Available online at www.sciencedirect.com

ScienceDirect

Procedia CIRP 33 (2015) 29 – 34

www.elsevier.com/locate/procedia

9th CIRP Conference on Intelligent Computation in Manufacturing Engineering - CIRP ICME '14

The Cyber Physical Implementation of Cloud Manufactuirng Monitoring Systems

Jeff Morgan *, Garret. E. O'Donnell

Mechanical and Manufacturing Engineering Department, Trinity College Dublin, Dublin 2, Ireland

* Corresponding author. *E-mail address*: MORGANJE@TCD.ie

Abstract

The rise of the industrial internet has been envisaged as a key catalyst for creating the intelligent manufacturing plant of the future through enabling open data distribution for cloud manufacturing. The context supporting these systems has been defined by Service Oriented Architectures (SOA) that facilitate data resource and computational functions as services available on a network. SOA has been at the forefront EU research over the past decade and several industrially implemented SOA technologies exist on the manufacturing floor. However it is still unclear whether SOA can meet the multi-layered requirements present within state-of-the-art manufacturing Cyber Physical Systems (CPS). The focus of this research is to identify the capability of SOA to be implemented at different execution layers present in a manufacturing CPS. The state-of-the-art for manufacturing CPS is represented by the ISA-95 standard and is correlated with different temporal analysis scales, and manufacturing computational requirements. Manufacturing computational requirements are identified through a review of open and closed loop machine control orientations, and continuous and discrete control methods. Finally the Acquire Recognise Cluster (ARC) SOA for reconfigurable manufacturing process monitoring systems is reviewed, to provide a topological view of data flow within a field level manufacturing SOA.

© 2014 The Authors. Published by Elsevier B.V. This is an open access article under the CC BY-NC-ND license
(http://creativecommons.org/licenses/by-nc-nd/4.0/).

Selection and peer-review under responsibility of the International Scientific Committee of "9th CIRP ICME Conference"

Keywords: Manufacturing; Monitoring; Service Orient Architecture

1. Introduction

Performance measurement is indispensable to manufacturing due to the fact that if the effective efficiency of an activity cannot be measured, it could not be effectively controlled [1]. Currently process and condition based monitoring systems are undergoing a transformation from static centralisation to dynamic decentralisation through the incorporation of Service Oriented Architectures (SOA) [2]. This transformation has been invoked from the convergence of the global industrial systems with the power of advanced computing, low-cost sensing and new levels of connectivity permitted by network technology, i.e. the industrial internet [3].

A SOA is a set of architectural tenets for building autonomous yet interoperable systems through loosely coupled Web Service (WS) [4]. SOA offers flexibility to process monitoring and control system, to meet the requirements of the heterogeneous manufacturing shop-floor, enabling adaption to manufacturing plants based on dynamic production plans, and facilitating interoperability through networked services. These envisaged advancements in data interoperability are aimed at offering the necessary system wide visibility of manufacturing data in complex collaborative automation systems, creating more open, flexible and agile environments [5]. Subsequently manufactures can now take a holistic approach to process monitoring with SOA enabled reconfigurable sensor fusion technology [6].

Research into SOA for manufacturing monitoring and control systems over the past decade has identified a means and a model to achieve factory wide interoperability, through the EU funded projects; Service Infrastructure for Real time

2212-8271 © 2014 The Authors. Published by Elsevier B.V. This is an open access article under the CC BY-NC-ND license
(http://creativecommons.org/licenses/by-nc-nd/4.0/).

Selection and peer-review under responsibility of the International Scientific Committee of "9th CIRP ICME Conference"
doi:10.1016/j.procir.2015.06.007

Embedded Networked Applications (SIRENA) [7], Service Oriented Cross layer Infrastructure for Distributed smart Embedded devices (SOCRADES) [8], ArchitecturE for Service Oriented Process Monitoring and Control (AESOP) [9]. However the substantial manufacturing SOA research contributions has given focus to the emergence of network technologies with little consideration given to the requirements of the process industry at lower field levels [10]. Current SOA requirements within manufacturing process monitoring and control have identified a need for research to be focused on how field level devices should integrate into higher-level systems to address seamless integration and transparent systems [10].

The focus of this research is to identify the capability of SOA to be implemented at different execution layers present in a manufacturing CPS. The cyber physical data processing and computational requirements of a manufacturing execution system are reviewed to identify systemic requirements for computation and communication. Additionally a topological view of data flow present in the Acquire Recognise Cluster (ARC) field level SOA for reconfigurable process monitoring systems is presented.

2. Manufacturing Cyber Physical Data Computation

2.1. Manufacturing Enterprise SOA Adoption

A manufacturing plant is an amalgamation of computational applications and physical devices, collaborating together to form complex CPS [11]. The ISA-95 international standard provide a layered hierarchical structure to provide structural context around manufacturing execution systems. The ISA-95 enterprise architecture defines models and terminology to determine which information has to be exchanged between systems for sales, finance and logistics and systems for production, maintenance and quality [12]. There are 5 levels in the ISA-95 model [13]; Level 0: is the production process itself, Level 1: is associated with all sensing and manipulating elements within the production process, Level 2: addresses monitoring, supervisory control and automatic control of the production process, Level 3: incorporates the management of the workflow to produce the desired end-products, maintaining records and optimising the production process, Level 4 aims at establishing the basic plant production schedule, material use, delivery and shipping, and inventory.

The ISA-95 model helps define boundaries between the different industrial enterprise levels. However the ISA-95 structure has been seen as rigidly hierarchical by limiting the capacity of cross layer interoperability due to highly integrated vendor-locked communication standards [13]. These standards exist due to the different functional control requirements present at each enterprise level. When data is acquired from a source the information is abstracted the further it is passed up the enterprise hierarchy, to meet different management requirements. This observation was made evident through Vijayaraghavan and Dornfelds work that identified the presence of different temporal analysis scales on different levels of manufacturing control [14], Fig.1. Each temporal level requires different computation and communication capabilities to provide for the control operating characteristics and data analysis. Subsequently a question is how can SOA be implemented across a manufacturing enterprise to enable more interoperable systems while maintaining the critical functional requirements at each level?

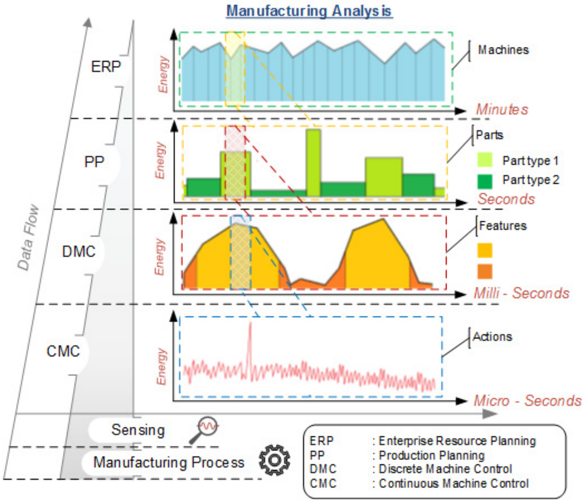


Fig. 1. Temporal Scales of Analysis and Manufactuirng execution systems, adapted from [14]

In 2012 Delsing et al. [13] proposed a migration procedure for ISA-95 Decentralised Control Systems (DCS) into enabled SOA systems. This work identified the presence of supporting service driven informatic systems at level 4 for Enterprise Resource Planning (ERP) and level 3 for production management. A key challenge identified in this work was for SOA to be adopted Real-Time (RT) control system execution must be preserved. This challenges the capability of SOA to meet the requirements of level 2 of a manufacturing enterprise which is associated with monitoring, supervisory control and autonomous control of the production process. To understand what role SOA can play to either meet the requirements of or coexist with machine control systems, a computational review of manufacturing control must be achieved to identify the functional requirements at this level.

2.2. Machine Control and Computation

Traditionally Manufacturing control systems utilise either closed or open loop control to regulate the operating characteristics of a system. Closed loop control utilises the feeding back of the measured system output to the system controller input allowing a control error to be determined and corrective action to be applied to reach the desired output [15]. Open loop control uses desired system output and potentially other measured disturbance inputs to reach the desired system output. A manufacturing enterprise utilises a mix of these methods as closed loop control cannot always be implemented due to the incapacity to measure system output on a continuous bases or in RT.

Additional to these control models different control methods are utilised; Continuous Control (CC) and Discrete Control (DC). CC maintains continuous response relationships between input and output, e.g. adaptive motion control of a CNC axis through a Proportional Integral Derivative (PID) controller [16]. Alternatively DC can exhibit multiple modes of operation and maintain discrete relationships within the system through discrete transitions between feedback measurements, or control adaption, e.g. the multifunctional control of machine events or scheduling of a production process [17]. Traditionally CC operate on a micro perspective of control systems relying on analogue controllers. However the speed, flexibility, accuracy, and reliability of digital controllers has exponentially increased over the past 20 years, uniquely offering greater advantages over analogue controllers, allowing for discrete controllers to achieve CC operations [18].

CC can be achieved by digital controllers through RT operating characteristics. RT systems can be characterised by achieving operation within defined jitter limits [19], and control latency as close to zero as possible to achieve just-in-time execution to minimise the disturbance input and control error which could lead to the degradation of system stability [15]. A divergence in RT definition can be identified through systems operating at RT speeds deterministically, namely Hard-Real Time (H-RT) and through systems operating at RT speeds non-deterministically, namely Soft-Real Time (S-RT) [19]. RT systems aim to achieve operational actions as close to RT as possible. In high performance motion control systems RT is obtained under 1 millisecond [20]. RT systems are designed to optimise computation through streamlining programming for high speed execution. Multiple operation states can be achieved with RT programming, as tasks are queued for computation via the CPU.

Other RT systems utilised within high performance computation control are Field-Programmable Gate Arrays (FPGA). FPGA's are reprogrammable silicon chips that consist of prebuilt logic blocks and programmable routing resources. FPGA's exceed the computing power of digital signal processors by breaking the paradigm of sequential execution and accomplish execution per clock cycle by hard coding operation directly to the processor [21].

Both RT CPU's and FPGA's are dedicated processing units that enable the creation of highly optimal systems, for both CC and DC systems, through reconfigurable programming means. However the function environment of these high performance units is static. When achieving deterministic execution all programming code needs to be specified and compiled together, to enable the optimal performance and execution time limits to be determined. Dynamic execution environments can only be achieved through undedicated processing units, where computation tasks are queued for execution by the CPU. These systems can operate at high computation speeds of under 1 millisecond in S-RT. However this flexibility comes at a price as the undedicated systems are not deterministic, and cannot achieve H-RT operation due to their ambiguity of execution from sharing of resources in a dynamic execution environment.

A topological view of manufacturing control computation systems identifies a separation in technology from RT deterministic static systems, to dynamic interoperable open systems Fig. 2. This separation can be identified by the divergence in systematic requirements from deterministic high speed performance, to multifunctional flexible network capable features. A system requires a greater flexibility when dealing with a dynamic environment. However to achieve guaranteed H-RT control systems must be optimised for reaction speed.

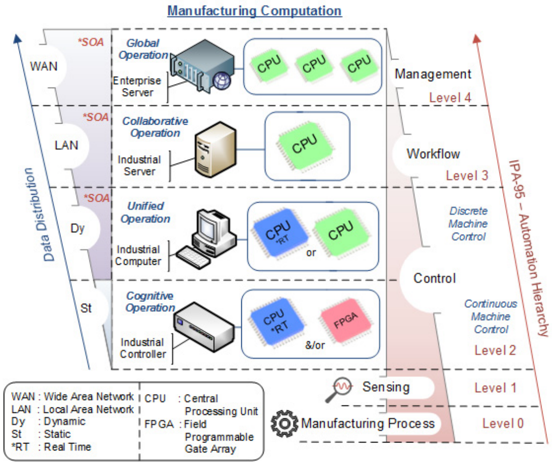


Fig. 2. Manufacturing execution control computation

The transitioning of data throughout a manufacturing hierarchy operates in a highly dynamic environment. Large amounts of varying sized data is required by multiple end users at different locations within a plant. Handling this data traffic requires greater undedicated computation capable units. The need for flexibility is greater than the need for H-RT reaction, as the requirements of the system has a high dependence on change, due to the ambiguity and diversity present within the end users requirements and connected technology. Achieving H-RT would require each system to be finalised and orchestrated for optimisation. This is an unrealistic goal for manufacturing management systems due the complexity and ever evolving nature of manufacturing systems.

2.3. SOA Implementation and Communication

SOA specifies that distributed resources and organisations should provide their functionalities in the form of services that requesters can have access to [22]. An entity or service can be discovered dynamically through asynchronous messaging by exposing its interface [23]. These characteristics meet the criteria of dynamic execution systems and require a dynamic communication medium for support. SOA originated from Web technology of Ethernet TCP/UPD, which enable the loose connectivity of hundreds or thousands of devices. However ultimately the use of asynchronous time-division multiplexed networking introduces time varying delays which are sources of potential instability for RT targets [24].

Subsequently this incapacity to utilise deterministic communication mediums has identified a incapability of SOA to meet the requirements of deterministic CC and DC present in level 2 of the ISA-95. Other solutions to meet these requirements can be seen with Profibus DP [25] and EtherNet/IP [20]. SOA may not be a primary interoperability member at the lowest point of computation in a manufacturing execution system. However these systems should be enabled to either provide their data for higher levels systems directly or indirectly from communicating their data to a mediator or orchestrator.

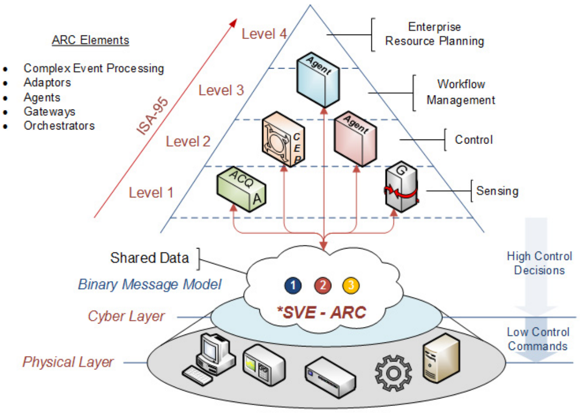
Traditional implementation of SOA within manufacturing systems identified the use of WS for communication protocol, e.g. HTTP in MTConnect [26] and DPWS in AESOP [27]. However these medium utilise a XML base message structure. XML was identified to not meet the high speed requirements of a for industrial machinery applications due to its verbose syntax and the need for parsing which can slow down processing speed and cause RT constraints [28]. Limitations with raditional XML structuring can be overcome through adoption of the Efficient XML Interchange (EXI), which utilises binary representation of data and is designed for compactness and high performance parsing and serialisation [29]. Examples of a binary representation model to achieve SOA high speed communication with TCP <1ms, was demonstrated by Morgan and O'Donnell [30]. Subsequently the introduction of binary messaging has identified a means for nondeterministic DC present in level 2 of the ISA-95 [30] [28].

The time restrictions present within level 2 of the ISA-95 are not present in the above levels due to the abstracting of data in the higher temporal scales of analysis in manufacturing execution systems. These systems favour flexibility and open connectivity rather than high speed deterministic behaviour. Due to the fact that latency within the millisecond range will not destabilise systems operating within a >1sec scale of temporal operation, e.g. scheduling, resource management, production planning. Subsequently traditional or high speed WS technology can be utilised to achieve SOA at levels 3 and 4 in the ISA-95.

3. Acquire Recognise Cluster a field level SOA

3.1. Topology

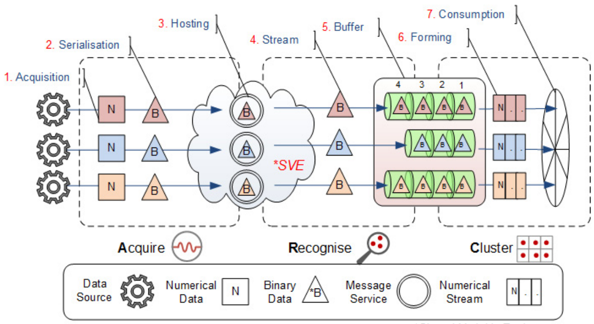
The ARC-SOA for reconfigurable process monitoring systems is a field level informatics system designed to meet the requirements of S-RT analysis, DC, and decision support on the manufacturing floor. The ARC-SOA utilises a dencentralised architecture with the National Instruments – Shared Variable Engine (SVE) for common platform interoperability [31], Fig. 3. The SVE is a communication interoperability medium and act as an orchestrator unit to represent data sources as services on a network. The SVE utilises the NI-Publish Subscribe Protocol (NI-PSP) that operates on Ethernet TCP/IP with use of the LogosXT transmission algorithm [32]. The NI-PSP enables the pulling and pushing of data within the network, as data variables can be referenced on request or subscribed to for event driven data acquisition. Ultimately the ARC-SOA enables the facilitation of process variables within a cloud that is accessible by multiple end users dynamically, to meet the varying manufacturing domain requirements present in process control, optimisation, management, and maintenance. More information on the ARC-SOA can be referenced in [30].



*Shared Variable Engine – Acquire Recognise Cluster

Fig. 3. Acquire Recognise Cluster Topology, adapted from [30].

The ARC-SOA can be reviewed in 7 steps as data flows from source to service and finally to consumer. These steps are characterised by name of the architecture; Acquire, Recognise, and Cluster (ARC), Fig. 4. The software elements in this example include a data adaptor for data sourcing, the SVE for data interoperability, and a data client agent for data consumption.



*Shared Variable Engine

Fig. 4. Acquire Recognise Cluster sequential data flow

1. Acquisition: data is collected via an adaptor software element and time-stamped; 2. Serialisation: data is serialised to a Binary Message Model (BMM) format [30], data can range from single values to thousands of values in an array which is bundled into a singular message packet; 3. Hosting: packets are published to and hosted by the SVE for internal and network wide data distribution; 4. Stream: data is acquired and streamed dynamically from the SVE by the data client agent which can be present locally on the same computer or remotely across a network; 5. Buffer: data is acquired from multiple source that arrive at different times

and are then loaded into a designated buffers for processing; 6. Forming: data buffers are emptied cyclically where message packages are deserialised, data streams are united, and further buffered for consumption; 7 Consumption: processed data streams are consumed depending on the functionality of the agent.

3.2. Bench Marking

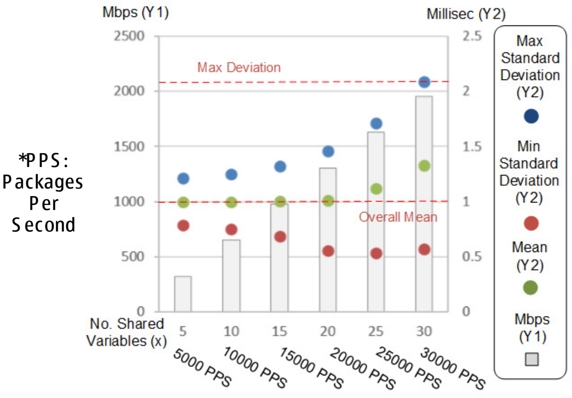


Fig. 5 Local communication time bench marking

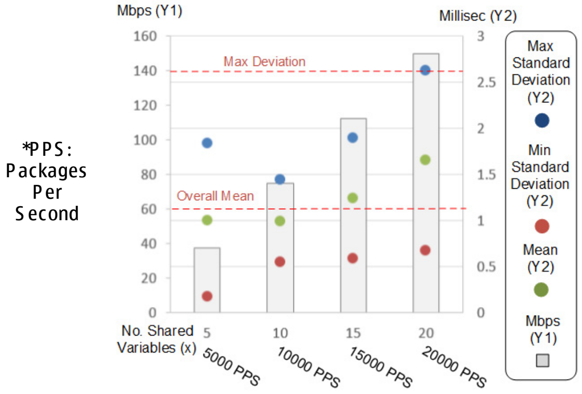


Fig. 6 Network communication time bench marking

Communication benchmarking was undertaken to identify the capability of the ARC-SOA to meet the high speed and high data throughput requirements present within S-RT DC at level 2 of a manufacturing execution system. Two experiments were undertaken to provide a contrasting view of how fast data can be communicated through the SOA reliably. Data was transmitted from a software adapter to a client agent via the SVE being hosted by a singular computer, Fig.5, and between two computers networked together via a 100 Mbps network switch, Fig.6. Network communication speed was identified through the measurement of message arrival times in the agent. The adaptor was set to communicate 1 message or package of set data size every 1 millisecond, which is 1000 Packages Per Second (PPS) per variable. The mean communication time was determined through 3 repeated tests. Additionally the quantity of variables being transmitted was incremented between test sets. Samples per package were varied between local and network tests, as the network tests were limited by 100 Mbps on the network. Local communication time experiments were undertaken with a 1000 samples per package with double point precision variables resulting in 8.134 kB per package. Network communication time experiments were undertaken with a 100 samples per package with double point precision variables resulting in 0.934 kB per package.

The ARC-SOA bench marking has identified; a local maximum capacity to maintain 20 M Hz of data across 20 shared variables, with a standard deviation of 0.32 milliseconds, at 1301.4 Mbps; And a network maximum capacity to maintain 1 M Hz of data across 10 shared variables, with a standard deviation of 0.83 milliseconds, at 74.2 Mbps. Results after these two set points in both experiments identify an increase in mean communication time above the 1 millisecond range, indicating an incapacity to maintain the throughput targets. Additionally faster and more reliable results could be achieved within network tests through increasing the connection speed >100Mbps, which is the bottle neck in the system's capacity.

It is important to note that a larger number of variables can be utilised depending on the systems requirements, meaning 5 to 30 variables is not the maximum limit. However message communication times will increase with increasing message sizes and message traffic. Additionally decreasing message sizes will enable a higher PPS throughput capacity, and vice versa.

4. Conclusion

This work investigated whether SOA could be integrated into a cyber-physical manufacturing execution system to enable cloud monitoring. The results identified a separation in systematic computation and functional control requirements at different CPS layers. This separation was identified as a divergence in functionality between enabling deterministic high speed performance, and enabling multifunctional flexible network capability. Subsequently SOA was identified to not meet the deterministic requirements of H-RT present in supervisory control, automatic control and monitoring systems. However SOA was identified to meet the requirements of all higher level manufacturing CPS layers due to the reduced time constraints present.

Additionally, to provide context around SOA implementation the ARC-SOA was reviewed, with focus given to mapping data flow and defining communication capacity locally and across a network. The results identified that the ARC-SOA could meet the high speed and data volume requirements present in S-RT DC systems.

In summary, SOA may not currently be able to meet the requirements of H-RT systems, however its implementation for nondeterministic S-RT DC can enable the creation of advanced algorithms and mechanisms which once perfected can be refined to exist on RT targets with deterministic execution. Ultimately the dynamic nature of SOA within manufacturing execution systems will create a new environmental to facilitate the innovation of the future through borderless data interoperability.

Acknowledgements

This research was funded under the Graduate Research Education Programme in Engineering (GREP-Eng) which is a PRTLI Cycle 5 funded programme and is co-funded under the European Regional Development Fund. The author would like to thank his work colleagues at Trinity College Dublin for their assistance in collating this publication.

References

[1] K. K. . Hon, "Performance and Evaluation of Manufacturing Systems," *CIRP Ann. - Manuf. Technol.*, vol. 54, no. 2, pp. 139–154, 2005.

[2] M. Riedl, H. Zipper, T. Bangemann, and M. Meier, "Distributed automation system supports process monitoring and control," *Industrial Electronics (ISIE), 2011 IEEE International Symposium on*. pp. 1711–1715, 2011.

[3] P. C. Evans and M. Annunziata, "Industrial internet: Pushing the Boundaries of Minds and Machines," 2012.

[4] J. Cassina, C. Alessandro, and M. Taisch, "Development of an Extended Product Lifecycle Management through Service Oriented Architecture," *Industrial Product-Service Systems CIRP IPS2*. Cranfield University, pp. 118–124, 2009.

[5] A. W. Colombo, F. Jammes, H. Smit, R. Harrison, J. L. M. Lastra, and I. M. Delamer, "Service-oriented architectures for collaborative automation," in *Industrial Electronics Society, 2005. IECON 2005. 31st Annual Conference of IEEE*, 2005, pp. 2649–2659.

[6] R. Teti, K. Jemielniak, G. O'Donnell, and D. Dornfeld, "Advanced monitoring of machining operations," *CIRP Ann. - Manuf. Technol.*, vol. 59, pp. 717–739, 2010.

[7] H. Bohn, A. Bobek, and F. Golatowski, "SIRENA - Service Infrastructure for Real-time Embedded Networked Devices: A service oriented framework for different domains," in *Networking, International Conference on Systems and International Conference on Mobile Communications and Learning Technologies, 2006. ICN/ICONS/MCL 2006. International Conference on*, 2006, p. 43.

[8] A. Cannata, M. Gerosa, and M. Taisch, "SOCRADES: A framework for developing intelligent systems in manufacturing," *Industrial Engineering and Engineering Management, 2008. IEEM 2008. IEEE International Conference on*. pp. 1904–1908, 2008.

[9] S. Karnouskos, A. W. Colombo, F. Jammes, J. Delsing, and T. Bangemann, "Towards an architecture for service-oriented process monitoring and control," *IECON 2010 - 36th Annual Conference on IEEE Industrial Electronics Society*. pp. 1385–1391, 2010.

[10] Processit.eu, "European Roadmap for Industrial Process Automation," 2013.

[11] S. Karnouskos, "Cyber-Physical Systems in the SmartGrid," *Industrial Informatics (INDIN), 2011 9th IEEE International Conference on*. pp. 20–23, 2011.

[12] ISA-European-Office, "Technology ISA-95," 2010. [Online]. Available: http://www.isa-95.com/subpages/technology/isa-95.php. [Accessed: 13-Nov-2013].

[13] J. Delsing, F. Rosenqvist, O. Carlsson, A. W. Colombo, and T. Bangemann, "Migration of industrial process control systems into service oriented architecture," *IECON 2012 - 38th Annual Conference on IEEE Industrial Electronics Society*. pp. 5786–5792, 2012.

[14] A. Vijayaraghavan and D. Dornfeld, "Automated energy monitoring of machine tools," *CIRP Ann. - Manuf. Technol.*, vol. 59, no. 1, pp. 21–24, 2010.

[15] J. Hellerstein, Y. Diao, S. Parekh, and D. Tilbury, *Feedback Control of Computing Systems*. New Jersey: John Wiley & Sons Inc, 2004, p. 429.

[16] Q. Zhong, Y. Shi, J. Mo, and S. Huang, "A Linear Cross-Coupled Control System for High-Speed Machining," *Int. J. Adv. Manuf. Technol.*, vol. 19, no. 8, pp. 558–563, 2002.

[17] L. Monostori, B. C. Csáji, B. Kádár, A. Pfeiffer, E. Ilie-Zudor, Z. Kemény, and M. Szathmári, "Towards adaptive and digital manufacturing," *Annu. Rev. Control*, vol. 34, no. 1, pp. 118–128, 2010.

[18] M. S. Fadali and A. Visioli, *Digital Control Engineering Analysis and Design*. Elsevier, 2013, p. 582.

[19] National-Instruments, "What is a Real-Time Operating System (RTOS)?," 2013.

[20] P. Didier, F. Macias, J. Harstad, R. Antholine, S. A. Johnston, S. Piyevsky, M. Schillace, G. Wilcox, D. Zaniewski, and S. Zuponcic, "Converged Plantwide Ethernet (CPwE) Design and Implementation Guide," 2011.

[21] National-Instruments, "Introduction to FPGA Technology: Top 5 Benefits," 2012.

[22] J. M. Mendes, P. J. Leitão, F. J. Restivo, A. W. Colombo, and A. Bepperling, "Engineering of service oriented automation systems: a survey," *Innovative Production Machines and Systems*. Cardiff University, pp. 33–38, 2008.

[23] F. Jammes, A. Mensch, and H. Smit, "Service-Oriented Device Communications Using the Devices Profile for Web services," in *Advanced Information Networking and Applications Workshops, 2007, AINAW '07. 21st International Conference on*, 2007, vol. 1, pp. 947–955.

[24] Y. Halevi and R. Asok, "Integrated Communication and Control Systems: Part I-Analysis," *J. Dyn. Syst. Meas. Control*, vol. 110, pp. 367 – 373, 1988.

[25] Acromag Inc, "Introduction to Profibus DP," 2002.

[26] MTConnect, "Getting Started with MTConnect - Connectivity Guide," 2011.

[27] F. Jammes, B. Bony, P. Nappey, A. W. Colombo, J. Delsing, J. Eliasson, R. Kyusakov, S. Karnouskos, P. Stluka, and M. Till, "Technologies for SOA-based distributed large scale process monitoring and control systems," *IECON 2012 - 38th Annual Conference on IEEE Industrial Electronics Society*. pp. 5799–5804, 2012.

[28] G. Starke, T. Kunkel, and D. Hahn, "Flexible collaboration and control of heterogeneous mechatronic devices and systems by means of an event-driven, SOA-based automation concept," *Industrial Technology (ICIT), 2013 IEEE International Conference on*. pp. 1982–1987, 2013.

[29] R. Kyusakov, H. Makitaavola, J. Delsing, and J. Eliasson, "Efficient XML Interchange in factory automation systems," *IECON 2011 - 37th Annual Conference on IEEE Industrial Electronics Society*. pp. 4478–4483, 2011.

[30] J. Morgan and G. E. O'Donnell, "A Service Oriented Reconfigurable Process Monitoring System – Enabling cyber physical Systems," *J. Mach. Eng.*, vol. 14, 2014.

[31] National-Instruments, "Network Variable Technical Overview," 2009.

[32] National-Instruments, "Using the LabVIEW Shared Variable," 2012.