

OPC UA Client Prozessanschluss für den Elvis Server Benutzerdokumentation

Änderungen

Version	Name	Datum	Geänderte Abschnitte	Änderungsgrund
3.3.1.1	wp	27.07.2022	Alle	Erzeugt
3.3.1.4	wp	15.10.2022	PackML-Unterstützung	Ergänzt
3.3.1.8	wp	29.11.2023	Prozessdatenimport	Ergänzt
3.3.1.9	wp	12.12.2023	Änderung von Datenpunktwerten	Hinzugefügt

Beschreibung

OPC UA ist der Nachfolger von OPC DA. OPC UA hat gegenüber OPC DA einige Vorteile:

OPC UA basiert auf Standardprotokollen und nicht auf Windows-zentrierten Technologien, wie COM und DCOM. OPC UA definiert ein optimiertes binäres TCP-Protokoll für den lokalen Datenzugriff.

Darauf aufbauend sind Webdienste, XML, HTTPS und andere Internetkommunikationen spezifiziert, um einen Firewall-freundlichen Zugriff von außerhalb zu ermöglichen. OPC UA verwendet eine Endpunkt-URL, um anzugeben, welches Verbindungsprotokoll verwendet werden soll. Das Binärprotokoll ist `opc.tcp://Server` und `https://Server` ist für Web-Service (SOAP).

Der hier beschriebene OPC UA Client-Prozessanschluss für den Elvis Server unterstützt beide Protokoll-Varianten. In der vorliegenden ersten Version dieses Pakets ist von den verschiedenen für OPC UA spezifizierten Modulen als einziges der Data Access (DA) implementiert.

Test-Server

Für einfache Tests können folgende frei verfügbare OPC UA – Server verwendet werden:

<https://downloads.prosysopc.com/opc-ua-simulation-server-downloads.php>

<https://www.unified-automation.com/downloads.html> (nach Anmeldung, z.B. OPC UA C++ Demo Server). Nach der Projektierung des Prozessanschlusses (s.u.) können die von den Servern angebotenen OPC UA-Knoten als Datenpunkte importiert werden.

Das Beispielprojekt Elvis-Projekt `OpcUaClient`, das von unserer Web-Seite geladen werden kann, enthält einen Test-Server (s. `Readme.txt`-Datei, ebenfalls in dem Projekt). Dieser Test-Server bietet unter `Machine_1` einige häufig in Elvis verwendete Variablen an. Zudem werden Variablen der Basistypen von dem PackML-Node-Set angeboten. Es ist zu beachten, dass im Unterschied zu vielen PLC-Systemen die Byte-Order auf Intel-PC's „little endian“. Daher ist, je nach verwendetem Server-System, der Konfigurationsparameter „Reverse Byte-Order“ zu setzen (s. Projektierung unten).

Projektierung

Nach dem Laden des Paketes Itgmbh.Elvis.OpcUaClient.3.3.x steht für die Projektierung der Prozessanschluss OpcUaClient zur Verfügung. Für den Prozessanschluss sind folgende Einstellung vorzunehmen:

Eigenschaft	Beschreibung
Host Name	Der Server-Name des OPC UA-Servers.
Opc-ServiceName	Der Name des Services
Port	Der TCP-Port, default: 4880
Scheme	Transportschema. Auswahl aus opc.tcp, http, https
UserName	Anmeldename
Passwort	Passwort.
Reverse Byte-Order	Keht die Anordnung der Bytes der Variablenwerte beim Senden und Empfangen um.

Prozessdatenimport

Setzen der Tag-Eigenschaft

Die Tag-Eigenschaft kann auf den Wert

Identify By Path

gesetzt werden. Damit wird erreicht, dass bei einem erneuten Prozessdatenimport ein Datenpunkt mit gleichem Namen, aber unterschiedlichem Pfad als neuer Datenpunkt angenommen werden. Ohne diesen Tag-Wert wird von einer Modifikation des gleichnamigen bereits vorhandenen Datenpunktes ausgegangen.

Automatisches Anlegen von Funktionen und Funktionstypen

Sind in der PLC (programmable logic controller) sogenannte Variablenblöcke angelegt, wird die übergeordnete Variable beim Prozessdatenimport als Funktion angelegt.

Beispiel für eine Konfiguration

Der Demo-Server von Prosys (Download-Link s.o.) zeigt nach dem Start eine Verbindungsadresse an, die in etwa so aussieht:

opc.tcp://rechner:53530/OPCUA/SimulationServer. Die Konfigurations-Eigenschaften für eine Verbindung mit diesem Server sind dann:

Host Name: rechner

Opc-ServiceName: OPCUA/SimulationServer

Scheme: opc.tcp

Der Demo-Server unterstützt verschiedene Verbindungspunkte (Endpoints), u.a. auch mit Security Policy None. Das bedeutet, dass eine anonyme Verbindung, ohne die Übertragung von Benutzername und Passwort zulässig ist und somit keine Eintragungen für die entsprechenden Eigenschaften der Konfiguration benötigt werden.

Konvertierung von Serverdatentypen

Wenn vom Server Datentypen verwendet werden, die nicht ohne Weiteres auf Datentypen des Elvis-Clients abgebildet werden können, besteht die Möglichkeit, die vom Server gelieferten Variablen-Werte

zur Laufzeit zu konvertieren. Hierfür müssen die Typen der vom Prozess importierten Variablen angepasst werden. Die folgende Tabelle zeigt die aktuell unterstützten Konvertierungen.

Server-Datentyp	Client-Datentyp	Prozess-TypInfo	Anmerkung
DT	DateTime	OPCUA_DT	Konvertiert ein Byte-Array der Länge 8 mit einem BCD kodierten Datum-/Zeitwert in das DateTime format.
CHAR	String	OPCUA_CHAR	Konvertiert ein UInt8-Array beliebiger Länge in einen String.

Beispiel: Die Definition eines Datenpunkttyps, der eine Konvertierung von UInt8-Array in einen String durchführt, sieht wie folgt aus:

Name: OPCStringFromCHAR

Datentyp: String

Prozess-TypInfo: OPCUA_CHAR

Änderung von Datenpunktwerten

Änderungen von Datenpunktwerten werden dem Client über Events mitgeteilt. Der Client muss, um Events zu erhalten, die entsprechenden Knoten auf der PLC abonniert haben. Bei der Projektierung ist zu berücksichtigen, dass die Anzahl der möglichen Abonnements auf den PLC's unterschiedlich begrenzt ist. Die Anzahl der möglichen Abonnements/Subscriptions entnehmen sie bitte der Beschreibung der eingesetzten PLC.