Towards Agile Role-based Decision Support for OPC UA Profiles

Dirk van der Linden, Maarten Reekmans
Electromechanics Research Group
Artesis University College of Antwerp
Belgium
{*dirk.vanderlinden, maarten.reekmans*}*@artesis.be*

Wolfgang Kastner
Automation Systems Group
Vienna University of Technology
Austria
k@auto.tuwien.ac.at

Herbert Peremans
Active Perception Lab
University of Antwerp
Belgium
herbert.peremans@ua.ac.be

***Abstract*—Interoperability has become a key factor in modern automation systems. In this context, the OPC (Open Productivity and Connectivity) standards are most prominent. The OPC Unified Architecture (UA) provides platform independent industrial communication using Web-based technologies. It also includes a meta-model guaranteeing interoperability not only at the protocol level but also regarding the semantics of exchanged data. The resulting increased complexity of its specification is an entry barrier for small and medium enterprises. It is challenging to decide which parts of OPC UA a specific company needs to implement. This paper proposes a mechanism for determining the most relevant OPC UA profiles in a specific application domain.**

Keywords-Automation; OPC UA; Profiles; Decision support; Survey; Software tool.

I. Introduction

In the last 10 years, industrial communication has become a key technology in modern industry. A continually growing number of manufacturing companies desire, even require, totally integrated systems. This integration should cover electronic automation devices such as Programmable Logic Controllers (PLCs) and microcontrollers as well as Human Machine Interfaces (HMI) and supervision, trending, and alarm software applications, e.g., Supervisory Control and Data Acquisition (SCADA) and Manufacturing Execution Systems (MES). Industrial communication encompasses the whole range from field management to process management and Enterprise Resource Planning (ERP) applications (business management).

Likewise, the past decade has seen a push towards the integration of building services and building management. Total integration in this field should not only encompass Direct Digital Control (DDC) and SCADA/Building Management Systems (BMS), but also Computer Aided Facility Management (CAFM) applications and HMI ranging from dedicated panels to Web-pads and visitor guidance systems.

The OPC Foundation started in the nineties to promote cross-vendor interoperability for automation projects. Initially, the OPC specifications focused on the Microsoft's proprietary DCOM communication technology. The more recent standard family, OPC Unified Architecture (UA), is designed to be more generic, abstract, technology independent and platform agnostic [1], [2]. However, the resulting increased complexity of OPC UA is an entry barrier for Small and Medium Enterprises (SMEs) that integrate automation systems or provide UA software. The scalability of OPC UA enables interoperability in various fields of application, but it is challenging for SMEs to decide which parts of OPC UA they should implement. The typical activities of SMEs target a specific market niche. In contrast, OPC UA has an almost unlimited field of application. This research project wants to facilitate the process of identifying subsets of the enhanced OPC UA standard for specific targets. Currently, the choice of UA profiles is pragmatically made according to the implementor's knowledge of the OPC UA specifications and the perceived requirements of the application. New concepts of OPC UA, for example information modeling, redundancy or events tend to be skipped due to a lack of awareness of these profiles. Its ultimate goal is a software tool that converts a company's needs to a list of recommended OPC UA profiles, ordered according to the benefits for their business.

This paper gives a short overview on OPC UA and its profiles, then focuses on two major issues concerning OPC UA. First, stakeholders claim that there is a lack of documentation. Second, the generic, abstract concepts of OPC UA result in complex specifications. To discover the most popular environments and technologies and relate the abstract specifications to applications, we performed a worldwide survey. The results of this survey are discussed. In addition, a strategy and tool to improve the choice of profiles are introduced.

The research presented was accomplished within the project "Web-based Communication in Automation" (WebCom), a Consortium Type Project within the EraSME funding program uniting representatives of Research and Technology Organizations and several local SMEs (vendors, system integrators, system developers, consultants). To provide first hand knowledge and expertise on the upcoming OPC UA specifications, the OPC Foundation also agreed to actively support the project and thus serves as an additional partner.

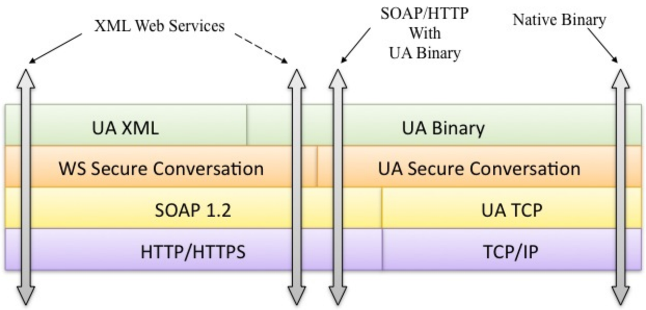

Figure 1. OPC UA transport (Source: OPC Foundation)

II. OPC Unified Architecture and Profiles

Web-based technology is the key to taking interoperability to a new level. Web Services (WS) are totally platform independent – they can be implemented using any programming language and run on any hardware platform or operating system. Easier than ever before, components can flexibly be arranged into applications, collaborating over the Internet as well as corporate Intranets.

OPC UA is considered one of the most promising incarnations of WS technology for automation [3], [4], [5]. From the very beginning, OPC UA was intended as an interface between systems, aggregating and propagating data through different application domains. Its design, thus, takes into account that the field of application for industrial communication differs from regular IT communication: embedded automation devices such as PLCs, Distributed Control Systems (DCSs) or DDCs provide another environment for Web-based communication than standard PCs.

The fundamental components of OPC UA are various transport mechanisms and unified data modeling. The transport mechanisms tackle platform independent communication while still allowing optimization with regard to the involved systems. While communication between industrial controllers or embedded systems may require high speed, business management applications may need high data volume and firewall friendly transport. As a consequence, two data encoding schemes are defined, named OPC UA Binary and OPC UA XML. Different compromises are possible to find a good balance between security and performance, depending on the application (Figure 1).

Data modeling defines the rules and basic building blocks necessary to expose an information model with OPC UA. Rather than supporting data communication, it facilitates the conversion of data to information. Rather than introducing unnecessary new formalisms, the OPC Foundation encourages definitions of complex data based on related industrial standards. Examples are FDI (Field Device Integration), EDDL (Electronic Device Description Language [6]), IEC 61131-3 (PLC programming languages [7]) and ISA 88 (batch control [8], [9]). Basically, an OPC UA information model is made up of nodes and references between nodes. Nodes can contain both online data (instances) and meta data (classes). OPC UA clients can browse through the nodes of an OPC UA server via the references, and gather semantic information about the underlying industrial standards. For clients, it is very convenient to program against these complex data types. They also bring a potential of code re-use. Note that the OPC Foundation provides dedicated OPC UA information models to structure the legacy OPC specifications (Figure 2). These information models facilitate the migration of legacy OPC interfaces to OPC UA interfaces [10].

OPC UA is designed in a way that individual implementations do not need to support all features, but can be downscaled to a limited scope if desired. At the same time, advanced products which allow a high degree of freedom will require the support of more sophisticated features. A service based OPC UA implementation can be tailored to be just as complex as needed for the underlying application.

Hence, what is needed is a way to describe (and test) which features are supported by an OPC UA compliant product. This is where the OPC UA terms *ConformanceUnit* and *Profile* come into play [11]. A specific set of features (e.g., a set of services or a part of an information model) that can be tested as a single entity is referred to as a ConformanceUnit. An example of a ConformanceUnit is the *Call service*. This service is used to call a method on an OPC UA server. ConformanceUnits are further combined into Profiles. An application (client or server) shall implement all of the ConformanceUnits in a Profile to be compliant with it. Some Profiles may contain optional ConformanceUnits which in turn may exist in more than one Profile (Figure 3). The term *facet* is used to refer to Profiles which are expected to be part of another larger Profile or which concern a specific part of OPC UA. Software certificates contain information about the supported Profiles. OPC UA Clients and Servers can exchange these certificates via services [12].

Up to now, more than 60 OPC UA Profiles have been released [13]. However, it is expected that the list will be extended over time – even by other organizations than the

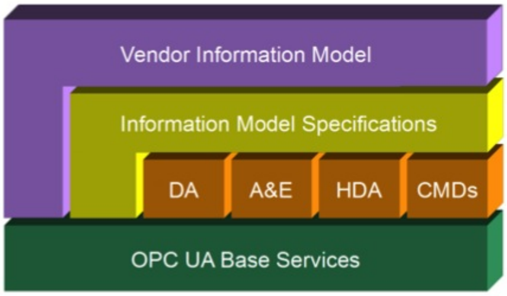

Figure 2. OPC UA Information Modeling (Source: OPC Foundation)

OPC Foundation. At this moment, OPC Foundation working groups are working on new, upcoming profiles.

III. Survey

We assume that complaints about OPC UA not being documented well and excessively complex originate mainly from the conflict between the wide range of features and possibilities of the standard on one hand and the strongly focused, niche-oriented business needs of SMEs on the other. Therefore, our goal was to identify which specific sets of profiles add the most value for SMEs. We anticipated that OPC UA implementations for SMEs could be significantly facilitated if the documentation to be provided could be narrowed according to their particular field of application. Consequently, our intent was to determine recommended sets of OPC UA profiles according to the industry sector a business is active in.

This approach follows the assumption that each industry sector requires specific automation applications, resulting in a typical set of automation technologies being used and, likewise, having typical requirements on data communication within and between these technologies. Knowing which OPC UA profile (or combination thereof) is designed to fulfil given communication requirements, it should in this case be possible to recommend a set of profiles based on the industry sector. For example, the redundancy profiles can be recommended for sectors like chemical industry, where high availability is important. Traceability is important for the pharmaceutical sector so the Auditing profiles will be included in the recommended profiles list.

We designed a survey to validate this assumption. The survey did not assume any detailed knowledge of OPC UA profiles on the part of the respondents, but focused on generalized questions regarding communication requirements that would allow drawing conclusions about required profiles. To make sure that these questions reflect the capabilities of the available OPC UA profiles well, we consulted one of the lead authors of the Profiles part of the OPC UA specifications for expert advice. To address a representative number and kind of stakeholders, the survey was distributed to OPC Foundation members as well as companies that figure on the Foundation's regular mailing list and several other industry specific mailing lists containing a wide variety of respondents in addition.

About 25,000 questionnaires were sent out, and a total of 719 responses were collected. The geographical distribution of all respondents is shown in Figure 4. It largely matches the geographical distribution of the OPC Foundation members. The most important industrial sectors of respondents (15% or more of responses) are listed in Table I. Multiple answers to this question were allowed. We observed that a significant number (10–15%) of respondents is active in up to 8 different industrial sectors, and still 5–10% are represented in up to 5 areas.

From the results, some conclusions can be drawn as to which parts of the standard are currently of greatest interest and therefore should receive particular attention in general. In particular, we see PLC and SCADA dominating the list of automation technologies in use by the respondents (Table II; PAC = Programmable Automation Controller, TFM = Technical Facility Management). Also, quite general management tasks such as alarm, event and user logging received high importance rankings among respondents.

Table I
Industry sectors most relevant to respondents

Oil & Gas Production	18%
Oil & Gas Distribution	15%
Chemical	18%
Food & Beverage	16%
Power Distribution	16%
Power Generation	20%
Building Automation	19%
Automotive Industry	16%
Industrial Automation	38%
Process Automation	30%
IT	19%

Table II
Technologies in use by respondents

ERP	MES	SCADA	PLC	PAC	DCS	TFM	BMS	DDC
30%	24%	66%	72%	29%	43%	9%	18%	13%

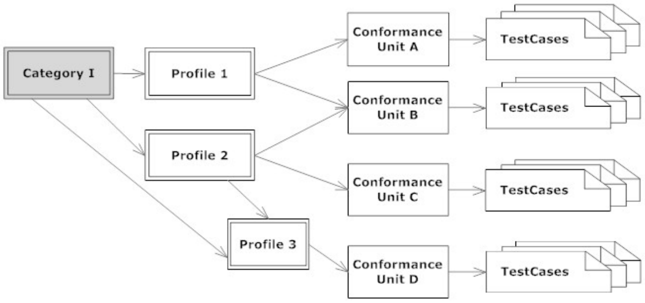

Figure 3. OPC UA Profiles and ConformanceUnits

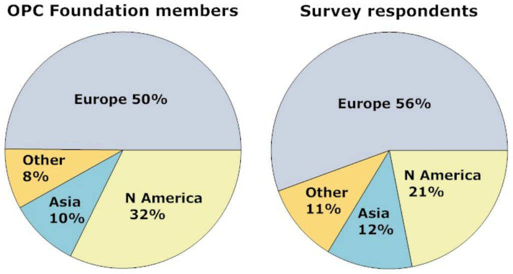

Figure 4. OPC Foundation members and respondents by region

Thus, while there is apparently the need for support of a wide range of different systems, initial implementation effort can be significantly reduced by focusing on these technologies. Considering that many applications follow a very basic pattern, many implementers will only need to provide the so called *Core Server* facet profile, in combination with one *Transport* facet. So an implementation with the same functionality as available with the former DCOM OPC DA, but in a platform independent way. To further reduce the programming effort, wrappers and proxies are available [10].

We found some key trends and assumptions behind the OPC UA technology that can be confirmed by the survey results. For example, 432 interviewees stated to be manufacturer of systems or products that use a communication network. 59% use a field device network, and 37% use the control network in a shared network set-up with the standard computer network. This illustrates the high importance of industrial data communications in general as well as the drive towards combined communication networks and totally integrated systems.

As far as the speed of communication is concerned, communication within less than one second is required by the majority (165/355) of PLC/PAC/DCS users. Also, the time frame for delivering data in the control network is typically short (15% say less than 1 ms; 55% say less than one second). However, a substantial percentage of PLC/PAC/DCS users (81/355) are satisfied with a delivery of data/messages within less than one minute.

This shows that on one hand, demand for fast and efficient transport as provided by UA Binary transport is significant. On the other hand, a large market segment exists where speed does not matter as much as other qualities of service. Also, lower speed may well be acceptable if compensated by other desirable properties such as firewall friendly communication, which would for example be a key property of the SOAP-HTTP WS-SC XML transport facet.

There is also a strong demand for security and robustness. The top three security related issues among respondents are authentication, restricted access and confidentiality of transferred data; for availability, utilizing redundant servers is seen as more relevant than deploying redundant clients.

Regarding operating systems and programming languages in use by the respondents, a technology shift begins to show. Though Windows is still the leading operating system being deployed, a trend towards Linux can be observed. Relevant programming languages are, in decreasing order of importance, C/C++, C#.NET, VB.NET, and Java. The rise of .NET indicates that DCOM is becoming a legacy technology. The use of C#.NET and C/C++ is significantly higher than the other languages ($p < 0.001$). Differences concerning the use of the programming languages in different regions are not significant (at a p-value of 0.05), which leads us to conclude that the technology shift is happening worldwide.

To confirm the suspected dependencies between industry sectors, automation technologies and communication requirements, we applied logistic regression analysis [14] to the survey results. In such an analysis, the estimates of the weight of variables with regards to a specific use provides an idea of the relevance of these variables.

We found the use of MES to be very high in the food and beverage industry. PLC systems are being used nearly everywhere except in power distribution and IT (with negative estimates of -0.54 and -0.89, respectively). The use of DCS systems is also very diverse, except in the automotive industry, which instead shows a significant use of PAC (at an estimate of 0.70). SCADA is present in power generation, industrial automation, food/beverage and oil production, with a negative estimate for the IT sector (-0.64). Overall, PLC and SCADA are quite correlated (0.55).

Again, using logistic regression analysis, we found differences of preferences of programming languages with regard to the type of automation technology in use. The majority of Java users can be found among ERP, MES, SCADA and TFM users (as confirmed by the Hosmer and Lemeshow Goodness-of-Fit test). The majority of C#.NET users work, in decreasing order, with MES, SCADA and ERP systems. The majority of C users focus on SCADA, DDC and BMS systems. The diversity of VB users is the biggest, they work with PLC, SCADA, MES, DCS and ERP systems. The selection of these technologies is based on the analysis of maximum likelihood estimates of a simplified model with an entry cutoff value of 0.15 and a stay cutoff value of 0.15.

Concerning the most common security issues, we found a good fitting logistic regression model showing that ERP users value rogue system detection, auditability of actions, confidentiality of proprietary data and network intrusion avoidance. PLC users have different priorities, with a focus on auditability of actions, availability of systems, restricted external access to proprietary data and network intrusion avoidance. PAC users place a similar (but lower) priority on network intrusion avoidance, availability of systems and restricted external access. MES users assign high importance to preventing the alteration of proprietary data, auditability of actions, network intrusion detection and authentication of users.

The users who need a very short time frame (less than 1 ms) for delivering data/messages via the control network are mostly MES and PLC users. Those who need the fastest message exchange via the computer network (less than one second) are mostly PLC, MES and SCADA users.

We however did not find a straightforward correlation between industry sectors, target technologies and communication requirements that could have been translated into a simple, static set of profile recommendations. We therefore chose to design an agile decision support tool.

IV. Role-based optimization strategy and tool

The logistic regression analysis of the survey results showed that a large number of parameters would be necessary to determine a recommended profile. This makes the regression models too complicated for practical use. In fact, it implies that the correlation between a “typical implementor” (as defined by a naive classification according to target industry sector or automation technology) and any set of recommended profiles is low. It is therefore infeasible to stick to one single set of questions to deal with all stakeholders’ needs. Arriving at this conclusion, we opted for another approach: an online tool to dynamically produce recommended sets of profiles on an individual, user-by-user basis. The tool is based on the generalized questions regarding communication requirements which reflect the capabilities of the various OPC UA profiles that were created for the survey. It is designed to easily accept new or updated questions to reflect newly released profiles. This *agility* is an additional advantage, as the definition of OPC UA profiles by the OPC Foundation working groups is an ongoing process.

Considering our initial goal of identifying which specific sets of profiles would add the most value for a SME, we wanted to have the tool take into consideration the economic dimension in addition to the technical one. Each vendor has its own target market, with a diverse set of customers and specific fields of application. While many profiles might make sense from a technical point of view (and thus may well all be requested by customers), implementing some profiles will provide more commercial benefits than implementing others. Vendors must meet the challenge to find the balance between satisfying customer requirements and return on investment for implementing these profiles.

To best support this decision, our tool should therefore assign a priority to each recommended profile. Also, it should be capable of linking an estimate of commercial benefits based on development time and budget to this prioritized list of recommended profiles. With this information, end-users of the tool can more accurately envision the development planning of a product even without detailed knowledge of OPC UA technologies, as this knowledge is embedded in the tool. Thus, for getting the most relevant results the implementation of the decision support tool takes into account normative constraints (i.e., it shall produce output that is consistent with the OPC UA specifications), budget constraints and maximum commercial benefit (Figure 5).

The decision support tool takes its input from three sources, each representing a particular competence or role. These inputs provide the functional parameters for the decision support tool. When the experts have entered these parameters, the end-user who typically has little knowledge of OPC UA profiles, can use the tool to help determine the list of recommended profiles for their company and application.

The first role is that of an OPC UA expert who is determining the normative constraints. The main task of the UA expert is to input a set of survey questions and possible answers. Each answer is then linked to one or more profiles. Using these relations a profile is produced according to the answer given by a respondent to the respective question. Besides, what we call *static* normative constraints have been hardcoded into the software. Some examples of these static normative constraints are that no product can be built with only one profile and that an application must at least support one of the core facets, one security facet and one transport facet. Another example of a static normative constraint relates to nested profiles: the basic profile must be implemented before an enhanced profile can be implemented (e.g., *Core Server* can only be implemented when *SecurityPolicy - None* has already been implemented).

The second role is that of a software architect who provides input regarding the development time required for implementing a specific profile. The software architect must have detailed knowledge of OPC UA profiles to do this. The development time is put into the tool once per profile. The end-user has to provide some additional parameters like the cost of programming labour in their company, the preferred programming language and an indication of the complexity of the application behind the OPC UA interface to get the total cost of implementation of a specific profile.

Third, the role of technical-commercial manager (sales / business) is to estimate the commercial benefit of implementing a specific profile. This commercial benefit can be estimated and used as a parameter to manage the development priority. Some of the commercial benefits can be estimated by the results of our technology survey. As mentioned, it should be noted that typically the technical-commercial role does not have enough OPC UA knowledge to estimate the benefit of a profile directly, which means that they especially profit from decision support as described in this section.

A survey is restricted to a static, limited set of questions and can never anticipate all upcoming changes and new profiles. In contrast, the decision support software handles every profile, cost and benefit as an abstract parameter. This

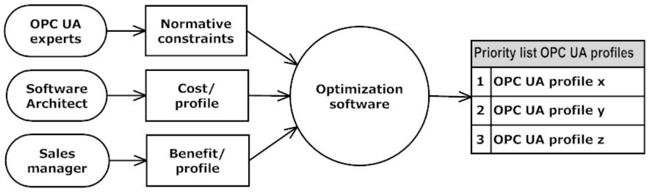

Figure 5. Role-based decision support

allows dealing with diversity and upcoming changes dynamically. New profile cost information and benefit parameters can easily be added.

The online tool is based on CakePHP [15], an open source web development framework. CakePHP follows the Model-View-Controller (MVC) design pattern, which greatly facilitates the creation of database-driven applications such as this. The decision support tool is agile because of the relational database being designed in a way to enable the straightforward addition of:

- New users with a specific role
- Newly released profiles, including their nesting relationship with existing profiles
- New questions and their answers, with an appropriate link to a profile
- New programming languages
- Cost parameters

V. Conclusion and outlook

Thanks to its technology-independence and scalability, OPC UA has a high success potential provided that implementers can reduce the overall complexity by focusing on their specific field of application.

In comparison with the legacy ("classic") OPC specifications, implementing OPC UA is in no way more complex as long as implementers do not attempt to provide more functionality than they need to. On the contrary, it can even be expected that many OPC UA implementations require less effort thanks to the increased development flexibility offered. Formerly, a server implementation had to support all mandatory interfaces, while OPC UA only requires the Core Server facet and a transport facet.

As the conducted survey did not yield a conclusive set of profiles associated with an application field, it was decided that a more personalized result on a user-by-user basis could be more valuable to implementers of OPC UA technology. The agile role-based decision support tool for OPC UA profiles has the goal to deliver a clear cut list of functionality that these implementors need to develop for their products. This will hopefully lead to faster adoption of the OPC UA technology, despite it being generally perceived as involving a very complex set of specifications.

Currently, the framework for the decision support tool is being tested and a working group of OPC UA experts is being assembled to synthesize the questions and answers. By using the input of these experts, a prioritized list of recommended profiles can be produced which is the main feature of the tool and is already implemented. Calculating the cost and commercial benefit is a complex matter and further research will determine the feasibility of these features. When the tool is released after beta testing it will be made freely available to the public in cooperation with the OPC Foundation.

Acknowledgment

This paper is part of the project WebCom [16] funded within the EraSME program by the Flemish Agency for Innovation by Science and Technology (IWT) and the Austrian Research Promotion Agency (FFG). The authors thank all respondents of the survey and the OPC Foundation for constructive collaboration. In particular, we would like to thank Paul Hunkar, one of the lead authors of the OPC UA profiles specification, who has provided valuable consulting expertise. The authors also thank Georg Neugschwandtner.

References

[1] OPC Foundation, OPC UA Part 1 – Overview and Concepts 1.01 Specification, 2009.

[2] International Electrotechnical Commission, OPC Unified Architecture Specification, 62541, 2010.

[3] W. Mahnke, S. Leitner, and M. Damm, OPC Unified Architecture, Springer, ISBN 978-3-540-68898-3, 2009.

[4] J. Lange, F. Iwanitz, and T. J. Burke, OPC – From Data Access to Unified Architecture, VDE-Verlag, ISBN 978-3-8007-3242-5, 2010.

[5] van der Linden D., Mannaert H., Kastner W., Vanderputten V., Peremans H., and Verelst J., "An OPC UA Interface for an Evolvable ISA88 Control Module", IEEE Conference on Emerging Technologies and Factory Automation, 2011.

[6] D. Grossmann, K. Bender, and B. Danzer, "OPC UA based Field Device Integration," Proc. Society of Instrument and Control Engineers Annual Conference 2008 (SICE 2008), pp. 933–938, 2008.

[7] PLCopen and OPC Foundation, OPC UA Information Model for IEC 61131-3 1.00 Companion Specification, 2010.

[8] J. Virta, I. Seilonen, A. Tuomi, and K. Koskinen, "SOA-Based integration for batch process management with OPC UA and ISA-88/95," Proc. 2010 IEEE Conference on Emerging Technologies and Factory Automation (ETFA 2010), pp. 1–8, 2010.

[9] van der Linden D., Mannaert H., and de Laet J., "Towards evolvable Control Modules in an industrial production process", ICIW 2011, 6^{th} International Conference on Internet and Web Applications and Services, St. Maarten, pp. 112-117, 2011.

[10] T. Hannelius, M. Salmenpera, and S. Kuikka, "Roadmap to adopting OPC UA," Proc. 6th IEEE International Conference on Industrial Informatics (INDIN 2008), pp. 756–761, 2008.

[11] OPC Foundation, OPC UA Part 7 – Profiles 1.00 Specification, 2009.

[12] van der Linden D., Mannaert H., and De Bruyn P., "Towards the Explicitation of Hidden Dependencies in the Module Interface", ICONS 2012, $7th$ International Conference on Systems, Reunion, pp. 73-78, 2012.

[13] "OPC UA Profiles," www.opcfoundation.org/profilereporting/ (last accessed 2012-01-19).

[14] S. Sharma, Applied Multivariate Techniques, John Wiley & Sons, ISBN 0-471-31064-6, 1996.

[15] "CakePHP: the rapid development PHP framework," www.cakephp.org (last accessed 2012-01-19).

[16] "Webcom Research Project," www.webcom-eu.org (last accessed 2012-01-19).