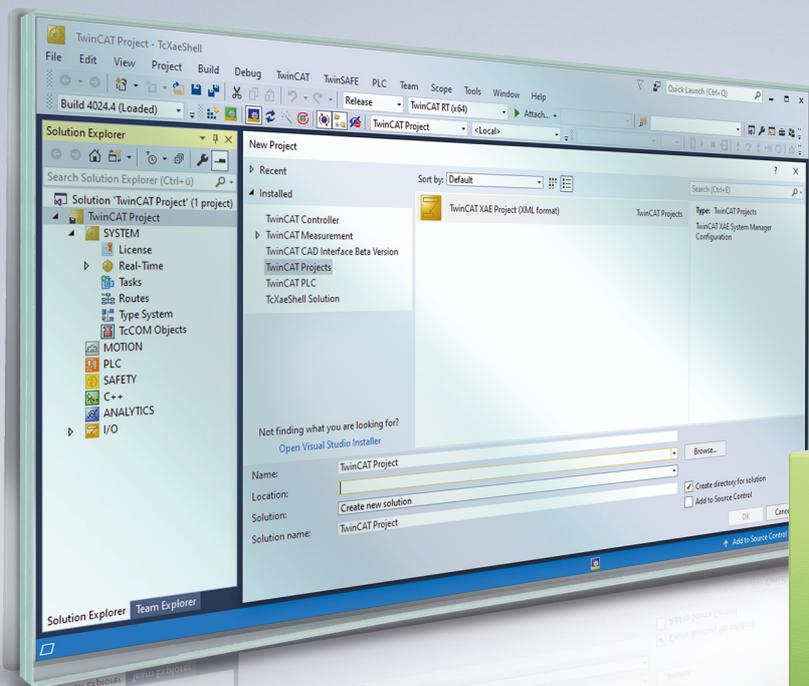


BECKHOFF New Automation Technology

Handbuch | DE

TE6100

TwinCAT 3 OPC UA Nodeset Editor



Inhaltsverzeichnis

1	Vorwort.....	5
1.1	Hinweise zur Dokumentation	5
1.2	Zu Ihrer Sicherheit.....	6
1.3	Hinweise zur Informationssicherheit	7
1.4	Ausgabestände der Dokumentation.....	7
2	Überblick.....	8
3	Installation	10
3.1	Systemanforderungen	10
3.2	Installation	10
3.3	Lizenzierung	10
4	Technische Einführung	13
4.1	Quick Start	13
4.1.1	Einfaches Nodeset mit Basistypen.....	13
4.1.2	Einfache Companion Specification	19
4.2	Unterstützte Features.....	23
4.3	Lizenz-Informationen.....	25
4.4	Benutzeroberfläche	25
4.4.1	Überblick	25
4.4.2	Type Explorer.....	26
4.4.3	Instance Explorer	27
4.4.4	Attribute Editor	27
4.4.5	Models.....	28
4.4.6	Menü	28
4.4.7	Message Viewer.....	29
4.4.8	TwinCAT Target Browser.....	29
4.4.9	Themes	30
4.5	Mapping	32
4.6	Applikationsverzeichnisse	34
4.7	Applikations- und Projekteinstellungen	35
5	Samples	38
6	Anhang.....	39
6.1	Begriffe.....	39
6.2	Troubleshooting	39
7	Support und Service	40

1 Vorwort

1.1 Hinweise zur Dokumentation

Diese Beschreibung wendet sich ausschließlich an ausgebildetes Fachpersonal der Steuerungs- und Automatisierungstechnik, das mit den geltenden nationalen Normen vertraut ist.

Zur Installation und Inbetriebnahme der Komponenten ist die Beachtung der Dokumentation und der nachfolgenden Hinweise und Erklärungen unbedingt notwendig.

Das Fachpersonal ist verpflichtet, stets die aktuell gültige Dokumentation zu verwenden.

Das Fachpersonal hat sicherzustellen, dass die Anwendung bzw. der Einsatz der beschriebenen Produkte alle Sicherheitsanforderungen, einschließlich sämtlicher anwendbaren Gesetze, Vorschriften, Bestimmungen und Normen erfüllt.

Disclaimer

Diese Dokumentation wurde sorgfältig erstellt. Die beschriebenen Produkte werden jedoch ständig weiterentwickelt.

Wir behalten uns das Recht vor, die Dokumentation jederzeit und ohne Ankündigung zu überarbeiten und zu ändern.

Aus den Angaben, Abbildungen und Beschreibungen in dieser Dokumentation können keine Ansprüche auf Änderung bereits gelieferter Produkte geltend gemacht werden.

Marken

Beckhoff®, TwinCAT®, TwinCAT/BSD®, TC/BSD®, EtherCAT®, EtherCAT G®, EtherCAT G10®, EtherCAT P®, Safety over EtherCAT®, TwinSAFE®, XFC®, XTS® und XPlanar® sind eingetragene und lizenzierte Marken der Beckhoff Automation GmbH.

Die Verwendung anderer in dieser Dokumentation enthaltenen Marken oder Kennzeichen durch Dritte kann zu einer Verletzung von Rechten der Inhaber der entsprechenden Bezeichnungen führen.

Patente

Die EtherCAT-Technologie ist patentrechtlich geschützt, insbesondere durch folgende Anmeldungen und Patente:

EP1590927, EP1789857, EP1456722, EP2137893, DE102015105702

mit den entsprechenden Anmeldungen und Eintragungen in verschiedenen anderen Ländern.



EtherCAT® ist eine eingetragene Marke und patentierte Technologie lizenziert durch die Beckhoff Automation GmbH, Deutschland

Copyright

© Beckhoff Automation GmbH & Co. KG, Deutschland.

Weitergabe sowie Vervielfältigung dieses Dokuments, Verwertung und Mitteilung seines Inhalts sind verboten, soweit nicht ausdrücklich gestattet.

Zu widerhandlungen verpflichten zu Schadenersatz. Alle Rechte für den Fall der Patent-, Gebrauchsmuster- oder Geschmacksmustereintragung vorbehalten.

1.2 Zu Ihrer Sicherheit

Sicherheitsbestimmungen

Lesen Sie die folgenden Erklärungen zu Ihrer Sicherheit.
Beachten und befolgen Sie stets produktspezifische Sicherheitshinweise, die Sie gegebenenfalls an den entsprechenden Stellen in diesem Dokument vorfinden.

Haftungsausschluss

Die gesamten Komponenten werden je nach Anwendungsbestimmungen in bestimmten Hard- und Software-Konfigurationen ausgeliefert. Änderungen der Hard- oder Software-Konfiguration, die über die dokumentierten Möglichkeiten hinausgehen, sind unzulässig und bewirken den Haftungsausschluss der Beckhoff Automation GmbH & Co. KG.

Qualifikation des Personals

Diese Beschreibung wendet sich ausschließlich an ausgebildetes Fachpersonal der Steuerungs-, Automatisierungs- und Antriebstechnik, das mit den geltenden Normen vertraut ist.

Signalwörter

Im Folgenden werden die Signalwörter eingeordnet, die in der Dokumentation verwendet werden. Um Personen- und Sachschäden zu vermeiden, lesen und befolgen Sie die Sicherheits- und Warnhinweise.

Warnungen vor Personenschäden

GEFAHR

Es besteht eine Gefährdung mit hohem Risikograd, die den Tod oder eine schwere Verletzung zur Folge hat.

WARNUNG

Es besteht eine Gefährdung mit mittlerem Risikograd, die den Tod oder eine schwere Verletzung zur Folge haben kann.

VORSICHT

Es besteht eine Gefährdung mit geringem Risikograd, die eine mittelschwere oder leichte Verletzung zur Folge haben kann.

Warnung vor Umwelt- oder Sachschäden

HINWEIS

Es besteht eine mögliche Schädigung für Umwelt, Geräte oder Daten.

Information zum Umgang mit dem Produkt



Diese Information beinhaltet z. B.:
Handlungsempfehlungen, Hilfestellungen oder weiterführende Informationen zum Produkt.

1.3 Hinweise zur Informationssicherheit

Die Produkte der Beckhoff Automation GmbH & Co. KG (Beckhoff) sind, sofern sie online zu erreichen sind, mit Security-Funktionen ausgestattet, die den sicheren Betrieb von Anlagen, Systemen, Maschinen und Netzwerken unterstützen. Trotz der Security-Funktionen sind die Erstellung, Implementierung und ständige Aktualisierung eines ganzheitlichen Security-Konzepts für den Betrieb notwendig, um die jeweilige Anlage, das System, die Maschine und die Netzwerke gegen Cyber-Bedrohungen zu schützen. Die von Beckhoff verkauften Produkte bilden dabei nur einen Teil des gesamtheitlichen Security-Konzepts. Der Kunde ist dafür verantwortlich, dass unbefugte Zugriffe durch Dritte auf seine Anlagen, Systeme, Maschinen und Netzwerke verhindert werden. Letztere sollten nur mit dem Unternehmensnetzwerk oder dem Internet verbunden werden, wenn entsprechende Schutzmaßnahmen eingerichtet wurden.

Zusätzlich sollten die Empfehlungen von Beckhoff zu entsprechenden Schutzmaßnahmen beachtet werden. Weiterführende Informationen über Informationssicherheit und Industrial Security finden Sie in unserem <https://www.beckhoff.de/secguide>.

Die Produkte und Lösungen von Beckhoff werden ständig weiterentwickelt. Dies betrifft auch die Security-Funktionen. Aufgrund der stetigen Weiterentwicklung empfiehlt Beckhoff ausdrücklich, die Produkte ständig auf dem aktuellen Stand zu halten und nach Bereitstellung von Updates diese auf die Produkte aufzuspielen. Die Verwendung veralteter oder nicht mehr unterstützter Produktversionen kann das Risiko von Cyber-Bedrohungen erhöhen.

Um stets über Hinweise zur Informationssicherheit zu Produkten von Beckhoff informiert zu sein, abonnieren Sie den RSS Feed unter <https://www.beckhoff.de/secinfo>.

1.4 Ausgabestände der Dokumentation

Version	Änderung
1.0.x	Erste Veröffentlichung

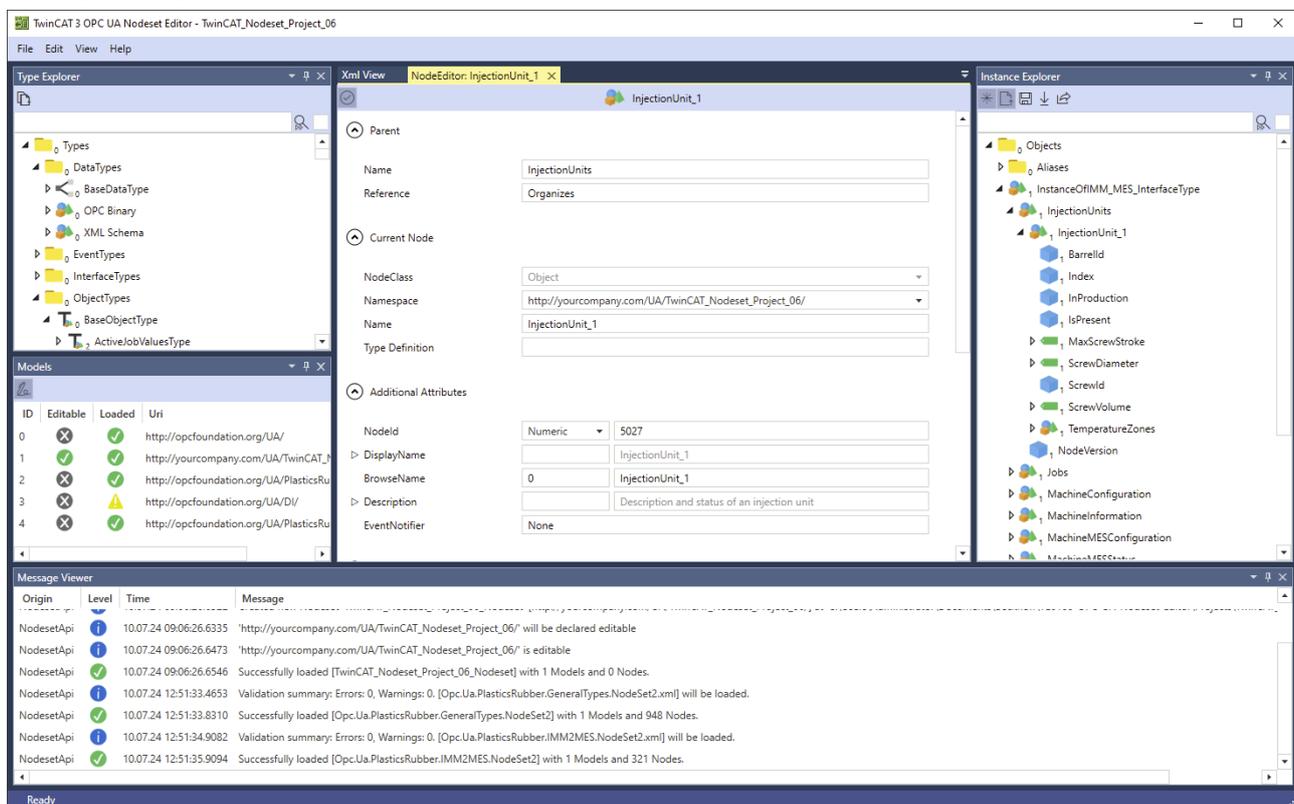
2 Überblick

Der TwinCAT OPC UA Nodeset Editor ist ein Engineering-Tool zur Erstellung und Bearbeitung von OPC-UA-Nodeset-Dateien, die insbesondere für Companion Specifications verwendet werden. Mit Hilfe des Editors können Sie vorhandene Companion Specifications oder eigene Informationsmodelle auf Ihrer Beckhoff-Steuerung abbilden und mit Datenpunkten aus der SPS verknüpfen.

Bei der Entwicklung einer Maschinenanwendung gibt es oft Anforderungen an die Struktur des OPC UA Namespace einer Maschine, die abgebildet werden soll. In diesem Fall muss die Konformität mit einer Companion Specification oder einem kundenspezifischen Informationsmodell hergestellt werden, damit die Maschine möglichst reibungslos in eine bestehende und standardisierte Umgebung integriert werden kann.

Der TwinCAT OPC UA Nodeset Editor ist die grafische Lösung, um diese Herausforderung zu meistern. Er erlaubt:

- Laden von Nodeset-Dateien aus Companion Specifications oder benutzerdefinierten Informationsmodellen.
- Erstellung von einfachen benutzerdefinierten Nodeset-Dateien.
- Bequeme Verknüpfung von Objekten aus der Steuerung mit solchen aus einem geladenen oder erstellten Nodeset.
- Verwendung des Nodesets für neue und bestehende Maschinen mit dem TwinCAT OPC UA Server (TF6100).



Use-Cases

Der TwinCAT OPC UA Nodeset Editor unterstützt den Anwender bei der Umsetzung von Anforderungen an Companion Specifications, indem er eine grafische Benutzeroberfläche bereitstellt, die den Import von Nodesets ermöglicht. Aber auch ohne Anforderungen an eine spezifische Companion Specification kann der TwinCAT OPC UA Nodeset Editor ein nützliches Werkzeug sein und bei den folgenden Anwendungsfällen helfen.

Erstellung eines eigenen Server-Adressraums

Bereits vor der Verbreitung von Companion Specifications gab es erste Anfragen, Informationen aus der SPS nach einer definierten Struktur auf dem TwinCAT OPC UA Server zur Verfügung zu stellen. Standardmäßig verwendet der Server das IEC61131-3 Informationsmodell, um Variablen von der SPS über

OPC UA bereitzustellen. Manchmal ist diese Datendarstellung auf dem Server jedoch nicht erwünscht, und es gab Anfragen nach einer "einfachen Modellierungsoption für Ordner und einfache Datentypen". Dieser Anwendungsfall kann mit Hilfe des vorliegenden Produkts gelöst werden.

Verknüpfung von Variablen aus bestehenden Nodesets

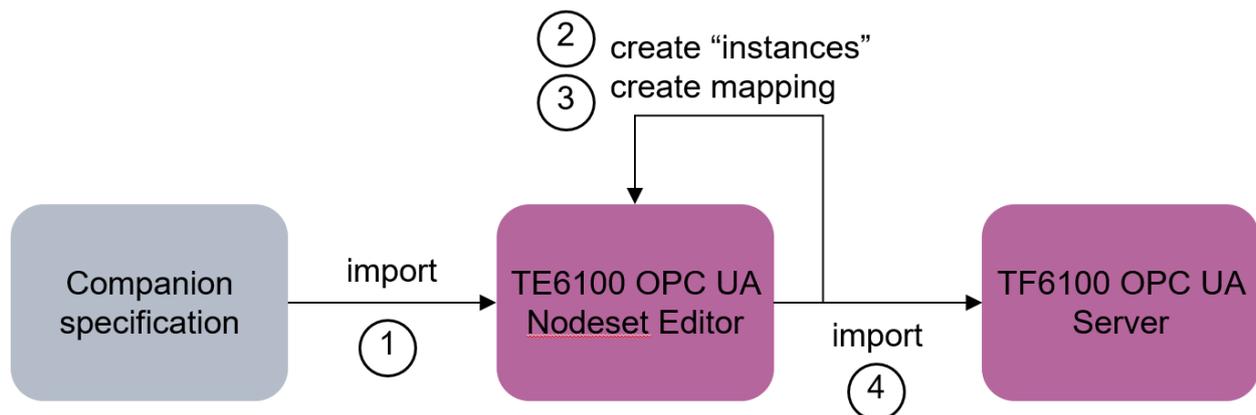
Nodesets, die mit einem Drittanbieter-Tool, wie z. B. dem UA Modeler (<https://www.unified-automation.com>), erstellt wurden, können in den TwinCAT OPC UA Server importiert werden. Da Drittanbieter-Tools jedoch keine Variableninformationen aus einem TwinCAT-Steuerungsprogramm haben, kann stattdessen der TwinCAT OPC UA Nodeset Editor verwendet werden. Die vorhandenen Nodesets können in die grafische Benutzeroberfläche des Nodeset Editors importiert und dort mit Hilfe des TwinCAT Target Browsers mit SPS-Variablen verknüpft werden. Die resultierende Nodeset-Datei kann dann auf dem TwinCAT OPC UA Server bereitgestellt werden.

Erstellen von Instanz-Nodesets aus einer Companion Specification

Der Import von Companion Specifications und die Erstellung von Instanzen aus den darin enthaltenen Daten und Objekttypen ist erforderlich, wenn eine Maschine mit einer Standardschnittstelle ausgestattet werden soll, die durch die Companion Specifications definiert wurde. Der TwinCAT OPC UA Nodeset Editor ermöglicht die Instanziierung solcher Daten- und Objekttypen und die anschließende Verknüpfung der Instanzen mit Variablen aus der TwinCAT PLC.

Workflow

Der Workflow bei der Verwendung des TwinCAT OPC UA Nodeset Editors zur Erstellung eines Nodesets und dem anschließenden Export zum TwinCAT OPC UA Server kann wie folgt dargestellt werden.



In Schritt (1) wird die Nodeset-Datei (die "Companion Specification") in den TwinCAT OPC UA Nodeset Editor importiert. Das bedeutet, dass die in der Datei definierten Datentypen in der Editorumgebung zur Verfügung stehen und für die Projektierung genutzt werden können.

In Schritt (2) und (3) findet die Projektierung statt, d.h. aus den Datentypen werden Objekte ("Instanzen") erzeugt und mit Variablen aus der SPS verknüpft.

In Schritt (4) wird das generierte Projekt in den TwinCAT OPC UA Server importiert, damit dieser das generierte Informationsmodell in seinem Adressraum zur Verfügung stellen kann.

Sehen Sie dazu auch

📖 Unterstützte Features [► 23]

3 Installation

3.1 Systemanforderungen

Die folgende Tabelle beschreibt die technischen Systemvoraussetzungen für den reinen Betrieb der Anwendung.

Technische Daten	Beschreibung
Betriebssystem	Windows 10, 11
Zielplattform	PC-Architektur (x86,x64)
Minimale TwinCAT-Version	TwinCAT 3.1 Build 4026
Erforderliches TwinCAT-Setup-Level	TwinCAT 3 XAE, XAR, ADS
Benötigte TwinCAT-Lizenz	TE6100 TC3 OPC UA Nodeset Editor L1 TE6101 TC3 OPC UA Nodeset Editor L2 (optionale Zusatzlizenz)

● TF6100 TC3 OPC UA Server

i Dieses Produkt arbeitet mit dem TwinCAT OPC UA Server zusammen, um ein erstelltes Nodeset auszuführen und es im Adressraum des Servers anzuzeigen. Bitte stellen Sie sicher, dass das Produkt TF6100 TC3 OPC UA Server auf Ihrem System installiert ist.

3.2 Installation

Wenn Sie TwinCAT 3.1 Build 4026 (und höher) auf dem Windows-Betriebssystem verwenden, können Sie diese Funktion über den TwinCAT Package Manager installieren, siehe [Dokumentation zur Installation](#). Normalerweise installieren Sie die Funktion über den entsprechenden Workload. Sie können die im Workload enthaltenen Pakete aber auch einzeln installieren. Diese Dokumentation beschreibt im Folgenden kurz den Installationsvorgang über den Workload.

Kommandozeilenprogramm TcPkg

Über das TcPkg Command Line Interface (CLI) können Sie sich die verfügbaren Workloads auf dem System anzeigen lassen:

```
tcpkg list -t workload
```

Über das folgende Kommando können Sie den Workload der Funktion installieren.

```
tcpkg install TE6100.OpcUaNodesetEditor.XAE
```

TwinCAT Package Manager UI

Über das User Interface (UI) können Sie sich alle verfügbaren Workloads anzeigen lassen und diese bei Bedarf installieren.

Folgen Sie hierzu den entsprechenden Anweisungen in der Oberfläche.

Die Installation des entsprechenden Pakets oder Workloads kann entweder über TcPkg CLI oder TcPkg UI erfolgen.

● Unvorbereiteter TwinCAT-Neustart kann Datenverlust erzeugen

i Die Installation dieser Funktion hat unter Umständen einen TwinCAT-Neustart zur Folge. Stellen Sie sicher, dass keine kritischen TwinCAT-Applikationen auf dem System laufen oder fahren Sie diese zunächst geordnet herunter.

3.3 Lizenzierung

Die TwinCAT 3 Function ist als Vollversion oder als 7-Tage-Testversion freischaltbar. Beide Lizenztypen sind über die TwinCAT-3-Entwicklungsumgebung (XAE) aktivierbar.

Lizenzierung der Vollversion einer TwinCAT 3 Function

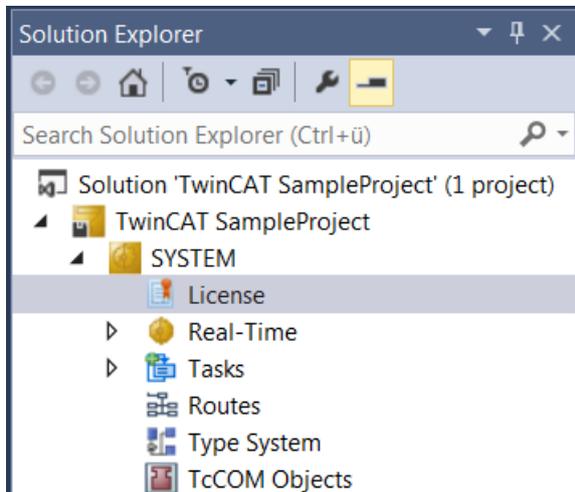
Die Beschreibung der Lizenzierung einer Vollversion finden Sie im Beckhoff Information System in der Dokumentation „[TwinCAT-3-Lizenzierung](#)“.

Lizenzierung der 7-Tage-Testversion einer TwinCAT 3 Function



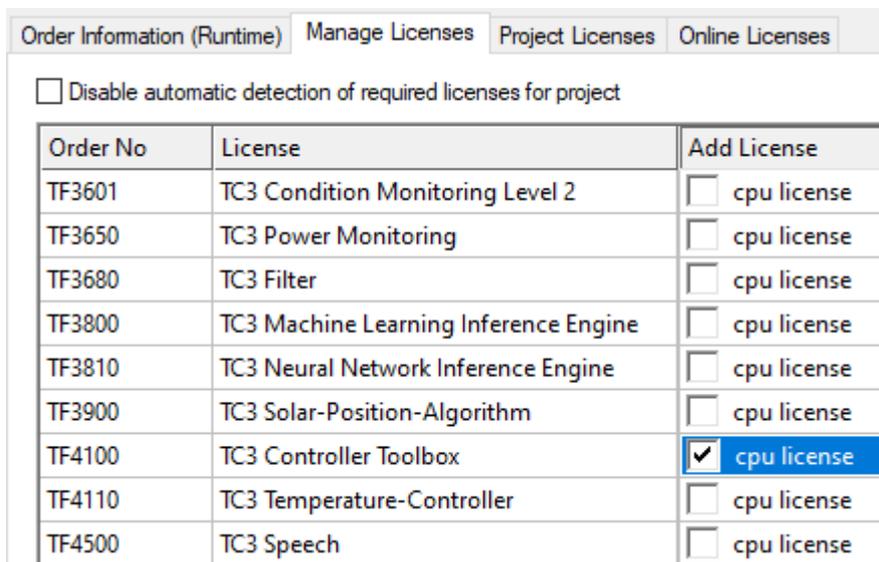
Eine 7-Tage-Testversion kann nicht für einen [TwinCAT-3-Lizenz-Dongle](#) freigeschaltet werden.

1. Starten Sie die TwinCAT-3-Entwicklungsumgebung (XAE).
2. Öffnen Sie ein bestehendes TwinCAT-3-Projekt oder legen Sie ein neues Projekt an.
3. Wenn Sie die Lizenz für ein Remote-Gerät aktivieren wollen, stellen Sie das gewünschte Zielsystem ein. Wählen Sie dazu in der Symbolleiste in der Drop-down-Liste **Choose Target System** das Zielsystem aus.
 - ⇒ Die Lizenzierungseinstellungen beziehen sich immer auf das eingestellte Zielsystem. Mit der Aktivierung des Projekts auf dem Zielsystem werden automatisch auch die zugehörigen TwinCAT-3-Lizenzen auf dieses System kopiert.
4. Klicken Sie im **Solution Explorer** im Teilbaum **SYSTEM** doppelt auf **License**.



⇒ Der TwinCAT-3-Lizenzmanager öffnet sich.

5. Öffnen Sie die Registerkarte **Manage Licenses**. Aktivieren Sie in der Spalte **Add License** das Auswahlkästchen für die Lizenz, die Sie Ihrem Projekt hinzufügen möchten (z. B. „TF4100 TC3 Controller Toolbox“).



6. Öffnen Sie die Registerkarte **Order Information (Runtime)**.

⇒ In der tabellarischen Übersicht der Lizenzen wird die zuvor ausgewählte Lizenz mit dem Status „missing“ angezeigt.

7. Klicken Sie auf **7 Days Trial License...**, um die 7-Tage-Testlizenz zu aktivieren.

Order Information (Runtime) Manage Licenses Project Licenses Online Licenses

License Device: Target (Hardware Id) Add...

System Id: 2DB25408-B4CD-81DF-5488-6A3D9B49EF19 Platform: other (91)

License Request

Provider: Beckhoff Automation Generate File...

License Id: Customer Id:

Comment:

License Activation

7 Days Trial License... License Response File...

⇒ Es öffnet sich ein Dialog, der Sie auffordert, den im Dialog angezeigten Sicherheitscode einzugeben.

Enter Security Code

Please type the following 5 characters: OK

Kg8T4

Cancel

8. Geben Sie den Code genauso ein, wie er angezeigt wird, und bestätigen Sie ihn.

9. Bestätigen Sie den nachfolgenden Dialog, der Sie auf die erfolgreiche Aktivierung hinweist.

⇒ In der tabellarischen Übersicht der Lizenzen gibt der Lizenzstatus nun das Ablaufdatum der Lizenz an.

10. Starten Sie das TwinCAT-System neu.

⇒ Die 7-Tage-Testversion ist freigeschaltet.

4 Technische Einführung

4.1 Quick Start

Das folgende Kapitel führt Sie durch einige Quickstart-Tutorials zur Verwendung des TwinCAT OPC UA Nodeset Editors. Bitte stellen Sie sicher, dass die Installationsanweisungen befolgt wurden und dass Ihr System die Systemanforderungen erfüllt. Die folgenden Quickstart-Tutorials sind verfügbar:

Tutorial	Beschreibung
Einfaches Nodeset mit Basistypen [► 13]	Demonstriert, wie ein Nodeset mit einfachen Datentypinstanzen erstellt wird.
Einfache Companion Specification [► 19]	Demonstriert, wie ein Nodeset mit Objektinstanzen aus einer Companion Specification erstellt wird.

4.1.1 Einfaches Nodeset mit Basistypen

Dieser Dokumentationsartikel soll Ihnen einen schnellen Einstieg in die Verwendung dieses Produkts ermöglichen. Nach der erfolgreichen [Installation \[► 10\]](#) und [Lizenzierung \[► 10\]](#) führen Sie die folgenden Schritte aus, um ein erstes Nodeset zu erstellen.

● OPC UA Base Nodeset

i Standardmäßig lädt der TwinCAT OPC UA Nodeset Editor automatisch das OPC UA Base Nodeset, das den Namespace <http://opcfoundation.org/UA/> definiert.

● TF6100 TC3 OPC UA Server

i Dieses Produkt arbeitet mit dem TwinCAT OPC UA Server zusammen, um ein erstelltes Nodeset auszuführen und es im Adressraum des Servers anzuzeigen. Bitte stellen Sie sicher, dass das Produkt TF6100 TC3 OPC UA Server auf Ihrem System installiert ist.

Das folgende Quickstart-Tutorial führt Sie durch die folgenden Schritte:

- Vorbereitung eines SPS-Projekts
- Neues Projekt anlegen
- Instanzen aus dem OPC UA Base Nodeset erstellen
- Erstellte Instanzen mit SPS-Variablen verknüpfen
- Deployment des Projekts auf dem TwinCAT OPC UA Server

Vorbereitung eines SPS-Projekts

Erstellen Sie ein neues TwinCAT-Projekt und fügen Sie ein SPS-Projekt hinzu. Erstellen Sie die folgende Datenstruktur im SPS-Projekt und instanziiieren Sie sie dreimal im MAIN-Programm.

```

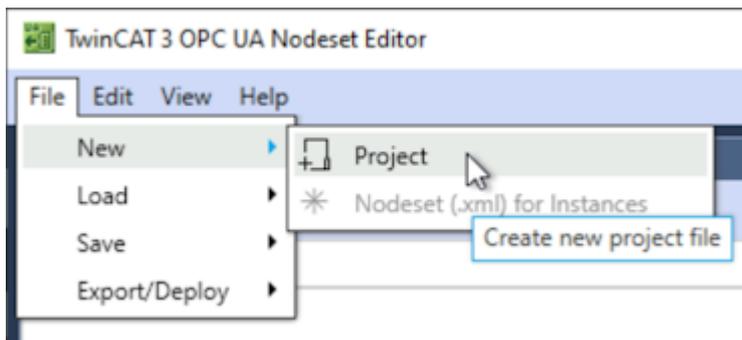
TYPE ST_MyStructure :
STRUCT
  MyBoolean : BOOL;
  MyInt16 : INT;
  MyFloat : REAL;
END_STRUCT
END_TYPE

PROGRAM MAIN
VAR
  MyStruct1 : ST_MyStructure;
  MyStruct2 : ST_MyStructure;
  MyStruct3 : ST_MyStructure;
END_VAR

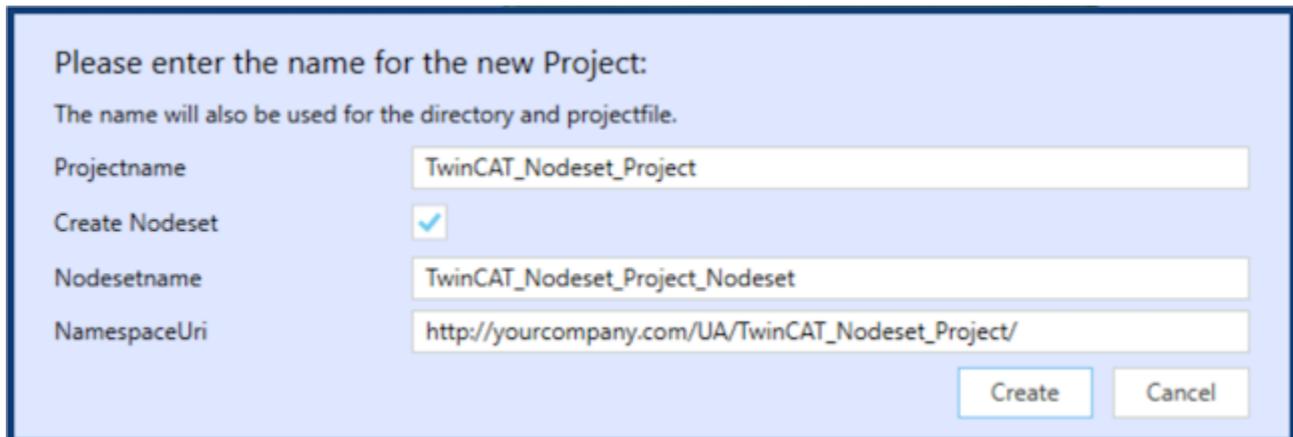
```

Neues Projekt anlegen

Klicken Sie im Menü auf **File > New > Project**, um ein neues Nodeset-Projekt zu erstellen.



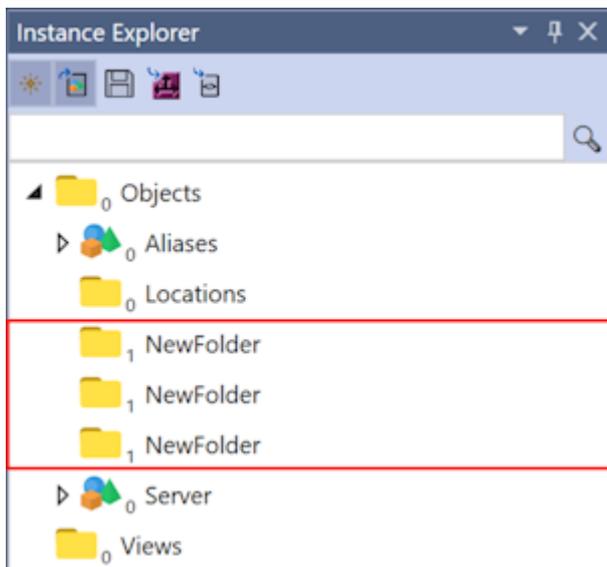
Im folgenden Auswahldialog können Sie den Namen und ein Zielverzeichnis für das neue Projekt angeben.



Instanzen erstellen

In diesem Tutorial wollen wir eine Reihe von Instanzen mit einfachen Datentypen (Integer, Boolean, Float) erstellen und sie in drei verschiedenen Ordnern gruppieren.

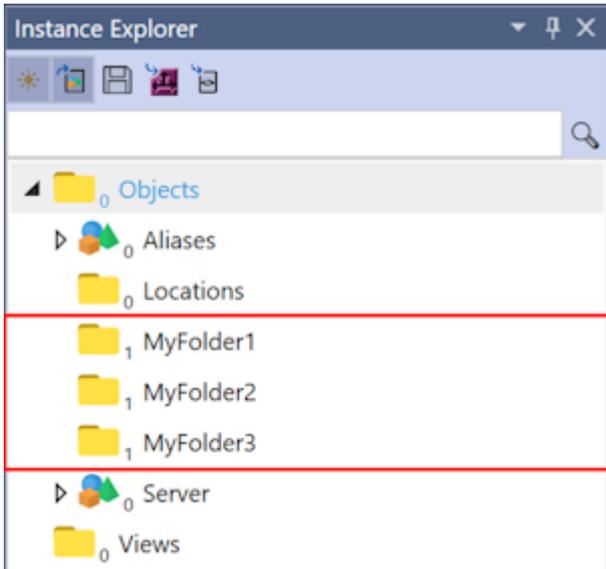
Gehen Sie zum Instance Explorer und fügen Sie drei Ordner hinzu, indem Sie den Node **Objects** markieren und im Kontextmenü **Add Folder** wählen. Die drei Ordner werden dann wie folgt angezeigt:



Benennen Sie die Ordner um, indem Sie auf jeden Ordner doppelklicken und das Attribut **Name** im Attribute Explorer entsprechend anpassen. Speichern Sie diese Änderung über die Schaltfläche **Save** in der Symbolleiste.



Führen Sie diese Prozedur für alle drei Ordner durch, so dass der Instance Explorer wie folgt aussieht:

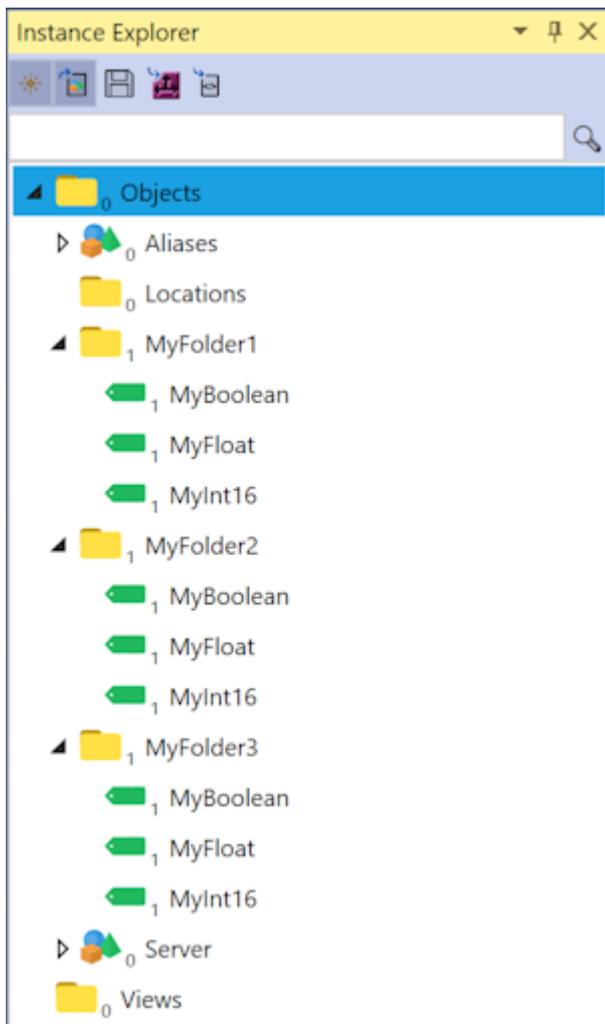


Im nächsten Schritt wollen wir einige Variablen zu den Ordnern hinzufügen.

Das Erstellen neuer Variablen- oder Objektinstanzen erfolgt in der Regel per Drag-and-Drop eines Datentyps im Type Explorer. Wählen Sie den ersten Ordner im Instance Explorer und navigieren Sie dann im Type Explorer zum gewünschten Datentyp. Drag-and-Drop den Datentyp auf den ausgewählten Ordner im Instance Explorer. Somit wird dort eine entsprechende Instanz dieses Datentyps angelegt. In unserem Beispiel verwenden wir den Datentyp Boolean.



Wir wiederholen diesen Schritt nun mit zwei weiteren Datentypen, z. B. Int16 und Float, und führen diesen Vorgang auch für die beiden anderen Ordner durch. Falls erforderlich, können Sie den Namen der hinzugefügten Variablen anpassen, indem Sie auf sie doppelklicken und sie dann im Attribute Explorer umbenennen.

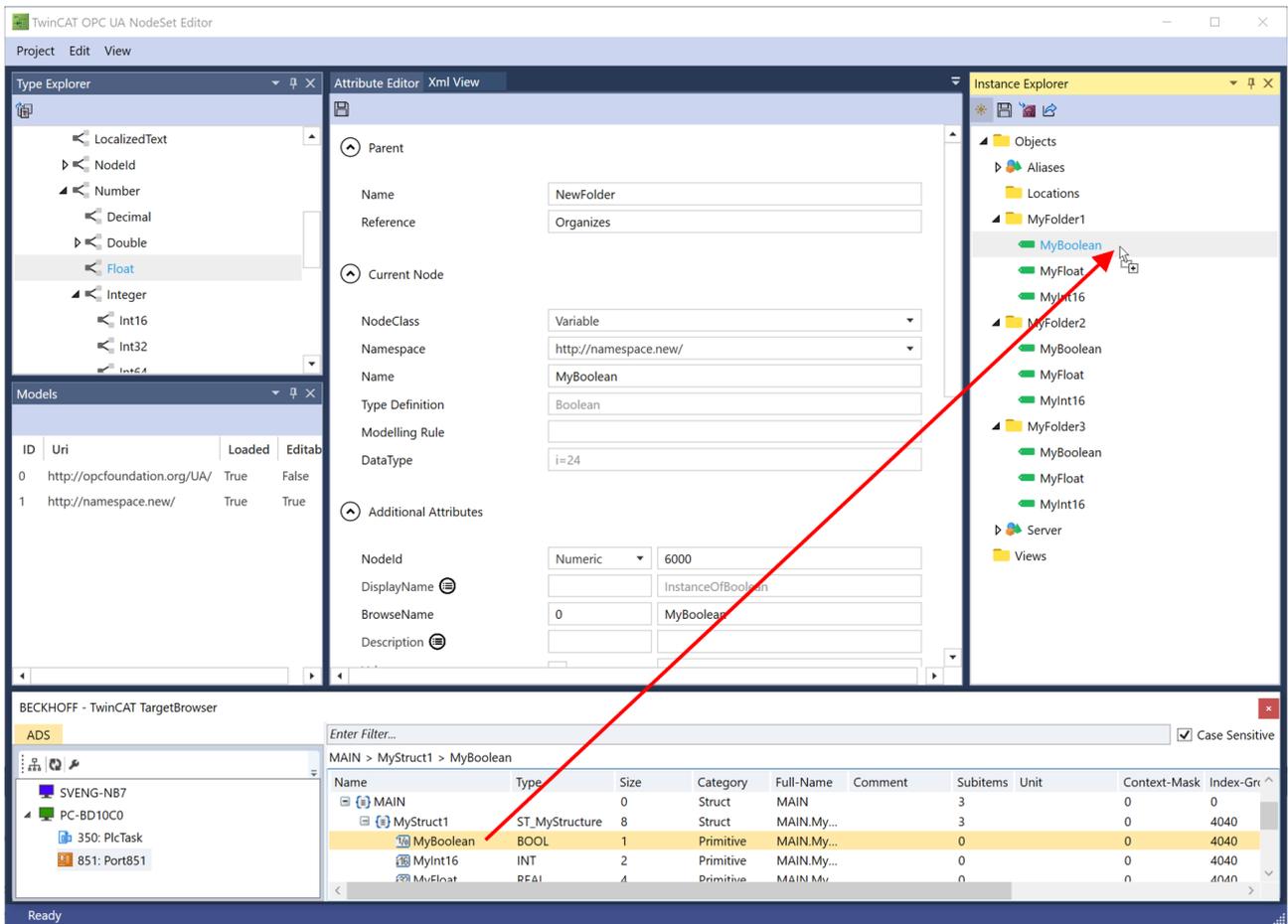


Verknüpfung mit SPS-Variablen

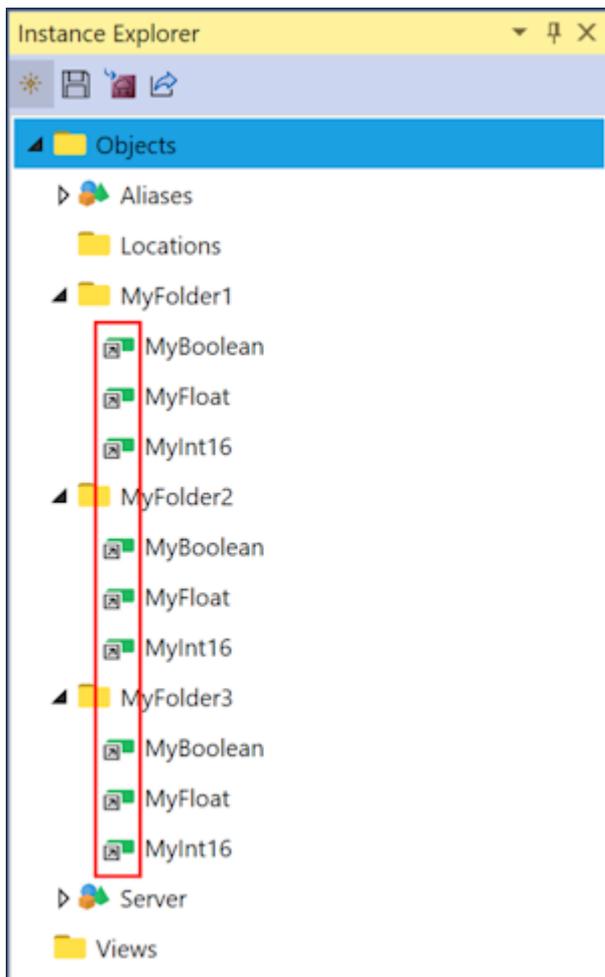
Wir wollen die nun erstellten Instanzen mit Variablen aus unserem vorbereiteten SPS-Projekt verknüpfen. Dazu kann der TwinCAT Target Browser verwendet werden, den Sie im Menü **View** öffnen können.

Navigieren Sie im Target Browser zum vorbereiteten SPS-Projekt und verknüpfen Sie die einzelnen Membervariablen der Datenstruktur ST_MyStructure mit den entsprechenden Instanzen, indem Sie sie per Drag-and-Drop auf den entsprechenden Node im Instance Explorer ziehen.

Die Verknüpfung ist dann in der Baumstruktur sichtbar und kann auch im Attribute Explorer überprüft werden.

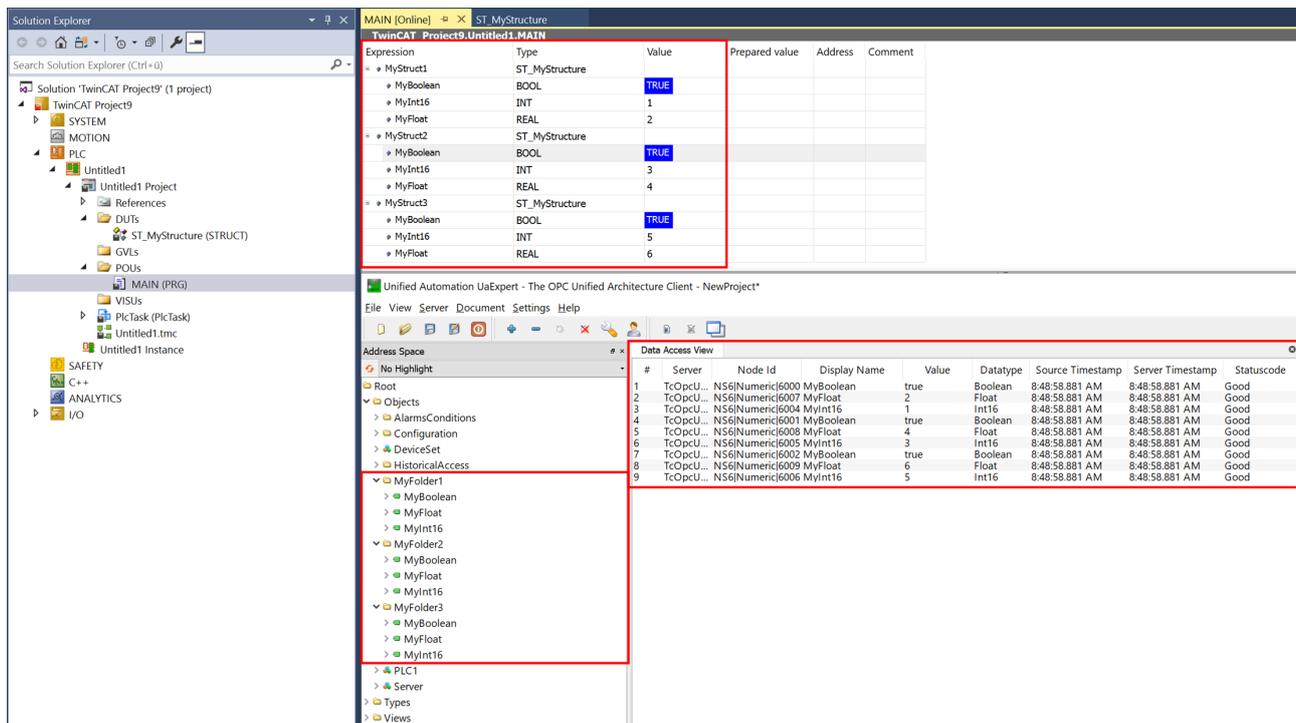


Führen Sie dies für alle Variablen durch.



Deployment des Projekts auf dem TwinCAT OPC UA Server

Um das Projekt auf den TwinCAT OPC UA Server bereitzustellen, klicken Sie auf die Schaltfläche **Export to TwinCAT OPC UA Server** in der Symbolleiste des Instance Explorers. Damit wird das Projekt-Nodeset auf den lokal installierten TwinCAT OPC UA Server exportiert und auch eine entsprechende Konfigurationsdatei für den Server erstellt. Bitte stellen Sie sicher, dass Sie den TwinCAT OPC UA Server nach dem Deployment des Nodesets neu starten, damit die Konfiguration geladen wird. Der Adressraum des Servers enthält dann die angelegten Nodes, die mit den entsprechenden SPS-Variablen verknüpft wurden.



4.1.2 Einfache Companion Specification

Dieser Dokumentationsartikel soll Ihnen einen schnellen Einstieg in die Verwendung dieses Produkts ermöglichen. Führen Sie nach erfolgreicher [Installation \[► 10\]](#) und [Lizenzierung \[► 10\]](#) die folgenden Schritte aus, um ein erstes Nodeset auf der Grundlage einer Companion Specification zu erstellen.

● OPC UA Base Nodeset

i Standardmäßig lädt der TwinCAT OPC UA Nodeset Editor automatisch das OPC UA Base Nodeset, das den Namespace <http://opcfoundation.org/UA/> definiert.

● TF6100 TC3 OPC UA Server

i Dieses Produkt arbeitet mit dem TwinCAT OPC UA Server zusammen, um ein erstelltes Nodeset auszuführen und es im Adressraum des Servers anzuzeigen. Bitte stellen Sie sicher, dass das Produkt TF6100 TC3 OPC UA Server auf Ihrem System installiert ist.

● Companion Specification Beispiel-Nodeset

i Als Beispiel für eine Companion Specification wird ein Nodeset verwendet, das einen ObjectType "MyObjectType" mit einigen Membervariablen definiert. Dieses Nodeset kann von unserem [TE6100 Samples Repository auf GitHub \[► 38\]](#) heruntergeladen werden.

Das folgende Quickstart-Tutorial führt Sie durch die folgenden Schritte:

- Vorbereitung eines SPS-Projekts
- Neues Projekt anlegen
- Nodeset der Companion Specification importieren
- Objektinstanzen erstellen
- Erstellte Instanzen mit SPS-Variablen verknüpfen
- Deployment des Projekts auf dem TwinCAT OPC UA Server

Vorbereitung eines SPS-Projekts

Erstellen Sie ein neues TwinCAT-Projekt und fügen Sie ein SPS-Projekt hinzu. Erstellen Sie die folgende Datenstruktur im SPS-Projekt und instanzieren Sie sie dreimal im MAIN-Programm.

```

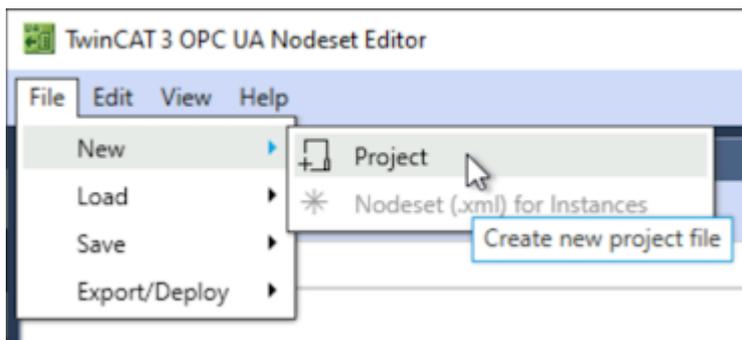
TYPE ST_MyStructure :
STRUCT
  MyBoolean : BOOL;
  MyInt16 : INT;
  MyFloat : REAL;
END_STRUCT
END_TYPE

PROGRAM MAIN
VAR
  MyStruct1 : ST_MyStructure;
  MyStruct2 : ST_MyStructure;
  MyStruct3 : ST_MyStructure;
END_VAR

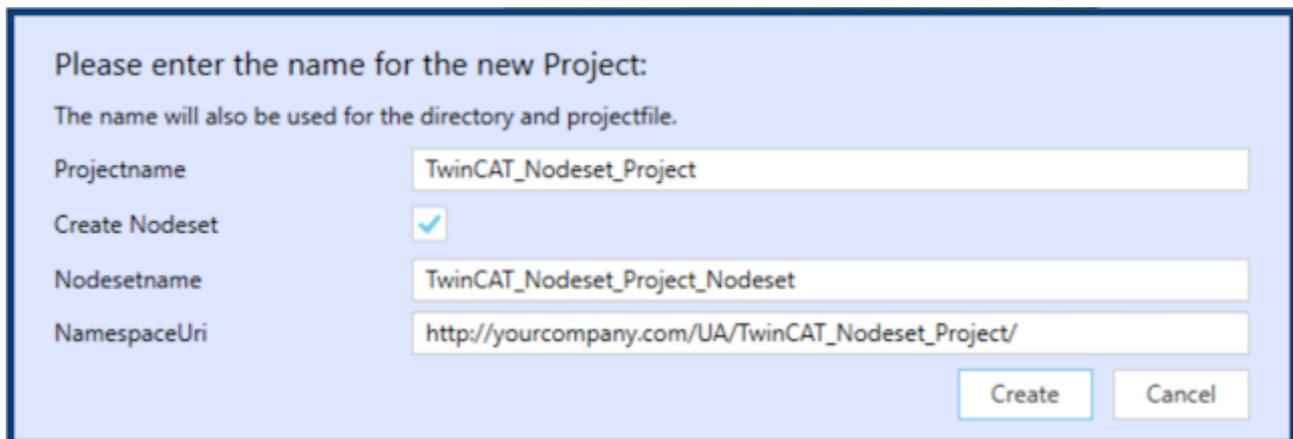
```

Neues Projekt anlegen

Klicken Sie im Menü auf **File > New > Project**, um ein neues Nodeset-Projekt zu erstellen.

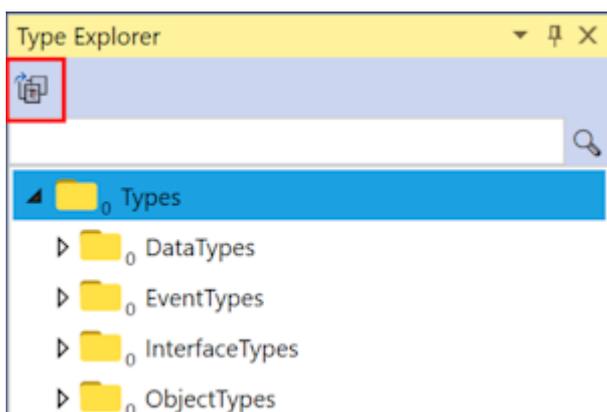


Im folgenden Auswahldialog können Sie den Namen und ein Zielverzeichnis für das neue Projekt angeben.

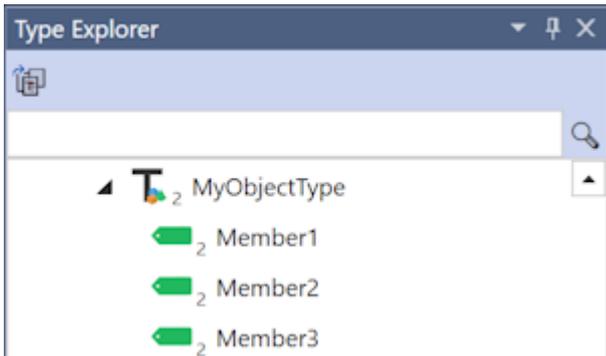


Nodeset der Companion Specification importieren

Importieren Sie die Datei "te6100_quickstart_companion_specification.xml" aus dem GitHub-Repository in den TwinCAT OPC UA Nodeset Editor, indem Sie im Type Explorer auf die Schaltfläche **LoadNodeset** klicken.

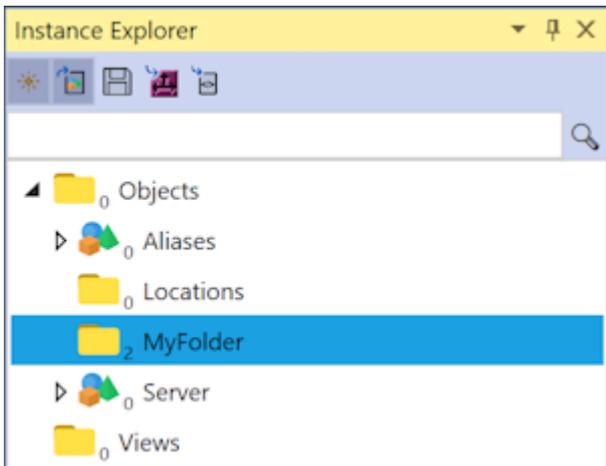


Der Import lädt das Typmodell aus dem Nodeset in den Type Explorer und zeigt es dort an. Das Typmodell enthält einen ObjectType "MyObjectType", den wir nun instanziiieren wollen. Sie können das Textfeld verwenden, um nach dem Namen des Typs zu suchen, oder zu diesem Typ unter dem Node ObjectTypes \ BaseObjectType scrollen.



Objektinstanzen erstellen

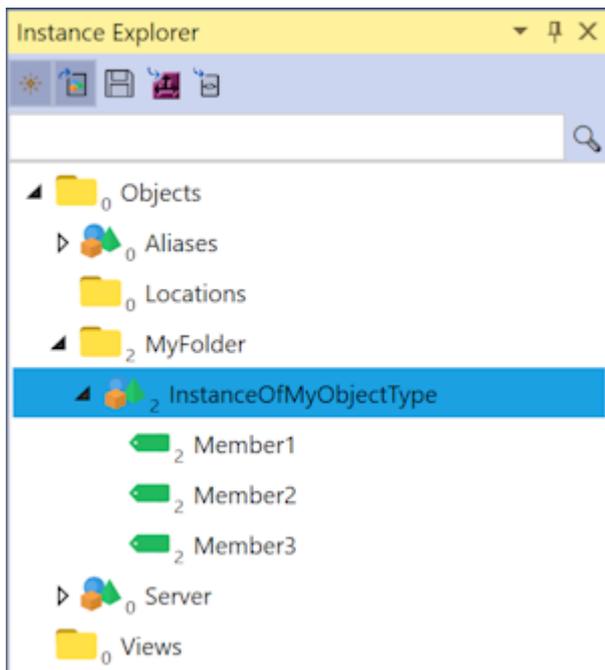
Verwenden Sie das Kontextmenü, um einen neuen Ordner im Instance Explorer zu erstellen. Sie können den Ordner umbenennen, indem Sie auf den Ordner doppelklicken und dann das Attribut "Name" im Attribute Explorer ändern. Speichern Sie diese Änderung dann über die Schaltfläche **Save** in der Symbolleiste des Attribute Explorers.



Dann den Typ "MyObjectType" aus dem Type Explorer per Drag-and-Drop in den erstellten Ordner im Instance Explorer ziehen.



Dadurch wird eine neue Objektinstanz des Typs "MyObjectType" erstellt und in dem erstellten Ordner abgelegt. Alle obligatorischen Membervariablen des Typs werden entsprechend in den Instance Explorer übertragen und dort als Node unterhalb der Objektinstanz angezeigt. Sie können diese Nodes nun mit SPS-Variablen verknüpfen.



Sie können die Instanz über den Attribute Editor umbenennen.

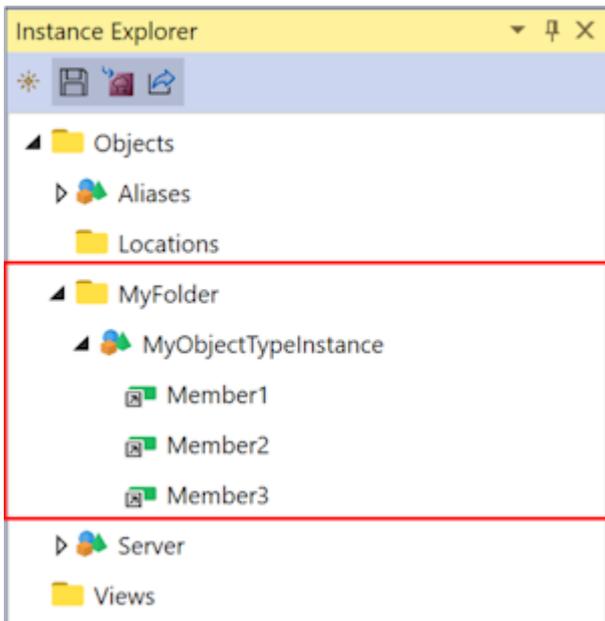
Erstellte Instanzen mit SPS-Variablen verknüpfen

Wir wollen nun die Membervariablen der erstellten Objektinstanz mit Variablen aus unserem vorbereiteten SPS-Projekt verknüpfen. Dazu kann der TwinCAT Target Browser verwendet werden, den Sie im Menü **View** öffnen können.

Navigieren Sie im Target Browser zum vorbereiteten SPS-Projekt und verknüpfen Sie die einzelnen Membervariablen der Datenstruktur mit den entsprechenden Membervariablen der Objektinstanz, indem Sie sie per Drag-and-Drop auf den entsprechenden Node im Instance Explorer ziehen. Die Datentypen sind hier wie folgt:

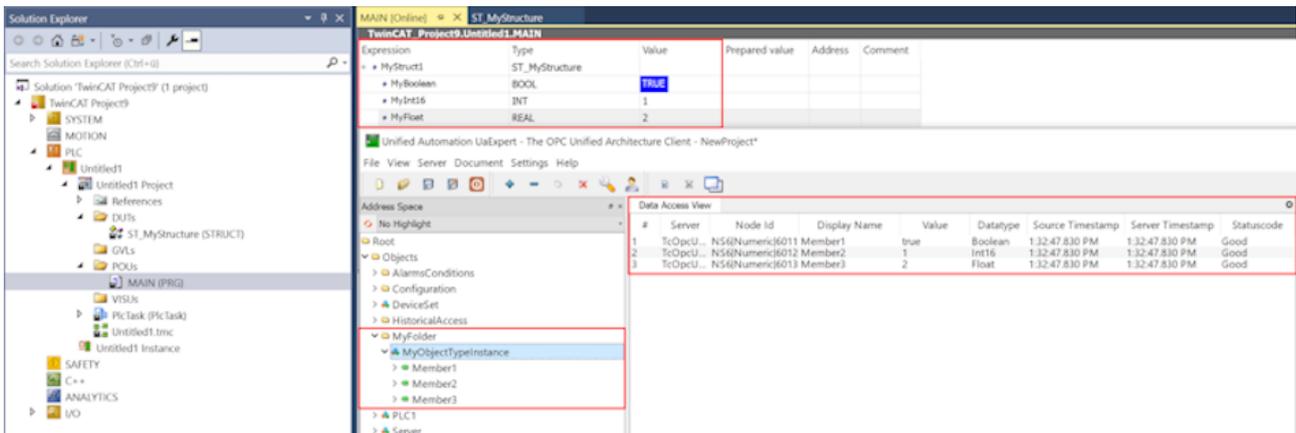
Membervariable	Datentype
Member1	Boolean (BOOL)
Member2	Int16 (INT)
Member3	Float (REAL)

Durch einen Doppelklick auf die Variable können diese Informationen auch über den Attribute Explorer abgerufen werden. Sobald alle Variablen verknüpft sind, kann im nächsten Schritt das Projekt auf dem TwinCAT OPC UA Server aktiviert werden.



Deployment des Projekts auf dem TwinCAT OPC UA Server

Um das Projekt auf den TwinCAT OPC UA Server bereitzustellen, klicken Sie auf die Schaltfläche **Export to TwinCAT OPC UA Server** in der Symbolleiste des Instance Explorers. Damit wird das Projekt-Nodeset auf den lokal installierten TwinCAT OPC UA Server exportiert und auch eine entsprechende Konfigurationsdatei für den Server erstellt. Bitte stellen Sie sicher, dass Sie den TwinCAT OPC UA Server nach dem Deployment des Nodesets neu starten, damit die Konfiguration geladen wird. Der Adressraum des Servers enthält dann die angelegten Nodes, die mit den entsprechenden SPS-Variablen verknüpft wurden.



4.2 Unterstützte Features

Mit Hilfe von OPC UA können komplexe Informationsmodelle erstellt werden. Diese Seite gibt einen Überblick über die derzeit verfügbaren Features des TwinCAT OPC UA Nodeset Editors. Zusätzliche Features werden im Laufe der Zeit als Updates zur Verfügung gestellt.

Feature	Beschreibung
Erstellung von verschachtelten Ordnern	Ermöglicht die Erstellung von (verschachtelten) Ordnern, z. B. zur Strukturierung von Instanzen.
Instanziierung von einfachen Datentypen	Erlaubt die Instanziierung von einfachen Datentypen aus dem OPC UA Base Nodeset. Die folgenden Datentypen werden unterstützt: Boolean, Sbyte, Byte, Int16, UInt16, Int32, UInt32, Int64, UInt64, Float, Double, DateTime, String, NodId, LocalizedText.
Instanziierung von Arrays einfacher Datentypen	Ermöglicht die Instanziierung von Arrays einfacher Datentypen aus dem OPC UA Base Nodeset. Die folgenden Datentypen werden unterstützt: Boolean, Sbyte, Byte, Int16, UInt16, Int32, UInt32, Int64, UInt64, Float, Double, DateTime, String, NodId, LocalizedText.
Instanziierung von Objekttypen	Ermöglicht die Instanziierung von Objekttypen aus einem importierten Nodeset, z. B. einer Companion Specification. Es können bestimmte Einschränkungen gelten, siehe unten.
Instanziierung von strukturierten Typen	Ermöglicht die Instanziierung von strukturierten Typen aus einem importierten Nodeset, z. B. einer Companion Specification. Es können bestimmte Einschränkungen gelten, siehe unten. Dies kann weitere Konfigurationsschritte erfordern, die in unserer Dokumentation über Mapping [► 32] näher erläutert werden.
Mapping von Nodes	Ermöglicht die Verknüpfung der erstellten Instanzen mit Variablen aus der SPS. Der TwinCAT Target Browser dient als Basis für das Feature.
Behandlung von obligatorischen/optionalen Nodes	Bei der Instanziierung eines Objekttyps werden alle Unterelemente automatisch hinzugefügt. Optionale Elemente müssen manuell entfernt werden.
Export zum TwinCAT OPC UA Server	Ermöglicht den Export des erstellten Nodeset-Projekts zum TwinCAT OPC UA Server.

Einschränkungen

Im Folgenden erhalten Sie eine Übersicht, welche Features derzeit noch nicht verfügbar sind. Zusätzliche Features werden im Laufe der Zeit durch Software-Updates für dieses Produkt hinzugefügt.

Feature	Beschreibung
Dynamische Array-Längen	Die Instanziierung von Arrays mit dynamischer Länge ist prinzipiell möglich, jedoch können solche Arrays derzeit nicht in der TwinCAT PLC oder dem TwinCAT OPC UA Server verwendet werden.
Methoden	Obwohl die Instanziierung von Methoden prinzipiell möglich ist, können solche Methoden derzeit nicht über den TwinCAT Target Browser verlinkt werden.
Erstellen/Ändern von Typen	Das Erstellen neuer Daten- oder Objekttypen oder das Ändern bestehender Typen wird derzeit nicht unterstützt.
Statische Werte für Nodes	Es ist derzeit nicht möglich, einen statischen Wert für einen Node zu setzen.
Referenzen	Derzeit können die Referenzen eines Nodes nicht geändert werden. Beim Hinzufügen von Ordnern oder Instanzen von Nodes werden diese mit Hilfe von Organizes -Referenzen referenziert.

4.3 Lizenz-Informationen

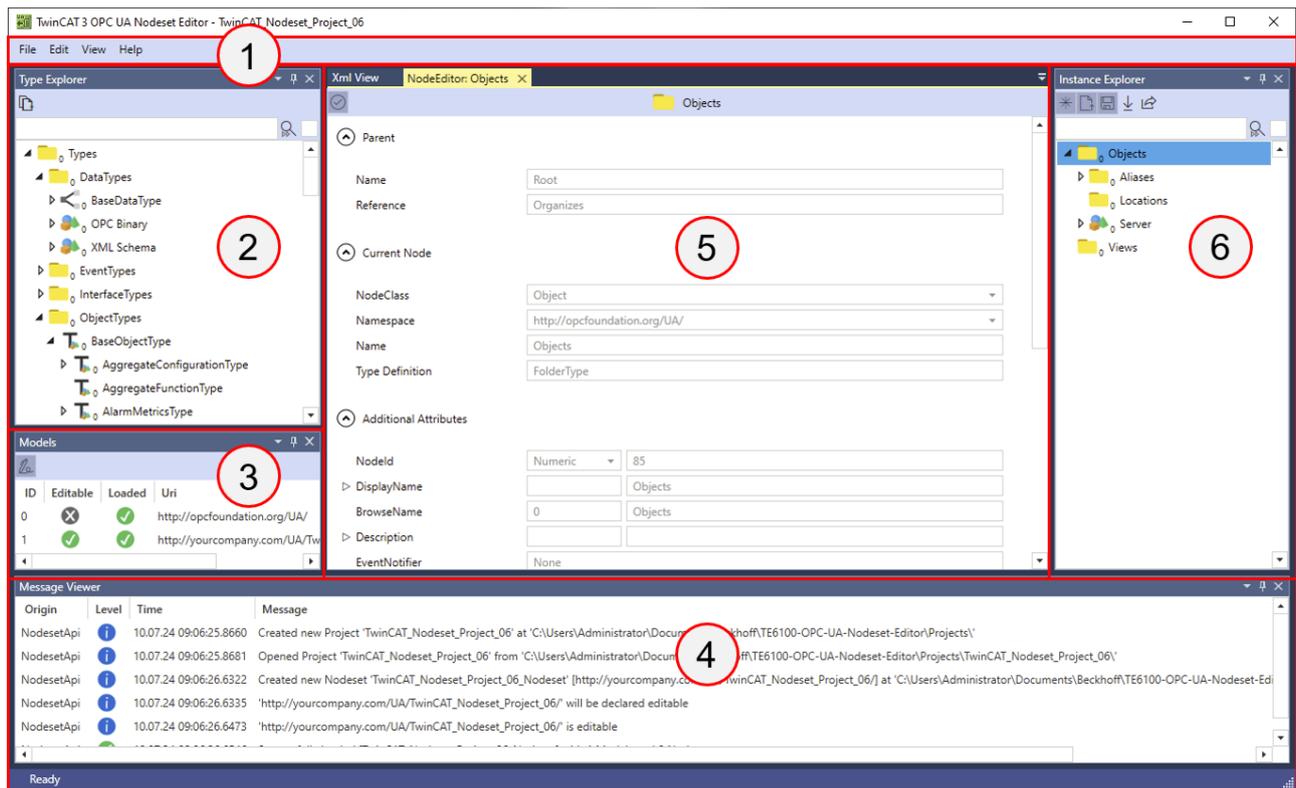
Es gibt verschiedene Lizenzen, die für den TwinCAT OPC UA Nodest Editor erworben werden können. Die folgende Tabelle gibt einen Überblick über die verschiedenen Lizenzen, die für dieses Produkt erhältlich sind.

Lizenz	Beschreibung
7-Tage-Testlizenz	Für dieses Produkt kann eine 7-Tage-Testlizenz verwendet werden. Sie bietet alle Funktionen der TE6100-Lizenz (siehe unten), erlaubt aber nicht die Speicherung oder Deployment des Projekts.
TE6100	Dies ist die Basislizenz, die für die Nutzung des Produkts erforderlich ist. Sie ermöglicht den Zugriff auf alle Produktfunktionen, mit Ausnahme derjenigen, die durch eine Zusatzlizenz definiert sind.
TE6101	Diese Lizenz ist eine optionale Zusatzlizenz, die es erlaubt, die automatische Codegenerierung zu nutzen.

4.4 Benutzeroberfläche

4.4.1 Überblick

Die Benutzeroberfläche ist in verschiedene Bereiche unterteilt, die im Folgenden näher beschrieben werden. Die folgende Abbildung veranschaulicht die verschiedenen Bereiche.

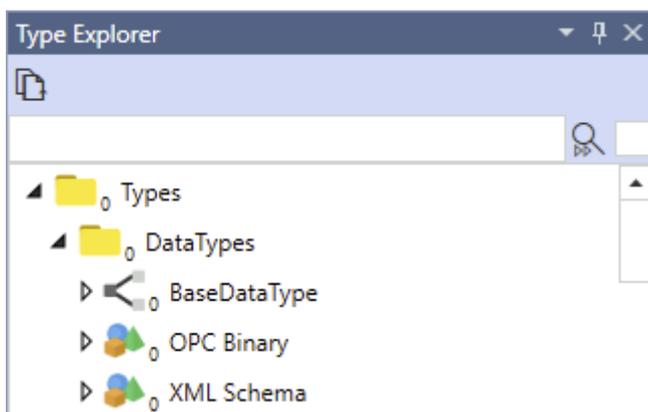


Bereich	Name	Beschreibung
1	Menü [▶ 28]	Zugriff auf Lade-/Speichermechanismen, sowie Einstellungen für die Benutzeroberfläche und den TwinCAT Target Browser.
2	Type Explorer [▶ 26]	Zeigt das Typsystem der aktuell geladenen Nodesets an. Ein Textfeld ermöglicht die Suche nach Nodes.
3	Models [▶ 28]	Zeigt die Namespaces der aktuell geladenen Nodesets an.
4	Message Viewer [▶ 29]	Enthält Protokollierungsinformationen über das aktuell geladene Projekt.
5	Attribute Editor [▶ 27]	Zeigt die Attribute des aktuell ausgewählten Nodes an. Hier können sowohl die Attribute der Nodes aus dem Type Explorer als auch aus dem Instance Explorer angezeigt werden.
6	Instance Explorer [▶ 27]	Zeigt die aktuell konfigurierten Instanzen an. Ein Textfeld ermöglicht die Suche nach Nodes.

Die einzelnen Bereiche sind in einem so genannten angedockten Fensterkonzept angeordnet und können je nach Benutzeranforderungen unterschiedlich gestaltet werden. Die Fenster des Type-, Attribute- und Instance-Explorers enthalten auch eine eigene Symbolleiste, mit der Sie Aktionen in diesem Bereich durchführen können. Diese Aktionen werden im Folgenden ausführlicher beschrieben.

4.4.2 Type Explorer

Der Type Explorer zeigt alle Datentypen aus den aktuell geladenen Nodesets an. Weitere Aktionen sind über die Symbolleiste verfügbar, und ein Textfeld ermöglicht die Suche nach Nodes.



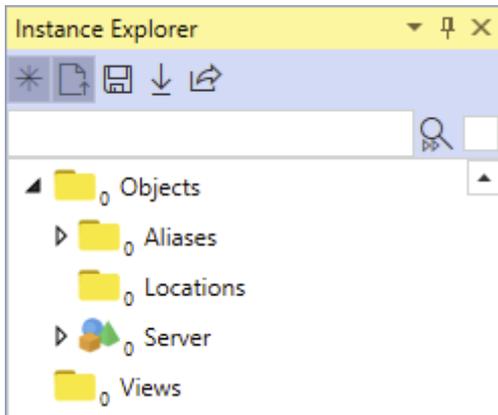
Die Zahlen auf jedem Node stellen den NamespaceIndex eines bestimmten Nodes dar und können zur leichteren Identifizierung von Nodes nützlich sein.

Die folgende Tabelle zeigt alle Aktionen, die Sie mit dem Type Explorer durchführen können.

Aktion	Beschreibung
Nodeset importieren	Ermöglicht das Hinzufügen weiterer Nodesets.
Drag&Drop	Indem Sie einen bestimmten Node per Drag&Drop in den Instance Explorer ziehen, können Sie eine Instanz dieses Typs erstellen.
Suchen	Sie können nach Nodes suchen, indem Sie den Namen oder die NodeId des Nodes in das entsprechende Textfeld eingeben. Weitere Suchbegriffe sind: ns=x (sucht nach Nodes in einem bestimmten NamespaceIndex)

4.4.3 Instance Explorer

Der Instance Explorer zeigt alle Instanzen in diesem Projekt an. Weitere Aktionen sind über die Symbolleiste verfügbar.

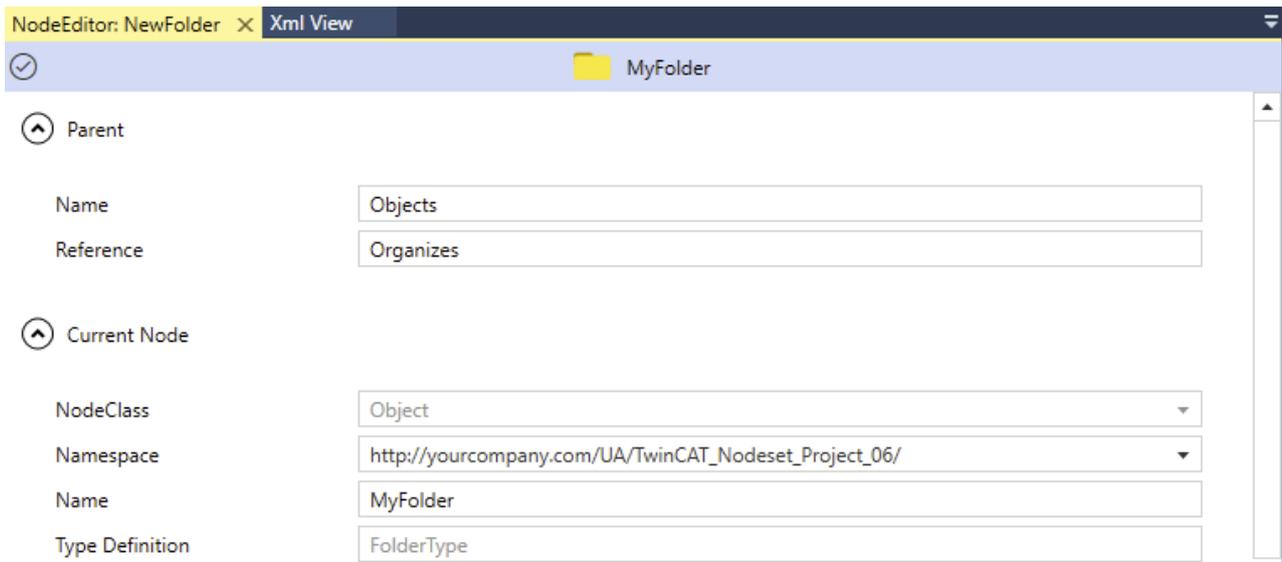


Die folgende Tabelle zeigt alle Aktionen, die Sie im Instance Explorer durchführen können.

Aktion	Beschreibung
Neues Nodeset erstellen	Erzeugt eine neue Instanz Nodeset.
Änderungen am Nodeset speichern	Speichert die aktuellen Änderungen auf der Festplatte.
Deployment auf TwinCAT OPC UA Server	Exportiert das aktuelle Projekt auf den TwinCAT OPC UA Server.
Nodesets exportieren	Exportiert den aktuellen Instanz Nodeset in eine Datei.
Suchen	Sie können nach Nodes suchen, indem Sie den Namen oder die NodeId des Nodes in das entsprechende Textfeld eingeben. Weitere Suchbegriffe sind: ns=x (sucht nach Nodes in einem bestimmten NameSpaceIndex) HasExtension (sucht nach Nodes mit einem Mapping zu einer SPS-Variablen) HasNoExtension (sucht nach Nodes ohne Mapping zu einer SPS-Variablen)
Kontextmenü (Rechtsklick) Node löschen	Entfernt den aktuell ausgewählten Node und die eng verbundenen Kindknoten. Bitte beachten Sie, dass andere Kindknoten (ohne ParentNodeId) nicht entfernt werden und als "verwaist" gelten, wenn sie keinen anderen Elternknoten haben.
Kontextmenü (Rechtsklick) Ordner anlegen	Erzeugt einen neuen Node vom Typ FolderType.

4.4.4 Attribute Editor

Der Attribute Explorer zeigt die Attribute des aktuell ausgewählten Nodes an. Es können sowohl die Attribute von Nodes aus dem Type Explorer als auch aus dem Instance Explorer angezeigt werden. Durch Doppelklick auf einen Node werden dessen Attribute geladen und entsprechend angezeigt. Weitere Aktionen sind über die Symbolleiste verfügbar.

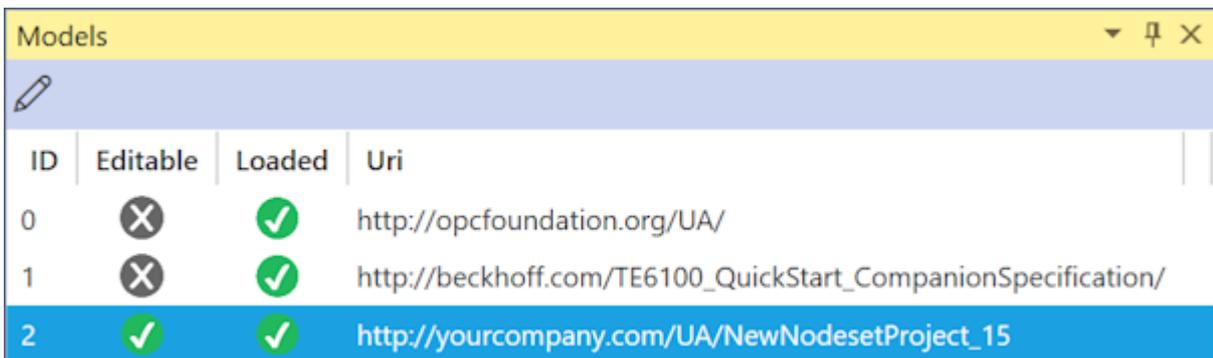


Die folgende Tabelle zeigt alle Aktionen, die im Attribute Editor durchgeführt werden können.

Aktion	Beschreibung
Attribute bearbeiten	Attribute des aktuell geöffneten Nodes bearbeiten.
Speichern	Übernimmt die Änderungen an den Attributen.
XML-Ansicht	Ermöglicht die Anzeige des aktuell ausgewählten Nodes als XML.

4.4.5 Models

Das Fenster **Informationsmodelle** zeigt alle aktuell geladenen Namespaces an - sowohl von allen geladenen Typ- als auch Instanz-Nodesets. In diesem Fenster können Sie auswählen, welchen Namespace Sie bei der Erstellung von Instanzen aus einem Daten- oder Objekttyp bearbeiten möchten. Der aktuell als bearbeitbar gekennzeichnete Namespace wird bei der Erstellung neuer Instanzen verwendet.



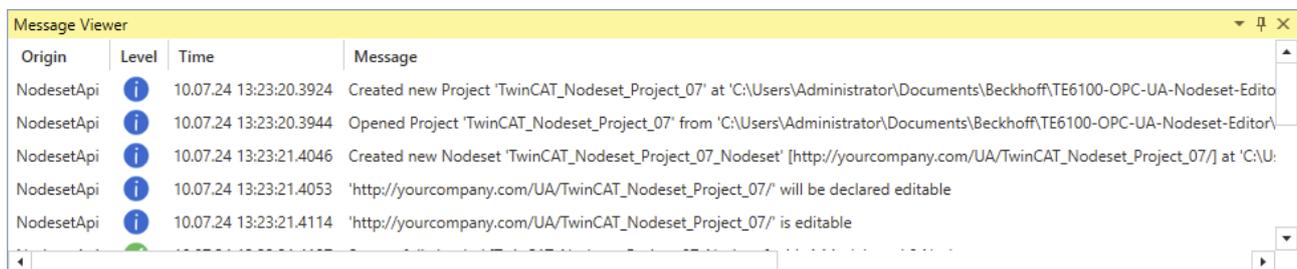
4.4.6 Menü

Das Menü enthält verschiedene Mechanismen, die sich auf das aktuell geöffnete Projekt beziehen. Die folgende Tabelle gibt weitere Informationen zu den einzelnen Menüeinträgen.

Menü	Unterpunkt	Beschreibung
File	New	Erzeugt ein neues Projekt.
File	Load	Lädt ein bestehendes Projekt.
File	Save	Speichert das aktuell geöffnete Projekt.
File	Export/Deploy	Exportiert das aktuell geöffnete Projekt in eine einzelne Nodeset-Datei. Zusätzlich kann das aktuell geöffnete Projekt auf den TwinCAT OPC UA Server bereitgestellt werden.
View	Theme	Ermöglicht den Wechsel zwischen verschiedenen visuellen Themen.
View	Target Browser	Öffnet den TwinCAT Target Browser.
View	Type Explorer	Blendet den Type Explorer ein/aus.
View	Models	Blendet das Fenster Models ein/aus.
View	Instance Explorer	Blendet den Instance Explorer ein/aus.
View	Message Viewer	Blendet den Message Viewer ein/aus.
View	XML View	Blendet die XML-Ansicht ein/aus.
View	Welcome	Blendet den Willkommensbildschirm ein/aus.
Help	Info	Zeigt Lizenzinformationen an.

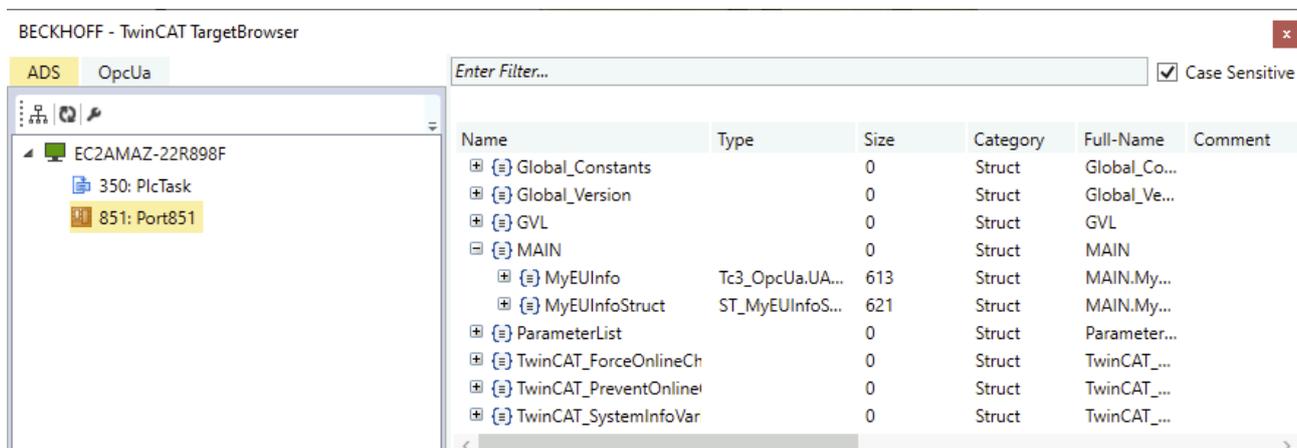
4.4.7 Message Viewer

Der Message Viewer enthält Protokollierungsinformationen zu Ihrem aktuell geöffneten Projekt. Die Meldungen werden in verschiedene Schweregrade und ihre Herkunft eingeteilt. Darüber hinaus hat jede Nachricht einen Zeitstempel, der angibt, wann die Nachricht ausgelöst wurde.



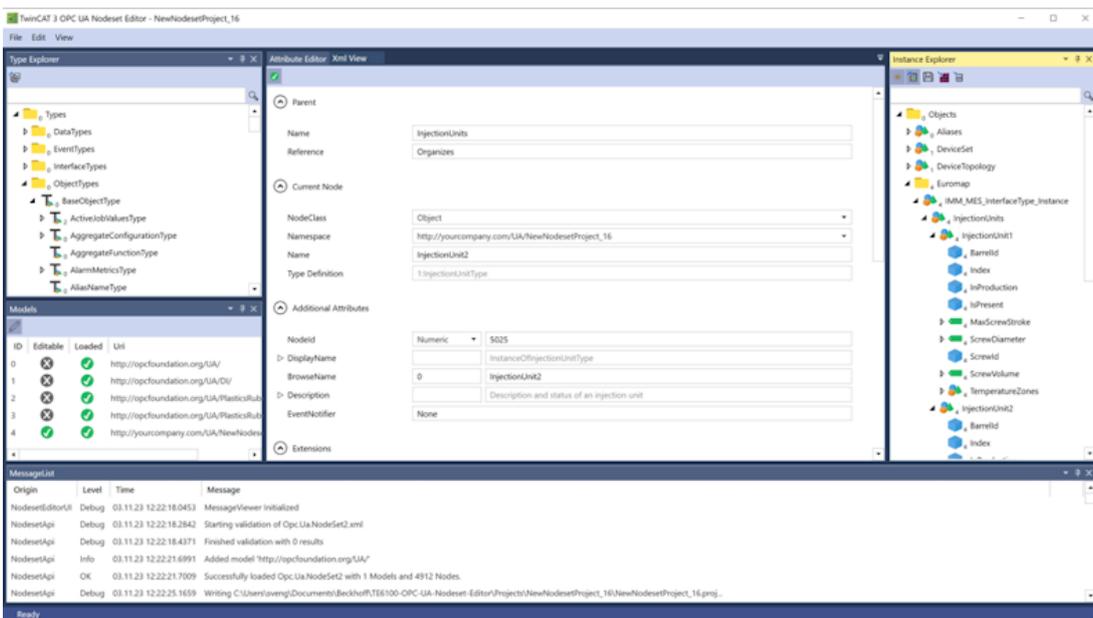
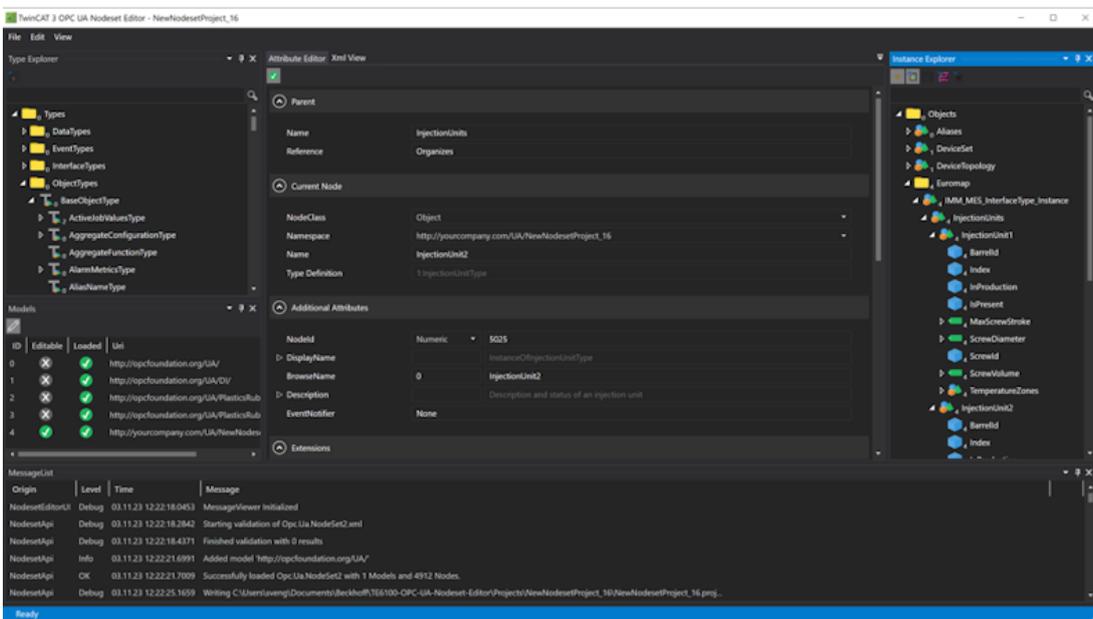
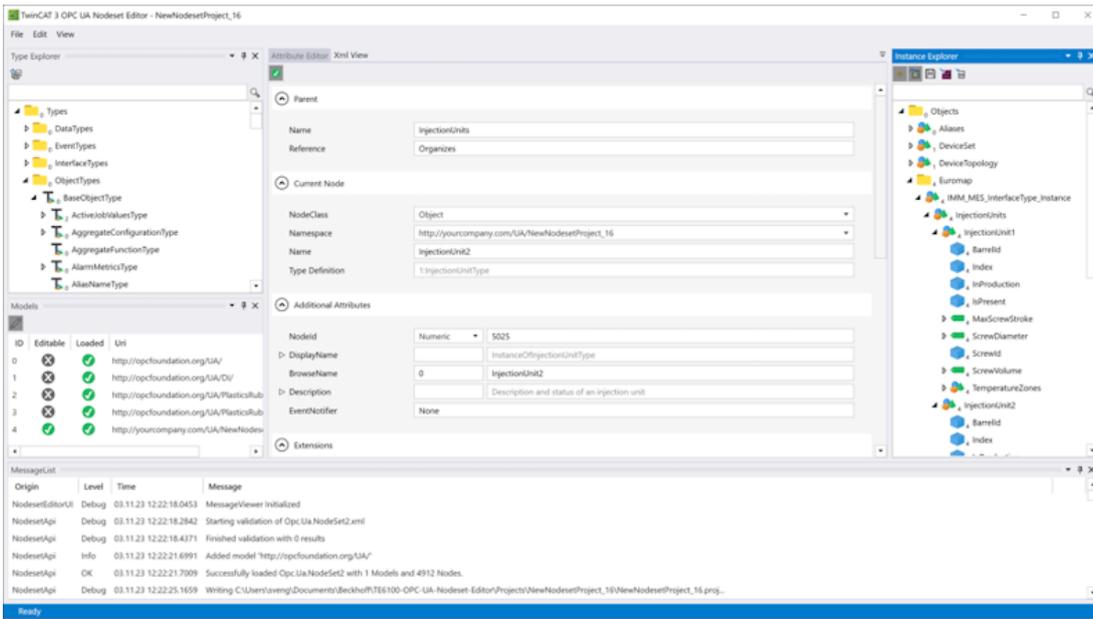
4.4.8 TwinCAT Target Browser

Der TwinCAT Target Browser kann über das Menü **View** geöffnet werden. Es ist derzeit als separates Fenster implementiert, wird aber in zukünftigen Versionen in die Hauptfensterstruktur (angedockt) integriert werden.



4.4.9 Themes

Der TwinCAT OPC UA Nodeset Editor unterstützt derzeit drei verschiedene Farbthemen: Hell, Dunkel und Blau (Standard). Sie können das Theme Ihrer Wahl über das Menü **View** auswählen. Die Einstellung kann in den Anwendungseinstellungen [[▶ 35](#)] beibehalten werden.



4.5 Mapping

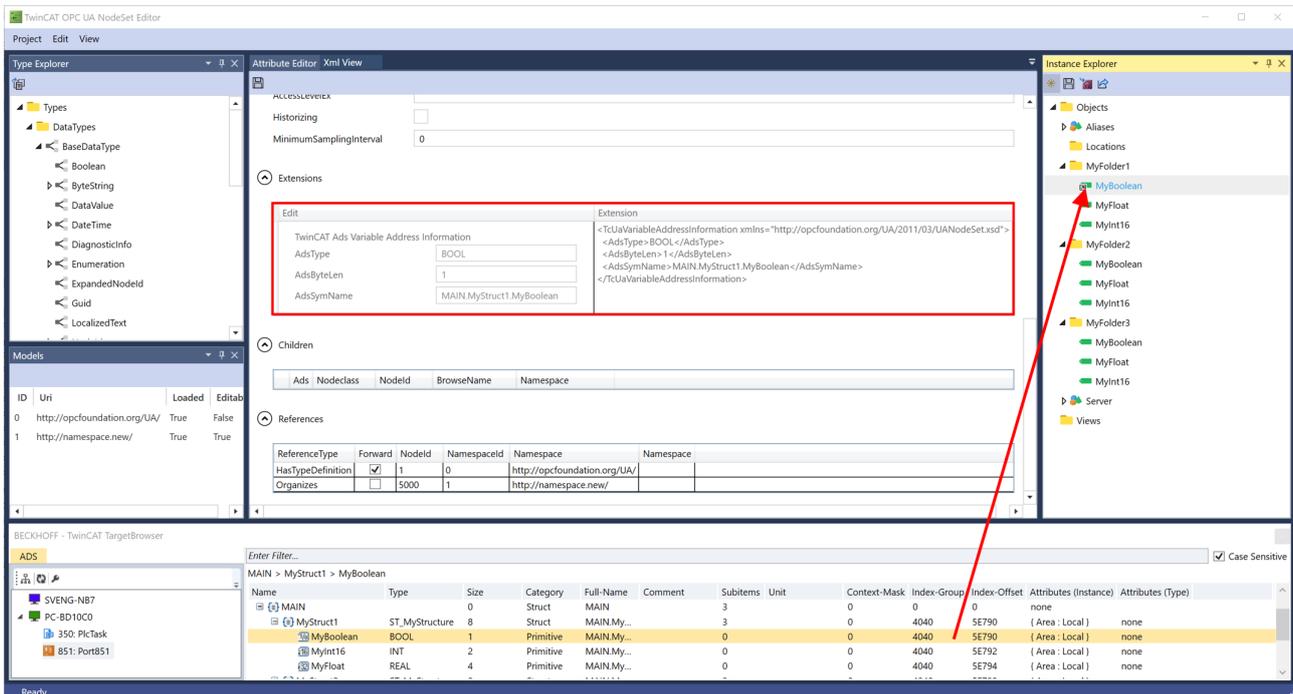
Für die Verknüpfung von OPC UA Nodes mit SPS-Symbolen steht der TwinCAT Target Browser zur Verfügung, der die Adressinformationen eines SPS-Symbols zu einem OPC UA Node hinzufügt. Das Noderset kann dann auf den TwinCAT OPC UA Server zur Verfügung gestellt werden.

Objects

Objects können nicht mit SPS-Symbolen verknüpft werden, da sie keinen "Wert" haben, der verknüpft werden könnte.

Variablen und Properties

Das Mapping von Variablen und Properties eines einfachen Datentyps (Int32, Double, Boolean, ...) kann mit Hilfe des TwinCAT Target Browsers durchgeführt werden. Bitte beachten Sie, dass der Datentyp der SPS-Variablen mit dem Datentyp des OPC UA Nodes "übereinstimmen" muss.



Die folgende Tabelle gibt einen Überblick über gängige, einfache Datentypen, wie sie im OPC UA Base Namespace <http://opcfoundation.org/UA/> definiert sind. Sie zeigt auch den entsprechenden IEC61131-Datentyp an.

OPC UA Datentyp	SPS-Datentyp
Boolean	BOOL
Sbyte	SINT
Byte	INT
Int16	INT
UInt16	UINT
Int32	DINT
UInt32	UDINT
Int64	LINT
UInt64	ULINT
Float	REAL
Double	LREAL
DateTime	DT
String	STRING

Strukturen

Strukturierte Typen können auch über den TwinCAT Target Browser mit einer SPS-Struktur verknüpft werden. Allerdings gibt es hier einige Besonderheiten zu beachten. Zum einen muss die Datenstruktur in der SPS zur OPC UA Datenstruktur "passen" und zum anderen benötigt die Datenstruktur in der SPS noch die OPC UA Typinformation als SPS-Attribut.

Die folgende OPC-UA-Struktur ist als Beispiel angegeben.

```
<UADataType NodeId="ns=1;i=3015" BrowseName="1:ModelDataType">
  <DisplayName>ModelDataType</DisplayName>
  <References>
    <Reference ReferenceType="HasEncoding">ns=1;i=5048</Reference>
    <Reference ReferenceType="HasEncoding">ns=1;i=5050</Reference>
    <Reference ReferenceType="HasEncoding">ns=1;i=5049</Reference>
    <Reference ReferenceType="HasSubtype" IsForward="false">i=22</Reference>
  </References>
  <Definition Name="1:ModelDataType">
    <Field DataType="Byte" Name="SwitchingObjectIdentification"/>
    <Field DataType="Byte" Name="SwitchingObjectNumber"/>
    <Field DataType="Date" Name="CurrentDate"/>
    <Field DataType="Int32" ValueRank="1" ArrayDimensions="1" Name="SwitchingTimes"/>
  </Definition>
</UADataType>
```

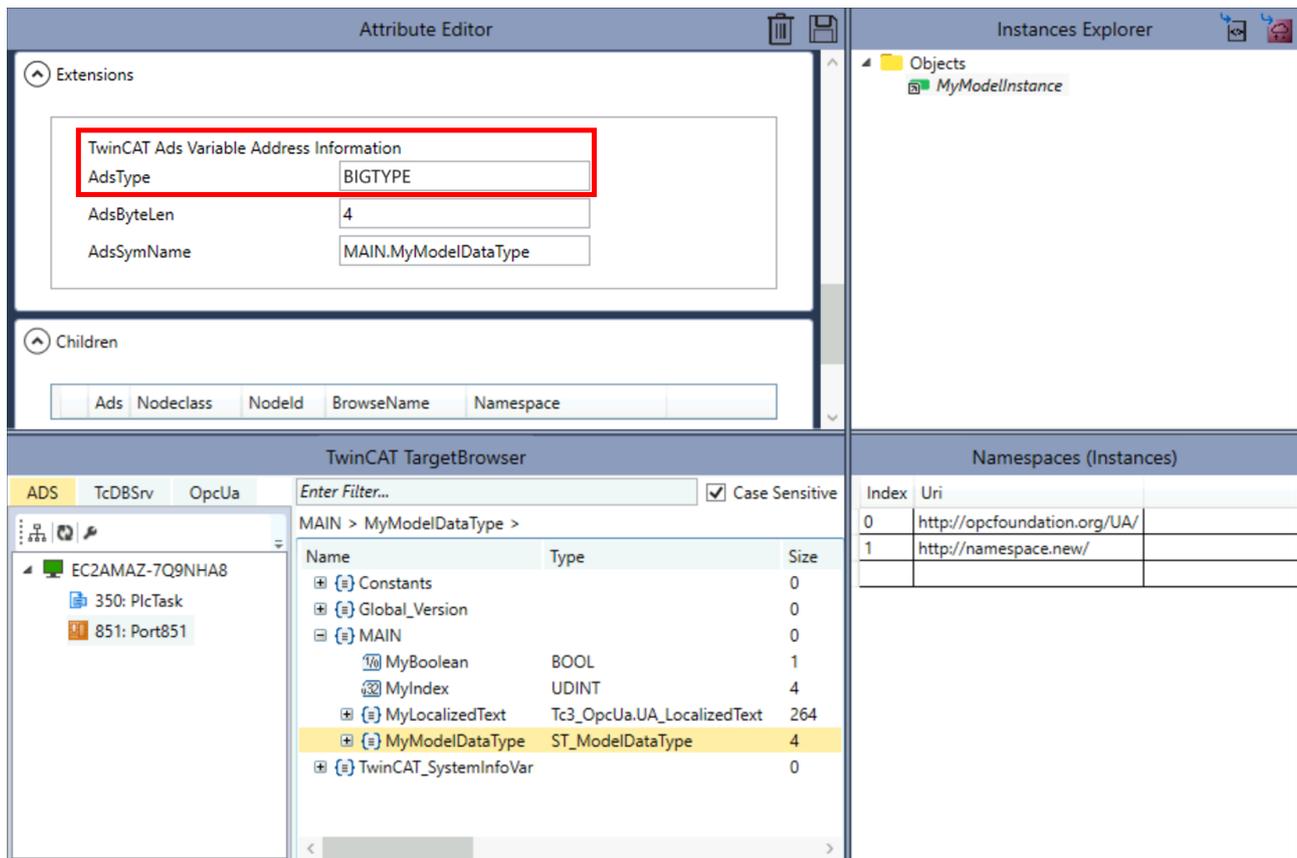
Die Struktur in der SPS muss dann wie folgt aufgebaut sein:

```
PROGRAM MAIN
VAR
  MyModel : ST_ModelDataType;
END_VAR

{attribute 'OPC.UA.DA' := '1'}
{attribute 'OPC.UA.DA.StructuredType' := '1'}
{attribute 'pack_mode' := '1'}
{attribute 'OPC.UA.AdditionalStructuredType.NamespaceName' := 'nameSpaceNameNodeset'}
{attribute 'OPC.UA.AdditionalStructuredType.Id' := 'i=3015'}
TYPE ST_ModelDataType :
STRUCT
  SwitchingObjectIdentification : BYTE;
  SwitchingObjectNumber : BYTE;
  CurrentDate : DT;
  SwitchingTimes : ARRAY[0..2] OF DINT;
END_STRUCT
END_TYPE
```

Die zusätzlichen Attribute OPC.UA.AdditionalStructuredType.NamespaceName und OPC.UA.AdditionalStructuredType.id sind notwendig, damit der TwinCAT OPC UA Server die Typinformationen zuordnen kann. Sie stellen den Namespace und die NodeID des Datentyps des StructuredType dar.

Der Link im TwinCAT Target Browser kann dann auf die Strukturinstanz MyTimeModel gesetzt werden. Bitte stellen Sie anschließend sicher, dass der AdsType "BIGINT" in der TcAdsExtension am OPC UA Node eingetragen ist.



4.6 Applikationsverzeichnisse

Diese Anwendung verwendet verschiedene Verzeichnisse, um relevante Informationen zu speichern, z. B. Konfigurationsdateien, Nodeset-Bibliotheken und Dateien im Zusammenhang mit dem aktuellen Projekt.

Installationsordner

Der Installationsordner dieser Anwendung befindet sich unter dem TwinCAT-Installationsverzeichnis.

```
%TcInstallDir%\Functions\TE6100-OPC-UA-Nodeset-Editor
```

Konfigurationsdateien

Diese Anwendung verwendet verschiedene Konfigurationsdateien, die sich in einem Unterordner des ProgramData-Verzeichnisses befinden:

```
%ProgramData%\Beckhoff\TE6100-OPC-UA-Nodeset-Editor
```

Bibliotheken

Es gibt ein zentrales Verzeichnis, in dem Sie Nodeset-Dateien ablegen können, die Sie in Ihren Projekten immer wieder als Bibliotheken verwenden müssen. Jedes Mal, wenn Sie eine Nodeset-Datei in Ihr aktuelles Projekt importieren möchten, öffnet der TwinCAT OPC UA Nodeset Editor dieses "Library"-Verzeichnis, das sich in einem Unterordner des ProgramData-Verzeichnisses befindet:

```
%ProgramData%\Beckhoff\TE6100-OPC-UA-Nodeset-Editor\Library
```

Projekte

Standardmäßig werden neue Projekte in einem Unterordner des Home-Verzeichnisses des Benutzers erstellt:

```
%UserProfile%\Documents\Beckhoff\TE6100-OPC-UA-Nodeset-Editor\Projects
```

Logdateien

Der TwinCAT OPC UA Nodeset Editor erstellt Logdateien, die nützliche Informationen enthalten können, falls etwas schief geht. Diese Logdateien befinden sich in einem Unterordner des ProgramData-Verzeichnisses:

```
%ProgramData%\Beckhoff\TE6100-OPC-UA-Nodeset-Editor\Log
```

4.7 Applikations- und Projekteinstellungen

Die Anwendung selbst und jedes Projekt enthält XML-basierte Konfigurationsdateien, die verschiedene Einstellungen enthalten, die das Verhalten der Anwendung beeinflussen. Solche Einstellungen werden in den folgenden Kapiteln beschrieben.

Applikationseinstellungen

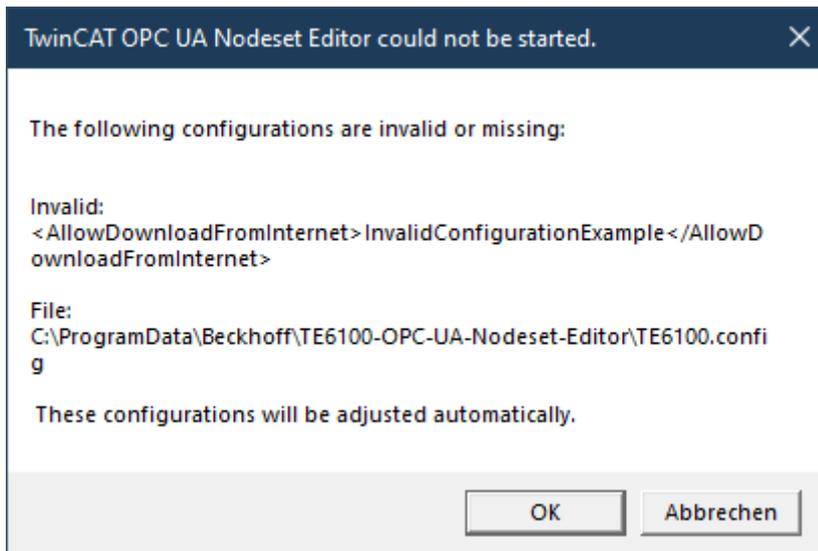
Im Verzeichnis der Konfigurationsdateien (siehe [Applikationsverzeichnisse \[▶ 34\]](#)) befindet sich die Datei TE6100.config, die Konfigurationsparameter für die Anwendung TwinCAT OPC UA Nodeset Editor enthält. Sie können diese Datei mit einem Texteditor Ihrer Wahl öffnen. Folgende Parameter stehen zur Verfügung.

Sektion	Parameter	Beschreibung
General	AllowDownloadFromInternet	Der Nodeset Editor enthält bereits eine bestimmte Version des Base-Nodesets. Wenn Sie diese Einstellung auf "true" setzen, wird stattdessen die aktuellste Version von der OPC Foundation heruntergeladen.
General	DefaultLocalOpcUaServerDir	Default-Speicherort der TwinCAT OPC UA Server Installation. Dieser Pfad wird verwendet, wenn ein Nodeset auf dem TwinCAT OPC UA Server bereitgestellt wird.
General	ProjekteDir	Standard-Basisverzeichnis für neue Projekte.
General	ShowNamespaceIndexInTree	Aktiviert/deaktiviert die Anzeige des NamespaceIndex als Zahl im Type- und Instance-Explorer.
General	Theme	Legt das Standard-Theme für die Anwendung fest.
Nodesets	CoreNodesetFilePath	Pfad zum OPC UA Basis Nodeset (wird für alle Basisdaten und Objekttypen verwendet). Dieses Nodeset wird standardmäßig für jedes neue Projekt geladen.
Nodesets	LibraryDir	Standard-Basisverzeichnis für Typ-Nodesets. Dieses Verzeichnis wird standardmäßig geöffnet, wenn Sie Typ-Nodesets importieren.
Nodesets	OrderAttributesOnExport	Wenn diese Option auf true gesetzt ist, werden die Attribute aller XML-Elemente in alphabetischer Reihenfolge sortiert, wobei gemeinsame Attribute an erster Stelle stehen (z. B. Nodeld und BrowseName) und andere in alphabetischer Reihenfolge folgen. Tipp: Wenn Sie ein von dieser Software gespeichertes Nodeset mit einem anderen Nodeset vergleichen wollen (z. B. mit "git diff"), kann es sinnvoll sein, dies auf false zu setzen, um unnötige Änderungen zu vermeiden.
Nodesets	OrderNodesOnExport	Bei der Einstellung "true" werden die Nodes einheitlich nach NamespaceId und Nodeld und Unterelementen (DisplayName usw.) geordnet. Tipp: Wenn Sie ein von dieser Software gespeichertes Nodeset mit einem anderen Nodeset vergleichen wollen (z. B. mit "git diff"), kann es sinnvoll sein, dies auf false zu setzen, um unnötige Änderungen zu vermeiden.
Nodesets	XmlSchemaDateipfad	Pfad zum OPC UA XmlSchemaFile, das für die (Syntax-) Validierung von Nodesets verwendet wird.

Projekteinstellungen

Parameter	Beschreibung
LocalOpcUaServer	Default-Speicherort der TwinCAT OPC UA Server Installation. Dieser Pfad wird verwendet, wenn ein Nodeset auf dem TwinCAT OPC UA Server bereitgestellt wird. Bei der Erstellung eines neuen Projekts wird dieser Pfad aus dem entsprechenden Parameter für die Applikationseinstellungen (siehe oben) übernommen, aber Sie können ihn projektspezifisch überschreiben.
Nodesets	Enthält eine Liste aller Nodesets, die zu diesem Projekt gehören, einschließlich Pfad, NamespaceName, Version und PublicationDate.

Wenn eine ungültige Konfiguration erkannt wird, erscheint ein Dialog, der Sie über den ungültigen Parameter informiert.

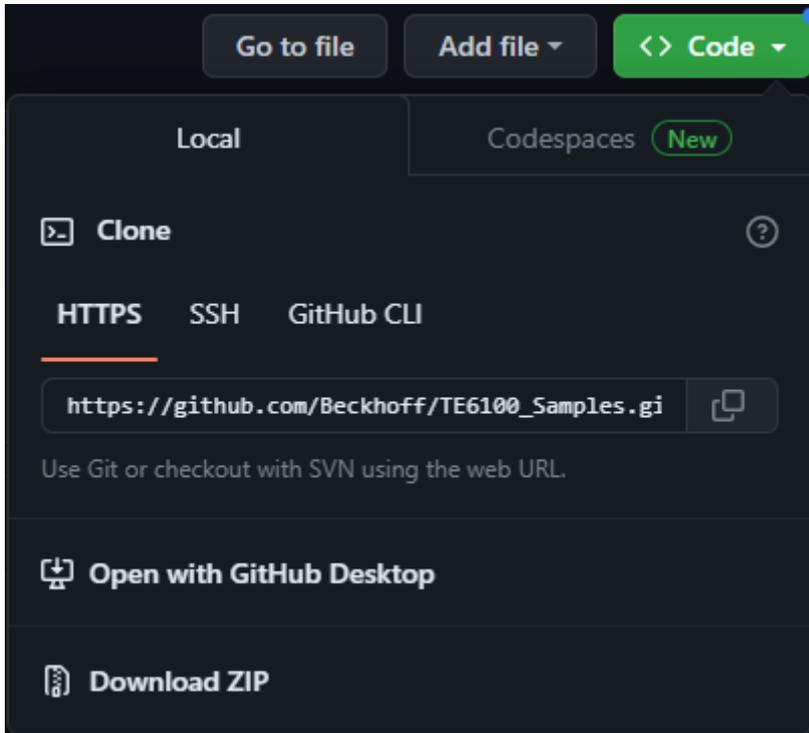


Die Software wird versuchen, ungültige Konfigurationen automatisch zu korrigieren, wenn Sie auf **OK** klicken. Dies geschieht, indem die Konfiguration auf den Standardwert zurückgesetzt wird. Beispiel:

```
] <General>  
<AllowDownloadFromInternet>false</AllowDownloadFromInternet>
```

5 Samples

Beispielcode und -konfigurationen für dieses Produkt können über das entsprechende Repository auf GitHub bezogen werden: https://github.com/Beckhoff/TE6100_Samples. Sie haben dort die Möglichkeit, das Repository zu clonen oder ein ZIP-File mit dem Sample herunterzuladen.



Wir empfehlen Ihnen auch unsere [Quickstart Tutorials \[▶_13\]](#), die einen geführten Durchgang durch verschiedene Anwendungsfälle der Konfiguration bieten.

6 Anhang

6.1 Begriffe

Nodeset

Ein Nodeset ist eine Datei, die ein OPC UA Informationsmodell formal beschreibt. Nodeset-Dateien sind in der Regel XML-basiert und müssen entsprechend einer XML-Schema-Definition strukturiert sein. Nodesets werden für die Verarbeitung von Informationsmodellen mit dem Computer verwendet. Ein Nodeset definiert Nodes, ihre Attribute und Referenzen. Eine Referenz kann sowohl auf Nodes im gleichen Dokument als auch auf Nodes außerhalb der Datei verweisen. OPC UA selbst definiert einige zentrale Informationsmodelle mit entsprechenden Nodesets, zum Beispiel:

Name	Namespace
Core („Base“)	http://opcfoundation.org/UA
DI	http://opcfoundation.org/UA/DI
GDS	http://opcfoundation.org/UA/GDS

Besonders erwähnenswert ist das Modell "Core", dessen Nodeset umgangssprachlich auch als "Base"-Nodeset bezeichnet wird. Das Core-Modell dient in der Regel als Basis für weitere Informationsmodelle und beschreibt die grundlegenden Datentypen, die in OPC UA verfügbar sind.

Companion Specifications

Obwohl die bestehenden OPC UA Informationsmodelle bereits in vielen Branchen anwendbar sind, ist es möglich, eigene Informationsmodelle zu erstellen, um spezifische Anforderungen zu erfüllen. Zu diesem Zweck schließen sich Unternehmen und Organisationen zusammen, um branchenspezifische Modelle, so genannte "Companion Specifications", zu definieren. Solche Companion Specifications gibt es zum Beispiel für die Verpackungsindustrie (PackML) oder für die Kunststoffverarbeitung (EUROMAP).

Weitere Informationen zu diesem Thema sind auf der [Website der OPC Foundation](#) zu finden.

Node

Ein Node ist ein wesentlicher Teil des Adressraums eines Servers und stellt ein Objekt, eine Variable oder eine Methode dar. Auch Daten und Objekttypen werden im Adressraum des Servers als Node dargestellt.

6.2 Troubleshooting

Verhalten	Beschreibung
Nodeset kann nicht zum TwinCAT OPC UA Server exportiert werden. Das Log zeigt einen Fehler "Failed to write ...".	Bitte stellen Sie sicher, dass Sie Schreibrechte auf das Installationsverzeichnis des TwinCAT OPC UA Servers haben. Wenn der TwinCAT OPC UA Server in das Verzeichnis %ProgramFiles% installiert wurde, stellen Sie bitte sicher, dass Sie den TwinCAT OPC UA Nodeset Editor als Administrator mit der Option "Als Administrator ausführen" starten.
Ich habe mein Nodeset auf dem TwinCAT OPC UA Server bereitgestellt, aber der Server-Namespace zeigt meine konfigurierten Nodes nicht an.	Bitte stellen Sie sicher, dass Sie den korrekten Installationspfad des TwinCAT OPC UA Servers in den Applikationseinstellungen [► 35] konfiguriert haben. Auf einem 32 Bit-System wird das Win32-Installationsverzeichnis verwendet, während auf einem 64 Bit-System das Win64-Verzeichnis zum Ausführen der Datei TcOpcUaServer.exe verwendet wird. Sie können auch über den Windows Task Manager überprüfen, welches Verzeichnis verwendet wird, indem Sie zum Prozess TcOpcUaServer navigieren und im Kontextmenü Open file location auswählen.

7 Support und Service

Beckhoff und seine weltweiten Partnerfirmen bieten einen umfassenden Support und Service, der eine schnelle und kompetente Unterstützung bei allen Fragen zu Beckhoff Produkten und Systemlösungen zur Verfügung stellt.

Downloadfinder

Unser [Downloadfinder](#) beinhaltet alle Dateien, die wir Ihnen zum Herunterladen anbieten. Sie finden dort Applikationsberichte, technische Dokumentationen, technische Zeichnungen, Konfigurationsdateien und vieles mehr.

Die Downloads sind in verschiedenen Formaten erhältlich.

Beckhoff Niederlassungen und Vertretungen

Wenden Sie sich bitte an Ihre Beckhoff Niederlassung oder Ihre Vertretung für den [lokalen Support und Service](#) zu Beckhoff Produkten!

Die Adressen der weltweiten Beckhoff Niederlassungen und Vertretungen entnehmen Sie bitte unserer Internetseite: www.beckhoff.com

Dort finden Sie auch weitere Dokumentationen zu Beckhoff Komponenten.

Beckhoff Support

Der Support bietet Ihnen einen umfangreichen technischen Support, der Sie nicht nur bei dem Einsatz einzelner Beckhoff Produkte, sondern auch bei weiteren umfassenden Dienstleistungen unterstützt:

- Support
- Planung, Programmierung und Inbetriebnahme komplexer Automatisierungssysteme
- umfangreiches Schulungsprogramm für Beckhoff Systemkomponenten

Hotline: +49 5246 963-157
E-Mail: support@beckhoff.com

Beckhoff Service

Das Beckhoff Service-Center unterstützt Sie rund um den After-Sales-Service:

- Vor-Ort-Service
- Reparaturservice
- Ersatzteilservice
- Hotline-Service

Hotline: +49 5246 963-460
E-Mail: service@beckhoff.com

Beckhoff Unternehmenszentrale

Beckhoff Automation GmbH & Co. KG

Hülshorstweg 20
33415 Verl
Deutschland

Telefon: +49 5246 963-0
E-Mail: info@beckhoff.com
Internet: www.beckhoff.com

Mehr Informationen:
www.beckhoff.com/te6100

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

