



Einleitung

OPC UA

OPC UA steht für OPC Unified Architecture, kurz OPC UA. Im Vergleich zum Vorgänger OPC unterscheidet sich OPC UA besonders durch die Fähigkeit, Maschinendaten (Messwerte, Parameter etc.) nicht nur zu transportieren, sondern auch maschinenlesbar semantisch zu beschreiben. OPC UA bedeutet: **O**peness **P**roductivity **C**onnectivity **U**nified **A**rchitecture (Offenheit, Produktivität, Konnektivität, einheitliche Architektur).

Node

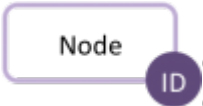
Der „Node“ ist das grundlegendste Element der OPC UA. Nahezu jedes Element ist quasi auf einen „Node“ „reduziert“. Die Nodes stehen dabei in entsprechender Relation zueinander.

Die Definition aus Wikipedia über die OPC Unified Architecture hat für den Begriff „Node“ eine treffende Beschreibung:

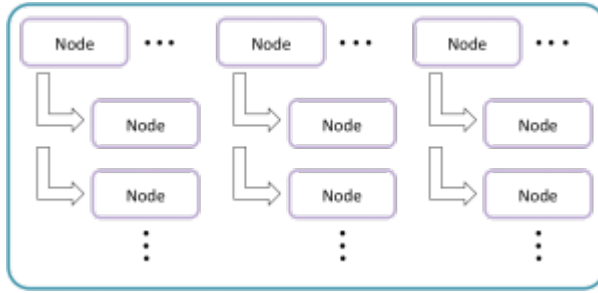
„Das OPC-Informationsmodell ist nicht mehr nur eine Hierarchie aus Ordnern, Items und Properties. Es ist ein sogenanntes Full-Mesh-Network aus Nodes, mit dem neben den Nutzdaten eines Nodes auch Meta- und Diagnoseinformationen repräsentiert werden.“ [Quelle: wikipedia.org/wiki/OPC_Unified_Architecture](https://wikipedia.org/wiki/OPC_Unified_Architecture)

- Ein Node ähnelt einem Objekt aus der objektorientierten Programmierung.
- Ein Node besitzt Attribute, die gelesen werden können (Data Access (DA), Historical Data Access (HDA)).
- Die Nodes werden sowohl für die Nutzdaten, als auch für alle anderen Arten von Metadaten verwendet.
- Der damit modellierte OPC Adressraum beinhaltet ein Typenmodell, mit dem sämtliche Datentypen spezifiziert werden.

NodId

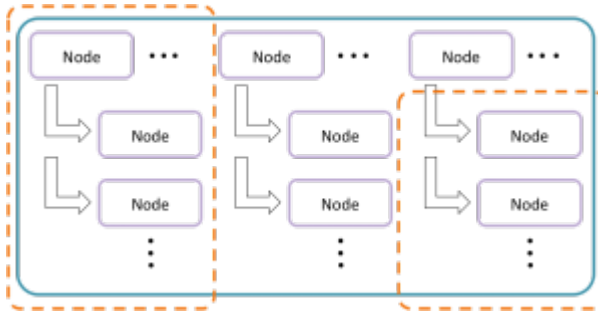
 Die OPC-Spezifikation definiert, dass jeder Node über einen Identifier (= **NodId**) eindeutig im Adressraum (engl. Address Space) identifiziert werden kann. Der **NodId** wird entweder durch eine GUID (Global Unique Identifier), einen numerischen Ausdruck, ein Array von Bytes oder einen String-Wert definiert. Der NodId enthält in der Regel den „**Namespace**“. Das muss aber nicht zwangsläufig der Fall sein.

Address Space



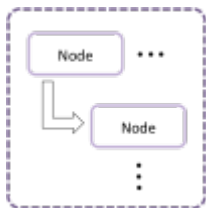
Sämtliche **Nodes**, die in OPC UA bereitgestellt und verarbeitet werden, werden im Rahmen eines sogenannten **Address Spaces** verwaltet. Der **Address Space** stellt dabei eine Art logischen Speicher dar. Innerhalb dieses „Speichers“ können die in ihm enthaltenen **Nodes** auf einen oder mehrere **Nodes** im selben oder in einem anderen **Address Space** (logisch) verweisen.

View



Der oben genannte / visualisierte „Address Space“ kann logisch in eine oder mehrere Views unterteilt werden. Während es eine **Standardansicht** gibt, können die **benutzerdefinierten Ansichten** einen oder mehrere Nodes enthalten.

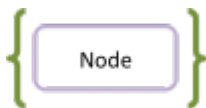
NodeManager



Der **Node Manager** stellt ein oder mehrere Nodes bereit und definiert ihre Beziehungen zueinander. Vordefinierte **System Node Manager** sind:

- Core Node Manager (definiert u.a. Type Nodes und System Nodes)
- Diagnostics Node Manager (stellt Nodes für Diagnosezwecke bereit)
- Master Node Manager (der „Verwalter“ aller Node Manager, er delegiert Aufrufe zu dem betreffenden Node Manager)

Service



OPC UA definiert eine Reihe verschiedener **Services**, über die der Client mit dem Server interagiert. Diese **Services** sind dabei serverseitig als Methoden implementiert und werden verwendet zum:

- Lesen und Schreiben von Node Attributen bzw. Werten
- Verwalten von Node Referenzen
- Browsen von Nodes
- Lesen und Schreiben historischer Werte
- Aufrufen von Methoden
- Verwalten von Subscriptions
- etc.

Inhaltsverzeichnis

OPC UA	1
Node	1
NodeId	1
Address Space	1
View	2
NodeManager	2
Service	2

