

A Migration Approach towards a SOA-based Next Generation Process Control and Monitoring

Jerker Delsing ^{*}, Jens Eliasson ^{*}, Rumen Kyusakov ^{*}, Armando W. Colombo [†],
François Jammes [‡], Johan Nessaether [§], Stamatis Karnouskos [¶], and Christian Diedrich ^{||}

^{*}Luleå University of Technology, Luleå, Sweden, Email: jerker.delsing@ltu.se, jens.eliasson@ltu.se, rumen.kyusakov@ltu.se

[†]University of Applied Sciences Emden & Schneider Electric, Germany. Email: awcolombo@et-inf.fho-emden.de

[‡]Schneider Electric, France. Email: francois2.jammes@schneider-electric.com

[§]Midroc, Sweden Email: johan.nessaether@midroc.se

[¶]SAP Research, Germany Email: stamatis.karnouskos@sap.com

^{||}IFAK, Germany Email: christian.diedrich@ifak.eu

Abstract—Interest in Service Oriented Architectures (SOA) in the automation domain has seen a rapid increase both from the academia as well as the industry recent years. Since green field plants today are not common, the partial migration of plant automation to SOA design is needed to introduce new functionalities. Thus strategies and approaches for migration from legacy to SOA architectures becomes of vital interest. This paper discusses different views on partial migration of a process monitoring and control system from legacy to SOA. The discussion includes a global top down view, a bottom up view, hardware/software considerations and a hint on training of personnel.

I. INTRODUCTION

Advanced Information and Communication Technologies (ICT) are increasingly getting integrated in industrial shop floor systems; this leads to significant IT dependency as well as shorter life-cycles in the introduction of new functionality. As we are moving toward the “Internet of Things” vision, millions of interconnected devices (e.g. sensors, actuators, SCADA/DCS, mobile devices etc.) will provide and consume information available on the network and cooperate. Service Oriented Architecture (SOA) seems to be a promising solution to realize the necessary cross-layer interaction of heterogeneous devices and systems and ensure higher degrees of interoperability.

To address the emerging integration challenges of very large numbers of (sub)-systems and devices, the AESOP [2] project builds upon promising work carried out in several European collaborative projects such as SIRENA [4], SODA [5], SOCRADES [9], [11], etc., all of which demonstrated that embedding Web Services at the device level and integrating these devices with MES and ERP systems at upper levels of an enterprise architecture was feasible [6]–[8].

The current ongoing R&D results around the industrial applications of the SOA paradigm aim to further advance the vision towards an infrastructure that will enable cross-layer service oriented collaboration not only at horizontal level, e.g. among cooperating devices and systems, but also at vertical level between systems located at different levels of a Plant-Wide System (PWS) enterprise architecture [6], [8], [9].

Large scale monitoring as well as management (soft control) functions at all levels of an enterprise [8] may benefit significantly by tapping in to the services offered by the collaborative SOA-based infrastructure. By taking advantage of the capabilities of Cooperating Objects [10] future “Perfect Plant- Wide System” [6], [11]–[13] will be able to seamlessly collaborate and enable monitoring and control information flow in a cross-layer way, i.e., within and between the different levels of the enterprise architecture. The different systems will be part of a SCADA/DCS ecosystem, where components can be dynamically added or removed and dynamic discovery enables the on-demand information combination and collaboration [3]. All current and future systems will be able to share information in a timely and open manner, enabling an enterprise-wide system of systems [14], [16] that will dynamically evolve based on business needs, and will present emergent behaviours as result of composition and orchestration of the services exposed in the ecosystem.

A critical issue that arises when the SOA infrastructure is introduced is the structural and behavioural integration of legacy devices and systems. Today by using SOA we can already target at some degree future compliance and follow concepts and approaches that will enable designing today the acceptable “legacy” SOA system of tomorrow, that may be easily integrated in long-running infrastructures (e.g. the chemical, pharmaceutical etc. ones who boast long operational lifecycles of 15-20 years). As explained later in section IV, “services encapsulating device and systems specifications” are used as wrapping information technology that enables to progressively migrate standalone devices as well as more complex systems and make them part of a fully service based ICT infrastructure.

This paper is organized as follows: Section II presents a SOA architecture and summarizes the major necessary specifications to be addressed to enable the proposed architecture. Following, Section III and IV present the major characteristics of a migration approach and related migration paths for the transition from legacy systems to the SOA architecture. Finally, Sections V present conclusions and suggestions for future work.

II. ARCHITECTURE

The SOA-based enterprise architecture allows devices and systems, from the shop floor to the business levels, in a cross-layer interaction mode:

- To expose their structural and behavioural properties, their functionality and networking capabilities as “services”, and
- To be accessed, approached and integrated in collaborative business relationships with other devices and systems as “services”.

Fig. 1 shows the proposed architecture, composed of services (marked as “S” and depicted in a green cube), which are wrapping via web services many different devices and systems in an independent way no-matter of the physical location of the devices and systems in the enterprise architecture. The common feature is the use of the Internet Protocol (IP) suite and mostly web services used throughout all layers and in all sub systems.

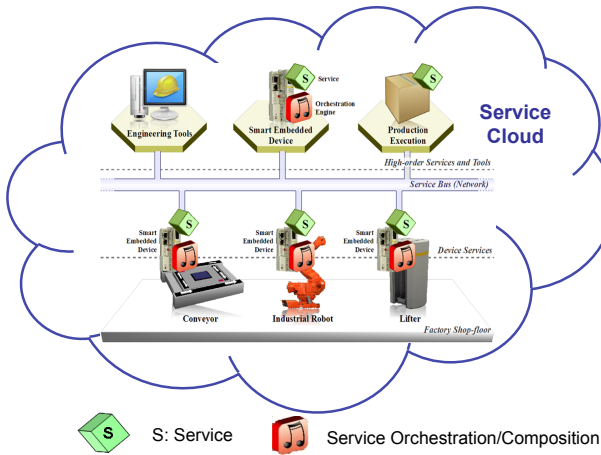


Fig. 1. SOA-driven architecture

This cloud-based approach allows multi-level composition of systems of systems and services of services, as shown in Fig. 2. The use of a cloud-based architecture enables tight integration and interaction between high-end business systems e.g. ERP/MES (see www.mesa.org) and factory floor systems e.g. SCADA/DCS, thereby allowing different management, control, monitoring, MES and other supervisory control functionalities, as a result of the composition and service orchestration [3]. It is this composition which allows the whole system to develop emergent behaviours that were not initially envisioned to be exposed by the constituent (composed) systems. The service composition can act as an enabling tool that will allow the whole system to reach common goals like optimized line workloads, energy management related to shop floor throughput, etc. Moreover, the proposed approach facilitates the integration of wireless sensor and actuator networks

(WSAN), whose functionality is also wrapped. The use of a large number of low-cost sensors enables new information to be collected by the system, and can allow for better utilization of available resources and assets through for example: better maintenance, localization, and fault detection to mention a few [15].

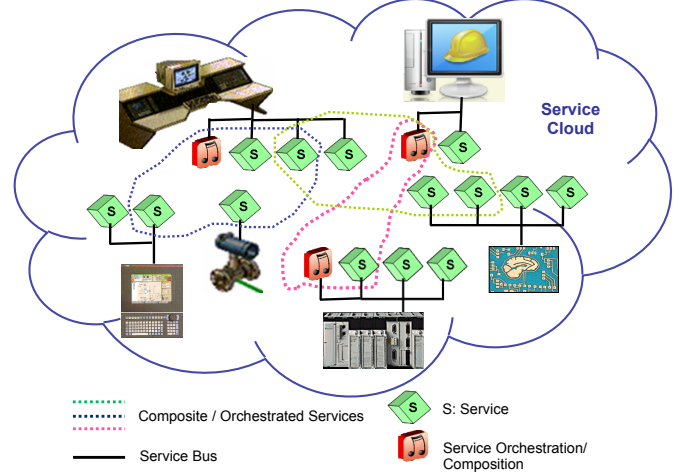


Fig. 2. Hierarchical composition of services enables abstract cross-layer functionality.

Having in mind that a SOA-based system behaves asynchronously (in opposition to the majority of current implemented industrial process control and monitoring systems), to seamlessly integrate a large number of devices and systems from different manufacturers into a single SOA ecosystem is a complex and challenging task.

A first step is the identification of the right wrapping ICT technologies. Here the specification, definition and implementation of standardized services is realised. Many of them are identified as “generic services” because are common for all devices and systems, and other labelled as “infrastructure services” by SOCRADES (www.socrades.eu) and NESSI (www.nessi-europe.com) projects.

The next step is the specification and implementation of composition, orchestration and choreography mechanisms, which have to mainly include two possible functions: (i) processing the information content of the services, and (ii) processing the events associated to the services. This is shown by the “music” icon on Figure 2.

Of course there are other complementary steps e.g. the definition, specification and implementation of control, monitoring and management mechanisms that exploit all the potential offered by the SOA-based systems but their discussion is not part of this work (the reader is recommended to consult material available by the SOCRADES (www.socrades.eu) project).

III. MIGRATION

A migration approach from current legacy to a SOA-based industrial system of systems [14], [16] will follow a set of basic steps, which can be summarized in Fig. 3. Current

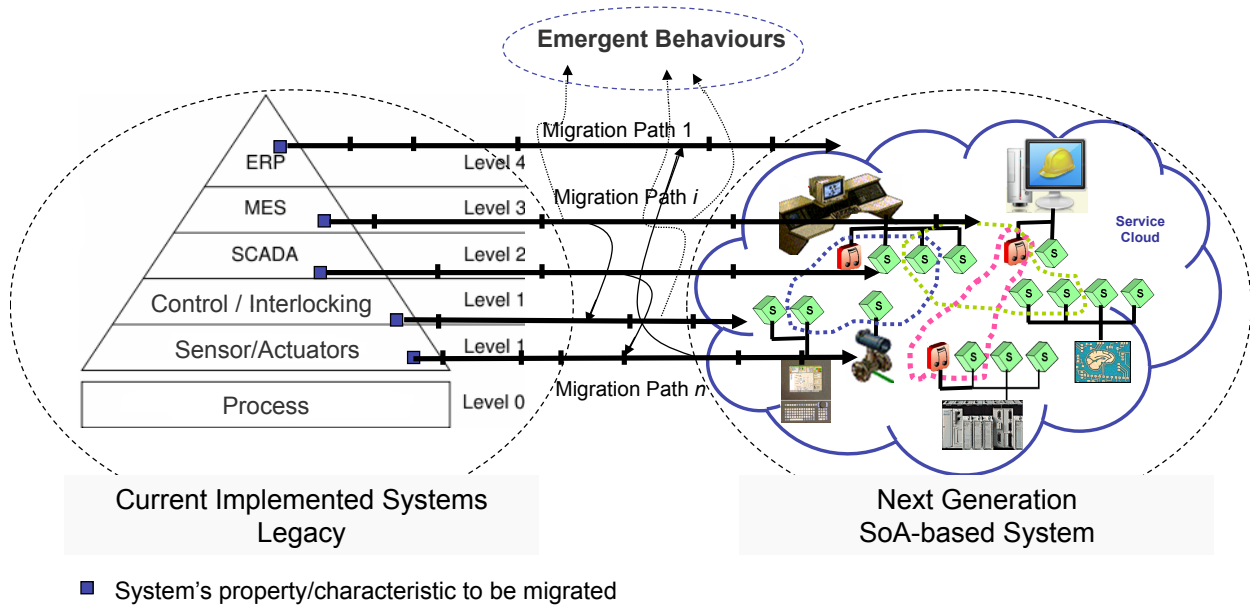


Fig. 3. Migration Approach from Legacy to SOA-based Systems.

legacy industrial systems are mainly specified, implemented and running following the ISA95 (www.isa-95.com) standard. This means, a migration to a SOA-based system cannot in general be performed in only one or some of the levels of the architecture shown on the left side of Fig. 3. This is because specifications and system characteristics in a defined level are closely related to specifications in other levels (e.g. a control specification in Level 1 will only be well implemented when it is considering information and actions performed in the neighbour levels like SCADA or MES above them). Thus a migration strategy has to address how the migrated part can represent the legacy functionality and how it is involved to other level of the control system.

For each specification and/or characteristic in a level, one or more migration paths will have to be identified. Each time that a migration approach will have to be applied, a first obligatory activity will be to define and understand the differences with the legacy infrastructure. Let take as an example the task of migrating to SOA-based monitoring and control system. To enable this migration, it is mandatory to define and understand differences to legacy monitoring and control systems that will arise. After a first analysis, we consider the following key characteristics of a legacy monitoring and control system can be identified from a:

- **Top down perspective:** Polling, or scan based, time triggered data sampling is often utilized. This enables good control of hard real-time properties. Often, there is a mixture of different communication hardware and structures, such as (i) Point to point links, (ii) Busses, and (iii) Protocols
- **Bottom up view:** In today's architectures, data transfer

can be either analogue or digital. Analogue data is per definition streaming, while digital data can be either streaming or by request.

Compared to a state-of-the-art SOA architecture, there are a number of differences. The resulting SOA-based Monitoring and Control System has the following properties:

- **Top down view:** A service-driven architecture allows devices and functions to be exposed as services. Distributed system models are necessary in order to fully capture and express the distributed approach of a cloud-based network composed of services (and services-of-services). In order to manage a large scale network, consisting of thousands of SOA-enabled devices, maintenance services, tools and methodologies are required. One important service process is automatic device and service discovery which overcomes some of the obstacles of manually discovering and configuring a large number of heterogeneous devices. Service orchestration is another key technology in order to be able to compose complex services, and to group services together.
- **Bottom up view:** In SOA-based approaches all data transfer is performed using layered protocols usually on top of IP. However, we distinguish between data by request (on-demand), i.e. scan-based, and event-based communication. Event-based data transfer holds some interesting properties and can help mitigate the overhead associated by the use of general-purpose SOA and its implementation protocols.

After identifying and understanding those differences and deciding to proceed with the application of the migration approach, we follow the specification of the migration path

for a given legacy system specification or characteristic, which will content a set of steps that will start with the analysis of the current structural and behavioural properties bound to the given specification. We will continue with sequential changes in static or dynamic (or both) aspects of the given property and will finish with a SOA-based properties or characteristics as shown on the right side of Figure 3.

It is expected that several migration paths will exist, possibly as many as system's specifications to be migrated, and each of those paths will additionally have its own number and type of migration steps. Moreover, since the specifications in different levels are related, the migration paths will also have correlations and possibly conflicts. This means, interdependencies among migration paths are due to architectural and functional relationships between systems' characteristics. The later has a great significance for the migration approach, because it is due to these cross-relationships among migration paths that some emergent behaviour in the new SOA-based system of systems will be possible [16].

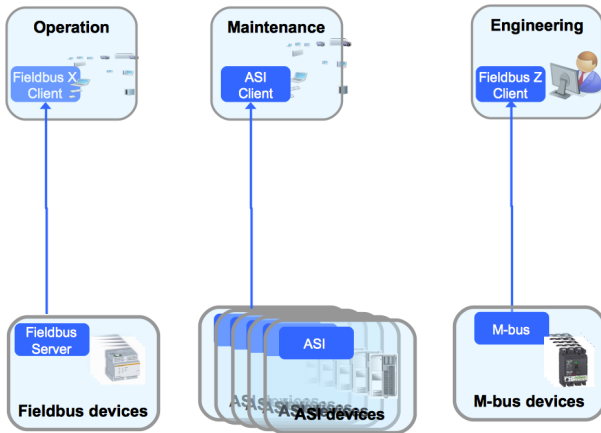


Fig. 4. A simple legacy process monitoring and control system design.

A break down and simplification of Fig. 3 can be seen in Fig. 4 and Fig. 5. In a basic legacy implementation, Fig. 4, we find three independent user domains (most often with no or minimal integration) necessary for a plant operation i.e. (i) operation, (ii) maintenance and (iii) engineering. Using a SOA approach these three domains can be integrated as shown in Fig. 5.

Migrating from legacy independent systems to SOA through an intermediate level: Legacy solutions, shown in Fig. 4, are usually using their own (proprietary) protocols to interconnect devices and systems, leading to isolated, independent systems which are not interoperable nor easy to migrate. A migration path to provide a full interoperable solution integrating all systems and devices together, is possible by adding an intermediate level providing a SOA compliant view of legacy devices through connectors to legacy devices. This approach, shown below in Fig. 5, does not require updating legacy devices in order to make them SOA compliant, which is a long-term path that cannot be always a priori understood, especially from a short term perspective. The above approach is given

from an operational view point. It is clear that an engineering configuration aspect has to be addressed as well.

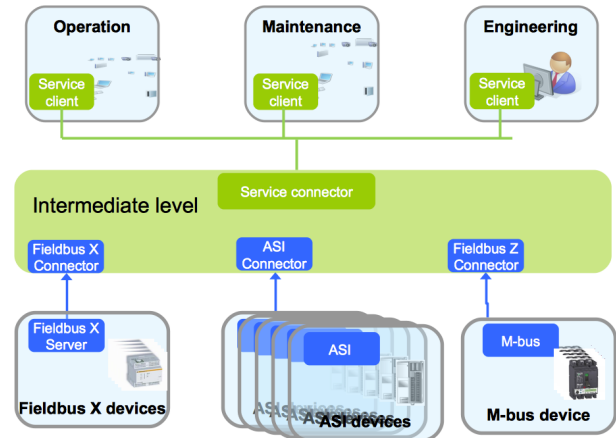


Fig. 5. The introduction of SOA at a higher level in an ISA95 based control system architecture using an intermediate level where field device and there data are converted to services.

It is important to recall here, that there must always be a well-thought purpose of the migration from legacy systems to state-of-the-art SOA-enabled large scale systems: some end users might strive for a well defined goal e.g. e-maintenance or a more standard-compliant extensible system, while other end-users might get SOA components virtually without knowing it; for instance when SOA components are integrated with other delivered functionality.

Today, for instance, gauges can transmit a reading using a 4–20 mA current loop but make the transformations using a microprocessor and also have a serial port to configure levels when to switch relays, virtually creating a combined gauge and switch in one device. Machine designers use such devices and sometimes do not know or ignore additional interfaces and features. Well-planned migration is essential because end users might hesitate due to risk of unforeseen/unplanned costs.

IV. MIGRATION STRATEGIES FROM LEGACY TO SOA

In this section, we present two typical scenarios and outline a migration strategy for each of them. The two scenarios are:

- A SOA based monitoring of maintenance information for a lubrication system interfaced to a legacy maintenance system
- A legacy PI control loop ensuring the oil pressure supply in a lubrication system interfaced to a governing SOA system

For the lubrication monitoring scenario we have a SOA-enabled level sensors, a valve with positioning monitoring, and torque sensors at motors driving the shafts whose bearings are lubricated by the lubrication system. The governing system is scanning data once per second.

The services provided by these devices are:

- Level, scaling, id, status, ...
- Desired valve position, actual valve position, id, ...
- Torque, scaling, id, status, ...

Each of the devices are connected to the governing SCADA system e.g. via an RS-232 link. For the pressure PI control scenario we have a local control reading a pressure sensor and actuating on a pump. The controller has a set of parameters that can be set through an M-bus interface. The M-bus is connected to a governing SOA based DCS system.

These two scenarios introduce a number of legacy-to-SOA monitoring and control interface situations that need to be addressed. From the device level, interfacing a lower level monitoring SOA monitoring and control block to a higher level scanning legacy monitoring and control system. Here the SOA block (services of services, virtual “sensors” e.g., aggregation, prediction, estimation, filtering) need to adapt and provide and interface to the higher level legacy system. This required interface will include issues like data update rate, data format, used protocols, etc. The configuration of the SOA block interface should ideally be obtainable as a service from a device/service management system.

Interfacing legacy analog data streaming from the PI controller to the SOA monitoring and control. The governing SOA system can schedule data fetch events to the M-bus based on the system model of required data. This constitutes a service event in the SOA M&C paradigm. The system model needs to determine when the obtained data should generate an event of interest to another SOA service in the monitoring and control system. Thus the governing SOA system replicate the streaming analog functionality in IT terms of SOA.

Interfacing a packet based data (on request) device to a SOA service is rather straightforward (SOA acts as a wrapper on an already event based solution). Here the SOA service will need device description including means of data request and data format (possibly obtainable as a service from a device management system providing information from the devices). Based on this, the service can be configure to appropriate actions.

One part of the migration strategy is how we physically translates from legacy approaches like 4–20 mA, M-bus, RS-232 etc. to a service interface. Such translation to Service-Oriented Architectures solutions may include:

- Using wireless solutions, providing topological and hierarchical flexibility not generally achievable using wired communication technologies is an approach which is slowly getting acceptance by the industry even though the academic world has long embraced it. New architectures, such as Wireless HART and 6LoWPAN, are starting to be used in various applications. The use of wireless, and even battery powered devices, allows for sensing physical properties not possible before due to high installation costs when using wires, or not even possible, e.g. mobile sensors worn by human users. However, even if wireless sensor and actuator networks can address properties that wired solution cannot, they suffer from a relatively low bandwidth, high delay, and a potentially high packet loss rate. The use of verbose SOA protocols further adds overhead. However, for low real-time sensor and actuator scenarios, the wireless approach offers interesting

properties.

- Using wired solutions, when determinism and high performance level are required. Wired solutions are mainly based, at the physical layer, on the use of Ethernet cabling solutions and of the corresponding switches. TCP/UDP standard communication protocols are used above, together with SOA protocol stacks such as DPWS or OPC-UA [1]. In such solutions, the architectures are no longer centralized, and the intelligence is distributed among all available resources as web services. However, in order to get accepted, the new SOA solution must provide comparable performance level to the one achieved using traditional field buses. Previous studies demonstrated that a DPWS stack can provide this high performance level, when using EXI [17] instead of standard text-based XML [18].

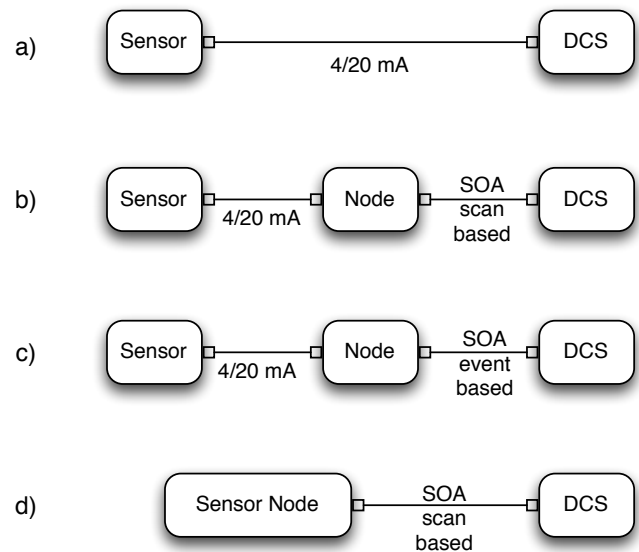


Fig. 6. Migration Strategies for a sensor.

In Fig. 6, a migration path is visualized. The path starts at (a) with a legacy system using proprietary or non-SOA protocols, often used on today’s shop floors. From this legacy architecture, one could add a new sensor node (wired or wireless) to the sensor in (b) using the legacy protocol. The new sensor node in turn acts as a mediator, transforming SOA messages into functionally feasible to and from upper layers, i.e. SCADA/DCS. It has to be pointed out that the type of communication between a sensor node and the DCS may have implications to cross layer control issues as indicated in Fig. 3.

It must be clarified that real-time SOA support is still an open issue, and that not all sensors and actuators will benefit from using it. However one has to consider a holistic view and weight the benefits for also other (not directly connected to a specific use-case) scenarios e.g. asset management. As a next step in (c), to mitigate the overhead of SOA protocols, the system could be event based. This reduces the network traffic

load. Finally, in (d) we envision that new devices will come with an integrated SOA interface, thus precluding the need of legacy system support.

An important part of a migration strategy is also training of staff on the new features and capabilities of the SOA-based infrastructure. This includes training of operators, maintenance staff, instrumentation staff, electricians staff, engineering staff etc.

V. CONCLUSION

We have introduced some major characteristics of a migration approach from legacy industrial process monitoring and control systems to a next generation of SOA-based process monitoring and control systems. The proposed migration strategy comprises top down and bottom up functional views and their implications. Furthermore some aspects on hardware migration to ensure legacy to SOA integration is presented.

Despite of the necessity to search for more general aspects of the introduced migration approach, issues that need to be investigated further include robustness of a process monitoring and control SOA-based infrastructure, real-time properties for closed loop control, security issues, deployment of large-scale installations, large system stability, evolution of new services and related architecture. The use of SOA protocols may add overhead in the communication layer which impacts i.e. real-time performance. The use of security, e.g. WS-Security, adds additional overhead in terms of memory, processing, and communication. Finally, the performance impact of verbose message exchange in large-scale sensor networks must be investigated. An important aspect of the migration is the consideration of the inter-relations among migration paths (depicted in Fig. 3), because these inter-relations will be one of the major enablers for emergent behaviours in the new migrated SOA-based systems of systems [16].

ACKNOWLEDGMENT

The authors would like to thank for their support the European Commission, and the partners of the EU FP7 project IMC-AESOP (www.imc-aesop.eu) for the fruitful discussions.

REFERENCES

- [1] W. Mahnke, S. H. Leitner, and M. Damm, "OPC Unified Architecture", Springer, May 2009. ISBN 3540688986.
- [2] S. Karnouskos, A. W. Colombo, F. Jammes, J. Delsing, and T. Bange-mann, "Towards an architecture for service-oriented process monitoring and control", in 36th Annual Conference of the IEEE Industrial Electronics Society (IECON), Phoenix, AZ, USA, 7–10 November 2010.
- [3] S. Karnouskos, A. W. Colombo, "Architecting the Next Generation of Service-based SCADA/DCS System of Systems", in 37th Annual Conference of the IEEE Industrial Electronics Society (IECON 2011), Melbourne, Australia, 7-10 Nov 2011.
- [4] H. Bohn, A. Bobek, and F. Golatowski, "SIRENA – service infrastructure for real-time embedded networked devices: A service oriented framework for different domains" in Proc. of the International Conference on Networking, Systems, Mobile Communications and Learning Technologies, IEEE Computer Society, 2006.
- [5] S. de Deugd, R. Carroll, K. E. Kelly, B. Millett, and J. Ricker, "SODA: Service Oriented Device Architecture", *Pervasive Computing*, IEEE, vol. 5, no. 3, pp. 94-96, Jul-Sep. 2006.
- [6] A. W. Colombo and S. Karnouskos, "Towards the factory of the future: service-oriented cross-layer infrastructure", in *ICT Shaping the World: A Scientific View*. European Telecommunications Standards Institute (ETSI), John Wiley and Sons, 2009, vol. 65-81.
- [7] S. Karnouskos, O. Baecker, L. M. S. de Souza, and P. Spiess, "Integration of SOA-ready networked embedded devices in enterprise systems via a cross-layered web service infrastructure", in *Proceedings of 12th IEEE International Conference on Emerging Technologies and Factory Automation, ETFA 2007*, Patras, Greece, Sep. 25–28, 2007.
- [8] S. Karnouskos, D. Savio, P. Spiess, D. Guinard, V. Trifa, and O. Baecker, "Real World Service Interaction with Enterprise Systems in Dynamic Manufacturing Environments", in *Artificial Intelligence Techniques for Networked Manufacturing Enterprises Management*, L. Benyoucef and B. Grabot, Eds., Springer, 2010, ISBN 978-1-84996-118-9.
- [9] A. W. Colombo, "Steps towards the flexible factory of the future", *eStrategies: Projects Europe*, Annual Edition, The British Publisher, pp. 18–21, 2009.
- [10] P. J. Marron, S. Karnouskos, D. Minder, A. Ollero, "The emerging domain of Cooperating Objects", Springer, 2011.
- [11] A. W. Colombo, S. Karnouskos, and J. M. Mendes, "Factory of the Future: A Service-Oriented System of Modular, Dynamic Reconfigurable and Collaborative Systems", in *Artificial Intelligence Techniques for Networked Manufacturing Enterprises Management*, L. Benyoucef and B. Grabot, Eds., Springer, 2010, ISBN 978-1-84996-118-9.
- [12] P. Kennedy, V. Bapat, and P. Kurchina, "In Pursuit of the Perfect Plant", *Evolved Technologist Press*, 2008.
- [13] Purdue Enterprise Reference Architecture (PERA), <http://pera.net/>.
- [14] M. Jamshidi, Ed. "Systems of Systems Engineering", CRC Press, November 2008.
- [15] O. Björk, "Nyttorealiser 2010", Technical Report, 2010, VSAP AB & Svenskt Projektforum: <http://tinyurl.com/nyttorealiser> (in Swedish)
- [16] S. Simanta, E. Morris, G. Lewis, D. Smith, "Engineering Lessons for Systems of Systems Learned from Service-Oriented Systems", in *Proc. of IEEE Systems Conference, 4th Annual IEEE*, 2010.
- [17] Efficient XML Interchange (EXI) Format 1.0, <http://www.w3.org/TR/2011/REC-exi-20110310/>
- [18] R. Kyusakov, J. Eliasson, and J. Delsing, "Efficient structured data processing for web service enabled shop floor devices", in *Industrial Electronics (ISIE)*, 2011 IEEE International Symposium on, June 2011, pp. 1716-1721.