



# Manual



Software Modules  
**MOVIKIT® OPC-UA,**  
**MOVIKIT® OPC-UA add-on SensorInterface**



## Table of Contents

<b>1</b>	<b>General information.....</b>	<b>5</b>
1.1	About this documentation .....	5
1.2	Content of the documentation.....	5
1.3	Structure of the warning notes .....	5
1.3.1	Meaning of signal words .....	5
1.3.2	Structure of section-related safety notes.....	5
1.3.3	Structure of embedded safety notes .....	6
1.4	Decimal separator in numerical values .....	6
1.5	Rights to claim under limited warranty .....	6
1.6	Product names and trademarks.....	7
1.7	Copyright notice .....	7
1.8	Applicable documentation.....	7
<b>2</b>	<b>Safety notes .....</b>	<b>8</b>
2.1	Preliminary information .....	8
2.2	Target group .....	8
2.3	IT security .....	8
2.4	Designated use .....	8
<b>3</b>	<b>MOVIKIT® OPC-UA.....</b>	<b>9</b>
3.1	Module description.....	9
3.1.1	Requirements .....	9
3.1.2	License type .....	9
3.1.3	Order information .....	10
3.2	Functions .....	10
3.3	Application examples .....	11
3.3.1	MOVI-C® CONTROLLER as a simple OPC UA server .....	11
3.3.2	MOVI-C® CONTROLLER as OPC UA server with OPC UA information model. 12	
3.3.3	MOVI-C® CONTROLLER or visualization as OPC UA client .....	13
<b>4</b>	<b>MOVIKIT® OPC UA add-on SensorInterface .....</b>	<b>14</b>
4.1	Module description.....	14
4.1.1	Requirements .....	14
4.1.2	License type .....	14
4.1.3	Order information .....	14
4.2	Functions .....	14
4.3	Advantages .....	15
<b>5</b>	<b>Startup .....</b>	<b>16</b>
5.1	General information .....	16
5.2	Software required for startup .....	16
5.3	Usecase: The simple OPC UA server without IT security.....	16
5.3.1	Settings .....	16
5.3.2	Testing the communication .....	18
5.4	Usecase: The OPC UA server with information model without IT security .....	20
5.4.1	Storing OPC UA information model .....	20

5.4.2	Transferring OPC UA declarations into the program .....	21
5.4.3	Testing communication .....	23
5.5	Use case: OPC UA server with certificates.....	24
5.6	Use case: OPC UA server with additional user management .....	26
5.7	MOVI-C CONTROLLER® or MOVI-C® visualization as OPC UA client .....	28
5.8	MOVIKIT® OPC UA add-on SensorInterface .....	29
<b>6</b>	<b>Further useful information.....</b>	<b>30</b>
6.1	Use of certificates .....	30
6.2	Security policies and performance.....	30
6.3	Changing OPC UA configuration at runtime .....	30

# 1 General information

## 1.1 About this documentation

This documentation is an integral part of the product. The documentation is intended for all employees who perform work on the product.

Make sure that this documentation is accessible and legible. Ensure that persons responsible for the systems and their operation as well as persons who work with the product independently have read through the documentation carefully and understood it. If you are unclear about any of the information in this documentation, or if you require further information, contact SEW-EURODRIVE.

## 1.2 Content of the documentation

The descriptions in this documentation refer to the software and firmware versions at the time of publication. These descriptions might differ if you install later software or firmware versions. In this case, contact SEW-EURODRIVE.

The latest edition of the documentation is also always available in [Online Support](#) on the website of SEW-EURODRIVE.

## 1.3 Structure of the warning notes

### 1.3.1 Meaning of signal words

The following table shows the grading and meaning of the signal words for safety notes:

Signal word	Meaning	Consequences if disregarded
<b>⚠ DANGER</b>	Imminent hazard	Severe or fatal injuries
<b>⚠ WARNING</b>	Possible dangerous situation	Severe or fatal injuries
<b>⚠ CAUTION</b>	Possible dangerous situation	Minor injuries
<b>NOTICE</b>	Possible damage to property	Damage to the product or its environment
<b>INFORMATION</b>	Useful information or tip: Simplifies handling of the product.	

### 1.3.2 Structure of section-related safety notes

Section-related safety notes do not apply to a specific action but to several actions pertaining to one subject. The hazard symbols used either indicate a general hazard or a specific hazard.

This is the formal structure of a safety note for a specific section:



#### SIGNAL WORD


Type and source of hazard.

Possible consequence(s) if disregarded.

- Measure(s) to prevent the hazard.

**Meaning of the hazard symbols**

The hazard symbols in the safety notes have the following meaning:

Hazard symbol	Meaning
	General hazard

**1.3.3 Structure of embedded safety notes**

Embedded safety notes are directly integrated into the instructions just before the description of the dangerous action.

This is the formal structure of an embedded safety note:

**⚠ SIGNAL WORD!** Type and source of hazard. Possible consequence(s) if disregarded. Measure(s) to prevent the hazard.

**1.4 Decimal separator in numerical values**

In this document, a period is used to indicate the decimal separator.

Example: 30.5 kg

**1.5 Rights to claim under limited warranty**

Read the information in this documentation. This is essential for fault-free operation and fulfillment of any rights to claim under limited warranty. Read the documentation before you start working with the product.

## 1.6 Product names and trademarks

The product names mentioned in this documentation are trademarks or registered trademarks of the respective titleholders.

## 1.7 Copyright notice

© 2023 SEW-EURODRIVE. All rights reserved. Copyright law prohibits the unauthorized reproduction, modification, distribution and use of this document – in whole or in part.

## 1.8 Applicable documentation

For all other components, refer to the corresponding documentation.

Always use the latest edition of the documentation and the software.

The SEW-EURODRIVE website ([www.sew-eurodrive.com](http://www.sew-eurodrive.com)) provides a wide selection of documents for download in various languages. If required, you can also order printed and bound copies of the documentation from SEW-EURODRIVE.



## 2 Safety notes

### 2.1 Preliminary information

The following general safety notes serve the purpose of preventing injury to persons and damage to property. They primarily apply to the use of products described in this documentation. If you use additional components, also observe the relevant warning and safety notes.

### 2.2 Target group

Specialist for working with software

Any work with the software may only be performed by a specialist with suitable training. A specialist in the context of this documentation is someone who has the following qualifications:

- Appropriate instruction
- Knowledge of this documentation and other applicable documentation
- SEW-EURODRIVE recommends additional training for products that are operated using this software.

### 2.3 IT security



If you need support with the configuration, contact SEW-EURODRIVE Service. You can obtain information about current security-related issues by e-mail or on the [Product Security Management website](#). There you will find various contact options for reporting security-related problems.

For detailed information on the IT security of the products used, refer to the respective documentation.

### 2.4 Designated use

The MOVIKIT® software modules described in this manual are software licenses for enabling communication according to the OPC UA specification based on the communication stack of the OPC Foundation.

Use the device-independent MOVISUITE® engineering software to enable communication with the IEC Editor and to apply it to a MOVI-C® CONTROLLER.



## **3 MOVIKIT® OPC-UA**

### **3.1 Module description**

The MOVIKIT® OPC-UA software module provides an OPC UA server and an OPC UA client on the MOVI-C® CONTROLLER of performance classes UHX25A, UHX45A, UHX65A, and UHX86A. Both the server and the client implement communication according to the OPC UA specification following the "Micro Embedded Device Server" profile based on the OPC Foundation communication stack. Communication takes place via the binary protocol and standard port 4840.

The OPC UA server publishes its data via the symbol configuration in the IEC Editor or via the information model in the communication manager. It can communicate with OPC UA clients in the MOVI-C® CONTROLLER from SEW-EURODRIVE as well as with devices from other manufacturers such as HMI panels, augmented reality solutions, or tablets for visualizations. The OPC UA client can communicate with other OPC UA servers via the data source manager.

In the simplest case, the client and server communicate unencrypted without certificates and with deactivated user management; logging in is anonymous. Encrypted communication with certificates and/or user management can also be used. Pub-Sub communication is not supported.

A MOVI-C® CONTROLLER can also be an OPC UA client for other servers and an OPC UA server for other clients.

#### **3.1.1 Requirements**

One of the following MOVI-C® CONTROLLER with the listed firmware version is required to use OPC UA. The scope of functions described in this manual refers to firmware version 8.0, regardless of the controller. For startup, at least the specified version of the MOVISUITE® engineering software is required. The function of OPC UA is only permanent if a valid purchase license is active. Switch the MOVI-C® CONTROLLER off and on to complete the activation of the purchase license over the runtime.

<b>MOVI-C® CONTROLLER</b>	<b>Firmware</b>	<b>MOVISUITE®</b>
UHX25A (standard)	from 7.0	V2.30
UHX45A (advanced)		
UHX65A (progressive)		
UHX86	from 8.0	V2.40

The OPC UA client is part of the MOVIKIT® Visualization software module as of version 1.3.4.0. The scope of functions described below refers to version 1.4.1.0.

#### **3.1.2 License type**

Performance license (performance-based license for a MOVI-C® CONTROLLER)

For more information on licensing, refer to the documents [MOVIKIT® Licensing](#) and [MOVI-C® Software Components](#). You can download the documents from the [SEW-EURODRIVE Website](#).

### 3.1.3 Order information

The following table shows the order data for MOVIKIT® OPC UA in relation to the various performance classes of the MOVI-C® CONTROLLER.

MOVI-C® CONTROLLER	Type designation	Part number
UHX25A (standard)	SMK1501-020	23373008
UHX45A (advanced)	SMK1501-040	23373016
UHX65A (progressive)	SMK1501-060	23372540
UHX86	SMK1501-080	23374683

## 3.2 Functions

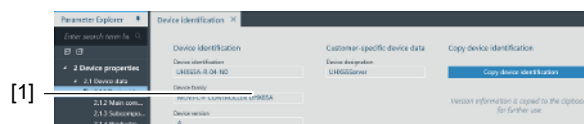
Overview of functions:

### Providing an OPC UA server

- Providing all data available and selected on the MOVI-C® CONTROLLER using the OPC UA communication protocol.
- Connecting a MOVI-C® CONTROLLER to a third-party component that is an OPC UA client.
- Connecting a third-party component with an OPC UA client to a MOVI-C® CONTROLLER.
- Provision of structures according to the OPC UA information model in function blocks via the communication manager. The use of methods is currently not supported.

### Assigning a name for the MOVI-C® CONTROLLER

Freely adjustable name of the MOVI-C® CONTROLLER [1], which is used in the symbolic designation of the node ID.



18014432319363979

### Providing an OPC UA client

Providing an OPC UA client for the MOVI-C® CONTROLLER via the data source manager or the MOVIKIT® Visualization software module. This means that, for example, the following functions are possible:

- Cross communication between two MOVI-C® CONTROLLER as an alternative to network variables.
- Connecting an actuator or sensor to the MOVI-C® CONTROLLER as an OPC UA server.
- Connecting a visualization (client) to a MOVI-C® CONTROLLER (server).

### Providing mechanisms for IT security

- Protection via certificates and encryption
- User and password can also be used as OPC UA server
- User group dependent enable of the variables in the symbol configuration

### Connecting the DUV40A sensor

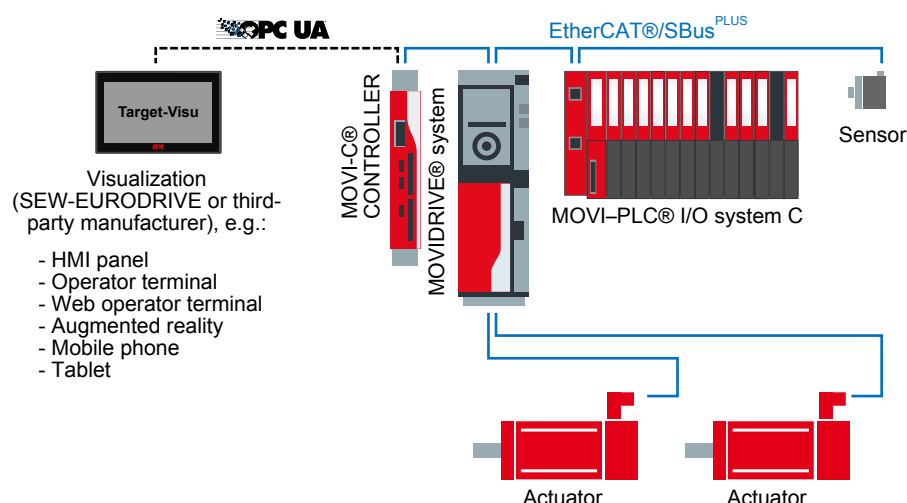
Use the "MOVIKIT® OPC UA add-on SensorInterface" (→ 14) software module to connect the DUV40A sensor from SEW-EURODRIVE.

## 3.3 Application examples

### 3.3.1 MOVI-C® CONTROLLER as a simple OPC UA server

The following figure shows an example implementation of a visualization (client) to a MOVI-C® CONTROLLER (server) using OPC UA.

#### OPC UA topology



63050425198757003

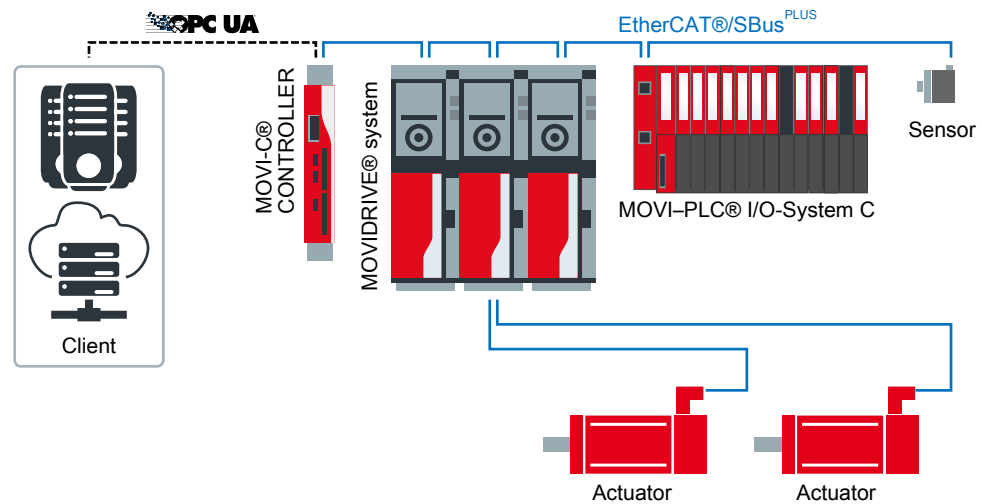
The HMI panel acts as the interface to the system/machine operator. Behind the user interface there is usually extensive code with corresponding programming effort that should not be lost by using a MOVI-C® CONTROLLER. In case of a replacement, this code can be retained as long as the HMI panel has an OPC UA client. This significantly reduces the effort required to change from any controller to a MOVI-C® CONTROLLER.

When using the MOVIKIT® OPC UA software module, you do not need to insert variables in a fieldbus telegram to transfer them to a higher-level controller and then send them to the visualization. Symbolic access is easy to handle, as is the architecture through a direct connection between the MOVI-C® CONTROLLER and the visualization.

The MOVIKIT® OPC-UA software module provides access to all sensor and actuator data provided in the IEC part of the MOVI-C® CONTROLLER. This allows numerous products that use the OPC UA standard to communicate with each other.

### 3.3.2 MOVI-C® CONTROLLER as OPC UA server with OPC UA information model

The following figure shows an example of the implementation of a visualization and control of an external software application (client) to a MOVI-C® CONTROLLER (server) using OPC UA.

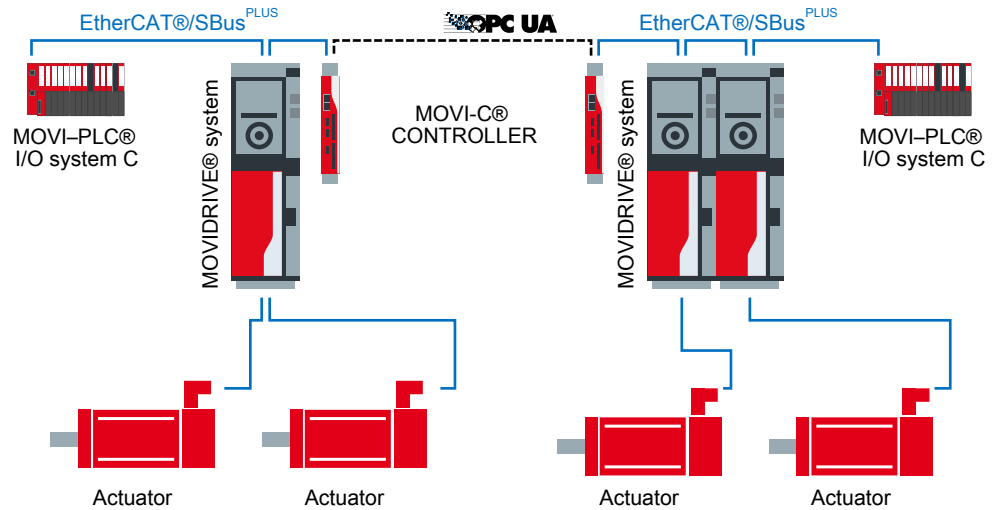


18014435110002571

The application on the MOVI-C® CONTROLLER is controlled by an external client in a standardized manner according to the OPC UA information model. This means that, for example, an application on a PC or another controller within the application can directly access and trigger actions. Methods are not yet supported.

### 3.3.3 MOVI-C® CONTROLLER or visualization as OPC UA client

The following figure shows an example of communication between two MOVI-C® CONTROLLER. One device takes over the role of the server and the other the role of the client.



18014435110033803

OPC UA is increasingly replacing proprietary solutions such as ARTI or network variables as the communication standard between SEW-EURODRIVE products. In addition, OPC UA offers a standardized interface for communication between corresponding clients and servers from various other manufacturers.

## 4 MOVIKIT® OPC UA add-on SensorInterface

### 4.1 Module description

The MOVIKIT® OPC-UA add-on SensorInterface offers easy connection of various sensors via OPC UA client. The various sensor connections are grouped together under the license. The DUV40A sensor from SEW-EURODRIVE with part number 19175892 is recommended.

#### 4.1.1 Requirements

The MOVIKIT® OPC-UA add-on SensorInterface software module is a product based on the MOVIKIT® OPC-UA software module. This is why both components are needed. The DUV40A sensor is required in addition to the requirements for the MOVIKIT® OPC-UA software module.

#### 4.1.2 License type

Single (license for a MOVI-C® CONTROLLER). A separate license instance is required for each sensor used.

For more information on licensing, refer to the documents [MOVIKIT® Licensing](#) and [MOVI-C® Software Components](#). You can download the documents from the [SEW-EURODRIVE Website](#).

#### 4.1.3 Order information

The following table shows the order data for MOVIKIT® OPC-UA add-on SensorInterface in relation to the various performance classes of the MOVI-C® CONTROLLER.

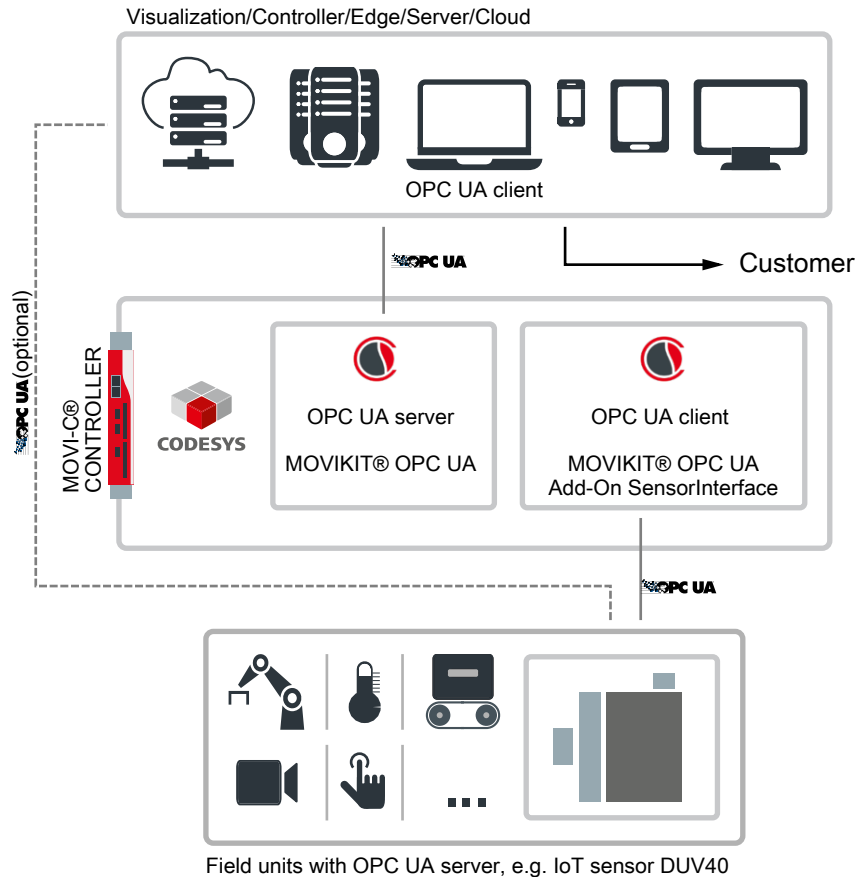
MOVI-C® CONTROLLER	Type designation	Part number
UHX25A (standard)	SMK1505-000	23373377
UHX45A (advanced)		
UHX65A (progressive)		
UHX86		

### 4.2 Functions

In addition to the MOVIKIT® OPC UA software module, the MOVIKIT® OPC UA add-on SensorInterface offers the following functions:

- Integrating a sensor as OPC UA server
- Transferring sensor data to the MOVIKIT® OPC UA software module within the controller.

The following figure shows an example using the DUV40 sensor on a MOVI-C® CONTROLLER as well as data transfer to a third-party product (e.g. monitor, HMI panel, higher-level controller, server or cloud).



36028830505332235

The path marked as "optional" from the sensor directly to the third-party product bypasses the controller and is therefore limited to the sensor data. It is not part of this software. The advantages of direct feedback from the MOVI-C® CONTROLLER to the process are not available in this setup.

### 4.3 Advantages

- Easy connection of the DUV40A sensor from SEW-EURODRIVE
- Feedback to the process is possible, since IEC data from the MOVI-C® CONTROLLER is available in addition to the sensor data.
- Flexible use of DUV40, not necessarily for the monitoring of motor and gear unit
- Cost-effective basis for existing and new IoT applications with products from SEW-EURODRIVE (also taking into account the costs for the MOVIKIT® OPC UA software module and the MOVI-C® CONTROLLER)



## 5 Startup

This chapter describes the startup steps specific to the software module in more detail. For more detailed information and notes on the startup of all other devices and software components used in the project, refer to the corresponding documentation.

### 5.1 General information

OPC UA (Open Platform Communications Unified Architecture) according to IEC 62541 is increasingly establishing itself as a communication standard in mechanical and plant engineering. This makes it easier to network devices and exchange information.

Unlike conventional fieldbus connections or visualization interfaces, a device has the ability to provide symbolic information about itself, which means about its properties, variables, process values, and capabilities. The other devices in the network can read and write these objects. In this way, OPC UA meets the requirements of a standardized, service-oriented architecture (SOA). Defined profiles include, for example, process data exchange (data access) and methods (historical access and alarms and conditions).

Conventional protection goals of IT security include among others to ensure that the data is only accessible to authorized participants, that the origin of the data is known, and that the data cannot be manipulated. Part 2 of the OPC UA specification (security model) describes the security mechanisms for this communication. A distinction is made between security settings (security policy and message security mode) to protect the data and authentication settings as a requirement on the clients and servers involved in the communication.

### 5.2 Software required for startup

The following software is required:

- MOVISUITE® engineering software
- OMH memory card with MOVIRUN®; an OMW memory card (Windows 10 IoT) is not required.

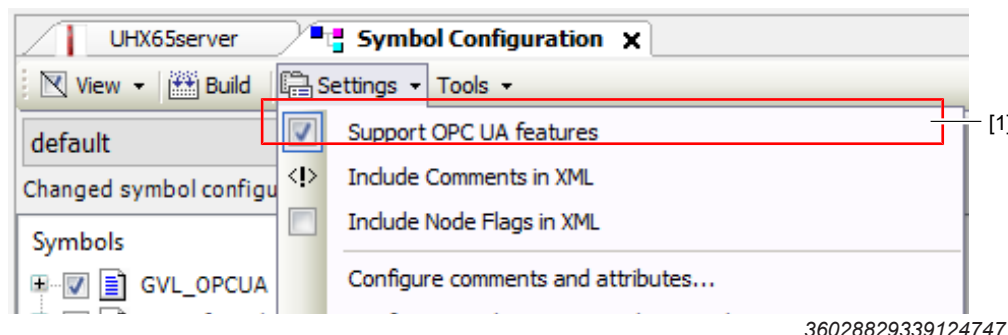
For more detailed information on the hardware requirements of the individual software components, see the documentation for the respective software.

### 5.3 Usecase: The simple OPC UA server without IT security

#### 5.3.1 Settings

The OPC UA server implements the "Micro Embedded Device Server" profile and is an additional software module on the real-time part of the MOVI-C® CONTROLLER (OMH card). The software module is a performance license. For the designation and part number of the software module matching your MOVI-C® CONTROLLER, refer to section "Order information" (→ 10).

If the property "Support OPC UA features" is active in the symbol configuration in the IEC editor, the software module provides the selected data for suitable OPC UA clients. The symbol configuration is automatically created when generating the IEC project.



[1] Menu item [Support OPC UA features]

The following modes are used:

- Security Policy = None
- Message Security Mode = None
- Authentication Settings = Anonymous

The binary protocol is used for communication. In the connection setup, it is typically identified as `opc.tcp://<Server_IP>:4840`. The IP address of the server corresponds to the IP address of the engineering interface X80 selected for the MOVI-C® CONTROLLER. Port 4840 is registered with IANA and reserved for OPC UA.

If the communication partner expects a symbolic host name, it can be specified via MOVISUITE® (preferred solution) or via a correspondingly adapted file `SewPlcIp.xml` in the folder `System` on the OMH card of the MOVI-C® CONTROLLER. In the delivery state of the OMH card, this file has the name `SewPlcIp.Example.xml` and is not active. In this case, adjust the host name via MOVISUITE® in the configuration of the MOVI-C® CONTROLLER in the configuration menu [Communication] > [Engineering] using the "Saved host name" field.

Otherwise, open the file `SewPlcIp.Example.xml` to assign the host name, adjust the host name there in the code accordingly, save the file under the name `SewPlcIp.xml` in the same folder and switch the MOVI-C® CONTROLLER off and on again.

```

</ipConfig_V4>
<!-- LAN 3: Engineering Interface -->
<IpConfig_V4 InterfaceNumber="2">
  <DHCP>false</DHCP>
  <IpAddress>192.168.10.160</IpAddress>
  <NetMask>255.255.255.0</NetMask>
  <GatewayAddress>192.168.10.1</GatewayAddress>
  <NameServer>0.0.0.0</NameServer>
  [1] <HostName>MOVI-C-ENG</HostName>
</IpConfig_V4>
<!-- LAN 4 -->
<IpConfig_V4 InterfaceNumber="1">

```

9007231574895627

[1] Symbolic host name

### 5.3.2 Testing the communication

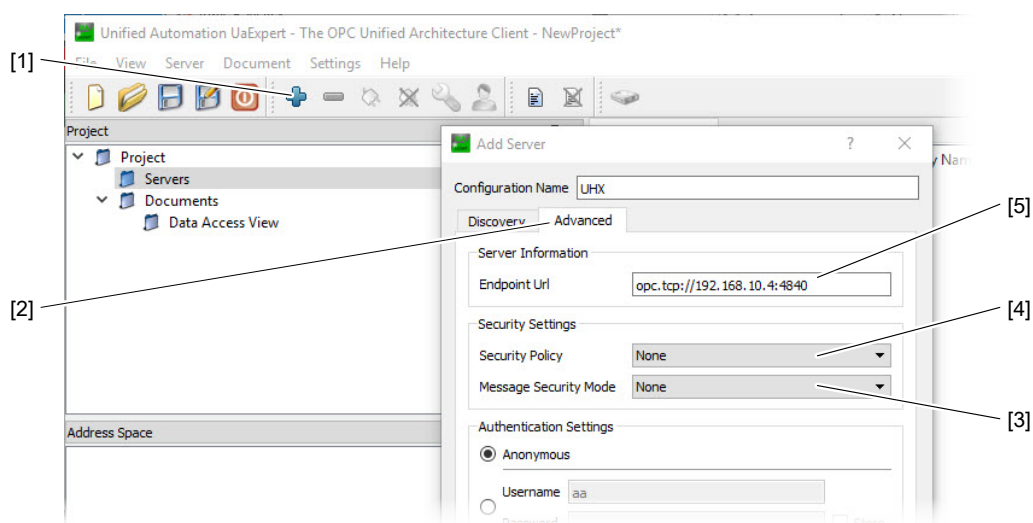
You can test the communication with the OPC UA server using the open source software UaExpert from Unified Automation. You can download UaExpert from the Internet and install it on your computer.

To perform a simple test, do the following:

- ✓ You have defined process data values for communication via OPC UA in the symbol configuration of the IEC program.
- ✓ You have loaded the IEC program onto the MOVI-C® CONTROLLER.
- ✓ You have installed the UaExpert software.

## 1. Start UaExpert.

⇒ The user interface is displayed:



18014430871850379

- [1] [Add Server] button
- [2] "Advanced" tab
- [3] "Message Security Mode" drop-down list
- [4] "Security Policy" drop-down list
- [5] "Endpoint Url" edit box

## 2. Click on the [Add Server] button.

⇒ The "Add Server" window opens.

## 3. Open the "Advanced" tab.

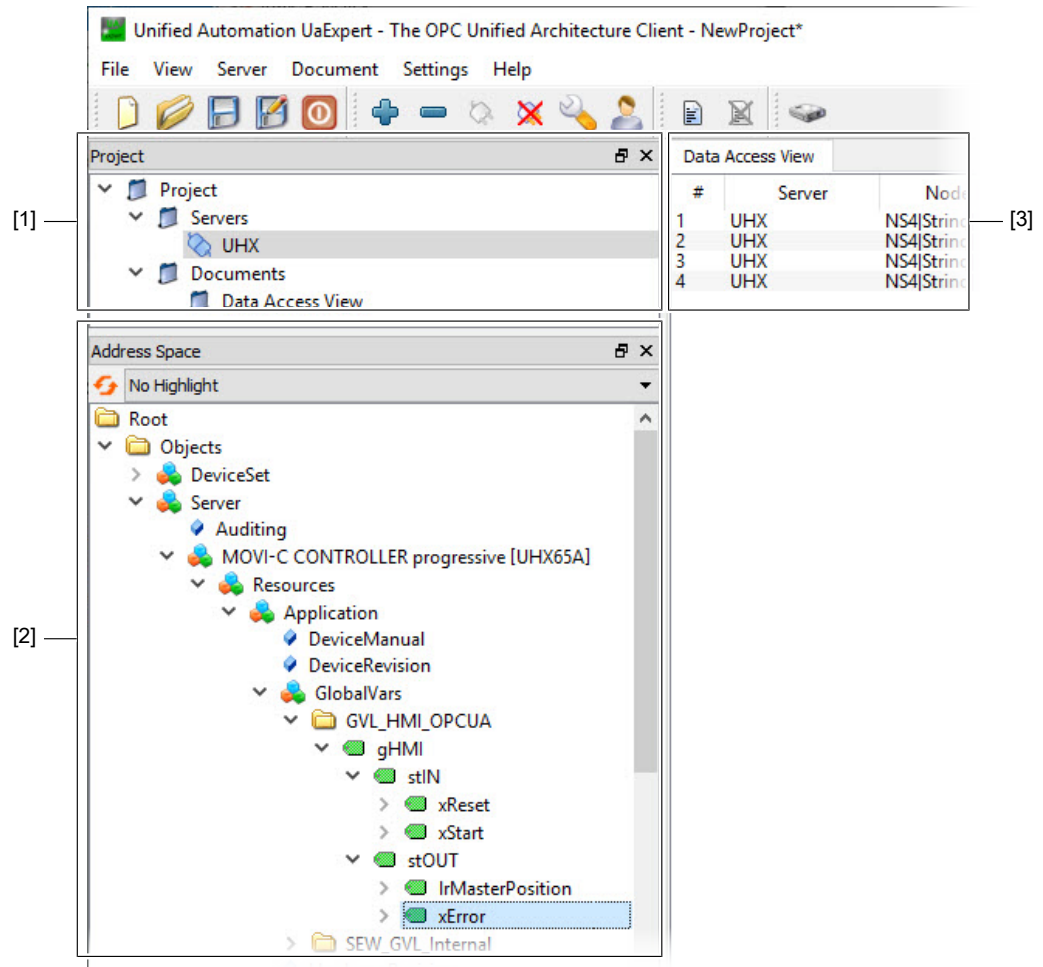
## 4. In the "Endpoint Url" edit box, enter the protocol, IP address and port of the OPC UA server.

## 5. Select "None" from the "Security Policy" drop-down list.

## 6. Select "None" from the "Message Security Mode" drop-down list.

## 7. Click on the [OK] button.

8. To establish a connection to the OPC UA server, click on the menu item [Server]/[Connect].
  - ⇒ The OPC UA server is shown in the "Project" window.



27021630131404043

- [1] "Project" window
- [2] Tree structure
- [3] "Data Access View" window

9. Expand the tree structure.
10. Drag and drop the variables provided by the OPC UA server into the "Data Access View" window.

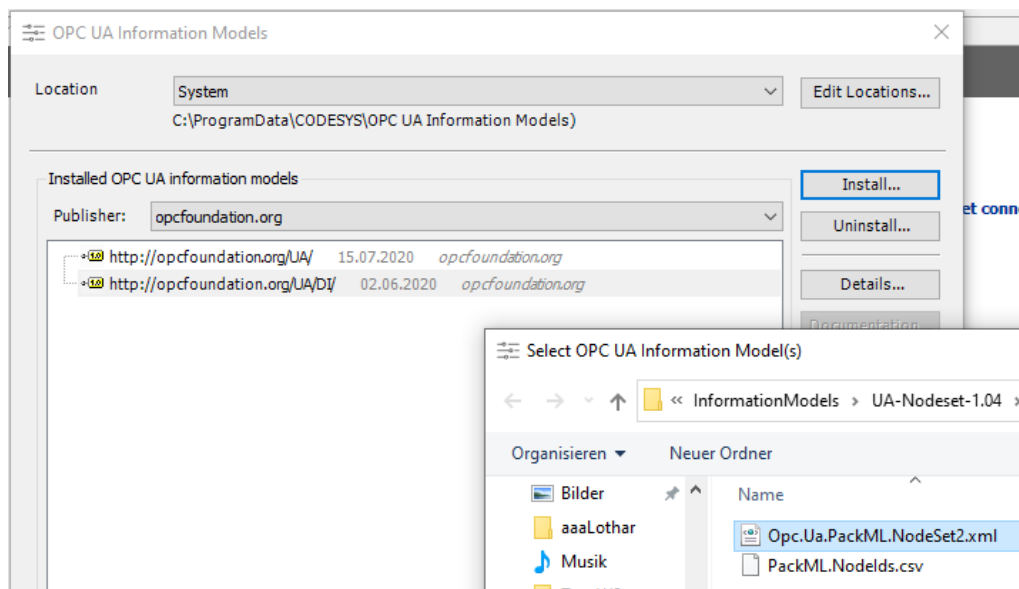
## 5.4 Usecase: The OPC UA server with information model without IT security

### 5.4.1 Storing OPC UA information model

If the OPC UA information model to be used is not yet installed, it can be installed in the IEC Editor (or in CODESYS) as follows:

- ✓ The IEC Editor or CODESYS is open.

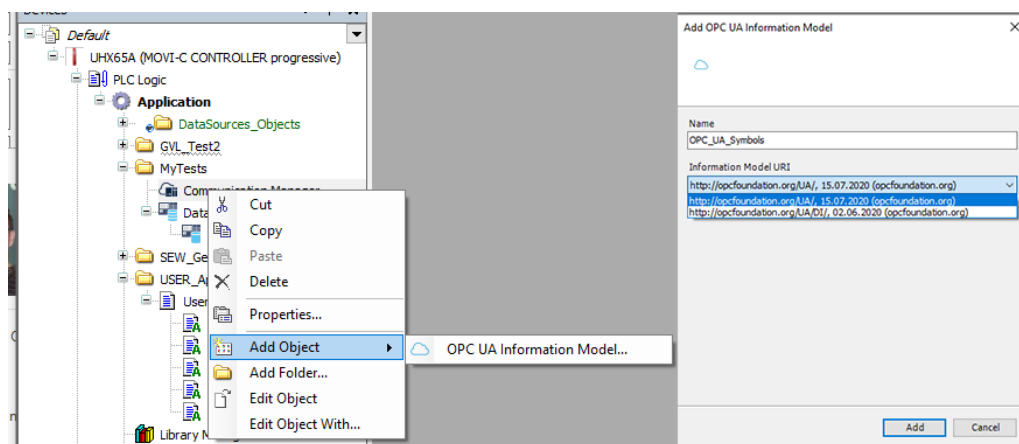
1. Open the [Tools] > [OPC UA Information Model Repository...] menu.
2. Click [Install...]  
⇒ The "Select OPC UA Information Model(s)" dialog box is displayed.
3. Select the XML file with the OPC UA information model and click [Open].



40154895371

If the "Communication Manager" object is included in the application, an OPC UA information model can be stored from the repository by means of an XML import as follows:

In the context menu of the "Communication Manager" object under "Add Object", open the entry "OPC UA Information Model..." and select the XML file with the OPC UA information model in the "Add OPC UA Information Model(s)" dialog box.



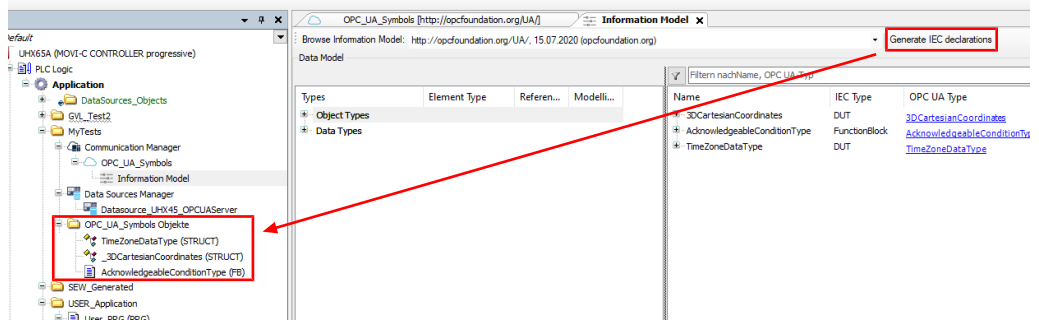
40154897803

#### 5.4.2 Transferring OPC UA declarations into the program

The OPC UA information model can be used to transfer declarations for standardized data structures and function blocks to the program. The following procedures are available for transferring declarations.

**Procedure 1: Create, instantiate and search declarations**

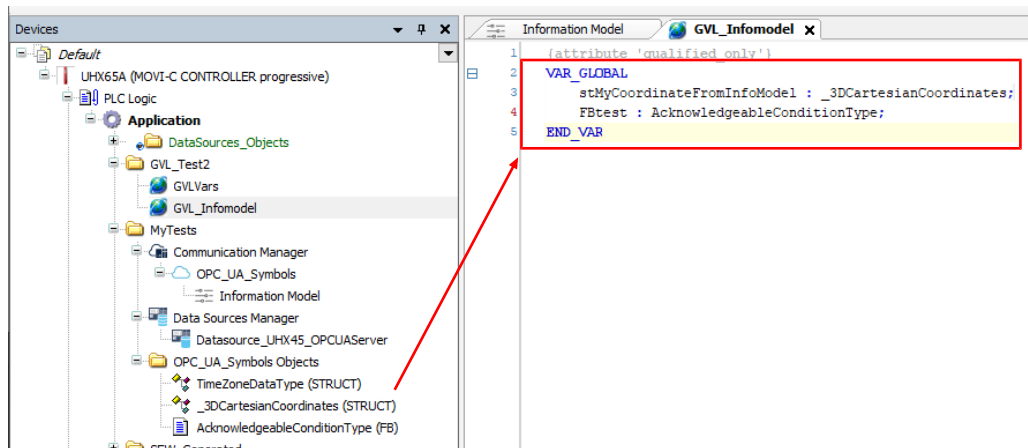
1. Open the object added in the device tree with the OPC UA information model stored.
2. Drag and drop the object and data types to be used in the project from the list with all object and data types in the left column to the right column.
3. Click [Generate IEC declarations].



40240089227

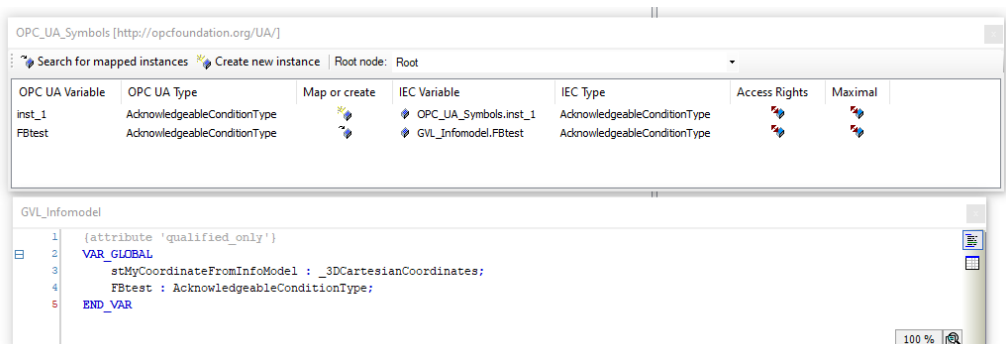
⇒ The selected object and data types are added as objects in the device tree.

4. Instantiate the selected object and data types.



40240098443

5. Search for the mapped instances shown and insert the instances mapped in the table using [Search for mapped instances].



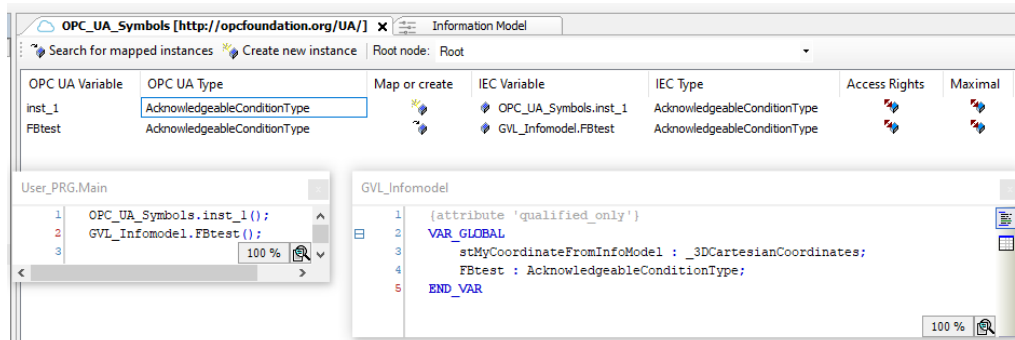
40154905099

⇒ The function block can be used in the program.



**Procedure 2: Inserting a new instance directly**

1. Insert a new instance in the table via [Create new instance].



40154907531

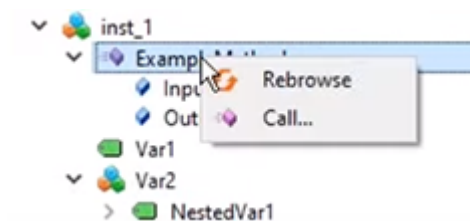
⇒ The function block can be used in the program.

**5.4.3 Testing communication**

You can test the communication with the OPC UA server using the open source software UaExpert from Unified Automation. You can download UaExpert from the Internet and install it on your computer.

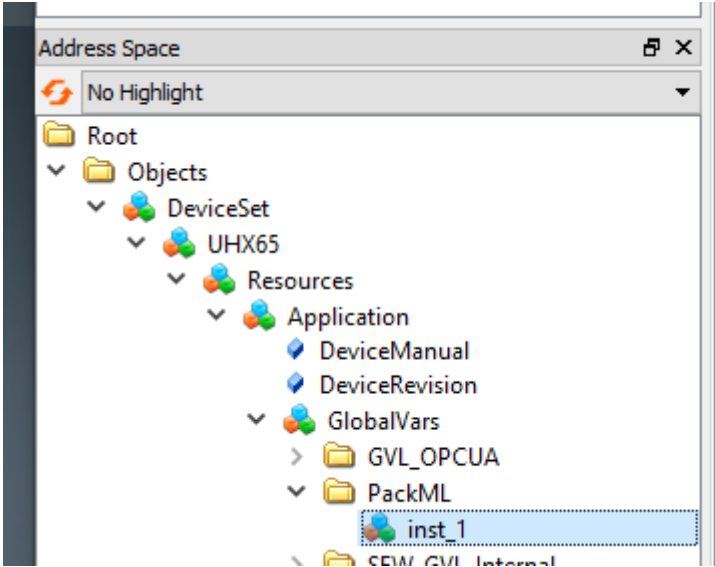
To perform a simple test, do the following:

- ✓ You have loaded the IEC program onto the MOVI-C® CONTROLLER.
  - ✓ You have installed the UaExpert software.
1. Start UaExpert.
  2. Click [Rebrowse] at the corresponding object.



40154909963

⇒ The variables released in the symbol configuration are visible in the test client.

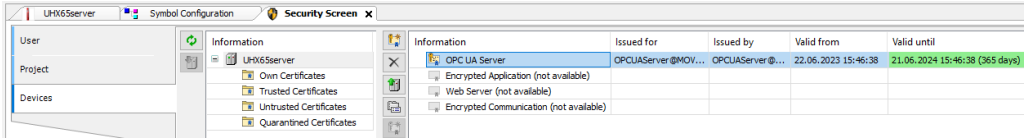


43088463627

5.5 Use case: OPC UA server with certificates

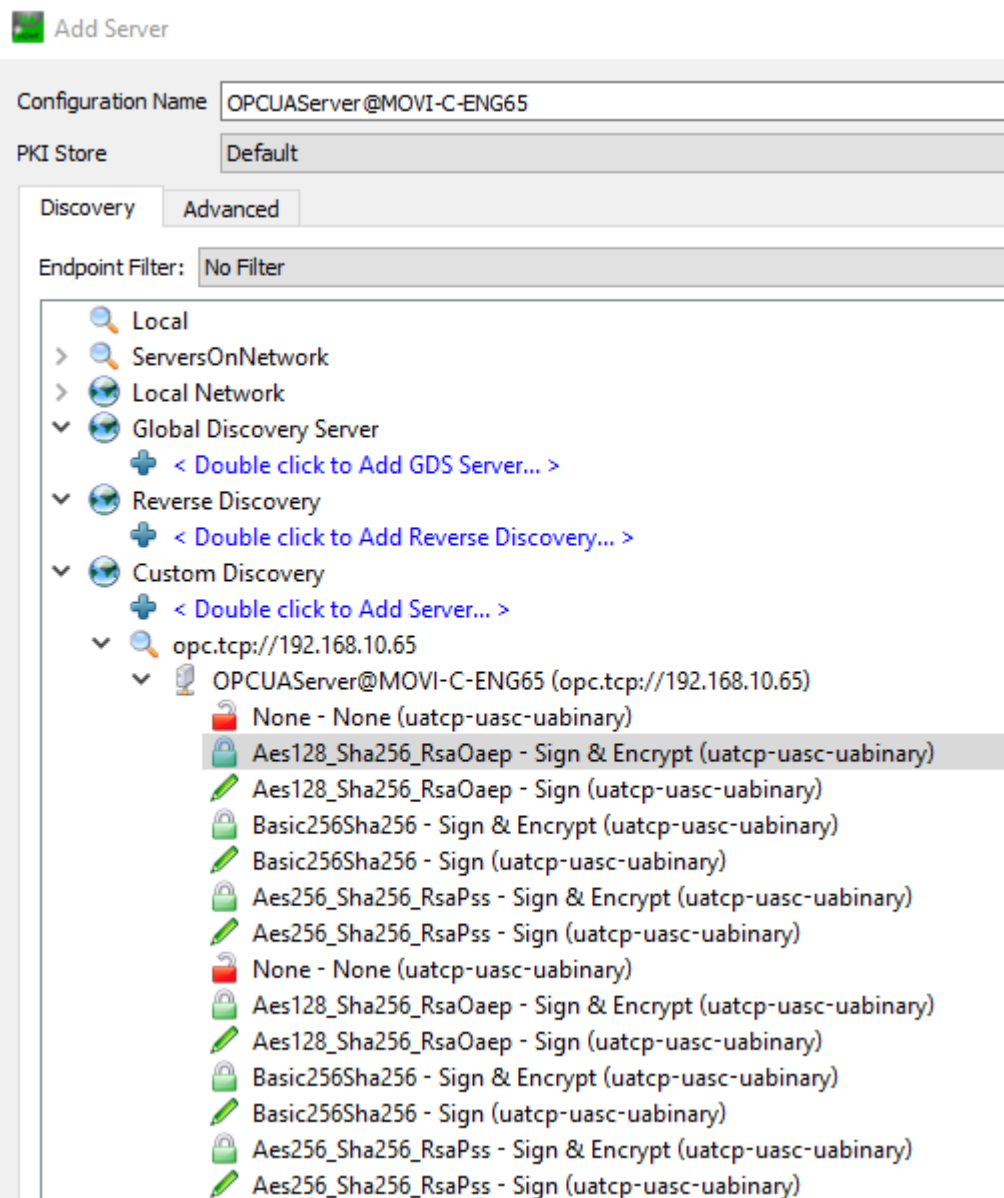
The communication partners can use certificates to familiarize themselves, exchange their public keys and establish a secure connection. The certificates of the server can be scanned in the dialog box under the [Devices] tab in the [View] [> Security Screen] dialog box. In the right-hand dialog, you create a self-signed certificate of the server.

The controller must be switched off and on for the change to take effect.



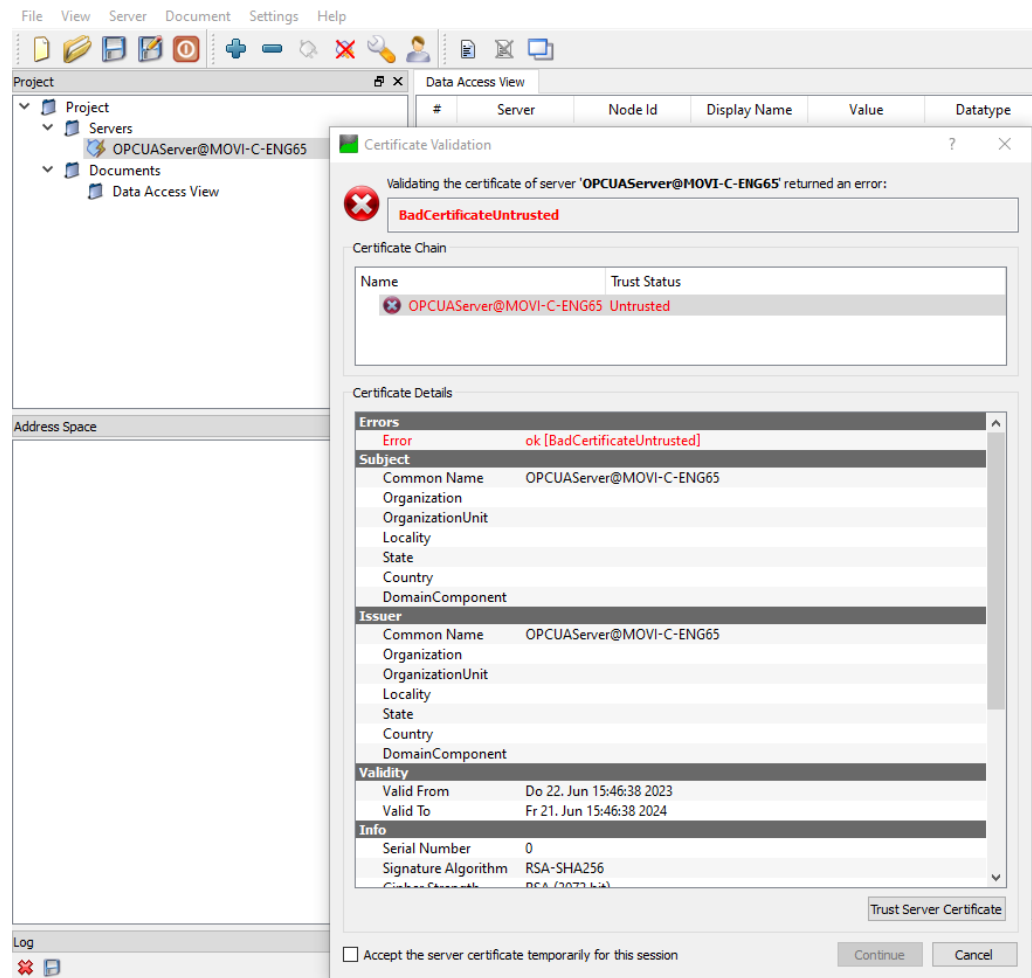
43089722507

The security settings supported by the server are visible and selectable in the client.



43089728011

In order for the self-signed certificate to be added to the list of "managed certificates", the operator must trust the certificate manually when establishing a connection with the server for the first time.



43089730827

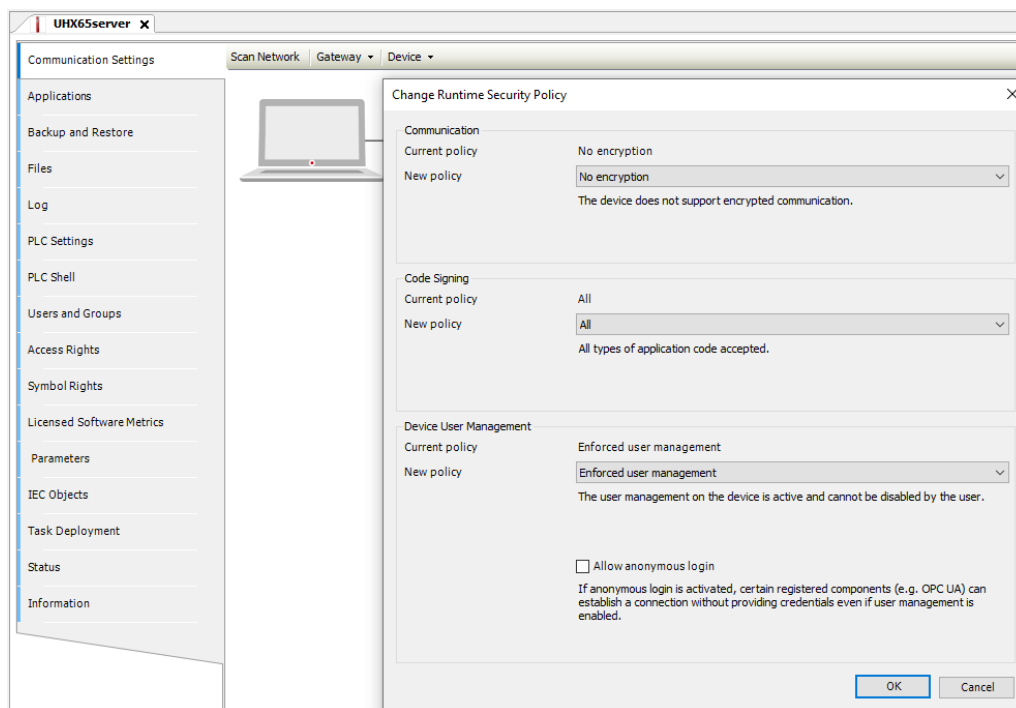
To establish the connection completely, you must trust the certificate of the client in the security screen of the IEC Editor. Drag and drop the certificate from the quarantine folder to the folder of the trusted certificates.

## 5.6 Use case: OPC UA server with additional user management

Set up your user and password in addition to using the certificates.

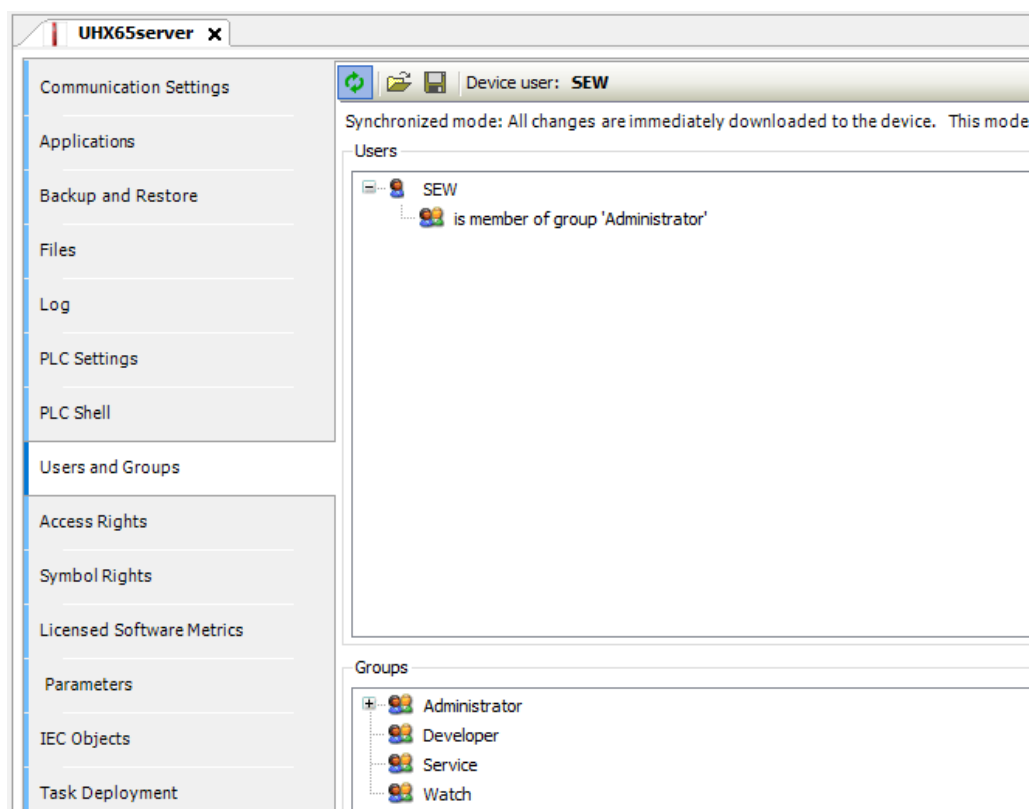
✓ To do so, set the following settings in the [Device] pull-down menu:

1. Open the [Change Runtime Security Policy] dialog.
2. Select [New policy = Enforced user management] and [Allow anonymous login] in the settings.



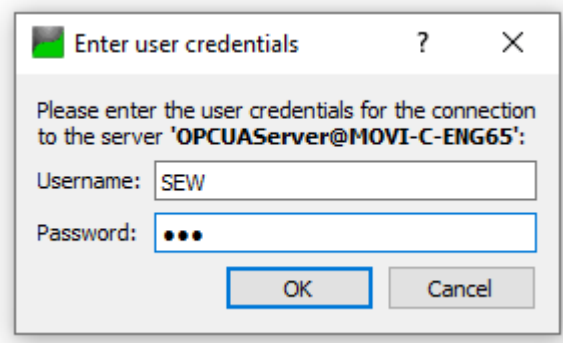
43090057739

Create a user with password in the [Users and Groups] menu. Follow the example of the administrator group with the user SEW as a member.



43090064139

Switch the control off and back on again. The user logs on to the test client with user name and password and has access to the variable values again.



43090425867

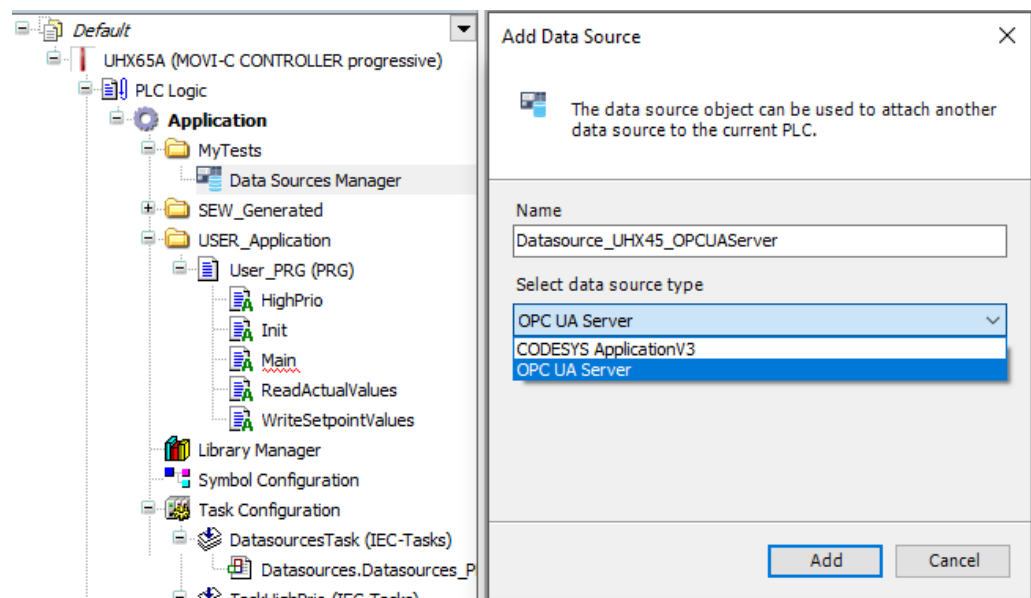
Additional symbol groups can be formed. Different user groups can read and write different variables.

For further information refer to the [Codesys](#) website.

## 5.7 MOVI-C CONTROLLER® or MOVI-C® visualization as OPC UA client

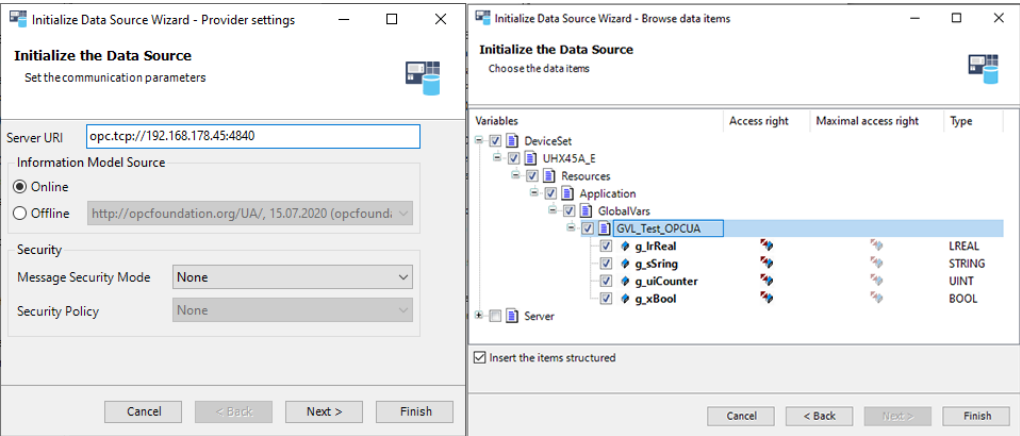
An OPC UA client can read the variables of the OPC UA server and exchange data in the IEC editor or in the editor of the MOVI-C® visualization. Proceed as follows to set up the data exchange:

1. In the IEC project, add a "Data Sources Manager" object in the device tree.
2. Add a "Data source" object below the "Data Sources Manager" object that establishes the connection with an OPC UA server. For this purpose, the OPC UA server must be active in the network and can be searched.



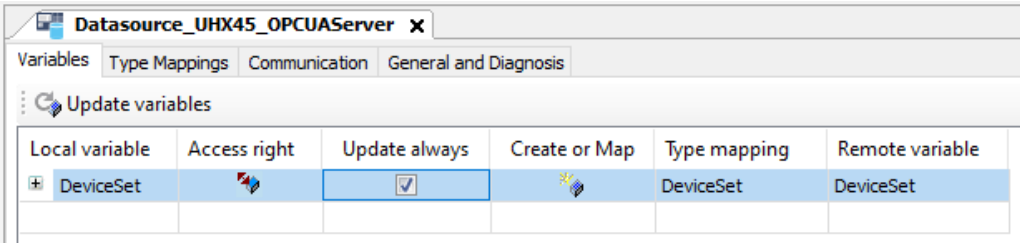
40151980043

3. Select the required variables.



40151982475

4. Activate the [Update always] option for the selected variables.

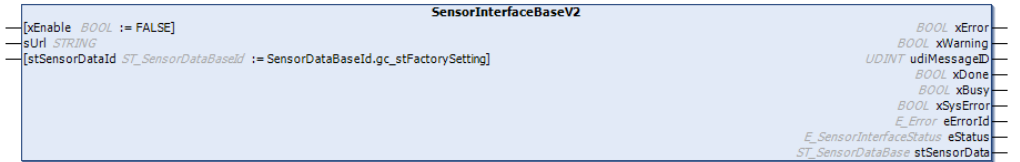


40154892939

- ⇒ The client does not consider certificates and users with passwords.
- ⇒ Communication takes place without IT security measures.
- ⇒ These are only possible with software based on CODESYS 3.5.18.40 or higher.

5.8 MOVIKIT® OPC UA add-on SensorInterface

A defined measurement task can be performed with the MOVIKIT® and the associated function block without connecting the DUV sensor as OPC UA server via the data source manager. Since this is possible in the OPC UA client, access can be made in addition to or instead via the variables in the data source manager. The library and the SensorInterfaceBaseV2 function block are available from SEW-EURODRIVE on request.



43092691979



## 6 Further useful information

### 6.1 Use of certificates

Certificates have a validity date. If they are not renewed, they expire. Therefore, it is necessary that the date and time are set to the same in the client and server. Both stations use an NTP client and the Internet to have the same time.

### 6.2 Security policies and performance

The following security policies are supported:

- BASIC256SHA256
- AES128SHA256RSAOAE
- AES256SHA256RSAPSS

Switch the controller off and on again if you have created or deleted OPC UA server certificates via the CODESYS Security Screen. Certain certificate checks can only be performed during boot-up.

OPC UA communication is implemented as a freefloat task so that the performance of the application is not affected in a controller with several cores if cores are not used by the application.

### 6.3 Changing OPC UA configuration at runtime

Settings can be changed using the `DatasourceOpcUaServer.UserConfigurationV1` variable type. By default, `SubscriptionPublishingInterval` is set to 1 second. You can change the time to retrieve the data of the OPC UA server more quickly. Set the data source to read the communication settings of variables.

Example:

A config variable of the type `DatasourceOpcUaServer.UserConfigurationV1` is declared. Select "config" under "Read from Variable" in the data source. Then set `xValid` of the variable to `TRUE` and the publishing interval to 400 ms.



43092695691

#### Mapping of the IP address to a symbolic host name

For example, under Microsoft Windows, there are OPC UA clients for which it is not possible to enter a fixed IP address in the connection establishment but a symbolic alias name must be used. The alias name is always possible.

If you need this function under Windows, change the `hosts` file in the `C:\Windows\System32\drivers\etc.` folder, for example to

```

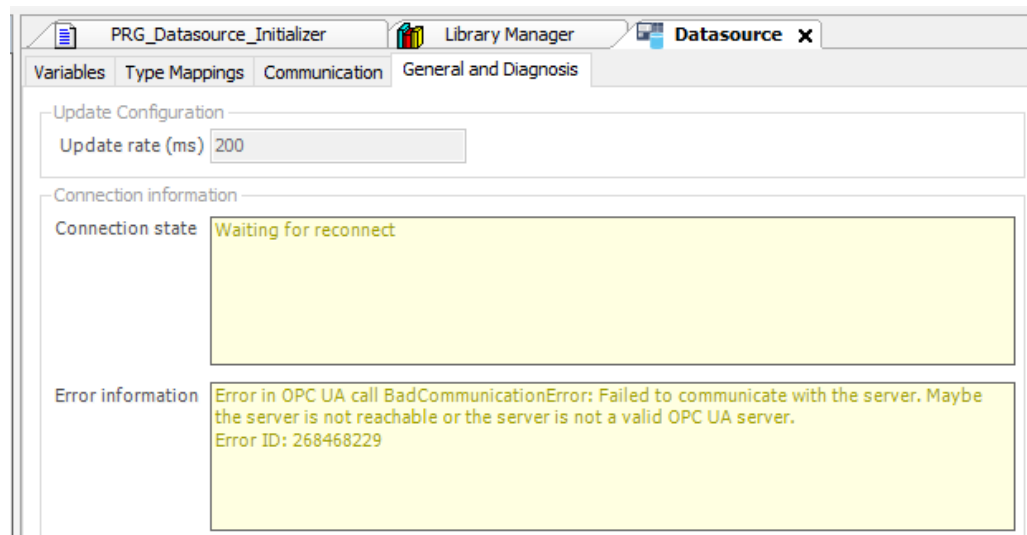
1  # Copyright (c) 1993-2009 Microsoft Corp.
2  #
3  # This is a sample HOSTS file used by Microsoft TCP/IP for Windows.
4  #
5  # This file contains the mappings of IP addresses to host names. Each
6  # entry should be kept on an individual line. The IP address should
7  # be placed in the first column followed by the corresponding host name.
8  # The IP address and the host name should be separated by at least one
9  # space.
10 #
11 # Additionally, comments (such as these) may be inserted on individual
12 # lines or following the machine name denoted by a '#' symbol.
13 #
14 # For example:
15 #
16 #       102.54.94.97       rhino.acme.com          # source server
17 #       38.25.63.10       x.acme.com              # x client host
18 #
19 # localhost name resolution is handled within DNS itself.
20 #   127.0.0.1       localhost
21 #   ::1             localhost
22 #   192.168.10.45    MOVI-C-ENG45
23 #   192.168.10.65    MOVI-C-ENG65
24
25

```

9007242347440011

### Diagnostics of the OPC UA connection

The status of the OPC UA connection can be read out and evaluated for diagnostic purposes. The connection status and error state are displayed in the IEC Editor and evaluated in the application program.



43092881035

Read the status of the runtime in the program with, for example with

PRG\_DataSource X

```

1  PROGRAM PRG_DataSource
2  VAR
3      source      : DataSources.IDatasources.IDatasource;
4      source3     : DataSources.IDatasources.IDatasource3;
5      state       : DatasourceOpcUaServer.IDatasources.MonitoringState;
6      udiError    : UDINT;
7      xResult     : BOOL;
8
9      _state      : DatasourceOpcUaServer.IDatasources.MonitoringState;
10     _udiError    : UDINT;
11 END_VAR
12
13
14     source := g_DataSources.GetDataSource ('Datasource');
15     xResult := __QUERYINTERFACE (source, source3);
16
17     IF xResult THEN
18         state := source3.State;
19         udiError := source3.Error;
20     END IF

```

43092884363

Where

MonitoringState (ENUM)

TYPE MonitoringState :

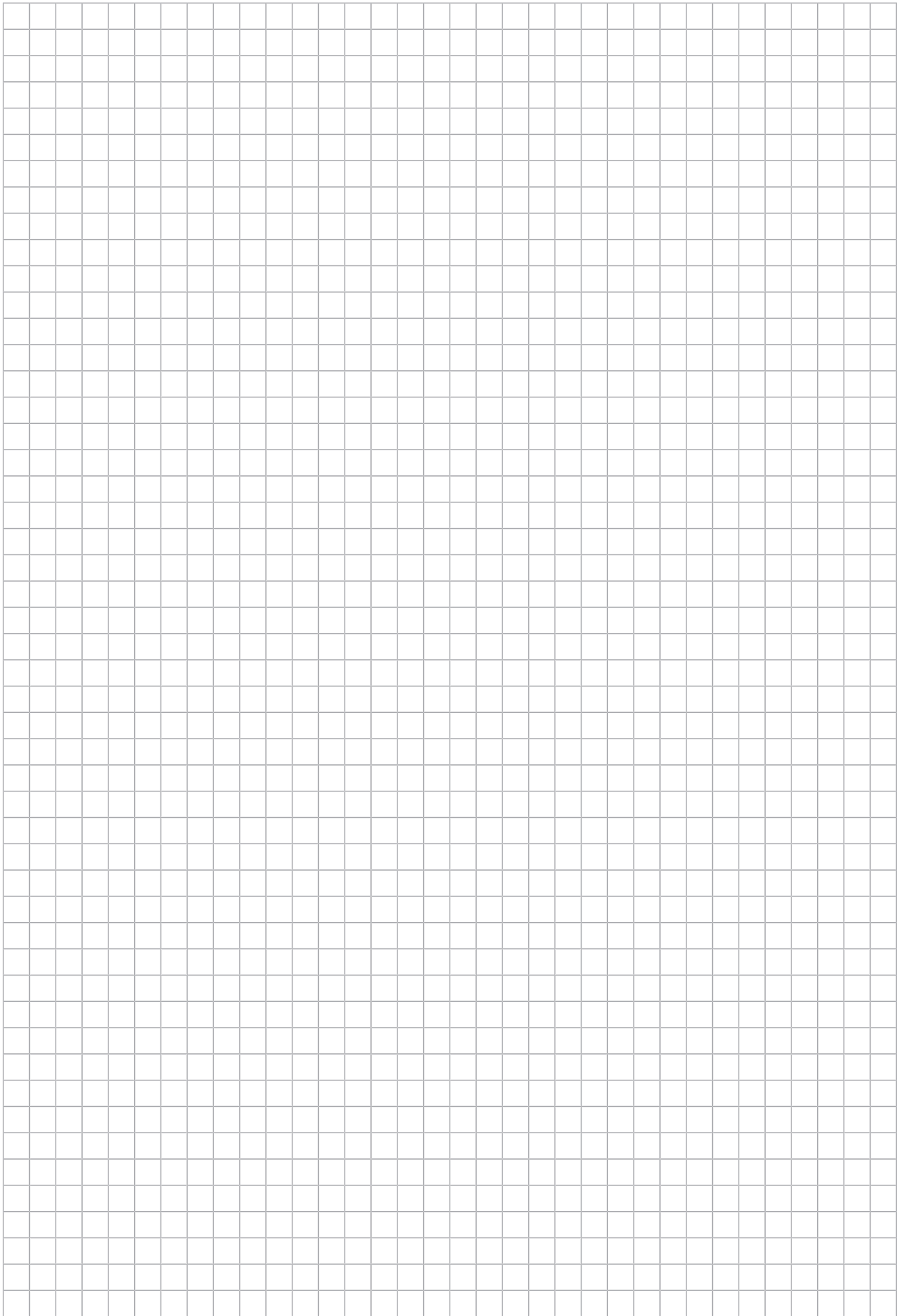
Is used to monitor the state of a data source for the data source editor or in an IEC application

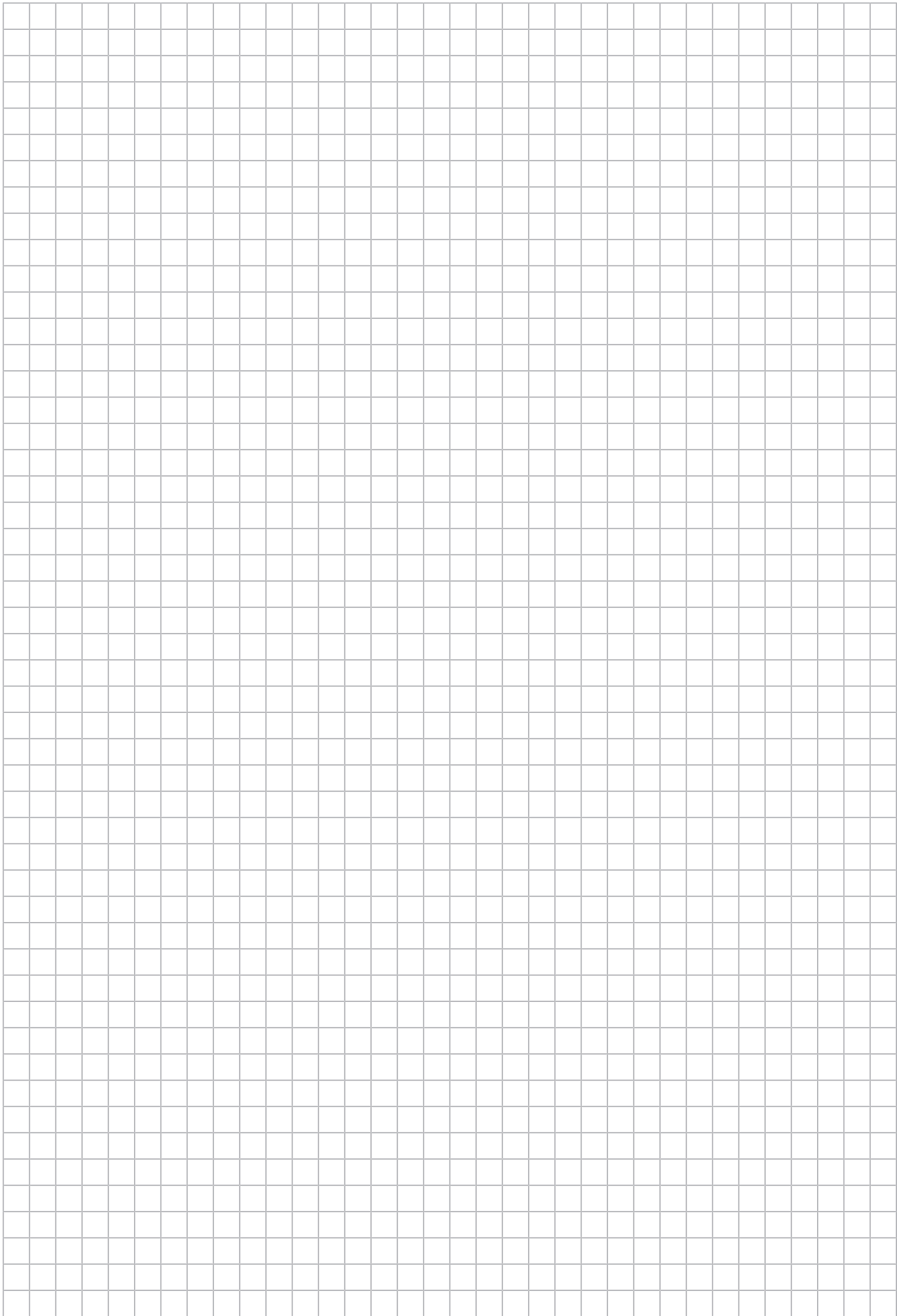
Attributes:  
qualified\_only

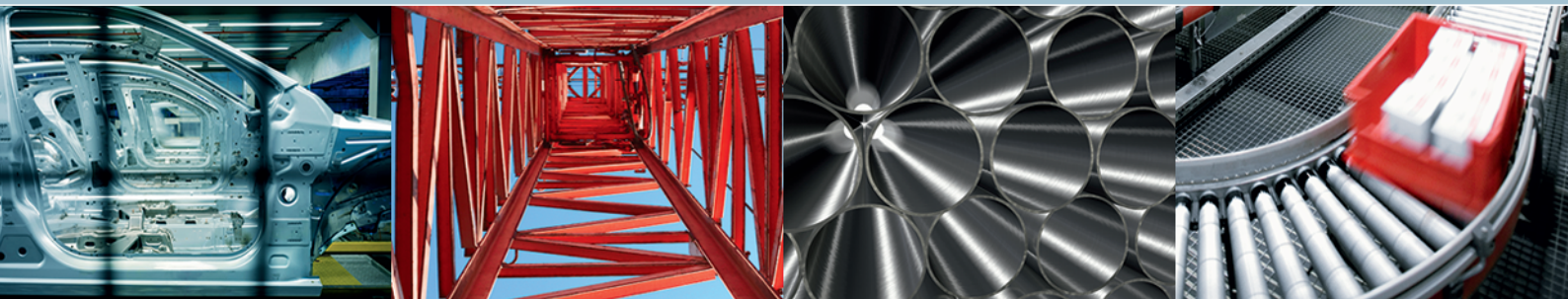
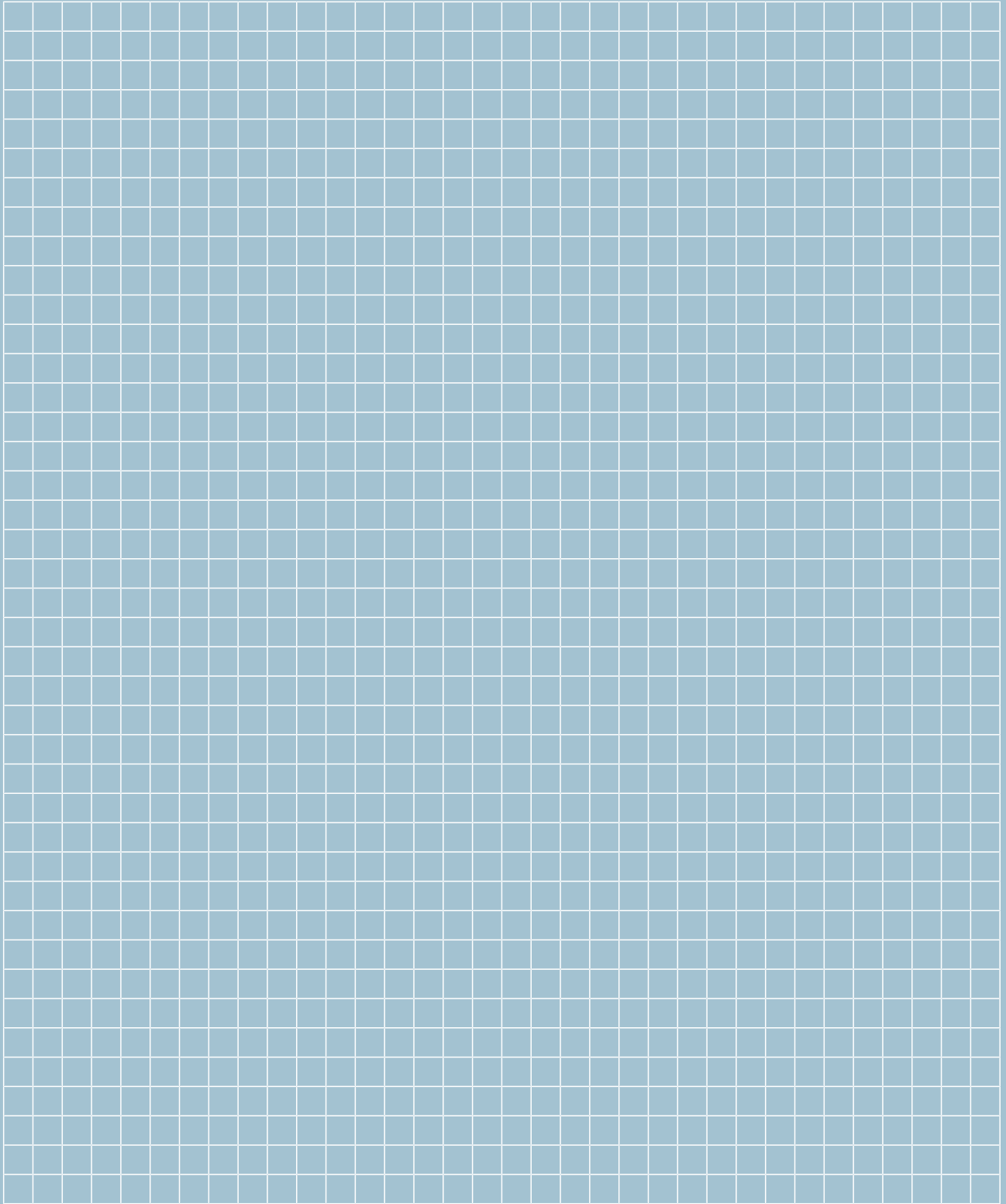
InOut:

Name	Comment
Initialized	Left after a call to StartConnect
Connecting	Switches to Connected after succeeded Connecting and to WaitingForReconnect after errors
Connected	The datasource is connected
ShuttingDown	Left to Shutdown after succeeded call to ShuttingDown
Shutdown	Terminal state
WaitingForReconnect	Left after some time in favor of Initialized
WaitingForCommSetting	Waiting that dynamic configuration setting has been ready
Unknown	

43092887179









**SEW-EURODRIVE**  
Driving the world