AUTOMATION SYSTEM ARCHITECTURE USING OPEN INDUSTRIAL STANDARDS

WILLIAM D. FERRAZ, RODRIGO P. PANTONI

Smar Equipamentos Industriais Ltda.
Av Antonio Furlan Jr. 1622 CEP 14160500 Sertaozinho - SP
E-mails: wilferraz.hm@hotmail.com , palucci@smar.com.br

Abstract— In modern industries the control systems are inserted inside a broader context other then the control problem domain. This broader context is generally referred to as Automation System, which also addresses peripheral issues like control data management and associated communication protocols, maintenance of the control system components (e.g. sensors and final control elements) known as 'Asset Management', management of data exchange among subsystems, interoperability among devices and among systems, etc. This article shows that most of the previous mentioned issues are addressed using not only electronics and network standards as well as open industrial software standards in the host level (control room workstations) established upon a common open architecture for automation systems based on SOA (Service Oriented Architecture).

Keywords— Industrial Standards, SOA, Network of things, Automation System Architecture

1- Introduction

In modern industries the control systems are inserted inside a broader context other then the control problem domain. This broader context is generally referred to as Automation System, which also addresses peripheral issues like control data management and associated communication protocols, maintenance of the control system components (e.g. sensors and final control elements) known as 'Asset Management', management of data exchange among subsystems, interoperability among devices and among systems, etc. This article shows that most of the previous mentioned issues are addressed using not only electronics and network standards as well as open industrial software standards in the host level (control room workstations). Thus, the automation systems evolved in the last decade from stand-alone specialized workstations using vendor-specific (proprietary) standards for hardware, software and communication protocols to full-fledged hardware and software architecture supporting high-speed data exchange among all sorts of entities, from biometrics and Internet-aware components to valve positioners (Peluso and Wallace, 2003). This augmentation in the amount of digital-communication enabled devices and the associated data processing generated by the interaction of these devices requires not only the state-of-the-art on software and hardware development but it requires the knowledge of the state-of-the-art technology to design the system architecture to handle the communication involved in such architecture. Dominance over such technologies that can properly deal with the huge amount of information available in the nowadays automation systems network is necessary to transform raw information in usable knowledge (Hayes and Alberts, 1996) and also to control the information exchange (Duchastel, 2001) between the network components.

In order to guarantee the accomplishment of common factory-floor production and engineering requirements like quality and reliability, stability and safety issues among others, it is necessary to have data and information flowing up and down through the several different layers that compose an enterprise structure, and thus through distinct information-domains e.g. Neural-Network and PID control (electronics engineering domain) affecting production efficiency and planning (Production Management domain). The automation systems community soon realized that it is not effective, and in some cases even not acceptable, to try to siege technology as well as knowledge development and usage by using proprietary solutions or proprietary artifacts. This context conducted the development of non-proprietary solutions. To harmonize different vendors' solutions (non-proprietary) for distinct information domains accomplishing the engineering requirements, it is common sense that the enterprise automation system might rely upon a common architecture (Emerson, 2005) (Gelle et al, 2003) to enable open standards interoperability.

In order to mitigate what have been exposed above, international non-profit organizations play a fundamental role. Some of them, like OPC Foundation (OPC – OLE for Process Control), Fieldbus Foundation™ (FF) and IEC (International Electrotechnical Commission) cooperate among themselves to enable the development of open and consistent automation system architectures. Examples of such efforts are the FF HSE (High Speed Ethernet) specifications, IEC 61804-2 (EDDL - Electronic Device Description Language) and OPC UA (Unified Architecture) specifications (Fieldbus Foundation, 1999a and 199b) (OPC Foundation, 2006).

As it's demonstrated in the next paragraphs, an automation system that combine the implementation of such technologies is able to succeed in the hard task of observe, classify and use information to cope with the enterprise ROI (Return Of Investment) and TCO (Total Cost Of Ownership).

2- FF HSE – Open Specifications to Implement a Common Network Architecture for Interoperability

Obviously, the Enterprise production infrastructure is the basic source of the Automation System's raw data. In the modern plants, however, the data generated varies from simple measurements like a pressure value to complex diagnostics information like a valve signature[1] or even measurement subsystems. Analog technologies are not suitable to transmit this amount of data, so the technology evolved towards digital-processor field devices in the last two decades.

However, technologies and standards appear and disappear all the time. Often, those who buy into the '*technology de jour*'[2] have later been disappointed and incurred in unnecessary expense to replace a 'promising' but unsupported technology.

Therefore, when discussing digital field automation technology architectures, users state their concerns a hundred different ways, but in the end, what users (Business and Process) seek are assurances the technology platform they choose provides (Zielinski, 2004):

- Freedom of choice in plant floor instrumentation and equipment independent of the Host - valves, transmitters, motor starters, remote IO, etc.;
- Consistency in how plant floor instrumentation and equipment is engineered;
- Flexibility and efficiency in how plant-floor data are shared throughout the enterprise;
- Ease of maintenance;
- Quantified proof that adopting manufacturers have long-term commitments to expand and improve the underlying technology;
- Easy access to production data;
- Easy connectivity throughout the business endpoint;
- Enough resources for data mining.

Among several solutions, the non-profit organization Fieldbus FOUNDATION™ presented an approach that meets the requirements of the modern enterprise's infrastructure network mentioned above. As shown in the figure 1, the FF HSE system architecture provides a framework for describing the automation system as a collection of physical devices interconnected by a fieldbus network (Fieldbus Foundation, 1999a).

The FF's architecture is comprised of a consistent hardware and software architecture envisaging an optimum integration with the end-user, through a layer called 'User-Layer', which, in the extent given by FF, has no counterpart in other open specification such as MODBUS.

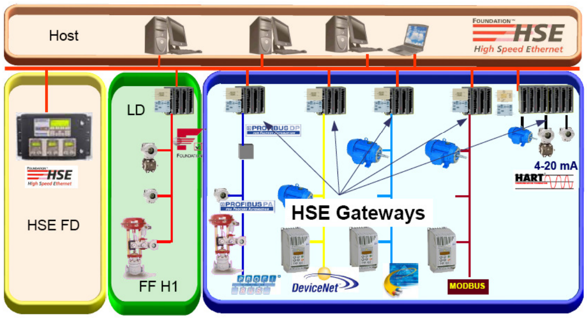

Figure 1: FF HSE Architecture

The FF systems' architecture is composed of two parts. The (low-speed) factory-floor part, named 'H1' and the systems' backbone communication network named 'HSE', taking the acronym from the Ethernet's definition *H*igh *S*peed *E*thernet, since the communication relies upon the well known and accepted Ethernet standard.

3- EDDL – The Experts Talk

FF HSE User layer uses the Electronic Device Description Language (EDDL) to describe the devices throughout the FF-HSE-based system. EDDL is a text-based language for describing the digital communication characteristics of intelligent field instrumentation and equipment parameters—device status, diagnostic data, and configuration details—in an operating system and Human Machine Interface (HMI) neutral environment.

Everyone is aware of the difficulties of getting device software drivers to work after a host platform OS upgrade. EDDL was specifically designed for the EDD (Electronic Device Description) file to avoid this issue. The IEC 61804-2 standard is not only OS (Operational System) and HMI independent, but also fieldbus communication protocol neutral.

Today EDDL technologies establishes the engineering and operating base on which all major digital fieldbus protocols—FOUNDATION Fieldbus™, HART®, and PROFIBUS—construct parametric and device descriptions. And, because EDDL is an open technology with international standard status, it can be easily and effectively applied to any device and any fieldbus protocol (Emerson , 2005).

From an end-user perspective, it's equally reassuring to know that a host workstation change from any OS to any other without require modification of the existing EDD files. It is possible to mention two typical examples like: migration of the host platform (OS) from Windows NT to Windows 2003 Server or from Windows to Linux.

Allying the FF HSE architecture and EDDL capability is possible to integrate not only the communication systems between networks but it's possible to integrate the device manufacturer know-how to the automation infrastructure. However, the automa-

[1] Valve signature plots actuator pressure versus travel (actuator stem position variation over time)

[2] French expression. Technology-of-the-day, not proven-in-use technology

tion system vendor is not limited to the device manufacturer know-how. It can also join its own knowledge to it.

In February 2003, representatives of, FF, HCF (HART Communication Foundation) and PNO (PROFIBUS NutzerOrganisation) met in a collaborative project to extend the capabilities of the IEC 618342 standard.

The scope of the co-operative project is to add robust organization and graphical visualization of device data as well as support for persistent data storage (i.e., permanent data storage) features to IEC 61804-2. Sophisticated devices. As it is protocol and technology independent (different from other technologies with the same purpose like FDT - Field Device Tool (Merrit, 2002)), it is even more efficient when using FF-HSE, with its user-layer, to empower data and information transfer.

4- OPC AND OPC-UA

The rising of OPC technology made the HMI software manufacturers have to develop only a driver for communication with devices, differently of the way before, which each manufacturer had to develop proprietary drivers for supporting its devices.

A common analogy for OPC DA (Data Access) is printer drivers in DOS versus Windows. In DOS, the application developer had to write a printer driver for every printer. So AutoCAD, WordPerfect, and Netscape all had to write a separate printer driver for every printer they wanted to support. In industrial automation, companies like Rockwell Automation, ICONICS and Citect wrote their own HMI software and a proprietary driver for each industrial device including every PLC (Programmable Logic Controller) brand in order to retrieve process data.

The standardization of these drivers was developed by a group of manufactures that have improved the OPC model due to them additions based on past experiences. It is mentioned for example the navigation system of the OPC tags through tree structure, and its division by sectors.

Since 2003, OPC Foundation and a group of IHM and device manufactures have been working in a joint venture to define and implement a new OPC Specification. This new Specification is the OPC UA.

The purpose of OPC UA is to provide enhancements for existing and next generation OPC products in the areas of security, reliability, and interoperability. OPC UA is designed to unify existing OPC specifications such as DA, DX (Data Exchange), HAD (Historical Data Access), and XML (eXtensible Markup Language) DA into an environment that will leverage Web-based technologies and standards such as Web Services, WSDL (Web Services Description Language), XML and SOAP (Simple Object Access Protocol).

The OPC UA is based on SOA (Service Oriented Architecture) through Web Service that transports XML data, which provides the communication among different software of different platforms, providing interoperability (state of the art in IT) (OPC Foundation, 2006).

The Extensible Markup Language (XML) is a non-proprietary specification for document interchange in the Internet that the World Wide Web Consortium (W3C) developed in 1998 (Roy and Ramanujan, 2000). XML lets interoperability of different software of different platforms to transfer information using a common language. Since XML is a meta-language, it is possible to create new languages to make a standard talk. In Mathematics for example, a XML based language called MathML is in use in the same way OPC uses SOAP (Gelle et al, 2003).

The OPC Foundation has joined the international cooperative team of the three leading fieldbus organizations, the Fieldbus Foundation™ (FF), HART Communication Foundation (HCF), and PROFIBUS NutzerOrganisation e.V. (PNO), to extend the reach of electronic device descriptions (EDDs) into the OPC unified architecture. This work provides the total integration between OPC UA and EDDL to describe data to the host reads, i.e., OPC Clients have access to complex device data with automatic integration. This lets an easier and cheaper development, which brings a better quality for the product to the final user (Fieldbus Foundation, 2006).

Thus, OPC UA is the key to improve the transformation of simple data in knowledge, due to the open high-level mechanism, open the doors to store the information by an easer way for MES (Manufacturing Execution System) and ERP (Enterprise Resource Planning) applications.

5- COMMON OPEN ARCHITECTURE AND XML

Although the above-mentioned technologies can enable the development of an efficient yet interoperable automation system in order to establish a common architecture, other features are necessary. These other requirements are necessary because FF-HSE, EDDL and OPC do not provide resources, for instance, to implement binary and sequential logic as well as enterprise representation (which are better provided by the IEC 61131 standard and ISA S-88).

Then it is very clear that another technology is necessary to integrate all the different standards used in the automation system. As can be inferred from the previous sections, the best approach to promote integration of data, communication and even graphical elements is the usage of XML (W3C, 2006) (and its variations like XML Schema, XSLT - eXtensible Stylesheet Language Transformations, etc). XML is used to integrate different software layers as well as the software in the same network layer because it is also open (in fact it is public), platform independent and has a great number of programs and APIs that allow easy manipulation (read and write operations) of XML files. Even other specifications are being produced having XML as basic element of information and data description, like SOAP, SAX (Simple API for XML), DOM (Document Object Model) and SVG (Scalable Vector Graphics) (Tantaleán, 2003) (Gelle et al, 2003).

Thus, XML can directly consolidate not only device internal data (described, for instance, by EDDL, CFF - Capability File Format and CFH - Capability File for FOUNDATION Fieldbus™ HSE) but also its relationship with the communication workspace (FF, OPC and CORBA - Common Object Request Broker Architeture) as well as configuration and visualization tools (IEC61131, ISA-S88, SGML, SVG). Having XML as the common language being used by the different components of the system, it is possible to extend the cognition to the upper layers of the automation system facilitating the integration of the system with business endpoints, MES, etc. The figure 2 represents the SOAP / XML as the integration technology.

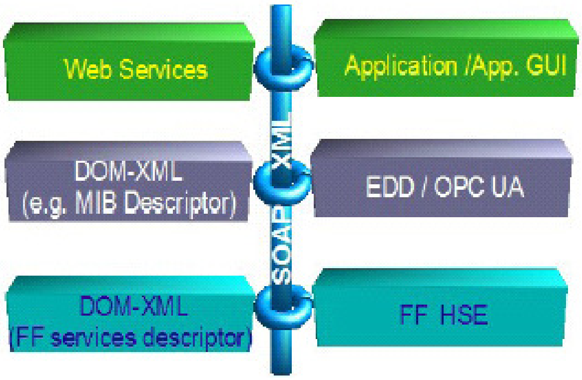

Figure 2: SOAP-XML as the core 'language' of the Common Automation Architecture

The adoption of such Automation System Architecture, can assure long-term integration of the Manufacturing Facility with other business endpoints, since SOAP-XML is currently being applied in all sorts of business automation structures.

6- CONCLUSIONS

The nowadays technology is presenting a new challenge to the mankind, which is the ability to construct networks of virtually anything. Some has called this as 'network of things' (Sang, 2006). This new reality, as shown in figure 3, will need not only new network approaches but also solutions to deal with such amount of information exchanged throughout the network. The new 'network of things' scenario will be constructed on devices that have an enhanced cognition ability, using SOAP /Web Services, characterizing the 'Total Information Age'.

The first implementations of systems that are using these technologies have demonstrated several benefits not only for technicians but also for enterprise managers and business agents.

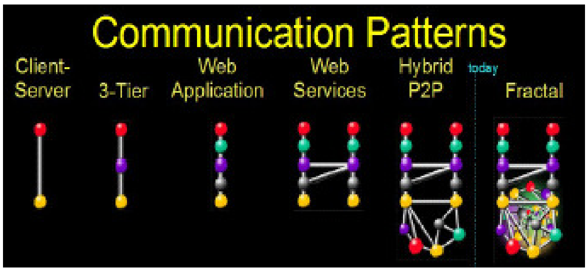

Figure 3: The 'Network of Things'

According to ARC (The ARC Strategies Report, 2001), one set of the benefits is shown in the following graph (Figure 4), which guarantee better ROI figures than conventional, proprietary systems (Verhappen, 2005).

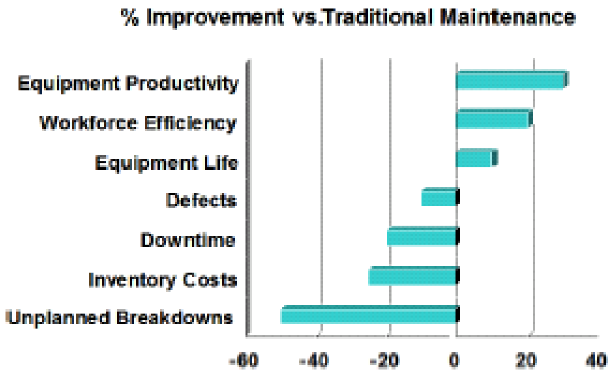

Figure 4: Benefits of a common open architecture
Source: ARC

This graph (Figure 4) reflects how import is to choose not only the best control strategy (or algorithms) but also the components and associated software used to implement the control system itself and its insertion in the enterprise automation. This means that the gains obtained by using a better control system algorithm will not be evident if a sensor used in the implementation is affected by defects not quickly detected. Using modern software the resolution of such defects could be done faster than using legacy software and equipment, reducing the downtime and consequently, reducing unplanned breakdown.

To fully accomplish the above-mentioned scenario this article has shown that the Automation System of the Total Information Age should be able to map the intricacies of its problem domain into SOAP/XML using open standards, constructing a common open architecture. It was shown that EDDL, FF-HSE and OPC UA are able to accomplish such task. TCO is then mitigated since this approach will prepare the enterprise to face the nowadays needs and also the future fractal network structure, where the users will have almost no concern about the network and software technology being used but they will be concerned about the services being offered.

GLOSSARY

API	Application Programming Interface
ARC	Automation Research Corporation
CFF	Capability File Format
CFH	Capability File for FOUNDATION Fieldbus
HSE	High Speed Ethernet
CORBA	Common Object Request Broker Architeture
DCS	Distributed Control System
DOM	Document Object Model
EDD	Electronic Device Description
EDDL	Electronic Device Description Language
ERP	Enterprise Resource Planning
FDT	Field Device Tool
FF	Fieldbus FOUNDATION^{TM}
HCF	HART Communication Foundation
HMI	Human Machine Interface
HSE	High Speed Ethernet
IEC	International Electrotechnical Commission
ISA	The Instrumentation, Systems, and Automation Society (Instrument Society of America)
MES	Manufacturing Execution System
OPC	OLE for Process Control
OPC DA	Data Access
OPC DX	Data Exchange
OPC HDA	Historical Data Access
OPC UA	Unified Architecture
OS	Operational Sytem
PLC	Programmable Logic Controller
PNO	PROFIBUS NutzerOrganisation
ROI	Return Of Investment
SAX	Simple API for XML
SGML	Standard Generalized Markup Language
SOA	Service Oriented Architecture
SOAP	Simple Object Access Protocol
SVG	Scalable Vector Graphics
TCO	Total Cost Of Ownership
XML	eXtensible Markup Language
XSLT	eXtensible Stylesheet Language Transformations
WSDL	Web Services Description Language

REFERENCES

Bourmakine E.; Krassi B (2006). *Distributed Automation and Control Systems.*

Duchastel, P.(2001). *Towards an information realm.* Digit.hvm. ISSN 1575-2275.

Emerson (2005). White paper: *EDDL Technology.*

Fieldbus Foundation (1999a). *FF-581-1.3*: System Architecture.

Fieldbus Foundation (1999b). *FD-043-2.0:* Technical Overview.

Fieldbus Foundation (2006). <www.fieldbus.org/endusersupport/enduserpresentations/prague05/presentations/presentation%20eddl%20pius%20barmet.pdf>

Gelle, E. et al (2003). *A Software Architecture for Industrial Automation.*

Hayes, R.E.; Alberts, D. S. (1996). *Information Warfare and Deterrence*- Appendix B. The Realm of Information Dominance: Beyond Information War.

Merrit, R (2002). *FDT Is Not a Fieldbus Panacea*. Control.

OPC Foundation (2006). <www.opc.org>.

Peluso, M.; Wallace T; (2002). *Distributed Intelligence: Hydrocarbon Engineering.*

Roy J.; Ramanujan A. (2000). *XML data's universal language*. IT PRO. p. 32-36.

Sang S. (2006). *Webservices course* at <www.javapassion.com/webservices> .

Santos P. (2004). XML Access, Formatting and Exporting – The Fieldbus Foundation' s MIB (in portuguese). - Monograph presented to UniCOC – Ribeirao Preto.

Tantaleán, D.(2003). *Applying ODP standard to Integrate CORBA and Fieldbus Architectures*. Dissertation-USP.

The ARC Strategies Report (2001). *Field Device and Sensor Strategies for the E-World*.

Verhappen I. (2005). *Fieldbus – Status and Future Directions*. FF EUC, Buenos Aires.

W3C (2006). *Extensible Markup Language.* <www.W3.org/XML>.

Zielinski, M (2004). *Digital fieldbus installations use EDDL for simplicity with advanced, full functionality*.