sure that you have backed up your configuration files beforehand.

TwinCAT 2 Setup

If you are using TwinCAT 2, you can install this supplement via a setup package, which you can download from the Beckhoff website at https://www.beckhoff.com/download.

The installation can be done on either the engineering or runtime side, depending on the system you need the supplement for. Please note that under TwinCAT 2, the supplement is licensed directly during installation. A separate dialog prompts you to enter the license key. If you would like to install a 30-day trial version of the supplement, please enter DEMO as the license key.



There is a separate setup download for TwinCAT 2 on Windows CE with the name TS6100-0030 OPC UA. This setup installs the CAB files, which you can then transfer to a Windows CE device and install them there.

TwinCAT 3 Package Manager

If you are using TwinCAT 3.1 Build 4026 (and higher) on the Microsoft Windows operating system, you can install this function via the TwinCAT Package Manager, see Installation documentation.

Normally you install the function via the corresponding workload; however, you can also install the packages contained in the workload individually. This documentation briefly describes the installation process via the workload.

NOTICE

Unprepared TwinCAT restart can cause data loss

The installation of this function may result in a TwinCAT restart.
Make sure that no critical TwinCAT applications are running on the system or shut them down in an orderly manner first.

Command line program TcPkg

You can use the TcPkg **C**ommand **L**ine **I**nterface (CLI) to display the available workloads on the system and to install a specific workload.

```
tcpkg list -t workload
tcpkg install TF6100.OpcUaConfigurator.XAE
```

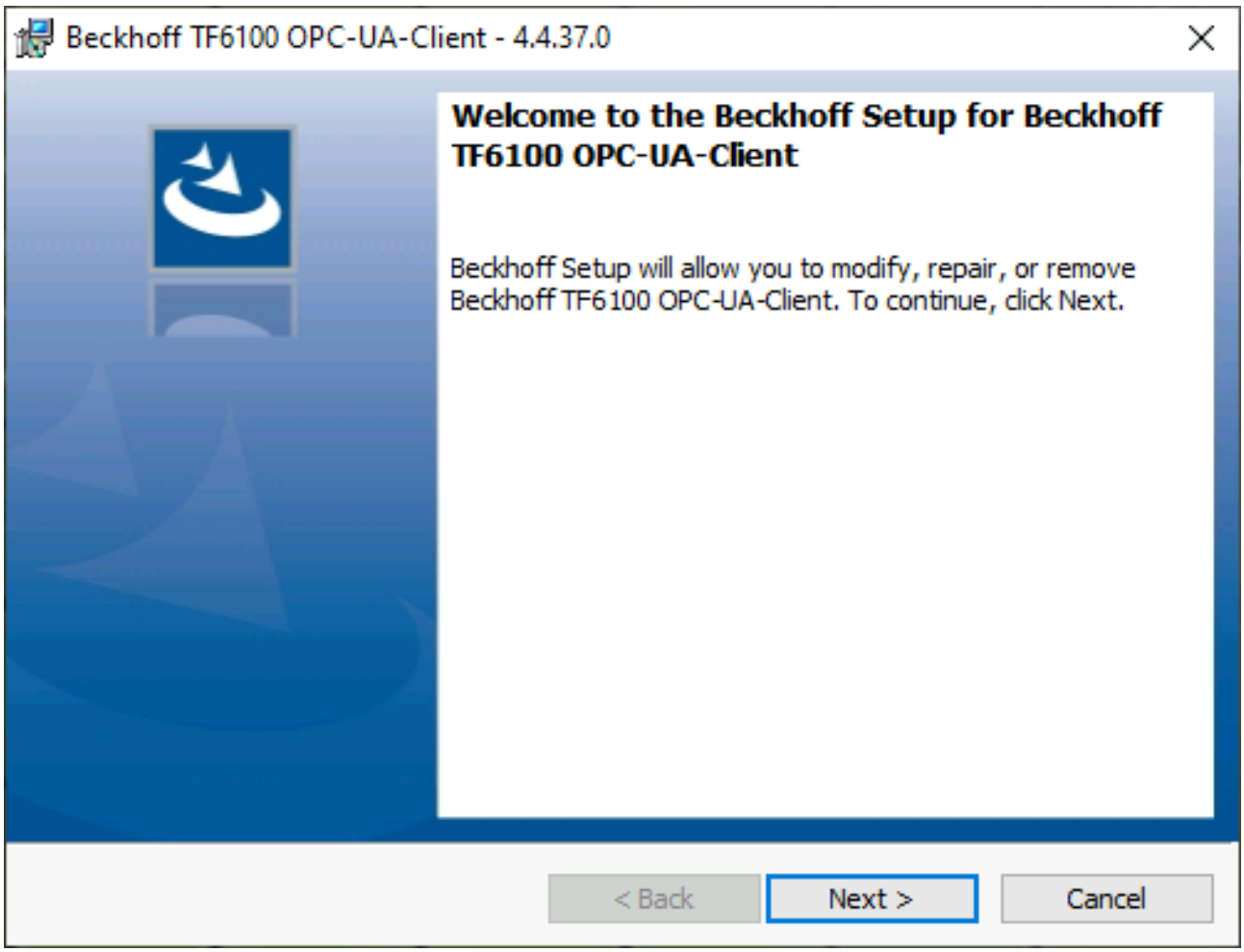
TwinCAT Package Manager UI

You can use the **U**ser **I**nterface (UI) to display all available workloads and install them if required. To do this, follow the corresponding instructions in the interface.

TwinCAT 3 Setup

If you are using TwinCAT 3.1 Build 4024 on the Microsoft Windows operating system, you can install this function via a setup package, which you can download from the Beckhoff website at https://www.beckhoff.com/download.

Depending on the system on which you need the function, the installation can be done on either the engineering or runtime side. The following screenshot shows an example of the setup interface using the TF6100 TwinCAT OPC UA Client setup.



To complete the installation process, follow the instructions in the Setup dialog.

NOTICE

Unprepared TwinCAT restart can cause data loss

The installation of this function may result in a TwinCAT restart.
Make sure that no critical TwinCAT applications are running on the system or shut them down in an orderly manner first.

Integration in Visual Studio and XAE Shell

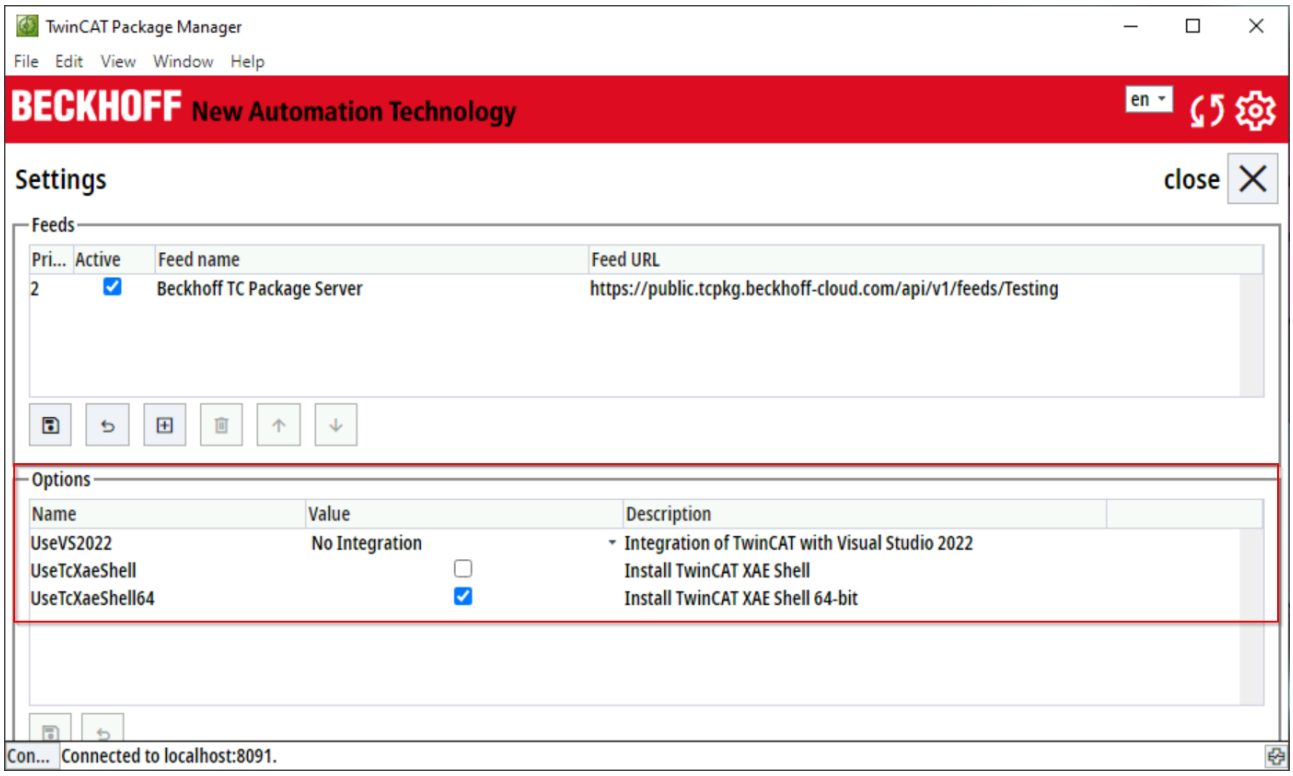
This product contains components that are integrated into Visual Studio and the XAE Shell. When installing via the TwinCAT Package Manager, you can use configuration parameters to define the existing Visual Studio versions into which the product is to be integrated.

You can use the following TcPkg CLI call to view the current Visual Studio integration settings and adjust them if necessary:

```
tcpkg config list
tcpkg config set -n UseVS2022
```



You will also find these settings in the TcPkg UI and can adjust them according to your engineering environment. The following screenshot shows a configuration in which only integration into the TwinCAT XAE Shell (64-bit) takes place.



Subsequent integration in Visual Studio

If you want to perform an integration in Visual Studio after you have already installed the product once, you can perform the necessary steps using either TcPkg CLI or UI.

TcPkg UI

By setting the Visual Studio integration configuration option above, all installed workloads with a VS integration are automatically searched for and integrated accordingly. No further manual adjustments are required.

TcPkg CLI

In the TcPkg CLI, after enabling the desired VS integration, you must enter the following commands to automatically perform the integration for the corresponding installed workloads.

```
tcpkg install vs2022.ext
```

4 Technical introduction

4.1 Quick start

The following chapter provides a quick introduction to the TwinCAT OPC UA Configurator. In these instructions, the Standalone Configurator is used to establish a connection with the locally installed TwinCAT OPC UA Server and to configure it. This requires both products to be installed - on the same system in this example.

The following steps are described in more detail below:

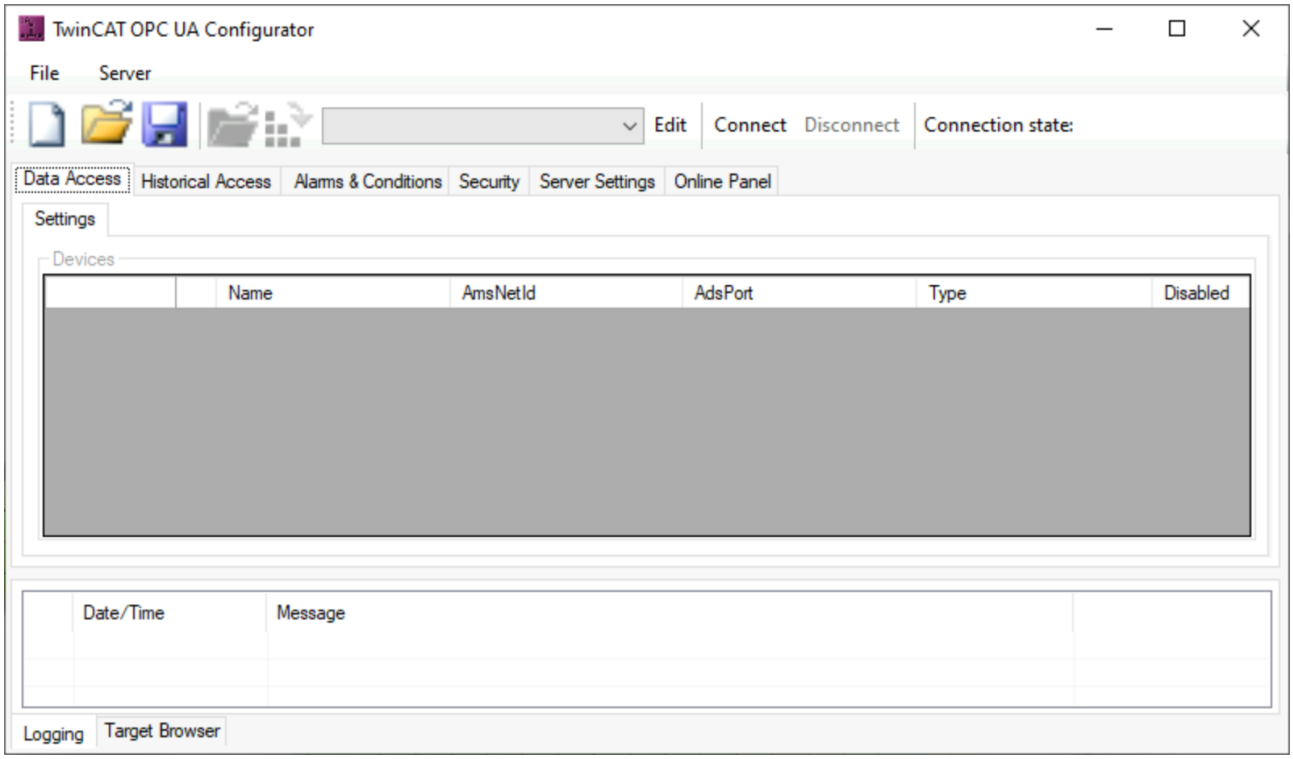
- Starting the Standalone Configurator
- Configuring a server connection
- Connecting to the server and reading the configuration
- Making changes to the configuration
- Activating the new configuration on the server

Starting the Standalone Configurator

The TwinCAT OPC UA Configurator is installed by default in a subdirectory of the TwinCAT installation directory. Further information can be found in the documentation chapter on the application directories [▶ 20].

When the product is installed, a shortcut is created in the Windows Start menu, which enables easy access to the application.

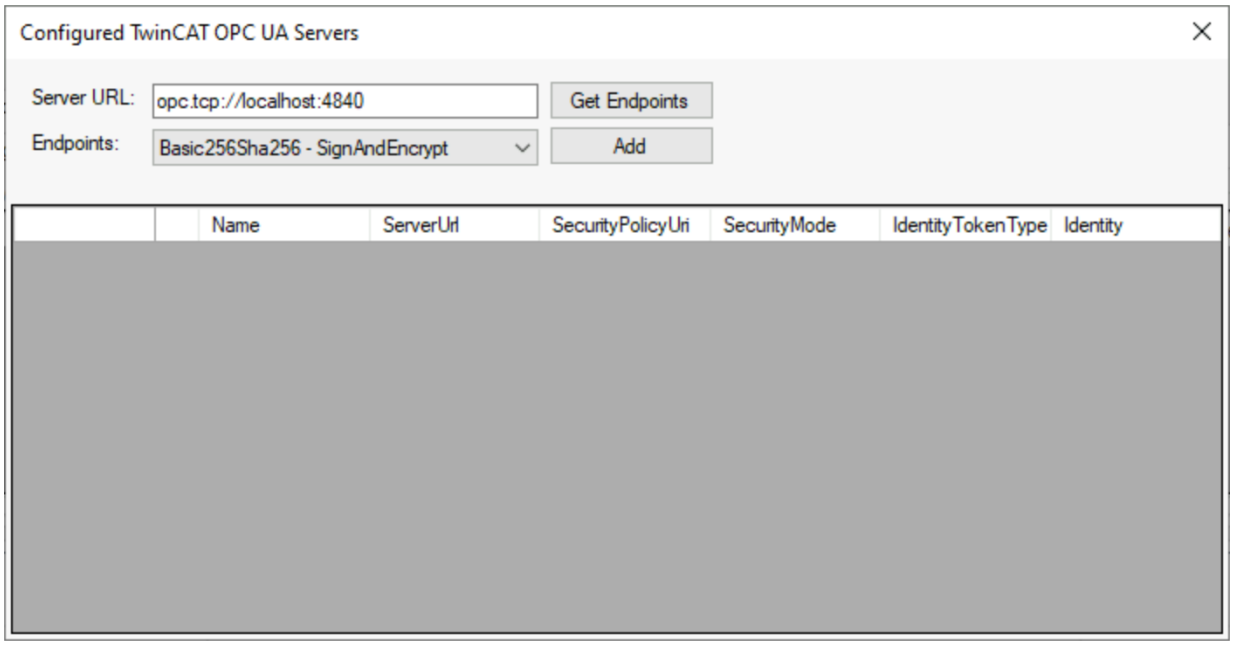
After starting the application, you must first configure a new server connection. In the next step, you will learn how to do this.



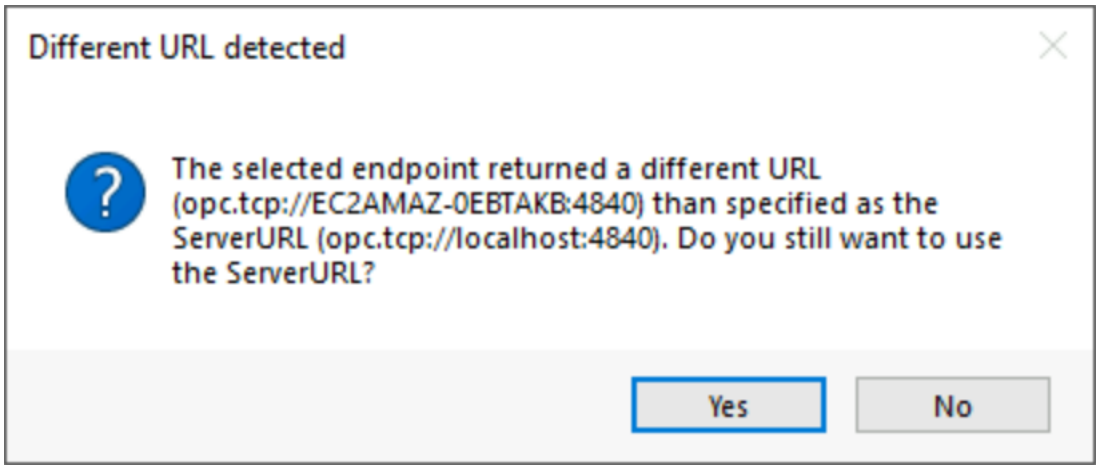
Configuring a server connection

1. Open the server selection dialog by clicking on the **Edit** button in the toolbar. Enter the server URL of the TwinCAT OPC UA Server to be configured in the dialog that opens. In this example, the server is installed on the same system and we can use the default address (opc.tcp://localhost:4840). Click on the

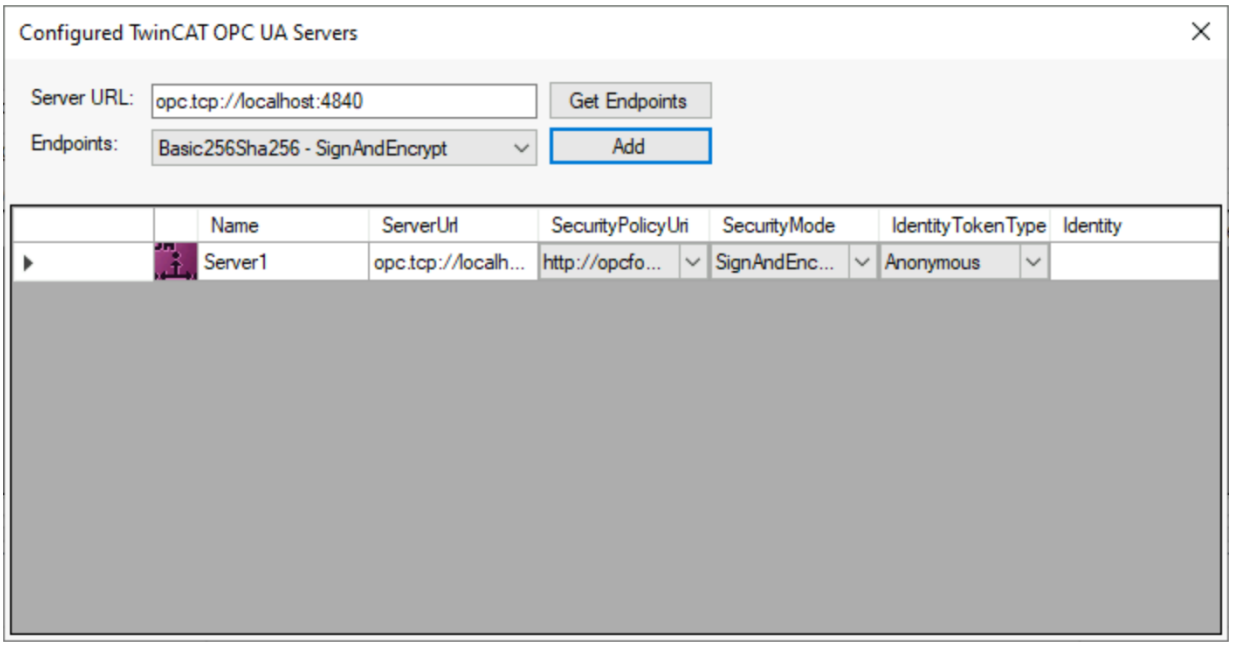
Get Endpoints button to obtain a list of all server endpoints. Select the endpoint "Basic256Sha256 - SignAndEncrypt".



2. Then click **Add** to add the server. If you receive a warning regarding the differences in the server URL, confirm this message with **Yes**.

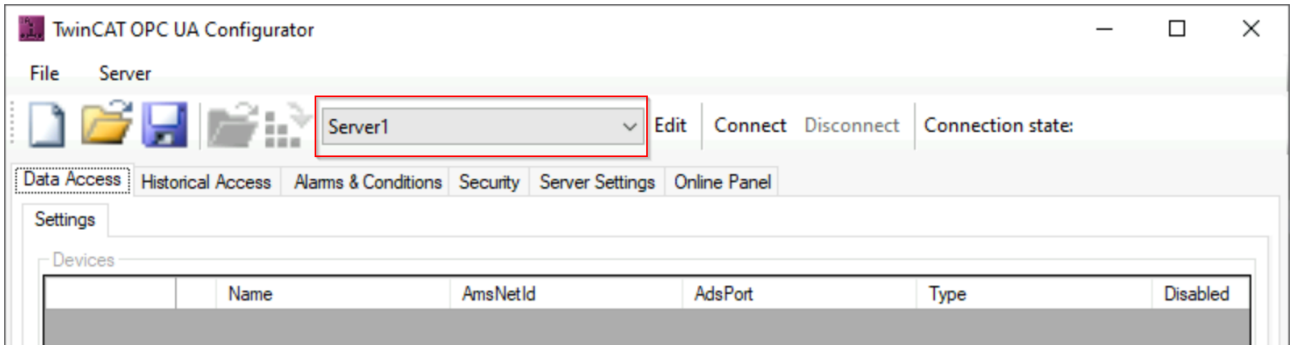


⇨ The TwinCAT OPC UA Server has now been added to the selection dialog.



Depending on the operating environment, further settings for the connection parameters may now be necessary, e.g. user name/password for access to the server. In this example, however, it is assumed that both the TwinCAT OPC UA Server and the TwinCAT OPC UA Configurator were used on this system for the first time. So we leave the default settings and close the dialog.

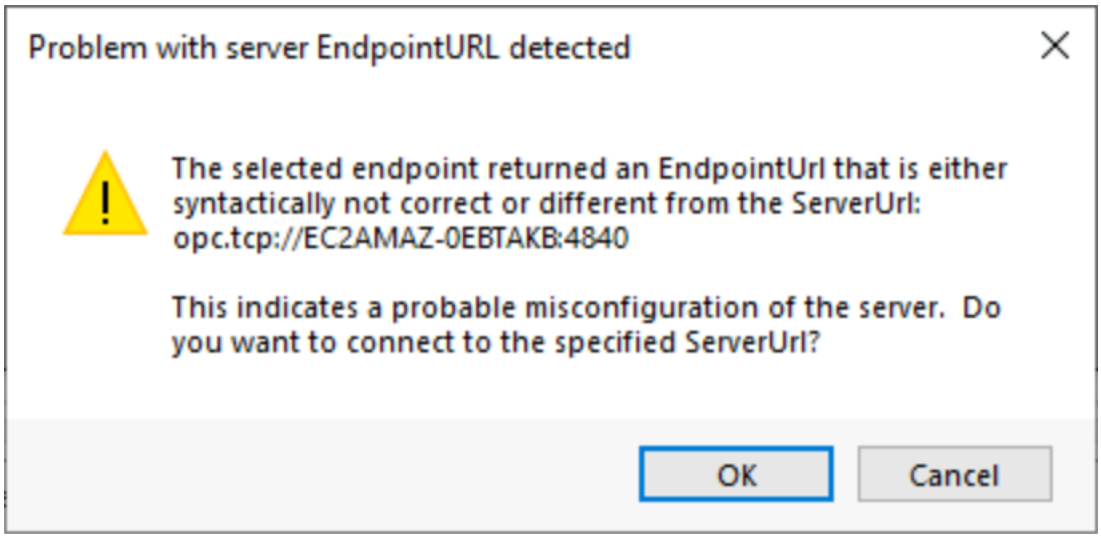
In the server selection list in the toolbar, you will now find a new entry with the connection profile you have just configured.



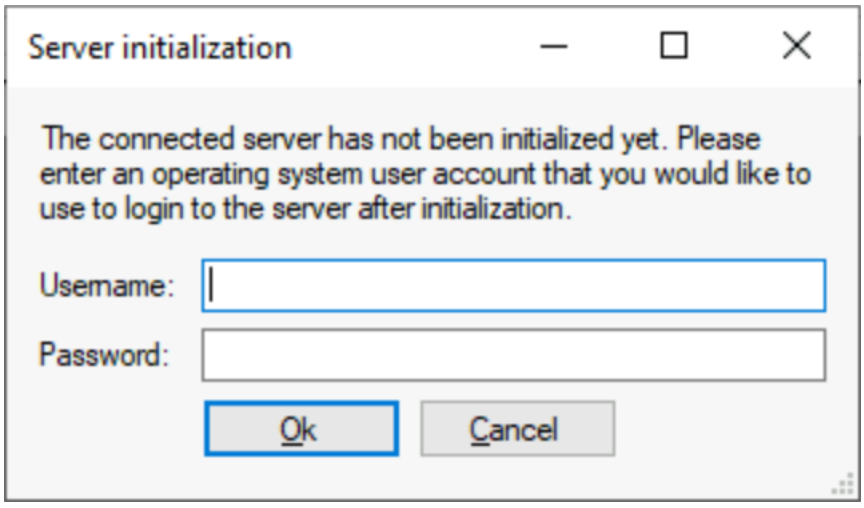
Connecting to the server and reading the configuration

✓ To establish a connection with the server, make sure the server you just configured is selected.

1. Click on **Connect** in the toolbar. If you receive a warning regarding a different server URL, please confirm this dialog with **Ok**.



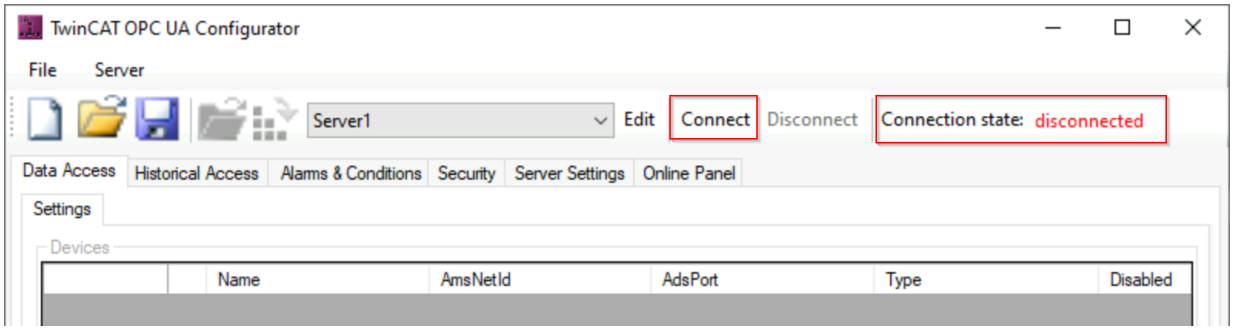
2. Since we assume a new installation in this tutorial, i.e. both the TwinCAT OPC UA Server and the TwinCAT OPC UA Configurator are used for the first time, the Configurator now recognizes that the server is an uninitialized server in the delivery state. For more information on the initialization concept (also known as TOFU Trust On First Use), we recommend the corresponding documentation chapter in the TwinCAT OPC UA Server documentation.
3. Enter a username/password combination to initialize the server. You can either use an existing user in the operating system or a new user, which is then automatically created in the operating system.

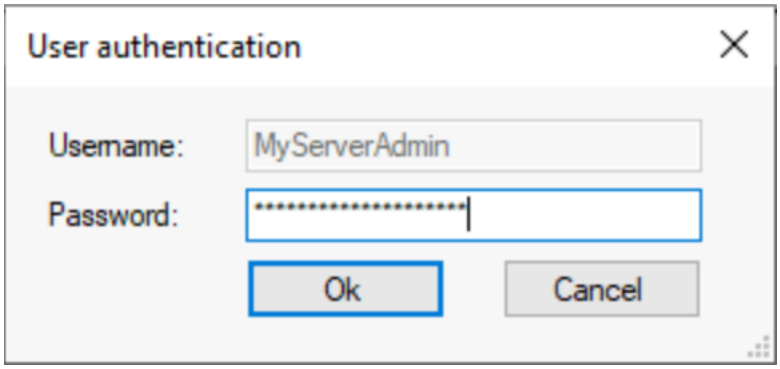


Please make a note of the username/password combination used, as this is required for later access to the server via OPC UA.

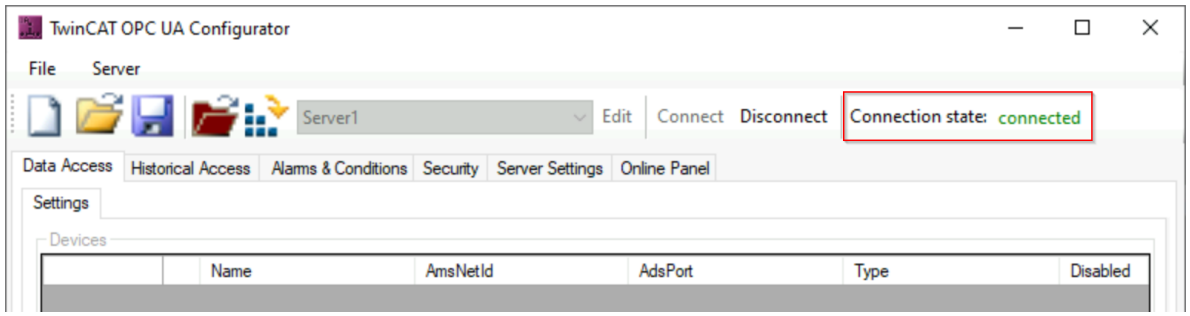
⇨ The server is now initialized and restarted.

4. You can now re-establish a connection to the server via the toolbar. The user used for initialization was automatically stored in the connection profile. Only the password needs to be re-entered for a connection.

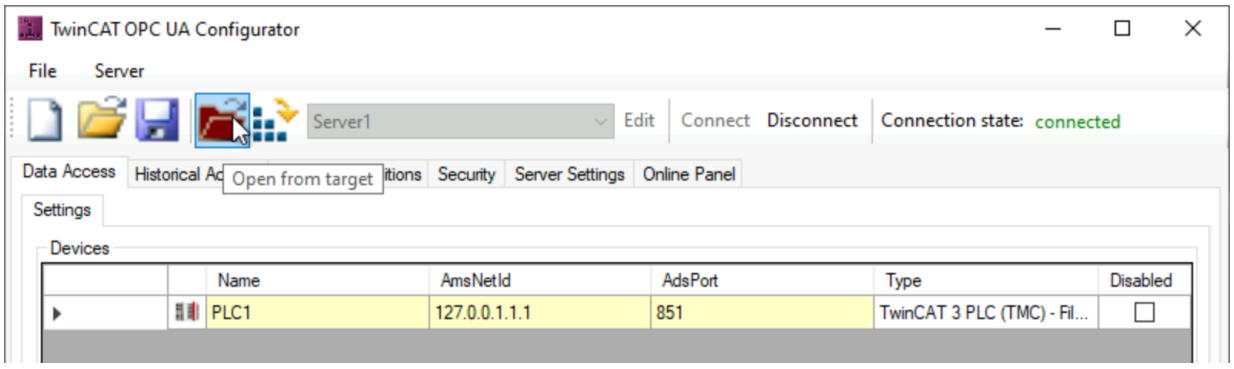




⇨ The **Connection State** will change to "Connected" (green) and you will be connected to the server.



5. You can now read the configuration of the server by clicking the **Open from Target** button in the toolbar.

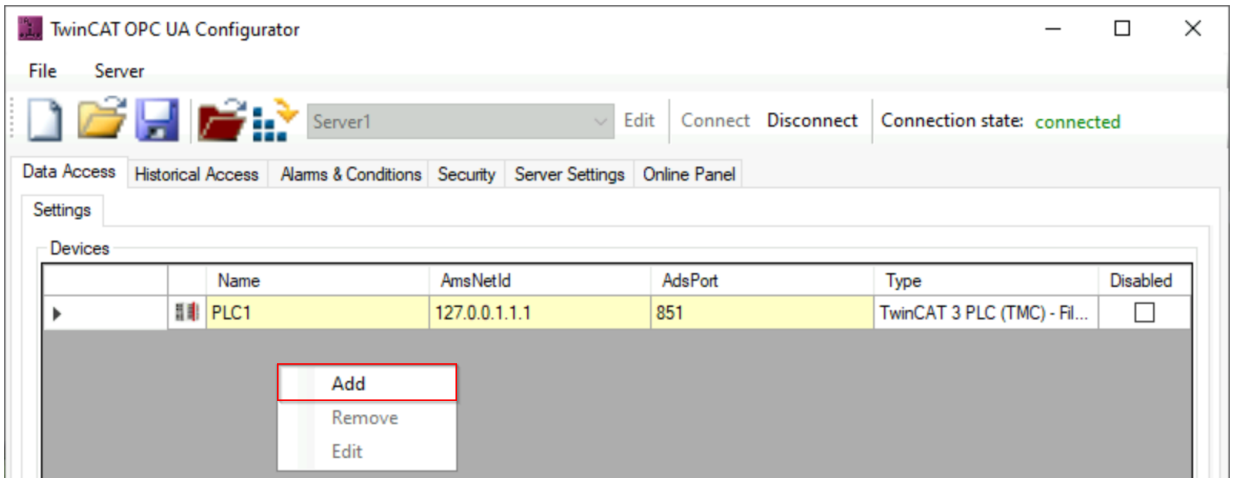


⇨ The configuration of the server is read and displayed in the user interface of the Configurator.

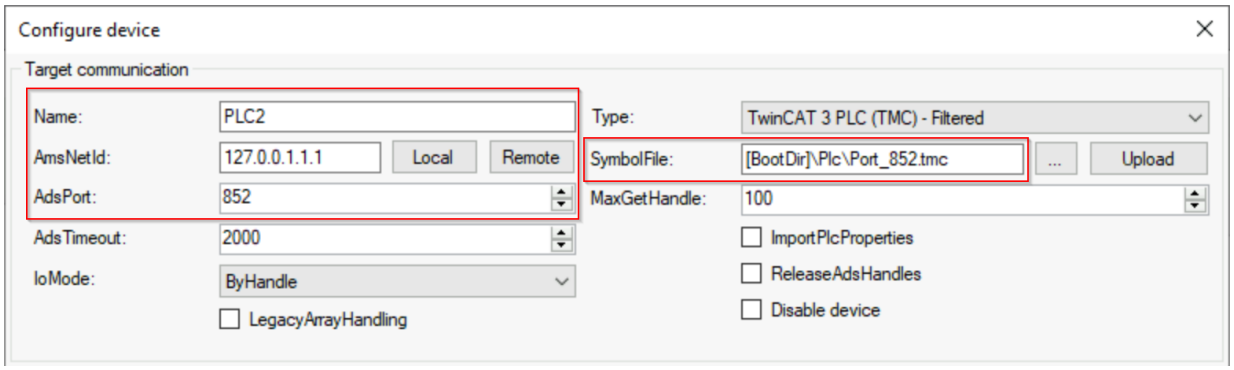
Making changes to the configuration

You can now make any changes to the configuration. In this example, we want to make an additional ADS device available via the TwinCAT OPC UA Server. By default, only the first, locally running PLC runtime from the server perspective is made available via OPC UA. This PLC runtime named "PLC1" can be viewed in the **Data Access** tab.

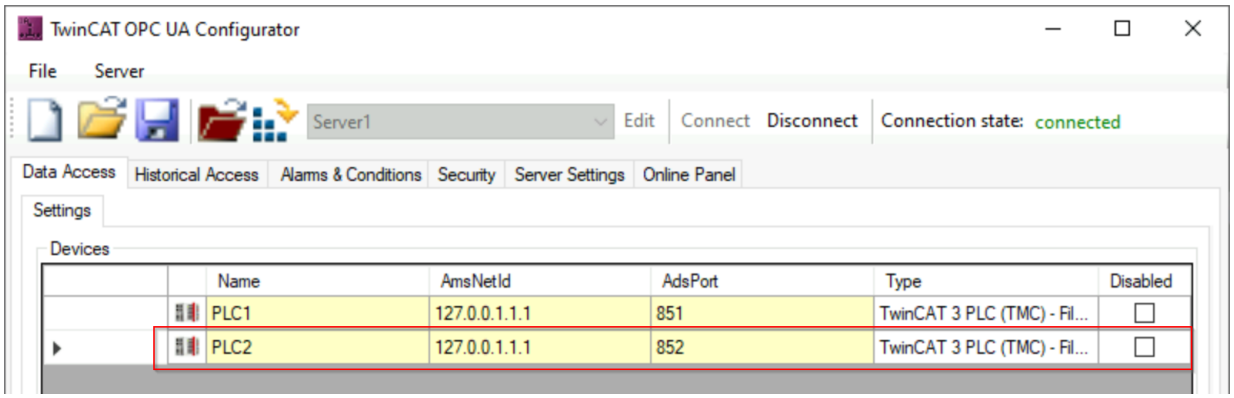
1. We now add another Data Access device via the context menu.



2. In the device properties, we set the parameters for **Name**, **AmsNetId**, **AdsPort** and **SymbolFile** to the settings shown below and save these settings using the **OK** button.

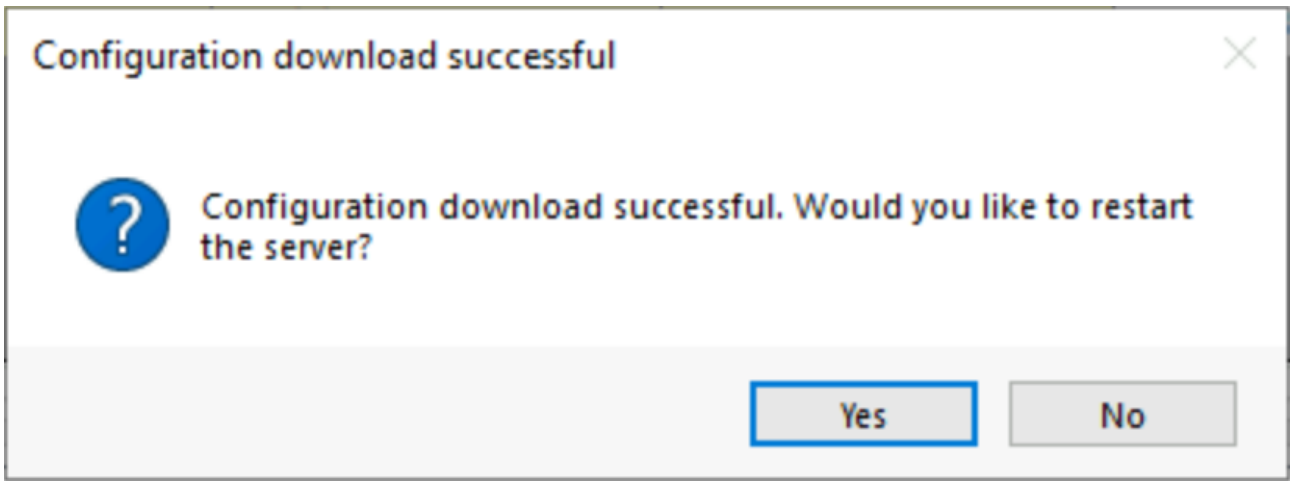


⇨ We have now added a second Data Access device to our configuration.

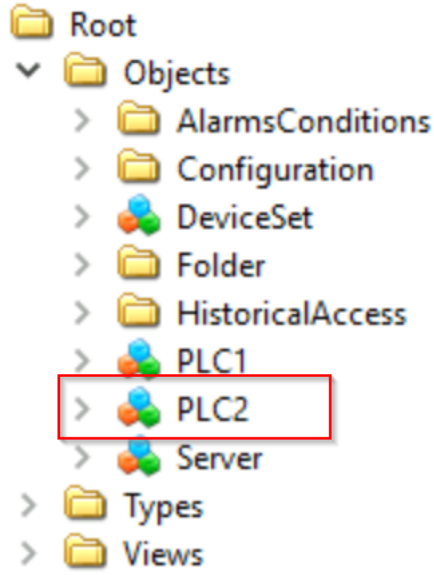


Activating the new configuration on the server

Finally, we need to download the configuration to the server. You can use the corresponding **Activate on Target** button in the toolbar to do this. A final dialog informs us that the configuration has been successfully transferred to the server and asks whether it should be restarted. Confirm this with **Yes**.



You have successfully used the TwinCAT OPC UA Configurator to make a configuration change to the TwinCAT OPC UA Server. In this example, we have added an additional Data Access device to the server. The additional device is the second PLC runtime on the local system. Any OPC UA Client connecting to the server will now find this second PLC runtime under the "PLC2" object in the server address space.



4.2 Application directories

This application uses various directories to store relevant information, such as configuration or certificate files.

Installation directory

The base installation directory of the application is relative to the TwinCAT installation directory.

```
%TcInstallDir%\Functions\TF6100-OPC-UA
```

The application is then installed in the following directory below this directory:

```
%TcInstallDir%\Functions\TF6100-OPC-UA\Win32\Configurator
```

The Visual Studio Configurator files are stored in the following directory:

```
%TcInstallDir%\Functions\TF6100-OPC-UA\Win32\Configurator\Vsix
```

Certificate directory

Certificate files, which are used to establish a secure communication connection, are stored in the following directory. There is a difference between the Standalone Configurator and the Visual Studio Configurator.

```
%ProgramData%\Beckhoff\TF6100-OPC-UA\TcOpcUaConfigurator\PKI
%ProgramData%\Beckhoff\TF6100-OPC-UA\TcOpcUaConfiguratorVs\PKI
```

Configuration files

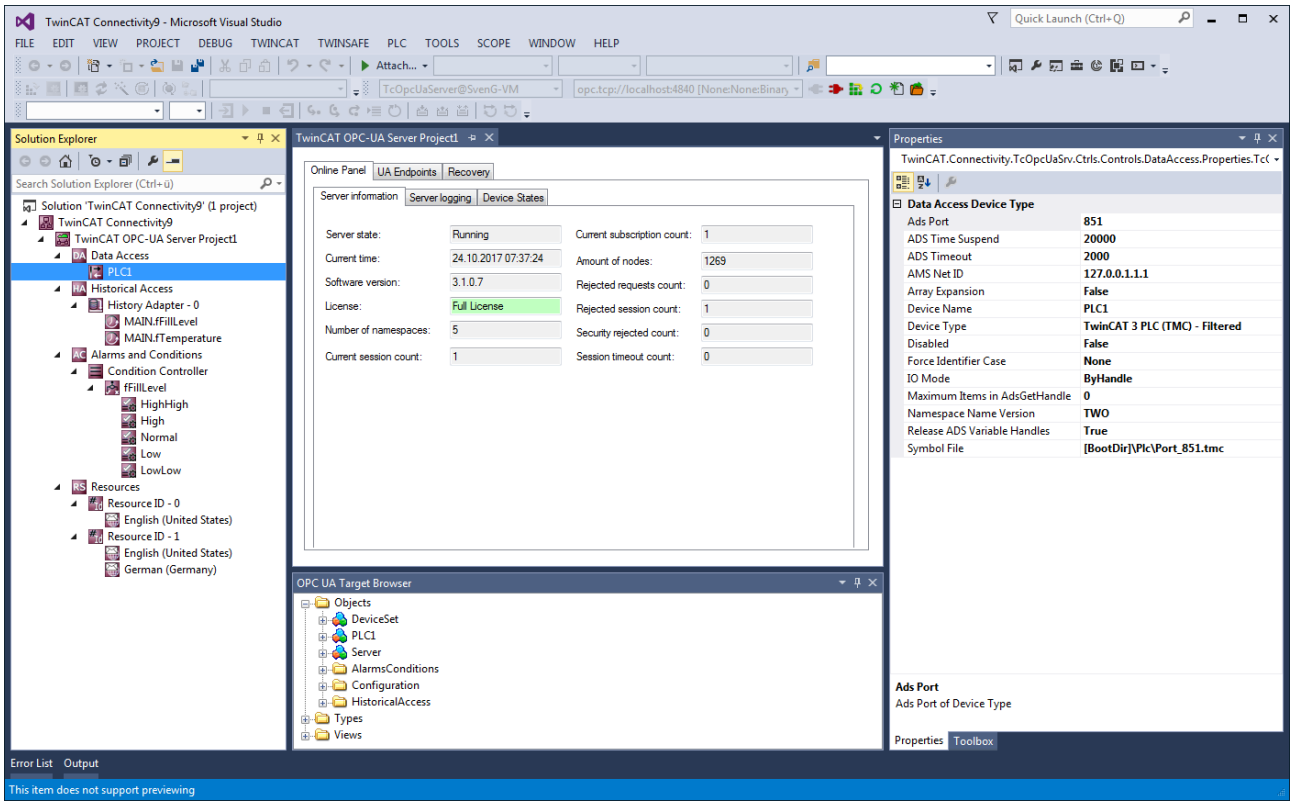
Both the Standalone Configurator and the Visual Studio Configurator use configuration files, e.g. for the server selection dialog. Depending on the tool, these configuration files are stored in the following directory.

```
%ProgramData%\Beckhoff\TF6100-OPC-UA\TcOpcUaConfigurator
%ProgramData%\Beckhoff\TF6100-OPC-UA\TcOpcUaConfiguratorVs
```

4.3 Visual Studio

4.3.1 Overview

The TF6100 setup (version 4.x.x and higher) contains the latest version of the OPC UA Server Configurator. This was integrated in Microsoft Visual Studio as a separate project type to provide an integrated and consistent engineering concept. You can configure all the different facets of the TwinCAT OPC UA Server and in doing so also use source control mechanisms such as Team Foundation Server or Subversion Integrations.



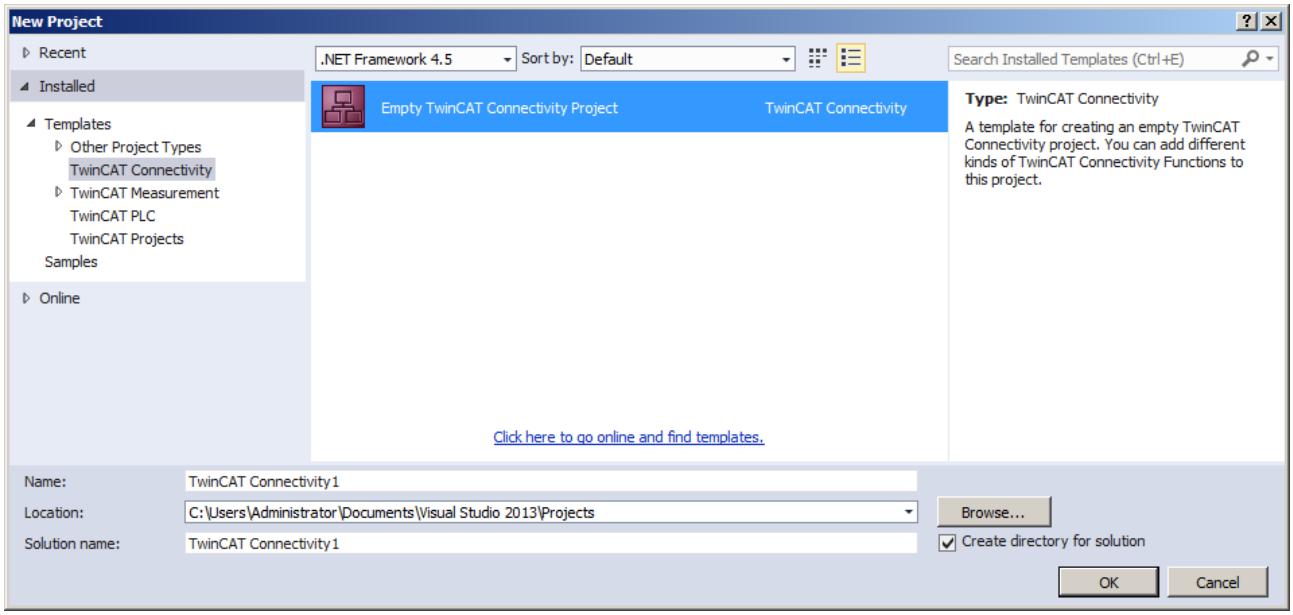
Requirements

Products	Setup versions	Target platform
TF6100	4.x.x	IPC or CX (x86, x64, Arm®)

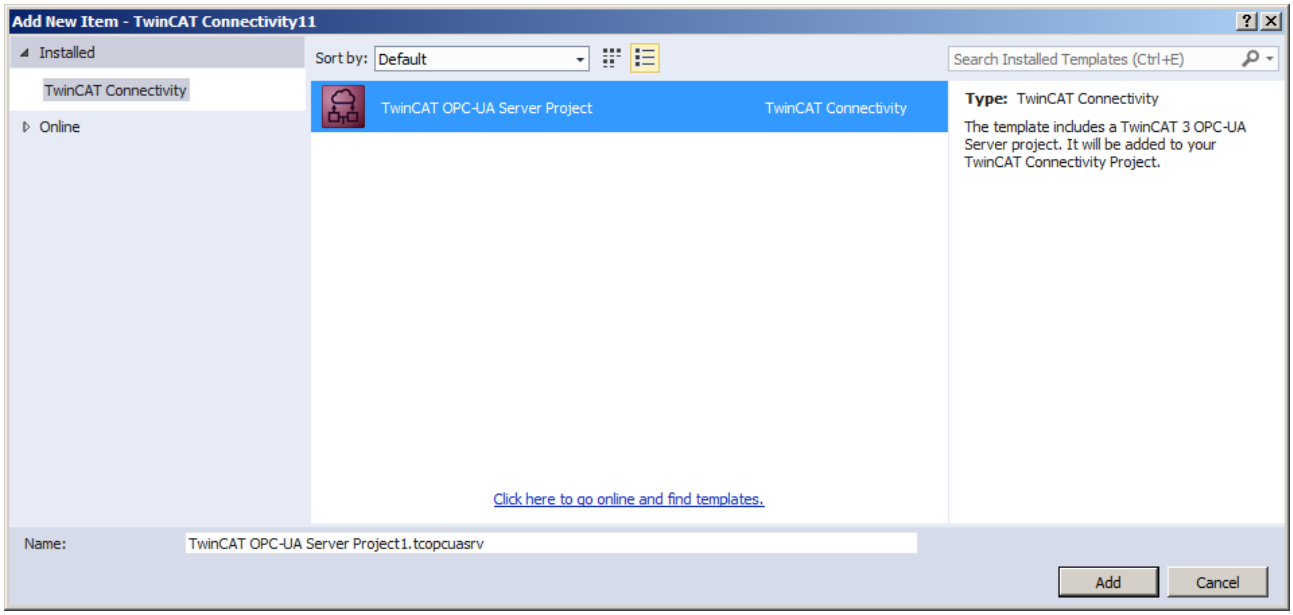
4.3.2 Creating a new project

The project package of the OPC UA Configurator integrates itself in the so-called connectivity package. You can select this when creating a new Visual Studio project.

Project template "TwinCAT Connectivity Project":



Project template "TwinCAT OPC-UA Server Project":



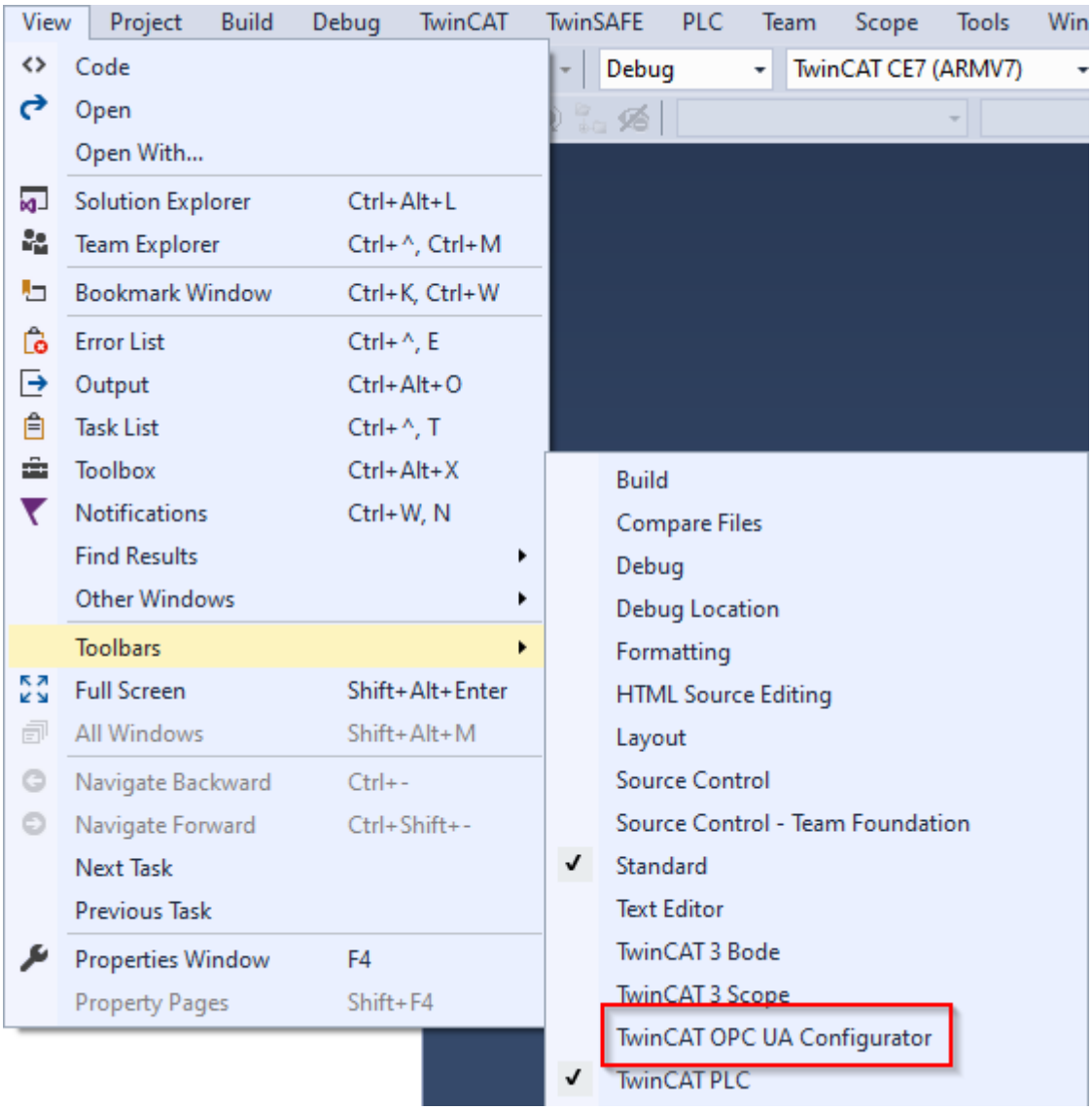
Requirements

Products	Setup versions	Target platform
TF6100	4.x.x	IPC or CX (x86, x64, Arm®)

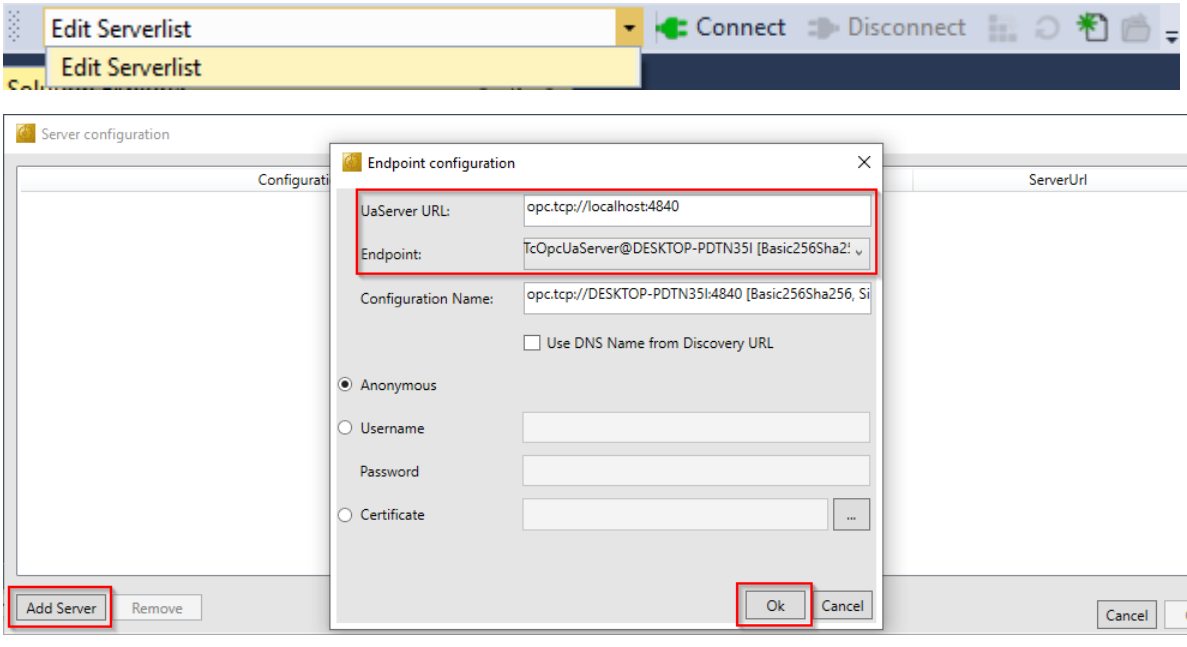
4.3.3 Connecting to a server

The OPC UA Configurator enables the complete parameterization of the Server via OPC UA. Similar to the TwinCAT XAE system, you can select an OPC UA Server to connect to via the toolbar.

1. To do this, first add the appropriate toolbar to your Visual Studio interface.

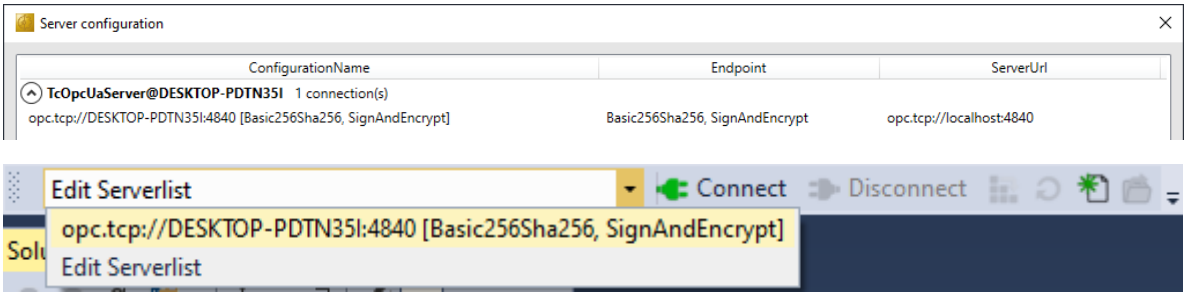


2. You can then add one or more server connections via the entry **Edit Serverlist** in the DropDownBox of the toolbar.



3. In the dialog **Endpoint configuration** you make all settings for the connection with the server, especially the server URL, the selection of an endpoint offered by the server and optionally also the IdentityToken (e.g. username/password) with which the configurator should connect to the server.

   ⇨ The server connection is then added to the server list under an automatically generated configuration name and can then be selected in the drop-down list of the toolbar.



⇨ By clicking on the **Connect** button, a connection to the server can now be established and the server configured.

Online configuration

All settings that you make in your project are carried out for the connected TwinCAT OPC UA Server.

Initialization of the server

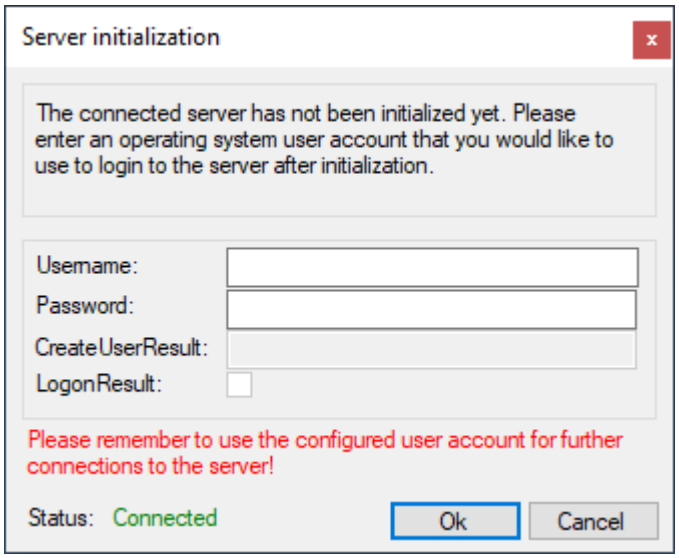
If the server is still in the (uninitialized) delivery state, you will receive a corresponding note for server initialization. This process is described in more detail in the chapter on Performing the server initialization [▶ 24].

Requirements

Products	Setup versions	Target platform
TF6100	4.x.x	IPC or CX (x86, x64, Arm®)

4.3.4 Performing the server initialization

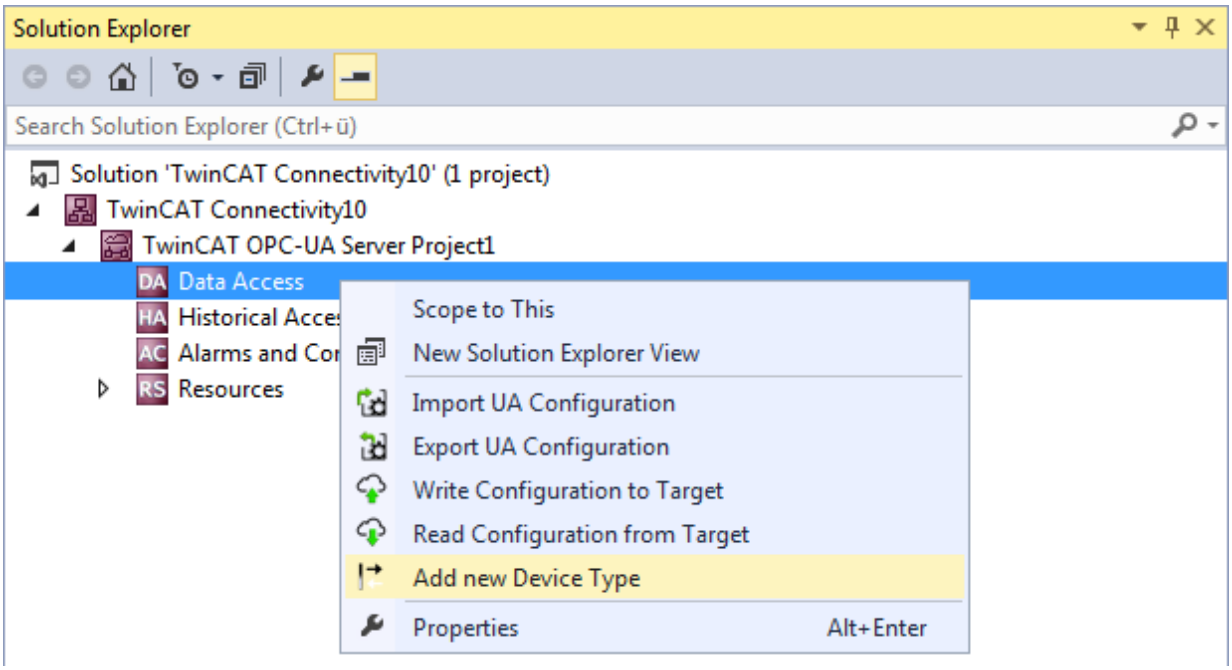
The TwinCAT OPC UA Server is delivered in an uninitialized mode, which is based on the so-called TOFU (Trust-On-First-Use) principle. Detailed information about this server feature and the corresponding background information can be found here. The TwinCAT OPC UA Configurator enables the initialization of the server during the first connection establishment. A corresponding warning message indicates the uninitialized server and enables an appropriate initialization.



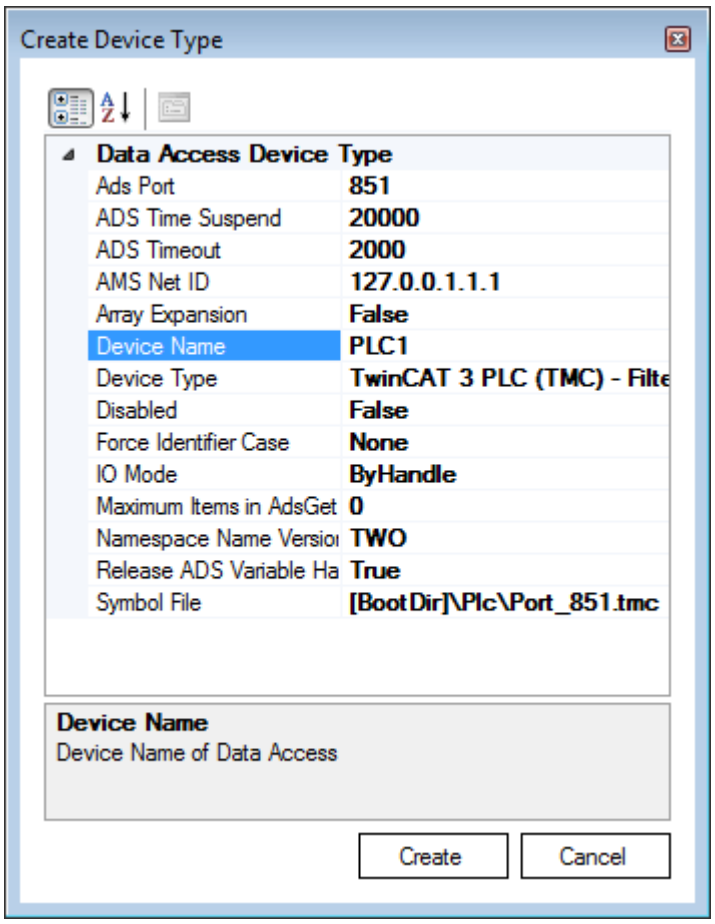
4.3.5 Adding ADS devices

The OPC UA Server can "talk" to one of more ADS devices. To establish a connection, a route to the respective ADS device is required. In the OPC UA Configurator, ADS devices are created, configured and thus announced to the OPC UA Server in the **Data Access** facet.

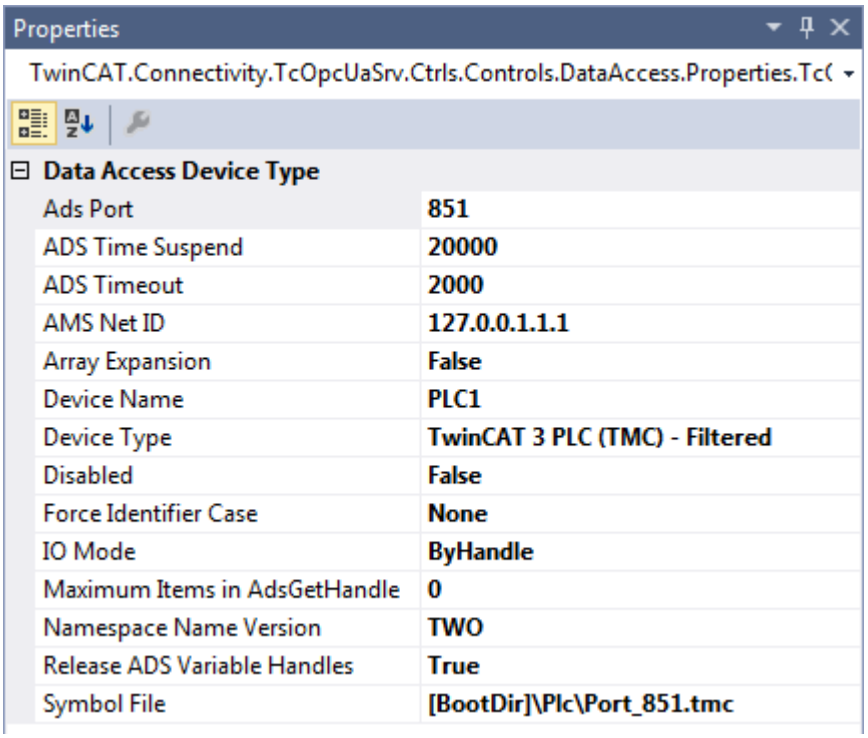
1. New ADS devices are added to the configuration via the context menu command **Add new Device Type**.



2. When the command is executed, a dialog box opens in which connection parameters can be configured for this device, e.g. AMS Net ID, ADS port or the symbol file.

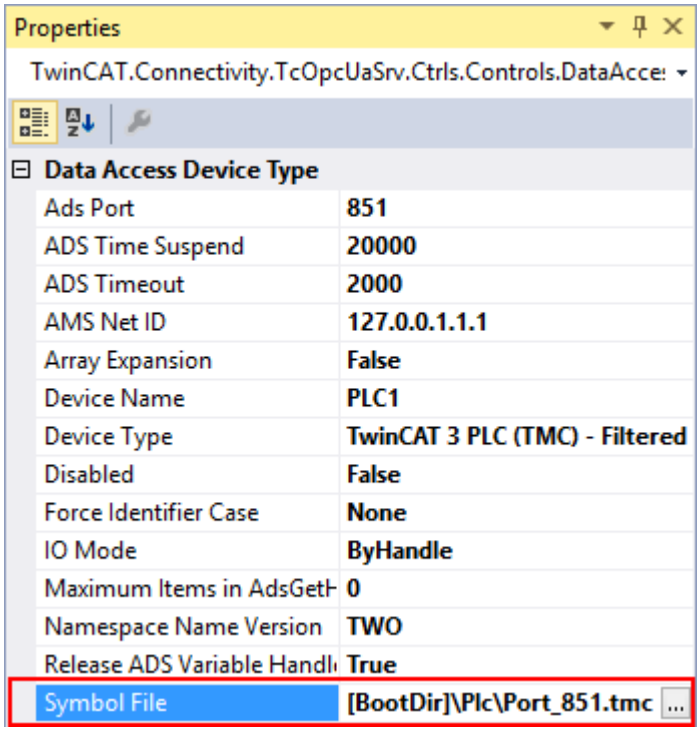


3. You can subsequently modify the connection parameters if necessary via the Properties window in Visual Studio.

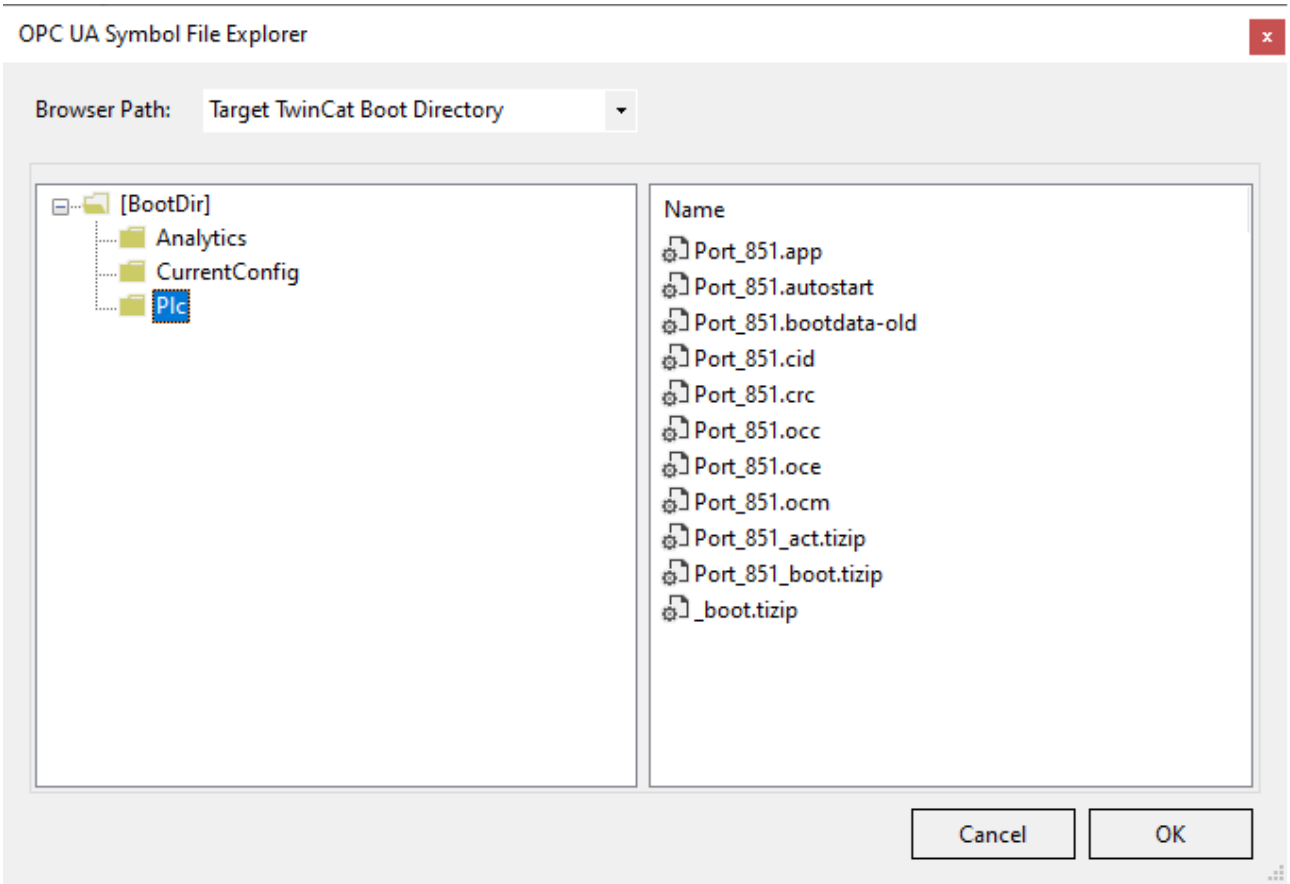


Selecting the symbol file

Symbol files that are present on the selected target device can be imported directly. These symbol files can be stored either in the TwinCAT boot directory or in the symbol directory of the OPC UA Server. You can select the files via the corresponding dialog during the symbol file configuration.



The TwinCAT OPC UA File Explorer can be connected to either the local TwinCAT directory or the remote boot directory. The latter can be read in via the configuration namespace of the server (see Configuration namespace).



Requirements

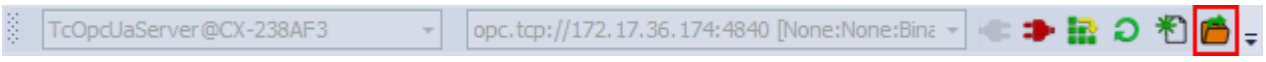
Products	Setup versions	Target platform
TF6100	4.x.x	IPC or CX (x86, x64, Arm®)

4.3.6 Reading and writing the configuration

Via the configurator you can initiate the download/upload of complete server configurations as well as loading every single facet (data access, historical access, etc.) individually to the target device and opening it there. The functions necessary for this are integrated both in the toolbar and in the context menu of the respective facet.

Opening a configuration from the target device

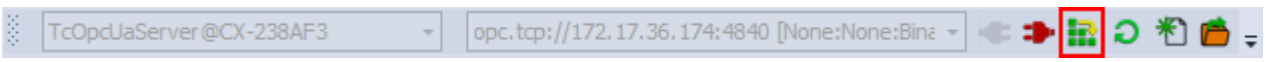
You can open the configuration of the selected target device via the corresponding button in the toolbar.



See also: Connecting to a server [▶ 22]

Activating the configuration on a target device

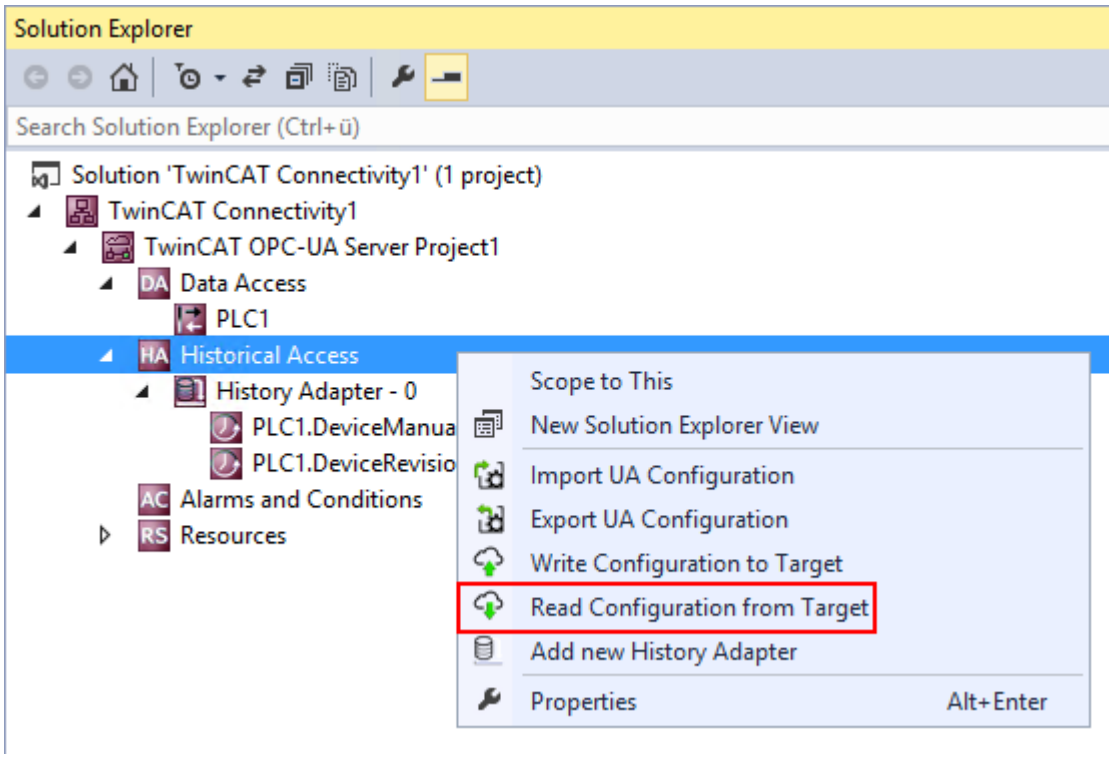
Use the corresponding button in the toolbar to download the currently open configuration to the selected target device.



See also: Connecting to a server [▶ 22]

Opening a partial configuration

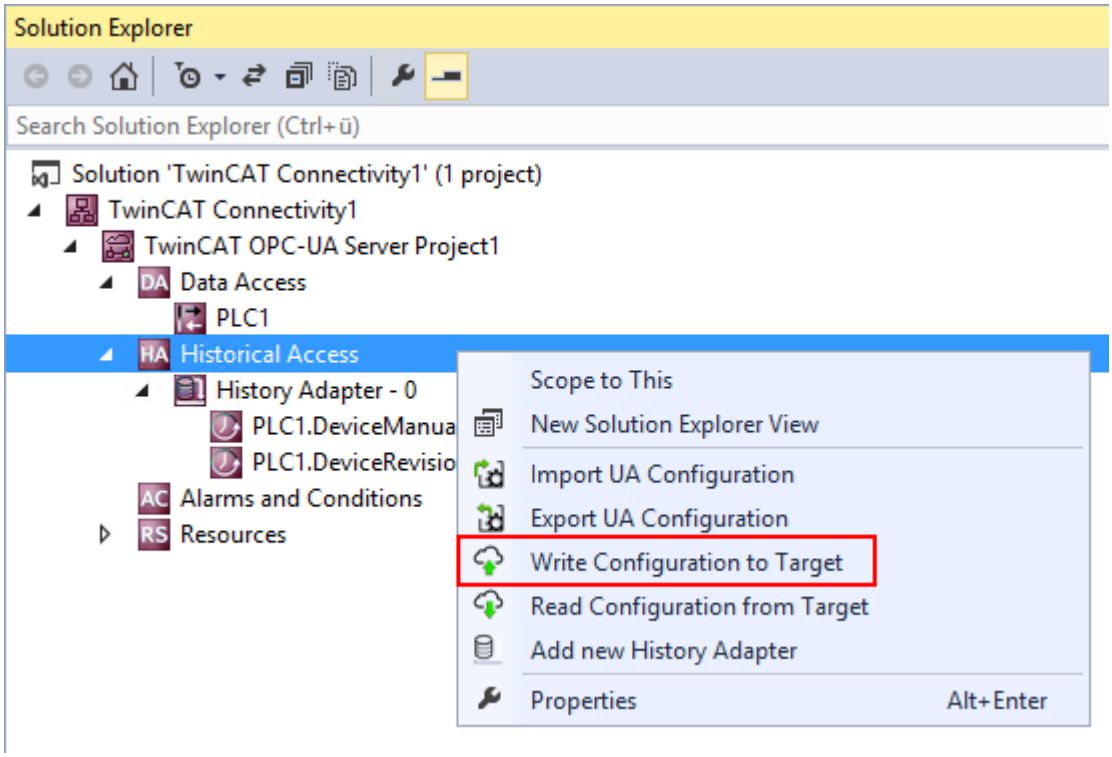
Use the command **Read Configuration from Target** in the context menu of a specific facet of the configuration to open the partial configuration of the selected target device.



See also: Connecting to a server [▶ 22]

Downloading a partial configuration

Use the command **Write Configuration to Target** in the context menu of a specific facet of the configuration to download the partial configuration to the selected target device.



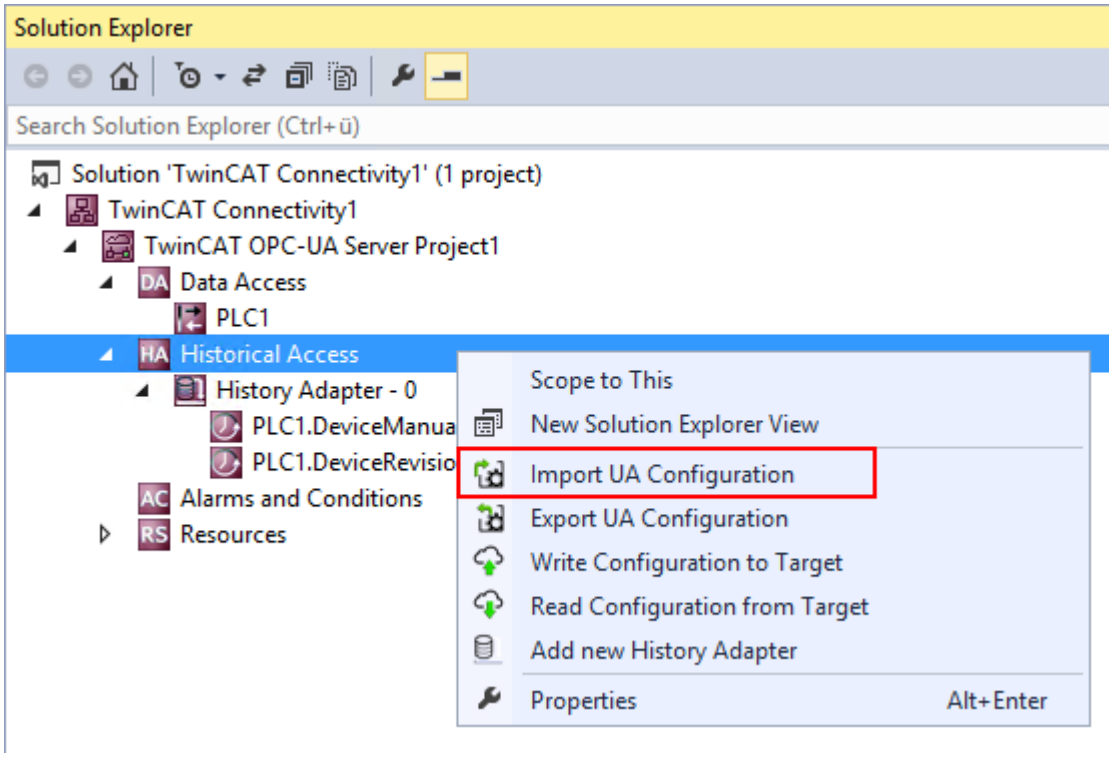
See also: Connecting to a server [▶ 22]

4.3.7 Importing and exporting configuration files

The context menu commands enable the import/export of configuration files of the OPC UA Server.

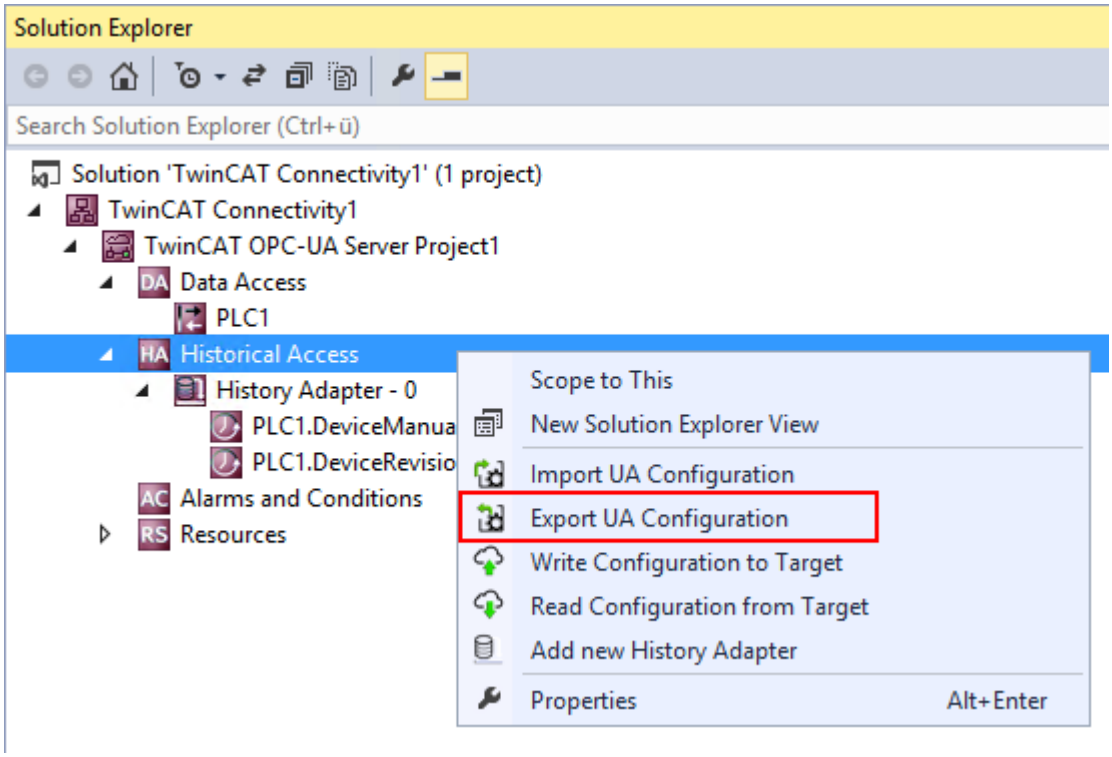
Importing a partial configuration

Use the **Import UA Configuration** command in the context menu of a specific facet of the configuration to import the partial configuration (e.g. Historical Access) from an XML configuration file.



Exporting a partial configuration

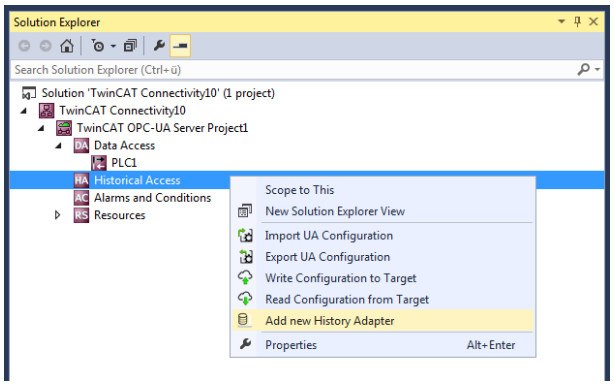
Use the command **Export UA Configuration** in the context menu of a specific facet of the configuration to export the partial configuration (e.g. Historical Access) to an XML configuration file.



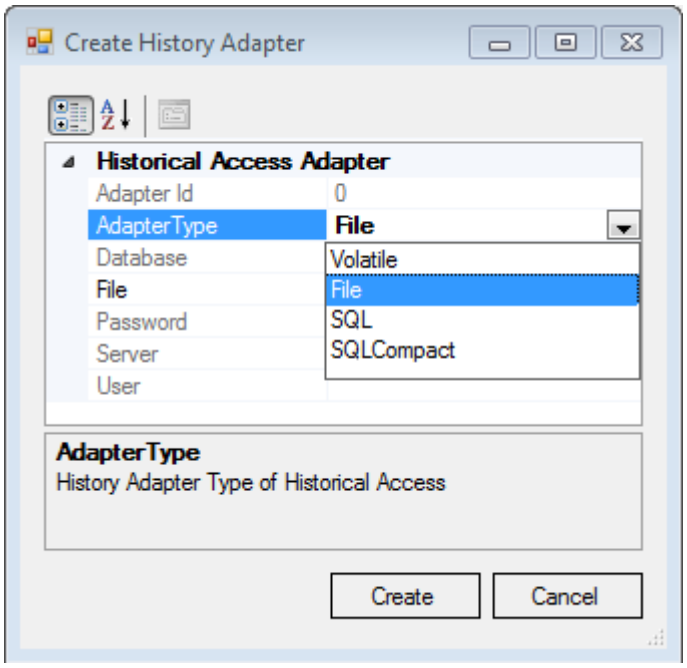
4.3.8 Configuring historical access

To configure Historical Access, you must first set up the History Adapters. These are the different locations for storing historical data, such as RAM, file, SQL Server.

History Adapters are added to the configuration using the context menu command **Add new History Adapter**.



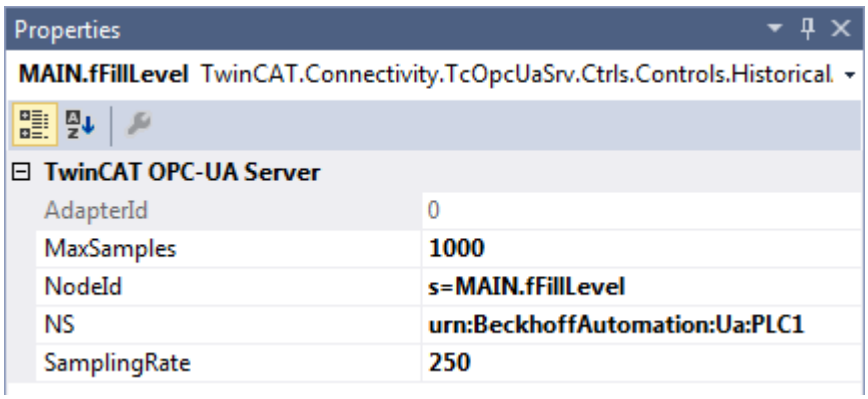
Depending on the adapter type you have to specify further parameters, e.g. the desired file storage path or the access data for the SQL Server.



After you have created a History Adapter you can add the desired variables to the adapter. These variables must already exist on the selected OPC UA Server when the engineering is implemented. You can use the integrated **OPC UA Target Browser** to select the variables and then add the variables from the target browser to the History Adapter by drag and drop.



Additional parameters can be specified in the properties window of the newly added variable, e.g. the desired SamplingRate or the size of the ring buffer to be used in the History Adapter.

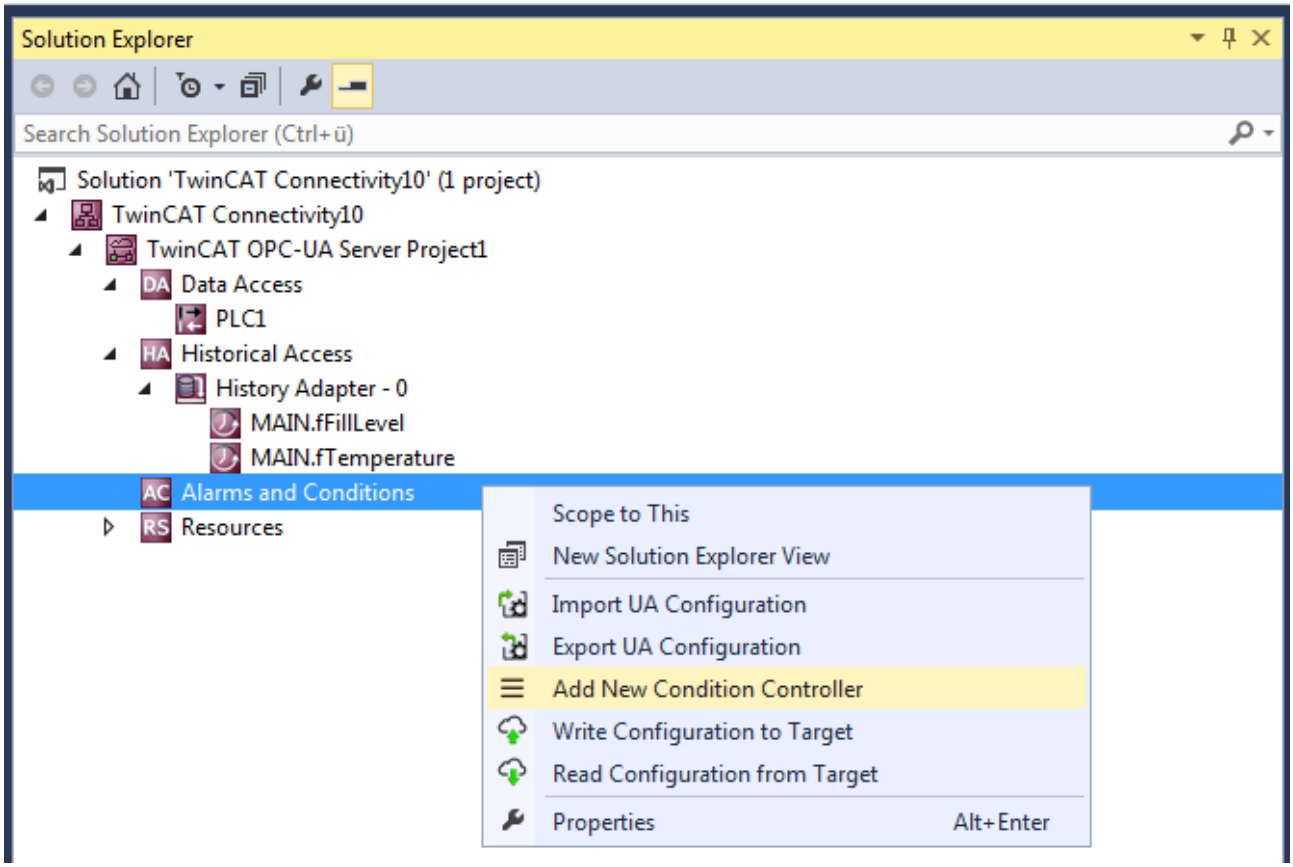


See also: Connecting to a server [▶ 22]

4.3.9 Configuring Alarms and Conditions

In order to configure Alarms and Conditions (A&C) you must first set up the Condition Controllers. These are container units that group together alarms.

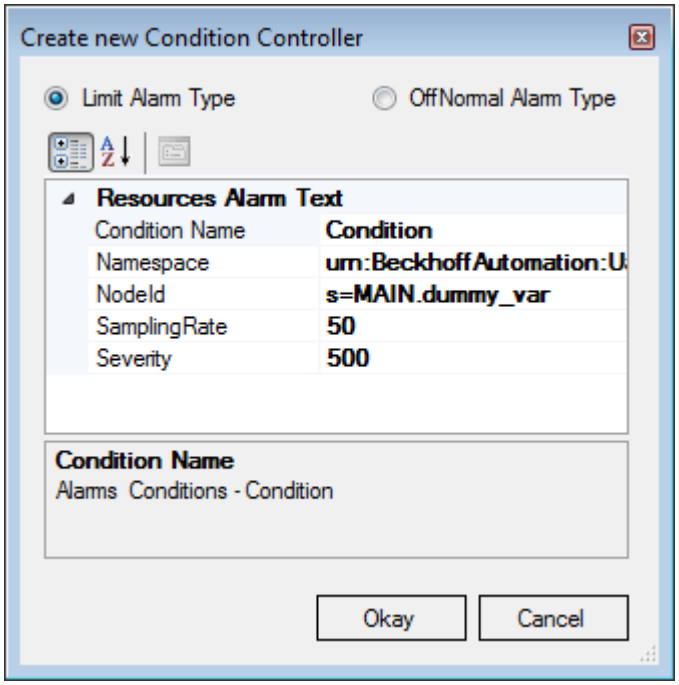
Condition Controllers are added to the configuration using the context menu command **Add New Condition Controller**.



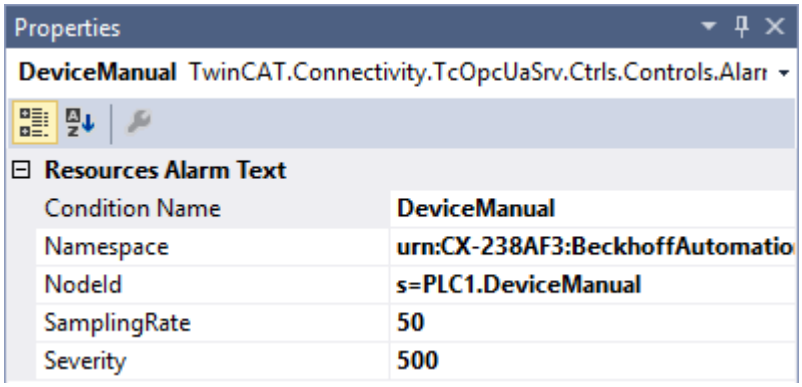
Once you have created a Condition Controller, add the desired variables to the controller and monitor them in terms of alarms and conditions. A Condition is created for each variable, which specifies the parameters for monitoring. These variables must already exist on the selected OPC UA Server when the engineering is implemented. You can use the integrated **OPC UA Target Browser** to select the variables and then add the variables from the Target Browser to the Condition Controller by drag and drop.



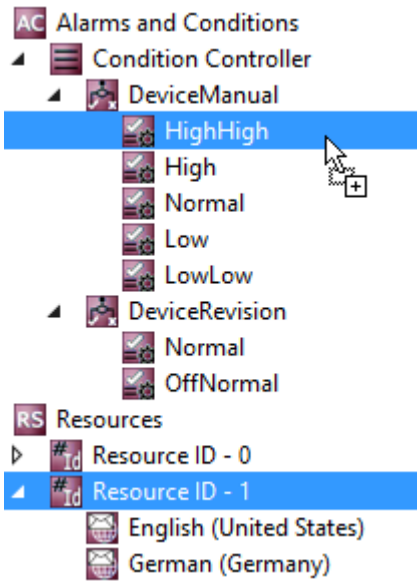
In the dialog box which then opens you can define the Condition type and further parameters for the monitoring, e.g. SamplingRate and Severity.



Depending on the Condition type selected, specify additional parameters in the Condition properties window. The threshold values for the respective Condition type are displayed as individual entries in the tree view of the configuration. Here too, you can configure the corresponding parameters in the properties window.



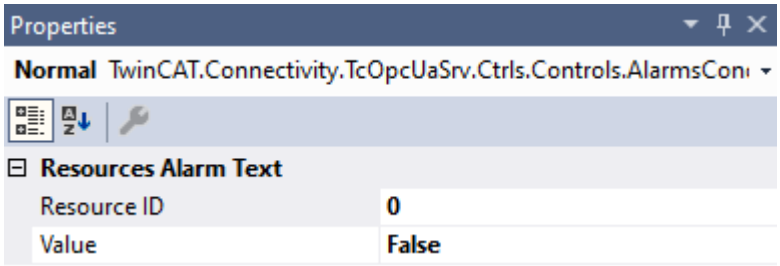
Subsequently you have to define the alarm texts that are to be sent to the OPC UA Client when a Condition is triggered. How to create alarm texts is described in the chapter Configuring alarm texts [▶ 34]. You can drag and drop the alarm texts onto the respective threshold value of a Condition.



Alarm type OffNormal

An OffNormal alarm type is used to define which state of a Boolean variable is evaluated as normal. An alarm is triggered if the variable value deviates from this. The PLC must be used for working with value ranges (e.g., integer or double variables). Depending on the value, a corresponding TRUE or FALSE state is then passed to the OPC UA Server.

State	Value range
Normal	TRUE or FALSE, depending on the user's decision.
OffNormal	TRUE or FALSE, depending on the configuration of the normal state. Cannot be configured by the user.

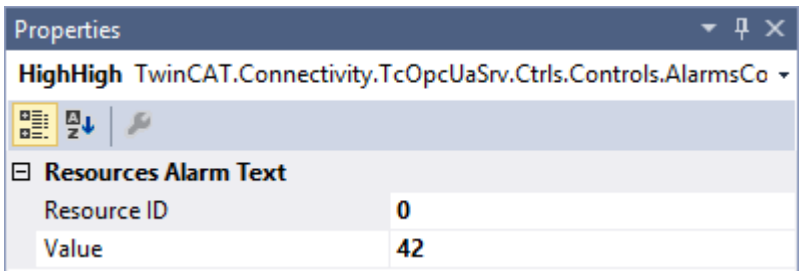


The first step is to configure the normal state as described above. The user then defines an alarm text for the respective state (OffNormal and Normal) via Resources. This can be done either by drag and drop or by selecting from the **Resource ID** drop-down list.

Alarm type limit

With an alarm type limit you define different threshold values upon whose reaching an alarm is to be sent. The following table describes the different threshold values using an example configuration.

State	Example threshold values	Associated value range (INT)
HighHigh	5000	5000-32767
High	2000	2000-4999
Normal	-	1000-1999
Low	1000	500-999
LowLow	500	-32768-499



In the first step, the various threshold values are configured as described above. The user then defines an alarm text for the respective state (HighHigh, High, Normal, Low, LowLow) via Resources. This can be done either by drag and drop or by selecting from the **Resource ID** drop-down list.

Requirements

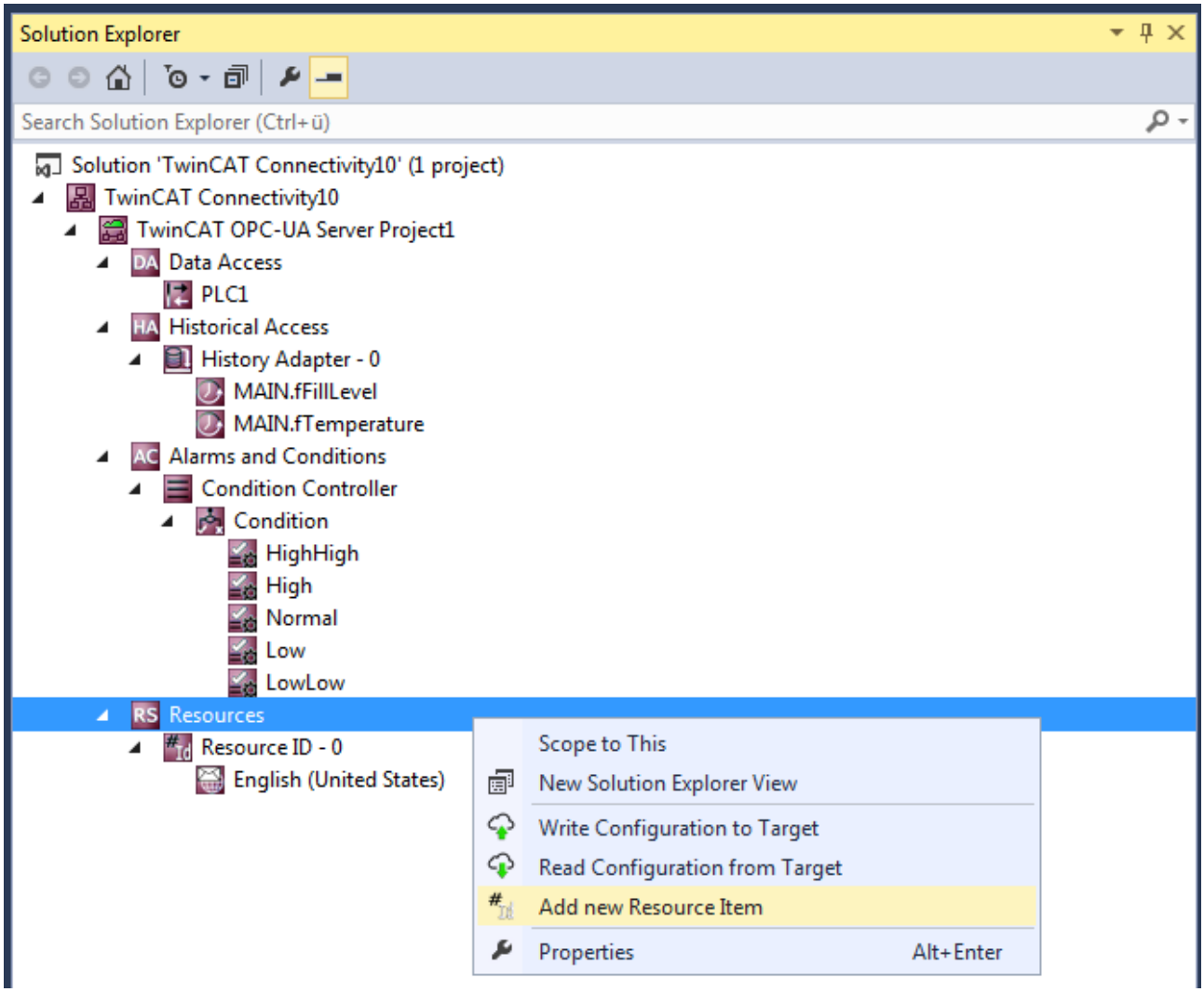
Products	Setup versions	Target platform
TF6100	4.x.x	IPC or CX (x86, x64, Arm®)

See also: Connecting to a server [▶ 22]

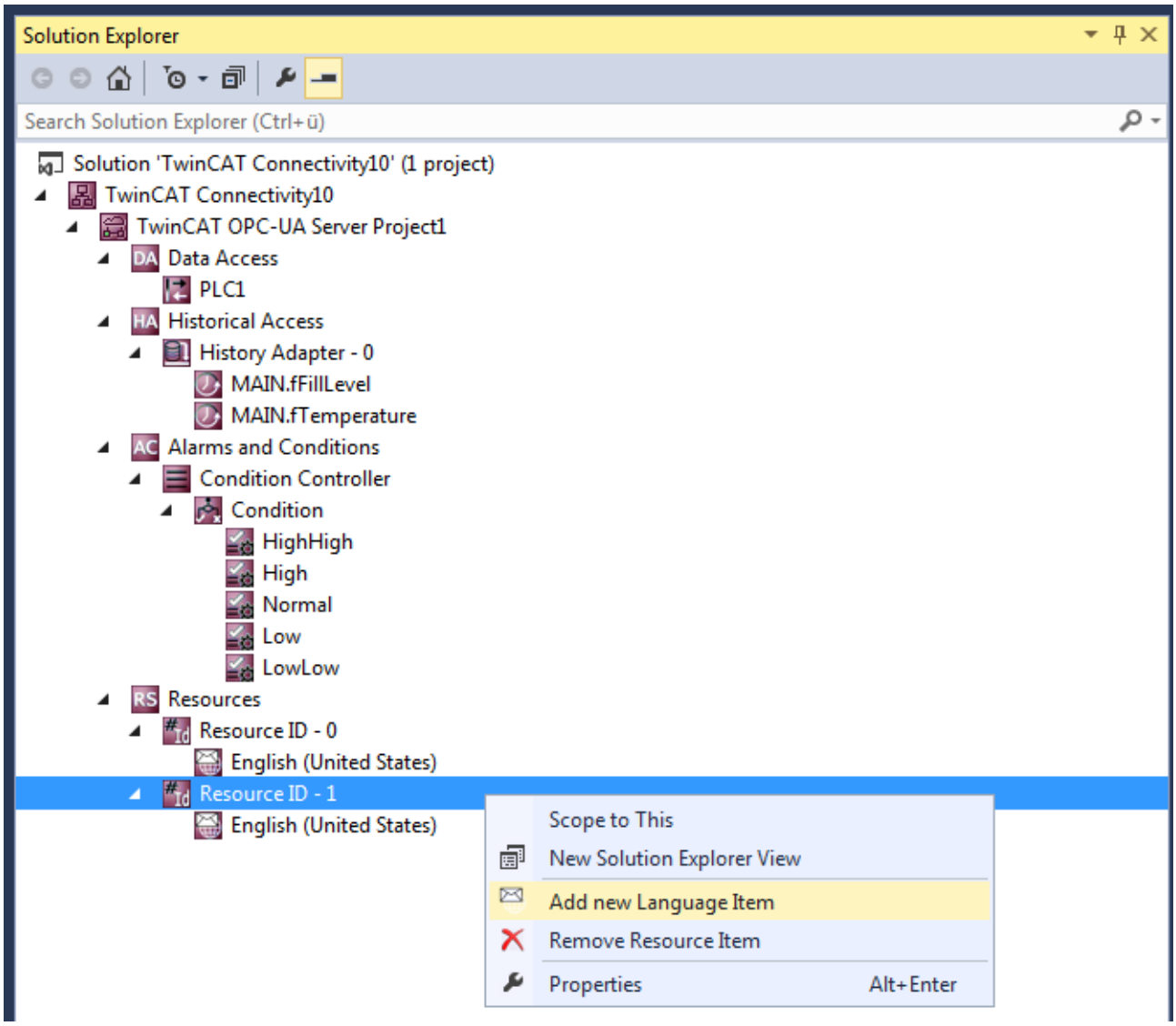
4.3.10 Configuring alarm texts

The OPC UA Configurator enables the (multilingual) management of alarm texts that are used, for example, with Alarms and Conditions [▶ 31]. The configuration of the alarm texts takes place in the **Resources** facet. Each alarm text is identified by a unique ID. Multiple language texts can then be assigned to this ID.

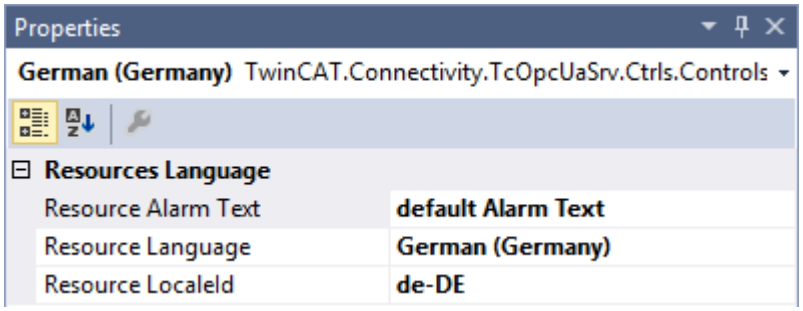
You can create so-called Resource Items using the context menu command **Add new Resource Item**.



You add new Language Items to a Resource Item using the command **Add new Language Item** in the context menu of a Resource Item.



You can further parameterize a Language Item, e.g. the language text and the assigned language, in the properties window. When you define the language, the associated LocaleID is automatically set. The LocaleID is requested by the OPC UA Client to indicate in which language it expects alarm texts.



Requirements

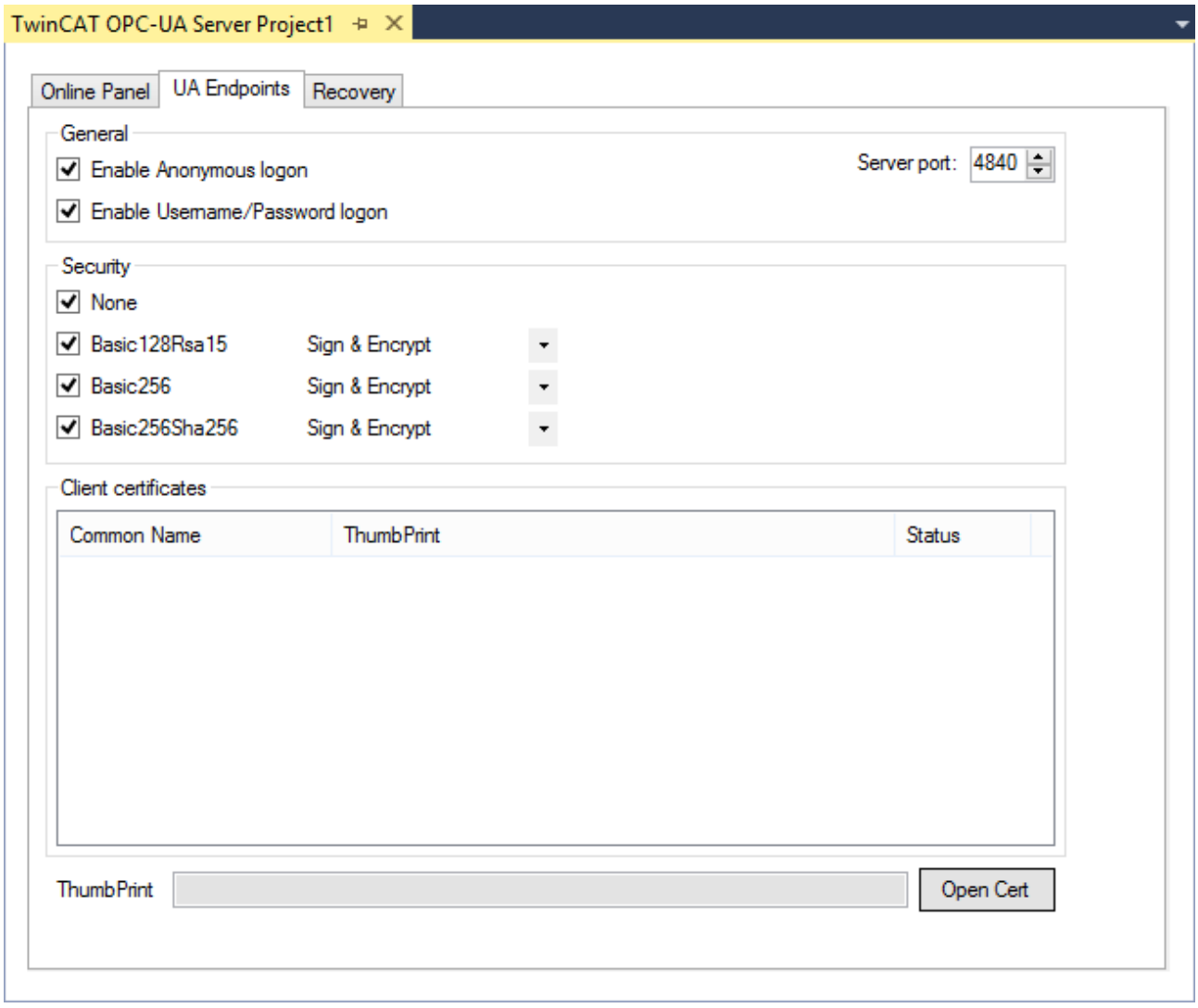
Products	Setup versions	Target platform
TF6100	4.x.x	IPC or CX (x86, x64, Arm®)

4.3.11 Configuring endpoints

The endpoints of the OPC UA Server indicate which security mechanisms are to be used during the connection establishment of a client. These range from "unencrypted" to "encrypted and signed", based on different key strengths.

The endpoints can be activated and deactivated using the configurator. It may be useful to deactivate the unencrypted endpoint so that all clients can only connect themselves with valid certificates that are classified as trustworthy.

The endpoints are configured directly at the level of the OPC UA Server project. By double-clicking on the project you can make the corresponding settings on the **UA Endpoints** tab. The settings become effective after an activation of the configuration and a subsequent restart of the server (see Reading and writing the configuration [▶ 27] and Restarting the server [▶ 45]).



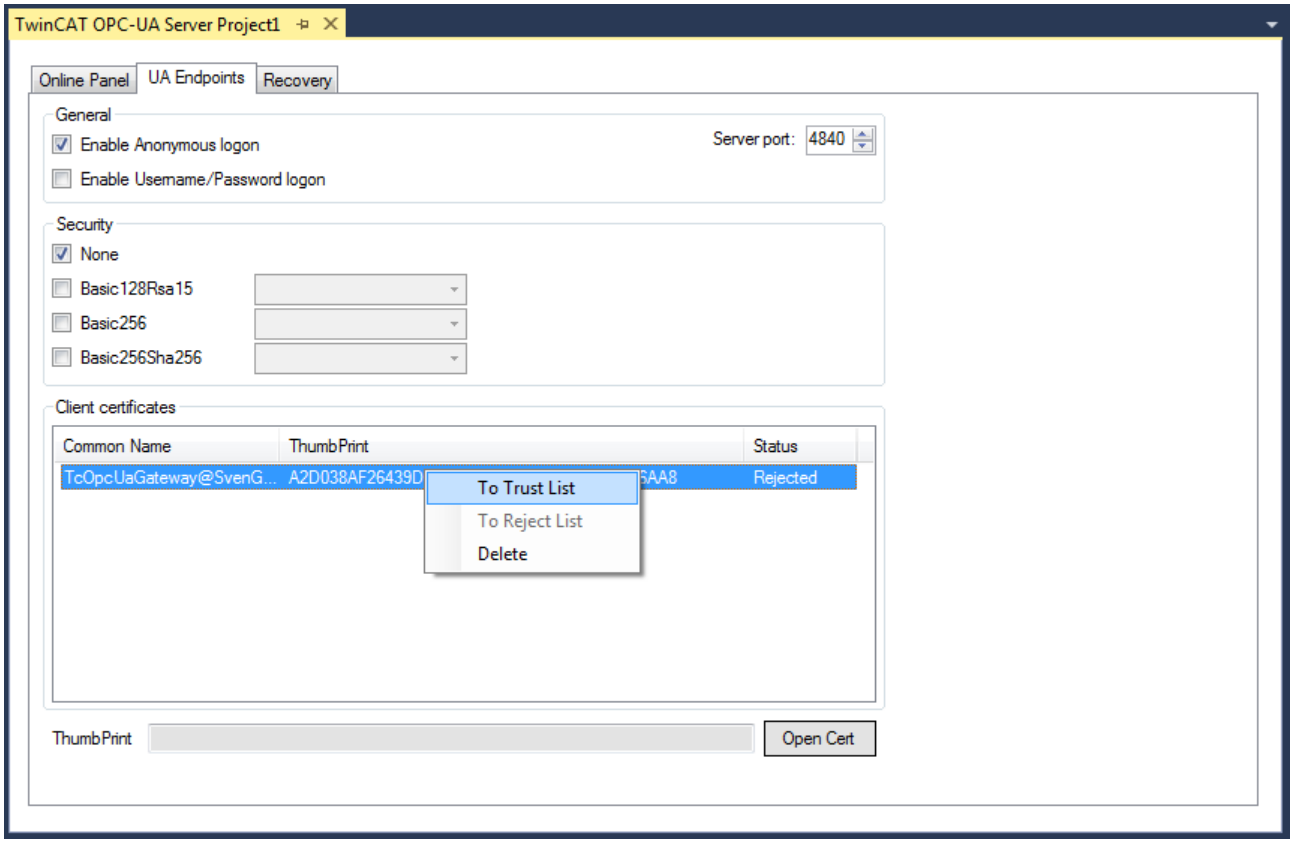
Requirements

Products	Setup versions	Target platform
TF6100	4.x.x	IPC or CX (x86, x64, Arm®)

4.3.12 Trust relationship for certificates

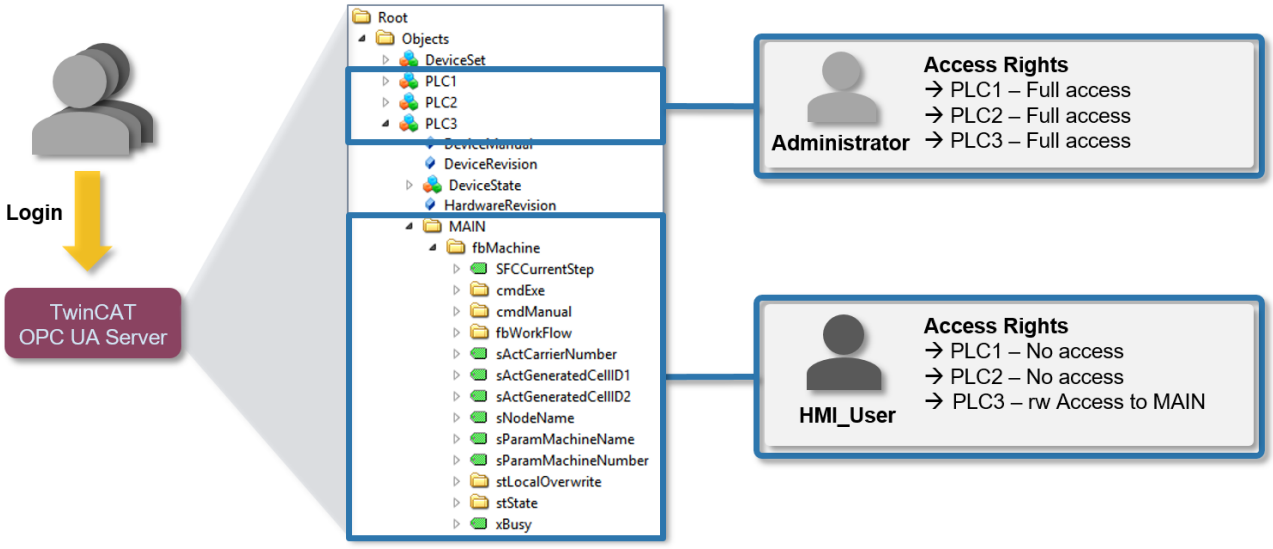
The Configurator facilitates management of the client certificates on the server. In the project settings you can classify the certificates as trustworthy or refuse them on the **UA Endpoints** tab in the **Client certificates** area.

After an OPC UA Client has attempted to connect to a secure server endpoint for the first time, the client certificate is deposited on the server and declared "rejected". The server administrator can subsequently enable the certificate. A subsequent connection attempt of the client with a secured endpoint will then be successful.



4.3.13 Configuring security settings

The OPC UA Server enables the configuration of permissions at namespace and node level. This allows you to fine-granulate the access to ADS devices (for example, to different PLC runtimes) as well as variables. These security settings are available for all ADS devices that can be displayed in the server namespace.



Configuration

The permissions are configured on the basis of an XML-based configuration file (*TcUaSecurityConfig.xml*), which is located in the same directory as the server. The configuration file consists of the three areas **Users**, **Groups**, and **AccessInfos**.

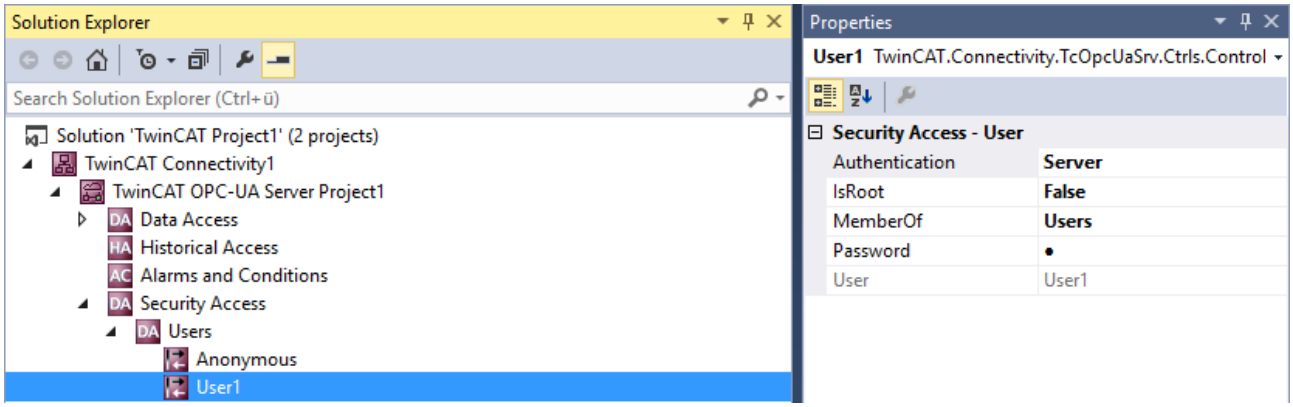
Users

In the **Users** area you can configure user accounts that are to be accepted by the OPC UA Server as logins. There are three different authentication methods:

OS (recommended authentication method)	The mechanisms of the operating system are used to validate user name and password. The user account is subject completely to the control of the operating system and/or domain.
Server (not recommended)	User name and password are known only to the OPC UA Server. Both pieces of information are stored in plain text in the XML file.
None	Only the user name of the server is evaluated, the password is ignored.

Users can be configured with a tag <DefaultAccess> that specifies the standard access of the user to a certain namespace.

Users can be members of one or more groups. You can specify this using the **MemberOf** attribute. In case of memberships of several groups, separate the groups by a semicolon.

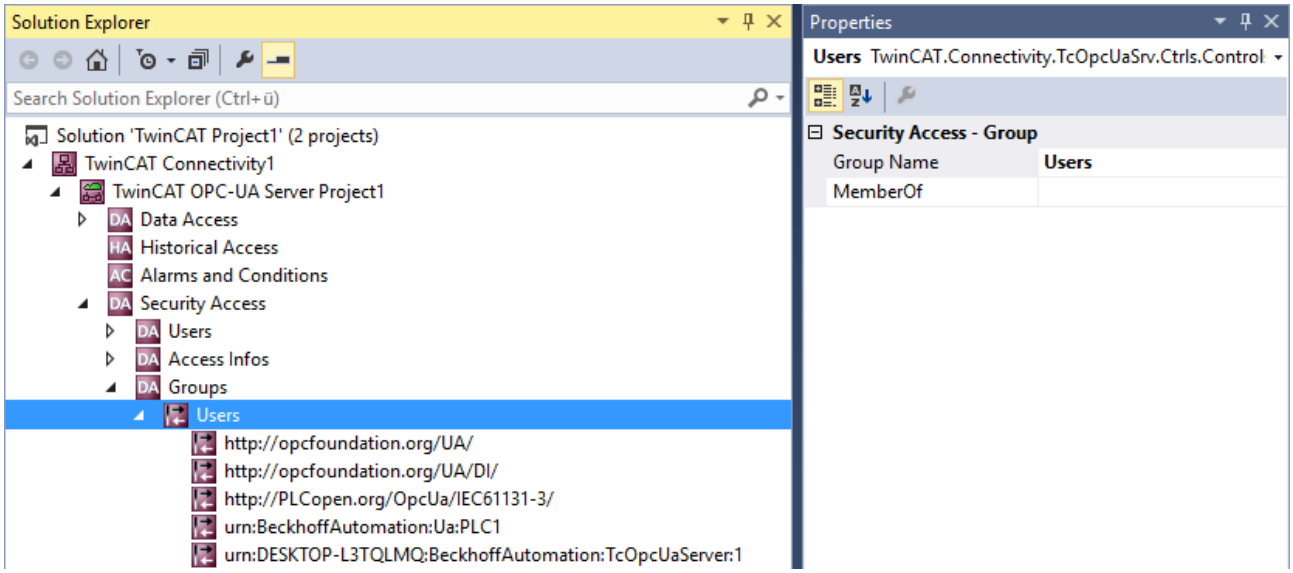


Groups

In order to enable a simpler configuration with several user accounts, you can combine the users into groups.

Groups can also be configured with a tag <DefaultAccess>.

You can nest groups using the **MemberOf** attribute. In case of memberships of several groups, separate the groups by a semicolon.

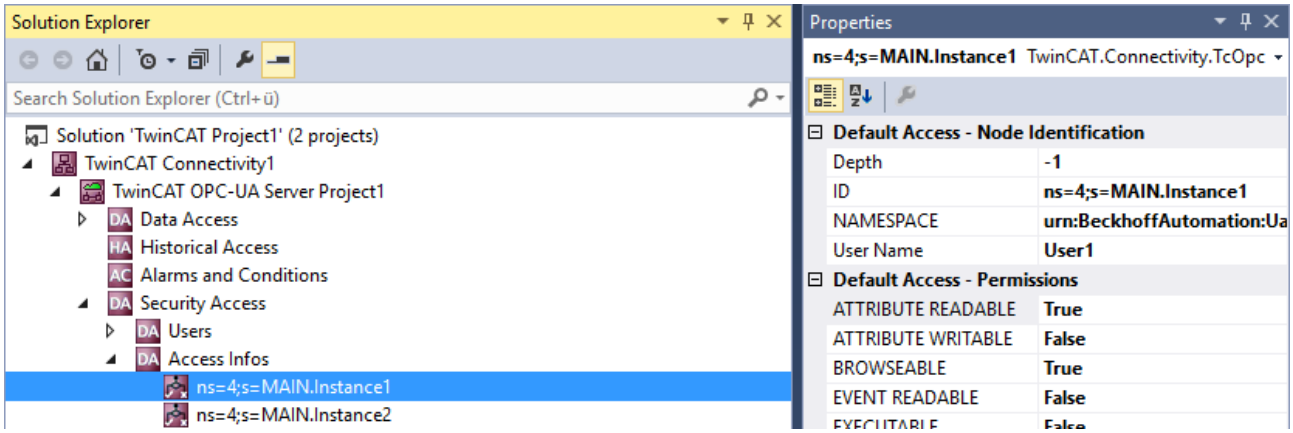


AccessInfos

If a fine-granular setting of permissions at the node level is to be implemented, then AccessInfos can be configured additionally, which specify the access permissions on nodes. Access rights can be passed on to subelements. Although AccessInfos allow the most fine-grained configuration of permissions, such a configuration can quickly become confusing. Therefore, check whether configuring access rights at the namespace level (see above) is not sufficient.

The AccessInfo for a node contains the following settings:

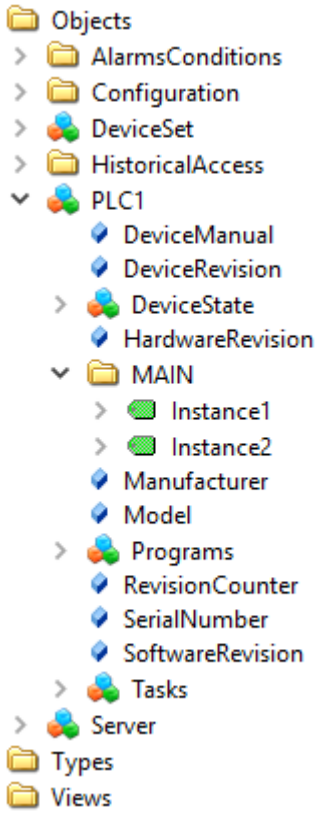
NS	Configures the NamespaceName in which the node is localized
Id	Configures the identifier of the node, including the IdentifierType (e.g. s = String)
Depth	Inheritance depth of permissions (-1 for infinite)
User/Group	User or group that is to be given access to this node, including the AccessLevels



AccessInfos can be configured by dragging & dropping variables from the Target Browser. The configurable permissions are cumulative.

Sample configuration

Let's take the following simple control program. The variables are already published in the OPC UA namespace of the server. The OPC UA Server is initially in the delivery state.



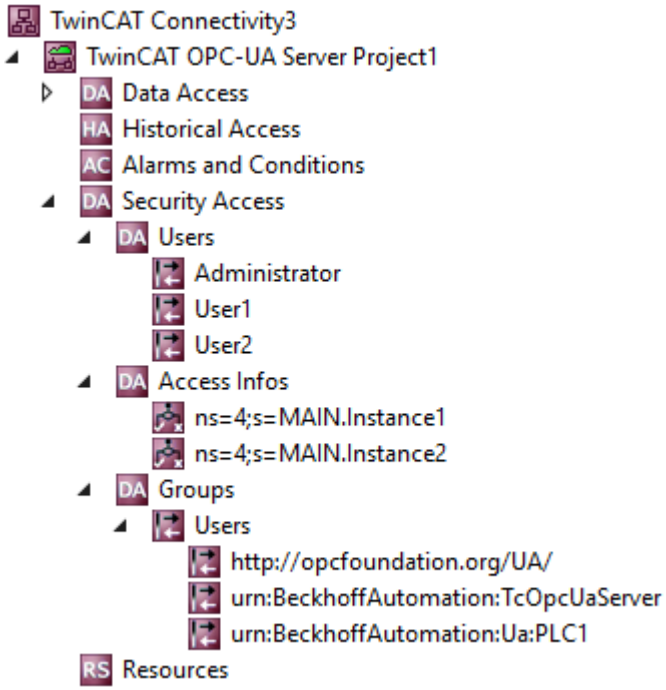
Access restrictions

Access to the server is to be restricted for clients as follows:

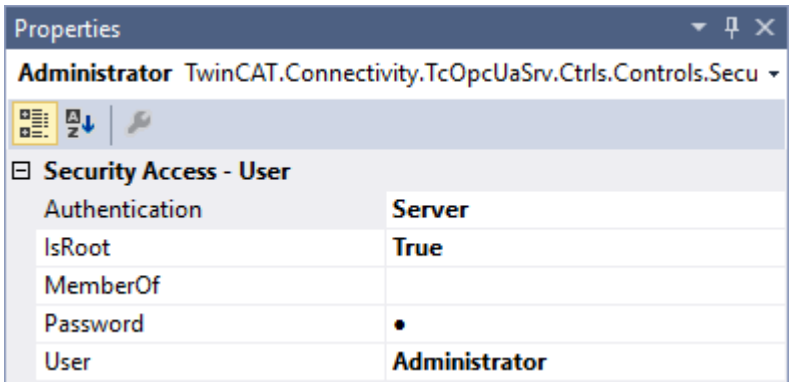
- Anonymous access is to be deactivated.
- There is to be a user - "Administrator" - who has full access to the complete server.
- There is to be a user - "User1" - who only has read access to MAIN.Instance1. The user should not come from the operating system here, but should only be used internally in the server.
- There is to be a user - "User2" - who only has read access to MAIN.Instance2. The user should not come from the operating system here, but should only be used internally in the server.
- General access permissions are to be configured for all users via a group called "Users".

Settings

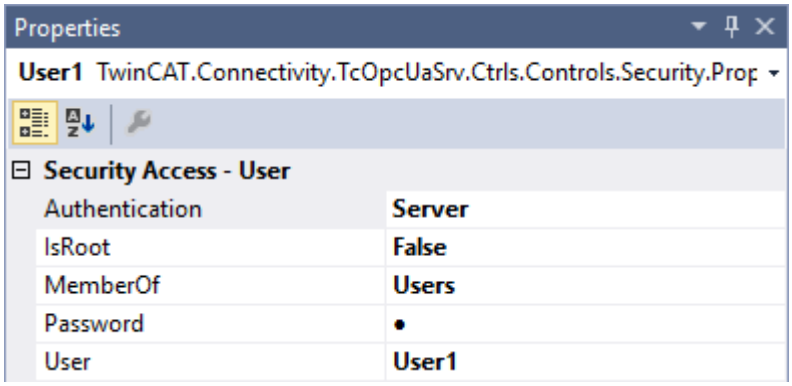
The configuration of the OPC UA Server is set as follows:



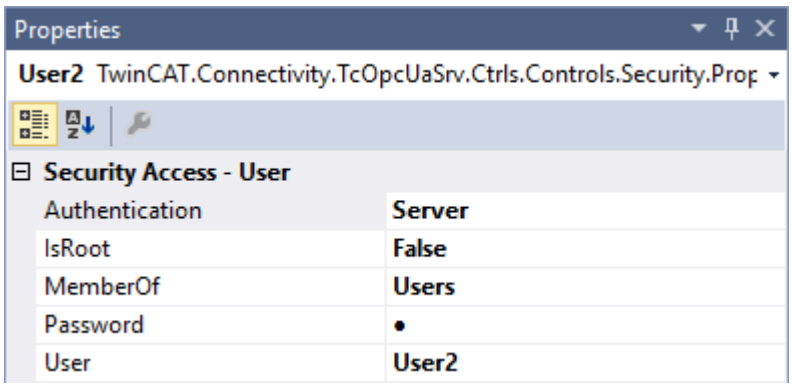
Settings for the user "Administrator":



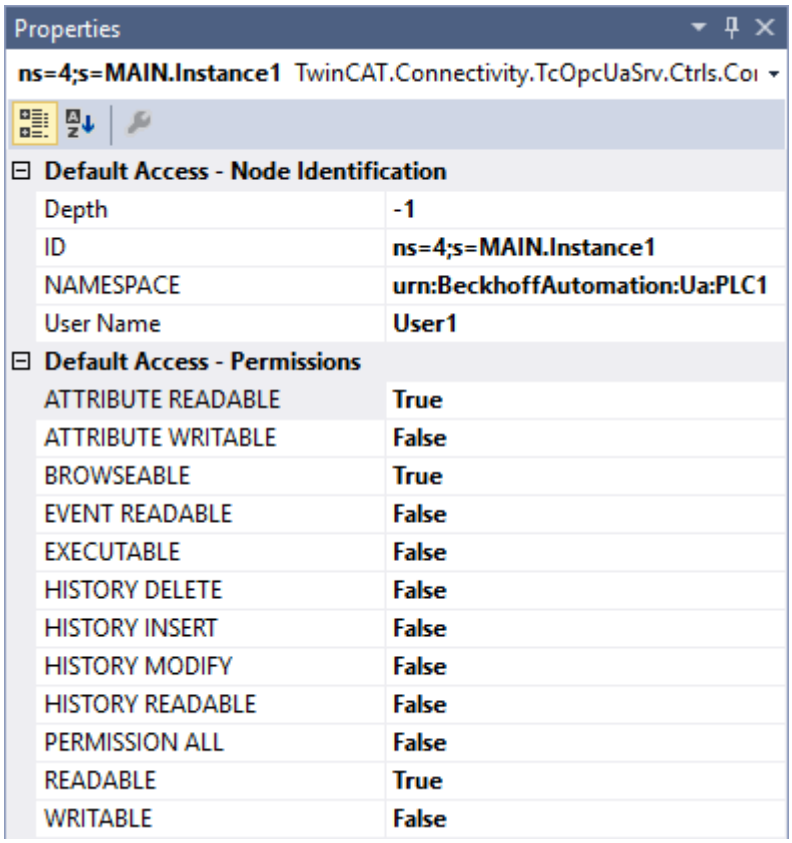
Settings for the user "User1":



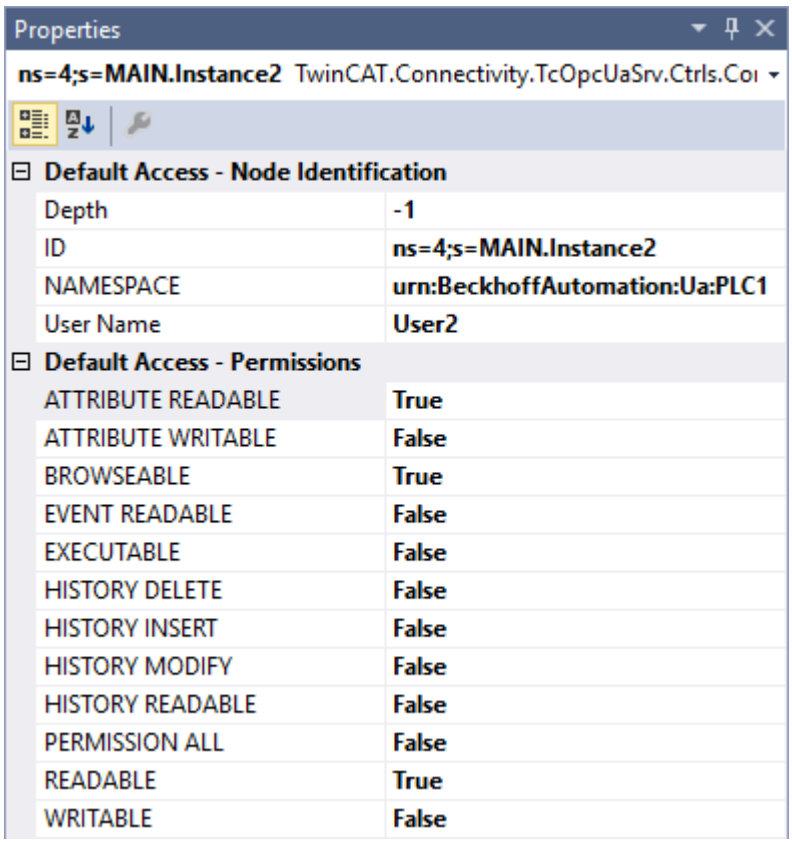
Settings for the user "User2":



Settings for AccessInfos "MAIN.Instance1":

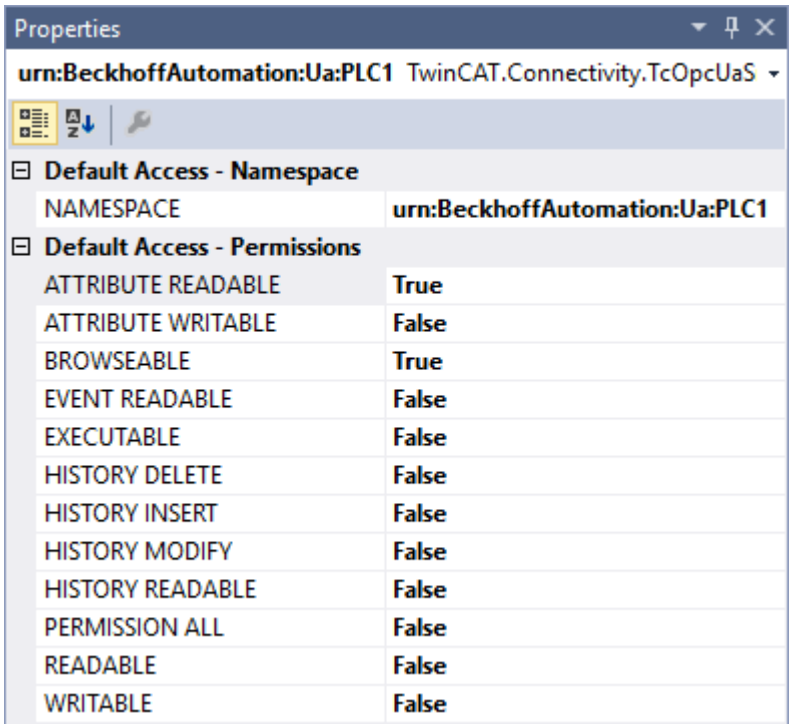


Settings for AccessInfos "MAIN.Instance2":



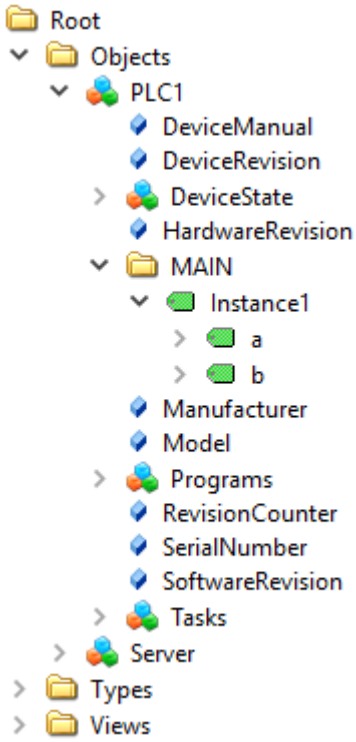
Settings for the group "Users":

The user group is equipped both with basic access to required server and type system namespaces and with read and browse permissions to the PLC1 namespace.



Result

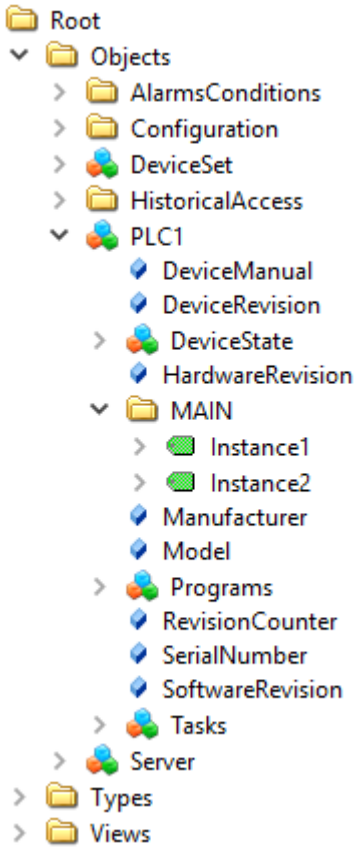
Following activation of the configuration, the namespace of the server for "User1" looks like the following after establishment of a connection:



The user has only read rights to the node "Instance1", which is clear from the attribute UserAccessLevel:

DataType	ST_Test
NamespaceIndex	4
IdentifierType	String
Identifier	<StructuredDataType>:ST_Test
ValueRank	-1
ArrayDimensions	BadAttributeIdInvalid (0x80350000)
AccessLevel	CurrentRead, CurrentWrite
UserAccessLevel	CurrentRead

The user "Administrator", conversely, has full access rights to all elements of the namespace:



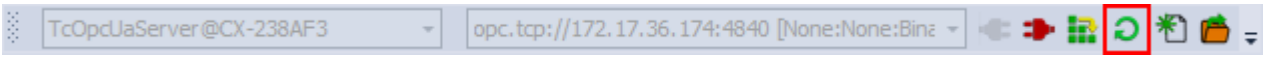
4.3.14 Restarting the server

The OPC UA Configurator enables the triggering of a restart of the OPC UA Server. This can be done locally or remotely and refers to the selected target device.

Loss of connection

A restart of the OPC UA Server always leads to a loss of the connection of all connected clients.

The restart is triggered via the toolbar.



Requirements

Products	Setup versions	Target platform
TF6100	4.x.x	IPC or CX (x86, x64, Arm®)

See also: Connecting to a server [▶ 22]

4.3.15 Logging

For an advanced diagnostics you can activate the logging function of the OPC UA Server.

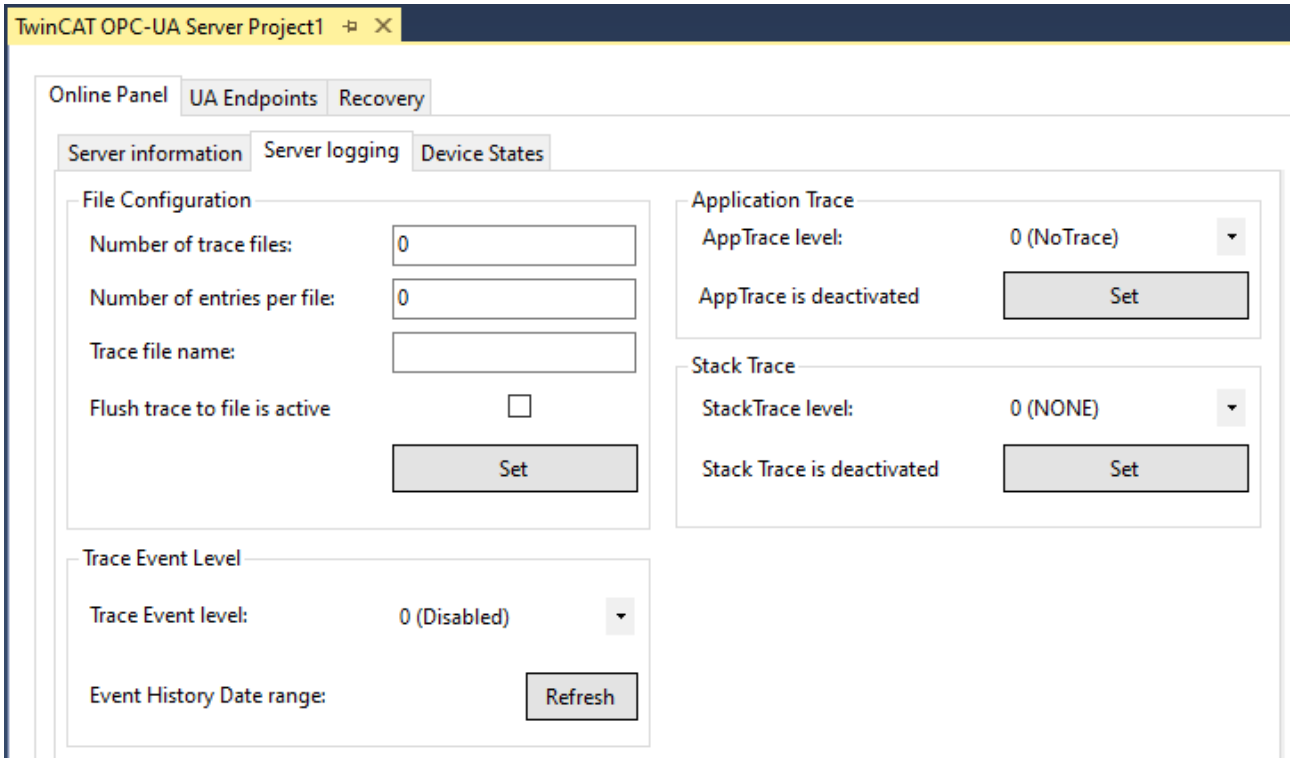
Writing the log file

Activating the logging function on the server causes a log file to be written on the file system. Make sure that there is sufficient storage space available and set the logging parameters accordingly (number of log files, size per log file).

Performance and timing behavior

Activation of the logging function will change the timing behavior of the OPC UA Server. As a result there may be losses of speed, depending on the platform and project.

The logging function is activated using the **Activate** button on the **Online Panel** tab in the project configurator.You can activate the function locally or remotely depending on the selected target device. The logging function remains active until it is deactivated again via the configurator or until the OPC UA Server is restarted.



Trace Level

In general, the higher the trace level, the more detailed (and more) data is written, but the more load is also placed on the server application, which changes the timing behavior accordingly. Please therefore only activate logging in the event of diagnostics and in consultation with Beckhoff Support.

Activate App Trace

In most cases it is sufficient to create a so-called "AppTrace". This logs information from the server application. To activate the AppTrace, please enter the number of TraceFiles and the number of entries per TraceFile in the corresponding text fields. Then select a trace level and click the button to activate the AppTrace. The values in the gray text boxes represent the current settings on the server.

Activate Stack Trace

In a few cases it is also necessary to create a so-called "StackTrace", whereby information from the OPC UA stack is logged. To activate the StackTrace please enter the number of TraceFiles as well as the number of entries per TraceFile into the corresponding text boxes. Then select a trace level and click on the button to activate the StackTrace. The values in the gray text boxes represent the current settings on the server.

Requirements

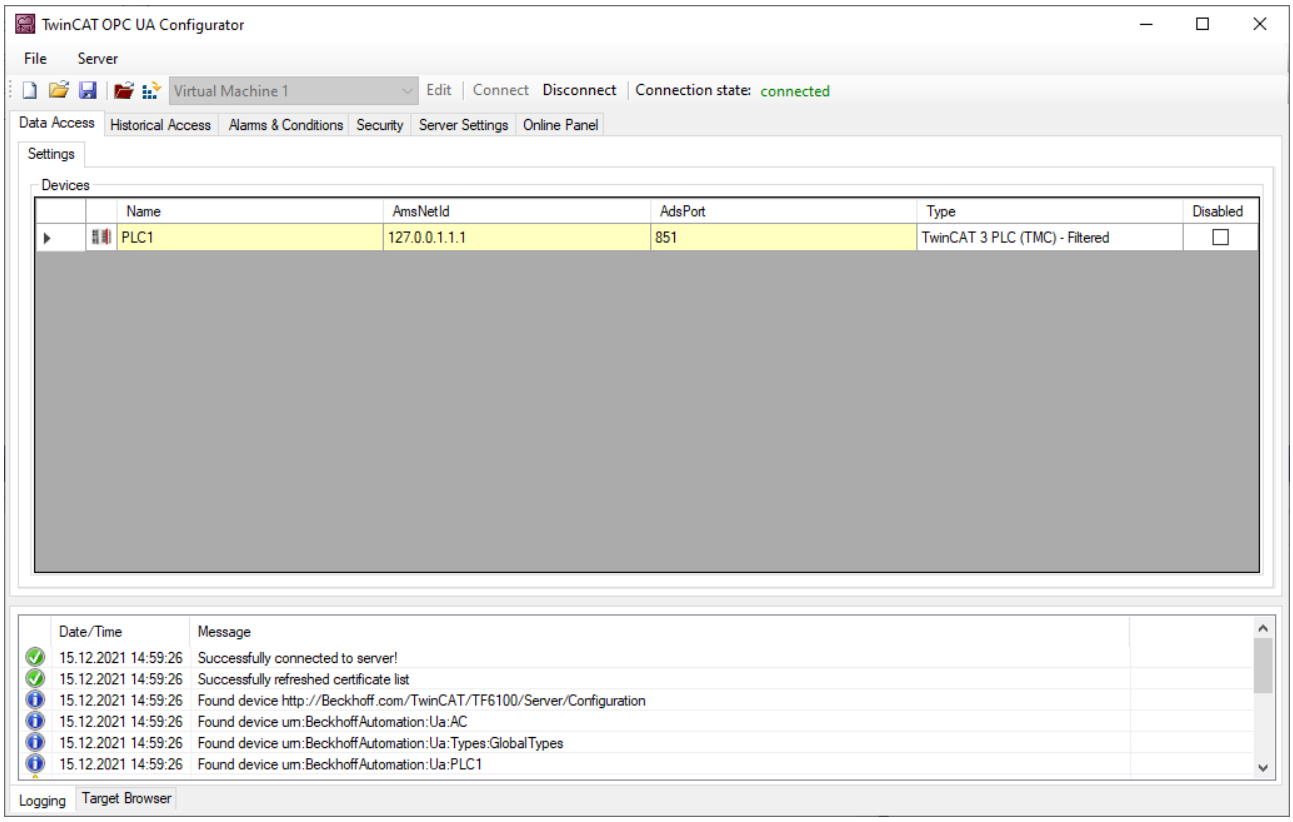
Products	Setup versions	Target platform
TF6100	4.x.x	IPC or CX (x86, x64, Arm®)

See also: Selecting a target device [▶ 22]

4.4 Standalone

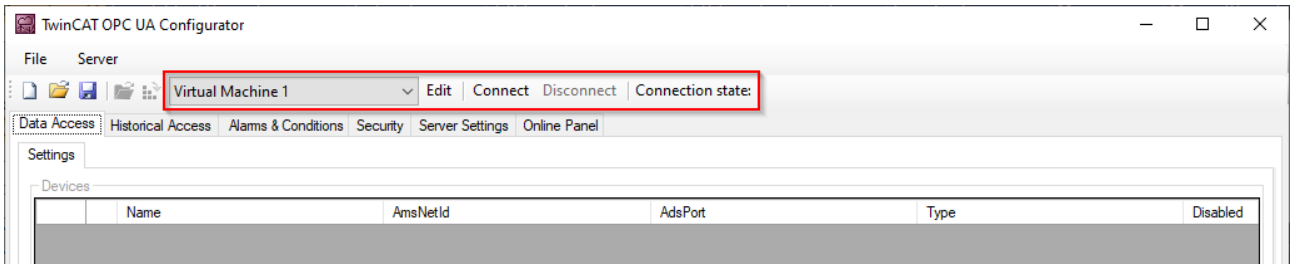
4.4.1 Overview

The standalone configurator enables parameterization of the TwinCAT OPC UA Server independently of Visual Studio. You can configure all the different features of the server.

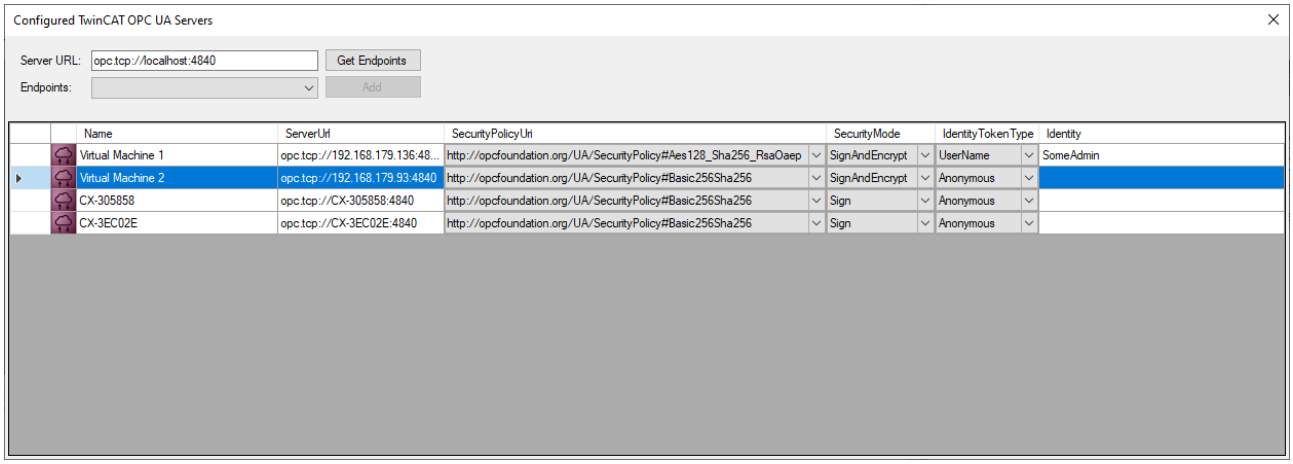


4.4.2 Connecting to a server

The OPC UA Configurator enables the complete parameterization of the Server via OPC UA. Similar to the TwinCAT XAE system, you can select an OPC UA Server to connect to via the toolbar.



Click on the **Edit** button to open the server list dialog. In this dialog you can add one or more server connections.



By entering a ServerURL and pressing the **Get Endpoints** button, a server connection can be added to the list. Any settings for the IdentityToken, e.g. whether the Configurator should connect as an anonymous user or with a user name/password combination, must be set manually.

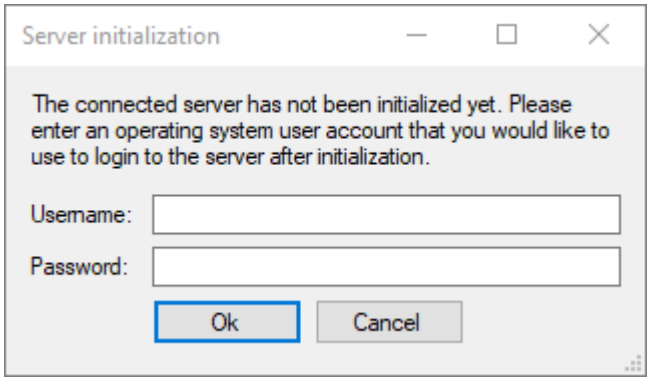
Confirming a configuration

Please always confirm changes to the entries with the **ENTER** key, as only then will they be automatically saved in the background.

After configuring a server connection, the corresponding entry is available in the DropDownBox and the connection can be established by clicking the **Connect** button.

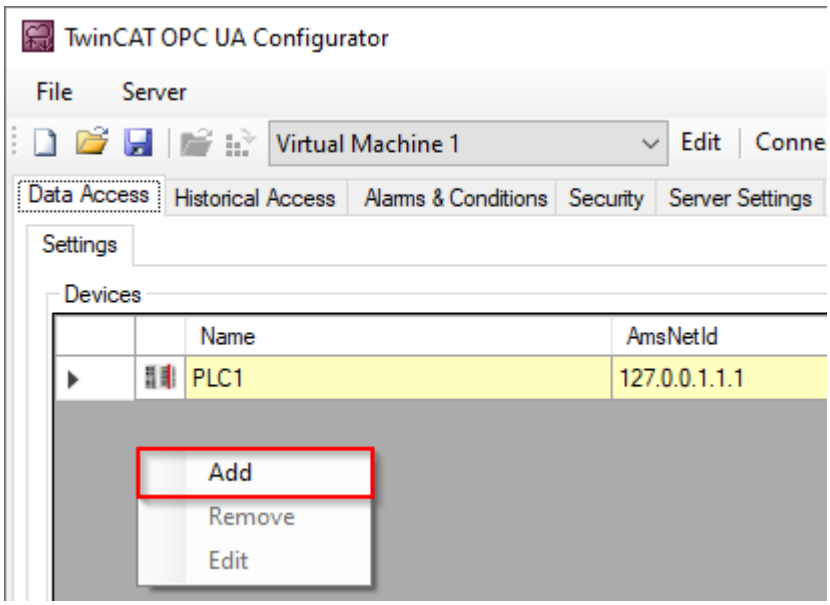
4.4.3 Performing the server initialization

The TwinCAT OPC UA Server is delivered in an uninitialized mode, which is based on the so-called TOFU (Trust-On-First-Use) principle. Detailed information about this server feature and the corresponding background information can be found here. The TwinCAT OPC UA Configurator enables the initialization of the server during the first connection establishment. A corresponding warning message indicates the uninitialized server and enables an appropriate initialization.

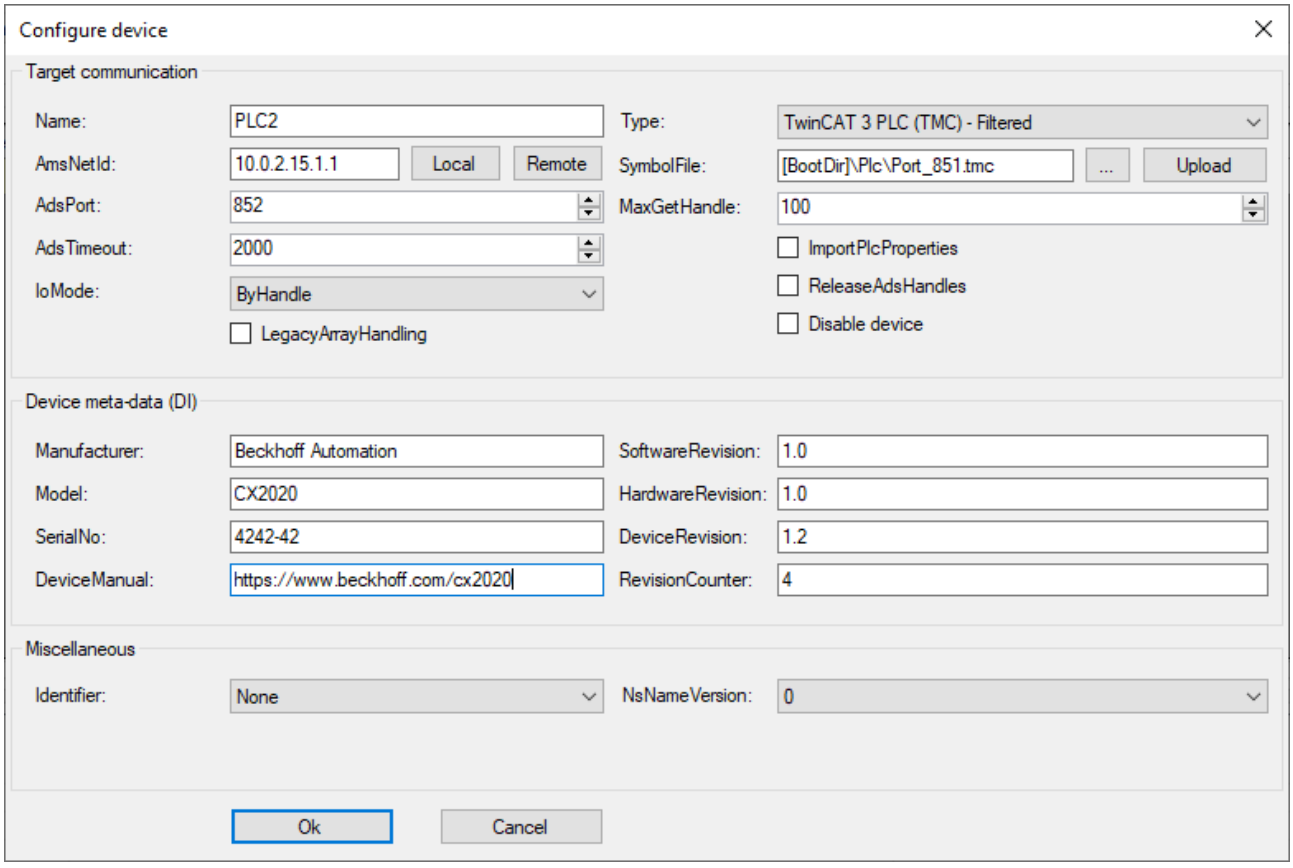


4.4.4 Adding ADS devices

ADS devices can be added to the TwinCAT OPC UA Server configuration via the **Data Access** tab. In the associated DataGrid, create a new device via the context menu.



Set the device-specific parameters in the subsequent dialog.



Selecting an AMS NetID

To select an AMS NetID, either the ADS devices from the local system or the connected TwinCAT OPC UA Server can be selected. An ADS device is a system that has an ADS route to the local system or server system. By clicking on the **Local** button the local ADS routes are displayed. By clicking the **Remote** button the ADS routes on the connected TwinCAT OPC UA Server are displayed.

Selecting a symbol file

The selection of a symbol file is always done from the local system. However, the symbol file can be uploaded to the connected TwinCAT OPC UA Server via the **Upload** button. The symbol file is stored in the subfolder "symbolfiles" of the TwinCAT OPC UA Server home directory and automatically referenced via a placeholder in the configuration file.

4.4.5 Reading and writing the configuration

The configurator enables both reading/writing of the configuration files from the TwinCAT OPC UA Server and loading/saving of the configuration files on the local system. These functionalities are available via the menu as well as the toolbar.

Local loading/saving

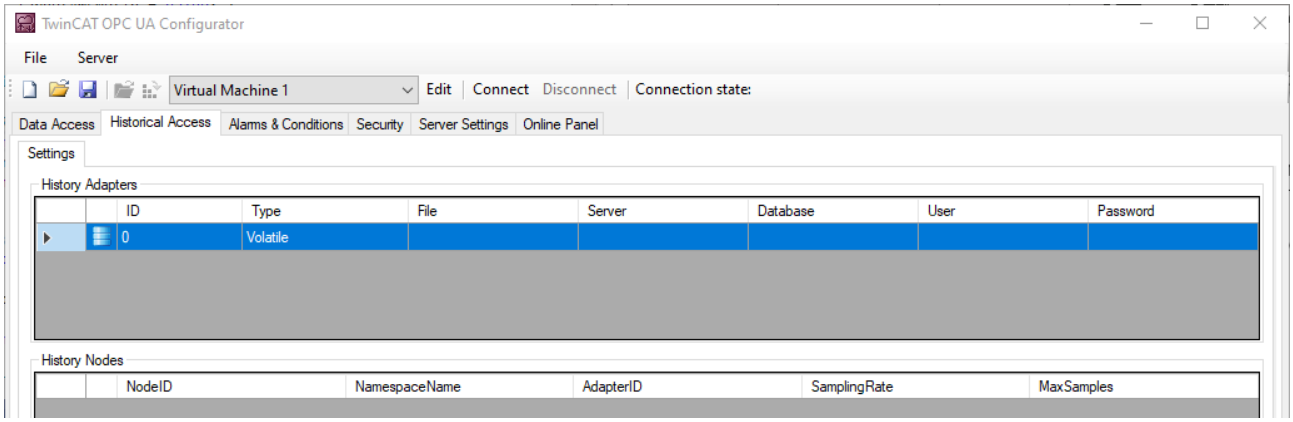
These functions are available via the **File** menu. The buttons available here **Open** and **Save** enable the configuration files to be loaded and saved. All configuration files are always loaded or saved.

Remote loading/saving

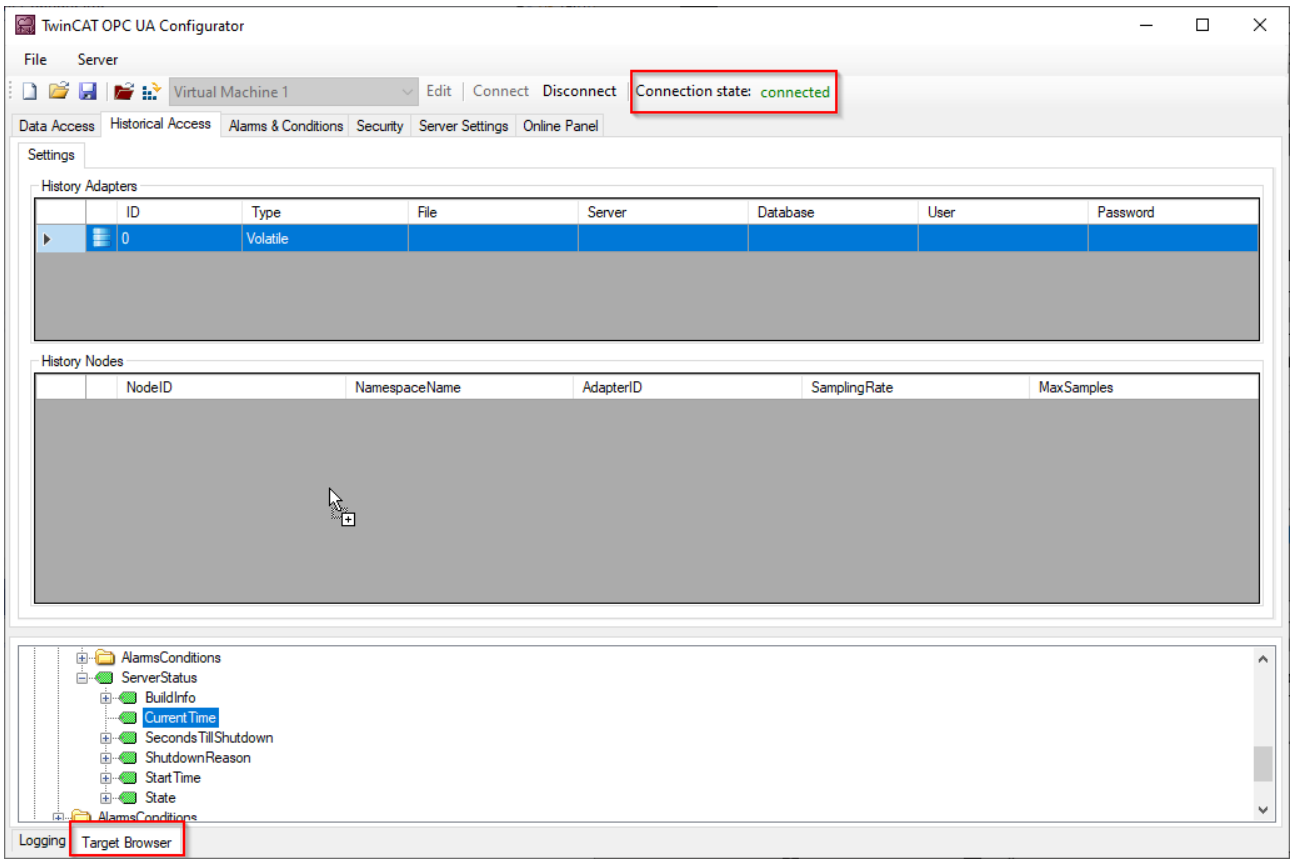
These functions are available from the **server** menu. The buttons **Open from target** and **Activate from target** available here enable loading and saving of the configuration files from the connected TwinCAT OPC UA Server. All configuration files are always loaded or saved.

4.4.6 Configuring historical access

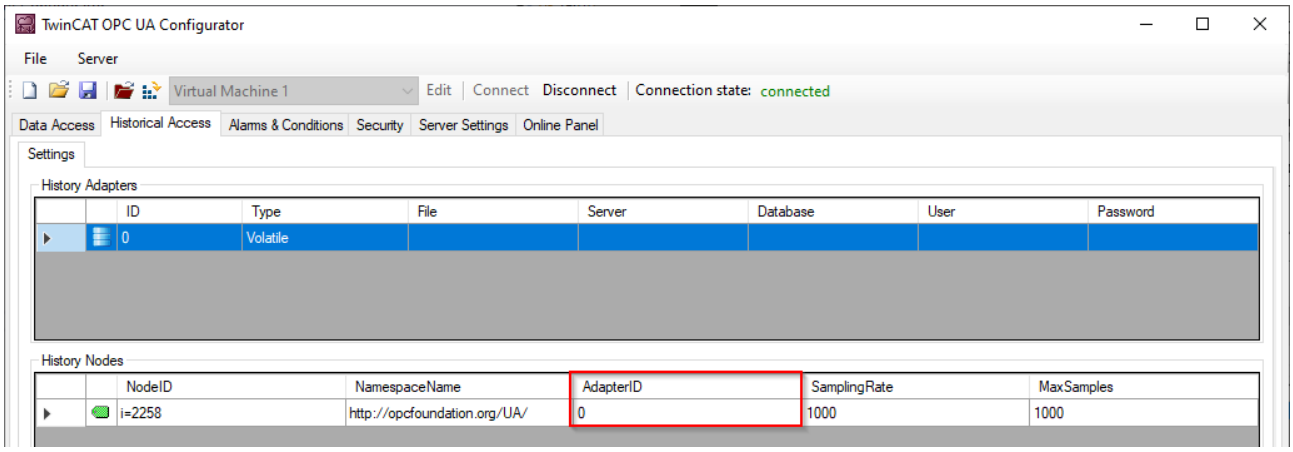
Use the **Historical Access** tab to configure both the **History Adapter** and the **History Nodes**. A **History Adapter** defines the type of data storage and a **History Node** the variable for which historical data should be saved in the data storage.



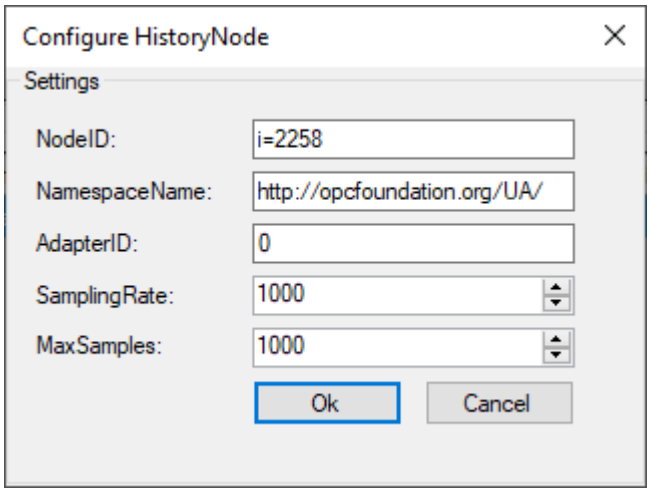
You can use the context menu to create both **History Adapter** and **History Nodes**. If you are connected to a TwinCAT OPC UA Server, you can also conveniently add the nodes to be configured via drag & drop from the **Target Browser** to the **History Nodes**.



Subsequently, a **History Node** can be linked to the respective **History Adapter** via the AdapterID.

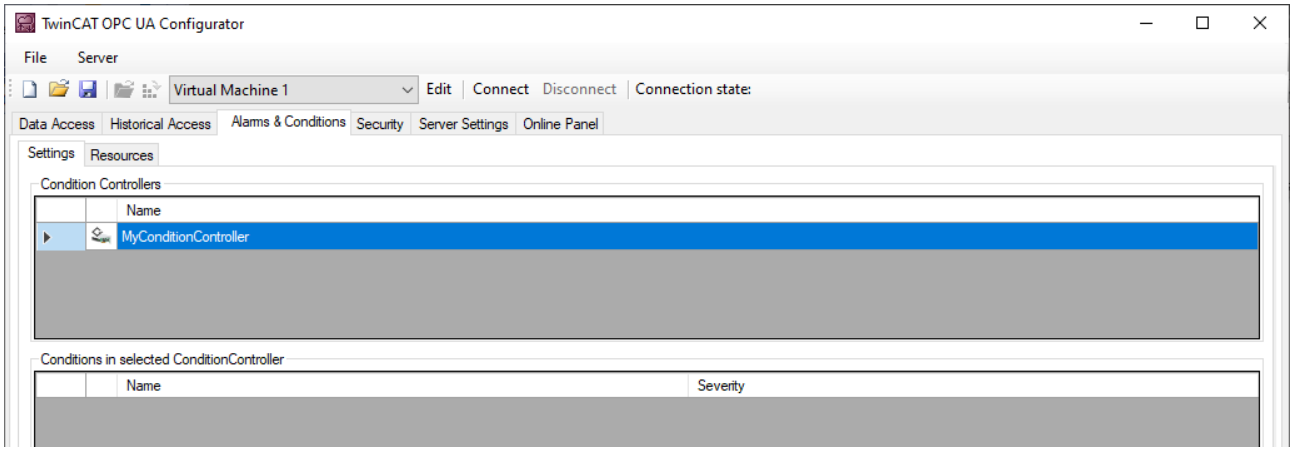


Double-click on the History Node to open the corresponding configuration dialog.

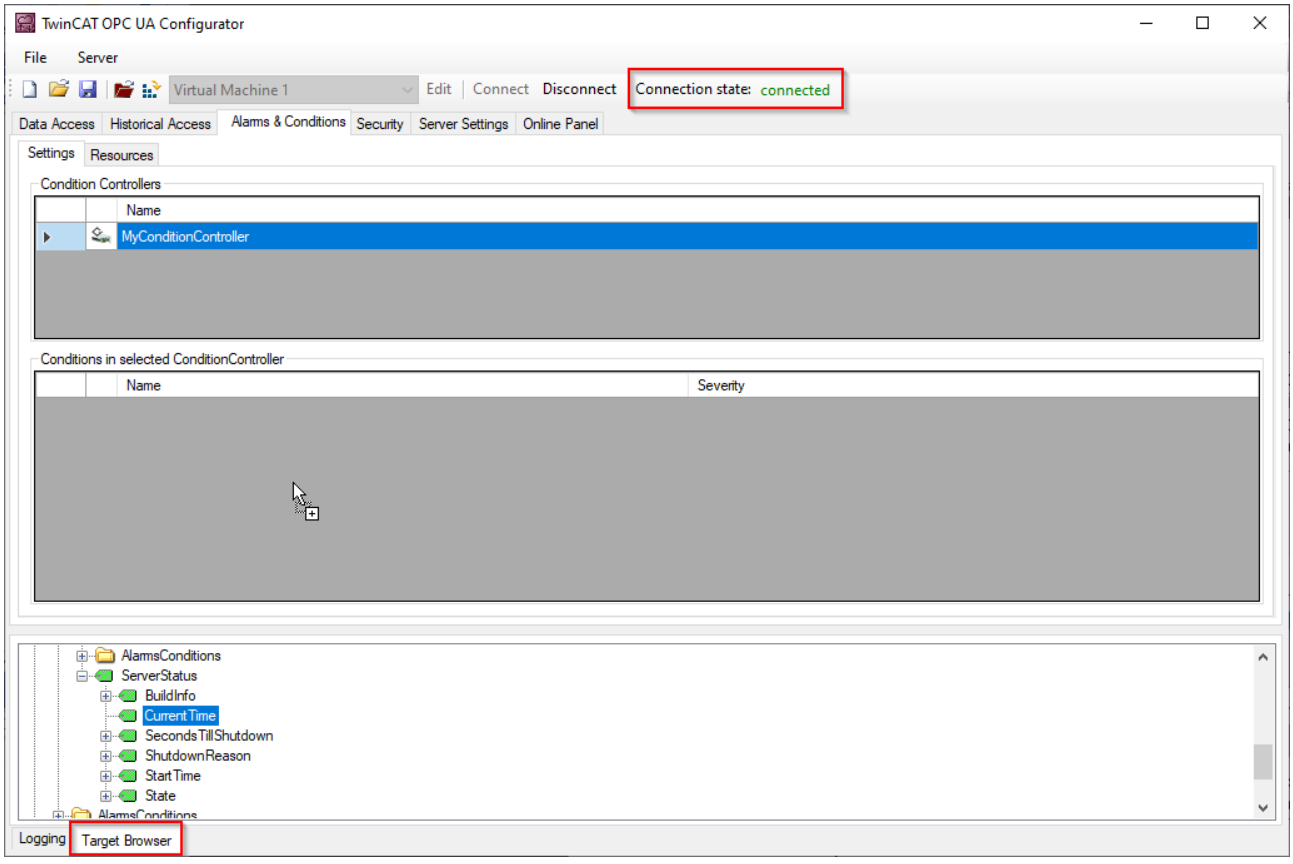


4.4.7 Configuring Alarms and Conditions

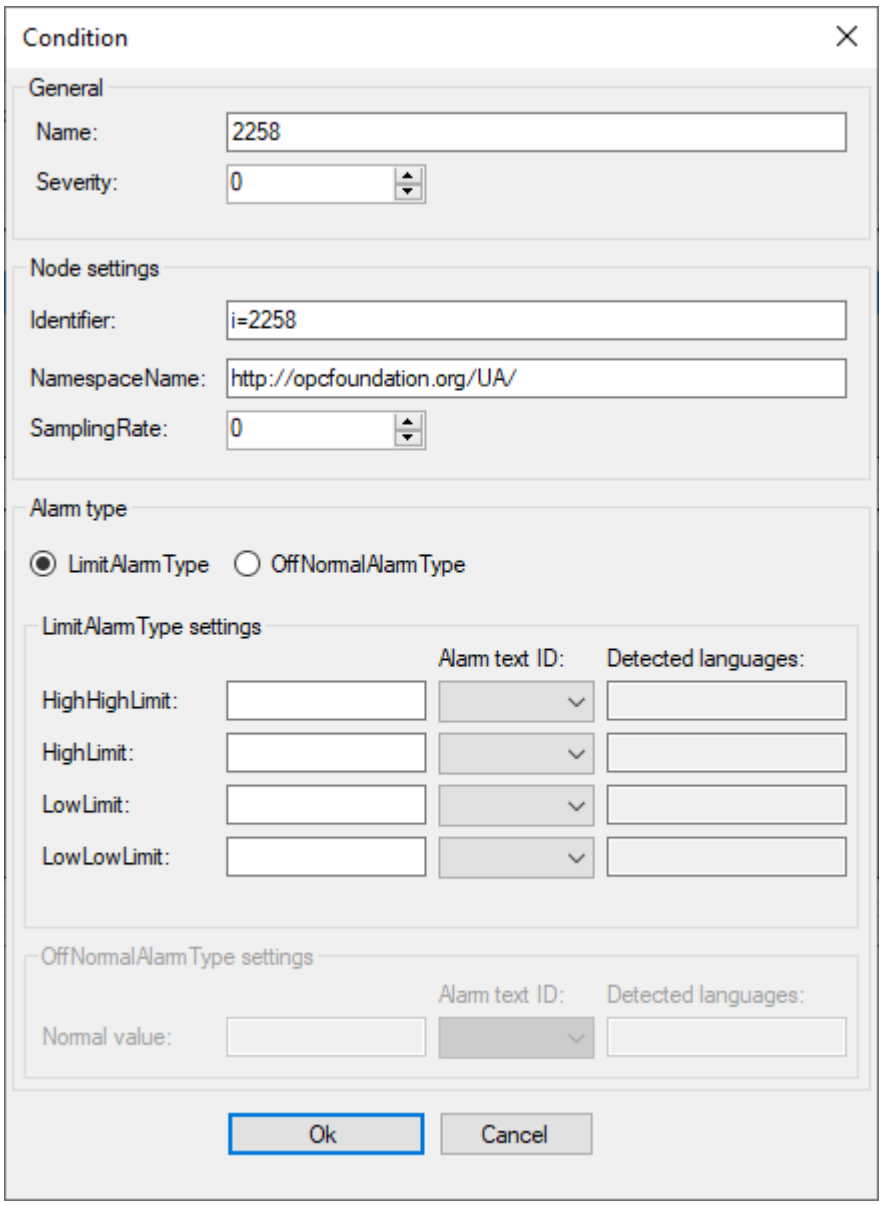
Use the **Alarms & Conditions** tab to configure both the **Condition Controller** and the **Conditions**. A **Condition Controller** is a management unit for organizing the individual **Conditions**. A **Condition** on the other hand reflects a variable which is to be monitored in the sense of **Alarms & Conditions** on the basis of configurable threshold values.



You can use the context menu to create both **Condition Controller** and **Conditions**. If you are connected to a TwinCAT OPC UA Server, you can also conveniently add the nodes to be configured to the **Conditions** via drag and drop from the **Target Browser**.



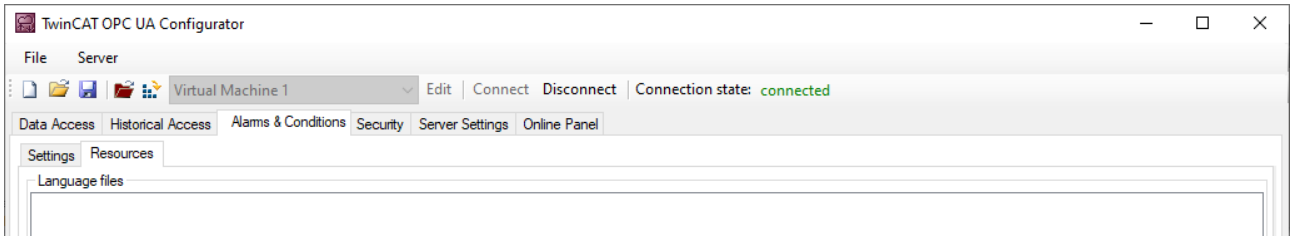
A **Condition** is always added to the currently selected **Condition Controller**. When using drag and drop, the configuration dialog of a **Condition** opens automatically.



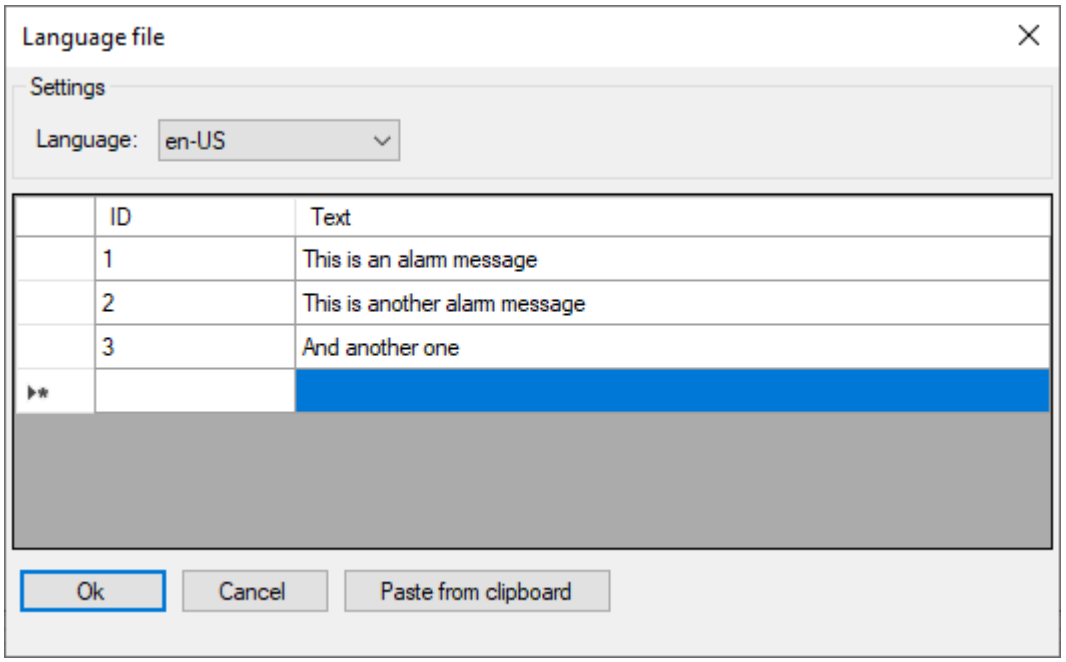
The alarm texts to be configured when selecting the respective **AlarmType** can be selected via the corresponding drop-down boxes. Please note that the alarm texts must already be available. Read the chapter Configuring alarm texts [▶ 53] to learn more about this topic.

4.4.8 Configuring alarm texts

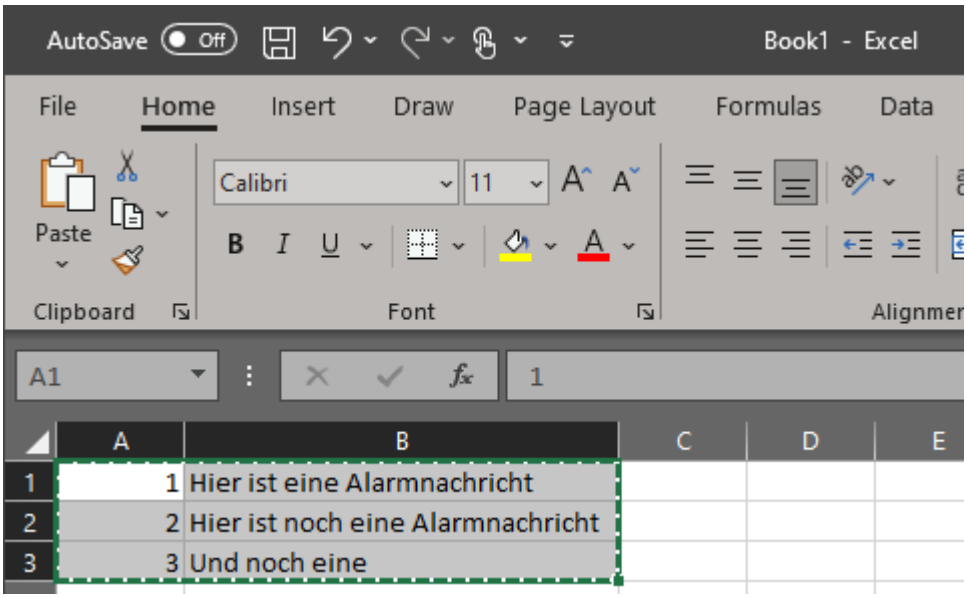
Within the **Alarms & Conditions** area, you can configure alarm texts via the **Resources** tab, which you can then use for a Condition.

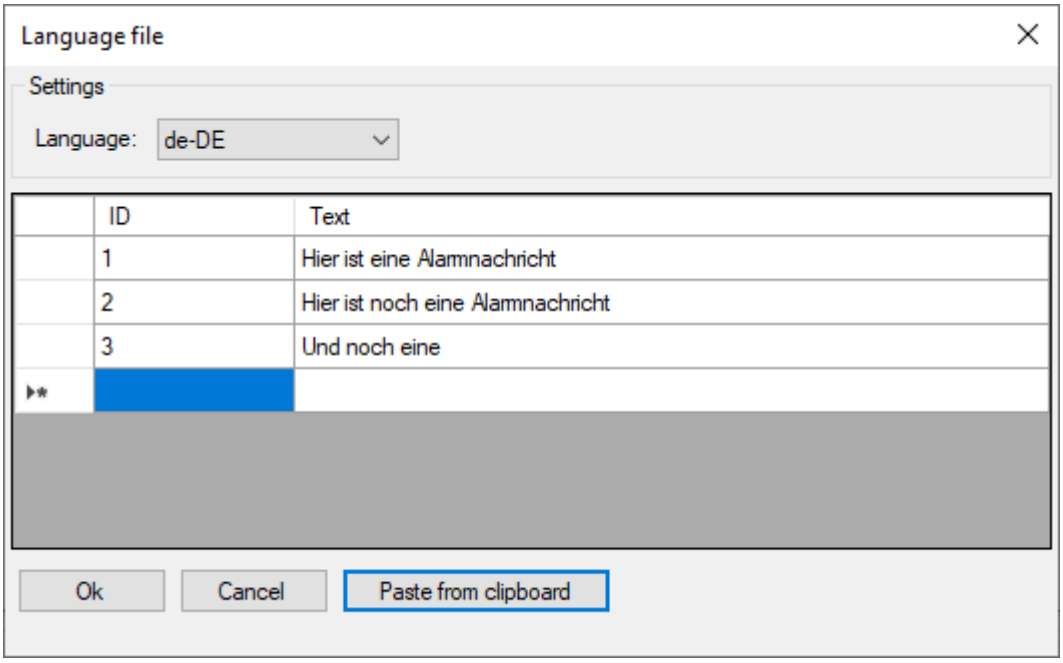


1. You can add a new alarm text file via the context menu. These files are grouped according to the language for which the alarm texts are defined.

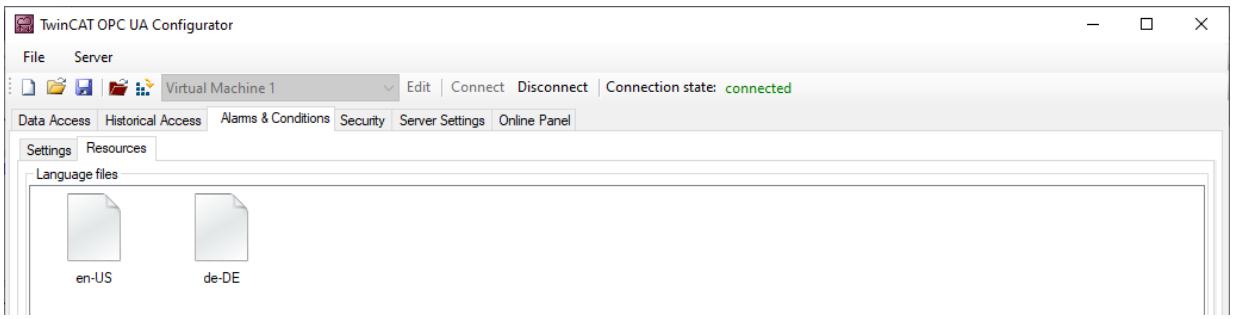


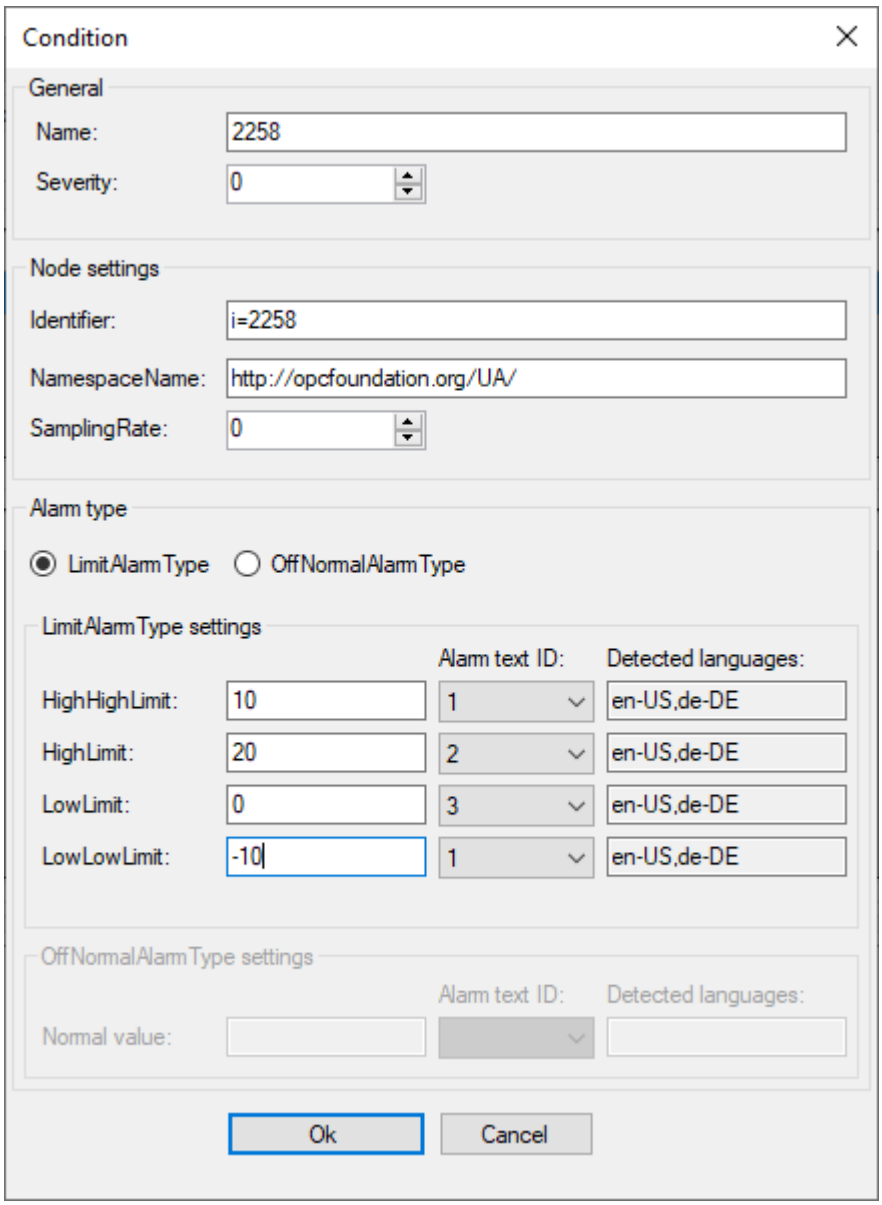
2. The **Paste from clipboard** button can be used to copy ID and text from an Excel spreadsheet by first copying them to the clipboard (CTRL+C) and then importing them via the button.





⇨ After configuring the language files, you can use the alarm texts on a Condition.





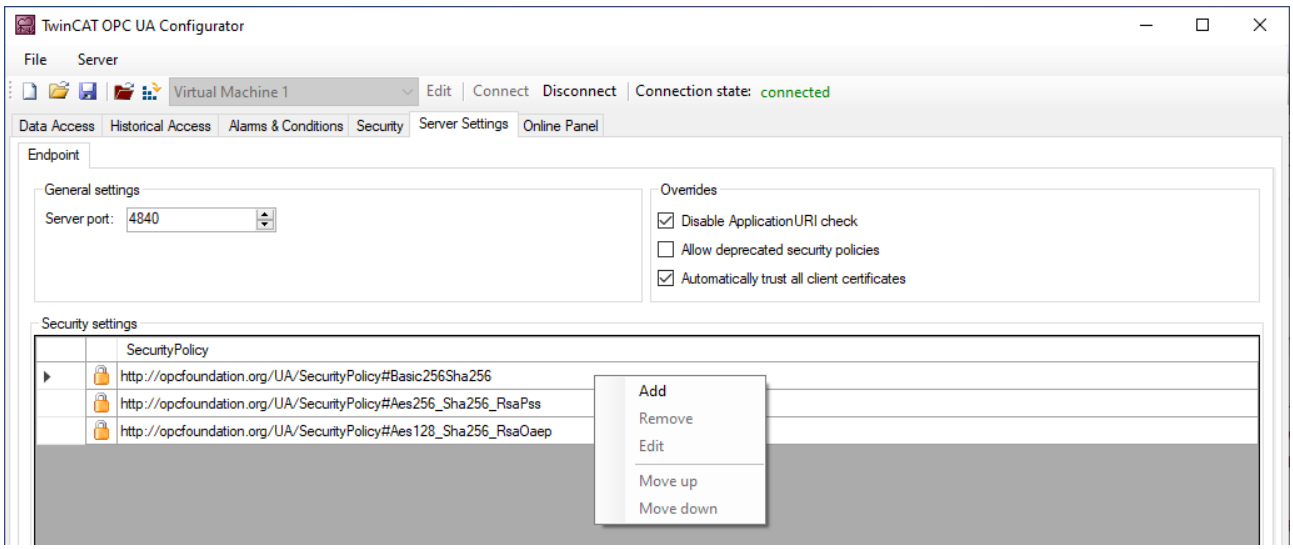
Using the **Detected languages** fields, you can quickly check whether you have defined the selected AlarmtextID for all languages, or whether a language may have been forgotten.

4.4.9 Configuring endpoints

The endpoints of the OPC UA Server indicate which security mechanisms are to be used during the connection establishment of a client. These range from "unencrypted" to "encrypted and signed", based on different key strengths.

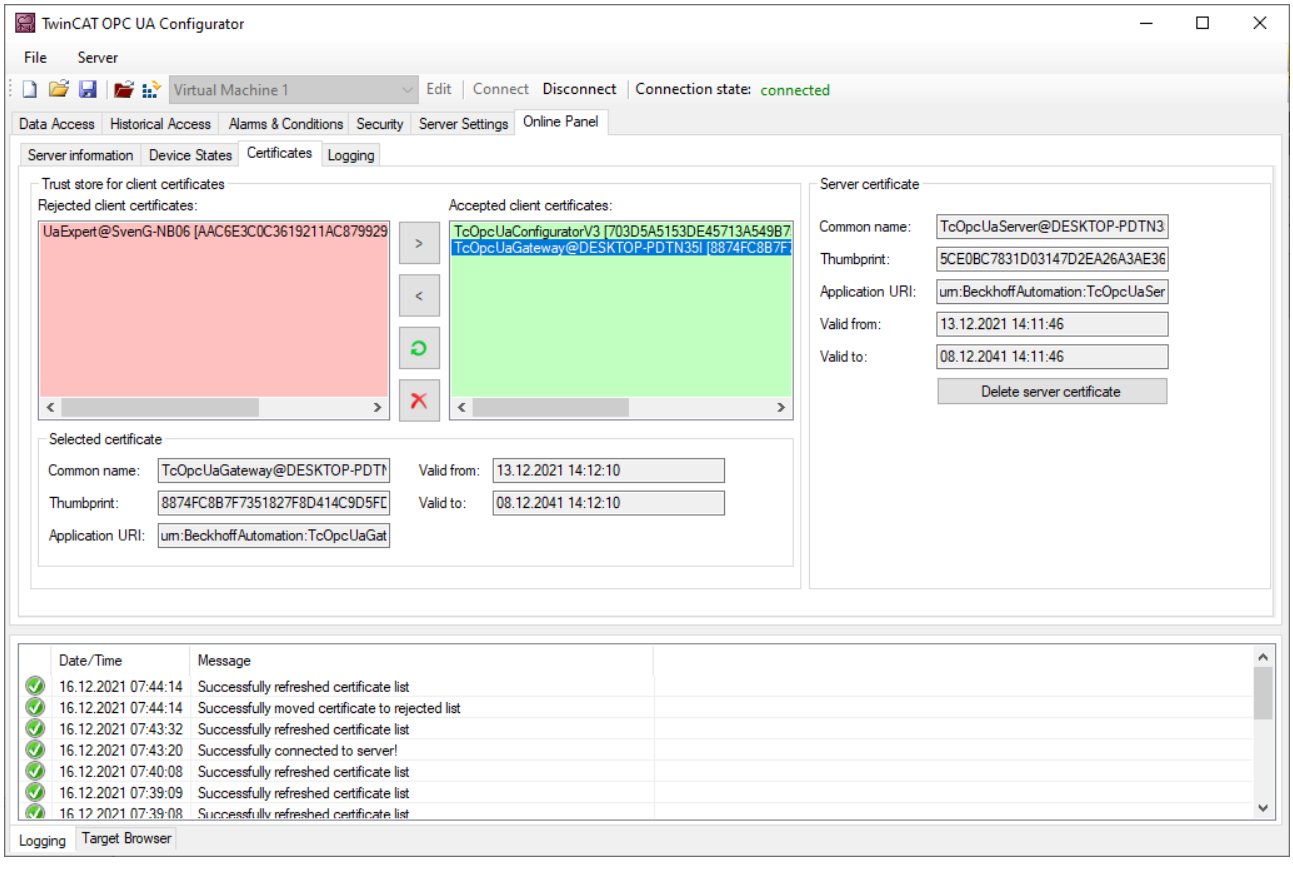
The endpoints can be activated and deactivated using the configurator. It may be useful to deactivate the unencrypted endpoint so that all clients can only connect themselves with valid certificates that are classified as trustworthy.

The **Server Settings** tab allows you to configure the endpoints, as well as some additional parameters. The context menu can be used to add or remove endpoints from the configuration.



4.4.10 Trust relationship for certificates

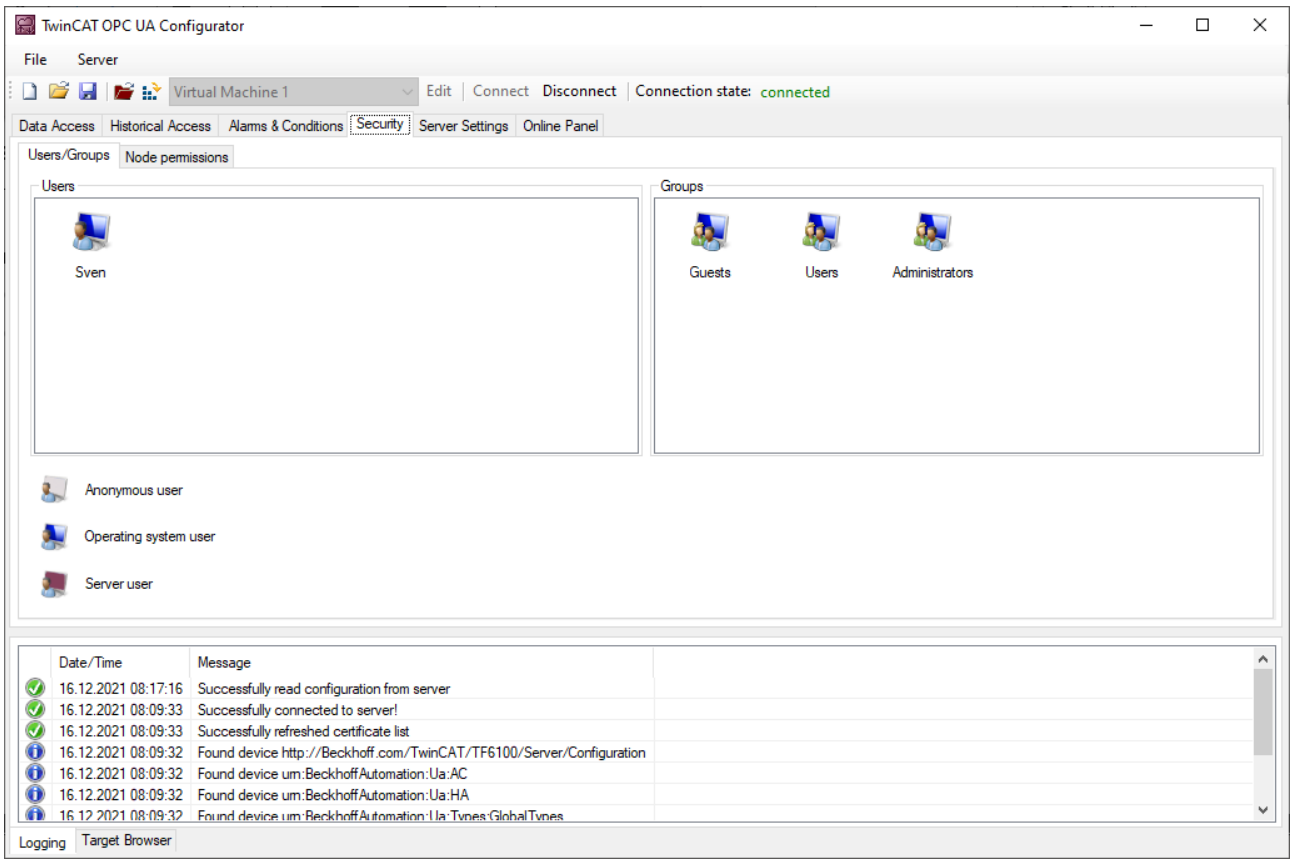
The trust relationships for client certificates on the TwinCAT OPC UA Server can be configured via the **Online Panel** tab and the **Certificates** section there. By selecting a client certificate in the respective TrustStore (Rejected/Accepted), certificate details can be displayed and moved between the TrustStores.



4.4.11 Configuring security settings

Security settings can be made on the server via the **Security** tab. These security settings may include the following items:

- Users and groups
- Access rights for groups to namespaces
- Access rights for groups to individual nodes



Users and groups

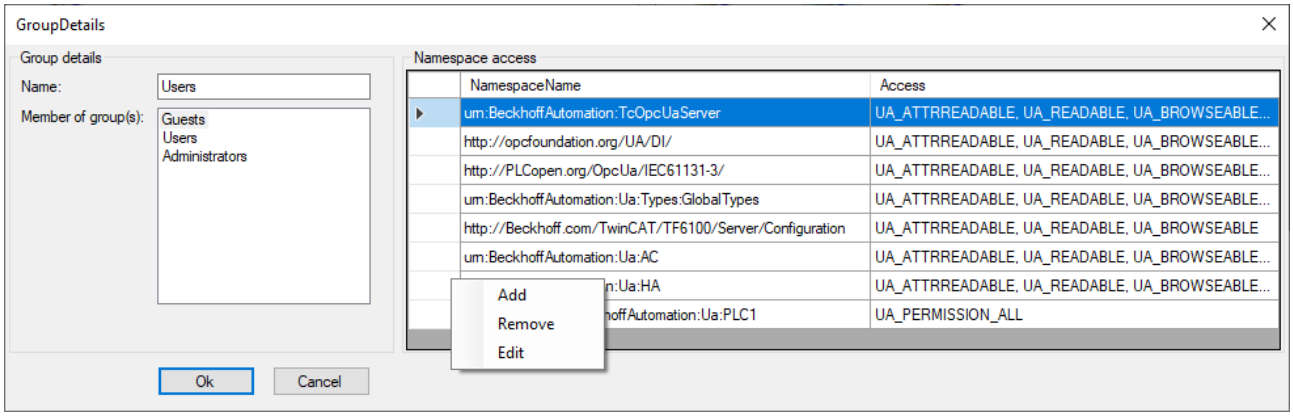
To configure access rights, users and user groups must first be created. Some groups are already predefined when the server is delivered. New users or groups can be added to the configuration via the context menu.

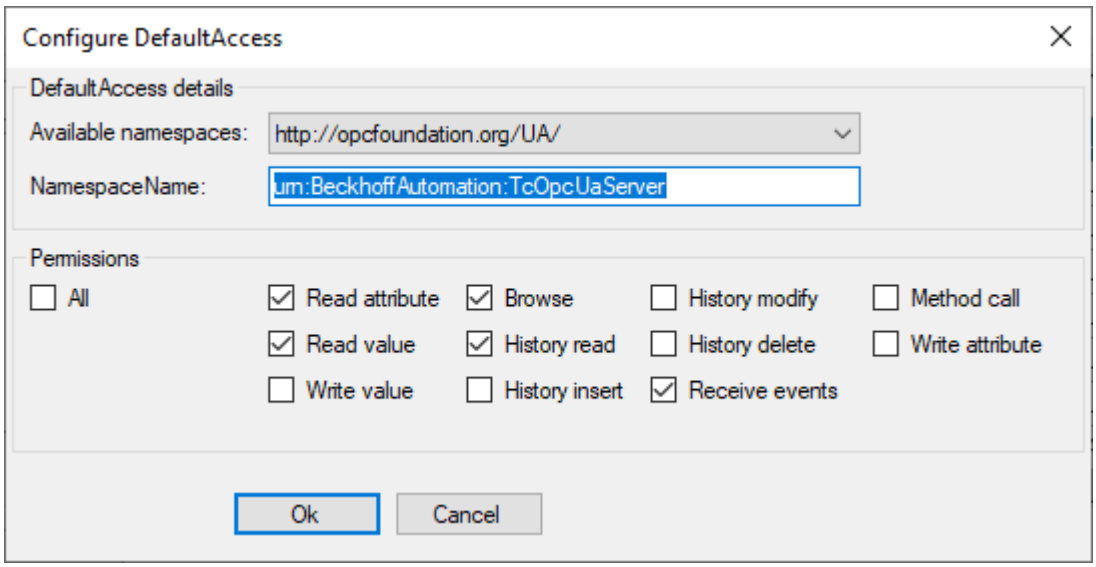
A user can be either the anonymous user, an operating system user or a server user. In any case, we recommend the configuration of operating system users.

A user group can have a so-called **default access** configured. These are access rights to a specific namespace.

Access rights to namespaces

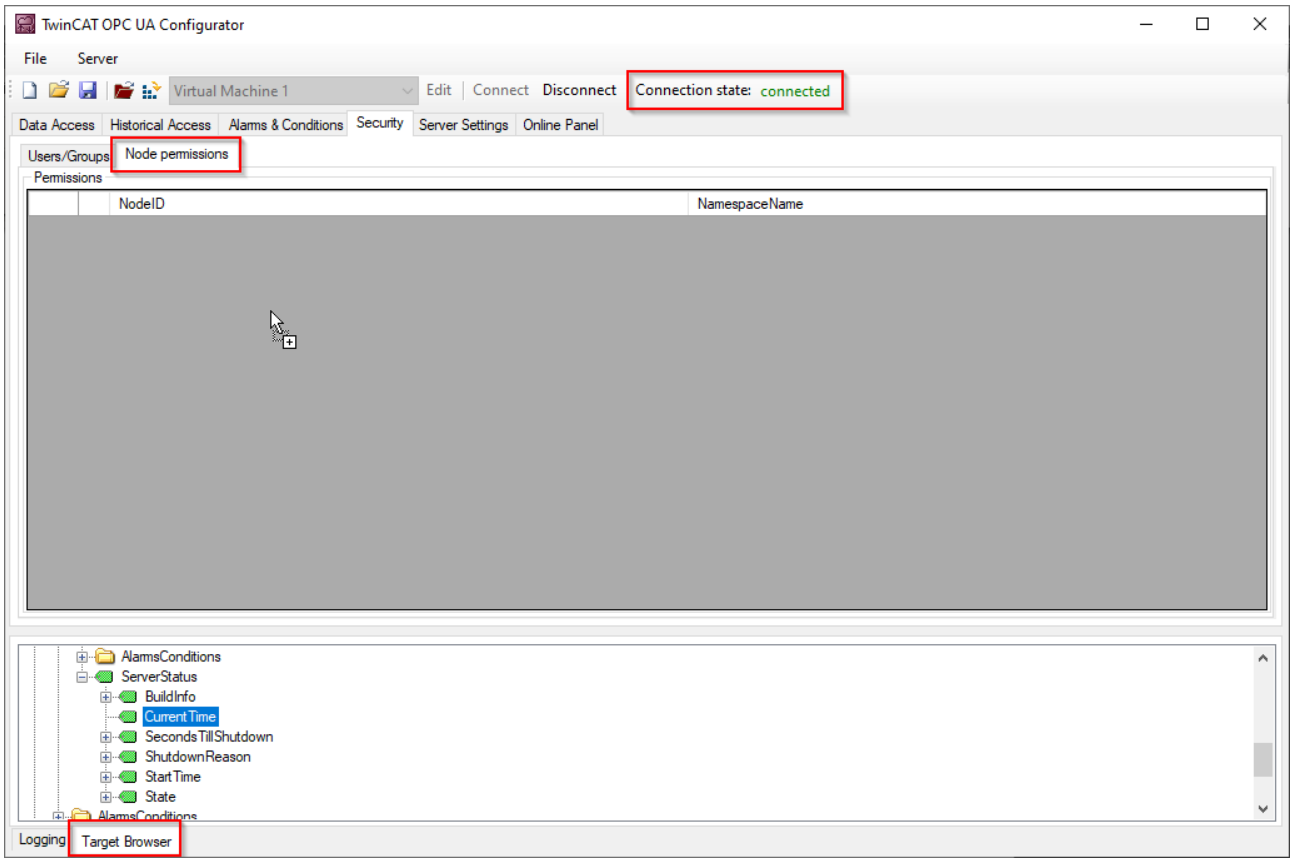
Access rights to certain namespaces can be defined at a user group. In the settings of the group there is a corresponding configuration area, which can be edited via the context menu.



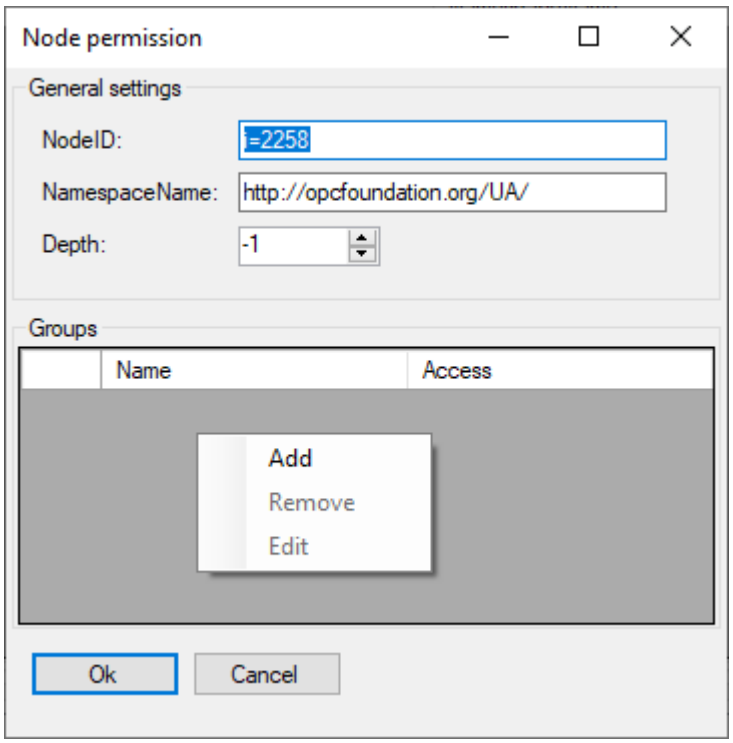


Access rights to individual nodes

The **Node permissions** tab can be used to define access rights to individual nodes and their child elements. You can configure the nodes manually via the context menu or conveniently add them to the configuration by dragging and dropping them from the **Target Browser**, provided you are connected to a server.



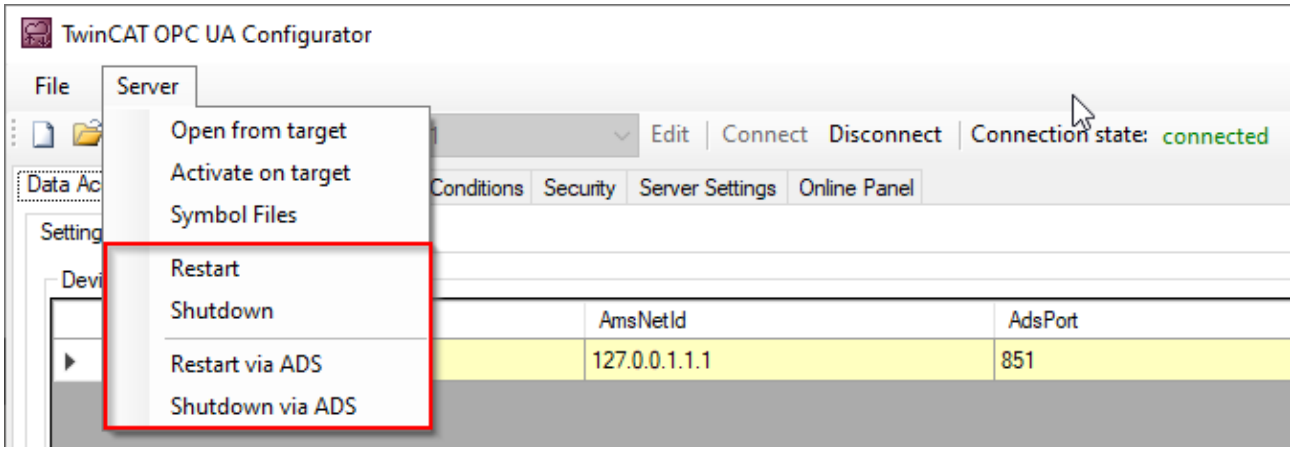
The user groups and access rights of the respective group can then be defined in the node configuration dialog.



You can use the parameter **Depth** to set whether the permissions should be inherited by child elements. The value "-1" indicates that all child elements should inherit the permissions.

4.4.12 Restarting the server

A TwinCAT OPC UA Server can be restarted via the **Server** menu. Usually you want to restart the server that is just connected via OPC UA. Alternatively, you can trigger the restart via ADS if you have established an ADS route to the server system.



4.4.13 Logging

For an advanced diagnostics you can activate the logging function of the OPC UA Server.

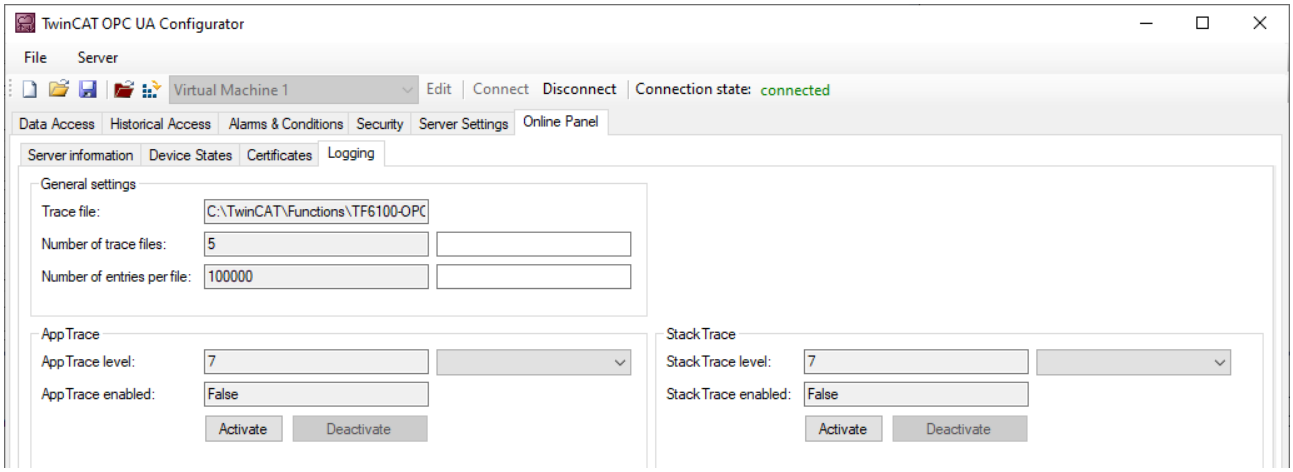
Writing the log file

Activating the logging function on the server causes a log file to be written on the file system. Make sure that there is sufficient storage space available and set the logging parameters accordingly (number of log files, size per log file).

Performance and timing behavior

Activation of the logging function will change the timing behavior of the OPC UA Server. As a result there may be losses of speed, depending on the platform and project.

The server logging functions can be activated via the **Online Panel** tab and the **Logging** section there.



Trace Level

In general, the higher the trace level, the more detailed (and more) data is written, but the more load is also placed on the server application, which changes the timing behavior accordingly. Please therefore only activate logging in the event of diagnostics and in consultation with Beckhoff Support.

Activate App Trace

In most cases it is sufficient to create a so-called "AppTrace". This logs information from the server application. To activate the AppTrace, please enter the number of TraceFiles and the number of entries per TraceFile in the corresponding text fields. Then select a trace level and click the button to activate the AppTrace. The values in the gray text boxes represent the current settings on the server.

Activate Stack Trace

In a few cases it is also necessary to create a so-called "StackTrace", whereby information from the OPC UA stack is logged. To activate the StackTrace please enter the number of TraceFiles as well as the number of entries per TraceFile into the corresponding text boxes. Then select a trace level and click on the button to activate the StackTrace. The values in the gray text boxes represent the current settings on the server.

5 Appendix

5.1 ADS Return Codes

Grouping of error codes:
Global error codes: 0x000 [▶ 62]... (0x9811_0000 ...)
Router error codes: 0x500 [▶ 62]... (0x9811_0500 ...)
General ADS errors: 0x700 [▶ 63]... (0x9811_0700 ...)
RTime error codes: 0x1000 [▶ 65]... (0x9811_1000 ...)

Global error codes

Hex	Dec	HRESULT	Name	Description
0x0	0	0x98110000	ERR_NOERROR	No error.
0x1	1	0x98110001	ERR_INTERNAL	Internal error.
0x2	2	0x98110002	ERR_NORTIME	No real time.
0x3	3	0x98110003	ERR_ALLOCLOCKEDMEM	Allocation locked – memory error.
0x4	4	0x98110004	ERR_INSERTMAILBOX	Mailbox full – the ADS message could not be sent. Reducing the number of ADS messages per cycle will help.
0x5	5	0x98110005	ERR_WRONGRECEIVEHMSG	Wrong HMSG.
0x6	6	0x98110006	ERR_TARGETPORTNOTFOUND	Target port not found – ADS server is not started, not reachable or not installed.
0x7	7	0x98110007	ERR_TARGETMACHINENOTFOUND	Target computer not found – AMS route was not found.
0x8	8	0x98110008	ERR_UNKNOWNCMDID	Unknown command ID.
0x9	9	0x98110009	ERR_BADTASKID	Invalid task ID.
0xA	10	0x9811000A	ERR_NOIO	No IO.
0xB	11	0x9811000B	ERR_UNKNOWNAMSCMD	Unknown AMS command.
0xC	12	0x9811000C	ERR_WIN32ERROR	Win32 error.
0xD	13	0x9811000D	ERR_PORTNOTCONNECTED	Port not connected.
0xE	14	0x9811000E	ERR_INVALIDAMSLENGTH	Invalid AMS length.
0xF	15	0x9811000F	ERR_INVALIDAMSNETID	Invalid AMS Net ID.
0x10	16	0x98110010	ERR_LOWINSTLEVEL	Installation level is too low –TwinCAT 2 license error.
0x11	17	0x98110011	ERR_NODEBUGINTAVAILABLE	No debugging available.
0x12	18	0x98110012	ERR_PORTDISABLED	Port disabled – TwinCAT system service not started.
0x13	19	0x98110013	ERR_PORTALREADYCONNECTED	Port already connected.
0x14	20	0x98110014	ERR_AMSSYNC_W32ERROR	AMS Sync Win32 error.
0x15	21	0x98110015	ERR_AMSSYNC_TIMEOUT	AMS Sync Timeout.
0x16	22	0x98110016	ERR_AMSSYNC_AMSERROR	AMS Sync error.
0x17	23	0x98110017	ERR_AMSSYNC_NOINDEXINMAP	No index map for AMS Sync available.
0x18	24	0x98110018	ERR_INVALIDAMSPORT	Invalid AMS port.
0x19	25	0x98110019	ERR_NOMEMORY	No memory.
0x1A	26	0x9811001A	ERR_TCPSEND	TCP send error.
0x1B	27	0x9811001B	ERR_HOSTUNREACHABLE	Host unreachable.
0x1C	28	0x9811001C	ERR_INVALIDAMSFRAGMENT	Invalid AMS fragment.
0x1D	29	0x9811001D	ERR_TLSSEND	TLS send error – secure ADS connection failed.
0x1E	30	0x9811001E	ERR_ACCESSDENIED	Access denied – secure ADS access denied.

Router error codes

Hex	Dec	HRESULT	Name	Description
0x500	1280	0x98110500	ROUTERERR_NOLOCKEDMEMORY	Locked memory cannot be allocated.
0x501	1281	0x98110501	ROUTERERR_RESIZEMEMORY	The router memory size could not be changed.
0x502	1282	0x98110502	ROUTERERR_MAILBOXFULL	The mailbox has reached the maximum number of possible messages.
0x503	1283	0x98110503	ROUTERERR_DEBUGBOXFULL	The Debug mailbox has reached the maximum number of possible messages.
0x504	1284	0x98110504	ROUTERERR_UNKNOWNPORTTYPE	The port type is unknown.
0x505	1285	0x98110505	ROUTERERR_NOTINITIALIZED	The router is not initialized.
0x506	1286	0x98110506	ROUTERERR_PORTALREADYINUSE	The port number is already assigned.
0x507	1287	0x98110507	ROUTERERR_NOTREGISTERED	The port is not registered.
0x508	1288	0x98110508	ROUTERERR_NOMOREQUEUES	The maximum number of ports has been reached.
0x509	1289	0x98110509	ROUTERERR_INVALIDPORT	The port is invalid.
0x50A	1290	0x9811050A	ROUTERERR_NOTACTIVATED	The router is not active.
0x50B	1291	0x9811050B	ROUTERERR_FRAGMENTBOXFULL	The mailbox has reached the maximum number for fragmented messages.
0x50C	1292	0x9811050C	ROUTERERR_FRAGMENTTIMEOUT	A fragment timeout has occurred.
0x50D	1293	0x9811050D	ROUTERERR_TOBEREMOVED	The port is removed.

General ADS error codes

Hex	Dec	HRESULT	Name	Description
0x700	1792	0x98110700	ADSERR_DEVICE_ERROR	General device error.
0x701	1793	0x98110701	ADSERR_DEVICE_SRVNOTSUPP	Service is not supported by the server.
0x702	1794	0x98110702	ADSERR_DEVICE_INVALIDGRP	Invalid index group.
0x703	1795	0x98110703	ADSERR_DEVICE_INVALIDOFFSET	Invalid index offset.
0x704	1796	0x98110704	ADSERR_DEVICE_INVALIDACCESS	Reading or writing not permitted. Several causes are possible. For example, an incorrect password was entered when creating routes.
0x705	1797	0x98110705	ADSERR_DEVICE_INVALIDSIZE	Parameter size not correct.
0x706	1798	0x98110706	ADSERR_DEVICE_INVALIDDATA	Invalid data values.
0x707	1799	0x98110707	ADSERR_DEVICE_NOTREADY	Device is not ready to operate.
0x708	1800	0x98110708	ADSERR_DEVICE_BUSY	Device is busy.
0x709	1801	0x98110709	ADSERR_DEVICE_INVALIDCONTEXT	Invalid operating system context. This can result from use of ADS blocks in different tasks. It may be possible to resolve this through multitasking synchronization in the PLC.
0x70A	1802	0x9811070A	ADSERR_DEVICE_NOMEMORY	Insufficient memory.
0x70B	1803	0x9811070B	ADSERR_DEVICE_INVALIDPARM	Invalid parameter values.
0x70C	1804	0x9811070C	ADSERR_DEVICE_NOTFOUND	Not found (files, ...).
0x70D	1805	0x9811070D	ADSERR_DEVICE_SYNTAX	Syntax error in file or command.
0x70E	1806	0x9811070E	ADSERR_DEVICE_INCOMPATIBLE	Objects do not match.
0x70F	1807	0x9811070F	ADSERR_DEVICE_EXISTS	Object already exists.
0x710	1808	0x98110710	ADSERR_DEVICE_SYMBOLNOTFOUND	Symbol not found.
0x711	1809	0x98110711	ADSERR_DEVICE_SYMBOLVERSIONINVALID	Invalid symbol version. This can occur due to an online change. Create a new handle.
0x712	1810	0x98110712	ADSERR_DEVICE_INVALIDSTATE	Device (server) is in invalid state.
0x713	1811	0x98110713	ADSERR_DEVICE_TRANSMODENOTSUPP	AdsTransMode not supported.
0x714	1812	0x98110714	ADSERR_DEVICE_NOTIFYHNDINVALID	Notification handle is invalid.
0x715	1813	0x98110715	ADSERR_DEVICE_CLIENTUNKNOWN	Notification client not registered.
0x716	1814	0x98110716	ADSERR_DEVICE_NOMOREHDLS	No further handle available.
0x717	1815	0x98110717	ADSERR_DEVICE_INVALIDWATCHSIZE	Notification size too large.
0x718	1816	0x98110718	ADSERR_DEVICE_NOTINIT	Device not initialized.
0x719	1817	0x98110719	ADSERR_DEVICE_TIMEOUT	Device has a timeout.
0x71A	1818	0x9811071A	ADSERR_DEVICE_NOINTERFACE	Interface query failed.
0x71B	1819	0x9811071B	ADSERR_DEVICE_INVALIDINTERFACE	Wrong interface requested.
0x71C	1820	0x9811071C	ADSERR_DEVICE_INVALIDCLSID	Class ID is invalid.
0x71D	1821	0x9811071D	ADSERR_DEVICE_INVALIDOBJID	Object ID is invalid.
0x71E	1822	0x9811071E	ADSERR_DEVICE_PENDING	Request pending.
0x71F	1823	0x9811071F	ADSERR_DEVICE_ABORTED	Request is aborted.
0x720	1824	0x98110720	ADSERR_DEVICE_WARNING	Signal warning.
0x721	1825	0x98110721	ADSERR_DEVICE_INVALIDARRAYIDX	Invalid array index.
0x722	1826	0x98110722	ADSERR_DEVICE_SYMBOLNOTACTIVE	Symbol not active.
0x723	1827	0x98110723	ADSERR_DEVICE_ACCESSDENIED	Access denied. Several causes are possible. For example, a unidirectional ADS route is used in the opposite direction.
0x724	1828	0x98110724	ADSERR_DEVICE_LICENSENOTFOUND	Missing license.
0x725	1829	0x98110725	ADSERR_DEVICE_LICENSEEXPIRED	License expired.
0x726	1830	0x98110726	ADSERR_DEVICE_LICENSEEXCEEDED	License exceeded.
0x727	1831	0x98110727	ADSERR_DEVICE_LICENSEINVALID	Invalid license.
0x728	1832	0x98110728	ADSERR_DEVICE_LICENSESYSTEMID	License problem: System ID is invalid.
0x729	1833	0x98110729	ADSERR_DEVICE_LICENSENOTIMELIMIT	License not limited in time.
0x72A	1834	0x9811072A	ADSERR_DEVICE_LICENSEFUTUREISSUE	Licensing problem: time in the future.
0x72B	1835	0x9811072B	ADSERR_DEVICE_LICENSETIMETOLONG	License period too long.
0x72C	1836	0x9811072C	ADSERR_DEVICE_EXCEPTION	Exception at system startup.
0x72D	1837	0x9811072D	ADSERR_DEVICE_LICENSEDUPLICATED	License file read twice.
0x72E	1838	0x9811072E	ADSERR_DEVICE_SIGNATUREINVALID	Invalid signature.
0x72F	1839	0x9811072F	ADSERR_DEVICE_CERTIFICATEINVALID	Invalid certificate.
0x730	1840	0x98110730	ADSERR_DEVICE_LICENSEOEMNOTFOUND	Public key not known from OEM.
0x731	1841	0x98110731	ADSERR_DEVICE_LICENSERESTRICTED	License not valid for this system ID.

Hex	Dec	HRESULT	Name	Description
0x732	1842	0x98110732	ADSERR_DEVICE_LICENSEDEMODENIED	Demo license prohibited.
0x733	1843	0x98110733	ADSERR_DEVICE_INVALIDFNCID	Invalid function ID.
0x734	1844	0x98110734	ADSERR_DEVICE_OUTOFRANGE	Outside the valid range.
0x735	1845	0x98110735	ADSERR_DEVICE_INVALIDALIGNMENT	Invalid alignment.
0x736	1846	0x98110736	ADSERR_DEVICE_LICENSEPLATFORM	Invalid platform level.
0x737	1847	0x98110737	ADSERR_DEVICE_FORWARD_PL	Context – forward to passive level.
0x738	1848	0x98110738	ADSERR_DEVICE_FORWARD_DL	Context – forward to dispatch level.
0x739	1849	0x98110739	ADSERR_DEVICE_FORWARD_RT	Context – forward to real-time.
0x740	1856	0x98110740	ADSERR_CLIENT_ERROR	Client error.
0x741	1857	0x98110741	ADSERR_CLIENT_INVALIDPARM	Service contains an invalid parameter.
0x742	1858	0x98110742	ADSERR_CLIENT_LISTEMPTY	Polling list is empty.
0x743	1859	0x98110743	ADSERR_CLIENT_VARUSED	Var connection already in use.
0x744	1860	0x98110744	ADSERR_CLIENT_DUPLINVOKEID	The called ID is already in use.
0x745	1861	0x98110745	ADSERR_CLIENT_SYNCTIMEOUT	Timeout has occurred – the remote terminal is not responding in the specified ADS timeout. The route setting of the remote terminal may be configured incorrectly.
0x746	1862	0x98110746	ADSERR_CLIENT_W32ERROR	Error in Win32 subsystem.
0x747	1863	0x98110747	ADSERR_CLIENT_TIMEOUTINVALID	Invalid client timeout value.
0x748	1864	0x98110748	ADSERR_CLIENT_PORTNOTOPEN	Port not open.
0x749	1865	0x98110749	ADSERR_CLIENT_NOAMSADDR	No AMS address.
0x750	1872	0x98110750	ADSERR_CLIENT_SYNCINTERNAL	Internal error in Ads sync.
0x751	1873	0x98110751	ADSERR_CLIENT_ADDHASH	Hash table overflow.
0x752	1874	0x98110752	ADSERR_CLIENT_REMOVEHASH	Key not found in the table.
0x753	1875	0x98110753	ADSERR_CLIENT_NOMORESYM	No symbols in the cache.
0x754	1876	0x98110754	ADSERR_CLIENT_SYNCRESINVALID	Invalid response received.
0x755	1877	0x98110755	ADSERR_CLIENT_SYNCPORTLOCKED	Sync Port is locked.
0x756	1878	0x98110756	ADSERR_CLIENT_REQUESTCANCELLED	The request was canceled.

RTime error codes

Hex	Dec	HRESULT	Name	Description
0x1000	4096	0x98111000	RTERR_INTERNAL	Internal error in the real-time system.
0x1001	4097	0x98111001	RTERR_BADTIMERPERIODS	Timer value is not valid.
0x1002	4098	0x98111002	RTERR_INVALIDTASKPTR	Task pointer has the invalid value 0 (zero).
0x1003	4099	0x98111003	RTERR_INVALIDSTACKPTR	Stack pointer has the invalid value 0 (zero).
0x1004	4100	0x98111004	RTERR_PRIOEXISTS	The request task priority is already assigned.
0x1005	4101	0x98111005	RTERR_NOMORETCB	No free TCB (Task Control Block) available. The maximum number of TCBs is 64.
0x1006	4102	0x98111006	RTERR_NOMORESEMAS	No free semaphores available. The maximum number of semaphores is 64.
0x1007	4103	0x98111007	RTERR_NOMOREQUEUES	No free space available in the queue. The maximum number of positions in the queue is 64.
0x100D	4109	0x9811100D	RTERR_EXTIRQALREADYDEF	An external synchronization interrupt is already applied.
0x100E	4110	0x9811100E	RTERR_EXTIRQNOTDEF	No external sync interrupt applied.
0x100F	4111	0x9811100F	RTERR_EXTIRQINSTALLFAILED	Application of the external synchronization interrupt has failed.
0x1010	4112	0x98111010	RTERR_IRQLNOTLESSOREQUAL	Call of a service function in the wrong context
0x1017	4119	0x98111017	RTERR_VMXNOTSUPPORTED	Intel VT-x extension is not supported.
0x1018	4120	0x98111018	RTERR_VMXDISABLED	Intel VT-x extension is not enabled in the BIOS.
0x1019	4121	0x98111019	RTERR_VMXCONTROLSMISSING	Missing function in Intel VT-x extension.
0x101A	4122	0x9811101A	RTERR_VMXENABLEFAILS	Activation of Intel VT-x fails.

Specific positive HRESULT Return Codes:

HRESULT	Name	Description
0x0000_0000	S_OK	No error.
0x0000_0001	S_FALSE	No error. Example: successful processing, but with a negative or incomplete result.
0x0000_0203	S_PENDING	No error. Example: successful processing, but no result is available yet.
0x0000_0256	S_WATCHDOG_TIMEOUT	No error. Example: successful processing, but a timeout occurred.

TCP Winsock error codes

Hex	Dec	Name	Description
0x274C	10060	WSAETIMEDOUT	A connection timeout has occurred - error while establishing the connection, because the remote terminal did not respond properly after a certain period of time, or the established connection could not be maintained because the connected host did not respond.
0x274D	10061	WSAECONNREFUSED	Connection refused - no connection could be established because the target computer has explicitly rejected it. This error usually results from an attempt to connect to a service that is inactive on the external host, that is, a service for which no server application is running.
0x2751	10065	WSAEHOSTUNREACH	No route to host - a socket operation referred to an unavailable host.
More Winsock error codes: Win32 error codes			

5.2 Support and Service

Beckhoff and their partners around the world offer comprehensive support and service, making available fast and competent assistance with all questions related to Beckhoff products and system solutions.

Download finder

Our download finder contains all the files that we offer you for downloading. You will find application reports, technical documentation, technical drawings, configuration files and much more.

The downloads are available in various formats.

Beckhoff's branch offices and representatives

Please contact your Beckhoff branch office or representative for local support and service on Beckhoff products!

The addresses of Beckhoff's branch offices and representatives round the world can be found on our internet page: www.beckhoff.com

You will also find further documentation for Beckhoff components there.

Beckhoff Support

Support offers you comprehensive technical assistance, helping you not only with the application of individual Beckhoff products, but also with other, wide-ranging services:

- support
- design, programming and commissioning of complex automation systems
- and extensive training program for Beckhoff system components

Hotline:	+49 5246 963-157
e-mail:	support@beckhoff.com

Beckhoff Service

The Beckhoff Service Center supports you in all matters of after-sales service:

- on-site service

- repair service
- spare parts service
- hotline service

Hotline:	+49 5246 963-460
e-mail:	service@beckhoff.com

Beckhoff Headquarters

Beckhoff Automation GmbH & Co. KG

Huelshorstweg 20
33415 Verl
Germany

Phone:	+49 5246 963-0
e-mail:	info@beckhoff.com
web:	www.beckhoff.com

Trademark statements

Beckhoff®, TwinCAT®, TwinCAT/BSD®, TC/BSD®, EtherCAT®, EtherCAT G®, EtherCAT G10®, EtherCAT P®, Safety over EtherCAT®, TwinSAFE®, XFC®, XTS® and XPlanar® are registered trademarks of and licensed by Beckhoff Automation GmbH.

Third-party trademark statements

Arm, Arm9 and Cortex are trademarks or registered trademarks of Arm Limited (or its subsidiaries or affiliates) in the US and/or elsewhere.

Intel, the Intel logo, Intel Core, Xeon, Intel Atom, Celeron and Pentium are trademarks of Intel Corporation or its subsidiaries.

Microsoft, Microsoft Azure, Microsoft Edge, PowerShell, Visual Studio, Windows and Xbox are trademarks of the Microsoft group of companies.

More Information:
www.beckhoff.com/TF6100

Beckhoff Automation GmbH & Co. KG
Hülshorstweg 20
33415 Verl
Germany
Phone: +49 5246 9630
info@beckhoff.com
www.beckhoff.com