Embedded service oriented monitoring, diagnostics and control: towards the asset-aware and self-recovery factory

Axel Vidales Ramos[1], Ivan M. Delamer[2] and Jose L. M. Lastra[1]
Tampere University of Technology[1] and Inico Technologies Ltd.[2]
axel.vidales@tut.fi, delamer@ieee.org and jose.lastra@tut.fi

***Abstract*-In Europe, the manufacturing sector represents nearly the 22% of the GDP. Since the beginning of the European Technology Platform MANUFUTURE, Service Oriented Architecture have become the *de facto* in the industrial informatics research community. In most cases, SOA is implemented by using Web Service technology based on RESTful services or WS-* standards. Several publications have shown the advantages of WS technologies in Factory Automation e.g. enterprise integration, monitoring and control, real-time decision taking, maintenance scheduling, and reconfiguration. Monitoring has become an effortless task, leading to massive data to be processed in order to present a diagnostic i.e. asset-awareness. Further, once problems are identified, an automatic self-recovery mechanism should be provided in order to minimize loses. This paper introduce the eSONIA project which aims to improve the flow of information, rather than raw data, in the factory floor by enhancing the system awareness and granting self-recovery to the factory. To achieve this goal, eSONIA uses pervasive heterogeneous embedded devices intercommunicated with WS technologies over IPv6.**

I. INTRODUCTION

European manufacturing industry represents 22% GDP and maintenance cost can be up to 60% of the production cost, depending on the area [1]. Maintenance costs needs to be reduced in order to improve the cost effectiveness of industries. The main problem of the factories is the fact that they are not well understood (e.g. WMS, LIMS) [2]. In order to reduce the cost of maintenance and improve the manufacturing sector the factories need to be understood in a holistic and continuous approach, such as the Evolvable Production System [3]. Service Oriented Architecture (SOA) has been recognized by the Information and Communication Technologies (ICT) as an enabler of sustainable manufacturing [4-5]. Additionally, SOA facilitates the task of factory integration by exposing services to all enterprise levels. SOA is usually realized with Web Service (WS) technology either using RESTful services or WS-* standards (also called SOAP WS) [6], although there are other alternatives e.g. OPC Unified Architecture [7]. With the advent of SOAs and the use of WS technology the task of monitoring and control of locally distributed heterogeneous devices have become an effortless task; although, in the case of geographically distributed networks the communication is still a challenge [8]. Network challenges are expected to be diminished by the use of IPv6. Nowadays, several raw data is collected; although, it is unusually processed in order to provide meaningful information. To alleviate this problem, this paper presents the embedded Service Oriented moNItoring, diagnostics and control: towards the Asset-aware and self-recovery factory (eSONIA) [9] project primarily targets to provide, for first time, Monitor / Diagnostic / Prognostic / Control (MDPC) to the factory and expose this information to all enterprise levels. The eSONIA project relies on pervasive heterogeneous IPv6 embedded devices which collect data (monitoring). The data are processed providing asset health assessment (diagnostic), maintenance scheduling (prognostic), and real-time optimization (control) in order to solve an important manufacturing problem, *factory obfuscation*. This includes efficient 3D visualization for indoors and outdoors, production track & trace, real-time route planning and optimum maintenance performance. This project relates to other European projects which work inline this goal, such as iLand.

This paper is divided as follows: section II presents background research in the areas of SOA, asset-awareness, and self-recovery. Section III discusses the different Internet Protocols. Section IV explains the architectural view of eSONIA, in order to provide MDPC. Section V shows a proof of concept of the eSONIA architecture. And finally, last section presents the conclusions.

II. BACKGROUND

This section debriefs the last ten years of research work in the area of Factory Automation using SOA as a middleware, and specifically, using WS technology. Furthermore, monitoring applications towards asset-aware and self-recovery, and IP optimizations for embedded devices are mentioned.

A. *Service Oriented Architecture*

The European Technology Platform MANUFUTURE [10] has as main objective to "develop and implement a strategy based on Research and Innovation, capable of speeding up the rate of industrial transformation to high-added-value products, processes and services, securing high-skills employment and winning a major share of world Manufacturing output in the future knowledge-driven economy." Consequentially, several European research projects have been created working together to reach this goal. Service Infrastructure for Real-Time Embedded Networked Applications (SIRENA) [11] project presents the Service-Oriented paradigm for industrial automation. SIRENA merges the factory floor and the enterprise levels by using WS technology on top of the Internet Protocol (IP); therefore, it reduces the networking costs. Following the same path, Service Oriented Cross-Layered InfRAstructure for Distributed smart Embedded deviceS (SOCRADES) [12] project designed a framework in order to monitor and control a SOA factory. SOCRADES defined methodologies for WS description [13], which are further used to allow effortless monitoring. Within the work, it is provided a level of data processing used towards control optimization. Although, SOCRADES creates a framework for data monitoring and control it has still many challenges [14], such as data aggregation, awareness and self-recovery, Wireless Sensor Networks (WSN). There many other projects such as InLife which uses DPWS, for monitoring and diagnostic, directly in the devices.

Moreover, in the past few years, IP standards have been gained momentum; hence the formation of the Internet Protocol for Smart Objects (IPSO) Alliance. IPSO alliance advocates for IP pervasive network devices for use in industry, energy, healthcare and consumer applications. Moreover, IPSO works closely in the generation of standards related with WS technologies. These standards focus on WSs for embedded devices (e.g. 6LoWPAN, EXI, CoAP) [15].

B. *Asset-aware and Self-Recovery*

In recent years, the asset-awareness and self-recovery techniques have gain importance. The research topics focus on embedded asset-awareness using semantics, 3D visualization, and monitoring. The research of [16] clarifies the inteLLigent Accountability Middleware Architecture (LLAMA) for embedded context-aware devices. In order to provide context- awareness the described architecture implements the Context Agent and Context Manager. Context Agents act as monitoring proxies and interact with the Context Manager which reports the system status to other architecture entities. Moreover, Karray et al. [17] explains the importance of using semantic mediation in order to exchange information between different systems, especially those of maintenance. The research community has also focused its efforts on using Device Profile for Web Services (DPWS) [18], for monitoring purposes. For example, in the area of 3D localization Feldhorst et al. [19] presents a 3D visualization tracking system. A WSN is also interconnected by using an Application Layer Gateway by [20]. Hilbrich [21] studies the performance of WSD in a Body Area Network, confirming that the most of the processing time is within the nodes (and not the communication channel). Other approaches involve the Evolvable Manufacturing Systems [3] which empathise the adaptability of the factory floor.

III. INTERNET PROTOCOL

IP was developed by the US Defence Advanced Research Project Agency (DARPA) for its packet switched network (ARPANET). IP is situated in the Network Layer of the ISO/OSI 7-Layer model positioned in the third layer. Despite IPv6 being a successor of IPv4 there is no compatibility between them. Moreover, due to the fact of IPv4 being widely deployed a co-existence of both Internet Protocols has been considered [22].

A. *Internet Protocol version 4*

IPv4 has been around since 1960 [23]. It is ubiquitous used by human end-users and, on the last decade, machine-to-machine communication. IPv4 has an address space of 2^{32} (almost 4.3 billion) to provide end-to-end communication. This address space is not enough even to provide a single IP address for every human being (approximately 6.79 billion). Nowadays, it has been completely allocated and its exhaustion is predicted for June of 2011. In order to alleviate the address depletion problem, several solutions have been proposed in the past, such as Network Address Translation (NAT) [24] and Application Layer Gateways (ALG) [25].

B. *Internet Protocol version 6*

IPv6 is the latest IP specification, defined in 1998 [26]. IPv6 simplifies the header of its predecessor and increases the size of source and destination addresses to 128 bits. IPv6 introduces a huge address space of 2^{128} (3.4×10^{38}) or 3.142×10^{11} addresses per cm^3 in the entire world. IPv6 improves response time, maintenance and security [21] by simplifying the end-to-end communication, as originally intended by the Internet. The response time is reduced by the omission of NATs and other ALG, given address space and the maintenance is improved by its Autoconfiguration features. Moreover, 6LoWPAN provides header compression given IPv6's simplicity [27].

C. *IPv4 vs. IPv6*

IPv6 excels IPv4 in many areas, in theory, although, in practice, the implementations of IPv4 are much more robust. The aspects of the protocol are briefly compared: accessibility, scalability, bandwidth, mobility, response time, and security. Accessibility and scalability are the main reasons for the definition of IPv6. There are sufficient addresses for IPv6 devices to access them globally without the need of gateways or NATs, unlike IPv4. In the basic form IPv4 has better utilization (bandwidth), due to its small address size compared to IPv6. Although when using header compression techniques (e.g. 6LoWPAN), which are only available on IPv6, the roles inverted (6LoWPAN address can be 16 bits long). Mobility is provided for both protocols, although if they need to be accessed from globally, IPv6 has better availability.

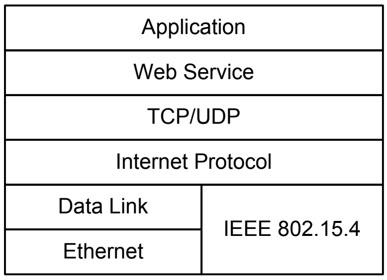

Fig. 1. Communication Stack

Response time is decreased in IPv6 due to the elimination of NATs and gateways. However, nowadays many implementations of IPv4 exhibit better response times than those of IPv6. Finally, the elimination of intermediaries allows the implementation of end-to-end security on IPv6 (e.g. IPSec [28]).

IV. System Architecture

In order to visualize the factories is required a loose couple architectural model, such as SOA, to ease integration and information flow. The eSONIA project uses Web Service technology to intercommunicate its heterogeneous pervasive devices. The proposed stack for embedded devices and, generally, every entity in the architecture is presented in Fig. 1. The layers describe the general protocols and standards used, although other technologies are also involved (e.g. Real-Time Ethernet, IPv6, 6LoWPAN, WS-* and RESTful WS). Hence in this paper, devices compliant with the presented stack are referred as Web Service Devices (WSD). The proposed stack provides support for wired and wireless WSD. Wired technology works on IPv6 on top of Ethernet while wireless uses 6LoWPAN on top of IEEE 802.15.4 standard.

Fig. 2 presents a use case of eSONIA where the lower part (below the Ethernet link) represents the factory, and the upper depicts the higher levels of the architecture. It is important to realize that there is not a hierarchy in the communication, this is a flat architecture i.e. any entity can exchange data with any other. To achieve this goal, every device is abstracted as a WS. Furthermore, entities could be located within (same site) or outside (different sites) the network and can be addressed by means of IPv6. Despite the fact that there are WSD for every factory requirement there is still the need of

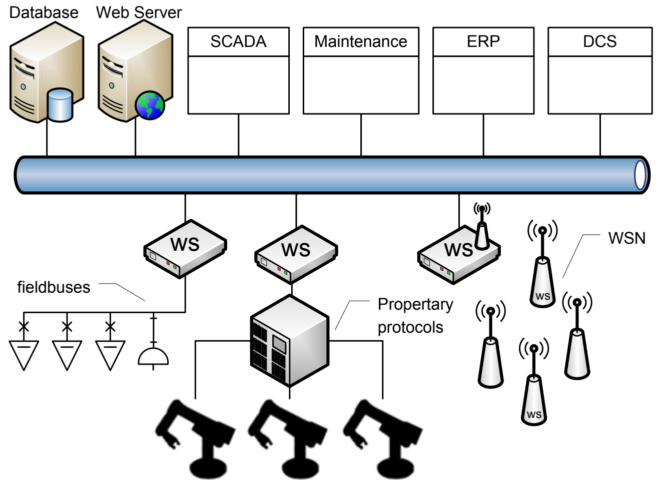

Fig. 2. Key architectural components

abstracting legacy systems to WS for different reasons (e.g. economical, expertise, temporal). In factories, there exist three cases of WS abstraction: WSD handling fieldbuses and/or IOs, WSD communicating with proprietary or legacy devices, and WSD in charge of WSN, as depict in Fig. 2. WSDs handling directly fieldbuses and/or IOs are the desired situation where every control unit is able to communicate directly with any other WSD. Therefore, high-level entities can tune production parameters and monitor factory data without the need of intermediates. The second case is the commonly undesired situation where the WS device acts as a gateway to proprietary protocols or legacy systems. Gateways are normally undesired because they increase the latency and maintenance efforts, while becoming the bottle neck, the single point of failure, and compromising the end-to-end security. On the other hand, gateways are broadly used on the industry; therefore, they cannot be left aside.

Finally, WSD handling WSN (see Fig. 2) can act as Edge Routers (ER) or gateways. When using IP compatible protocols on the wireless mesh (e.g. 6LoWPAN), the WSD can act as an ER. In the case of 6LoWPAN, the ER is responsible for compress/decompress IPv6 header in both directions. In this scenario, upper layer protocols are accessed just by the end points, therefore end-to-end security can be applied. In the case of using non-IP protocols (e.g. WirelessHART, ZigBee) the WSD can still provide communication by acting as a gateway.

Normally, higher enterprise levels do not have resource limitations in contrast with lower levels. As a result, most systems are Web Service compliant, and have been used already for several years in the business interactions. The eSONIA project abstracts high-level systems, such as Supervisory Control And Data Acquisition (SCADA), Maintenance, Enterprise Resource Planning (ERP), Distributed Control System (DCS), as WSDs. The architecture of eSONIA is divided in modules to achieve the aims of MDPC.

V. TEST BED

The test bed used to present the eSONIA concept in public events is the demo box illustrated in Fig. 3 which consist of a set of Smart Remote Terminal Units (Smart RTU) equipped with Ethernet and glued though a Service Oriented middleware over IPv6, a wireless sensor, an analog and digital IO panel, and an energy monitor.

The demo exhibits link between high level communication, via WS technology, and the low level IO logic. Additionally, wireless devices are exposed as WSs collecting information from the environment (e.g. temperature, humidity, luminosity) and health of assets (e.g. battery). The exposed data can be monitored using WS-Subscription (DPWS) by the Supervisory Control and Data Acquisition System (SCADA) over a secure connection. Smart RTU uses a Web Server for configuration and monitoring, therefore, no additional software needs to be installed. In configuration mode, a common browser can be used to modify the business logic (e.g. structured text) and the Web Service interface.

Fig. 4 presents the logical view of the demo box. The box is a portable demo factory which is communicated to a SCADA system, contained on a mobile computer, using (and exposing) DPWS technology on IPv6. The test bed demonstrates the development in the following areas: embedded DPWS communication over IPv6, and techniques for asset-awareness and self-recovery.

A. *DPWS over IPv6*

The Smart RTU uses DPWS technology over IPv6. This is the result of our collaborative work in IPv6 for industrial controllers, within the framework of the eSONIA project. The IPv6 stack has been

Fig. 3. Physical image of the demo box

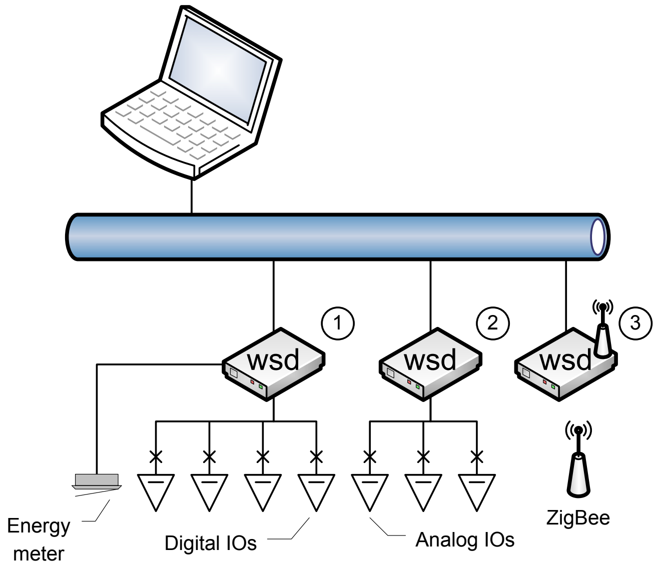

Fig. 4. Logical view of the demo box

accepted by the Lightweight IP (LwIP) open source community, to be included in the next planned release [29]. The external computer contains a SCADA system and a database. The SCADA system discover the devices in the IPv6 network by either receiving a hello message from a WSD or by sending a probe and obtaining probe matches, as described in WS-Discovery. Once a device is discovered, the user defines the relevant information to be monitored, either for the type of service or the specific device. This information is stored in a formal representation within the SCADA system. Once device start operating, events are generated, and received notifications are stored in the database for further processing.

B. Properties of asset-aware and self-recovery

The most important property of asset-awareness is availability i.e. if a certain service is accessible. The demo factory, contained in the box, uses a heartbeat mechanism to assure availability. A device is considered unavailable in the following two cases: a bye message is received, or a consecutive number of heartbeats are missing. Once a device is reported as unavailable, the device is removed and, moreover, notifies is subscribers about its removal. Wireless nodes also provide a health assessment by providing information about the environment and themselves.

The implemented self-recovery mechanism is applied when a device that has stop working is made available again. By using the formal information collected when the device was introduced to the system (e.g. relevant subscriptions) it is possible to recover the system to its intended state, without human intervention. The system remembers and uses this information to renew its subscriptions.

VI. CONCLUSIONS

The eSONIA project contributes for a better understanding of factories by propagating information across all enterprise levels, while providing asset-awareness and self-recovery mechanisms. In order to allow the effortless flow of information, IPv6 is pushed to every device. IPv6 makes it easier to interconnect devices geographically distributed and ease the maintenance of the network. Additionally, allow header compression for WSN. Integration of different devices at various enterprise levels is studied by using WS technology on top of IPv6. WSD are seamless interconnected to the architecture, while legacy devices and non-IP technologies are easily integrated through the use of gateways. A technique for asset availability is provided. And it is further enhanced by a self-recovering mechanism in the case of reintegration.

Also, the status of a wireless sensor is provided i.e. asset-health. These described techniques can be used in different automation domains, even outside of the intended. For example, waste control is indirectly targeted by reducing maintenance, therefore reducing the amount of spare parts and other expendables. Or the energy reduction in database server by reducing the amount of data stored by aggregating it already in the factory floor. Although various techniques are presented, further work is still required in the areas of asset-awareness, self-recovery, and maintenance scheduling.

VII. FURTHER WORK

This paper is a positioning paper for the eSONIA project and further papers will expand its various technological achievements. The main research areas of eSONIA include the following topics: IPv6 in embedded devices, data aggregation, efficient KPI calculation, asset-awareness and self-recovery.

ACKNOWLEDGMENT

The research leading to these results has received funding from the ARTEMIS Second Call 2009 Programme under Grant Agreement n° 100223, correspondent to the project shortly entitled eSONIA Embedded Service Oriented Monitoring, Diagnostics and Control: Towards the Asset-aware and Self-Recovery Factory.

REFERENCES

[1] R.K. Mobley, An introduction to predictive maintenance, Butterworth-Heinemann, 2002.

[2] C. Lam, K. Choy, and S. Chung, “Framework to measure the performance of warehouse operations efficiency,” Industrial Informatics (INDIN), 2010 8th IEEE International Conference on, 2010, pp. 634-639.

[3] Ribeiro, L., J. Barata, et al. (2010). Emergent Diagnosis for Evolvable Production Systems. International Symposium on Industrial Electronics. Bari, IEEE.

[4] Ribeiro, L., J. Barata, et al. (2010). An Agent-Based Interaction-Oriented Shop Floor to Support Emergent Diagnosis. IEEE International Conference on Industrial Informatics. Osaka, Japan, IEEE

[5] Kwei-Jay Lin and M. Panahi, “A real-time service-oriented framework to support sustainable cyber-physical systems,” Industrial Informatics (INDIN), 2010 8th IEEE International Conference on, 2010, pp. 15-21.

[6] G. Moritz, E. Zeeb, S. Prüter, F. Golatowski, D. Timmermann, and R. Stoll, “Devices Profile for Web Services and the REST,” *Industrial Informatics (INDIN), 2010 8th IEEE International Conference on*, 2010, pp. 584-591.

[7] G. Candido, F. Jammes, J. de Oliveira, and A. Colombo, “SOA at device level in the industrial domain: Assessment of OPC UA and DPWS specifications,” Industrial Informatics (INDIN), 2010 8th IEEE International Conference on, 2010, pp. 598-603.

[8] R. Messerschmidt, C. Diedrich, and P. Neumann, “Integrated web services and tunneling enabling field bus communication via heterogeneous (wide area) networks,” *Industrial Informatics (INDIN), 2010 8th IEEE International Conference on*, 2010, pp. 543-548.

[9] Embedded Service Oriented Monitoring, Diagnostics and Control: Towards the Asset-Aware and Self-Recovery Factory (eSONIA) www.esonia.eu

[10] European Commission: MANUFUTURE Platform, http://www.manufuture.org

[11] F. Jammes and H. Smit, “Service-oriented paradigms in industrial automation,” Industrial Informatics, IEEE Transactions on, vol. 1, 2005, pp. 62-70.

[12] 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, 2008, pp. 1904-1908.

[13] V. Herrera, A. Lee, and J. Lastra, “A Methodology for identifying Web Services in electronic manufacturing systems,” Industrial Informatics (INDIN), 2010 8th IEEE International Conference on, 2010, pp. 571-577.

[14] A. Cannata, S. Karnouskos, and M. Taisch, “Evaluating the potential of a service oriented infrastructure for the factory of the future,” Industrial Informatics (INDIN), 2010 8th IEEE International Conference on, 2010, pp. 592-597.

[15] Z. Shelby, “Embedded web services,” Wireless Communications, IEEE, vol. 17, 2010, pp. 52-57.

[16] Jinhwan Lee and Kwei-Jay Lin, “A context management framework for real-time SOA,” Industrial Informatics (INDIN), 2010 8th IEEE International Conference on, 2010, pp. 559-564.

[17] M. Karray, B. Chebel-Morello, and N. Zerhouni, “A contextual semantic mediator for a distributed cooperative maintenance platform,” Industrial Informatics (INDIN), 2010 8th IEEE International Conference on, 2010, pp. 181-188.

[18] Devices Profile for Web Services http://schemas.xmlsoap.org/ws/2006/02/devprof

[19] S. Feldhorst, M. Fiedler, M. Heinemann, M. ten Hompel, and H. Krumm, “Event-based 3D-monitoring of material flow systems in real-time,” Industrial Informatics (INDIN), 2010 8th IEEE International Conference on, 2010, pp. 195-200.

[20] A. Sleman and R. Moeller, “Integration of Wireless Sensor Network Services into other Home and Industrial networks; using Device Profile for Web Services (DPWS),” Information and Communication Technologies: From Theory to Applications, 2008. ICTTA 2008. 3rd International Conference on, 2008, pp. 1-5.

[21] R. Hilbrich, “An Evaluation of the Performance of DPWS on Embedded Devices in a Body Area Network,” Advanced Information Networking and Applications Workshops (WAINA), 2010 IEEE 24th International Conference on, 2010, pp. 520-525.

[22] H. Miyata and M. Endo, “Design and evaluation of a fieldbus protocol over IPv6,” in Industrial Informatics (INDIN), 2010 8th IEEE International Conference on, 2010, pp. 136–141.

[23] J. Postel et al., Internet protocol. STD 5, RFC 791, September 1981, 1981.

[24] K. Egevang and P. Francis, “RFC 1631: The IP network address translator (NAT),” Status: INFORMATIONAL, May 1994.

[25] H. Miyata, K. Miyazawa, Y. Akisada, and M. Endo, “The analysis of IPv6 adaptation to the industrial network protocol,” in Industrial Informatics, 2009. INDIN 2009. 7th IEEE International Conference on, 2009, pp. 614–619.

[26] S. Deering, R. Hinden et al., Internet protocol, version 6 (IPv6) specification. RFC 2460, Dec. 1998.

[27] Z. Shelby and C. Bormann, 6LoWPAN: The Wireless Embedded Internet. Wiley-Blackwell, Nov. 2009.

[28] K. Seo and S. Kent, “Security architecture for the internet protocol,” http://tools.ietf.org/html/rfc4301, Dec. 2005.

[29] Lightweight TCP/IP Protocol suite https://savannah.nongnu.org/projects/lwip/