



SMART
SAFETY.

Handbuch

OPC UA-Server®

SILworX



Alle in diesem Handbuch genannten HIMA Produkte sind mit dem Warenzeichen geschützt. Dies gilt ebenfalls, soweit nicht anders vermerkt, für weitere genannte Hersteller und deren Produkte.

HIQuad®, HIQuad X®, HIMax®, HIMatrix®, SILworX®, XMR®, HICore® und FlexSILon® sind eingetragene Warenzeichen der HIMA Paul Hildebrandt GmbH.

Alle technischen Angaben und Hinweise in diesem Handbuch wurden mit größter Sorgfalt erarbeitet und unter Einschaltung wirksamer Kontrollmaßnahmen zusammengestellt. Bei Fragen bitte direkt an HIMA wenden. Für Anregungen, z. B. welche Informationen noch in das Handbuch aufgenommen werden sollen, ist HIMA dankbar.

Technische Änderungen vorbehalten. Ferner behält sich HIMA vor, Aktualisierungen des schriftlichen Materials ohne vorherige Ankündigungen vorzunehmen.

Alle aktuellen Handbücher können über die E-Mail-Adresse documentation@hima.com angefragt werden.

© Copyright 2020, HIMA Paul Hildebrandt GmbH

Alle Rechte vorbehalten.

Kontakt

HIMA Paul Hildebrandt GmbH

Postfach 1261

68777 Brühl

Tel.: +49 6202 709-0

Fax: +49 6202 709-107

E-Mail: info@hima.com

Revision	Änderung	Art der Änderung	
		Technisch	Redaktionell
12.00	Erstausgabe des OPC UA-Server Handbuchs		

Inhaltsverzeichnis

1 Einleitung	6
1.1 Aufbau und Gebrauch des Handbuchs	7
1.1.1 Zielgruppe und erforderliche Kompetenz	7
2 Sicherheit	8
2.1 Bestimmungsgemäßer Einsatz	8
2.2 Restrisiken	8
2.3 Sicherheitsvorkehrungen	8
2.4 Notfallinformation	8
2.5 Automation Security bei HIMA Systemen	9
2.5.1 Produkteigenschaften	9
2.5.2 Risikoanalyse und Planung	10
2.6 Darstellungskonventionen	10
2.6.1 Sicherheitshinweise	11
2.6.2 Gebrauchshinweise	11
2.7 Safety Lifecycle Services	12
3 OPC Unified Architecture	13
3.1 Systemanforderungen und Mengengerüst für OPC UA-Server	13
4 Konzept des HIMA OPC UA-Servers	15
4.1 Redundanz von OPC UA-Servern	17
5 Konfiguration eines HIMA OPC UA-Servers	18
5.1 Konfiguration in SILworX	18
5.2 Namensraum konfigurieren	19
5.2.1 OPC UA-Objekt im Namensraum	19
5.2.1.1 Erzeugen von OPC UA-Objekten	19
5.2.1.2 Bearbeiten von OPC UA-Objekten	20
5.2.2 OPC UA-Typen im Namensraum	20
5.2.2.1 Erzeugen von OPC UA-Typen	20
5.2.2.2 Bearbeiten von OPC UA-Typen	20
5.2.3 OPC UA-Variablen im Namensraum	20
5.2.3.1 Erzeugen von OPC UA-Variablen	20

5.2.3.2 Erzeugen von OPC UA-Variablen per Drag&Drop	21
5.2.3.3 Datentyp der OPC UA-Variablen auswählen	21
6 OPC UA-Set	22
6.1 OPC UA-Set-Editor	22
6.1.1 Register Set-Elemente	22
6.1.2 Register Eigenschaften	22
6.2 COM-Referenz	23
6.2.1 Register Eigenschaften	23
6.3 Namensraum-Editor	24
6.3.1 Register Objekte und Typen	24
6.3.2 Register Eigenschaften	26
6.4 Typ-Referenz (im Namensraum)	27
6.4.1 Editieren der Typ-Referenz	27
6.4.2 Bearbeiten von OPC UA-Variablen	27
6.4.3 Editieren der Globale-Variable-Referenz	27
6.4.4 Fehlerhafte Typ-Referenzen	28
6.4.4.1 Ungültige Typ-Referenzen	28
6.4.4.2 Zirkelschlüsse	28
6.4.4.3 Zirkelschluss in der Typ-Verwendung	28
6.4.4.4 Zirkelschluss in der Typ-Ableitung von Typen	29
6.5 Zertifikate	30
6.5.1 Zertifikate-Editor einer Ressource anlegen	32
6.5.1.1 Server-Zertifikat laden	32
6.5.1.2 Client-Zertifikat laden	33
7 Alarm&Events	34
7.1 Alarm&Events einer Ressource aktivieren	34
7.2 Daten einer an einen OPC UA-Client ausgelieferten Event-Notifikation	35
7.3 Alarm&Events-Editor	36
7.3.1 Anzeige im OPC Client	37
8 Control-Panel (Online)	38
8.1 Trace-Logging (Online)	39
8.2 Diagnose des OPC UA-Servers	40

8.2.1 Öffnen einer Session	40
8.2.2 Aktivieren einer Session	40
8.2.3 Schließen einer Session	40
8.2.4 Erstellen einer Subscription	41
8.2.5 Löschen einer Subscription	41
9 Codegenerierung und Reload	42
9.1 Codegenerierung	42
9.2 Reload	42
9.3 Zertifikate	42
10 Versionsvergleich	43
10.1 OPC UA-Server	43
10.1.1 Sektion OPC UA	43
10.1.2 Sektion Namespace	43
10.1.3 Sektion Node	43
10.1.4 Sektion Variable	44
10.1.5 Sektion Reference	44
10.1.6 Sektion EventSource	44
10.1.7 Sektion Condition	44
10.2 OPC UA-Zertifikate	45
10.2.1 Sektion Own	45
10.2.2 Sektion Client	45

1 Einleitung

Dieses Handbuch beschreibt die Konfiguration des OPC UA-Servers in SILworX für die Steuerungen der Systemfamilien HIMax, HIMatrix und HIQuad X.

Voraussetzung für die risikolose Installation und Inbetriebnahme sowie für die Sicherheit bei Betrieb und Instandhaltung des Systems sind:

- Die Kenntnis von Vorschriften.
- Die technisch einwandfreie Umsetzung der in diesem Handbuch enthaltenen Sicherheitshinweise durch qualifiziertes Personal.

Durch Störungen oder Beeinträchtigungen von Sicherheitsfunktionen können in folgenden Fällen schwere Personen-, Sach- oder Umweltschäden eintreten, für die HIMA keine Haftung übernehmen kann:

- Bei nicht qualifizierten Eingriffen in die Systeme.
- Bei Abschalten oder Umgehen (Bypass) von Sicherheitsfunktionen.
- Bei Nichtbeachtung von Hinweisen dieses Handbuchs.

HIMA entwickelt, fertigt und prüft die HIMA Systeme unter Beachtung der einschlägigen Sicherheitsnormen. Die Verwendung der Systeme ist nur zulässig, wenn die folgenden Voraussetzungen erfüllt sind:

- Die in den Beschreibungen vorgesehenen Einsatzfälle wurden eingehalten.
- Die spezifizierten Umgebungsbedingungen wurden eingehalten.

1.1 Aufbau und Gebrauch des Handbuchs

Das Handbuch enthält die folgenden Hauptkapitel:

- Einleitung und Darstellungskonventionen
- Sicherheit
- Produktbeschreibung

Zusätzlich sind die folgenden Dokumente zu beachten:

Name	Inhalt	Dokumenten-Nr.
HIMax Systemhandbuch	Hardware-Beschreibung HIMax System	HI 801 000 D
HIMax Sicherheitshandbuch	Sicherheitsfunktionen HIMax System	HI 801 002 D
HIMatrix Sicherheitshandbuch	Sicherheitsfunktionen HIMatrix System	HI 800 022 D
HIMatrix Kompakt Systemhandbuch	Hardware-Beschreibung HIMatrix System	HI 800 140 D
HIMatrix Modular Systemhandbuch	Hardware-Beschreibung HIMatrix System	HI 800 190 D
HIQuad X Systemhandbuch	Hardware-Beschreibung HIQuad X System	HI 803 210 D
HIQuad X Sicherheitshandbuch	Sicherheitsfunktionen HIQuad X System	HI 803 208 D
Kommunikationshandbuch	Beschreibung Kommunikationsprotokolle	HI 801 100 D
Erste Schritte SILworX	Einführung in SILworX	HI 801 102 D

Die aktuellen Handbücher können über die E Mail Adresse documentation@hima.com angefragt werden. Für registrierte Kunden stehen die Produktdokumentationen im HIMA Extranet als Download zur Verfügung.

1.1.1 Zielgruppe und erforderliche Kompetenz

Dieses Dokument wendet sich an Planer, Projektoren, Programmierer und Personen, die zur Inbetriebnahme, zur Wartung und zum Betreiben von Automatisierungsanlagen berechtigt sind. Vorausgesetzt werden spezielle Kenntnisse auf dem Gebiet der sicherheitsbezogenen Automatisierungssysteme.

2 Sicherheit

Sicherheitsinformationen, Hinweise und Anweisungen in diesem Dokument unbedingt lesen. Die HIMA Systeme nur unter Beachtung aller Richtlinien und Sicherheitsrichtlinien einsetzen.

Die HIMA Steuerungen werden mit SELV oder PELV betrieben. Von diesen Steuerungen selbst geht kein Risiko aus. Der Einsatz im Ex-Bereich ist nur mit zusätzlichen Maßnahmen erlaubt.

2.1 Bestimmungsgemäßer Einsatz

Für den Einsatz von HIMA Systemen, sind die jeweiligen Bedingungen einzuhalten, siehe Handbücher in Kapitel 1.1.

2.2 Restrisiken

Von einem HIMA System selbst geht kein Risiko aus.

Restrisiken können ausgehen von:

- Fehlern in der Projektierung.
- Fehlern im Anwenderprogramm.
- Fehlern in der Verdrahtung.

2.3 Sicherheitsvorkehrungen

Am Einsatzort geltende Sicherheitsbestimmungen beachten und vorgeschriebene Schutzausrüstung tragen.

2.4 Notfallinformation

Ein HIMA System ist Teil der Sicherheitstechnik einer Anlage. Der Ausfall einer Steuerung bringt die Anlage in den sicheren Zustand.

Im Notfall ist jeder Eingriff, der die Sicherheitsfunktion der HIMA Systeme verhindert, verboten.

2.5 Automation Security bei HIMA Systemen

HIMA unterscheidet zwischen den Begriffen *Safety* im Sinne der funktionalen Sicherheit und *Security* im Sinne von Schutz eines Systems vor Manipulationen.

Industrielle Steuerungen (PES) müssen gegen IT-typische Problemquellen geschützt werden, z. B.:

- Unzureichender Schutz von IT-Einrichtungen (z. B. offenes WLAN, veraltete Betriebssysteme).
- Fehlendes Bewusstsein für den richtigen Umgang mit Betriebsmitteln (z. B. USB-Stick).
- Direkte Zugänge zu schützenswerten Bereichen.
- Angreifer innerhalb von Betriebsgeländen.
- Angreifer über Kommunikations-Netzwerke innerhalb und außerhalb von Betriebsgeländen.

HIMA Safety-Systeme bestehen aus folgenden zu schützenden Teilen:

- Sicherheitsbezogenes Automatisierungssystem.
- PADT.
- Optionale X-OPC Server (auf einem Host-PC).
- Optionale Kommunikationsverbindungen zu externen Systemen.

2.5.1 Produkteigenschaften

HIMA Systeme erfüllen bereits in den Grundeinstellungen Anforderungen an Automation Security.

In Steuerungen und im Programmierwerkzeug sind Schutzmechanismen integriert, die versehentliche oder nicht genehmigte Veränderungen verhindern:

- Jede Änderung am Anwenderprogramm oder an der Konfiguration einer Steuerung führt zu einem neuen Konfigurations-CRC.
- In der Steuerung können Online-Änderungen der Sicherheitsparameter deaktiviert werden. Dadurch sind Änderungen der Sicherheitsparameter nur durch Download oder Reload möglich.
- Der Anwender kann eine Benutzerverwaltung einrichten, um die Security zu erhöhen. Hier werden Benutzergruppen, Benutzerkonten, Zugriffsrechte für das PADT und für die Steuerungen (PES) projektbezogen festgelegt. In einer Benutzerverwaltung kann der Anwender definieren, ob für das Öffnen des Projekts und für den Login in eine Steuerung eine Autorisierung erforderlich ist.
- Der Zugang zu Daten einer Steuerung ist nur dann möglich, wenn im PADT das gleiche Anwenderprojekt geladen wurde wie in der Steuerung. Die CRCs müssen identisch sein (Archiv-Pflege!).
- Eine physikalische Verbindung zwischen einem PADT und einer Steuerung (PES) ist im Betrieb nicht notwendig und muss aus Gründen der Security getrennt werden. Das PADT kann für Diagnose- und Wartungszwecke erneut mit der Steuerung verbunden werden.

Die Anforderungen der Normen für Safety und Security sind zu beachten. Die Autorisierung von Personal und die notwendigen Schutzmaßnahmen unterliegen der Verantwortung des Betreibers.

WARNUNG

Personenschaden durch unbefugte Manipulationen an Steuerungen möglich!



Steuerungen sind gegen unbefugte Zugriffe zu schützen:

- Standardeinstellungen für Logins und Passworte sind zu ändern.
- Zugänge zu Steuerungen und PADTs sind zu kontrollieren!
- Weitere Schutzmaßnahmen entnehmen Sie dem Automation Security Handbuch (HI 801 372 D).

2.5.2 Risikoanalyse und Planung

Security ist kein Produkt, sondern ein Prozess. So helfen z. B. gepflegte Netzwerkpläne sicherzustellen, dass sichere Netzwerke dauerhaft von öffentlichen Netzwerken getrennt sind. Sinnvollerweise sollte nur ein definierter Übergang über eine Firewall oder ein eigenständiges Subnetz bestehen.

Eine sorgfältige Planung nennt die erforderlichen Maßnahmen. Nach erfolgter Risikoanalyse sind die benötigten Maßnahmen zu ergreifen, wie z. B.:

- Zugriffsrechte für Benutzergruppen und Benutzerkonten gemäß den vorgesehenen Aufgaben zuweisen.
- Passwörter verwenden, die den Anforderungen an die Security entsprechen.

Ein regelmäßiges Review (z. B. jährlich) der Security-Maßnahmen ist erforderlich.

 **Die für eine Anlage geeignete Umsetzung der benötigten Maßnahmen liegt in der Verantwortung des Betreibers!**

Weitere Informationen finden Sie im HIMA Automation Security Handbuch HI 801 372 D.

2.6 Darstellungskonventionen

Zur besseren Lesbarkeit und zur Verdeutlichung gelten in diesem Dokument folgende Schreibweisen:

Format	Beschreibung
Fett	Hervorhebung wichtiger Textteile. Bezeichnungen von Schaltflächen, Menüpunkten und Registern im Programmierwerkzeug, die angeklickt werden können.
<i>Kursiv</i>	Parameter, Systemvariablen und Referenzen.
<code>Courier</code>	Wörtliche Benutzereingaben.
RUN	Bezeichnungen von Betriebszuständen (Großbuchstaben).
Kapitel 1.2.3	Querverweise sind Hyperlinks, auch wenn sie nicht besonders gekennzeichnet sind. Im elektronischen Dokument (PDF): Wird der Mauszeiger auf einen Hyperlink positioniert, verändert er seine Gestalt. Bei einem Klick springt das Dokument zur betreffenden Stelle.

Sicherheits- und Gebrauchshinweise sind besonders gekennzeichnet.

2.6.1 Sicherheitshinweise

Um ein möglichst geringes Risiko zu gewährleisten, sind die Sicherheitshinweise unbedingt zu befolgen.



Die Sicherheitshinweise werden wie folgt dargestellt.

- Signalwort: Warnung, Vorsicht, Hinweis.
- Art und Quelle des Risikos.
- Folgen bei Nichtbeachtung.
- Vermeidung des Risikos.


Die Bedeutung der Signalworte ist:

- Warnung: Bei Missachtung droht schwere Körperverletzung bis Tod.
- Vorsicht: Bei Missachtung droht leichte Körperverletzung.
- Hinweis: Bei Missachtung droht Sachschaden.

SIGNALWORT

-  **Art und Quelle des Risikos!**
-  **Folgen bei Nichtbeachtung.**
- Vermeidung des Risikos.**

HINWEIS

-  **Art und Quelle des Schadens!**
- Vermeidung des Schadens.**

2.6.2 Gebrauchshinweise

Zusatzinformationen sind nach folgendem Beispiel aufgebaut:



An dieser Stelle steht der Text der Zusatzinformation.

2.7 Safety Lifecycle Services

HIMA unterstützt Sie in allen Phasen des Sicherheitslebenszyklus einer Anlage: Von der Planung, der Projektierung, über die Inbetriebnahme, bis zur Aufrechterhaltung der Sicherheit.

Für Informationen und Fragen zu unseren Produkten, zu Funktionaler Sicherheit und zu Automation Security stehen Ihnen die Experten des HIMA Support zur Verfügung.

Für die geforderte Qualifizierung gemäß Sicherheitsstandards führt HIMA produkt- oder kundenspezifische Seminare in eigenen Trainingszentren oder bei Ihnen vor Ort durch. Das aktuelle Seminarangebot zu Funktionaler Sicherheit, Automation Security und zu HIMA Produkten finden Sie auf der HIMA Webseite.

Safety Lifecycle Services:	
Onsite+ / Vor-Ort-Engineering	In enger Abstimmung mit Ihnen führt HIMA vor Ort Änderungen oder Erweiterungen durch.
Startup+ / Vorbeugende Wartung	HIMA ist verantwortlich für die Planung und Durchführung der vorbeugenden Wartung. Wartungsarbeiten erfolgen gemäß der Herstellervorgabe und werden für den Kunden dokumentiert.
Lifecycle+ / Lifecycle-Management	Im Rahmen des Lifecycle-Managements analysiert HIMA den aktuellen Status aller installierten Systeme und erstellt konkrete Empfehlungen für Wartung, Upgrade und Migration.
Hotline+ / 24-h-Hotline	HIMA Sicherheitsingenieure stehen Ihnen für Problemlösung rund um die Uhr telefonisch zur Verfügung.
Standby+ / 24-h-Rufbereitschaft	Fehler, die nicht telefonisch gelöst werden können, werden von HIMA Spezialisten innerhalb vertraglich festgelegter Zeitfenster bearbeitet.
Logistic+ / 24-h-Ersatzteilservice	HIMA hält notwendige Ersatzteile vor und garantiert eine schnelle und langfristige Verfügbarkeit.

Ansprechpartner:	
Safety Lifecycle Services	https://www.hima.com/de/unternehmen/ansprechpartner-weltweit/
Technischer Support	https://www.hima.com/de/produkte-services/support/
Seminarangebot	https://www.hima.com/de/produkte-services/seminarangebot/

3 OPC Unified Architecture

OPC UA steht für *Open Platform Communications Unified Architecture* und ist für die Kommunikation in der Industrieautomation und der Konnektivität zwischen Produkten verschiedener Hersteller spezifiziert.

3.1 Systemanforderungen und Mengengerüst für OPC UA-Server

Element	Beschreibung
Programmierwerkzeug	SILworX ab V12
Steuerung und Betriebssystem	<ul style="list-style-type: none"> HIMax CPU BS \geq V13, COM BS \geq V13 HIMatrix CPU BS \geq V17, COM BS \geq V22 HIQuad X CPU BS \geq V13, COM BS \geq V13
Lizenz	Eine Lizenz für die Freischaltung des OPC UA-Server-Protokolls pro COM-Modul, auf dem OPC UA laufen soll.
Anzahl OPC UA-Server	<p>Die Anzahl der OPC UA-Server auf einer Steuerung ist abhängig von der Anzahl der Kommunikationsmodule. Auf jedem Kommunikationsmodul kann ein OPC UA-Server betrieben werden:</p> <ul style="list-style-type: none"> HIMax: 20 (mit 20 COM-Modulen) HIMatrix: 1 (1 COM-Modul fest eingebaut) HIQuad H51X: 10 (mit 10 COM-Modulen) HIQuad H41X: 2 (mit 2 COM-Modulen)
Anzahl OPC UA Clients	Der OPC UA-Server kann parallel bis zu 4 OPC UA-Client-Sessions betreiben.
Sicherheitsbezogen	Der OPC UA-Server ermöglicht HIMA Steuerungen den Prozessdaten-Austausch mit Fremdsystemen, die über eine OPC UA-Client-Funktionalität verfügen. Der OPC UA-Server darf dabei nicht zur sicherheitsbezogenen Kommunikation verwendet werden.
Automation-Security	<p>Die Automation-Security ist ein zentraler Bestandteil der OPC-UA-Spezifikationen.</p> <p>Der OPC UA-Server unterstützt folgende Security-Profile:</p> <ul style="list-style-type: none"> SecurityPolicy - None. SecurityPolicy [B] Basic256Sha256 <p>Der OPC UA-Server bietet in Abhängigkeit der Konfiguration (Server-Zertifikat und Schalters <i>Nur verschlüsselte Verbindungen zulassen</i>) verschiedene Endpunkte (Endpoints) für OPC UA-Verbindungen an:</p> <ul style="list-style-type: none"> None - None (Endpunkt für unverschlüsselte OPC UA-Verbindungen) Bedingung: Schalter <i>Nur verschlüsselte Verbindungen zulassen</i> deaktiviert. Basic256Sha256 - Sign (Endpunkt für OPC UA-Verbindungen mit Signatur-Prüfung) Bedingung: Schalter <i>Nur verschlüsselte Verbindungen zulassen</i> deaktiviert und ein konfiguriertes Server-Zertifikat. Basic256Sha256 - Sign & Encrypt (Endpunkt für verschlüsselte OPC UA-Verbindungen mit Signatur-Prüfung). Bedingung: Ein konfiguriertes Server-Zertifikat.

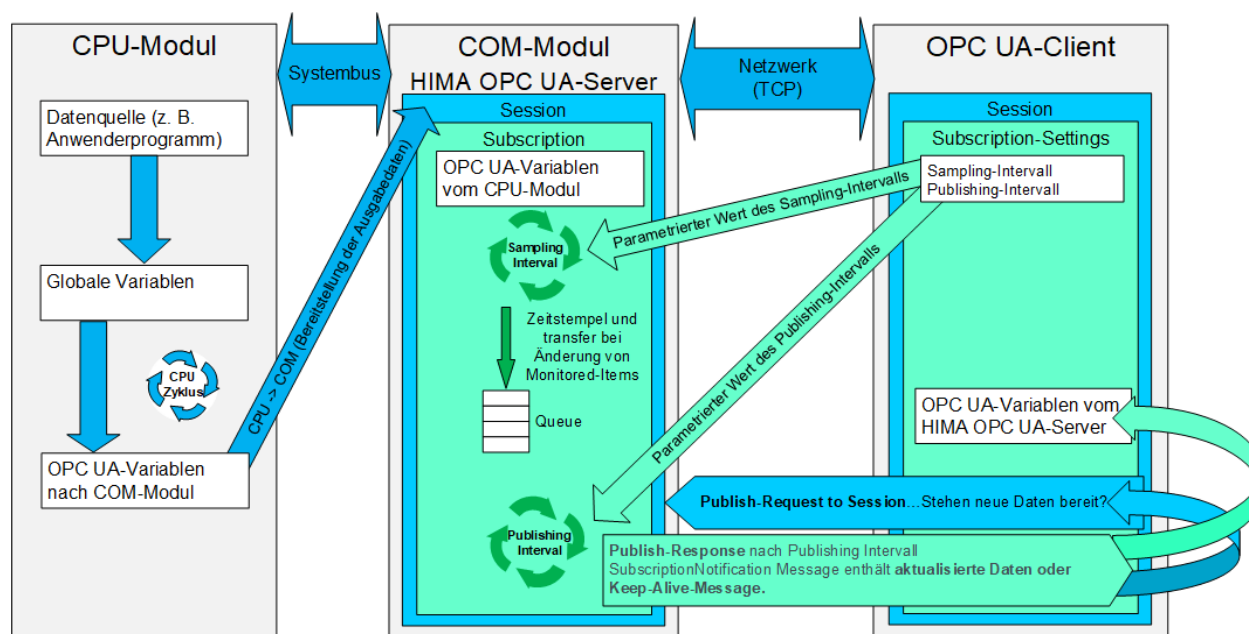
Element	Beschreibung
Schnittstellen	Ethernet-Schnittstellen des Kommunikationsmoduls.
Anzahl Namespaces	1 pro OPC UA-Server
Anzahl Nodes	<ul style="list-style-type: none">• Variablen: Maximal 10000• Objekte: Maximal 4000• Typen:Maximal 1000
Anzahl References	45 000
Reload-Änderungen	Maximal 1000 neue Nodes und maximal 1000 Attribut-Änderungen an vorhandenen Nodes in der Reload-Konfiguration.
Ereignis-Konfiguration	Ereignis-Konfiguration kann bei maximal 4 OPC UA- und X-OPC-Servern aktiviert werden.
TCP-Port	4840

4 Konzept des HIMA OPC UA-Servers



Der HIMA OPC UA-Server läuft direkt auf der Steuerung in einem Kommunikationsmodul. Ein OPC UA-Client verbindet sich direkt mit der Kommunikationsmodul.


Ein OPC UA-Client kann Variablen mittels Subscriptions abonnieren und die Variablen in diesen Subscriptions vom OPC UA-Server überwachen lassen. Nur bei einer Änderung dieser Variablen wird der OPC UA-Client benachrichtigt. Diese Methode reduziert die Menge der zu übertragenden Variablen und führt zu einer erheblichen Reduzierung der erforderlichen Bandbreite.

Die folgende Abbildung zeigt den Ablauf von der Datenquelle im CPU-Modul bis zum Empfang der OPC UA-Variablen, die via Subscriptions aboniert wurden.



Element	Beschreibung
CPU-Modul	Prozessormodul, auf dem die Ein-/Ausgangsdaten und das Anwenderprogramm abgearbeitet werden.
COM-Modul	Kommunikationsmodul, auf dem der HIMA OPC UA-Server läuft.
OPC UA-Client	Der OPC UA-Client greift auf die vom OPC UA-Server bereitgestellten Prozessdaten zu und stellt diese z. B. einem Leitsystem zur Verfügung. Der Zugriff auf die Subscriptions wird im OPC UA-Client mit den beiden Parametern Sampling-Intervall und Publishing-Intervall parametrisiert.
Systembus	Der Systembus verbindet das CPU- und COM-Modul untereinander.
Netzwerk (TCP)	Das Ethernet-Netzwerk verbindet das COM-Modul und den OPC UA-Client miteinander.
Daten-Quelle	Prozessdaten der Steuerung (z. B. aus dem Anwenderprogramm oder aus Hardware-Eingängen).
Globale Variable	Globale Variable können auf verschiedenen Ebenen des SILworX Projektbaums erstellt werden und gelten für alle zu diesem Geltungsbereich gehörenden untergeordneten Ebenen. Die globalen Variablen sind das Verbindungsmedium zwischen dem OPC UA-Server und

Element	Beschreibung
	den Datenquellen der Steuerung.
OPC UA-Variable	Eine OPC UA-Variable hat einen beliebigen von SILworX unterstützten IEC-Datentyp. Zudem können für jede OPC UA-Variable die Zugriffsrechte (lesen, schreiben) parametrisiert werden.
Sessions	Die Session ist die Verbindung zwischen OPC UA-Client und dem OPC UA-Server. Maximal 4 OPC UA-Client-Sessions pro OPC UA-Server.
Subscription	<p>Ein UA-Client kann eine Auswahl von Monitored-Items in sogenannten Subscriptions abonnieren. Bei Änderung der Monitored-Items benachrichtigt der Server den Client.</p> <div>  Mehrere Subscription ermöglichen unterschiedliche Einstellungen und eine optimierte Lastenbalance. </div> <p>Maximal 20 Subscriptions pro OPC UA-Server. Maximal 10 Subscriptions pro Session</p>
Monitored Items	Ein Monitored-Item ist eine Variable, die in einer Subscription beobachtet wird.
Queue	<p>Nur geänderte Monitored-Items-Werte werden in die Queue übertragen. Für jedes Monitored-Item ist eine Queue vorhanden.</p> <p>Der Anwender kann für ein Monitored-Item die Queue-Size einstellen.</p> <p>Wertebereich: 1 ... 2</p>
CPU-Zyklus	<p>Der Zyklus eines Prozessormoduls (CPU-Zyklus) besteht vereinfacht dargestellt aus folgenden Phasen:</p> <ol style="list-style-type: none"> 1. Verarbeitung der Eingabedaten. 2. Abarbeitung des Anwenderprogramms. 3. Bereitstellung der Ausgabedaten. <p>Für weitere Informationen, siehe Systemhandbuch des jeweiligen Systems.</p>
Sampling-Intervall	<p>Der Anwender kann im OPC UA-Client das Sampling-Intervall für die Monitored-Items einstellen. Alle Monitored-Items einer Subscription haben das gleiche Sampling-Intervall. Existiert bereits ein Monitored-Item innerhalb einer Subscription, so erhält jedes hinzugefügte Monitored-Item das gleiche Sampling-Intervall wie die bereits vorhandenen Monitored-Items der Subscription.</p> <p>Die Verwendung eines Sampling-Intervalls pro Monitored-Item, wie dies in der Spezifikation OPC Unified Architecture Specification Part 4: Services vorgesehen ist, würde die COM bei großen Prozessdatenmengen rechnerzeitmäßig zu stark belasten.</p> <p>Wertebereich: 10.0 ... 10000.0 ms</p> <div>  Das Sampling-Intervall hat erhebliche Auswirkungen auf die COM-Last. Je kleiner das Sampling-Intervall, desto größer ist die COM-Last. </div>

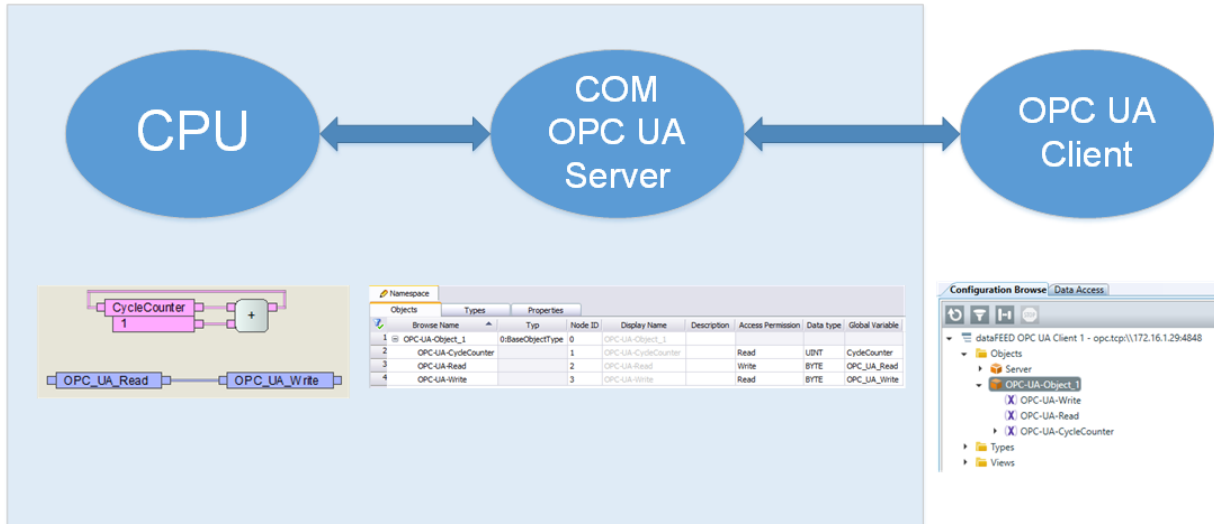
Element	Beschreibung
Publishing-Intervall	<p>Intervall in dem die Notification-Messages vom Server an den Client gesendet werden. Der Anwender kann für eine Subscription das Publishing-Intervall einstellen.</p> <p>Das Publishing-Intervall dient als 'Bremse', damit nicht zu oft Daten an den Client gesendet werden.</p> <p>Wertebereich: 10.0 ... 10000.0 ms</p> <div>  Je kleiner das Publishing-Intervall desto schneller die Client-/Server-Kommunikation und die Last für den Client. </div>
Keep Alive	<p>Der Anwender kann für eine Subscription den <i>Max Keep Alive Count</i> einstellen.</p> <p>Wenn keine neuen Daten in einer Queue vorhanden sind und daher keine Datenübertragung vom Server erforderlich ist, löst der Server nach dem <i>Keep Alive Timeout</i> ein leeres <i>Publish-Response-Telegramm</i> aus.</p>

4.1 Redundanz von OPC UA-Servern

Eine redundante OPC UA-Server-Verbindung wird durch die Konfiguration eines zweiten OPC UA-Servers erreicht. Die Redundanzbildung aus den OPC-UA-Variablen müssen in den redundanten OPC UA-Clients realisiert werden.

5 Konfiguration eines HIMA OPC UA-Servers

In diesem Beispiel wird eine Verbindung zwischen einem HIMA OPC UA-Server und einem OPC UA-Client erstellt. Hierzu die folgende Übersicht:



5.1 Konfiguration in SILworX

Erstellen eines OPC UA-Servers in SILworX.

So erstellen Sie ein **OPC UA-Server-Set** in SILworX:

- Im Strukturbaum die **Ressource** öffnen, in der der OPC UA-Server erstellt werden soll.
- Rechtsklick auf **Protokolle** und **Neu, OPC UA-Server-Set** wählen.
Ein neues **OPC UA-Server-Set** Objekt wird angelegt.

So öffnen Sie den **OPC UA-Server-Set-Editor** in SILworX:

- Rechtsklick auf **OPC UA-Server-Set** und im Kontextmenü **Edit** wählen.
Das "OPC UA-Set" auf Seite 22 enthält die Register **Set-Elemente** und **Eigenschaften**.

So wählen Sie das **COM-Modul** aus:

- Register **Set-Elemente** wählen.
- Rechtsklick auf **COM-Referenz** und **Edit** wählen, um den **COM-Referenz-Editor** zu öffnen.
- Im **COM-Referenz-Editor** das **COM-Modul** auswählen, auf dem der OPC UA-Server laufen soll.



Es kann immer nur ein OPC UA-Server pro COM-Modul angelegt werden.

5.2 Namensraum konfigurieren

Im Kontextmenü von Namensraum **Edit** wählen, um den Namensraum-Editor zu öffnen. Dieser enthält die folgenden Register:

- Objekte
- Typen
- Eigenschaften

So öffnen Sie den **Namensraum-Editor** im OPC UA-Server-Set:

- Rechtsklick auf **Namensraum** und **Edit** wählen, um den Namensraum-Editor zu öffnen.

Namespace								
Objects		Types		Properties				
	Browse Name	Typ	Node ID	Display Name	Description	Access Permission	Data type	Global Variable
1	OPC-UA-Object_1	0:BaseObjectType	0	OPC-UA-Object_1				
2	OPC-UA-CycleCounter		1	OPC-UA-CycleCounter		Read	UINT	CycleCounter
3	OPC-UA-Read		2	OPC-UA-Read		Write	BYTE	OPC-UA_Read
4	OPC-UA-Write		3	OPC-UA-Write		Read	BYTE	OPC-UA_Write

5.2.1 OPC UA-Objekt im Namensraum

Das Element OPC UA-Objekt erfüllt im Adressraum eine strukturierende Funktion.

Für weitere Informationen siehe Kapitel 5.5 der Spezifikation der OPC UA-Foundation *OPC Unified Architecture Specification Part 3: Address Space Model*.

5.2.1.1 Erzeugen von OPC UA-Objekten

So fügen Sie ein **OPC UA-Objekt** auf der obersten Ebene hinzu:

- Im Namensraum-Editor das Register **Objekte** wählen.
- Rechtsklick auf eine freie Stelle im Arbeitsbereich des Namensraum-Editors und im Kontextmenü **Stammelement erzeugen** wählen.
Der Objekt-Auswahldialog öffnet sich.
- Im Objekt-Auswahldialog **OPC UA-Objekt** wählen, um ein OPC UA-Objekt anzulegen.
- In die Tabellenzelle **Typ** den Typ-Namen direkt eingeben oder die Schaltfläche ... anklicken und den gewünschten Typ aus der Liste auswählen.
Wenn gewünscht einen Namen vergeben, und den Dialog mit **OK** bestätigen.

So fügen Sie ein **OPC UA-Objekt** unter einem bestehenden OPC UA-Objekt hinzu:

- Rechtsklick auf ein bestehendes **OPC UA-Objekt** im Arbeitsbereich und im Kontextmenü **Neu** wählen.
Der Objekt-Auswahldialog öffnet sich.
- Im Objekt-Auswahldialog **OPC UA-Objekt** wählen, um eine OPC UA-Variable anzulegen.
Wenn gewünscht einen Namen vergeben, und den Dialog mit **OK** bestätigen.

5.2.1.2 Bearbeiten von OPC UA-Objekten

Sie können OPC UA-Objekte einschließlich aller Unterelemente löschen, kopieren oder ausschneiden und anschließend an möglichen Einfüge-Positionen (siehe oben) einfügen.



Der Browse-Name der OPC UA-Objekte kann durch einen Doppelklick auf das jeweilige Objekt gewählt und geändert werden.

5.2.2 OPC UA-Typen im Namensraum

Das Element **OPC UA-Typ** ermöglicht die Verwendung vorkonfigurierter Strukturen bei der Modellierung von OPC UA-Objekten und OPC UA-Variablen.

Für weitere Informationen siehe Kapitel 4.5 der Spezifikation der OPC UA-Foundation *OPC Unified Architecture Specification Part 3: Address Space Model*

5.2.2.1 Erzeugen von OPC UA-Typen

So fügen Sie im Register *Typen* **OPC UA-Typen** auf der obersten Ebene hinzu:

➤ Rechtsklick auf eine freie Stelle im Arbeitsbereich des Namensraum-Editors und im Kontextmenü **Stammelement erzeugen** wählen.
Der Objekt-Auswahldialog öffnet sich.

➤ Im Objekt-Auswahldialog **OPC UA-Typ** wählen, um ein OPC UA-Typ anzulegen.
Wenn gewünscht einen Namen vergeben, und den Dialog mit **OK** bestätigen.

5.2.2.2 Bearbeiten von OPC UA-Typen

Sie können OPC UA-Typen einschließlich aller eventueller Unterelemente löschen, kopieren oder ausschneiden und danach an möglichen Einfüge-Positionen (siehe oben) wieder einfügen.

5.2.3 OPC UA-Variablen im Namensraum

Das Element OPC UA-Variable repräsentiert im Informationsmodell konkrete Werte: Eigenschaften oder Daten.

Weitergehende Informationen bietet die Spezifikation der OPC UA-Foundation *OPC Unified Architecture Specification Part 3: Address Space Model*, Kap. 5.6.

Zusätzlich zu den Eigenschaften, die für jeden Node verfügbar sind, hat eine Variable noch weitere editierbare Eigenschaften:

5.2.3.1 Erzeugen von OPC UA-Variablen

So fügen Sie im Register *Objekte* eine **OPC UA-Variable** auf der obersten Ebene hinzu:

➤ Im Namensraum-Editor das Register **Objekte** wählen.

➤ Rechtsklick auf eine freie Stelle im Arbeitsbereich des Namensraum-Editors und im Kontextmenü **Stammelement erzeugen** wählen.
Der Objekt-Auswahldialog öffnet sich.

➤ Im Objekt-Auswahldialog **OPC UA-Variable** wählen, um eine OPC UA-Variable anzulegen.

Wenn gewünscht einen Namen vergeben, und den Dialog mit **OK** bestätigen.

So fügen Sie eine **OPC UA-Variable** unter einem bestehenden OPC UA-Objekt hinzu:

- Rechtsklick auf ein bestehendes **OPC UA-Objekt** im Arbeitsbereich und im Kontextmenü **Neu** wählen.
Der Objekt-Auswahldialog öffnet sich.
- Im Objekt-Auswahldialog **OPC UA-Variable** wählen, um eine OPC UA-Variable anzulegen.

Wenn gewünscht einen Namen vergeben, und den Dialog mit **OK** bestätigen.

5.2.3.2 Erzeugen von OPC UA-Variablen per Drag&Drop

So fügen Sie eine **OPC UA-Variable** unter einem bestehenden OPC UA-Objekt oder OPC UA-Typ per Drag&Drop hinzu:

- In der Objektauswahl eine oder mehrere gewünschte **globale Variablen** wählen und per Drag&Drop auf ein vorhandenes **OPC UA-Objekt** oder einen **OPC UA-Typ** ziehen.
Pro globaler Variable wird eine **OPC UA-Variable** erzeugt. Diese erhält den Namen, die Beschreibung und den Datentyp der globalen Variablen. Im Register **Objekte** wird eine Referenz der **OPC UA-Variable** auf die globale Variable erzeugt.

5.2.3.3 Datentyp der OPC UA-Variablen auswählen

So wählen Sie den Datentyp der **OPC UA-Variable** aus:

- Doppelklicken auf das jeweilige Feld **Datentyp** und den erforderlichen Datentyp aus der Dropdown-Liste auswählen.

So verbinden Sie die **OPC UA-Variablen** mit globalen Variablen:

- In der Objektauswahl eine **Globale Variable** wählen und per Drag&Drop in die Spalte Globale Variable ziehen, um diese mit der gewünschten OPC UA-Variable zu verbinden.

So wählen Sie die Zugriffsrechte der **OPC UA-Variable** aus:

- Doppelklicken auf das jeweilige Feld **Zugriffsrecht** und das erforderliche Zugriffsrecht aus der Dropdown-Liste auswählen.

Siehe auch [Zertifikat](#)

6 OPC UA-Set

Der Anwender kann beliebig viele OPC UA-Server-Sets im Ordner *Protokolle* anlegen. Die Anzahl der tatsächlich verwendbaren OPC UA-Server-Sets wird durch die Anzahl der COM-Module der Steuerung begrenzt.

6.1 OPC UA-Set-Editor

Im Kontextmenü von OPC UA-Server-Set **Edit** wählen, um den Editor für das OPC UA-Server-Set zu öffnen. Dieser enthält die folgenden Register:

6.1.1 Register Set-Elemente

Das Register *Set-Elemente* eines OPC UA-Server-Sets listet die Elemente auf, die im Projektbaum unterhalb des Sets angezeigt werden. Der Anwender kann hier Elemente erzeugen, löschen oder deren Namen editieren.

[COM-Referenz](#)

[Namensraum](#)

Das Element "Zertifikate" auf Seite 30 kann zusätzlich vom Anwender angelegt werden.

6.1.2 Register Eigenschaften

Das Register *Eigenschaften* enthält die folgenden Register:

Element	Beschreibung
Typ	OPC UA-Server-Set.
Name	Beliebiger, eindeutiger Name für ein OPC UA-Server-Set. Typ: String Länge: 1 ... 120 ASCII-Zeichen
Max. µP-Budget in [%]	Maximales µP-Budget des COM-Moduls, das bei der Abarbeitung des Protokolls produziert werden darf. Wertebereich: 0 ... 100% Standardwert: 50%
Warnung bei µP-Budget-Überschreitung in [%]	Warnschwelle für das µP-Budget des COM-Moduls, bei deren Überschreiten der OPC UA-Server eine Kommunikations-Warnung melden und im Online-Dienst anzeigen muss. Der Wert muss niedriger als der Wert des Maximalen µP-Budget sein. Wertebereich: 1 ... 100% Standardwert: Deaktiviert

Element	Beschreibung
Events aktivieren	Alarm&Events einer Ressource für dieses OPC UA-Set aktivieren.
	Aktiviert: Alarm&Events aktiviert.
	Deaktiviert: Alarm&Events deaktiviert.
	Standardwert: Aktiviert
Nur verschlüsselte Verbindungen zulassen	Der OPC UA-Server kann für Verbindungen verschiedene Endpunkte anbieten, siehe Automation-Security . Mit diesem Schalter wird die Menge der angebotenen Endpunkte konfiguriert.
	Aktiviert: Der OPC UA-Server schränkt die Auswahl der angebotenen Endpunkte auf Sign & Encrypt ein. Sind bei aktiviertem Schalter keine Zertifikate vorhanden, meldet SILworX einen Fehler.
	Deaktiviert: Der OPC UA-Server schränkt die Auswahl der angebotenen Endpunkte nicht ein. Sind bei deaktiviertem Schalter keine Zertifikate vorhanden, meldet SILworX eine Warnung und weist auf die Möglichkeit hin, die Verbindungen auf <i>nur verschlüsselt</i> zu begrenzen. Sind keine Zertifikate vorhanden, meldet SILworX eine Information, dass unverschlüsselte Verbindungen zugelassen werden.
	Standardwert: Aktiviert

6.2 COM-Referenz

Das OPC UA-Server-Set enthält standardmäßig das Element *COM-Referenz* zur Konfiguration des COM-Moduls, auf dem der OPC UA-Server laufen soll. Für eine erfolgreiche Codegenerierung muss ein COM-Modul ausgewählt sein.

6.2.1 Register Eigenschaften

Im Kontextmenü von COM-Referenz **Edit** wählen, um den COM-Referenz-Editor zu öffnen. Dieser enthält das Register *Eigenschaften* mit den folgenden Parametern.

Element	Beschreibung
Typ	COM-Referenz
Name	Beliebiger, eindeutiger Name für eine COM-Referenz Typ: String Länge: 1 ... 120 ASCII-Zeichen
Modul	Das COM-Modul, auf dem der OPC UA-Server läuft. In der Dropdown-Liste der verfügbaren COM-Module kann der Anwender das passende Modul auswählen.

6.3 Namensraum-Editor

Der Anwender hat im Namensraum die Möglichkeit, die Daten, die per OPC UA übertragen werden sollen, entsprechend seinen Anforderungen zu strukturieren. Die Gesamtheit der vom Anwender modellierten Daten bildet den OPC UA-Adressraum.

Der Namesraum 0 ist durch die OPC Foundation vorgegeben und enthält die Basisdefinitionen für OPC UA-Modelle. Die Anzahl der Elemente, die direkt unterhalb dieses Namesraums 0 angelegt werden können, ist auf 80 beschränkt. Das beinhaltet:

- Alle vom Anwender erzeugten OPC UA-Typen, die von Namesraum-0-Typen abgeleitet sind.
- Alle vom Anwender auf der Stamm-Ebene erzeugten OPC UA-Objekte und -Variablen, da diese auf Namesraums-0-Objects verweisen.

Weitergehende Informationen bietet die Spezifikation der OPC UA-Foundation *OPC Unified Architecture Specification Part 3: Address Space Model*.

Der Namensraum muss mindestens ein Element enthalten und darf maximal 10000 OPC UA-Variablen, 4000 OPC UA-Objekte und 1000 OPC UA-Typen enthalten. Eine Unter- oder Überschreitung der unterstützten Grenzwerte wird im Logbuch als Fehler gemeldet.

Die Nodes des Namensraums sind:

- OPC UA-Objekte
- OPC UA-Typen
- OPC UA-Variablen

Im Register Typen kann der Anwender Typen erzeugen und editieren, die anschließend für die Modellierung der Objekte verwendet werden können.

6.3.1 Register Objekte und Typen

Die Nodes in den Registern **Objekte** und **Typen** verfügen über die folgenden gemeinsamen Parameter:

Element	Beschreibung
Browse-Name	<p>Maximale Länge: 511 Bytes UTF-8 Zeichen.</p> <p>Einschränkungen:</p> <ul style="list-style-type: none"> • Der Browse-Namen darf keine Punkte (.) enthalten, damit ein Tabellen-Import oder -Export möglich ist. • Der Browse-Name darf nicht mit einem Sternchen (*) beginnen. • Der Browse-Name darf nicht mit einer Zahl, gefolgt von einem Doppelpunkt (:) beginnen. <p>Browse-Name-Qualifizierung:</p> <p>Der Browse-Name eines OPC UA-Nodes wird durch einen Namensraum-Index einem Namensraum zugeordnet ("qualifiziert") z. B. '2:MeineVariable'.</p> <p>Ein fremder Namensraum-Index wird gespeichert und angezeigt wie eingegeben. Im Logbuch wird dies als Fehler gemeldet.</p>
Typ	<p>Der in der Spalte Typ angezeigte Referenztext entspricht dem Browse-Namen des referenzierten Typ. Er kann daher auch durch einen Namensraum qualifiziert werden (siehe Browse-Namen-Qualifizierung). Aktuell wird nur der Index des Namensraums akzeptiert, in dem das Element sich befindet, andere Namensräume werden durch die Validierung als Fehler gemeldet. Eine Ausnahme bildet der Typ "0:BaseObjectType", dieser Basis-Typ im</p>

Element	Beschreibung						
	<p>Namensraum 0 ist immer verfügbar.</p> <ul style="list-style-type: none"> Referenz auf einen OPC UA-Typ, wobei der Referenztext dem Browse-Namen des referenzierten Typ-Nodes entspricht. OPC UA-Objekte werden hiermit zu Instanzen des Typs. OPC UA-Typen werden hiermit zu Ableitungen dieses Typs. 						
Obligatorisch	<p>Die Eigenschaft <i>Obligatorisch</i> entspricht der ModellingRule Mandatory im OPC UA-Adressraum und ist nur im Register Typen für die Elemente (OPC UA-Objekte und -Variablen) eines Typs verfügbar.</p> <table border="1"> <tr> <td>Aktiviert</td><td>Dieses Unterelement muss in einem OPC UA-Objekt dieses Typs vorhanden sein.</td></tr> <tr> <td>Deaktiviert</td><td>Dieses Unterelement ist nicht zwingend erforderlich.</td></tr> </table> <p>Standardeinstellung: Aktiviert</p>	Aktiviert	Dieses Unterelement muss in einem OPC UA-Objekt dieses Typs vorhanden sein.	Deaktiviert	Dieses Unterelement ist nicht zwingend erforderlich.		
Aktiviert	Dieses Unterelement muss in einem OPC UA-Objekt dieses Typs vorhanden sein.						
Deaktiviert	Dieses Unterelement ist nicht zwingend erforderlich.						
Node-ID	<p>Die Node-ID ist der eindeutige numerische Identifizierer eines Elements im Adressraum. Die Node-ID wird automatisch vergeben und kann nachträglich manuell geändert werden.</p> <p>Wertebereich: 0 ... 1073741823</p>						
Display-Name	<p>Der Display-Name ist der Name, mit der das Element im OPC UA-Client angezeigt wird. SILworX übernimmt automatisch den Browse-Namen für den Display-Namen. Dieser kann nachträglich manuell geändert werden.</p> <p>Wenn der Anwender den vorgegebenen Display-Name vergibt, dann entspricht der Display-Name dem Browse-Name. Das hat den Vorteil, dass in der Konfigurationsdatei kein zusätzlicher Speicherplatz für den Display-Namen benötigt wird.</p> <p>Im Editor wird der vorgegebenen Display-Name in grauer Schrift angezeigt.</p> <p>Maximale Länge: 511 Bytes UTF-8 Zeichen.</p>						
Beschreibung	<p>Allgemeine Beschreibung oder Kommentar des Knotens.</p> <p>Maximale Länge: 4095 Bytes UTF-8 Zeichen.</p>						
Zugriffsrecht	<p>Mit dem Zugriffsrecht wird festgelegt, ob der aktuelle Wert einer OPC UA-Variable lesbar-/beschreibbar ist.</p> <p>Doppelklick auf das Feld Zugriffsrecht der OPC UA-Variable und das gewünschten Zugriffsrecht aus der Dropdown-Liste wählen.</p> <table border="1"> <tr> <td>Lesen</td><td>Ein OPC UA-Client kann den Wert der Variable lesen.</td></tr> <tr> <td>Schreiben</td><td>Ein OPC UA-Client kann den Wert der Variable verändern.</td></tr> <tr> <td>Lesen + Schreiben</td><td>Ein OPC UA-Client kann den Wert der Variable lesen und verändern. Wenn ein OPC UA-Client einen Wert schreibt, so wird dieser zunächst auf das CPU-Modul übertragen. Der Wert zum Lesen ist immer der resultierende Wert aus dem CPU-Modul.</td></tr> </table> <p>Standardeinstellung: Lesen</p>	Lesen	Ein OPC UA-Client kann den Wert der Variable lesen.	Schreiben	Ein OPC UA-Client kann den Wert der Variable verändern.	Lesen + Schreiben	Ein OPC UA-Client kann den Wert der Variable lesen und verändern. Wenn ein OPC UA-Client einen Wert schreibt, so wird dieser zunächst auf das CPU-Modul übertragen. Der Wert zum Lesen ist immer der resultierende Wert aus dem CPU-Modul.
Lesen	Ein OPC UA-Client kann den Wert der Variable lesen.						
Schreiben	Ein OPC UA-Client kann den Wert der Variable verändern.						
Lesen + Schreiben	Ein OPC UA-Client kann den Wert der Variable lesen und verändern. Wenn ein OPC UA-Client einen Wert schreibt, so wird dieser zunächst auf das CPU-Modul übertragen. Der Wert zum Lesen ist immer der resultierende Wert aus dem CPU-Modul.						

Element	Beschreibung
Datentyp	<p>Festlegung des beliebigen, elementaren Datentyps.</p> <p>Eine neue OPC UA-Variable wird immer mit Datentyp BOOL erstellt. Doppelklick auf das Feld Datentyp der OPC UA-Variable und den gewünschten Datentyp aus der Dropdown-Liste wählen.</p> <p>Standardeinstellung: BOOL</p>
Globale Variable:	<p>Referenz auf eine globale Variable, deren Datentyp zum Datentyp der OPC UA-Variable passen muss. Der Referenztext entspricht dem Namen der globalen Variablen. Für eine OPC UA-Variable mit Referenz auf eine konstante globale Variable ist das Zugriffsrecht <i>Lesen</i> möglich.</p> <p>Nur für Register Objekte</p>

Tabelleninhalt des Namensraum-Editors mit den Kontextmenüfunktionen **Tabelleninhalt als CSV sichern** oder **Tabelleninhalt aus CSV importieren**, siehe OLH CSV-Import und Export.

6.3.2 Register Eigenschaften

Das Register **Eigenschaften** eines Namensraums enthält die folgenden Parameter:

Element	Beschreibung
Typ	Namensraum, nicht änderbar.
Name	<p>Beliebiger, eindeutiger Name für einen Namensraum Typ: String. Dieser Name dient nur der Identifizierung des Namensraum innerhalb von SILworX und wird im OPC UA-Adressraum nicht verwendet.</p> <p>Typ: ASCII-String</p> <p>Länge: 1 ... 120 Zeichen</p>
Namensraum-URI	<p>Beliebiger, eindeutiger und nicht leerer Name für ein Namensraum-URI (Uniform Resource Identifier).</p> <p>Die Namensraum-URI dient im OPC UA-Adressraum zur Identifizierung eines Namensraums.</p> <p>Typ: ASCII-String</p> <p>Die Namensraum-Länge: 1 ... 255 Zeichen</p>
Namensraum-Index	<p>Eindeutiger, numerischer Wert zur Identifizierung des Namensraums.</p> <p>Wertebereich: 2 ... 9</p> <p>Standardwert: 2</p>

6.4 Typ-Referenz (im Namensraum)

Mit dem Referenzieren eines OPC UA-Typs erzeugt SILworX automatisch alle Unterelemente des Typ im referenzierenden Element. Eventuell schon vorhanden Unterelemente bleiben unverändert.

Sollte das Element bereits Unterelemente besitzen, deren Browse-Namen identisch zu Unterelementen des Typs sind, so wird der Typ nicht zugewiesen. Eine Fehlermeldung im Logbuch listet die nicht eindeutigen Elemente auf.

Während der Verifikation wird für OPC UA-Objekte und -Typen geprüft, dass alle obligatorischen Unterelemente des zugewiesenen Typs vorhanden sind. Weiterhin müssen die Werte von *Display-Name*, *Typ*, *Datentyp* und *Zugriffsrecht* aller Unter-Elemente, d.h. auch der optionalen Elemente, mit den Werten der korrespondierenden Elemente im Typ übereinstimmen.



Der Typ-Auswahldialog bietet im Kontextmenü den Export der Tabelle in eine csv-Datei an.

6.4.1 Editieren der Typ-Referenz

Editieren der **Typ-Referenz** im Namensraum Editor für **OPC UA-Objekte** und **OPC UA-Typen**.

So editieren Sie die **Typ-Referenz** eines Elements zu einem OPC UA-Typ:

- Rechtsklick auf **Namensraum** und **Edit** wählen, um den Namensraum-Editor zu öffnen.
- Im Namensraum-Editor das Register **Objekte** oder **Typen** wählen.
- In die Tabellenzelle **Typ** den Typ-Namen direkt eingeben oder die Schaltfläche ... anklicken und den gewünschten Typ aus der Liste auswählen.

6.4.2 Bearbeiten von OPC UA-Variablen

Sie können OPC UA-Variablen löschen, kopieren oder ausschneiden und danach an möglichen Einfüge-Positionen (siehe oben) wieder einfügen.

6.4.3 Editieren der Globale-Variable-Referenz

So können Sie die **Referenz** einer OPC UA-Variablen zu einer globalen Variable editieren:

- Im Namensraum-Editor das Register **Objekte** wählen.
- In der Objektauswahl eine oder mehrere gewünschte **globale Variablen** wählen und per Drag&Drop in die Spalte **Globale Variable** der OPC UA-Variable ziehen..

SILworX führt eine automatische Zuordnung von IEC-Datentypen auf OPC UA-Datentypen entsprechend folgender Tabelle durch:

IEC-Datentypen	OPC UA-Datentypen	Wertebereich
BOOL	Boolean	TRUE oder FALSE
BYTE	Byte	0 ... 255
WORD	UInt16	0 ... 65535

IEC-Datentypen	OPC UA-Datentypen	Wertebereich
DWORD	UInt32	0 ... 4294967295
LWORD	UInt64	0 ... 18446744073709551615
SINT	SByte	-128 ... 127
INT	Int16	-32768 ... 32767
DINT	Int32	-2147483648 ... 2147483647
LINT	Int64	-9223372036854775808 ... 9223372036854775807
USINT	Byte	0 ... 255
UINT	UInt16	0 ... 65535
UDINT	UInt32	0 ... 4294967295
ULINT	UInt64	0 ... 18446744073709551615
REAL	Float	Gleitkommazahl (IEEE 754-1985 single precision)
LREAL	Double	Gleitkommazahl (IEEE 754-1985 double precision)
TIME	UInt64	Ganzzahl von 0 ... 18446744073709551615

Für weitere Informationen siehe Kapitel 8 der Spezifikation der OPC UA-Foundation *OPC Unified Architecture Specification Part 3: Address Space Model*.

6.4.4 Fehlerhafte Typ-Referenzen

6.4.4.1 Ungültige Typ-Referenzen

Ungültige OPC UA-Typ-Referenzen können durch folgende Aktionen entstehen und werden von der Verifikation als Fehler gemeldet:

- Löschen oder Editieren bereits verwendeter Typen.
- CSV-Import von unbekannten Typ-Namen.

6.4.4.2 Zirkelschlüsse

Gegenseitige oder Selbst-Referenzierungen sind in einem hierarchischen System nicht möglich, da diese zu einer endlosen Kette gegenseitiger Abhängigkeiten führen würden. SILworX kann in diesem Fall keine korrekte Modellierung der Objekt-Elemente durchführen.

Im Register Typen werden daher im Typ-Auswahldialog Typen, die zu Selbst- oder gegenseitiger Referenzierung führen würden, nicht angeboten.

Sind solche Typen dennoch unbeabsichtigt entstanden und verwendet worden, z.B. durch Tabellen-Import, Kopieren oder Verschieben von Elementen, so müssen Anwender zunächst die fehlerhaften Zirkelschlüsse in den Typen entfernen, bevor fehlerhafte Typ-Zuweisungen in den Verwendungen behoben werden können.

6.4.4.3 Zirkelschluss in der Typ-Verwendung

Wenn im Register **Typen** OPC UA-Objekte vom Anwender erstellte Typen referenzieren, sind folgende Zustände nicht erlaubt und werden von der Verifikation als Fehler gemeldet:

- Selbst-Referenzierung eines Typs. Ein Typ darf nirgendwo in seinen Elementen ein Objekt enthalten, das auf eben diesen Typ verweist.
Beispiel: T1 und T2 sind für T1.O1 nicht gültig.

Objekte	Typen	Eigenschaften
Browse-Name	Typ	
1 T1	0:BaseObjectType	
2 O1	0:BaseObjectType	
3 T2	T1	
4 O1	0:BaseObjectType	

- Gegenseitige Referenzierung zweier Typen. Ein Typ darf nirgendwo in seinen Elementen ein Objekt enthalten, das auf einen Typ verweist, der wiederum Elemente des ersten Typs enthält.
Beispiel: T1, T2 und T3 sind für T1.O1 nicht gültig.

Objekte	Typen	Eigenschaften
Browse-Name	Typ	
1 T1	0:BaseObjectType	
2 O1	0:BaseObjectType	
3 T2	0:BaseObjectType	
4 O1	T1	
5 O1	0:BaseObjectType	
6 T3	T2	
7 O1	T1	
8 O1	0:BaseObjectType	

6.4.4.4 Zirkelschluss in der Typ-Ableitung von Typen

Wenn im Register **Typen** OPC UA-Typen von anderen OPC UA-Typen abgeleitet werden, sind folgende Zustände nicht erlaubt und werden von der Verifikation als Fehler gemeldet:

- Selbst-Ableitung eines Typs.
Ein Typ darf nicht von sich selbst abgeleitet werden.
Beispiel: T1, T2 und T3 sind für T1 nicht gültig.

Objekte	Typen	Eigenschaften
Browse-Name	Typ	
1 T1	0:BaseObjectType	
2 T2	T1	
3 T3	T2	

- Gegenseitige Ableitung zweier Typen.
Ein Typ darf nicht von einem Typ oder dessen Ableitungen abgeleitet werden, für den er selbst bereits als Basis-Typ dient.

6.5 Zertifikate

Der HIMA OPC UA-Server authentifiziert sich selbst mit dem Server-Zertifikat gegenüber Clients, die sich mit ihm verbinden wollen. Die Clients wiederum müssen durch den Server verifiziert werden und dafür über ein Client-Zertifikat verfügen, das in der Zertifikat-Verwaltung des Servers geladen ist. Nur unter diesen Voraussetzungen können verschlüsselte Verbindungen hergestellt werden.

Der HIMA OPC UA-Server unterstützt folgende Security Profile:

- SecurityPolicy - None.
- SecurityPolicy [B] - Basic256Sha256.

Wenn das Security-Profil SecurityPolicy [B] - Basic256Sha256 verwendet werden soll, wird ein konfiguriertes Server-Zertifikat und ein konfiguriertes Client-Zertifikat benötigt.

Das OPC UA-Server-Set ermöglicht die Verwaltung von Zertifikaten, die für verschlüsselten und signierten Transport von OPC UA-Daten nötig sind. Dafür wird immer ein Server-Zertifikat und mindestens ein Client-Zertifikat benötigt. Sind Client-Zertifikate geladen, aber kein Server-Zertifikat, so meldet SILworX einen Fehler.



Die benötigte Server-Zertifikat-Schlüssel-Datei kann von dem zuständigen Systemadministrator generiert werden.

Die Zertifikat-Dateien müssen dem ITU-T-Standard X.509v3 entsprechen. Die Codierung der Zertifikatsdateien muss unter Anwendung der durch die ITU-T-Empfehlung X.690 definierten Distinguished Encoding Rules (DER) erfolgen.

Server-Zertifikat

Ein Server-Zertifikat wird durch einen Namen gekennzeichnet und besteht aus einer Zertifikat-Datei und einer Schlüssel-Datei. Beide Dateien müssen vom Anwender zur Verfügung gestellt und in das Projekt geladen werden. Pro OPC UA-Server kann nur jeweils ein Server-Zertifikat vorhanden sein.

Um ein Server-Zertifikat in das SILworX Projekt zu laden, wählen Sie im Kontextmenü des Server-Zertifikate-Registers den Menüpunkt **Server-Zertifikat laden** aus. Im Dialog **Server-Zertifikat** wählen, wenn gewünscht einen Namen vergeben, und den Dialog bestätigen. Im anschließenden Dialog **Server-Zertifikat laden** können Sie die Zertifikat-Datei (mit der Datei-Erweiterung **.der**) und die Schlüssel-Datei (mit der Datei-Erweiterung **.pem**) auswählen.

Änderungen an der Zertifikat- oder der Schlüssel-Datei können im SILworX nicht ausgeführt werden. Bei Bedarf löschen Sie bitte das Zertifikat-Element aus dem Editor und laden sie das Zertifikat wie oben beschrieben erneut ins Projekt.

Die Schlüsseldatei für das Server-Zertifikat enthält den privaten Schlüssel, das Gegenstück zum öffentlichen Schlüssel aus der Server-Zertifikat-Datei.

Die Codierung der Schlüssel-Datei muss dem PEM-Format entsprechen.

Hierbei handelt es sich um eine Base64-codierte Variante des Schlüssels, umschlossen von einer Kopfzeile (header) und Fußzeile (footer), welche das Schlüsselformat beschreibt.

Server-Zertifikat-Datei

Die Server-Zertifikat-Datei muss nachfolgende Felder enthalten:

Element	Beschreibung
Version	Die Version muss V3 sein.
Seriennummer	Durch den Aussteller festgelegte eindeutige Seriennummer des Zertifikats.
Signaturalgorithmus	Der verwendete Signaturalgorithmus für den Fingerabdruck.
Fingerabdruck	Signatur des Zertifikats.
Aussteller	Name des Ausstellers (issuer) des Zertifikates, wie unter RFC 3280 beschrieben.
Gültigkeitsdauer	Festlegung der Gültigkeitsdauer, wann das Zertifikat gültig wird und wann es abläuft.
Antragsteller	Name des Antragstellers (subject) des Zertifikates, wie unter RFC 3280 beschrieben.
Alternativer Antragstellername	Der alternativer Antragstellername (subject alternative name) sollte zwei Einträge enthalten: <ol style="list-style-type: none"> 1. Ein einheitlicher Bezeichner, ein Uniform Resource Identifier (URI), was der URN des OPC UA-Server entsprechen sollte. 2. Die IP-Adresse des COM-Moduls, auf dem der OPC UA-Server konfiguriert wird.
Öffentlicher Schlüssel	Enthält den öffentliche Schlüssel (public key) sowie den verwendeten Verschlüsselungsalgorithmus.
Schlüsselverwendung	Die Schlüsselverwendung (key usage) des Zertifikates sollte folgendes beinhalten: <ul style="list-style-type: none"> • Digitale Signatur (digital signature) • Nichtabstreitbarkeit (non repudiation) • Schlüsselverschlüsselung (key encipherment) • Datenverschlüsselung (data encipherment)
Erweiterte Schlüsselverwendung	Die erweiterte Schlüsselverwendung (extended key usage) sollte Serverauthentifizierung (1.3.6.1.5.5.7.3.1) enthalten.
Stellenschlüsselkennung	Die Stellenschlüsselkennung (authority key identifier) gibt eine Informationen über den Schlüssel, der für das Signieren des Zertifikats verwendet wurde. Dies ist notwendig für Zertifikate, die über eine Zertifizierungsstelle (certificate authority) erstellt werden und sollte auch für selbst signierte Zertifikate angegeben werden.

Client-Zertifikate

Ein Client-Zertifikat wird durch einen Namen gekennzeichnet und besteht aus einer Zertifikat-Datei. Die Datei muss vom Anwender zur Verfügung gestellt und ins Projekt geladen werden. Pro OPC UA-Server können jeweils bis zu 8 Client-Zertifikate verwaltet werden.

Um ein Client-Zertifikat in das SILworX-Projekt zu laden, wählen Sie im Kontextmenü des Client-Zertifikate-Registers den Menüpunkt **Client-Zertifikat laden** aus. Im Dialog **Client-Zertifikat** wählen, wenn gewünscht einen Namen vergeben, und den Dialog bestätigen. Im anschließenden Dialog **Client-Zertifikat laden** können Sie die Zertifikat-Datei (mit der Datei-Erweiterung **.der**) auswählen.

Änderungen an der Zertifikat- oder der Schlüssel-Datei können im SILworX nicht ausgeführt werden. Bei Bedarf löschen Sie das Zertifikat-Element aus dem Editor und laden Sie das Zertifikat wie oben beschrieben erneut ins Projekt.

6.5.1 Zertifikate-Editor einer Ressource anlegen

So erstellen Sie den Zertifikate-Editor in einer Ressource:

- Im Strukturbaum **Ressource, Protokolle, OPC UA-Server Set** wählen.
- Rechtsklick auf **OPC UA-Server-Set** und im Kontextmenü **Neu** wählen.
Der Objekt-Auswahldialog öffnet sich.
- Im Objekt-Auswahldialog **Zertifikate** wählen, um ein Zertifikate-Objekt anzulegen.
Der Objekt-Auswahldialog *Zertifikate* wird neu hinzugefügt. Dieser beinhaltet die Register *Server-Zertifikate* und *Client-Zertifikate*.

So öffnen Sie den Zertifikate-Editor einer Ressource:

- Rechtsklick auf **Zertifikate** und **Edit** wählen, um den Zertifikate-Editor zu öffnen.

Während der Datei-Auswahl prüft SILworX, ob:

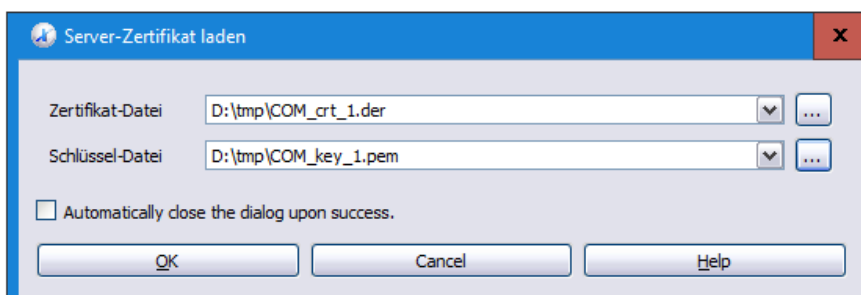
- Die Dateien vorhanden sind.
- Die minimale Dateigröße von 1 Byte nicht unterschritten ist.
- Die maximale Dateigröße von 10 KiB für eine Zertifikat-Datei bzw. 4 KiB für eine Schlüsseldatei nicht überschritten sind.

Die Datei-Auswahl kann erst abgeschlossen werden, wenn alle Bedingungen erfüllt wurden.

6.5.1.1 Server-Zertifikat laden

So laden Sie ein Server-Zertifikat in den Server-Zertifikate-Editor:

- Register **Server-Zertifikat** wählen.
- Rechtsklick auf eine freie Stelle im Arbeitsbereich des Server-Zertifikat-Editors und im Kontextmenü **Server-Zertifikat laden** wählen.
Der Objekt-Auswahldialog öffnet sich.
- Im Objekt-Auswahldialog **Server-Zertifikat** wählen, um den Dialog **Server-Zertifikat laden** zu öffnen.

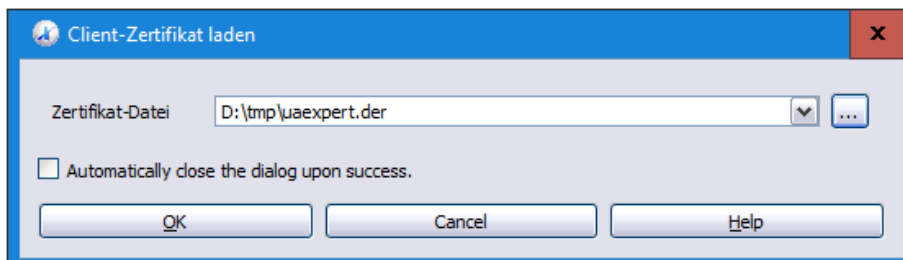


- **Zertifikat-Datei** und **Schlüssel-Datei** des HIMA OPC UA-Servers aus dem Speicherort wählen und mit **OK** bestätigen.

6.5.1.2 Client-Zertifikat laden

So laden Sie ein Client-Zertifikat in den Server-Zertifikate-Editor:

- Register **Client-Zertifikat** wählen.
- Rechtsklick auf eine freie Stelle im Arbeitsbereich des Client-Zertifikat-Editors und im Kontextmenü **Client-Zertifikat laden** wählen.
Der Objekt-Auswahldialog öffnet sich.
- Im Objekt-Auswahldialog **Client-Zertifikat** wählen, um den Dialog **Client-Zertifikat laden** zu öffnen.



- **Zertifikat-Datei** des OPC UA-Clients aus dem Speicherort wählen und mit **OK** bestätigen.



Gültigkeitsdauer der OPC UA-Zertifikate beachten!

Eventuell kann ein gültiges Zertifikat nicht ausgeführt werden, wenn auf dem OPC UA-Server oder OPC UA-Client nicht das aktuelle Datum und Uhrzeit eingestellt sind.

7 Alarm&Events

Werden in einem OPC UA-Client Events abonniert, beginnt der OPC UA-Server die Event-Daten aus dem System zu holen und liefert sie an den OPC UA-Client aus.

Das System unterstützt maximal 5000 Events pro Ressource. Bei Überschreitung dieser Menge meldet SILworX einen Fehler und eine Codegenerierung ist nicht möglich.

Events können im "Alarm&Events-Editor" auf Seite 36 konfiguriert werden. Die Message-Texte und Prioritäten der Event-Definitionen werden in die OPC UA-Konfigurationsdatei geschrieben.

7.1 Alarm&Events einer Ressource aktivieren

So aktivieren Sie die Events in einer Ressource:

- Rechtsklick auf **OPC UA-Server-Set** und **Edit** wählen.
- Register **Eigenschaften** wählen und **Events aktivieren** aktivieren.

Events in einer Ressource können für maximal 4 OPC UA-Server aktiviert werden, ohne Nutzung vom X-OPC Server oder safeEDR-Set mit aktivierten *Alarm&Events*.

Die folgenden Systeme erlauben die maximale Anzahl konfigurierter Zugriffe auf die Events:

- HIMax: 4
- HIQuad X: 4
- HIMatrix: 1

Der OPC UA-Server unterstützt die *System-Ereignisse*, die bei der Erfüllung der vom System vordefinierten Bedingungen auftreten.

Die folgenden *System-Ereignisse* können an einen OPC UA-Client ausgeliefert werden:

- System gestartet.
- System gestoppt.
- Reload aktiviert.
- Kommunikationsaufnahme zwischen System-CPU und SOE-E/A-Modul.
- Kommunikationsverlust zwischen System-CPU und SOE-E/A-Modul.
- Neuinitialisierung der Ereignisse.
- Keine freien Speicherplätze für Ereignis-Einträge im Ereignis-Puffer vorhanden.

7.2 Daten einer an einen OPC UA-Client ausgelieferten Event-Notifikation

Für jede Event-Definition erzeugt SILworX in der Konfigurationsdatei einen Eintrag (Sektion EventSource) mit folgenden Attributen:

Element	Beschreibung
Event-ID	Event-ID ist eindeutig und identifiziert eine Event-Notifikation. Länge: 12 Byte Array. Nicht vom Anwender änderbar.
Event-Type	Entspricht dem Typ <i>BaseEventType</i> für ein Ereignis, welches nicht <i>System-Ereignis</i> ist, oder <i>SystemEventType</i> für ein <i>System-Ereignis</i> .
Source-Name	Entspricht dem Namen der Event-Definition im Alarm&Events-Editor.
Source-Node	Entspricht der Node-ID der ersten OPC UA-Variable mit einem Globale-Variable-Verweis auf die Event-Definition im Alarm&Events-Editor. Wenn keine OPC UA-Variable einen Globale-Variable-Verweis auf die Event-Definition hat, entspricht der <i>SourceNode</i> dem <i>Server Node</i> . Findet SILworX während der Codegenerierung weitere OPC UA-Variablen mit dem gleichen Globale-Variable-Verweis, so werden diese ignoriert und verweisen nicht auf die Event-Definition.
Severity	Entspricht der Priorität aus der Event-Definition im Alarm&Events-Editor.
Message	Entspricht dem Text aus der Event-Definition im Alarm&Events-Editor.
Time	Entspricht dem Zeitstempel des Auftretens des Ereignisses.
ReceiveTime	Entspricht dem Zeitstempel des Empfangs des Ereignisses auf der COM.

7.3 Alarm&Events-Editor

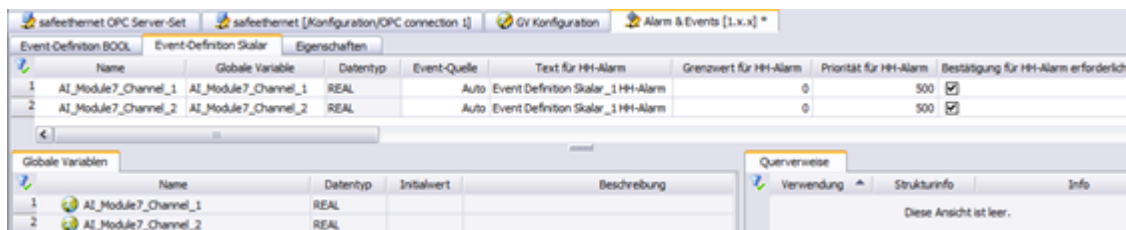
Die Konfiguration der Alarm&Events findet im Alarm&Events-Editor der Ressource statt. Die im Alarm&Events-Editor angelegten Ereignisse werden automatisch über die konfigurierte **safeethernet** Verbindung übertragen.

So erstellen Sie den **Alarm&Events-Editor** in einer Ressource:

- Im Strukturbaum **Konfiguration, Ressource** wählen.
- Rechtsklick auf **Ressource** und im Kontextmenü **Neu, Alarm&Events** wählen.
Der Alarm&Events-Editor wird neu hinzugefügt. Dieser beinhaltet die Event-Definitionen und Eigenschaften.

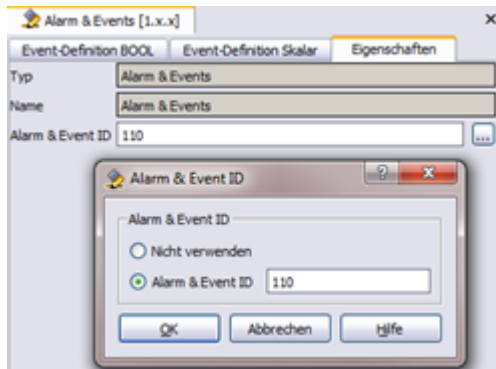
So ziehen Sie **globale Variablen** in den Alarm&Events-Editor:

- Rechtsklick auf **Alarm&Events** und **Edit** wählen.
- Register **Event Definition Bool** für boolsche Ereignisse wählen.
- Register **Event Definition Skalar** für skalare Ereignisse wählen.
- In der Objektauswahl auf die **globale Variable** klicken und per Drag&Drop auf eine freie Stelle im Arbeitsbereich des Alarm&Events Editors ziehen.



So legen Sie die **Alarm&Events ID** für Generierung eindeutiger Cookies der Ressource an:

- Register **Eigenschaften** des Alarm&Events Editors wählen und neben dem Feld **Alarm&Event-ID** die Schaltfläche ... anklicken.
Das Dialogfenster **Alarm&Event-ID** öffnet sich.
- **Alarm&Event-ID** wählen und eine eindeutige **Alarm&Event-ID** eintragen.



7.3.1 Anzeige im OPC Client

Der im OPC Client angezeigte Name des OPC UA-Servers setzt sich zusammen aus:

HIMA (Hersteller).Dienstname-DA (Data Access).

Die Verbindung zum OPC Server herstellen. Die konfigurierten Data Access Daten sollten jetzt zum OPC Client übertragen werden.

Verbindung zum X-OPC Server herstellen. Die konfigurierten Alarmer und Ereignisse sollten jetzt zum OPC Client übertragen werden.

8 Control-Panel (Online)

Im Control-Panel kann der Anwender die Einstellungen des OPC UA überprüfen und steuern. Zudem werden aktuelle Statusinformationen des OPC-UA-Servers angezeigt.

So öffnen Sie das Control-Panel zur Überwachung des OPC-UA-Servers:

- Im Strukturbaum Rechtsklick auf **Hardware** und im Kontextmenü **Online** wählen.
- Im **System-Login** Zugangsdaten eingeben, um die Online-Ansicht der Hardware zu öffnen.

So öffnen Sie die Diagnose des COM-Moduls zur Diagnose des OPC UA-Servers:

- Rechtsklick auf **COM-Modul** und im Kontextmenü **Diagnose** wählen, um die Diagnoseansicht des COM-Moduls zu öffnen.
- In der Diagnoseansicht **Alle Einträge** wählen, um die Diagnose des OPC UA-Servers anzuzeigen.
[Diagnose des OPC UA-Servers.](#)

So öffnen Sie die aktuelle Statusansicht des OPC UA-Servers:

- Doppelklick auf **COM-Modul** und im Strukturbaum OPC UA-Server wählen.
Hier wird das konfigurierte und das aktuelle μ P-Budget des OPC UA-Servers angezeigt.

Im Control-Panel werden die folgenden Online-Informationen angezeigt:

Element	Beschreibung
Konfiguriertes μ P-Budget	Das maximale μ P-Budget des COM-Moduls aus der Konfiguration des OPC UA-Server-Sets.
Aktuelles μ P-Budget	Aktuelles μ P-Budget des COM-Moduls für die Abarbeitung des OPC UA-Server-Sets.
Konfigurierte Warnung bei μ P-Budget-Überschreitung in [%]	Konfigurierte Warnschwelle für das μ P-Budget des COM-Moduls, bei deren Überschreiten der OPC UA-Server eine Kommunikations-Warnung melden muss.
Konfigurierte Warnung für μ P-Budget ist überschritten	Der OPC UA-Server hat eine Kommunikations-Warnung vor Überschreitung des konfigurierten μ P-Budgets gemeldet.

8.1 Trace-Logging (Online)

Die Funktion *Trace-Logging* bietet Analysedaten, wenn beim Betrieb des OPC UA-Protokolls Störungen auftreten. Sobald eine Trace-Logging-Nachricht vom OPC UA-Server generiert wurde, wird diese sofort an den Syslog-Client weitergeleitet. Zum Empfang und Aufzeichnung der Trace-Logging-Nachrichten muss ein Syslog-Client bereitgestellt werden (z. B. auf dem SILworX PC).

Während des Normalbetriebs muss das Trace-Logging deaktiviert bleiben, da es die Rechenzeit und Kommunikationslast erhöht.



Der Anwender darf das Trace-Logging erst nach Aufforderung eines HIMA Mitarbeiters aktivieren, um eine Log-Datei zu erstellen, die dann von HIMA analysiert werden kann.

So öffnen Sie im Online Control-Panel des COM-Moduls den Dialog zum Editieren der Trace-Logging-Parameter:

➤ Rechtsklick auf **OPC UA-Server** und im Kontextmenü **Trace-Logging** wählen.

Im Dialog werden die aktuell eingestellten Trace-Logging-Werte angezeigt. Sollten z.B. aufgrund einer Störung diese Werte nicht verfügbar sein, werden Standardwerte angezeigt. Der Trace-Logging-Dialog kann mit oder ohne Übernahme von Änderungen geschlossen werden.

Parameter	Beschreibung
Name	OPC-UA-Server, nicht änderbar.
Level	Der Parameter <i>Level</i> bestimmt die Stufen der Trace-Ausgaben. Der benötigte Wert wird auf Anforderung von einem HIMA Mitarbeiter bestimmt. Standardwert: 0
Facility	Der Parameter <i>Facility</i> bestimmt die Art der Trace-Ausgaben. Der benötigte Wert wird auf Anforderung von einem HIMA Mitarbeiter bestimmt. Standardwert: 0
Ziel-IP-Adresse	Dies ist die IP-Adresse des Syslog-Clients, zu dem die Trace-Logging-Nachrichten gesendet werden. Standardwert: 127.0.0.1 (ausgeschaltet).
Ziel-Port	Dies ist der IP-Port des Syslog-Clients, zu dem die Trace-Logging-Nachrichten gesendet werden. Standardwert: 514

8.2 Diagnose des OPC UA-Servers

Der OPC UA-Server erzeugt die nachfolgenden Einträge in der Kurzzeitdiagnose des COM-Moduls auf dem der OPC UA-Server läuft. Die Kurzzeitdiagnose kann der Anwender die Diagnoseansicht des COM-Moduls auslesen, siehe Kapitel [Control-Panel \(Online\)](#).

8.2.1 Öffnen einer Session

Der folgende Diagnoseeintrag wird beim Verbindungsaufbau zu einem OPC UA-Client erzeugt:

*Eine OPC UA-Session (Session-ID = **sID**, Kanal-ID = **chID**) wurde für den OPC UA-Client (IP-Adresse = <Client IPv4 Adresse>) erstellt. Anzahl OPC UA-Sessions: **N** von **Nmax***

Die gesamte Kommunikation zwischen OPC UA-Client und OPC UA-Server erfolgt innerhalb einer Sitzung (Session). Der Server verwaltet die Sessions zu den Clients. Die Datenübertragung erfolgt im OPC UA-Kommunikations-Stack über einen sicheren Kanal (SecureChannel). Die erzeugte Session ist erst nach ihrer Aktivierung für die Datenübertragung bereit.

Parameter	Beschreibung
sID	Vom OPC UA-Server vergebene eindeutige Identifikation für die Session.
chID	Vom OPC UA-Server vergebene eindeutige Identifikation für den Kanal.
N	Anzahl bereits erstellter Sessions auf dem OPC UA-Server, inklusive dieser Session.
Nmax	Maximale Anzahl der Sessions, die vom OPC UA-Server gleichzeitig geöffnet werden können.

8.2.2 Aktivieren einer Session

Der folgende Diagnoseeintrag wird bei der Aufforderung des OPC UA-Servers zur Aktivierung einer Session erzeugt.

*Die OPC UA-Session mit der ID = **sID** und mit dem Session-Timeout = **TO** ms wurde auf dem Kanal mit der ID = **chID** aktiviert.*

Wenn nach Ablauf des eingestellten TimeOut-Zeitfensters keine Aktivität vom OPC UA-Client vorhanden ist, dann wird die Sitzung vom Server geschlossen. Der vom OPC UA-Client angeforderte TimeOut-Wert darf den vom Server vorgegebenen minimalen/maximalen Wert nicht unter-/überschreiten, sonst wird der angeforderte TimeOut-Wert entsprechend den vom Server vorgegebenen Grenzwerten (min, max) angepasst.

Parameter	Beschreibung
sID	Vom OPC UA-Server vergebene eindeutige Identifikation für die Session.
chID	Vom OPC UA-Server vergebene eindeutige Identifikation für den Kanal.
TO	Vom OPC UA-Client angeforderte Session-Timeout (in ms).

8.2.3 Schließen einer Session

Der folgende Diagnoseeintrag wird beim Schließen einer Session zum OPC UA-Client erzeugt.

*Die OPC UA-Session mit der ID = **sID** auf dem Kanal mit der ID = **chID** wurde geschlossen*

Parameter	Beschreibung
sID	Vom OPC UA-Server vergebene eindeutige Identifikation für die Session.
chID	Vom OPC UA-Server vergebene eindeutige Identifikation für den Kanal.

8.2.4 Erstellen einer Subscription

Der folgende Diagnoseeintrag wird beim Anlegen einer Subscription erzeugt. In der Gesamtanzahl der noch verfügbaren Subscriptions (**N**) ist das Anlegen dieser Subscription noch nicht berücksichtigt.

*Ein Request des OPC UA-Clients (OPC UA-Server-Session-ID = **sID**) zum Erstellen einer Subscription mit folgenden Parametern wurde empfangen:*

- *Publish-Intervall = **Tp** ms*
- *Maximale Anzahl von Notifications pro Publish-Response = **Nmax***
- *Life-Time-Counter = **ClT***
- *Keep-Alive_Counter = **Cka***

*Aktuell können **N** Subscriptions angelegt werden.*

Für eine genaue Erläuterung der Parameter **Tp**, **Nmax**, **ClT** und **Cka**, siehe Kapitel 5.3.12, OPC UA Unified Architecture Specification, Part 4.

Parameter	Beschreibung
sID	Vom OPC UA-Server vergebene eindeutige Identifikation für die Session.
Tp	Zyklisches Zeitintervall in ms für die Übertragung von Benachrichtigungen.
Nmax	Maximale Anzahl von Benachrichtigungen in einer Antwort.
ClT	Lifetime-Zähler.
Cka	Keep-Alive-Zähler.
N	Anzahl der Subscriptions, die auf dem OPC UA-Server noch angelegt werden können.

8.2.5 Löschen einer Subscription

Der folgende Diagnoseeintrag wird beim Löschen einer Subscription erzeugt.

*Ein Request des OPC UA-Clients (OPC UA-Server-Session-ID = **sID**) zum Löschen von **N** Subscriptions wurde empfangen. Aktuell existieren auf dem OPC UA-Server **Nc** von **Nmax** möglichen Subscriptions.*

In der Anzahl der vorhandenen Subscriptions (Parameter **Nc**) ist das Löschen dieser Subscription noch nicht berücksichtigt.

Parameter	Beschreibung
sID	Vom OPC UA-Server vergebene eindeutige Identifikation für die Session.
N	Anzahl zu löschender Subscriptions.
Nc	Aktuelle Anzahl auf dem OPC UA-Server vorhandener Subscriptions.
Nmax	Maximale Anzahl der Subscriptions, die auf dem OPC UA-Server angelegt werden können.

9 Codegenerierung und Reload

Während Sie einen Download nur durchführen können, wenn sich ein Programmierbares Elektronisches System (PES) im STOP befindet, brauchen Sie für einen Reload das System nicht anzuhalten.



Ein Reload ist ein Eingriff in ein laufendes, sicherheitsbezogenes System. Sie können einen solchen Eingriff nur durchführen, wenn Sie zuvor den CPU-Schalter **Reload erlaubt** in den Eigenschaften der Ressource aktiviert haben. Wenn Sie im Anwenderprogramm die Systemvariable *Reload-Deaktivierung* verwenden, muss auch die Variable auf FALSE gesetzt sein.

9.1 Codegenerierung

Während der Codegenerierung erzeugt SILworX pro referenziertem COM-Modul eine Konfigurationsdatei *opcuaserver.config*. Die Konfigurationsdatei hat den Anzeigenamen *OPC UA-Server* und die minimale Version SILworX V12.

Das System unterstützt maximal 15000 Nodes pro Konfiguration, bei Überschreitung dieser Menge meldet SILworX einen Fehler und eine Codegenerierung ist nicht möglich.

9.2 Reload

Der integrierte OPC UA-Server auf dem COM-Modul liefert auch während eines Reloads konsistente Daten des Systems. Während der Umstellungsphase im System kann es daher zu längeren Verarbeitungszeiten im Datenverkehr kommen.

Das System unterstützt Reload-Codegenerierung nach folgenden Änderungen im Informationsmodell:

- Das Hinzufügen von maximal 1000 Nodes in einem Vorgang, wenn diese eine höhere Node-ID haben als alle bisher im Informationsmodell vorhandenen Nodes.
- Das Ändern von maximal 1000 Node-Beschreibungen in einem Vorgang.

Findet SILworX während einer Reload-Codegenerierung nicht unterstützte Änderungen im Informationsmodell, dann kann für das COM-Modul kein Reload durchgeführt werden. Der Anwender erhält eine Reload-Warnung, dass ein Cold-Reload möglich ist.

9.3 Zertifikate

Wenn Zertifikate konfiguriert wurden, erzeugt SILworX in der Codegenerierung pro referenziertem COM-Modul eine Konfigurationsdatei *opcucertificates.config* mit dem Anzeigenamen *OPC UA-Zertifikate* und der minimalen Version SILworX V12.

Das System unterstützt Änderungen in der Zertifikat-Konfiguration durch Reload-Codegenerierung nicht. Änderungen können nur über Cold-Reload auf die Steuerung geladen werden.

Weitere Informationen zu "Zertifikate" auf Seite 30.

10 Versionsvergleich

Der Versionsvergleich erfolgt anhand der vom Codegenerator erstellten Prüfsummen (CRCs) des Projekts. Für weitere Informationen, siehe das Handbuch Versionsvergleich HI 801 285 D.

Beim Versionsvergleich werden verschiedene Ressourcenkonfigurationen miteinander verglichen und die Unterschiede zwischen den einzelnen Konfigurationsdateien angezeigt. Das Ergebnis des Versionsvergleichs hat SIL3-Qualität und beruht auf den Konfigurationsdateien, welche den ausführbaren Code beschreiben.

10.1 OPC UA-Server

Für die Konfigurationsdatei *opcuaserver.config* bietet SILworX einen Versionsvergleich mit der Beschreibung *OPC UA-Server*.

10.1.1 Sektion OPC UA

Der Versionsvergleich meldet Änderungen für folgende OPC UA-Server-Parameter:

- Standardprotokoll-ID
- Anzahl Namensräume
- Anzahl Nodes
- Anzahl Referenzen

Diese Meldungen werden in der Detailansicht des Vergleichs am Anfang der Tabelle aufgelistet.

10.1.2 Sektion Namespace

Der Versionsvergleich meldet Änderungen für folgende Namensraum-Parameter:

- Namensraum-URI
- Anzahl Objects
- Anzahl Variablen
- Anzahl Typen
- Anzahl Referenzen
- Anzahl Strings

Diese Namensraum-Meldungen erscheinen in der Detailansicht des Vergleichs nach den Einträgen für die Sektion *OpcUa*. Sie werden anhand ihres Namensraum-Index identifiziert und sortiert.

10.1.3 Sektion Node

Der Versionsvergleich meldet entfernte oder hinzugefügte Nodes sowie Änderungen für folgende allgemeine Node-Parameter:

- Node-Klasse
- Browse-Name
- Display-Name
- Beschreibung

Diese Node-Meldungen erscheinen in der Detailansicht des Vergleichs nach den Einträgen für die Sektion *Namespace*. Sie werden anhand ihrer Node-ID identifiziert und sortiert.

10.1.4 Sektion Variable

Der Versionsvergleich meldet Änderungen für folgende Variablen-Parameter:

- Dataview-Identifizier
- Zugriffsrecht
- Datentyp

Diese Meldungen erscheinen in der Detailansicht des Vergleichs unterhalb des Nodes, der die Variable repräsentiert.

10.1.5 Sektion Reference

Der Versionsvergleich meldet entfernte oder hinzugefügte Referenzen.

Die Darstellung erfolgt als String, der aus den Node-IDs von Quelle, Ziel und Typ der Referenz gebildet wird.

Beispiel: 0:85-2:10-0:47

- 0:85 - entspricht dem Objekte-Ordner im Namensraum 0 der OPC Foundation
- 2:10 - entspricht einem Node mit der ID 10 im Namensraum 2 des Anwenders
- 0:47 - entspricht der Beziehung HasComponent im Namensraum 0 der OPC Foundation

Diese Meldungen erscheinen in der Detailansicht des Vergleichers unterhalb des Nodes, der die Quelle der Referenz darstellt.

10.1.6 Sektion EventSource

Der Versionsvergleich meldet entfernte oder hinzugefügte Events sowie Änderungen für folgende allgemeine Event-Parameter:

- Event-Name
- Event-ID
- Event-Typ
- Node-ID der Globalen Variablen

Diese Event-Meldungen erscheinen in der Detailansicht des Vergleichs nach den Einträgen für die Sektion Node. Sie werden anhand des Event-Namens identifiziert und sortiert.

10.1.7 Sektion Condition

Der Versionsvergleich meldet Änderungen für folgende Event-Parameter:

- Texte für Alarm (auch HH-, H-, L-, LL- und Rückkehr zu Normalzustand-Texte)
- Priorität für Alarm (auch HH-, H-, L-, LL- und Rückkehr zu Normalzustand-Prioritäten)

Diese Meldungen erscheinen in der Detailansicht des Vergleichs unterhalb der Event-Definition, die die Condition enthält.

10.2 OPC UA-Zertifikate

Für die Konfigurationsdatei *opcuacertificates.config* bietet SILworX einen Versionsvergleiche mit der Beschreibung *OPC UA-Zertifikate*.

Der Versionsvergleich meldet Änderungen für folgende Parameter der Client-Zertifikate:

- Nur verschlüsselte Verbindungen zulassen
- Anzahl Client-Zertifikate

Diese Meldungen werden in der Detailansicht des Vergleichs am Anfang der Tabelle aufgelistet.

10.2.1 Sektion Own

Der Versionsvergleich meldet Änderungen für folgende Parameter des Server-Zertifikats:

- Zertifikat-Datei
- Schlüssel-Datei

Diese Meldungen werden in der Detailansicht des Vergleichs nach den Einträgen für die Sektion Zertifikates aufgelistet. Die geänderten Werte werden als SHA1-Hash-Strings angezeigt. Der eigentliche Inhalt des Zertifikats oder des Schlüssels wird nicht angezeigt.

10.2.2 Sektion Client

Der Versionsvergleich meldet Änderungen für folgende Parameter der Client-Zertifikate:

- Zertifikat-Datei

Diese Client-Zertifikate-Meldung erscheint in der Detailansicht des Vergleichs nach den Einträgen für die Sektion Server-Zertifikat. Sie werden anhand der Reihenfolge in der Konfigurationsdatei *Index* identifiziert und sortiert. Die geänderten Werte werden als SHA1-Hash-String angezeigt. Der eigentliche Inhalt des Zertifikats wird nicht angezeigt.

HI 801 548 D (2025)


Für weitere Informationen kontaktieren Sie:

HIMA Paul Hildebrandt GmbH

Albert-Bassermann-Str. 28
68782 Brühl, Deutschland

Telefon: +49 6202 709-0

E-Mail: info@hima.com

 www.hima.com/de/

