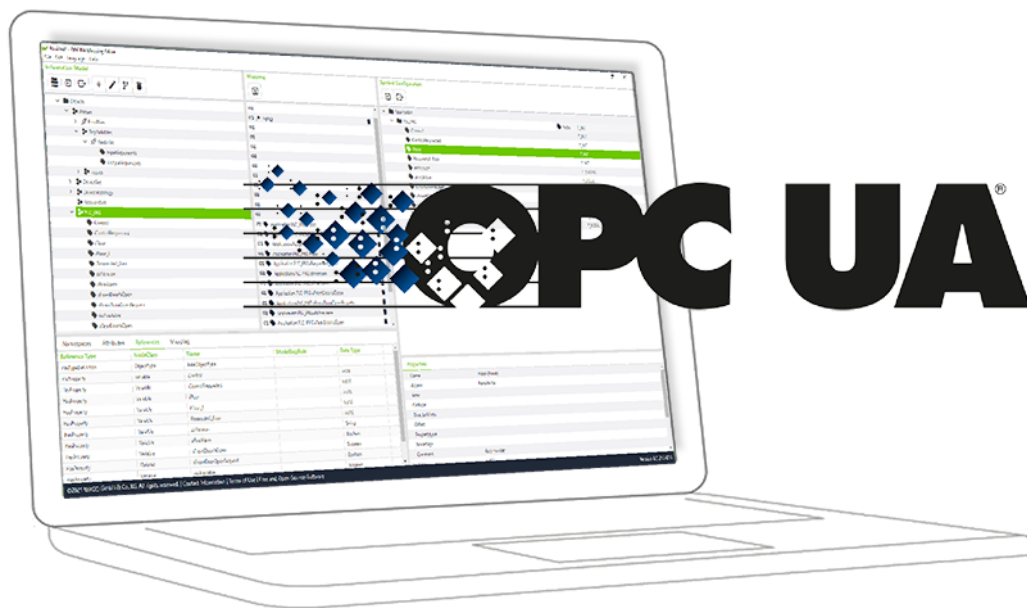


# WAGO Software

## OPC UA Mapping Editor



© 2022 WAGO GmbH & Co. KG  
Alle Rechte vorbehalten.

**WAGO GmbH & Co. KG**

Hansastraße 27  
D-32423 Minden

Tel: +49 (0) 571/887 – 0  
Fax: +49 (0) 571/887 – 844 169  
E-Mail: ✉ [info@wago.com](mailto:info@wago.com)  
Web: 🌐 [www.wago.com](http://www.wago.com)

**Technischer Support**

Tel: +49 (0) 571/887 – 44555  
Fax: +49 (0) 571/887 – 844555  
E-Mail: ✉ [support@wago.com](mailto:support@wago.com)

Es wurden alle erdenklichen Maßnahmen getroffen, um die Richtigkeit und Vollständigkeit der vorliegenden Dokumentation zu gewährleisten. Da sich Fehler, trotz aller Sorgfalt, nie vollständig vermeiden lassen, sind wir für Hinweise und Anregungen jederzeit dankbar.

E-Mail: ✉ [documentation@wago.com](mailto:documentation@wago.com)

Wir weisen darauf hin, dass die im Handbuch verwendeten Soft- und Hardwarebezeichnungen und Markennamen der jeweiligen Firmen im Allgemeinen einem Warenzeichenschutz, Markenzeichenschutz oder patentrechtlichem Schutz unterliegen.

**WAGO ist eine eingetragene Marke der WAGO Verwaltungsgesellschaft mbH.**

# Inhaltsverzeichnis

<b>Bestimmungen</b> .....	<b>5</b>
1.1 Bestimmungsgemäße Verwendung .....	5
1.2 Darstellungskonventionen .....	6
1.3 Rechtliche Informationen .....	7
<b>Sicherheit</b> .....	<b>9</b>
2.1 Allgemeine Sicherheitsbestimmungen .....	9
<b>Überblick</b> .....	<b>10</b>
<b>Voraussetzungen</b> .....	<b>12</b>
<b>Installieren</b> .....	<b>13</b>
<b>Grafische Benutzeroberfläche</b> .....	<b>14</b>
6.1 Datei und Projekt.....	15
6.2 Informationsmodell .....	16
6.3 Mapping.....	18
6.4 Eigenschaften Informationsmodell und Mapping .....	19
6.5 Symbolkonfiguration.....	22
6.6 Eigenschaften Symbolkonfiguration .....	23
<b>Bedienen</b> .....	<b>24</b>
7.1 Vorbereitung: Symbolkonfiguration in e!COCKPIT erstellen.....	24
7.2 Neues Projekt erstellen .....	24
7.3 Basismodelle auswählen.....	25
7.4 Neue Basismodelle hinzufügen.....	26
7.5 Informationsmodell bearbeiten.....	26
7.5.1 Instanz hinzufügen .....	26
7.5.2 Instanz bearbeiten .....	27
7.5.3 Instanzen löschen .....	28
7.5.4 Referenzen hinzufügen .....	28
7.5.5 Referenz löschen .....	29
7.5.6 Informationsmodell importieren .....	29
7.5.7 Informationsmodell exportieren .....	29
7.6 Symbolkonfiguration mit Applikationsvariablen importieren .....	30
7.7 Applikationsvariablen auf Variablen/Objekte des Informationsmodells abbilden ..	31
7.7.1 Applikationsvariable direkt auf eine OPC-UA-Variable abbilden .....	31
7.7.2 OPC-UA-Variable automatisch erstellen .....	32
7.7.3 OPC-UA-Objektstruktur automatisch erstellen .....	33
7.7.4 Applikationsvariable manuell auf OPC-UA-Variable abbilden .....	34
7.7.5 Variablen statische Werte zuweisen .....	34
7.7.6 Reguläre Ausdrücke zum Filtern der Applikationsvariablen verwenden .....	35
7.7.7 Mapping löschen .....	39

7.7.8	Mapping exportieren.....	39
7.8	Mapping in den Controller laden .....	40
7.9	Tastenkombinationen (Shortcuts) verwenden.....	41
	<b>Anhang.....</b>	<b>42</b>
8.1	Übersicht und Mapping von OPC-UA- und CODESYS Datentypen .....	42
8.2	Schutzrechte .....	45

# Bestimmungen

Die vorliegende Dokumentation gilt für die Software OPC UA Mapping Editor, Softwareversion 1.1.

## Hinweis

### Produktdokumentation beachten!

Die Software darf nur nach Anweisungen der vollständigen Gebrauchsanleitung installiert und betrieben werden. Kenntnis der vollständigen Gebrauchsanleitung ist Voraussetzung für die bestimmungsgemäße Verwendung.

1. Lesen Sie dieses Produkthandbuch.
2. Lesen Sie außerdem die Produkthandbücher Ihrer verwendeten Produkte (z. B. Feldbuskoppler, Controller, Panels).

## 1.1 Bestimmungsgemäße Verwendung

Der OPC UA Mapping Editor ist eine eigenständige, kostenfreie Software, um Programmstrukturen plattformunabhängig auf verschiedene OPC-UA-Modelle abzubilden.

### Sachwidrige Verwendung

Eine sachwidrige Verwendung der Produkte ist nicht gestattet. Die sachwidrige Verwendung ist insbesondere in den folgenden Fällen gegeben:

- Nichtbeachten der bestimmungsgemäßen Verwendung
- Verwendung der Produkte in Bereichen mit besonderem Risiko, die einen fehlerfreien Dauerbetrieb erfordern und in denen ein Ausfall oder Betrieb der Software zu einer unmittelbaren Gefahr für Leben, Körper oder Gesundheit oder zu erheblichen Sach- oder Umweltschäden führen kann (wie der Betrieb von Kernkraftwerken, Waffensystemen, Luft- und Kraftfahrzeugen)

### Gewährleistung und Haftung

Es gelten die Bestimmungen der allgemeinen Geschäfts- und Vertragsbedingungen für Lieferungen und Leistungen der WAGO GmbH & Co. KG sowie für Softwareprodukte und Produkte mit integrierter Software der WAGO Softwarelizenzvertrag, beide abrufbar unter: [www.wago.com](https://www.wago.com). Danach ist die Gewährleistung insbesondere in folgenden Fällen ausgeschlossen:

- Das Produkt wird sachwidrig verwendet.
- Der Mangel beruht auf speziellen Vorgaben (Hard- und Softwarekonfigurationen).
- Es wurden Modifikationen der Hard- oder Software durch den Nutzer oder Dritte durchgeführt, die nicht in dieser Dokumentation beschrieben sind und für das Auftreten des Mangels zumindest mitursächlich sind.

Einzelvertragliche Abreden haben stets Vorrang.

### Pflichten von Errichter/Betreiber

Die Verantwortung für die Sicherheit einer mit dem Produkt errichteten Anlage bzw. eines Systems liegt beim Errichter/Betreiber. Der Errichter/Betreiber ist für den sachgemäßen Einbau und die Sicherheit in den Anlagen verantwortlich. Dieser muss die geltenden Ge-

setze, Normen, Bestimmungen, örtlichen Vorschriften, den Stand und die Regeln der Technik zum Zeitpunkt der Installation einhalten und die in der Gebrauchsanleitung beschriebenen Vorgaben beachten. Ferner müssen die Errichtungsbestimmungen der Zulassungen eingehalten werden. Bei Nichteinhaltung darf das Produkt nicht im Geltungsbereich der Zulassung betrieben werden.

## 1.2 Darstellungskonventionen





### Zahlensysteme

100	Dezimal: Normale Schreibweise
0x64	Hexadezimal: C-Notation
'100'	Binär: In Hochkomma
'0110.0100'	Nibbles durch Punkt getrennt

### Textauszeichnungen

<i>kursiv</i>	Namen von Pfaden oder Dateien
<b>fett</b>	Bezeichnungen von Menüpunkten, Eingabe- oder Auswahlfelder, Hervorhebungen
Code	Ausschnitte von Programmcode
>	Auswahl eines Menüpunktes aus einem Menü
„Wert“	Werteingaben
[F5]	Beschriftungen von Schaltflächen oder Tasten

### Querverweise/Links

	Querverweis/Link zu einem Thema im Dokument
	Querverweis/Link zu einer Dokumentation
	Querverweis/Link zu einer Website
	Querverweis/Link zu einer E-Mail-Adresse

### Handlungsanweisung

- ✓ Dieses Symbol kennzeichnet eine Voraussetzung.
- 1. Handlungsschritt
- 2. Handlungsschritt
  - ⇒ Dieses Symbol kennzeichnet ein Zwischenergebnis.
  - ⇒ Dieses Symbol kennzeichnet ein Handlungsergebnis.

### Aufzählung

- Aufzählung erste Ebene
  - Aufzählung zweite Ebene

### Abbildungen

Abbildungen in dieser Dokumentation dienen dem besseren Verständnis und können von der tatsächlichen Ausführung der Produkte abweichen.

## Hinweise

### **GEFAHR**

#### Art und Quelle der Gefahr

Mögliche Folge der Gefahr, die auch Tod oder irreversible Verletzung umfasst

- Handlungsschritt zur Risikoreduktion

### **WARNUNG**

#### Art und Quelle der Gefahr

Mögliche Folge der Gefahr, die auch schwere Verletzung umfasst

- Handlungsschritt zur Risikoreduktion

### **VORSICHT**

#### Art und Quelle der Gefahr

Mögliche Folge der Gefahr, die zumindest leichte Verletzung umfasst

- Handlungsschritt zur Risikoreduktion

### **ACHTUNG**

#### Art und Quelle der Störung (nur Sachschäden)

Mögliche Störungen, die den Funktionsumfang bzw. die Ergonomie des Produktes einschränken, aber nicht vorhersehbar zu Gefährdung von Personen führen

- Handlungsschritt zur Risikoreduktion

### **Hinweis**

#### Hinweis und Information

Kennzeichnet Informationen, Erklärungen, Empfehlungen, Verweise etc.

## 1.3 Rechtliche Informationen

### Geistiges Eigentum

Vorbehaltlich anderslautender gesetzlicher Bestimmungen ist die Weitergabe oder Vervielfältigung dieses Dokumentes sowie die Verwertung und Mitteilung seines Inhalts ausdrücklich untersagt, es sei denn, es wurden abweichende Vereinbarungen getroffen. Fremdprodukte werden stets ohne Vermerk etwaiger Patentrechte genannt. Alle Rechte für den Fall der Patent-, Gebrauchsmuster- oder Designeintragung sind der WAGO GmbH & Co. KG, bei Fremdprodukten dem jeweiligen Hersteller, vorbehalten.

In der Dokumentation der Produkte werden Marken Dritter verwendet. Im Weiteren wird auf das Mitführen der Zeichen „®“ und „™“ verzichtet. Die Marken sind im Anhang aufgeführt (🔗 [Schutzrechte \[▶ 45\]](#)).

### **Änderungsvorbehalt**

Die in diesem Handbuch aufgeführten Vorschriften, Richtlinien, Normen usw. entsprechen dem Stand während der Ausarbeitung und unterliegen keinem Änderungsdienst. Sie sind vom Errichter/Betreiber in Eigenverantwortung in ihrer jeweils gültigen Fassung anzuwenden. Die WAGO GmbH & Co. KG behält sich das Recht vor, jederzeit technische Änderungen und Verbesserungen der Produkte und der Daten, Angaben und Abbildungen dieses Handbuchs vorzunehmen. Ein Anspruch auf Änderung oder Nachbesserung von bereits ausgelieferten Produkten ist – mit Ausnahme von Nachbesserungen im Rahmen der Gewährleistung – ausgeschlossen.

# Sicherheit

## 2.1 Allgemeine Sicherheitsbestimmungen

- Diese Dokumentation ist Teil des Produktes. Bewahren Sie deshalb die Dokumentation während der gesamten Nutzungsdauer des Produktes auf. Geben Sie die Dokumentation an den nachfolgenden Benutzer des Produktes weiter. Stellen Sie darüber hinaus sicher, dass gegebenenfalls jede erhaltene Ergänzung in die Dokumentation mit aufgenommen wird.
- Sämtliche Arbeitsschritte, die im Zusammenhang mit der Verwendung von WAGO Software stehen, dürfen nur von Fachkräften durchgeführt werden, die über ausreichende Kenntnisse im Umgang mit dem jeweils eingesetzten PC-System verfügen. Arbeitsschritte, in deren Folge Dateien auf dem PC-System erzeugt oder verändert werden, dürfen nur von Fachkräften durchgeführt werden, die zusätzlich zu den oben genannten auch über ausreichende Kenntnisse in der Administration des eingesetzten PC-Systems verfügen.  
Arbeitsschritte, in deren Folge das Verhalten des PC-Systems in einem Netzwerk verändert wird, dürfen nur von Fachkräften durchgeführt werden, die zusätzlich zu den oben genannten auch über ausreichende Kenntnisse in der Administration des jeweils eingesetzten Netzwerks verfügen.
- Halten Sie die geltenden Gesetze, Normen, Bestimmungen, örtlichen Vorschriften, den Stand der Technik und die Regeln der Technik zum Zeitpunkt der Installation ein.

# Überblick

„OPC Unified Architecture“ (OPC UA) ist eine plattformunabhängige und serviceorientierte Architektur. Sie wird verwendet, um Daten zu beschreiben und zu transportieren. Durch die Unabhängigkeit der Services können Geräte verschiedener Anbieter miteinander verbunden werden. Um den Anforderungen verschiedener Industrien mit gleichartigen Produkten und Maschinen gerecht zu werden, wurden sogenannte „Companion Specifications“ definiert, die an den Anwendungsfall angepasste Informationsmodelle (sogenannte Basismodelle) beinhalten.

Standardmäßig verwenden WAGO Controller das Informationsmodell „PLCopen“, um Daten für andere Anwendungen bereitzustellen. Mit dem OPC UA Mapping Editor ist es möglich, dieses Informationsmodell auf ein beliebiges anderes OPC-UA-Modell abzubilden (zu „mappen“), beispielsweise auf OPC-UA-Basismodelle wie „Robotik“ oder „Euro-map77“.

Die Mapping-Funktionalität wird von allen Controllern PFC200 der 2. Generation (750-821x), Control Panels TP600 (762-430x) und dem Edge Controller (752-8303/8000-0002) unterstützt.

Der OPC UA Mapping Editor bietet folgende Funktionen:

## **Editieren eines OPC-UA-Informationsmodells**

Über den Editor kann das OPC-UA-Informationsmodell erstellt und editiert werden. Dabei können optional bestehende Informationsmodelle als Basis importiert werden, z. B. „Robotik.xml“ oder „Euromap77.xml“, es können aber auch eigene firmen- oder anwendungsspezifische Modelle erstellt werden.

## **Importieren der Symbolkonfiguration (Applikationsdaten)**

Die Symbolkonfiguration beschreibt die Datenquelle mit den Variablen der Applikation. Diese wird mit einem *e!COCKPIT*-Projekt erstellt und in den OPC UA Mapping Editor importiert.

## **Verknüpfen der Datenquellen mit dem Informationsmodell**

Applikationsdaten werden mit den Variablen des Informationsmodells verknüpft. Somit wird die entsprechende OPC-UA-Variable zur Laufzeit den Wert der Applikationsvariable liefern.

## **Exportieren der Mapping-Datei**

Aus der Verknüpfung zwischen OPC-UA-Informationsmodell und der Datenquellen resultiert eine Mapping-Datei. Diese wird im Anschluss auf der Steuerung importiert.

## **Zugreifen auf Applikationsdaten der Steuerung**

Nachdem die Mapping-Datei auf der Steuerung importiert wurde, bietet die Steuerung die Applikationsdaten in dem entsprechenden Informationsmodell an. Die Daten und die neue Struktur gemäß des Informationsmodells können daraufhin mit entsprechenden OPC-UA-Clients weiterverarbeitet werden.

** Hinweis**

**Weitere Informationen zu OPC UA bei WAGO finden Sie hier:**

**Handbuch „WAGO OPC-UA-Server“**

Das Handbuch beschreibt die Funktionen des WAGO OPC-UA-Servers, das Herstellen einer Verbindung mittels OPC-UA-Client und das Konfigurieren via Web-Based-Management.

**Handbücher der PFC200, 2. Generation (750-821x)****Handbücher zu den Control Panels TP600 (762-430x)****Handbuch zum Edge Controller (752-8303/8000-0002)**

Die Handbücher beschreiben die jeweiligen Produkte, für welche der OPC UA Mapping Editor verwendet werden kann. Informationen zum Download von Lizenzen für diese Geräte finden Sie auf der Internetseite <http://www.wago.com>.

**Weitere Informationen zu Informationsmodellen finden Sie im Internet unter:**

<https://www.plcopen.org/technical-activities/communication>

<https://www.euromap.org/i40/overview>

<https://opcfoundation.org/about/opc-technologies/opc-ua/ua-companion-specifications/>

Download der Nodesets von der OPC-UA-Foundation:

<https://github.com/OPCFoundation/UA-NodeSet>

# Voraussetzungen

## Systemvoraussetzung

Tabelle 1: Systemvoraussetzung

Komponente	Voraussetzungen
Betriebssystem	Windows 10
Arbeitsspeicher	4 GB
Freier Festplattenspeicher	800 MB
Prozessor	Dualcore CPU
Bildschirmauflösung	Mindestens: 1.366 x 768 Pixel Empfohlen: 1.920 x 1.080 Pixel

## Lizenzen

Der OPC UA Mapping Editor, mit dem das Informationsmodell angepasst wird, ist als Download kostenfrei und lizenzfrei auf der WAGO Internetseite <https://www.wago.com/de/offene-automatisierung/connectivity-hero/opc-ua> verfügbar.

Grundsätzlich wird die Mapping-Funktionalität von allen Controllern PFC200 der 2. Generation (750-821x), TP600 Control Panels (762-430x) und Edge Controller (752-8303/8000-0002) unterstützt.

Um das angepasste Informationsmodell über die importierte Mapping-Datei auf der Steuerung verwenden zu können, ist jedoch eine erweiterte **e!RUNTIME** Lizenz des WAGO OPC-UA-Servers „OPC UA Server Extended“ auf dem Gerät erforderlich (Artikelnummer für PFC200(G2): 2759-2233/0210-1000 und Artikelnummer für TP600/Edge Controller: 2759-2236/0210-1000).

# Installieren

Die Installation des WAGO OPC UA Mapping Editors erfolgt als Setup.

1. Öffnen Sie die WAGO Internetseite zu OPC UA:  
[🌐 https://www.wago.com/de/offene-automatisierung/connectivity-hero/opc-ua](https://www.wago.com/de/offene-automatisierung/connectivity-hero/opc-ua)
2. Um einen Download-Link für den WAGO OPC UA Mapping Editor zu erhalten, klicken Sie auf den Link **[Zur Registrierung]** und füllen das Formular aus.
3. Laden Sie die Software herunter.
4. Starten Sie den Installationsprozess per Doppelklick auf die Installationsdatei.
5. Folgen Sie den Schritten im Installationsassistenten und wählen Sie ein Zielverzeichnis für die Installation aus.
6. Um die Installation zu starten, klicken Sie auf **[Installieren]**.
7. Um die Installation abzuschließen, klicken Sie auf **[Fertigstellen]**.
8. Starten Sie die Software über das Start-Menü **WAGO Software > WAGO OPC UA Mapping Editor**.

# Grafische Benutzeroberfläche

Die grafische Benutzeroberfläche gliedert sich in folgende Bereiche:

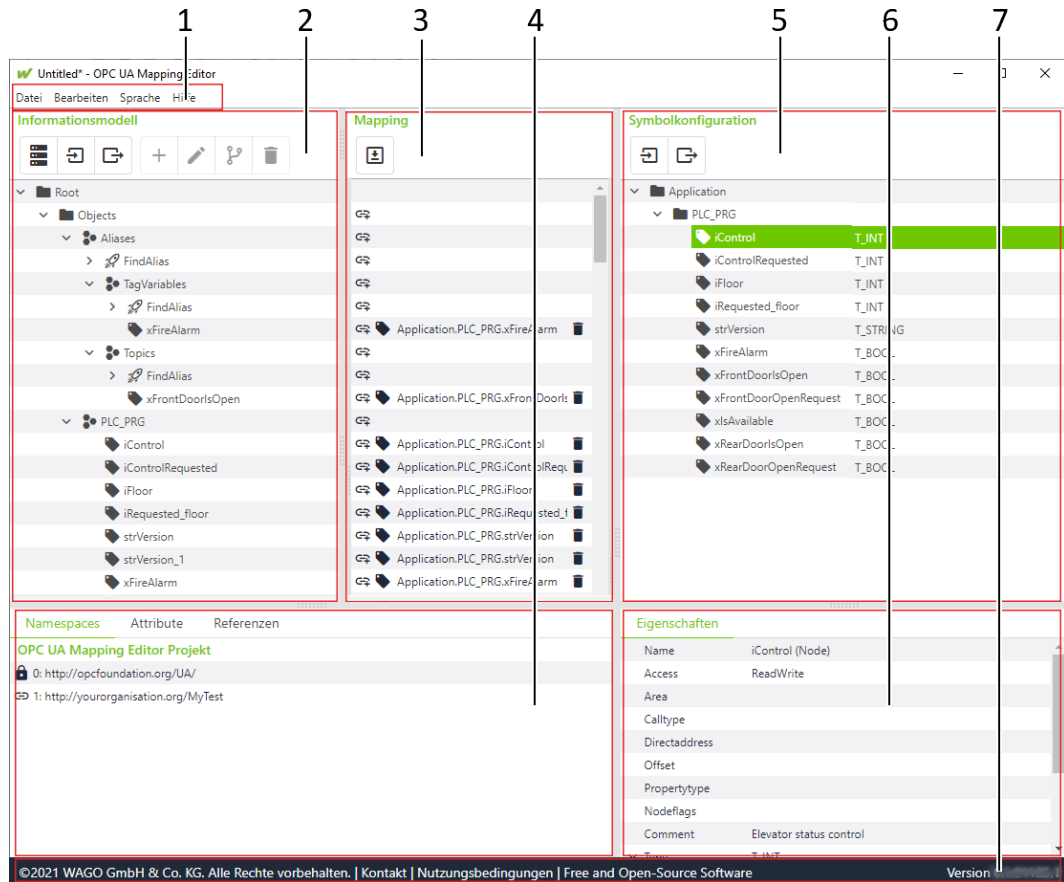


Abbildung 1: Grafische Benutzeroberfläche

Tabelle 2: Grafische Benutzeroberfläche

Pos.	Beschreibung
1	<p><b>Datei und Projekt</b> [▶ 15]</p> <p>Die Schaltflächen werden für allgemeine Einstellungen verwendet.</p>
2	<p><b>Informationsmodell</b> [▶ 16]</p> <p>In diesem Bereich werden Informationsmodelle angezeigt und angepasst.</p>
3	<p><b>Mapping</b> [▶ 18]</p> <p>In diesem Bereich werden Variablen aus der Symbolkonfiguration auf Variablen des Informationsmodells abgebildet und als Mapping-Datei exportiert.</p>
4	<p><b>Eigenschaften Informationsmodell und Mapping</b> [▶ 19]</p> <p>In diesem Bereich werden Informationen zu einem selektierten Element aus dem Informationsmodell angezeigt.</p>
5	<p><b>Symbolkonfiguration</b> [▶ 22]</p> <p>In diesem Bereich werden Applikationsdaten, die aus einem <b>e!COCKPIT</b>-Projekt resultieren, angezeigt, importiert und exportiert.</p>
6	<p><b>Eigenschaften Symbolkonfiguration</b> [▶ 23]</p> <p>In diesem Bereich werden Informationen zu einem selektierten Element der Symbolkonfiguration angezeigt.</p>
7	<p><b>Statusleiste</b></p> <p>Über Links werden Kontaktdaten der WAGO GmbH &amp; Co. KG sowie Nutzungsbedingungen und Lizenzbedingungen der Software erreicht.</p> <p>Außerdem wird die Version der Software angezeigt.</p>

## 6.1 Datei und Projekt

Das Menü bietet folgende Funktionen für allgemeine Einstellungen, die Datei und Projekt betreffen:

Tabelle 3: Menü „Datei“

Menüpunkt	Funktion
<b>Datei</b>	
<b>Neues Projekt</b>	Es wird ein Dialogfenster zum Erstellen eines neuen Projektes geöffnet. Dabei wird im Bereich „Informationsmodell“ initial ein Modell gemäß der OPC-UA-Standardspezifikation angelegt (siehe <a href="#">Neues Projekt erstellen [▶ 24]</a> ).
<b>Projekt öffnen</b>	Es wird ein Projekt geöffnet (*.opcprj), welches zuvor mit dem OPC UA Mapping Editor erstellt wurde.
<b>Projekt speichern (unter)</b>	Das Projekt wird gespeichert (*.opcprj).
<b>Beenden</b>	Der OPC UA Mapping Editor wird geschlossen.
<b>Bearbeiten</b>	
<b>Basismodelle auswählen</b>	Es wird ein Dialogfenster geöffnet, über welches gespeicherte Basismodelle ausgewählt oder neue Basismodelle hinzugefügt werden können (siehe <a href="#">Basismodelle auswählen [▶ 25]</a> ). Basismodelle sind Modelle, die über offene Plattformen zugänglich und bereits auf verschiedene Anwendungsbereiche angepasst sind.
<b>Informationsmodell importieren</b>	Es wird ein Dialogfenster geöffnet, über welches Informationsmodelle im Format *.xml importiert werden können (siehe <a href="#">Informationsmodell importieren [▶ 29]</a> ).
<b>Informationsmodell exportieren</b>	Es wird ein Dialogfenster geöffnet, über welches angepasste Informationsmodelle im Format *.xml exportiert werden können (siehe <a href="#">Informationsmodell exportieren [▶ 29]</a> ).
<b>Informationsmodell Einstellungen</b>	Es wird ein Dialogfenster geöffnet, über welches Modell-URI, Version und Datum des verwendeten Informationsmodells geändert werden können. Diese Daten werden erstmals beim Erstellen eines neuen Projektes eingegeben.
<b>Instanz hinzufügen</b>	Es wird ein Dialogfenster geöffnet, über welches Instanzen (Variablen oder Objekte) hinzugefügt werden können (siehe <a href="#">Instanz hinzufügen [▶ 26]</a> ).
<b>Instanz bearbeiten</b>	Es wird ein Dialogfenster geöffnet, über welches Instanzen (Variablen oder Objekte) bearbeitet werden können (siehe <a href="#">Instanz bearbeiten [▶ 27]</a> ).
<b>Instanz löschen</b>	Die ausgewählte Instanz wird gelöscht (siehe <a href="#">Instanzen löschen [▶ 28]</a> ).
<b>Referenz hinzufügen</b>	Es wird ein Dialogfenster geöffnet, über welches Referenzen hinzugefügt werden können (siehe <a href="#">Referenzen hinzufügen [▶ 28]</a> ).
<b>Sprache</b>	
<b>Deutsch/Englisch</b>	Die Oberflächensprache des Mapping Editors wird geändert.
<b>Hilfe</b>	
<b>Hilfe anzeigen</b>	Die Onlinehilfe des Mapping Editors wird in einem separaten Fenster geöffnet.

## 6.2 Informationsmodell

Im Bereich „Informationsmodell“ wird das OPC-UA-Modell angezeigt, welches für das Mapping verwendet wird.

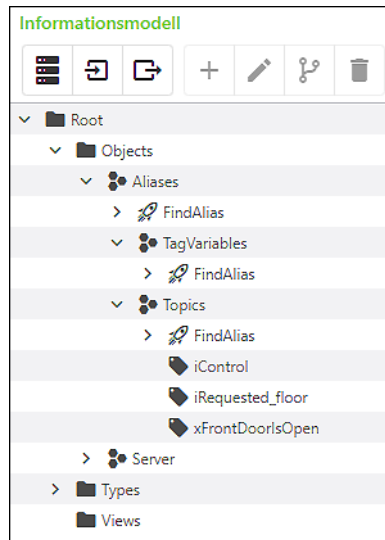


Abbildung 2: Bereich „Informationsmodell“

Das Informationsmodell wird in diesem Bereich wie folgt erstellt:

- Über „**Neues Projekt**“ wird zunächst ein initiales Modell in Standardstruktur gemäß OPC UA angelegt.
- Über „**Basismodell auswählen**“ werden Basismodelle mit bereits vordefinierten Strukturen gemäß Anwendungsbereich auf das Modell angewendet.
- Über „**Informationsmodell importieren**“ sind weitere Modelle importierbar, die als Datei vorliegen.






Folgende Symbolik wird verwendet:

- Ordner
- Objekt
- Objekttyp
- Methode
- Variable
- Datentyp

Im Bereich „Informationsmodell“ stehen über Schaltflächen folgende Funktionen zur Verfügung:

Tabelle 4: Funktionen im Bereich „Informationsmodell“

Schaltfläche	Funktion
[Basismodelle auswählen]	Es wird ein Dialogfenster geöffnet, über welches gespeicherte Basismodelle ausgewählt oder neue Basismodelle hinzugefügt werden können (siehe <a href="#">Basismodelle auswählen [ 25]</a> ). Basismodelle sind Modelle, die über offene Plattformen zugänglich und bereits auf verschiedene Anwendungsbereiche angepasst sind.
[Informationsmodell importieren]	Es wird ein Dialogfenster geöffnet, über welches Informationsmodelle im Format *.xml importiert werden können (siehe <a href="#">Informationsmodell importieren [ 29]</a> ). <b>Hinweis:</b> Beachten Sie, dass mit dem Import eines neuen Informationsmodells das bestehende überschrieben wird.

Schaltfläche		Funktion
	<b>[Informationsmodell exportieren]</b>	Es wird ein Dialogfenster geöffnet, über welches angepasste Informationsmodelle im Format *.xml exportiert werden können (siehe <a href="#">🔗 Informationsmodell exportieren [▶ 29]</a> ).
	<b>[Instanz hinzufügen]</b>	Es wird ein Dialogfenster geöffnet, über welches Instanzen (Variablen oder Objekte) hinzugefügt werden können (siehe <a href="#">🔗 Instanz hinzufügen [▶ 26]</a> ).
	<b>[Instanz bearbeiten]</b>	Es wird ein Dialogfenster geöffnet, über welches Instanzen (Variablen oder Objekte) bearbeitet werden können (siehe <a href="#">🔗 Instanz bearbeiten [▶ 27]</a> ).
	<b>[Referenz hinzufügen]</b>	Es wird ein Dialogfenster geöffnet, über welches eine Referenz hinzugefügt werden kann (siehe <a href="#">🔗 Referenzen hinzufügen [▶ 28]</a> ).
	<b>[Instanz löschen]</b>	Die ausgewählte Instanz wird gelöscht (siehe <a href="#">🔗 Instanzen löschen [▶ 28]</a> ).

Funktionen zum Hinzufügen von Instanzen und Referenzen sowie das Ausschneiden, Kopieren, Einfügen und Löschen von Elementen sind außerdem über das Kontextmenü erreichbar.

### 6.3 Mapping

Im Bereich „Mapping“ wird die Verknüpfung zwischen Applikationsdaten und Informationsmodell angezeigt. Dabei wird die Applikationsvariable der Symbolkonfiguration per Drag & Drop in den Bereich „Mapping“ neben die entsprechende Variable oder das Objekt des Informationsmodells gezogen (siehe [🔗 Applikationsvariablen auf Variablen/Objekte des Informationsmodells abbilden \[▶ 31\]](#)).

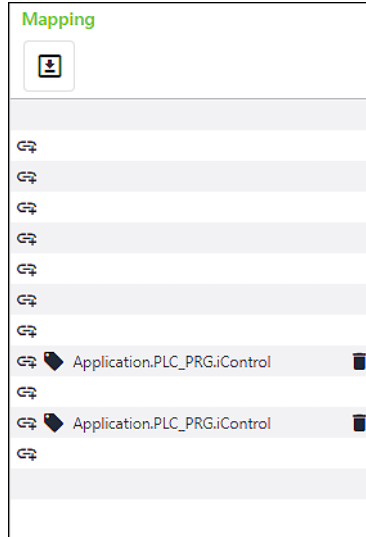


Abbildung 3: Auf das Informationsmodell abgebildete Variablen im Bereich „Mapping“

Folgende Symbolik wird verwendet:

- 🔗 Verknüpfbares Element (hier ist ein Mapping möglich)
- ◆ Variable
- ⑤ Statischer Wert
- .\* Regulärer Ausdruck

Im Bereich „Mapping“ stehen folgende Funktionen zur Verfügung:

Tabelle 5: Funktionen im Bereich „Mapping“

Schaltfläche	Funktion
[Mapping exportieren]	Es wird ein Dialogfenster zum Exportieren des Mappings als XML-Datei geöffnet (siehe <a href="#">🔗 Mapping exportieren [▶ 39]</a> ). Die Mapping-Datei wird verwendet, um sie über das WBM in den Controller zu laden.
[Mapping löschen]	Das Löschen einzelner Mappings zwischen Symbolkonfiguration und Informationsmodell ist über das Papierkorb-Symbol in der jeweiligen Zeile möglich (siehe <a href="#">🔗 Mapping löschen [▶ 39]</a> ).

## 6.4 Eigenschaften Informationsmodell und Mapping

Abhängig von dem Element, welches im Bereich „Informationsmodell“ ausgewählt wird, werden in diesem Bereich kontextsensitiv weitere Informationen angezeigt.

### Register „Namespaces“

In diesem Register wird der Namensraum angezeigt, auf welchem das Informationsmodell basiert.

Namespaces	Attribute	Referenzen	Mapping
<b>OPC UA Mapping Editor Projekt</b>			
🔒 0: http://opcfoundation.org/UA/			
1: http://yourorganisation.org/NewProject			

Abbildung 4: Register „Namespaces“

### Register „Attribute“


In diesem Register werden die Attribute und Werte des jeweils ausgewählten Elementes angezeigt.

Namespaces	Attribute	Referenzen	Mapping
	<b>Attribut</b>		<b>Value</b>
	Nodeld		1;i=7001
	NodeClass		Variable
	BrowseName		1, "iControl"
	DisplayName		""; "iControl"
	Description		""; ""
	Datentyp		Int16

Abbildung 5: Register „Attribute“

### Register „Referenzen“

In diesem Register werden verwendete Referenztypen mit Knotenklasse, Name, Modellregel und Datentyp für das ausgewählte Element angezeigt (siehe auch [🔗 Referenzen hinzufügen \[▶ 28\]](#)).

Ist das Löschen der Referenz möglich (dies gilt z. B. für selbst erstellte Referenzen), wird ein Papierkorb-Symbol  zum Löschen des Eintrages angezeigt (siehe [🔗 Referenz löschen \[▶ 29\]](#)).


Namespaces	Attribute	Referenzen	Mapping			
	<b>Referenztyp</b>	<b>NodeClass</b>	<b>Name</b>	<b>ModellingRule</b>	<b>Datentyp</b>	
	HasTypeDefinition	VariableType	BaseDataVariableType			
	AliasFor	Variable	iFloor		Int16	

Abbildung 6: Register „Referenzen“

### Register „Mapping“

Dieses Register wird nur dann angezeigt, wenn ein Element des Informationsmodells ausgewählt ist, für welches ein Mapping mit Applikationsvariablen möglich ist.

Bei Auswahl einer **Variable** werden im Register „Mapping“ folgende Einstellungen angezeigt:

Abbildung 7: Register „Mapping“ (hier bei Auswahl einer Variable)

Tabelle 6: Anzeige und Auswahl des Mapping-Typs

Mapping-Typ	Beschreibung
<b>Kein Mapping</b>	„Kein Mapping“ wird angezeigt, wenn für die im Informationsmodell ausgewählte Variable (noch) kein Mapping zu einer Applikationsvariable besteht. <b>Hinweis:</b> Ist bereits ein Mapping vorhanden und wird zu dieser Option gewechselt und auf <b>[Bestätigen]</b> geklickt, dann wird das bestehende Mapping gelöscht und der Eintrag im Bereich „Mapping“ entfernt.
<b>Mapping auf Applikationsvariable</b>	„Mapping auf Applikationsvariable“ wird angezeigt, wenn für die im Informationsmodell ausgewählte Variable ein Mapping zu einer Applikationsvariable besteht (siehe <a href="#">🔗 Applikationsvariablen auf Variablen/Objekte des Informationsmodells abbilden [▶ 31]</a> ). In diesem Fall werden Datentyp und Symbolpfad der verknüpften Applikationsvariable angezeigt. <b>Hinweis:</b> Ist zuvor noch kein Mapping erfolgt und es wird zu dieser Option gewechselt, dann kann der Symbolpfad an dieser Stelle auch manuell eingetragen werden (siehe <a href="#">🔗 Applikationsvariable manuell auf OPC-UA-Variable abbilden [▶ 34]</a> ). Bei Klicken auf <b>[Bestätigen]</b> wird der Symbolpfad auf Gültigkeit geprüft.
<b>Mapping auf statischen Wert</b>	Der Variable wird ein statischer Wert zugewiesen (siehe <a href="#">🔗 Variablen statische Werte zuweisen [▶ 34]</a> ). <b>Hinweis:</b> Beachten Sie, dass der eingetragene Wert zum verwendeten Datentyp passen muss. Es erfolgt keine Prüfung auf Gültigkeit der Werte. Beachten Sie dazu die OPC-UA-Attribut-Referenz unter <a href="https://reference.opcfoundation.org">🔗 https://reference.opcfoundation.org</a> .

Bei Auswahl eines **Objektes** werden im Register „Mapping“ folgende Einstellungen angezeigt:

Abbildung 8: Register „Mapping“ (hier bei Auswahl eines Objektes)

Tabelle 7: Einstellungen für reguläre Ausdrücke

Parameter	Beschreibung
DataSourceIdentifizier	Mit diesem regulären Ausdruck werden die Symbolpfade aller zur Verfügung stehenden Applikationsvariablen gefiltert.
Nodeld/ NodeldTarget	Für jeden bereits über den „DataSourceIdentifizier“ gefilterten Symbolpfad wird über die Eigenschaften „Nodeld“ und „NodeldTarget“ ein Nodeld-String generiert. <b>Hinweis:</b> Beachten Sie, dass dieser Nodeld-String gemäß der OPC-UA-Spezifikation eindeutig sein muss.  Die Eigenschaft „Nodeld“ gibt dabei einen regulären Ausdruck an, der auf die bereits gefilterten Symbolpfade angewendet wird.  Die Eigenschaft „NodeldTarget“ setzt aus den Bestandteilen des regulären Ausdrucks der „Nodeld“ den endgültigen Nodeld-Wert der OPC-UA-Variable zusammen.
BrowseName/ BrowseNameTarget	Analog zur Eigenschaft „Nodeld“ wird über diese Eigenschaften der OPC-UA-BrowseName aus dem bereits gefilterten Symbolpfad gebildet.
DisplayName/ displayNameTarget	Analog zur Eigenschaft „Nodeld“ wird über diese Eigenschaften der OPC-UA-DisplayName aus dem bereits gefilterten Symbolpfad gebildet.

Die Beschreibung zur Verwendung von regulären Ausdrücken finden Sie unter [🔗 Reguläre Ausdrücke zum Filtern der Applikationsvariablen verwenden \[▶ 35\]](#).

### 6.5 Symbolkonfiguration

Im Bereich „Symbolkonfiguration“ werden die Inhalte einer importierten Symbolkonfiguration (XML) angezeigt, die aus einer Applikation resultieren. Die Struktur orientiert sich dabei für Steuerungen standardmäßig an der PLCopen-Struktur.

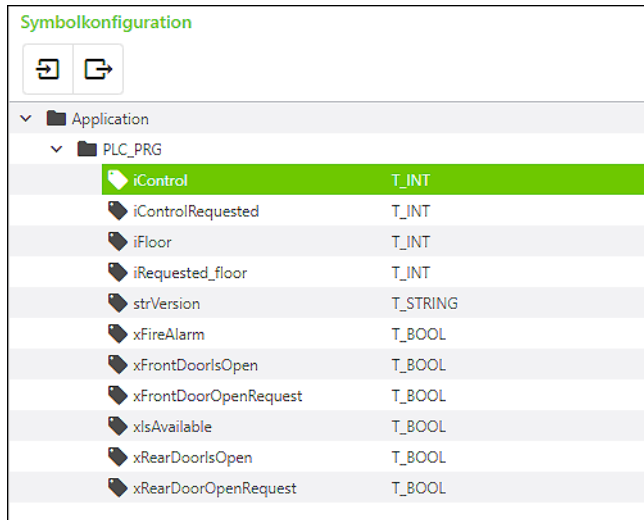




Abbildung 9: Bereich „Symbolkonfiguration“

Im Bereich „Symbolkonfiguration“ stehen folgende Funktionen zur Verfügung:

Tabelle 8: Funktionen im Bereich „Symbolkonfiguration“

Schaltfläche	Funktion
 <b>[Symbolkonfiguration importieren]</b>	Es wird ein Dialogfenster geöffnet, über welches Symbolkonfigurationen im Format *.xml importiert werden können (siehe <a href="#">Symbolkonfiguration mit Applikationsvariablen importieren [▶ 30]</a> ).
 <b>[Symbolkonfiguration exportieren]</b>	Es wird ein Dialogfenster geöffnet, über welches die Symbolkonfiguration im Format *.xml exportiert werden kann. Obwohl im OPC UA Mapping Editor keine Änderungen an der Symbolkonfiguration selbst vorgenommen werden, kann die Exportfunktion beispielsweise verwendet werden, wenn die Symbolkonfiguration in einem anderen Programm bearbeitet werden soll. Es kann vorkommen, dass ganze Projekte aus dem Mapping Editor weitergegeben wurden und die ursprüngliche Symbolkonfiguration deshalb nicht mehr vorliegt.

## 6.6 Eigenschaften Symbolkonfiguration

In diesem Bereich werden die Eigenschaften von ausgewählten Variablen der Symbolkonfiguration angezeigt.

Eigenschaften	
Name	iControl (Node)
Access	ReadWrite
Area	
Calltype	
Directaddress	
Offset	
Propertytype	
Nodeflags	
Comment	Elevator status control
▼ Type	T_INT
lecName	INT
TypeClass	Int
Size	2
BitOffset	
SwapSize	2

Abbildung 10: Eigenschaften zur Symbolkonfiguration

# Bedienen

## 7.1 Vorbereitung: Symbolkonfiguration in e!COCKPIT erstellen

1. Öffnen Sie **e!COCKPIT**.
2. Öffnen Sie die Programmstruktur und erstellen Sie Ihr Programm.
3. Klicken Sie mit der rechten Maustaste auf die Applikation (offline) und wählen Sie über das Kontextmenü **Symbolkonfiguration** aus.
4. Wählen Sie im Dialogfenster die Option „OPC-UA-Funktionalitäten unterstützen“ aus.
5. Um die Symbolkonfiguration anzulegen, klicken Sie auf **[Hinzufügen]**.  
⇒ Das Register „Symbolkonfiguration“ wird angezeigt.
6. Klicken Sie zunächst auf **[Erstellen]**, da das Programm für die weiteren Schritte übersetzbar sein muss.
7. Wählen Sie diejenigen Variablen aus der Liste aus, die über OPC-UA vom Server veröffentlicht werden sollen.  
⇒ Die gewählte Symbolkonfiguration wird beim nächsten Verbinden automatisch per Download/Online-Change auf den Controller übertragen. Die Parameter liegen damit zunächst in dem standardmäßigen Format nach PLCopen-Spezifikation auf dem Controller vor.  
Die Symbolkonfiguration wird als Datei in dem Projektverzeichnis gespeichert, in dem auch die Projektdatei „.ecp“ liegt.

## 7.2 Neues Projekt erstellen

1. Öffnen Sie den OPC UA Mapping Editor.
2. Klicken Sie in der Menüleiste auf **Datei > Neues Projekt**.
3. Geben Sie im Dialogfenster im Feld „Modell-URI“ einen Namespace ein. Dieser wird dazu verwendet, in einem Client-Programm Daten zu filtern.
4. Geben Sie außerdem eine Version und ein Datum für das Modell ein, welches Sie anlegen möchten.
5. Klicken Sie auf **[OK]**.  
⇒ Die Grundstruktur des Informationsmodells wird dargestellt.

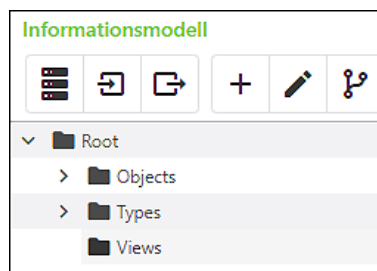


Abbildung 11: Grundstruktur Informationsmodell

Sie können mit diesem Modell weiterarbeiten. Für verschiedene industrielle Anwendungsbereiche sind jedoch auch Basismodelle verfügbar, die bereits an den jeweiligen Anwendungsbereich angepasste Datenmodelle beinhalten (siehe [🔗 Basismodelle auswählen \[▶ 25\]](#)).

### 7.3 Basismodelle auswählen

Basismodelle werden von der OPC-UA-Organisation als „Companion Specifications“ für die jeweiligen Anwendungsbereiche veröffentlicht.

- Um ein Basismodell zu verwenden, klicken Sie auf die Schaltfläche **[Basismodelle auswählen]**.

⇒ Das Dialogfenster „Basismodelle auswählen“ wird geöffnet.

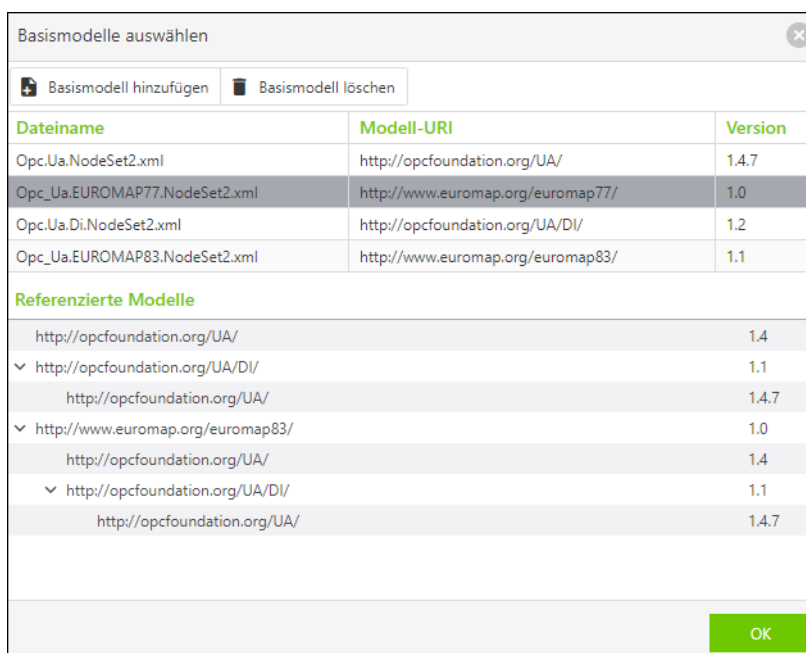


Abbildung 12: Dialogfenster „Basismodelle auswählen“

- Wählen Sie aus der Liste gespeicherter Basismodelle das Modell aus, welches Sie für Ihr Projekt verwenden möchten, beispielsweise „EuroMap“.

Optional laden Sie zunächst das gewünschte Basismodell herunter (siehe [🔗 Neue Basismodelle hinzufügen \[▶ 26\]](#))

- Klicken Sie auf **[OK]**.

⇒ Ihr Informationsmodell wird nun durch das referenzierte Basismodell an den jeweiligen Anwendungsbereich angepasst.

Sie können das Informationsmodell verwenden oder individuell weiter ergänzen (siehe [🔗 Informationsmodell bearbeiten \[▶ 26\]](#)).

### 7.4 Neue Basismodelle hinzufügen

1. Um die Liste auswählbarer Basismodelle im Dialogfenster „Basismodelle auswählen“ zu erweitern, laden Sie zunächst weitere Basismodelle unter <https://opcfoundation.org/ua/> herunter.
  2. Wählen Sie über die Schaltfläche **[Basismodell hinzufügen]** das zuvor gespeicherte Basismodell aus.
  3. Klicken Sie auf **[Öffnen]**.
- ⇒ Die Liste im Dialogfenster „Basismodelle auswählen“ wird entsprechend erweitert.

### 7.5 Informationsmodell bearbeiten

Im Bereich „Informationsmodell“ können Sie die Elemente des OPC-UA-Informationsmodells bearbeiten.

Auf der obersten Ebene besteht das Modell gemäß der OPC-UA-Basispezifikation standardmäßig aus den Ordnern „Objects“, „Types“ und „Views“. Im Ordner „Objects“ können Sie Instanzen (Objekte und Variablen) hinzufügen, löschen und bearbeiten.

**Hinweis:** Die Ordner „Types“ und „Views“ sind nicht veränderbar. Falls Sie hier eigene Typdefinitionen erstellen möchten, können Sie diese über ein anderes XML-Tool erstellen/verändern. Anschließend importieren Sie das geänderte Modell wieder in den OPC UA Mapping Editor.

#### 7.5.1 Instanz hinzufügen

1. Wählen Sie im Informationsmodell den Ordner „Objects“ oder einen darunterliegenden Objektknoten aus.
  2. Klicken Sie auf die Schaltfläche + **[Instanz hinzufügen]**.
- ⇒ Der Dialog „Instanz hinzufügen“ wird geöffnet.

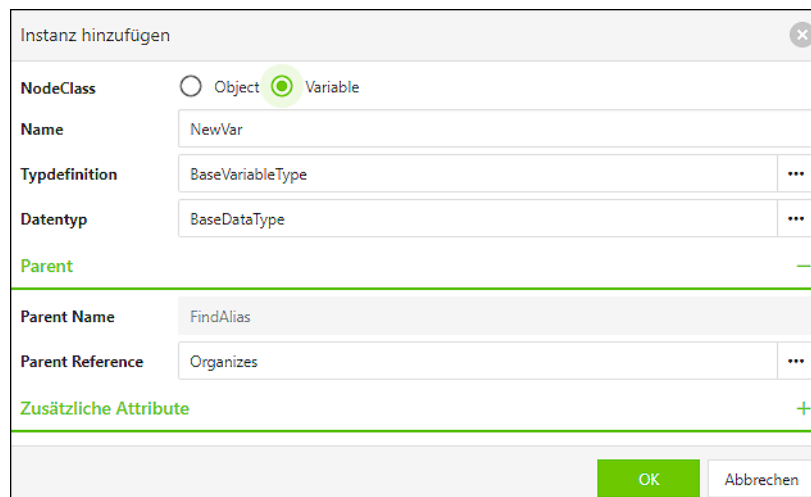



Abbildung 13: Instanz hinzufügen

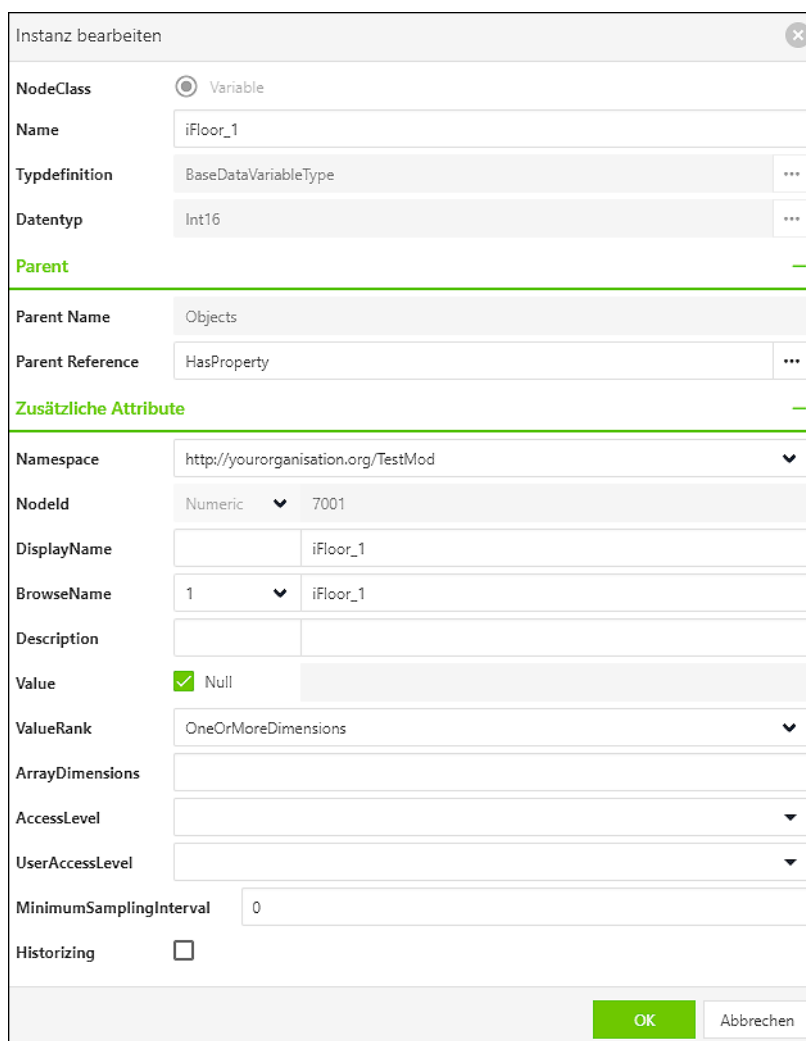
3. Wählen sie unter „NodeClass“ aus, ob sie ein Objekt oder eine Variable hinzufügen möchten.
4. Geben Sie den Namen des Objektes bzw. der Variablen in das entsprechende Textfeld ein.

5. Wählen Sie den Typ des Objektes bzw. der Variable aus.
  - ⇒ Die zur Verfügung stehenden Typen hängen von den im Projekt ausgewählten OPC-UA-Basismodellen (beispielsweise einer „Companion Specification“) ab. Ohne Hinzufügen einer Basisspezifikation werden lediglich die Datentypen angeboten, die für OPC UA standardmäßig definiert sind.
6. Optional können Sie weitere OPC-UA-Eigenschaften der neuen Instanz bearbeiten:
  - Im Bereich „Parent“ können Sie die Referenz ändern, welche die Beziehung zum Parent-Knoten definiert, beispielsweise auf den Referenztyp „Organizes“ oder „Has-Component“.
  - Im Bereich „Zusätzliche Attribute“ können Sie weitere OPC-UA-Attribute setzen.
7. Schließen Sie den Dialog mit **[OK]**.

### 7.5.2 Instanz bearbeiten

Die Eigenschaften eines bereits bestehenden Objektes oder einer Variablen können nachträglich bearbeitet werden.

1. Wählen Sie die zu bearbeitende Instanz im Informationsmodell aus.
2. Klicken Sie auf die Schaltfläche  **[Instanz bearbeiten]**.
  - ⇒ Der Dialog „Instanz bearbeiten“ öffnet sich.



The screenshot shows the 'Instanz bearbeiten' dialog box with the following fields and values:

- NodeClass:** Variable (selected with a radio button)
- Name:** iFloor\_1
- Typdefinition:** BaseDataVariableType
- Datentyp:** Int16
- Parent:** (Section header)
- Parent Name:** Objects
- Parent Reference:** HasProperty
- Zusätzliche Attribute:** (Section header)
- Namespace:** http://yourorganisation.org/TestMod
- NodeId:** Numeric 7001
- DisplayName:** iFloor\_1
- BrowseName:** 1 iFloor\_1
- Description:** (empty)
- Value:**  Null
- ValueRank:** OneOrMoreDimensions
- ArrayDimensions:** (empty)
- AccessLevel:** (dropdown menu)
- UserAccessLevel:** (dropdown menu)
- MinimumSamplingInterval:** 0
- Historizing:**

At the bottom right, there are two buttons: **OK** and **Abbrechen**.

Abbildung 14: Instanz bearbeiten

3. Bearbeiten Sie die gewünschten Eigenschaften der Instanz.  
**Hinweis:** Die Eigenschaft „NodeClass“ einer Instanz lässt sich nachträglich nicht mehr ändern. Falls Sie diese ändern möchten, legen Sie die Instanz (Objekt oder Variable) neu an.
4. Schließen Sie den Dialog mit **[OK]**.

### 7.5.3 Instanzen löschen

Sie können erstellte Instanzen (Objekte und Variablen) wie folgt löschen:

1. Wählen Sie das zu löschende Objekt bzw. die zu löschende Variable im Informationsmodell aus.
2. Klicken Sie im Kontextmenü auf **[Instanz löschen]**.  
 ⇒ Das Objekt bzw. die Variable wird aus dem Informationsmodell gelöscht.

**Hinweis:** Objekte oder Variablen, die standardmäßig im Basismodell definiert sind, können nicht gelöscht werden.

### 7.5.4 Referenzen hinzufügen

Sie können Referenzen einer Instanz im OPC-UA-Informationsmodell hinzufügen.

1. Wählen Sie ein Objekt oder eine Variable im OPC-UA-Informationsmodell aus, welches/welche nicht Bestandteil des Basismodells ist.
2. Klicken Sie auf die Schaltfläche **[Referenz hinzufügen]**.  
 ⇒ Der Dialog „Referenz hinzufügen“ öffnet sich.

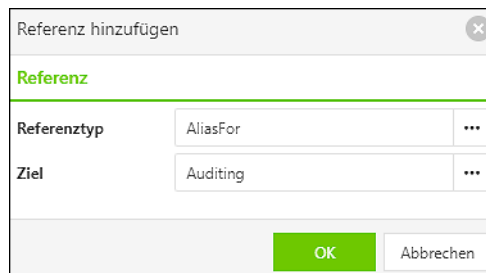


Abbildung 15: Referenz hinzufügen

3. Wählen Sie den Typ der Referenz aus.  
 ⇒ Die zur Auswahl stehenden Typen hängen von den im Projekt ausgewählten OPC-UA-Basismodellen (beispielsweise einer „Companion Specification“) ab. Ohne Hinzufügen einer Basisspezifikation werden lediglich die Referenztypen angeboten, die in OPC UA standardmäßig definiert sind.
4. Wählen Sie das Ziel der Referenz aus. Die Auswahl besteht aus allen Knoten im aktuellen Informationsmodell.
5. Bestätigen Sie den Dialog mit **[OK]**.  
 ⇒ Die neu hinzugefügte Referenz wird im Register „Referenzen“ angezeigt.

Namespaces	Attribute	Referenzen	Mapping		
<b>Referenztyp</b>	<b>NodeClass</b>	<b>Name</b>	<b>ModellingRule</b>	<b>Datentyp</b>	
HasTypeDefinition	VariableType	BaseDataVariableType			
AliasFor	Variable	Auditing	Mandatory	Boolean	

Abbildung 16: Neu angelegte Referenz

### 7.5.5 Referenz löschen

Hinzugefügte Referenzen können wieder gelöscht werden.

1. Wählen Sie das Objekt bzw. die Variable im Informationsmodell aus, für welches/welche Sie eine Referenz erstellt haben.
2. Wählen Sie im Bereich „Eigenschaften Informationsmodell“ im Register „Referenzen“ die zu löschende Referenz aus.




Namespaces	Attribute	Referenzen	Mapping			
Referenztyp	NodeClass	Name	ModellingRule	Datentyp		
HasTypeDefinition	VariableType	BaseDataVariableType				
AliasFor	Variable	Auditing	Mandatory	Boolean		

Abbildung 17: Referenz löschen


3. Klicken Sie in der rechten Spalte des Eintrages auf das Symbol .
- ⇒ Die Referenz wird gelöscht.

**Hinweis:** Wird kein Papierkorb-Symbol angezeigt, ist die Referenz nicht löscherbar. Dies ist beispielsweise für standardmäßig im Basismodell definierte Referenzen der Fall. Auch Referenzen auf untergeordnete (Kind-)Strukturen werden als nicht-löscherbar angezeigt. Von der übergeordneten Struktur aus, sind Bearbeitungen nicht zugelassen. Zum Ändern oder Löschen wechseln Sie deshalb direkt in die untergeordnete (Kind-)Instanz.


### 7.5.6 Informationsmodell importieren

1. Um ein Informationsmodell zu importieren, klicken Sie auf die Schaltfläche  **[Informationsmodell importieren]**.  
**Hinweis:** Beachten Sie, dass mit dem Import eines neuen Informationsmodells das bestehende überschrieben wird.
  2. Wählen Sie ein Informationsmodell im Format \*.xml aus. Dies können Modelle sein, die mit einer anderen Software erstellt wurden (z. B. UaModeler).  
**Hinweis:** Die XML-Datei muss ein gültiges OPC UA Nodeset gemäß dem XML-Schema der OPC Foundation beinhalten.
  3. Klicken Sie auf **[Öffnen]**.
- ⇒ Das Informationsmodell wird angezeigt.

### 7.5.7 Informationsmodell exportieren

1. Um ein Informationsmodell zu exportieren, klicken Sie auf die Schaltfläche  **[Informationsmodell exportieren]**.
  2. Wählen Sie einen Speicherort aus und geben Sie einen Dateinamen für das Informationsmodell ein.
  3. Klicken Sie auf **[Speichern]**.
- ⇒ Das Informationsmodell wird im Format \*.xml gespeichert.

## 7.6 Symbolkonfiguration mit Applikationsvariablen importieren

- Um die in **e!COCKPIT** erstellte XML-Datei mit den Applikationsvariablen (Symbolkonfiguration) zu öffnen, klicken Sie im Bereich „Symbolkonfiguration“ auf die Schaltfläche  **[Symbolkonfiguration importieren]**.

⇒ Der Inhalt der XML-Datei wird angezeigt.

**Hinweis:** Es wird dabei keine Validierung durchgeführt.

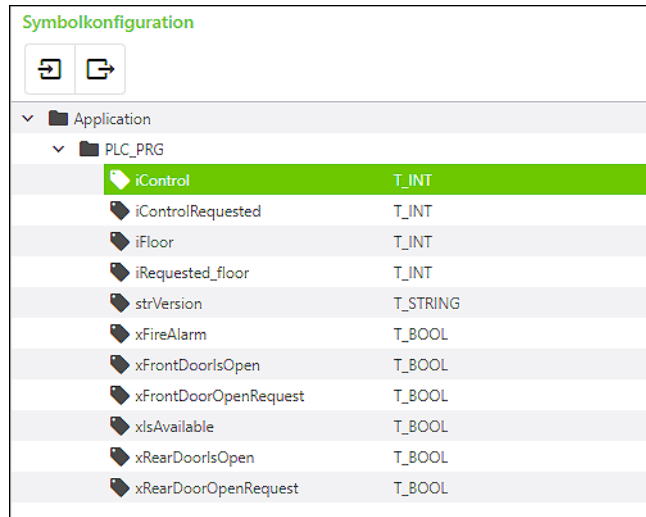



Abbildung 18: Symbolkonfiguration importieren

### Tipps:

- Für das Mapping ist in diesem Bereich eine Mehrfachauswahl möglich, sofern die Applikationsvariablen neben ein OPC-UA-Objekt gezogen werden.
- Bei Auswahl einzelner Variablen erhalten Sie im unteren Bereich „Eigenschaften“ weitere Informationen.
- Über  **reguläre Ausdrücke** [▶ 35](#)] kann die Symbolkonfiguration zur Laufzeit gefiltert werden.

## 7.7 Applikationsvariablen auf Variablen/Objekte des Informationsmodells abbilden

Der OPC UA Mapping Editor bietet verschiedene Möglichkeiten, ein Mapping zwischen Applikationsvariablen und OPC-UA-Modell zu erstellen. Sie können Variablen sowohl automatisch per Drag & Drop als auch manuell durch Eingabe der Zielvariablen miteinander verknüpfen. Ist im Informationsmodell keine passende Variable für das Mapping vorhanden, wird automatisch eine neue Variable oder eine entsprechende Ordnerstruktur im Informationsmodell angelegt. Zusätzlich kann über reguläre Ausdrücke eine Auswahl an Applikationsvariablen aus der Symbolkonfiguration definiert werden, für welche zur Laufzeit OPC-UA-Variablen generiert werden.

**Hinweis:** Beachten Sie, dass Applikationsvariablen nur auf Variablen/Objekte des Informationsmodells abgebildet werden können, die im Bereich „Mapping“ mit dem Symbol ⇄ gekennzeichnet sind.

### 7.7.1 Applikationsvariable direkt auf eine OPC-UA-Variable abbilden

Hierbei wird eine bereits angelegte Variable im OPC-UA-Modell mit einer Applikationsvariable verknüpft. Dabei müssen die Datentypen der Applikationsvariable und der OPC-UA-Variable kompatibel sein.

- Ziehen Sie dazu die Variable aus der Symbolkonfiguration per Drag & Drop in den Bereich „Mapping“ neben die jeweilige OPC-UA-Variable.
- ⇒ Die Applikationsvariable wird im Bereich „Mapping“ neben der Variable des Informationsmodells angezeigt.



Abbildung 19: Direktes Mapping

Die Datentypen müssen für das Mapping zusammenpassen. Die Prüfung auf Übereinstimmung der Datentypen erfolgt gemäß [Übersicht und Mapping von OPC-UA- und CODESYS Datentypen](#) [▶ 42].

#### **i** Hinweis

#### **Hinweis zum Mapping von Aufzählungstypen (ENUM) und Strukturdefinitionen (STRUCTS)!**

Variablen in der Symbolkonfiguration, die vom Datentyp „ENUM“ oder „STRUCT“ sind, können nicht direkt auf eine Variable im OPC-UA-Informationsmodell abgebildet werden. Variablen dieser Datentypen können jedoch per Drag & Drop in den Bereich „Mapping“ neben ein Objekt des Informationsmodells gezogen werden (siehe [OPC-UA-Variable automatisch erstellen](#) [▶ 32]). Der OPC UA Mapping Editor legt daraufhin eine neue Variable im OPC-UA-Informationsmodell an. Weiterhin wird (sofern der Datentyp „ENUM“ oder „STRUCT“ zum ersten Mal verwendet wird) ein entsprechender OPC-UA-Datentyp angelegt und der neuen Variable zugeordnet.

### 7.7.2 OPC-UA-Variable automatisch erstellen

Sie können OPC-UA-Variablen im Informationsmodell wie folgt automatisch anlegen:

- Ziehen Sie eine Variable aus der Symbolkonfiguration per Drag & Drop in den Bereich „Mapping“ neben ein Objekt im OPC-UA-Informationsmodell.

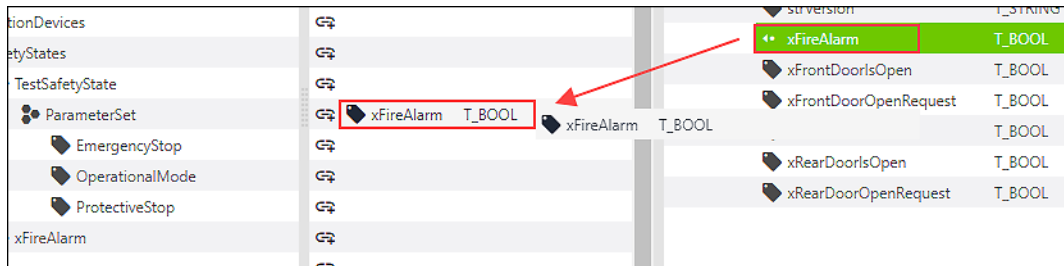


Abbildung 20: Variable in den Bereich „Mapping“ ziehen

⇒ Dem OPC-UA-Objekt wird daraufhin eine neue Variable hinzugefügt.

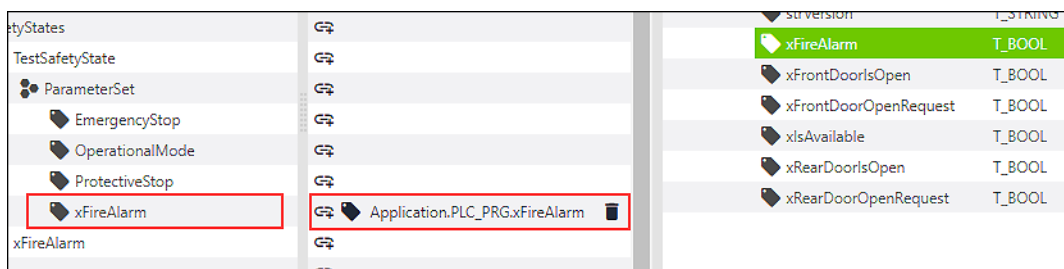


Abbildung 21: Neu erstellte OPC-UA-Variable

Der Name und der Datentyp dieser Variable wird entsprechend der Variable aus der Symbolkonfiguration angelegt (siehe [Übersicht und Mapping von OPC-UA- und CO-DESYS Datentypen \[► 42\]](#)).

**Tipp:** Via Mehrfachauswahl können mehrere Variablen gleichzeitig erstellt werden.

### 7.7.3 OPC-UA-Objektstruktur automatisch erstellen

Die Symbolkonfiguration ist als Baumstruktur organisiert. Dieser Baum enthält auch Knoten, die nicht direkt einer Variablen in einer Applikation entsprechen. Stattdessen repräsentieren diese Knoten einen Ordner wie z. B. ein SPS-Programm (hier „PLC\_PRG“), welches dann wiederum Variablen beinhaltet.

Der OPC UA Mapping Editor bietet die Möglichkeit, für einen solchen Knoten ein Objekt im OPC-UA-Informationsmodell anzulegen. Alle Variablen und ggf. Unterordner werden dabei ebenfalls als Variablen bzw. weitere Kind-Objekte angelegt.

- Ziehen Sie dazu den Ordner aus der Symbolkonfiguration per Drag & Drop in den Bereich „Mapping“ neben ein OPC-UA-Objekt.

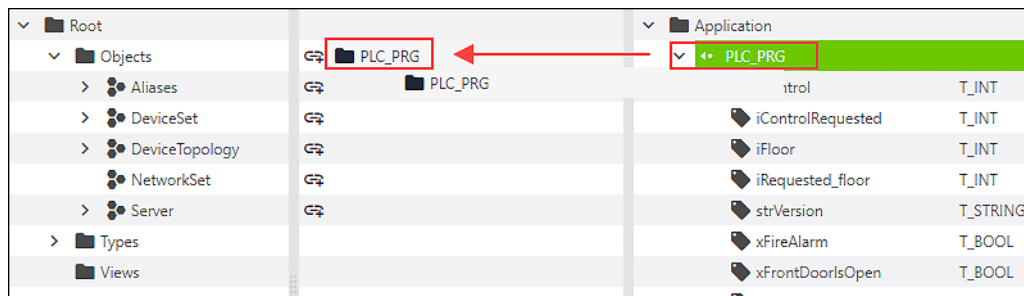


Abbildung 22: Ganzen Ordner auf ein OPC-UA-Objekt ziehen

⇒ Das neue Objekt wird mit allen enthaltenen Variablen im Informationsmodell angelegt. Das Mapping erfolgt automatisch.

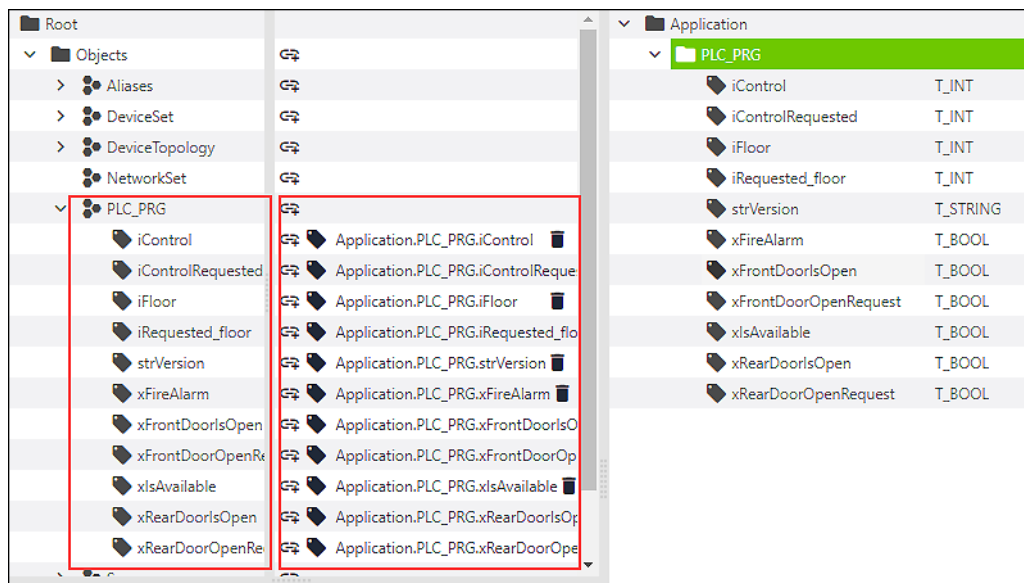


Abbildung 23: Angelegtes OPC-UA-Objekt mit Variablen

**Hinweis:** Objekte im OPC-UA-Informationsmodell von Objekttyp „Folder“ werden im Baum als Ordner dargestellt, sind aber grundsätzlich auch Objekte im Sinne der OPC-UA-Spezifikation. Somit macht es keinen Unterschied, ob Sie den Ordner aus der Symbolkonfiguration neben einen OPC-UA-Ordner oder neben ein anderes OPC-UA-Objekt ziehen.

### 7.7.4 Applikationsvariable manuell auf OPC-UA-Variable abbilden

Zusätzlich zu der Möglichkeit, eine Applikationsvariable per Drag & Drop mit einer OPC-UA-Variable zu verknüpfen, können Sie den Pfad der Applikationsvariable auch manuell eintragen.

1. Wählen Sie die OPC-UA-Variable im OPC-UA-Informationsmodell aus.
2. Öffnen Sie unter dem Informationsmodell das Register „Mapping“.
3. Wählen sie im Auswahlfeld „Mapping-Typ“ die Option „Mapping auf Applikationsvariable“ aus.
  - ⇒ **Hinweis:** Der Datentyp der OPC-UA-Variable dient an dieser Stelle nur als zusätzliche Information. Falls Sie ihn ändern möchten, verwenden Sie die Funktion „Instanz bearbeiten“.
4. Geben Sie im Feld „Symbolpfad“ den Pfad der Applikationsvariable in der Symbolkonfiguration an.

Abbildung 24: Applikationsvariable manuell auf OPC-UA-Variable abbilden

5. Um die Änderungen zu übernehmen, klicken Sie auf **[Bestätigen]**.
  - ⇒ Der eingegebene Wert wird im Bereich „Mapping“ neben der OPC-UA-Variablen angezeigt

**Hinweis:** Bei der Übernahme der Eingabe findet eine Prüfung der Gültigkeit des Pfades statt. Falls der eingegebene Pfad nicht auf eine gültige Variable in der Symbolkonfiguration verweist, wird ein Fehler angezeigt. Änderungen werden in dem Fall nicht übernommen.

### 7.7.5 Variablen statische Werte zuweisen

Für den Wert einer Variablen in einem OPC-UA-Informationsmodell kann ein statischer Wert angegeben werden.

1. Wählen Sie die OPC-UA-Variable im OPC-UA-Informationsmodell aus.
2. Öffnen Sie das Register „Mapping“.
3. Wählen Sie im Auswahlfeld „Mapping-Typ“ die Option „Mapping auf statischen Wert“ aus.
  - Hinweis:** Der Datentyp der OPC-UA-Variable dient an dieser Stelle nur als zusätzliche Information. Falls Sie ihn ändern möchten, verwenden Sie die Funktion „Instanz bearbeiten“.
4. Geben Sie den statischen Wert ein.
  - Hinweis:** Beachten Sie, dass der eingetragene Wert zum verwendeten Datentyp passen muss. Es erfolgt keine Prüfung auf Gültigkeit der Werte. Beachten Sie dazu die OPC UA Attribut-Referenz unter [🌐 https://reference.opcfoundation.org](https://reference.opcfoundation.org).

Abbildung 25: Statischen Wert eingeben


5. Um die Änderungen zu übernehmen, klicken Sie auf **[Bestätigen]**.  
⇒ Der eingegebene Wert wird im Bereich „Mapping“ neben der OPC-UA-Variablen angezeigt und mit dem Symbol  gekennzeichnet.

Abbildung 26: Variable im Bereich „Mapping“

### 7.7.6 Reguläre Ausdrücke zum Filtern der Applikationsvariablen verwenden

Der OPC-UA-Server bietet die Möglichkeit, OPC-UA-Variablen zu Laufzeit zu generieren und auf Applikationsvariablen abzubilden. Die abzubildenden Applikationsvariablen werden dabei über reguläre Ausdrücke aus allen vorhandenen Applikationsvariablen herausgefiltert.

Um eine solche Abbildungsvorschrift zu erstellen, gehen Sie wie folgt vor:

1. Wählen Sie ein Objekt (keine Variable) im Informationsmodell aus.
2. Öffnen Sie unter dem Informationsmodell das Register „Mapping“.  
⇒ Für OPC-UA-Objekte wird hier kontextsensitiv die Mapping-Konfiguration über reguläre Ausdrücke angeboten.

Abbildung 27: Reguläre Ausdrücke im Register „Mapping“

3. Konfigurieren Sie die regulären Ausdrücke, mit denen zur Laufzeit automatisch Variable unterhalb des ausgewählten OPC-UA-Objektes generiert werden (siehe auch nachfolgendes Beispiel).
4. Um die Änderungen zu übernehmen, klicken Sie auf **[Bestätigen]**.

### Beispiel für die Verwendung von regulären Ausdrücken

In diesem Beispiel bilden Sie alle Applikationsvariablen unterhalb einer OPC-UA-Variablen ab, deren Namen das Präfix „opc\_“ vorangestellt ist. Dabei soll unerheblich sein, ob diese in dem Programm „PLC\_PRG“ oder der globalen Variablenliste „GVL“ definiert sind. Als Ausgangspunkt des Beispiels dient die folgende Symbolkonfiguration:

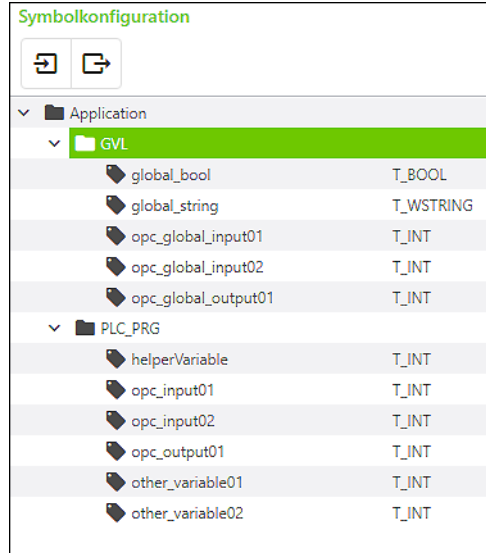


Abbildung 28: Symbolkonfiguration (Beispiel)

1. Legen Sie zunächst eine OPC-UA-Objektinstanz an (z. B. mit dem Namen „Reg-ExpMapping“). Für das Beispiel können Sie den Objekttyp auf dem Standardwert („BaseObjectType“) belassen.
2. Selektieren Sie das neu angelegte Objekt und bearbeiten Sie die regulären Ausdrücke im Register „Mapping“ unter dem Informationsmodell wie folgt:
3. **DataSourceIdentifizier**  
 Geben Sie unter „DataSourceIdentifizier“ einen regulären Ausdruck ein, mit dem die Symbolpfade aller zur Verfügung stehenden Applikationsvariablen gefiltert werden.  
**Hinweis:** Die Symbolpfade enthalten Punkte als Trennzeichen („.“). Stellen Sie im regulären Ausdruck diesem Punkt einen Backslash („\“) voran, da der Punkt sonst als Wildcard interpretiert wird.  
 (Es sei denn, Sie möchten den Punkt tatsächlich als Wildcard verwenden.)  
 ⇒ **Beispiel:** .\*\.opc\_.\*
4. **NodeId**  
 Für jeden bereits über den „DataSourceIdentifizier“ gefilterten Symbolpfad wird über die Eigenschaften „NodeId“ sowie „NodeIdTarget“ ein NodeId-String generiert. Geben Sie als Eigenschaft „NodeId“ einen regulären Ausdruck ein, der auf den bereits gefilterten Symbolpfad angewendet wird.  
 ⇒ **Beispiel:** .\*\. (.\*)\.opc\_ (.\*)

### 5. NodeldTarget

Geben Sie die Eigenschaft „NodeldTarget“ ein, um aus den Bestandteilen des regulären Ausdrucks unter „Nodeld“ den endgültigen Nodeld-Wert der OPC-UA-Variable zusammen zu setzen.

⇒ **Beispiel:** `nodeid_$1_$2`

Für den Symbolpfad `advanced.GVL.opc_global_input01` würde sich somit als „Nodeld“ `nodeid_GVL_global_input01` ergeben.

⇒ **Hinweis:** Beachten Sie, dass der Nodeld-String gemäß der OPC-UA-Spezifikation eindeutig sein muss. Um dies sicherzustellen, wird im Beispiel ein Teil des Symbolpfades (GVL bzw. PLC\_PRG) mit einbezogen, da sonst die Gefahr einer doppelten Nodeld bestünde.

### 6. BrowseName / BrowseNameTarget

Geben Sie die Eigenschaften „BrowseName“ und „BrowseNameTarget“ ein, um den OPC-UA-BrowseName aus dem bereits gefilterten Symbolpfad zu bilden.

⇒ **Beispiel:**

BrowseName: `*\.(.*)\.opc_(.*)`

BrowseNameTarget: `browsename_$1_$2`

Das Ergebnis für `advanced.GVL.opc_global_input01` lautet `browsename_GVL_global_input01`.

### 7. DisplayName / DisplayNameTarget

Geben Sie die Eigenschaften „DisplayName“ und „DisplayNameTarget“ ein, um den OPC-UA-DisplayName aus dem bereits gefilterten Symbolpfad zu bilden.

⇒ **Beispiel:**

DisplayName: `*\.(.*)\.opc_(.*)`

DisplayNameTarget: `$2 from $1`

Das Ergebnis für `advanced.GVL.opc_global_input01` lautet `global_input01 from GVL`.

### 8. Klicken Sie auf **[Bestätigen]**.

Namespaces	Attribute	Referenzen	Mapping
<b>Mapping: Regular expression</b>			
DataSourceIdentifier	<input type="text" value="*\.opc_*"/>		
Nodeld	<input type="text" value="*\.(\.*)\.opc_(*)"/>		
NodeldTarget	<input type="text" value="nodeid_\$1_\$2"/>		
BrowseName	<input type="text" value="*\.(\.*)\.opc_(*)"/>		
BrowseNameTarget	<input type="text" value="browsename_\$1_\$2"/>		
DisplayName	<input type="text" value="*\.(\.*)\.opc_(*)"/>		
DisplayNameTarget	<input type="text" value="\$2 from \$1"/>		
<input type="button" value="Bestätigen"/>			

Abbildung 29: Reguläre Ausdrücke verwenden

### 9. Das Ergebnis der durch reguläre Ausdrücke gefilterten Variablen können Sie sich mit einem OPC-Client ansehen (Ansicht hier beispielhaft für den OPC-UA-Client „UAexpert“).

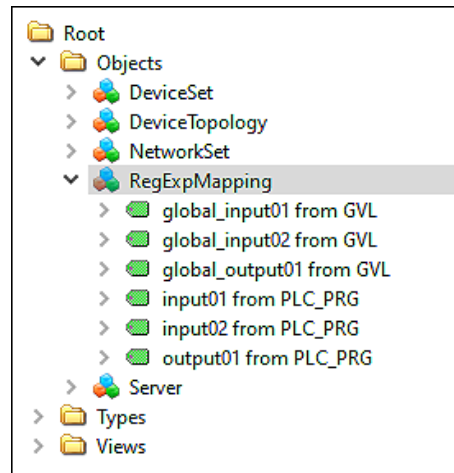


Abbildung 30: Trefferliste der regulären Ausdrücke (Ansicht im OPC UA Client „UAexpert“)

- Wenn Sie sich die Attribute ansehen, finden Sie beispielsweise hier die NodeId, den BrowseName und den DisplayName wieder. Diese wurden aus den regulären Ausdrücken gebildet, die Sie bei NodeID, NodeIdTarget etc. eingegeben haben.

Attribute	Value
▼ NodeId	ns=5;s=nodeid_GVL_global_input01
NamespaceIndex	5
IdentifierType	String
Identifier	nodeid_GVL_global_input01
NodeClass	Variable
BrowseName	5, "browsename_GVL_global_input01"
DisplayName	"", "global_input01 from GVL"
Description	BadAttributeValue (0x80350000)
WriteMask	0
UserWriteMask	0
RolePermissions	BadAttributeValue (0x80350000)
UserRolePermissions	BadAttributeValue (0x80350000)
AccessRestrictions	BadAttributeValue (0x80350000)
▼ Value	
SourceTimestamp	30.05.2021 16:52:17.139
SourcePicoseconds	0
ServerTimestamp	30.05.2021 16:52:17.139
ServerPicoseconds	0
StatusCode	Good (0x00000000)
Value	0
▼ DataType	Int16
NamespaceIndex	0
IdentifierType	Numeric
Identifier	4 [Int16]
ValueRank	-1 (Scalar)
ArrayDimensions	BadAttributeValue (0x80350000)
AccessLevel	CurrentRead, CurrentWrite
UserAccessLevel	CurrentRead, CurrentWrite
AccessLevelEx	BadAttributeValue (0x80350000)
MinimumSamplingInterval	BadAttributeValue (0x80350000)
Historizing	false

Abbildung 31: Ergebnis der regulären Ausdrücke (Ansicht im OPC UA Client „UAexpert“)

### **i Hinweis**

#### **Generieren der OPC-UA-Variablen erfolgt zur Laufzeit!**


Die OPC-UA-Variablen werden erst zur Laufzeit durch den OPC-UA-Server generiert. Aus diesem Grund werden diese im OPC UA Mapping Editor nicht unter dem ausgewählten OPC-UA-Objekt im Informationsmodell dargestellt.

### **i Hinweis**

#### **Filtern über reguläre Ausdrücke auch ohne Importieren einer Symbolkonfiguration möglich!**

Die regulären Ausdrücke werden durch den OPC-UA-Server zur Laufzeit immer auf die Symbolkonfiguration der gerade aktiven Applikation angewendet. Aus diesem Grund können Sie den OPC UA Mapping Editor auch direkt zur Eingabe regulärer Ausdrücke verwenden, ohne dass Sie eine Symbolkonfiguration importieren müssen. Sie exportieren lediglich das Mapping und laden es in den Controller. Die regulären Ausdrücke werden zur Laufzeit auf die jeweils aktive Applikation angewendet.

#### **7.7.7 Mapping löschen**

- Klicken Sie im Bereich „Mapping“ auf das Papierkorb-Symbol  des Mappings, welches Sie löschen möchten.

⇒ Das Mapping wird entfernt.

Alternativ wählen Sie die Variable im OPC-UA-Informationsmodell aus und wählen im Register „Mapping“ als Mapping-Typ „Kein Mapping“ aus. **[Bestätigen]** entfernt das Mapping.

#### **7.7.8 Mapping exportieren**

1. Klicken Sie auf die Schaltfläche  **[Mapping exportieren]**.

⇒ Ein Dialogfenster mit Exporteinstellungen wird geöffnet.

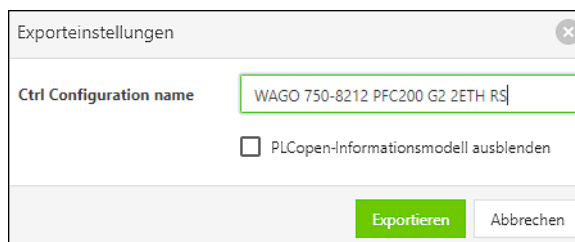


Abbildung 32: Mapping exportieren

2. Geben Sie als „Name der Konfiguration“ den Namen des Controllers ein, für welchen Sie das Mapping speichern.

#### **Hinweis: Identischen Namen wie im WBM verwenden!**

Sie finden den zu verwendenden Namen im WBM im Bereich „OPC-UA“, Feld „Ctrl Configuration name“. Verwenden Sie für den Export exakt diesen Namen!

3. Um das Mapping in einer XML-Datei zu speichern, klicken Sie auf **[Exportieren]**.

4. Wählen Sie den Speicherort aus und klicken Sie auf **[Speichern]**.

⇒ Der Controller kann die exportierte XML-Datei verwenden, um die Variablen seiner Applikation im modellierten Format darzustellen.

⇒ **Hinweis:** Die XML-Struktur des Exports entspricht der Struktur des OPC-UA-Informationsmodells, welches um die entsprechende Struktur aus der Symbolkonfiguration (PLCopen-konform) erweitert wurde.

Ist dies nicht gewünscht, sorgt das Aktivieren der Option „PLCopen-Informationsmodell ausblenden“ im Dialogfenster dafür, dass nur die Struktur des Informationsmodells verwendet wird.

## 7.8 Mapping in den Controller laden

Öffnen Sie das Web-Based-Management des Controllers.

1. Öffnen Sie die das Register „Fieldbus“ > Seite „OPC UA“.
2. Passen Sie bei Bedarf unter „Configuration“ die Konfiguration des Controllers an.
3. Aktivieren Sie beispielsweise die Optionen „Security Policy - None“ sowie „Trust all clients“, sofern ihr Client unverschlüsselt arbeitet oder führen Sie Änderungen an Zertifikaten durch (weitere Informationen zur Konfiguration, siehe Handbuch des PFC200, 750-821x).
4. Öffnen Sie die Seite „OPC UA“ > „Information Model“.
5. Damit der Controller das neue Informationsmodell verwendet, aktivieren Sie zunächst die Option „Feature enabled“.
6. Klicken Sie auf **[Submit]**.
7. Wählen Sie im darunterliegenden Feld mittels **[Upload]** die XML-Datei aus, die Sie über den OPC UA Mapping Editor exportiert haben.
  - ⇒ Der Server stellt die Daten nun in der neuen Struktur dar, die Sie nun auch in Ihrem Client wiederfinden.

## 7.9 Tastenkombinationen (Shortcuts) verwenden

Verwenden Sie Tastenkombinationen, um schnell auf oft benötigte Funktionen zuzugreifen.

- Um Tastenkombinationen einzugeben, drücken Sie die in der folgenden Liste angegebenen Tasten. Gleichzeitig zu drückende Tasten werden durch ein Pluszeichen (+) gekennzeichnet.

Tabella 9: Tastenkombinationen (Shortcuts)

Funktion	Tastenkombination
Neues Projekt	[STRG] + [N]
Projekt öffnen	[STRG] + [O]
Projekt speichern	[STRG] + [S]
Projekt speichern unter...	[ALT] + [STRG] + [S]
Beenden	[ALT] + [F4]
Basismodelle auswählen	[STRG] + [B]
Informationsmodell importieren	[STRG] + [I]
Informationsmodell exportieren	[STRG] + [E]
Informationsmodell Einstellungen	[STRG] + [Komma]
Instanz hinzufügen	[ALT] + [A]
Instanz bearbeiten	[ALT] + [E]
Instanz löschen	[ENTF]
Referenz hinzufügen	[ALT] + [L]
Hilfe anzeigen	[F1]

# Anhang

## 8.1 Übersicht und Mapping von OPC-UA- und CODESYS Datentypen

Um exportierte IEC-Variablen aus CODESYS im OPC-UA-Server abzubilden, muss die Kompatibilität der unterschiedlichen, verfügbaren Datentypen gewährleistet sein. In den folgenden beiden Tabellen werden Standard OPC-UA-Datentypen und Standard CODESYS Datentypen aufgeführt. In einer dritten Tabelle wird das Mapping zwischen OPC-UA- und CODESYS Datentypen dargestellt.

### Standard OPC-UA-Datentypen

Tabelle 10: Standard OPC-UA-Datentypen

Datentyp	Beschreibung	Anmerkung
<b>Logik-Typen</b>		
Boolean	Wahrheitswert	TRUE, FALSE
<b>Integrale Typen</b>		
SByte	Vorzeichenbehafteter 8 Bit Integer	[ -128, 127 ]
Byte	Vorzeichenloser 8 Bit Integer	[ 0, 255 ]
Int16	Vorzeichenbehafteter 16 Bit Integer	[ -2 <sup>15</sup> , 2 <sup>15</sup> - 1 ]
UInt16	Vorzeichenloser 16 Bit Integer	[ 0, 2 <sup>16</sup> - 1 ]
Int32	Vorzeichenbehafteter 32 Bit Integer	[ -2 <sup>31</sup> , 2 <sup>31</sup> - 1 ]
UInt32	Vorzeichenloser 32 Bit Integer	[ 0, 2 <sup>32</sup> - 1 ]
Int64	Vorzeichenbehafteter 64 Bit Integer	[ -2 <sup>63</sup> , 2 <sup>63</sup> - 1 ]
UInt64	Vorzeichenloser 64 Bit Integer	[ 0, 2 <sup>64</sup> - 1 ]
<b>Gleitkomma-Typen</b>		
Float	IEEE 754-1985 single precision floating point value	
Double	IEEE 754-1985 double precision data type	
<b>String-Typen</b>		
String	Unicode-String	
ByteString	Eine Folge von Byte-Werten	
<b>Zeit- und Datumstypen</b>		
DateTime	Ein Datum im gregorianischen Kalender mit Uhrzeit	
UtcTime	Zeitangabe in koordinierter Weltzeit (UTC)	Alle Zeitangaben zwischen OPC-UA-Servern und -Clients sind in UTC. Clients sollten Umwandlung in lokale Zeit anbieten.
Duration	Angabe einer Zeitdauer als Gleitkommazahl	<i>Double</i> -Wert, der ein Zeitintervall in Millisekunden angibt. Sub-Millisekunden-Angaben können durch Brüche dargestellt werden.
TimeString	Repräsentiert einen Zeitpunkt als String konform zu ISO 8601-2000	Verwendet 24-Stunden-Uhrzeit Einfaches Format: [hh][mm][ss] Erweitertes Format: [hh]:[mm]:[ss]
DurationString	Repräsentiert eine Zeitdauer als String konform zu ISO 8601-2000	Format: P[n]Y[n]M[n]DT[n]H[n]M[n]S „P“ (period) leitet die Angabe ein. Beispiel: „P3Y6M4DT12H30M5S“ = 3 Jahre, 6 Monate, 4 Tage, 12 Stunden, 30 Minuten, 5 Sekunden

Datentyp	Beschreibung	Anmerkung
<b>Benutzerdefinierte Typen</b>		
Enumeration	Aufzählung	
Structure	Struktur, Komposition aus Feldern	

## Standard CODESYS Datentypen

Tabelle 11: Standard CODESYS Datentypen

Datentyp	Beschreibung	Anmerkung
<b>Logik-Typen</b>		
BOOL	Wahrheitswert	TRUE, FALSE
BIT	1 Bit Logikwert	1, 0
BYTE	8 Bit Logikwert	[ 16#00, 16#FF ]
WORD	16 Bit Logikwert	[ 16#0000, 16#FFFF ]
DWORD	32 Bit Logikwert	[ 16#00000000, 16#FFFFFFFF ]
LWORD	64 Bit Logikwert	[ 16#0000000000000000, 16#FFFFFFFFFFFFFFFF ]
<b>Integrale Typen</b>		
SINT	Vorzeichenbehafteter 8-Bit-Integer	[ -128, 127 ]
USINT	Vorzeichenloser 8-Bit-Integer	[ 0, 255 ]
INT	Vorzeichenbehafteter 16-Bit-Integer	[ -2 <sup>15</sup> , 2 <sup>15</sup> - 1 ]
UINT	Vorzeichenloser 16-Bit-Integer	[ 0, 2 <sup>16</sup> - 1 ]
DINT	Vorzeichenbehafteter 32-Bit-Integer	[ -2 <sup>31</sup> , 2 <sup>31</sup> - 1 ]
UDINT	Vorzeichenloser 32-Bit-Integer	[ 0, 2 <sup>32</sup> - 1 ]
LINT	Vorzeichenbehafteter 64-Bit-Integer	[ -2 <sup>63</sup> , 2 <sup>63</sup> - 1 ]
ULINT	Vorzeichenloser 64-Bit-Integer	[ 0, 2 <sup>64</sup> - 1 ]
<b>Gleitkomma-Typen</b>		
REAL	Gleitkommazahl, 32 Bit	IEC 60559 bzw. IEEE 754-2008
LREAL	Gleitkommazahl, 64 Bit	IEC 60559 bzw. IEEE 754-2008
<b>String-Typen</b>		
STRING	Single-Byte-Zeichenkette	
WSTRING	Double-Byte-Zeichenkette	
<b>Zeit- und Datumstypen</b>		
DATE	Datum	Präfix: DATE# oder D# Beispiele: DATE#1984-06-25 oder D#1984-06-25
TIME	Zeitdauer	Präfix: TIME# oder T# Beispiele: TIME#14ms T#14.7m T#5d_14h_12m_18s_3.5ms
LTIME	Lange Zeitdauer	
TIME_OF_DAY / TOD	Tageszeit	Präfix: TIME_OF_DAY# oder TOD# Beispiel: TIME_OF_DAY#15:36:30.123 oder TOD#15:36:30.123
DATE_AND_TIME / DT	Datumsangabe mit Uhrzeit	Präfix: DATE_AND_TIME# oder DT# Beispiel: DATE_AND_TIME#1996-05-06-15:36:30 oder DT#1996-05-06-15: 36:30

### Besonderheiten

- Es kann eine direkte Ableitung von elementaren Typen erfolgen.  
Beispiel: `TYPE RU_REAL : REAL ; END_TYPE`
- Es können Untermengen (*Subranges*) von Typen verwendet werden.  
Beispiel: `TYPE ANALOG_DATA : INT (-4095..4095) ; END_TYPE`

### Program Organization Units nach IEC 61131-3

- Funktionen
- Funktionsblöcke
- Programme

### Mapping zwischen OPC-UA- und CODESYS Datentypen

Die folgende Tabelle zeigt das Mapping zwischen den Datentypen, die für CODESYS und OPC UA verwendet werden.

Tabelle 12: Mapping zwischen OPC-UA- und CODESYS Datentypen

CODESYS Datentyp	OPC-UA-Datentyp
BOOL	Boolean
BIT	Boolean
BYTE	Byte
WORD	UInt16
DWORD	UInt32
LWORD	UInt64
SINT	SByte
USINT	Byte
INT	Int16
UINT	UInt16
DINT	Int32
UDINT	UInt32
LINT	Int64
ULINT	UInt64
REAL	Float
LREAL	Double
STRING	String
WSTRING	String
DATE	DateTime
TIME	Int64
LTIME	Int64
TIME_OF_DAY / TOD	DateTime
DATE_AND_TIME / DT	DateTime
ENUM	Enumeration
STRUCT	Structure

## 8.2 Schutzrechte

- Adobe® und Acrobat® sind eingetragene Marken der Adobe Systems Inc.
- Android™ ist eine Marke von Google LLC.
- Apple, das Apple-Logo, iPhone, iPad und iPod touch sind eingetragene Marken von Apple Inc., registriert in den U.S.A. und anderen Staaten. "App Store" ist eine Dienstleistungsmarke von Apple Inc.
- AS-Interface® ist eine eingetragene Marke der AS-International Association e.V.
- BACnet® ist eine eingetragene Marke der American Society of Heating, Refrigerating and Air Conditioning Engineers, Inc. (ASHRAE).
- *Bluetooth*® ist ein registriertes Warenzeichen der Bluetooth SIG, Inc.
- CiA® und CANopen® sind eingetragene Marken des CAN in AUTOMATION – International Users and Manufacturers Group e. V.
- CODESYS ist eine eingetragene Marke der CODESYS Development GmbH.
- DALI ist eine eingetragene Marke der Digital Illumination Interface Alliance (DiiA).
- EtherCAT® ist eine eingetragene Marke und patentierte Technologie, lizenziert durch die Beckhoff Automation GmbH, Deutschland
- EtherNet/IP™ ist eine eingetragene Marke der Open DeviceNet Vendor Association, Inc (ODVA).
- EnOcean® ist eine eingetragene Marke der EnOcean GmbH.
- Google Play™ ist ein eingetragenes Markenzeichen von Google Inc.
- IO-Link ist eine eingetragene Marke der PROFIBUS Nutzerorganisation e.V.
- KNX® ist eine eingetragene Marke der KNX Association cvba.
- Linux® ist eine eingetragene Marke von Linus Torvalds.
- LON® ist eine eingetragene Marke der Echelon Corporation.
- Modbus® ist eine registrierte Marke der Schneider Electric, lizenziert für die Modbus Organization, Inc.
- OPC UA ist eine registrierte Marke der OPC Foundation.
- PROFIBUS® ist eine registrierte Marke der PROFIBUS Nutzerorganisation e.V. (PNO).
- PROFINET® ist eine registrierte Marke der PROFIBUS Nutzerorganisation e.V. (PNO).
- QR Code ist eine registrierte Marke von DENSO WAVE INCORPORATED.
- Subversion® ist eine Marke der Apache Software Foundation.
- Windows® ist eine registrierte Marke der Microsoft Corporation.

# Abbildungsverzeichnis

Abbildung 1	Grafische Benutzeroberfläche .....	14
Abbildung 2	Bereich „Informationsmodell“ .....	16
Abbildung 3	Auf das Informationsmodell abgebildete Variablen im Bereich „Mapping“ .....	18
Abbildung 4	Register „Namespaces“ .....	19
Abbildung 5	Register „Attribute“ .....	19
Abbildung 6	Register „Referenzen“ .....	19
Abbildung 7	Register „Mapping“ (hier bei Auswahl einer Variable) .....	20
Abbildung 8	Register „Mapping“ (hier bei Auswahl eines Objektes) .....	20
Abbildung 9	Bereich „Symbolkonfiguration“ .....	22
Abbildung 10	Eigenschaften zur Symbolkonfiguration .....	23
Abbildung 11	Grundstruktur Informationsmodell .....	24
Abbildung 12	Dialogfenster „Basismodelle auswählen“ .....	25
Abbildung 13	Instanz hinzufügen .....	26
Abbildung 14	Instanz bearbeiten .....	27
Abbildung 15	Referenz hinzufügen .....	28
Abbildung 16	Neu angelegte Referenz .....	28
Abbildung 17	Referenz löschen .....	29
Abbildung 18	Symbolkonfiguration importieren .....	30
Abbildung 19	Direktes Mapping .....	31
Abbildung 20	Variable in den Bereich „Mapping“ ziehen .....	32
Abbildung 21	Neu erstellte OPC-UA-Variable .....	32
Abbildung 22	Ganzen Ordner auf ein OPC-UA-Objekt ziehen .....	33
Abbildung 23	Angelegtes OPC-UA-Objekt mit Variablen .....	33
Abbildung 24	Applikationsvariable manuell auf OPC-UA-Variable abbilden .....	34
Abbildung 25	Statischen Wert eingeben .....	35
Abbildung 26	Variable im Bereich „Mapping“ .....	35
Abbildung 27	Reguläre Ausdrücke im Register „Mapping“ .....	35
Abbildung 28	Symbolkonfiguration (Beispiel) .....	36
Abbildung 29	Reguläre Ausdrücke verwenden .....	37
Abbildung 30	Trefferliste der regulären Ausdrücke (Ansicht im OPC UA Client „UAexpert“) .....	38
Abbildung 31	Ergebnis der regulären Ausdrücke (Ansicht im OPC UA Client „UAexpert“) .....	38
Abbildung 32	Mapping exportieren .....	39

---

# Tabellenverzeichnis

Tabelle 1	Systemvoraussetzung.....	12
Tabelle 2	Grafische Benutzeroberfläche .....	14
Tabelle 3	Menü „Datei“ .....	15
Tabelle 4	Funktionen im Bereich „Informationsmodell“ .....	16
Tabelle 5	Funktionen im Bereich „Mapping“ .....	18
Tabelle 6	Anzeige und Auswahl des Mapping-Typs.....	20
Tabelle 7	Einstellungen für reguläre Ausdrücke .....	21
Tabelle 8	Funktionen im Bereich „Symbolkonfiguration“ .....	22
Tabelle 9	Tastenkombinationen (Shortcuts).....	41
Tabelle 10	Standard OPC-UA-Datentypen.....	42
Tabelle 11	Standard CODESYS Datentypen .....	43
Tabelle 12	Mapping zwischen OPC-UA- und CODESYS Datentypen .....	44

**WAGO GmbH & Co. KG**  
Postfach 2880 · 32385 Minden  
Hansastraße 27 · D-32423 Minden  
✉ [info@wago.com](mailto:info@wago.com)  
🌐 [www.wago.com](http://www.wago.com)

Zentrale	+49 (0) 571/887 – 0
Vertrieb	+49 (0) 571/887 – 44 222
Auftragsservice	+49 (0) 571/887 – 44 333
Fax	+49 (0) 571/887 – 844 169

WAGO ist eine eingetragene Marke der WAGO Verwaltungsgesellschaft mbH.  
Copyright – WAGO GmbH & Co. KG – Alle Rechte vorbehalten. Inhalt und Struktur der WAGO Websites, Kataloge, Videos und andere WAGO Medien unterliegen dem Urheberrecht. Die Verbreitung oder Veränderung des Inhalts dieser Seiten und Videos ist nicht gestattet. Des Weiteren darf der Inhalt weder zu kommerziellen Zwecken kopiert, noch Dritten zugänglich gemacht werden. Dem Urheberrecht unterliegen auch die Bilder und Videos, die der WAGO GmbH & Co. KG von Dritten zur Verfügung gestellt wurden.