**OPC UA PROTOCOL**

What is OPC UA?

OPC Unified Architecture (OPC UA) is a machine-to-machine communication protocol used for industrial automation and developed by the OPC Foundation. The OPC UA platform in an platform-independent service-oriented architecture that integrates individual OPC Classic specifications into an extensible framework. OPC UA simplifies industrial connectivity so you can integrate all your devices, automation systems and software applications using a secure and platform-independent standard.

Why use OPC UA?

Using OPC UA you can easily connect your remote devices to Cumulocity IoT either in the cloud or at the edge, enabling device diagnostics, asset management, monitoring, reporting and other applications.

With Cumulocity IoT OPC UA integration, you can:

* Integrate any OPC-UA device into the platform without coding
* Plug in any device and the platform will automatically accommodate the capabilities and the structure of the data it provides
* Scale your infrastructure using multiple gateways

No VPN is required. The gateway establishes a secure, encrypted outgoing connection to our cloud platform, and you can use OPC UA at the edge as well as in the cloud.

By way of example, one company uses OPC UA integration for industrial connectivity—communication with different ovens in a baking process. Bakers get an alarm if they leave the bread in the oven too long and it is at risk of overcooking. The alarm moves the process from manual to automated, substantially reducing loss and waste.

Benefits of OPC UA

Self service:

As no coding is required, operational teams can easily connect to any OPC UA server. And they can use device auto-discovery to fully automate the detection and integration of devices with Cumulocity IoT.

Support for generic device types:

You only need to define each device once. The device type definition defines the mapping from a machine to the corresponding IoT asset and the configuration—for example, which data to synchronize at which frequency.

Cloud-based configuration:

We provide zero-touch installation of the OPC UA IoT gateway by a field engineer, through cloud-based remote configuration. So, no matter where engineers are, they can connect to different devices, without needing to be in the same room.

### OPC UA Transport Layers

Many of the common Industrial Automation (IA) protocol technologies limit the available transports. Devices that want to communicate with Programmable Controllers must use the transport that is defined for the communication technology supported by that brand of Programmable controller (PLC). However, the transports of OPC UA are not limited like other protocols. The technology space where OPC UA operates is much more extensive and requires support for many different transports and the capability to add new transports in the future. Transports are about how an OPC UA message is moved from your node to some other node on the network.

Once an OPC UA application forms a UA message or a response, it must send it somewhere. Transports are the low-level mechanisms for moving those serialized messages from one place to another.

OPC UA operates in a very broad technology space and the devices can support multiple transports or even custom or proprietary transports. OPC UA devices can be anything from a factory floor sensor or actuator to a Programmable Controller, a Human Interface Device, a Windows Server operating a massive Oracle database, or an undersea pipeline controller. A rich set of support transports are required to support the OPC UA mission of being a completely scalable solution.

The OPC UA specification defines several transports that Clients must support:

* SOAP / HTTP TRANSPORTS – HTTP (HyperText Transfer Protocol) is the connectionless, stateless, request-response protocol that you use every time you access a web page. SOAP (Simple Object Access Protocol) is an XML messaging protocol that provides a mechanism for applications to encode messages to other applications.
* HTTPS TRANSPORT – HyperText Transfer Protocol Secure (HTTPS) is the secure version of HTTP. It means that all communications between your browser and that website are encrypted. Just as with HTTP, SOAP is used as the request-response protocol to move the OPC UA requests between Clients and Servers.
* UA TCP TRANSPORT – UA TCP is a simple TCP-based protocol designed for Server devices lacking the resources to implement XML encoding and HTTP/SOAP type transports. UA TCP uses binary encoding and a simple message structure that can be implemented on low-end Servers.

What makes OPC UA transports so powerful is that OPC UA requests to Read or Write an Attribute can use standard Web Services technologies. UA requests and responses can be encoded as XML, placed inside a SOAP request, and transferred to an IT application that already knows how to handle the HTTP, the SOAP message, and the XML. OPC UA has an essentially “free” mechanism for sending messages between the factory floor and IT devices.

Data Model for OPC UA

It all begins with a single thing. A process, a system, or an entire plant might be as basic as a single piece of data or as complex as a process, a system, or an entire plant.

It might be a mix of data values, meta-data, and connections. Take a twin loop controller, for example. The setpoints and actual values for each loop would be linked by the dual loop controller object. The temperature units, high and low setpoints, and text descriptions would all be referenced by those variables.

Subscriptions to receive alerts of changes to the data values or meta-data for that data value may also be made accessible by the object. A client can acquire as little data as it wishes (a single data value) or a highly extensive set of data that defines the controller and its activity in great detail.

OPC UA is made up of a client and a server, just like its factory-floor siblings. The client device makes a request for data. It is provided by the server device. However, as the loop controller example demonstrates, the OPC UA server is far more advanced than an EtherNet/IP, Modbus TCP, or PROFINET IO server.

An OPC UA server represents data, information, processes, and systems as objects and exposes them to clients in a fashion that is helpful to a wide range of client applications. Even better, the OPC UA server offers complex services to the client, such as:

* OPC UA Services Discovery Services: Clients are able to find out what things are accessible, how they are related to other objects, what sort of data and what type of data is available, and what meta-data is available to organize, categorize, and characterize those items and values
* Subscription Services: Clients can utilize subscription services to figure out what type of data is accessible for alerts. Clients can use these services to choose how much, when, and how little they want to be alerted about changes to data values, as well as meta-data and object structure.

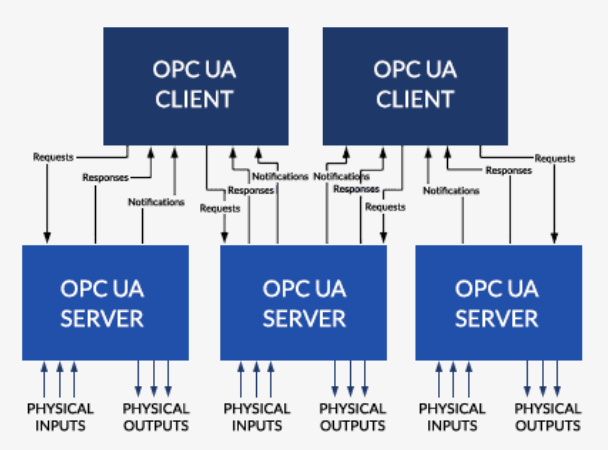
Query Services:

Services that provide a client with bulk data, such as historical data for a data value

* Node Services: Clients can use these services to create, delete, and modify the structure of the server's data.
* Method Services: Clients can use these services to make object-related function calls.

The OPC UA server is a data engine that collects data and presents it in ways that are useful to various types of OPC UA client devices, such as HMIs on the factory floor, proprietary control programs like recipe managers, or databases, dashboards, and sophisticated analytics programs on enterprise servers.

Data isn't always bound to a single physical node. Objects can refer to other objects, data variables, data types, and more in nodes located elsewhere on the subnet, in the architecture, or even on the Internet.



#### OPC UA Server

All industrial Servers provide the physical interface to the real world. Servers measure physical properties, indicate status, initiate physical actions and do all sorts of physical measurements and activations in the real world under the direction of a remote Client device. Servers are where the physical world meets the digital world.

The specific capabilities of an OPC UA Server are described by the Profile it supports. A Profile indicates to other devices (electronically) and to people (human-readable form) what specific features of the OPC UA specification are supported. Engineers can determine from the Profile if this device is suitable for an application. A Client device can interrogate the Server and determine if it is compatible with the Client and its application and if it should initiate the connection process with the device.

An OPC UA Server announces its availability to interested Client devices, it provides a list of its capabilities and functionality to interested Clients, it provides notifications of different kinds of events, it executes small pieces of logic called methods, it provides address space information in bulk to Clients (Query service), it provides browsing services so that a Client can walk through its address space, and it can allow Clients to modify the node structure of its address space.

An OPC UA Server models data, information, processes, and systems as Objects and presents those Objects to Clients in ways that are useful to vastly different types of Client applications. Better yet, the UA Server provides sophisticated services that the Client can use including:

* Discovery Services – services that Clients can use to know what Objects are available, how they are linked to other Objects, what kind of data and what type is available, what meta-data is available that can be used to organize, classify and describe those Objects and Values
* Subscription services – services that the Clients can use to identify what kind of data is available for notifications. Services that Clients can use to decide how little, how much and when they wish to be notified about changes, not only to data values but to the meta-data and structure of Objects
* Query Services – services that deliver bulk data to a Client like historical data for a data value
* Node Services – services that Clients can use to create, delete and modify the structure of the data maintained by the Server
* Method Services – services that the Clients can use to make function calls associated with Objects

Unlike the standard industrial protocols, an OPC UA Server is a data engine that gathers information and presents it in ways that are useful to various types of OPC UA Client devices. Devices that could be located on the factory floor like an HMI, a proprietary control program like a recipe manager or a database, dashboard or sophisticated analytics program that might be located on an Enterprise Server.

#### 

#### OPC UA Client

In most industrial networking technologies, there is a controlling device: a device that connects to and controls one or more end devices. In OPC UA, a device of this type is known as an OPC UA Client. Like controlling devices in these other technologies, an OPC UA Client device sends message packets to Server devices and receives responses from its Server devices. But beyond this basic functionality, an OPC UA Client device is fundamentally more sophisticated than controllers in other technologies.

There are eight concepts that are important to remember when thinking about OPC UA Clients:

1. Client devices request services from OPC UA Server devices. Server devices send response messages and notifications to the OPC UA Client device.
2. The Subscription Service Set, which drives notifications, and the Read Service of the Attribute Service Set are the primary services that OPC UA Clients use to interact with the address space on an OPC UA Server.
3. Clients find OPC UA Server devices in multiple ways. Clients can find Servers using traditional configuration, by using a Local Discovery Server, by using a Local Discovery Server with a Multicast Extension, or by using a Global Discovery Server.
4. Once a Client finds a Server, it obtains the list of available endpoints and selects an endpoint that supports the security profile and transport that matches its application requirements.
5. Clients begin the process of accessing an OPC UA Server by creating a channel, a long term, a connection between it and an OPC UA Server. Channels are the authenticated connections between two devices.
6. Once the channel is established, Clients create sessions, long term, logical connections between OPC UA applications. A session is the authorized connection between the Client’s application and the Server’s address space.
7. Clients can subscribe to data value changes, alarm conditions, and any results from programs executed by Servers. Servers publish notifications back to the Client when those items are triggered.
8. Clients invoke methods, which are small program segments. Programs can return results to the Client in the Method call or in a Notification if the Client subscribes to it.