Still a Thrill: OPC UA Device Discovery

By John Rinaldi

Published: 2016-04-08 · Archived: 2026-04-05 18:23:53 UTC

A long while back, I when I started working on industrial connectivity, I had hair. It's now gone. Completely gone. I've lost the hair but I've gained 30 years of experience. And over all those years in the automation industry, the Discovery mechanism in OPC UA is one of the most fascinating things I've seen.

OPC UA is a very flexible and sophisticated mechanism for Client devices to find and identify Server devices. The traditional mechanism for matching a Client (also called an Initiator or Master device) to a Server (or a Target or Slave device) is for the user to identify the Server to the Client during some sort of configuration process. These mechanisms are effective for the kinds of factory floor applications we've used in the past. In fact, for low-level sensor/actuator networks, they would still be recommended.

But OPC UA is designed to function equally well on the factory floor and the Enterprise, so what's really interesting is how OPC UA created a more robust and flexible mechanism that works equally well on the factory floor with embedded devices as it does in the Enterprise with IT devices.

OPC UA designed Device Discovery using the Service Oriented Architecture (SOA) model. What makes SOA so interesting, and the backbone of a lot of Enterprise and Internet applications, is that there is a standard mechanism for Clients to discover Server devices, interrogate them to see what services they offer, and connect to them. OPC UA extends the SOA model to define a process in which Clients can find Server devices that choose to reveal themselves, interrogate them to determine the various ways that the Client can interact with them, and determine what capabilities they have that might be of interest.

To understand OPC UA Device Discovery there is a term that you may have heard but not fully understood. That term is endpoint. An endpoint is a connection to a device that offers some specific functionality, that is sometimes only available through that specific connection. For example, you can think of the programming port on a

Programmable Controller as an endpoint. It uses a specific physical layer with a specific transport to accomplish downloading a program. That programming port is an endpoint, and it's dedicated to programming, it will differ from every other communication port (endpoint) on the programmable controller.

Typical Ethernet devices in Industrial Automation have a single endpoint. ProfiNet IO devices have an endpoint that supports ProfiNet IO connections with cyclic and acyclic message transfer. EtherNet/IP devices have an endpoint that supports CIP (Common Industrial Protocol) messaging. Modbus TCP devices have an endpoint that supports Modbus messages over Ethernet.

The only real cases where devices we are familiar with in Industrial Automation have multiple endpoints are those devices that support multiple protocols. If a device supports EtherNet/IP and Modbus TCP, then there will actually be two: http://192.168.0.100:502/ for Modbus TCP and http://192.168.0.100:44818/ for EtherNet/IP. That hasn't been important to us in the past because a Modbus Client or an EtherNet/IP Scanner device (Client/Initiator) "knows" to send messages to a Modbus Server or an Adapter device (Slave/Initiator) using those endpoints.

Game-changing Endpoints

But now, with OPC UA, endpoints are not only much more important, they are much more sophisticated.

OPC UA devices can have any number of endpoints. Some may have only one or two. Others might have as many as five or ten.

Some endpoints will use HTTP, HTTPS, or UA Secure Channel as the transport.

Some endpoints will require signed messages, others will require signed and encrypted messages, and still others may not use any security at all.

Some endpoints will exist for particular purposes, such as the Discovery endpoint, which Clients use to get the endpoint information. Those endpoints will usually not provide the functionality that you might find on other endpoints.

To understand OPC UA Device Discovery, it is important to understand the basic steps that are used in the Device Discovery process where an OPC UA Client device finds, interrogates, and connects to OPC UA Server devices:

1. A Server powers on, and if it chooses to provide its own Discovery service support, it opens its Discovery port for messages from Client devices that want endpoint information.

2. If a Server chooses to let another Server provide Discovery services, it registers itself with Local Discovery Servers (LDS) or multicast Discovery Servers (LDS-ME). The Discovery servers may be resident on the same platform as the Server, or on a platform someplace else on the network. Note that some Servers may choose to remain private and only be available to Clients that are configured to know about them.

3. A Client that finds a Server in an LDS can retrieve the application description for the Server and information on accessing the Server's Discovery endpoint where it can get more detailed information on the Server.

4. If a Client is interested in connecting to a Server, it uses the Server's Discovery endpoint to get the list of endpoints supported. The information on each endpoint includes the transport and security it supports.

5. If the Client finds an endpoint that meets its application requirements, security requirements, and transport requirements, it begins the connection process with the Server on that endpoint.

That's a very unique and sophisticated way of connecting two devices together. I find it fascinating. It might be more fascinating if I still had hair but I've given up on that!

John Rinaldi is a renowned automation strategist and the CEO of Real Time Automation (RTA). A leading voice in industrial networking, John has authored six books on essential protocols like EtherNet/IP, OPC UA and Industrial Ethernet, along with 500+ technical articles. He is a sought-after speaker and consultant, known for his ability to simplify complex data communication challenges and drive innovation in global industrial applications.
