

# Benutzerhandbuch



ASNeG

Server

Neukirchen

August 2015

## **Kontakt:**

Kai Hübl  
Lambertsberg 17  
D-34626 Neukirchen  
[kai.huebl@asneg.de](mailto:kai.huebl@asneg.de)

---

# Contents

<b>1</b>	<b>Einleitung.....</b>	<b>5</b>
1.1	Inhalt .....	5
1.2	OPC UA Client Stack .....	5
1.3	OPC UA Server Stack.....	6
1.4	OpcUaServer.....	6
1.5	ASNeG-Demo Server .....	7
<b>2</b>	<b>Installation .....</b>	<b>10</b>
2.1	Windows 7 .....	10
2.1.1	Systemvoraussetzungen .....	10
2.1.2	Installation .....	10
2.1.3	Starten und Stoppen des Server .....	13
2.2	ubuntu 14.04 LTS .....	15
2.2.1	Systemvoraussetzungen .....	15
2.2.2	Installation .....	15
2.2.3	Starten und Stoppen des Server .....	17
2.3	openSuse 13.3 .....	17
2.3.1	Systemvoraussetzungen .....	17
2.3.2	Installation .....	17
2.3.3	Starten und Stoppen des Server .....	18
<b>3</b>	<b>Konfiguration.....</b>	<b>18</b>
3.1	Überblick .....	18
3.2	OpcUaServer.....	19
3.2.1	Vordefinierte Makros.....	19
3.2.2	Allgemeine Konfiguration .....	20
3.2.3	Logging .....	20
3.2.4	Anwendung .....	20
3.2.5	Endpoints .....	22
3.2.6	Information Model.....	22
3.3	ASNeG-Demo OPC UA Informationsmodell .....	23
<b>4</b>	<b>FAQ.....</b>	<b>23</b>
4.1	Allgemein.....	23
4.2	Linux .....	24
4.2.1	Windows .....	26



# 1 Einleitung

## 1.1 Inhalt

Das Installations-Paket OpcUaStack enthält mehrere Libraries und Anwendungen, die für verschiedene Zwecke eingesetzt werden können. Die folgende Liste gibt einen Überblick über den Inhalt des Installations-Paketes.

- OPC UA Client Stack
- OPC UA Server Stack
- OpcUaServer
- ASNeG-Demo Server

## 1.2 OPC UA Client Stack

Der OPC UA Client Stack enthält Funktionen des OPC UA Standards für den Zugriff auf einen OPC UA Server. Der OPC UA Client Stack wird in Form einer dynamischen Library bereitgestellt und kann in neue und bestehende C/C++ Anwendungen integriert werden. Die Anwendung kann somit über die OPC UA Client Schnittstelle auf einen OPC UA Server zugreifen.

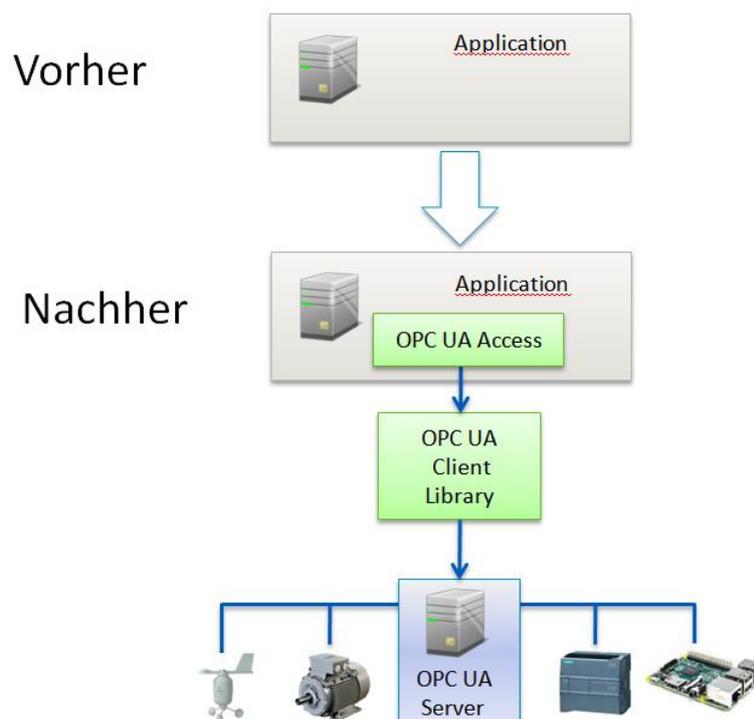


Abbildung 1 – OpcUaClient Stack

## 1.3 OPC UA Server Stack

Der OPC UA Server Stack enthält Funktionen des OPC UA Standards für den Betrieb eines OPC UA Servers. Der OPC UA Server Stack wird in Form einer dynamischen Library bereitgestellt und kann in neue und bestehende C/C++ Anwendungen integriert werden. Die Anwendung kann somit um eine OPC UA Standard Schnittstelle erweitert werden.

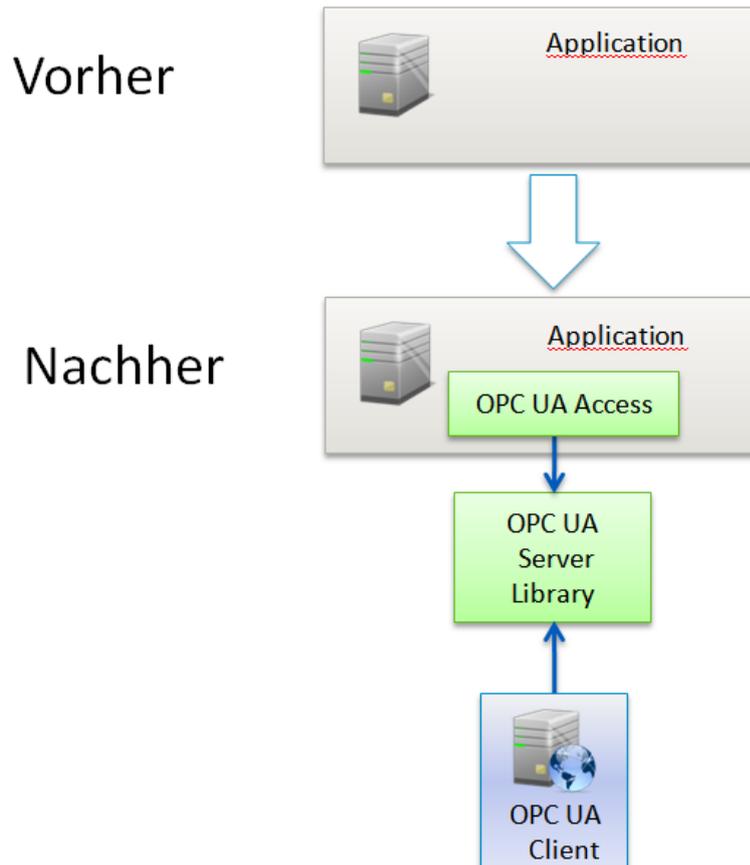


Abbildung 2 - OpcUaServer Stack

## 1.4 OpcUaServer

Der OpcUaServer ist ein Anwendungsserver mit dem beliebige OPC UA Client Server Anwendungen auf unterschiedlichen Betriebssystemen ausgeführt werden können. Der OpcUaServer besteht aus einem OPC UA Server, einem OPC UA Client und einer Produktschnittstelle. Die Produkt Library enthält die eigentliche Logik der Anwendung, die durch den OpcUaServer ausgeführt wird. Die Verknüpfungen zwischen den einzelnen Komponenten des OpcUaServers werden über die Konfiguration hergestellt. Somit können vom OpcUaServer beliebige OPC UA Client Server Anwendungen ausgeführt werden.

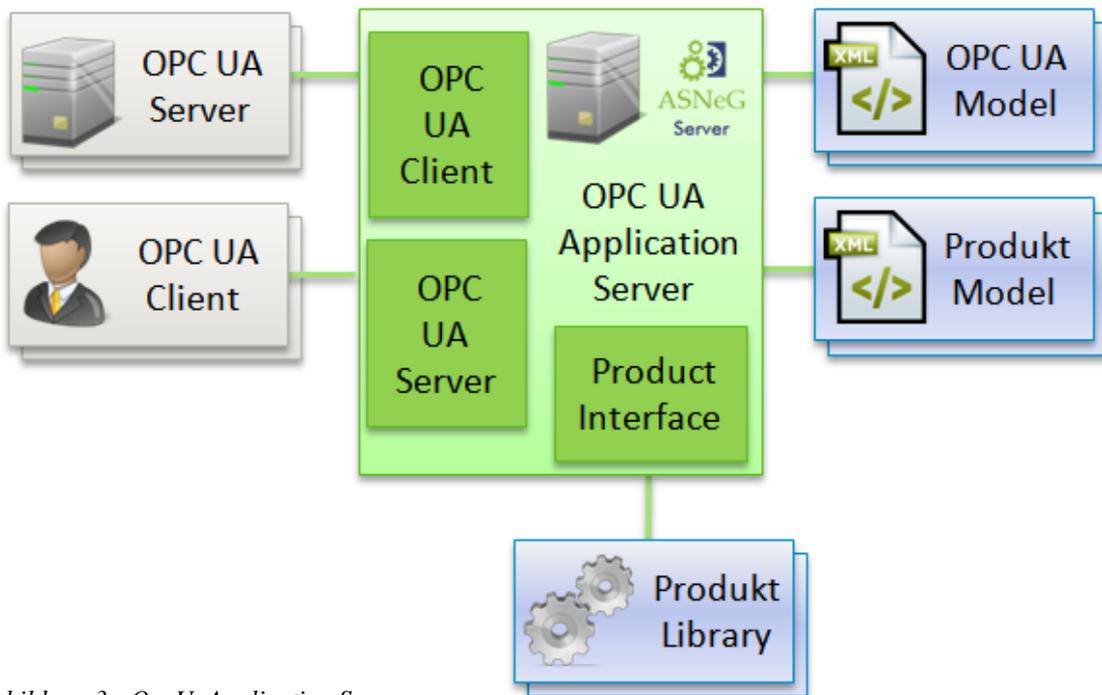


Abbildung 3 - OpcUaApplication Server

Der OPC UA Server stellt das Informationsmodell sowie eine Serverschnittstelle auf Basis des OPC UA Protokolls bereit. OPC UA Clients können über diese Schnittstelle auf die Funktionen des OPC UA Servers zugreifen. Das Informationsmodell des OPC UA Servers kann in der Konfiguration statisch festgelegt oder von der Produkt Library dynamisch während der Laufzeit des OpcUaServers geändert werden.

Der OPC UA Client stellt eine Clientschnittstelle auf Basis des OPC UA Protokolls bereit. Die Produkt Library kann über diese Schnittstelle auf die Funktionen eines oder mehrerer anderer OPC UA Server zugreifen.

Die Produktschnittstelle wird vom OpcUaServer zum Laden einer oder mehrerer Produkt Libraries verwendet. Eine Produkt Library enthält die eigentliche Anwendung des OpcUaServers. Durch dieses Konzept können vom OpcUaServer beliebige OPC UA Client Server Anwendungen ausgeführt werden. Beispiele für OPC UA Client Server Anwendungen sind ein OPC UA Modbus Gateway, ein OPC UA Dienst Server für die Überwachung eines Produktionsprozesses oder eine Historische Datenbank für die Speicherung und Verwaltung von historischen Daten. Die auszuführenden Produkt Libraries werden in der Konfigurationsdatei des OpcUaServer festgelegt.

## 1.5 ASNeG-Demo Server

Der ASNeG-Demo Server stellt für Testzwecke ein Informationsmodell mit zwei Folder zur Verfügung.

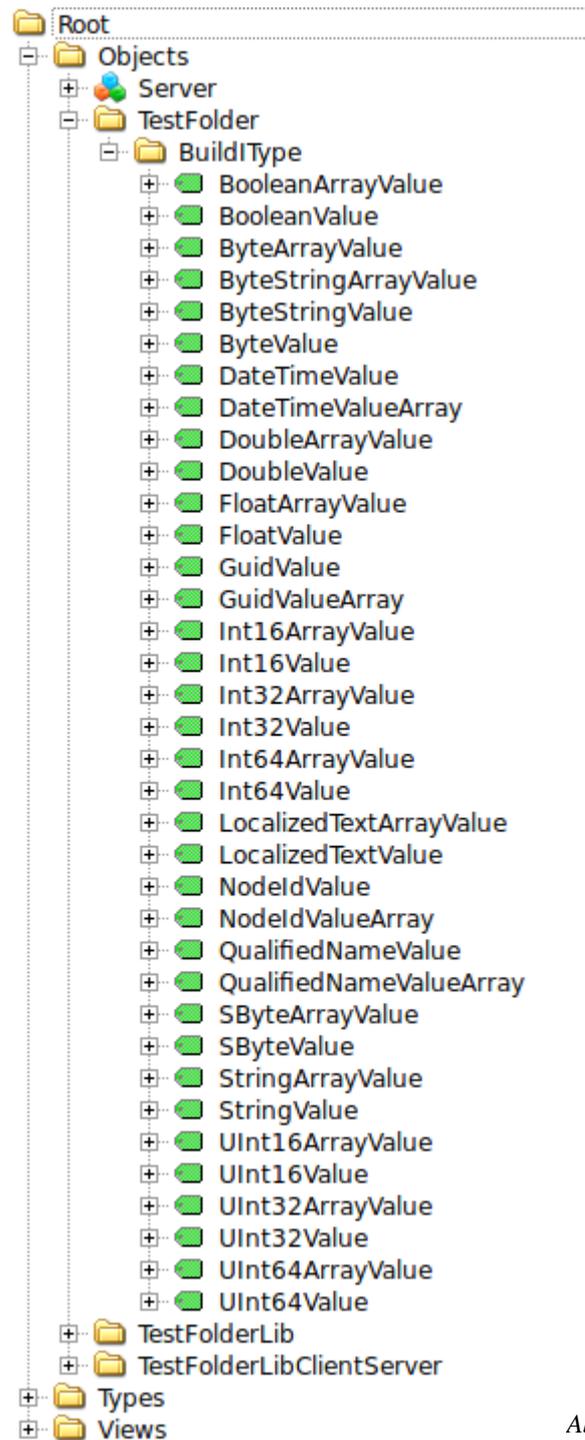


Abbildung 4 - Informationsmodell

Im Folder *Root/Objects/TestFolder/BuildInType* ist für jeden OPC UA BuildInType eine Variable vorhanden. Diese Variablen können von einem OPC UA Client gelesen, geschrieben oder überwacht werden.

Alle OPC UA Anfragen werden durch das Informationsmodell des OpcUaServer beantwortet.

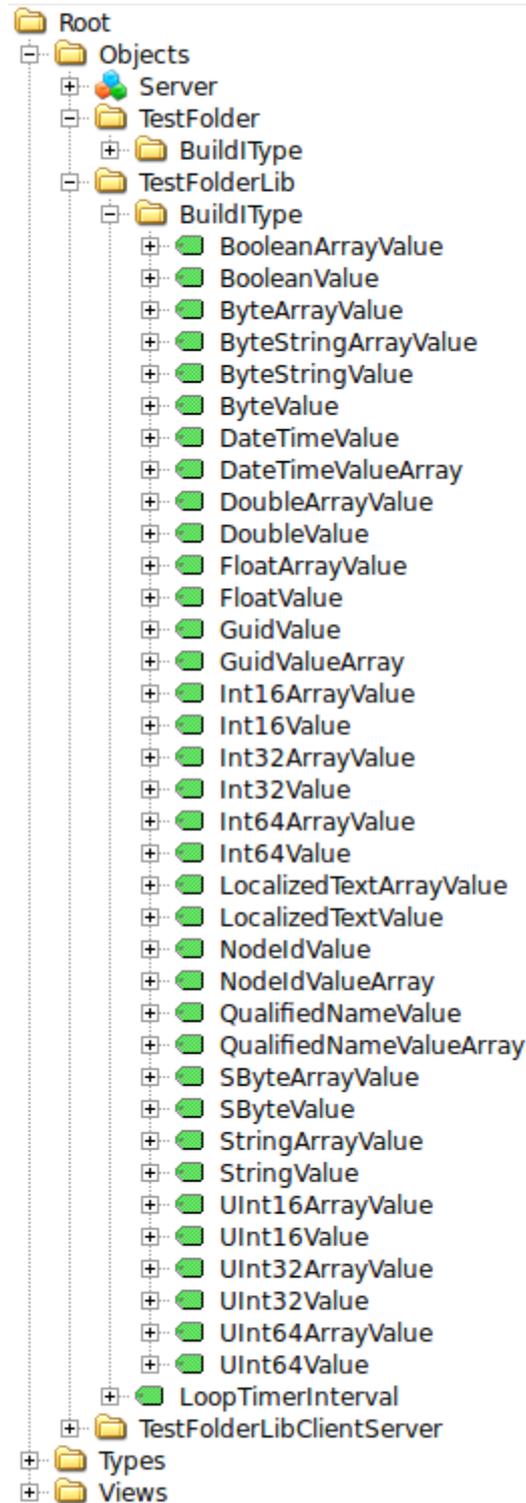


Abbildung 5 – Informationsmodell  
mit Loop

Im Folder *Root/Objects/TestFolderLib/BuildInType* ist für jeden OPC UA BuildInType eine Variable vorhanden. Diese Variablen können von einem OPC UA Client gelesen, geschrieben oder überwacht werden.

Alle OPC UA Anfragen werden nicht durch das Informationsmodell des OpcUaServer beantwortet, sondern an eine Produkt Library weitergeleitet. Die Produkt Library verwaltet

in diesem Fall die Variablen und die Zugriffe auf diese Variablen. Das Informationsmodell dient nur als Schnittstelle.

Die Variable *LoopTimerInterval* hat eine Sonderbedeutung. Diese Variable enthält ein Zeitintervall in Millisekunden. Ist das *LoopTimerInterval* ungleich 0 so werden alle Variablen im vorgegebenen Zeitintervall durch die Produkt Library geändert. Die Variable *LoopTimerInterval* kann wie alle Variablen durch ein OPC UA Client gesetzt bzw. gelesen werden.

## 2 Installation

Dieser Abschnitt beschreibt wie der OpcUaServer auf verschiedenen Betriebssystemen installiert werden kann.

### 2.1 Windows 7

Dieser Abschnitt beschreibt wie der OpcUaServer unter dem Betriebssystemen Windows 7 installiert werden kann.

#### 2.1.1 Systemvoraussetzungen

Für die Installation des OpcUaServer müssen die folgenden Voraussetzungen vor der Installation erfüllt sein.

- Installation der Boost Library in der Version 1.54 32 Bit.  
(<http://sourceforge.net/projects/boost/files/boost-binaries/1.54.0/>)  
Für die Installation muss das Paket *boost\_1\_54\_0-msvc-11.0-32.exe* ausgewählt werden. Als Installationsverzeichnis während der Installation der Boost Library muss das Verzeichnis *C:/local/boost\_1\_54\_0* ausgewählt werden.
- Installation der OpenSSL in der Version 1.0.2d (32Bit)  
(<http://heise.de/download/win32-openssl.html>)

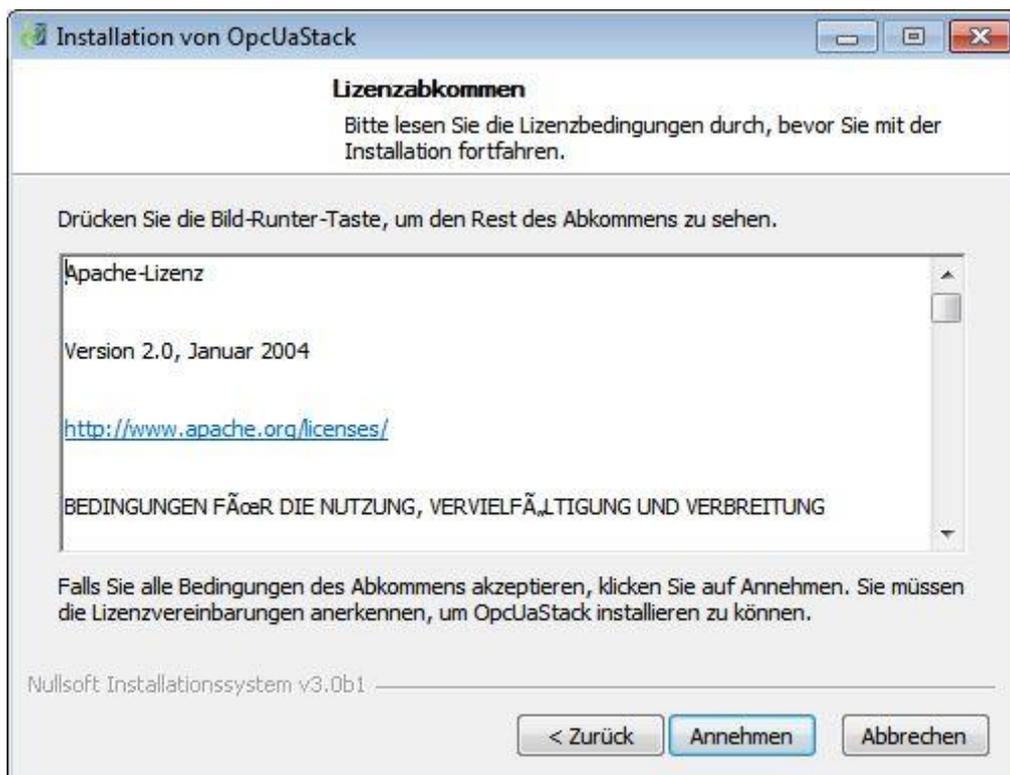
#### 2.1.2 Installation

Für die Installation des OpcUaServer müssen die folgenden Schritte durchgeführt werden.

- Laden Sie den OpcUaServer für das Betriebssystem Windows 7 von der ASNeG Webseite herunter (<http://asneg.de/download.html>). Die Installation startet nach dem Download selbstständig. Für die Installation des OpcUaServers auf Ihrem Windows 7 Betriebssysteme werden Administrator Rechte benötigt. Nach dem Ende der Installation des OpcUaServers muss Ihr Rechner neu gestartet werden.



- Drücken Sie *Weiter*.



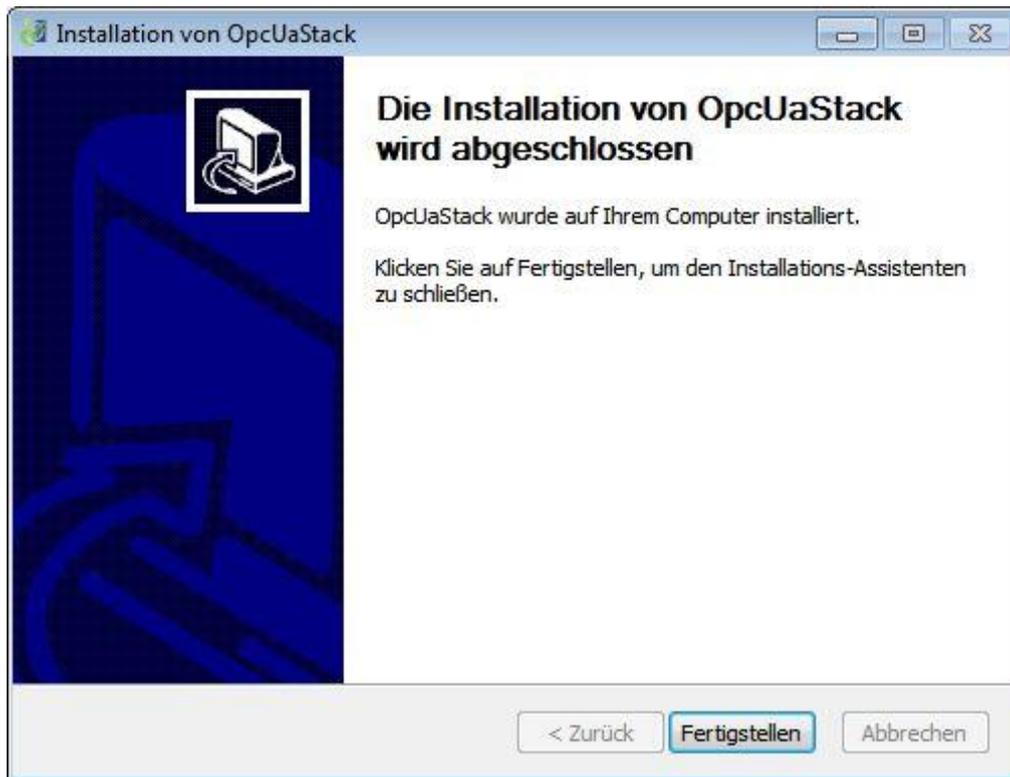
- Lesen Sie das Lizenzabkommen und bestätigen Sie dieses durch das Drücken von *Annehmen*.



- Drücken Sie *Weiter*.



- Drücken Sie *Installieren*.



Drücken Sie *Fertigstellen*.

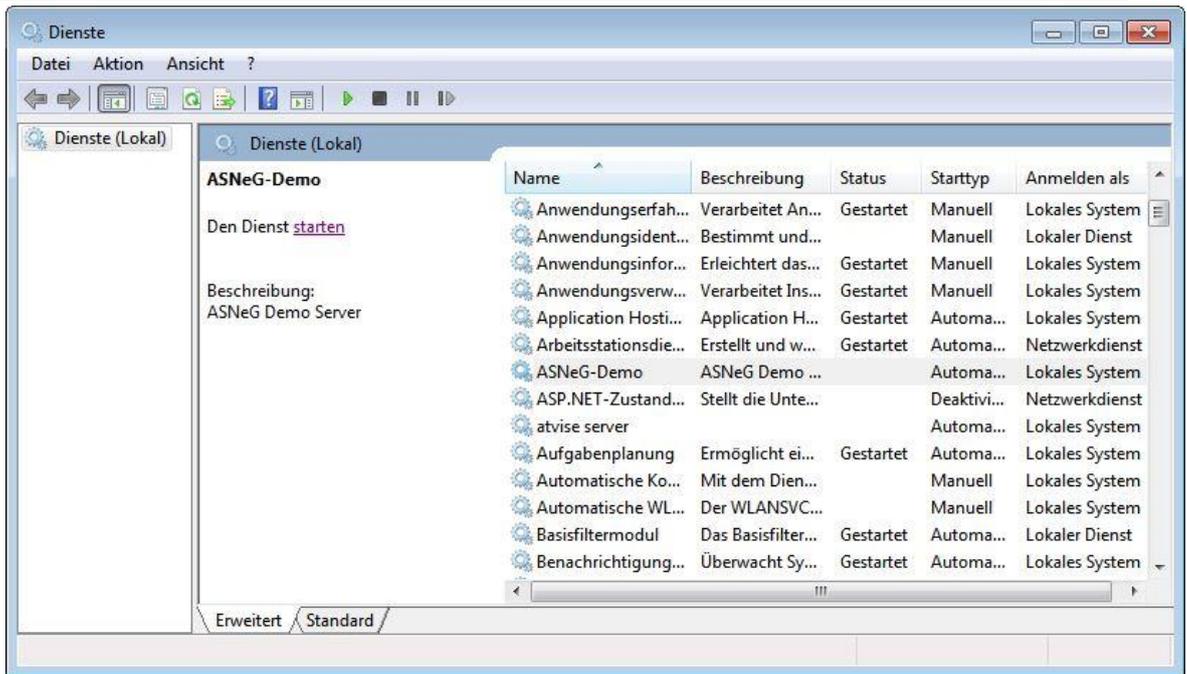
- Führen Sie jetzt einen Neustart des Rechners durch.
- Nach dem Neustart des Rechners ist die Installation des OpcUaServer abgeschlossen.

### 2.1.3 Starten und Stoppen des Server

Der OpcUaServer wird als Windows Dienst auf dem Windows 7 Betriebssystem installiert. Am Ende der Installation sowie nach einem Neustart des Rechners wird der ASNeG-Demo Server automatisch neu gestartet. Für das manuelle Starten und Stoppen des ASNeG-Demo Servers kann der Windows Dienste Manager verwendet werden.

## Starten des ASNeG-Demo Servers

- Wählen Sie das Menü „Windows System Menü“
- Wählen Sie das Menü „Systemsteuerung“
- Wählen Sie das Menü „Verwaltung“
- Wählen Sie das Menü „Dienste“

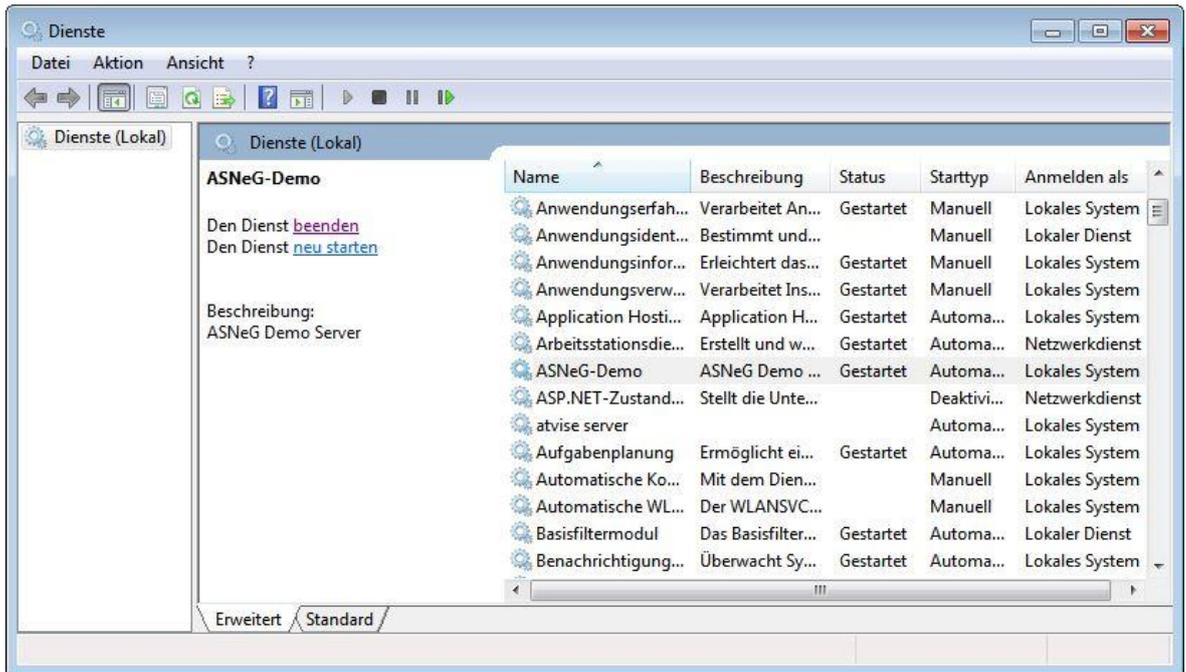


- Wählen Sie im rechten Fenster die Zeile *ASNeG-Demo* aus.
- Drücken Sie im mittleren Fenster *starten*. Der Windows Dienst ist nun gestartet.

Ob der ASNeG-Demo Server gestartet ist erkennen Sie am Statusfeld im rechten Fenster. Ist der ASNeG-Demo Server gestoppt, so enthält das Statusfeld keinen Wert. Ist der ASNeG-Demo Server gestartet, so enthält das Statusfeld den Inhalt „Gestartet“

## Stoppen des ASNeG-Demo Servers

- Wählen Sie das Menü „Windows System Menü“
- Wählen Sie das Menü „Systemsteuerung“
- Wählen Sie das Menü „Verwaltung“
- Wählen Sie das Menü „Dienste“



- Wählen Sie im rechten Fenster die Zeile *ASNeG-Demo* aus.
- Drücken Sie im mittleren Fenster *beenden*. Der Windows Dienst wird nun gestoppt.

## 2.2 ubuntu 14.04 LTS

Dieser Abschnitt beschreibt wie der OpcUaServer unter dem Betriebssystem ubuntu 14.04 LTS installiert werden kann.

### 2.2.1 Systemvoraussetzungen

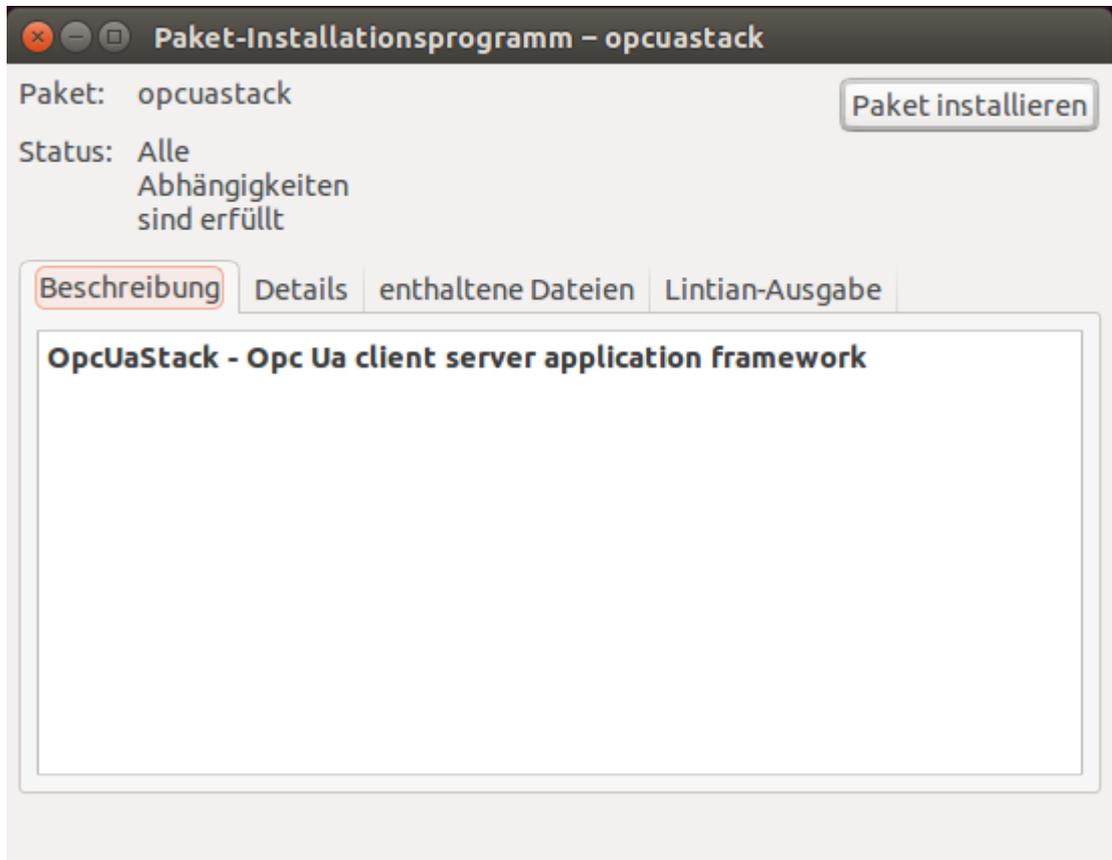
Für die Installation des OpcUaServer müssen die folgenden Voraussetzungen vor der Installation erfüllt sein.

- Installation der Boost Library in der Version 1.54.
  - >> `sudo apt-get install libboost1.54-dev`
  - >> `sudo apt-get install libboost-filesystem1.54-dev`
  - >> `sudo apt-get install libboost-thread1.54-dev`
- Installation der OpenSSL

### 2.2.2 Installation

Für die Installation des OpcUaServers müssen die folgenden Schritte durchgeführt werden.

- Laden Sie den OpcUaServer für das Betriebssystem ubuntu von der ASNeG Webseite herunter (<http://asneg.de/download.html>).
- Führen Sie das Installations Programm *gdebi-gtk* aus.  
>> sudo gdebi-gtk OpcUaStack.x.x.x-x86\_64.deb



- Drücken Sie *Paket installieren*.



- Drücken Sie *Schließen*. Damit ist die Installation abgeschlossen.

### 2.2.3 Starten und Stoppen des Server

Für das Starten und Stoppen der ASNeG-Demo Anwendung wird unter dem Betriebssystem das Init-System *upstart* eingesetzt. Nach der Installation und nach einem Neustart wird die ASNeG-Demo Anwendung automatisch neu gestartet. Zum Starten, Stoppen und Abfragen vom aktuellen Status können die folgenden Kommandos verwendet werden.

#### **Starten der ASNeG-Demo Anwendung:**

```
>> sudo initctl start ASNeG-Demo
```

#### **Stoppen der ASNeG-Demo Anwendung:**

```
>> sudo initctl stop ASNeG-Demo
```

#### **Abfrage des Status der ASNeG –Demo Anwendung:**

```
>> sudo initctl status ASNeG-Demo
```

## 2.3 openSuse 13.3

Dieser Abschnitt beschreibt wie der OpcUaServer unter dem Betriebssystem openSuse 13.3 installiert werden kann.

### 2.3.1 Systemvoraussetzungen

Für die Installation des OpcUaServer müssen die folgenden Voraussetzungen vor der Installation erfüllt sein.

- Installation der Boost Library in der Version 1.54 mit Yast.

### 2.3.2 Installation

Für die Installation des OpcUaServer müssen die folgenden Schritte durchgeführt werden.

- Laden Sie den OpcUaServer für das Betriebssystem openSuse von der ASNeG Webseite herunter (<http://asneg.de/download.html>).
- Führen Sie das Installations-Programm *rpm* aus.  
>> `sudo rpm -i OpcUaStack.x.x.x-x86_64.rpm`

### 2.3.3 Starten und Stoppen des Server

Für das Starten und Stoppen der ASNeG-Demo Anwendung wird unter dem Betriebssystem das Init-System *systemd* eingesetzt. Nach der Installation und nach einem Neustart wird die ASNeG-Demo Anwendung automatisch neu gestartet. Zum Starten, Stoppen und Abfragen vom Status können die folgenden Kommandos verwendet werden.

#### **Starten der ASNeG-Demo Anwendung:**

>> `sudo systemctl start ASNeG-Demo.service`

#### **Stoppen der ASNeG-Demo Anwendung:**

>> `sudo systemctl stop ASNeG-Demo.service`

#### **Abfrage des Status der ASNeG –Demo Anwendung:**

>> `sudo systemctl status ASNeG-Demo.service`

## 3 Konfiguration

Dieser Abschnitt beschreibt wie der OpcUaServer konfiguriert werden kann.

### 3.1 Überblick

Das Programm OpcUaServer kann auf einem Rechner mehrfach ausgeführt werden. Daher muss für jeden ausgeführten OpcUaServer eine eigene Konfiguration in einem eigenen Verzeichnis bereitgestellt werden. Das Verzeichnis wird wie folgt festgelegt.

---

<INSTALL\_DIR>/etc/OpcUaStack/<Service>

- <INSTALL\_DIR> - Installations-Verzeichnis des OpcUaServer. Das Installations-Verzeichnis ist Abhängig vom Betriebssystem.
- <SERVICE> - Name des Service der durch den OpcUaServer ausgeführt werden soll.

Im Installations-Paket des OpcUaStacks ist zusätzlich zum Programm OpcUaServer eine Demo Anwendung mit dem Name ASNeG-Demo enthalten. Die folgende Abbildung zeigt das bereitgestellte Verzeichnisse und die Dateien für die Konfiguration der Demo Anwendung ASNeG-Demo.

**TODO:** Bild: ASNeG-Demo Konfiguration

## 3.2 OpcUaServer

Die Konfiguration des OpcUaServers wird in einer XML Datei abgelegt. Für die Beschreibung der einzelnen XML-Tags in der Konfiguration wird die folgende Syntax verwendet.

- [<Tag>] Der Tag ist in der Konfiguration keinmal oder genau einmal enthalten
- <Tag> Der Tag ist in der Konfiguration genau einmal enthalten
- <Tag>\* Der Tag ist in der Konfiguration keinmal oder mehrfach enthalten
- <Tag>+ Der Tag ist in der Konfiguration einmal oder mehrfach enthalten

### 3.2.1 Vordefinierte Makros

Der OpcUaServer kennt einige vordefinierte Makros die in der Konfigurations-Datei verwendet werden können. Diese Makros werden durch den OpcUaServer beim Start vom OpcUaServer durch den realen Wert ersetzt. Die folgende Tabelle gibt einen Überblick über die Makros.

Makro	Beschreibung
@INSTALL_DIR@	Installations-Verzeichnis der Anwendung, die durch den OpcUaServer ausgeführt werden soll.
@BIN_DIR@	Verzeichnis in dem sich die Binärdateien befinden.
@CONF_DIR@	Verzeichnis in dem sich die Konfigurationsdateien befinden.
@LOG_DIR@	Verzeichnis in dem sich die Logdateien befinden.

### 3.2.2 Allgemeine Konfiguration

<b>OpcUaServer</b>	Die gesamte Konfiguration wird mit diesem Tag umschlossen.
<b>Logging</b>	In diesem Bereich wird das Logging für den OpcUaServer festgelegt.
<b>Application</b>	In diesem Bereich werden die Libraries festgelegt, die durch den OpcUaServer geladen werden sollen.
<b>Endpoints</b>	In diesem Bereich werden die vom OpcUaServer verwendeten OPC UA Endpunkte festgelegt.
<b>InformationModel</b>	In diesem Bereich werden die Informationsmodelle festgelegt, die durch den OpcUaServer geladen werden.

### 3.2.3 Logging

<b>Logging</b>	Mit diesem Tag wird das Logverhalten für den OpcUaServer festgelegt.
<b>[FileLogger]</b>	Mit diesem Tag wird festgelegt, dass alle Logausgaben durch den Datei Logger ausgegeben werden sollen. Ein Datei Logger schreibt alle Logausgaben in eine Ausgabedatei.
<b>LogFileName</b>	Verzeichnis und Name der Logdatei.
<b>MaxLogFileNumber</b>	Maximale Anzahl der Logdateien.
<b>MaxLogFileSize</b>	Maximale Größe einer Logdatei.
<b>LogLevel</b>	Gibt die Priorität der auszugebenen Logmeldungen an. Mögliche Werte sind: Trace Debug Info Warning Error

### 3.2.4 Anwendung

<b>Application</b>	In diesem Bereich werden die Libraries festgelegt die durch den OpcUaServer geladen werden sollen.
--------------------	--

---

<b>ApplicationLibrary+</b>	<p>Mit diesem Tag wird der Name der Library festgelegt, die vom OpcUaServer geladen werden soll. Je nach verwendetem Betriebssystem wird der Name durch den OpcUaServer erweitert.</p> <p>Windows: <i>&lt;LibraryName&gt;.DLL</i> Linux: <i>lib&lt;LibraryName&gt;.so</i></p> <p>Die Libraries werden vom OpcUaServer im Verzeichnis <i>&lt;INSTALL_DIR&gt;/usr/lib/OpcUaStack</i> erwartet.</p>
----------------------------	--

### 3.2.5 Endpoints

<b>Endpoints</b>	In diesem Bereich werden die vom OpcUaServer verwendeten OPC UA Endpunkte festgelegt.
<b>EndpointDescription+</b>	Mit diesem Tag wird ein OPC UA Endpunkt festgelegt.
<b>EndpointUrl</b>	Mit diesem Tag wird die URL des OPC UA Endpunkt festgelegt.
<b>ApplicationUrl</b>	Mit diesem Tag wird der URI des OPC UA Endpunkt festgelegt.
<b>ProductUri</b>	Mit diesem Tag wird die Produkt URI des OPC UA Endpunkt festgelegt.
<b>ApplicationName</b>	Mit diesem Tag wird der Name des OPC UA Endpunkt festgelegt.
<b>DiscoveryUrl</b>	Mit diesem Tag wird die Discovery URL des OpcUaServer festgelegt.
<b>SecurityPolicyUri</b>	Mit diesem Tag wird die Security Policy URI festgelegt. In der aktuellen Implementierung des OpcUaServer ist hier nur der Wert „ <i>http://opcfoundation.org/UA/SecurityPolicy#None</i> “ gültig.
<b>UserTokenPolicy</b>	Mit diesem Tag wird die Policy für den Benutzer festgelegt.
<b>PolicyId</b>	Mit diesem Tag wird die Id des Benutzers festgelegt. In der aktuellen Implementierung des OpcUaServer ist hier nur der Wert „ <i>OpcUaStack</i> “ gültig.
<b>TokenType</b>	Mit diesem Tag wird der Type des Benutzers festgelegt. In der aktuellen Implementierung des OpcUaServer ist hier nur der Wert „ <i>Anonymous</i> “ gültig.
<b>TransportProfileUri</b>	Mit diesem Tag wird der Transport Profile Uri festgelegt. In der aktuellen Implementierung ist nur der Wert „ <i>http://opcfoundation.org/UA-Profile/Transport/uatcp-uasc-uabinary</i> “ gültig.
<b>SecurityLevel</b>	Mit diesem Tag wird der Security Level des Endpunktes festgelegt.

### 3.2.6 Information Model

<b>InformationModel</b>	In diesem Bereich werden die Informationsmodelle festgelegt die durch den OpcUaServer geladen werden.
<b>NodeSetFile+</b>	Mit diesem Tag wird der Name und das Verzeichnis der Node Set Datei festgelegt, die durch den OpcUaServer geladen werden soll.

---

### 3.3 ASNeG-Demo OPC UA Informationsmodell

Das in einem OpcUaServer verwendete Informationsmodell wird in einer oder in mehreren Node Set Dateien abgelegt. Als Format für eine Node Set Datei wird das XML Format verwendet. Das Schema für eine Node Set Datei ist in der OPC UA Spezifikation festgelegt. Daher wird an dieser Stelle nicht weiter auf das Format einer Node Set Datei eingegangen.

## 4 FAQ

### 4.1 Allgemein

- **Welchen Port öffnet der OPC UA Server?**

Der OPC UA Server öffnet den Port 8888 auf allen Interfaces. Als Endpoint Url in einem OPC UA Client (z.B. UaExpert) kann "opc.tcp://127.0.0.1:8888" bzw. "opc.tcp://<Hostname>:8888" verwendet werden. Den String <Hostname> müssen Sie durch die IP-Adresse ihres Rechners ersetzen.

- **Welche Libraries benötigt der OPC UA Server?**

Vor dem Start des OPC UA Server muss sichergestellt werden, dass die folgenden Libraries auf dem System installiert sind.

- Boost-Library
- OpenSSL

- **Wo liegen die Konfigurationsdateien für den ASNeG-Demo-Server**

Die Konfigurationsdateien des ASNeG-Demo-Server befinden sich in dem folgenden Verzeichnis.

Windows:

*C:/Program Files (x86)/OpcUaStack/etc/OpcUaStack/ASNeG-Demo*

Linux:

*/etc/OpcUaStack/ASNeG-Demo*

Die Datei OpcUaServer.xml enthält die Konfigurationsdatei des ASNeG-Demo-Server. Alle anderen Dateien enthalten Nodesets, in denen der Adressraum des OPC UA Servers beschrieben wird.

---

## 4.2 Linux

- **Wie findet man heraus ob der Port des OPC UA geöffnet ist?**

Mit dem netstat Kommando kann man prüfen, ob der Port 8888 geöffnet ist. Führen Sie hierzu das netstat Kommando auf der Kommandozeile als Root-Benutzer wie folgt aus.

```
> netstat -na | grep 8888
```

Der Port ist geöffnet wenn in der Ausgabe des netstat Kommando die folgende Zeile erscheint.

```
tcp    0    0 0.0.0.0:8888      0.0.0.0:*        LISTEN
```

- **Wie findet man heraus ob der OPC UA Server gestartet ist?**

Mit dem Kommando ps kann man prüfen, ob der OPC UA Server gestartet ist. Führen Sie hierzu das ps Kommando auf der Kommandozeile als Root-Benutzer wie folgt aus.

```
> ps waux | grep ASNeG
```

Der OPC UA Server ist gestartet wenn in der Ausgabe des ps Kommando die folgende Zeile erscheint.

```
huebl 6113 1.5 0.1 221312 24840 pts/15  Sl+  21:30  0:11 OpcUaServer \
ASNeG-Demo
```

- **Wie startet und stoppt man den OPC UA Server?**

Der OPC UA Server wird nach der Installation automatisch als Service gestartet. In Abhängigkeit der verwendeten Linux Distribution kann eines der Kommandos initctl, systemctl oder service verwendet werden.

- **Kann man den OPC UA Server auch auf der Kommandozeile starten?**

Ja, der OPC UA Server kann auch über die Kommandozeile gestartet werden. Führen Sie hierzu das die folgenden Kommandos auf der Kommandozeile als Root-Benutzer aus.

```
> sudo su
```

---

```
> export LD_LIBRARY_PATH=/usr/lib/OpcUaStack/:$LD_LIBRARY_PATH
> OpcUaServer ASNeG-Demo
```

Startet der Server, so erscheint die folgende Ausgabe auf der Kommandozeile.

```
DemoLibrary::construct
DemoLibrary::startup
Application::receive
Index=0 Namespace=http://opcfoundation.org/UA/
Index=1 Namespace=http://yourorganisation.org/Test-Server/
Index=2 Namespace=http://yourorganisation.org/Test-Server-Lib/
Index=3 Namespace=http://yourorganisation.org/Test-Client-Server-Lib/
Application::receive
Application::receive
Application::receive
```

- **Wurden alle abhängige Libraries installiert?**

Mit dem Kommando ldd kann man prüfen, ob alle benötigten Libraries korrekt installiert sind.. Führen Sie hierzu das ldd Kommando auf der Kommandozeile als Root-Benutzer wie folgt aus.

```
> ldd /usr/bin/OpcUaServer
```

Sind alle benötigten Libraries auf dem System vorhanden, so erscheint die folgende Ausgabe auf der Kommandozeile. Die Versionen können sich je nach verwendetem Betriebssystem unterscheiden.

```
/usr/lib/arm-linux-gnueabi/libcofi_rpi.so (0x76f50000)
libOpcUaStackCore.so => /usr/lib/OpcUaStack/libOpcUaStackCore.so (0x7679f000)
libOpcUaStackServer.so => /usr/lib/OpcUaStack/libOpcUaStackServer.so (0x7631a000)
libOpcUaStackClient.so => /usr/lib/OpcUaStack/libOpcUaStackClient.so (0x7620c000)
libdl.so.2 => /lib/arm-linux-gnueabi/libdl.so.2 (0x761f0000)
libboost_system.so.1.49.0 => /usr/lib/libboost_system.so.1.49.0 (0x761e6000)
libboost_unit_test_framework.so.1.49.0 => /usr/lib/libboost_unit_test_framework.so.1.49.0 (0x7615a000)
libboost_filesystem.so.1.49.0 => /usr/lib/libboost_filesystem.so.1.49.0 (0x76134000)
libboost_thread.so.1.49.0 => /usr/lib/libboost_thread.so.1.49.0 (0x76112000)
libpthread.so.0 => /lib/arm-linux-gnueabi/libpthread.so.0 (0x760f3000)
libboost_date_time.so.1.49.0 => /usr/lib/libboost_date_time.so.1.49.0 (0x760dd000)
libboost_chrono.so.1.49.0 => /usr/lib/libboost_chrono.so.1.49.0 (0x760cf000)
```

---

```

libboost_regex.so.1.49.0 => /usr/lib/libboost_regex.so.1.49.0 (0x75fea000)
libssl.so.1.0.0 => /usr/lib/arm-linux-gnueabi/libssl.so.1.0.0 (0x75f99000)
libcrypto.so.1.0.0 => /usr/lib/arm-linux-gnueabi/libcrypto.so.1.0.0 (0x75e35000)
libstdc++.so.6 => /usr/lib/arm-linux-gnueabi/libstdc++.so.6 (0x75d63000)
libm.so.6 => /lib/arm-linux-gnueabi/libm.so.6 (0x75cf1000)
libgcc_s.so.1 => /lib/arm-linux-gnueabi/libgcc_s.so.1 (0x75cc9000)
libc.so.6 => /lib/arm-linux-gnueabi/libc.so.6 (0x75b99000)
/lib/ld-linux-armhf.so.3 (0x76f5e000)
librt.so.1 => /lib/arm-linux-gnueabi/librt.so.1 (0x75b8a000)
libcuc.so.48 => /usr/lib/arm-linux-gnueabi/libcuc.so.48 (0x75a5a000)
libcui18n.so.48 => /usr/lib/arm-linux-gnueabi/libcui18n.so.48 (0x758e8000)
libcudata.so.48 => /usr/lib/arm-linux-gnueabi/libcudata.so.48 (0x74770000)
libz.so.1 => /lib/arm-linux-gnueabi/libz.so.1 (0x74752000)

```

Werden Libraries nicht gefunden so müssen diese noch installiert werden. Verwenden Sie hierzu das Kommando apt-get.

## 4.2.1 Windows

- **Wie findet man heraus ob der Port des OPC UA Server geöffnet ist?**

Mit dem netstat Kommando kann man prüfen, ob der Port 8888 geöffnet ist. Führen Sie hierzu das netstat Kommando auf der Kommandozeile als Administrator wie folgt aus.

```
> netstat -na
```

Der Port ist geöffnet wenn in der Ausgabe des netstat Kommando die folgende Zeile erscheint.

```
TCP  0.0.0.0:8888      0.0.0.0:0        ABHÖREN
```

- **Wie findet man heraus ob der OPC UA Server gestartet ist?**

Starten Sie den Windows TaskManager und wählen Sie den Tab Dienste. Der Dienst ASNeG-Demo muss sich im Zustand "wird ausgeführt" befinden.

- **Wie startet und stoppt man den OPC UA Server?**

Der OPC UA Server wird nach der Installation automatisch als Dienst gestartet. Verwenden Sie den Windows Dienste Manager zum Starten und Stoppen des OPC UA Server.

- **Kann man den OPC UA Server auch auf der Kommandozeile starten?**  
NBC
- **Wurden alle abhängige Libraries installiert?**  
NBC