



Part 2 of 8:

Origin, Development and Objectives of OPC UA

OPC interfaces today provide easy to use, high-performance connectivity between automation components, control hardware and field devices, thus bridging the divide between heterogeneous automation worlds. In the second part of our series of articles on OPC UA, we will now take a look at the history of OPC, which also gives an understanding of the importance that the new OPC Unified Architecture standard has for automation.

OPC is unquestionably one of the most successful de-facto standards since the invention of the computer. Today, a user can choose from over 20,000 OPC products offered by more than 3,500 vendors. Millions of installed OPC based products are used in production, processes, building automation, and many other industries around the world. This renders the OPC technology as the undisputed standard for interoperable data exchange between software applications from different vendors. OPC allows automating data transfer between

widely distributed installations. OPC interfaces provide easy to use, high-performance connectivity between automation components, control hardware and field devices, thus bridging the divide between heterogeneous automation worlds. The OPC technology is used today for practically all types of data acquisition, for vertical and horizontal data integration, and data management. OPC is the essential link for HMI/SCADA systems for process visualization, for DCS systems or PLCs for process control, and MES and ERP systems to access the underlying automation components. In the early days of the OPC technology only process data or individual parameters were transferred over the OPC interface. Today, OPC is used to transport entire ERP documents, parameter sets, control sequences, or to drive control applications.

Never touch a running system – why a new OPC?

OPC is proven, well-established, and globally successful technology. What motivates the OPC Foundation to introduce the OPC Unified Architecture? Is OPC UA a “new OPC”? Will OPC UA supersede Classic OPC? What advantages does the Unified Architecture offer compared to Classic OPC? Considerations to build a new OPC architecture were embraced as early as 2003, as the Alarms&Events working group of the OPC Foundation was developing the next generation of the AE specification and its migration to Web Services. These considerations led to the formation of a completely new working group in late 2003. The primary goal of this working group initially was to convert the access to process data (Data Access), alarms and events (Alarms&Events) and historical data (Historical Data Access) to Web Services, and standardize the way they are accessed. The Unified Architecture – in short, OPC UA – was born. For over five years employees from 30 companies – several of them market leaders in their respective industries – worked to develop the new OPC architecture under the auspices of the OPC Foundation. Besides converting Classic OPC to Web Services and unifying DA, AE and HDA, many additional new requirements were also placed on the new OPC UA. They were added by the OPC Foundation after conducting market surveys and consulting a vast number of OPC users, system integrators and vendors. Based on the findings, the OPC Foundation defined the following guidelines and key objectives:

- Keep it simple: To make UA technology easy to use through UA components, despite the multitude of functional requirements and complexity
- Evolution instead of revolution: To maintain the terminology, object models and essential communication principles of Classic OPC; to protect investments in the development of Classic OPC products by ensuring their continued usability
- Platform independence and scalability: To supersede DCOM as the technology base with a service oriented architecture (SOA) to allow using OPC technology at the IT level and in embedded systems
- Access protection: Protection against spying, sabotage, attacks and faults caused by negligent behavior
- Data security: Robust architecture, reliable communication mechanisms, redundancy concepts and other precautions against data loss
- Strong performance: Lean, high-performance data transport to achieve highest performance requirements

OPC UA was not just planned as a new version of the OPC interface standard. OPC UA is the vision of “global” interoperability and a standardized data exchange between software applications independently of vendor, programming language, operating system or location.

New possibilities with OPC UA

OPC UA complements the existing OPC industrial standard by adding essential new properties such as platform independence, scalability, high availability, Internet capability, and many more. Platform independence and scalability, in particular, open up many possibilities for completely new and cost-saving automation concepts. Embedded field devices, process control systems, PLCs, gateways or operator panels can comprise lean OPC UA server implementations that have been ported directly to operating systems including embedded Linux, VxWorks, QNX, RTOS, and many more. This eliminates the need for a separate Windows PC for the OPC server, which previously provided access to the data on devices running non-Windows platforms. OPC UA components can also be used in IT systems, enterprise resource planning systems (WWS/ERP), production planning and control software and other e-business applications using Unix operating systems such as Solaris, HP-UX, AIX, etc. This broad applicability of OPC UA technology allows implementing whole new vertical integration concepts. By cascading OPC UA components, information can be transported securely and reliably from the factory floor all the way through to the production planning or ERP system. For this purpose, client and server enabled UA components at the automation level connect embedded UA servers at the field level with integrated UA clients in ERP systems at the enterprise level. The individual UA components can be geographically distributed and separated from each other by firewalls without problems. OPC UA does not replace Classic OPC. DCOM based OPC products and UA products can coexist. The migration strategy pursued by the OPC Foundation allows combining Classic OPC and OPC UA products. This way, several thousands of Classic OPC products installed around the world can be used together with the new OPC UA products right from the start. This offers many advantages for users as they can mix and match products from different vendors for communication not only on the intranet – as before – but also via the Internet, and will be able to scale them for different platforms in future. OPC UA offers standardization organizations the possibility to use UA Services as transports for their information models. To achieve this, the OPC Foundation is cooperating with various standards bodies, including PLCopen, ISA (Instrumentation, Systems and Automation Society), MIMOSA (Machinery Information Management Open Systems Alliance), ECT (EDDL Cooperation Team) and others to define companion standards (which will be the focus of Part 5 of the OPC UA Series). Within these cooperations, the OPC Foundation specifies “how” the data is transported, while the cooperating organizations define “what” data and information is to be transported according to the standardized information models.

In the next issue: “Ten Reasons for OPC UA”

www.softing.com

Author: Jürgen Lange, Area Account Manager Embedded Technology Products, Softing Industrial Automation GmbH

OPC Day Europe 2012

The OPC Foundation will be holding this year's OPC Day Europe on May 16, 2012. The event will take place at Endress+Hauser in Reinach, Switzerland, and focus on the use of OPC UA in process automation.

Series of Articles in SPS-MAGAZIN

Issue	Date of Publication	Topic
3	24.02.2012	OPC UA Status
4	16.03.2012	OPC UA: Origin, Development and Objectives
HMI Special	13.04.2012	Ten Reasons for OPC UA
5	10.05.2012	OPC UA Specifications
6	01.06.2012	OPC UA Companion Standards
7	30.06.2012	OPC UA Compliance Test
8	27.07.2012	OPC UA Toolkits
9	31.08.2012	OPC UA Outlook