Applying Semantic Web Services and Wireless Sensor Networks for System Integration

Gian Ricardo Berkenbrock, Celso Massaki Hirata,
Frederico Guilherme Álvares de Oliveira Júnior,
and José Maria Parente de Oliveira

Instituto Tecnológico de Aeronáutica, S. J. Campos, Brazil
{gian,hirata,fred,parente}@ita.br

Abstract. In environments like factories, buildings, and homes automation services tend to often change during their lifetime. Changes are concerned to business rules, process optimization, cost reduction, and so on. It is important to provide a smooth and straightforward way to deal with these changes so that could be handled in a faster and low cost manner. Some prominent solutions use the flexibility of Wireless Sensor Networks and the meaningful description of Semantic Web Services to provide service integration. In this work, we give an overview of current solutions for machinery integration that combine both technologies as well as a discussion about some perspectives and open issues when applying Wireless Sensor Networks and Semantic Web Services for automation services integration.

Keywords: Wireless Sensor Networks, Integration, Semantic Web-Services.

1 Introduction

Nowadays changes happen in a faster way. Companies always try to predict when such changes will happen. Buildings, factories, and homes are built to serve for different purposes during their lifetime. And for each purpose we have a distinct requirements set. From this perspective, automation services will change during their lifetime in such environments. Changes are concerned to business rules, process optimization, cost reduction, and so on. And some common changes are re-layout of production line to make a new product or to optimize the process, factory modernization, production lines adjustment for customized products, adaptation to a new layout on a building floor, adjustment for new security and safety rules, owner changes, and so on.

Hence a smooth and straightforward way to deal with machinery automation integration could be using a combining approach of wireless sensor networks and semantic web services. Combining both technologies we could reach an integration among machineries, and between machineries and third-party software systems. The term third-party software is used for software systems that are not deployed in sensor nodes.

M. Ulieru, P. Palensky, and R. Doursat (Eds.): IT Revolutions 2008, LNICST 11, pp. 161–170, 2009.

Wireless sensor networks (WSN) are composed of variety number of small, low cost, and dispensable devices called as sensor nodes or just nodes and these nodes can communicate over-the-air to each other using an embedded radio. The network size can change from a few nodes to hundreds of thousand nodes. WSN have some constraints such as low processing capability, limited energy, low transmission rate, and short transmission range. Due to these constraints WSN are very application-oriented. WSN communication can be single or multi hop. Then to increase communication capability and efficiency many protocols were proposed.

Semantic Web Services (SWS) are services with semantic ontology-based annotations. Contrary to syntactic web services, where parameters are basically described with data types, semantic web services parameters are associated with concepts described by ontologies allowing discovery, composition, execution and monitoring automation of services. By composing existing simple services, it is possible to build dynamic applications in order to perform more complex task to meet requirements. For instance, a complex query (task) that needs to obtain information from several sensors (services).

Therefore, applying together WSN and SWS could give us a smooth way to deliver integration for an automation system placed with different devices and systems. This approach can facilitate both an integration in an inside context for WSN and in an integration role between WSN and third-party systems. The remainder of this paper is organized as follow. Section 2 gives you a background information for the discussions in Section 4. Section 3 presents how SWS can be used in WSN. Section 4 also points out some open issues and Section 5 gives the final remarks.

2 Background

2.1 Wireless Sensor Networks

Wireless sensor networks (WSN) are composed of variety number of small, low cost, and dispensable devices called as sensor nodes or just nodes and these nodes can communicate over-the-air to each other using an embedded radio. Nodes have sensing capability and usually are applied to monitor some phenomenon. Nodes also can be deployed very close to some phenomenon or inside it. In order to deliver the collected data to base station, it is common that nodes communicate in an ad-hoc fashion until the data is delivered to the base station. WSN components like nodes, field (deployed area), base station (BS), third-party system, and the relationships among them are shown in Figure 1. In WSN the network size can change from a few to hundreds of thousand nodes.

WSN have some constraints such as low processing capability, limited energy, low transmission rate, and short transmission range [1]. Due to these constraints WSN are application-oriented. WSN can be used for military applications, environment monitoring, factory instrumentation, clean-room monitoring, etc. Currently, the most used operational system in WSN is TinyOS [2]. TinyOS is an event-based OS developed by Berkeley University for its sensor nodes.

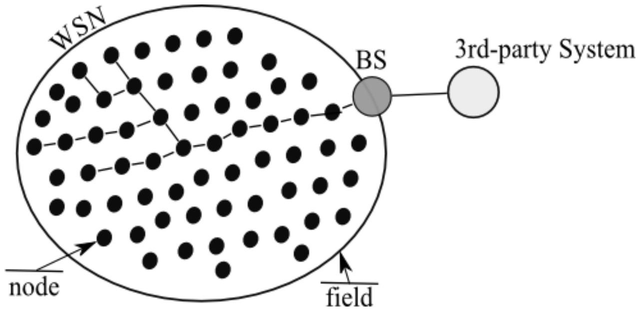

Fig. 1. Components of Wireless Sensor Network

WSN communication can be single or multi hop. So, in order to increase the communication capability and efficiency many protocols were proposed. Each protocol concentrates its effort on a specific service or network layer. Thus different protocols are used for medium access control (B-MAC [3], IEEE 802.15.4 [4]), routing (directed diffusion [5]), data aggregation (Collection Tree Protocol [6]) and dissemination (Location-Aided Flooding [7]), middleware (MANNA [8], TinyDB [9]), and so on. Considering protocols effort, a highlight is the recent approach to adapt TCP/IP stack for WSN. For example, 6loWPAN [10], μIP [11], and a new alliance called IP for Smart Objects (IPSO)[12]. Besides being able to communicate through a wireless communication, WSNs operate on a low-consumption energy [13].

Moreover, after an industry decision on standards about IP adaptation, a large amount of services might emerge and then integration services can be dealt in a easy way. These communication standards could give a step forward in a WSN distinct vendors environment working together properly. For example, electronics from different vendors that are use in ordinary homes could exchange data and then the owner could have a detailed information about energy consumption of those electronics. In addition, semantic web services can have an important role in this new environment.

2.2 Semantic Web Service

A web service can be defined as a piece of software that conforms to a set of open interoperability standards [14] such as Web Service Description Language (WSDL), Simple Object Access Protocol (SOAP), and Universal Description, Discover and Integration (UDDI), for description, messaging protocol and discovering, respectively.

In service-oriented architecture, applications can be accomplished by composing simple services in order to perform a given task. There are several approaches for automating services composition such as Business Process Execution Language (BPEL) [15]. However, most of those approaches give just a syntactic description of the offered service allowing only a predefined service composition. In summary, syntactic services requires human involvement.

On the other hand, Semantic Web Services aim at automating the discovery, composition, invoking and monitoring of Web Services [16] by providing an ontology-based description. The main advantage of semantic web services against syntactic ones is the possibility to associate a parameter to a concept rather than just to a data type. While a syntactic service can only define a parameter isbn of the type String for instance, a semantic service allows the association of this parameter with a International Standard Book Numbers (ISBN) semantic model (e. g. ontology or taxonomy) so that this parameter is able to expect values related to other concepts which has some semantic relationship with ISBN concepts.

In this context, automatic composition consists of dynamic discovery and combine existing services in a registry, given a set of input and output concepts. The matching process should take into account some criteria to determine if two concepts can be considered as the same.

Regarding to dynamic discovery and composition, Paolucci [17] has developed a Matchmaking algorithm that compares advertised services' parameters with request's ones. Kaufer et al. [18] proposed an hybrid service matchmaking based on both logic-based reasoning and matching based on syntactic information retrieval based similarity computation. Sirin et al. [19] proposed a semi-automatic approach for composing semantic web services which means that it is given to the user a set of services whose inputs matches to the output of the previously selected by him. Weise et al. [20] proposed a genetic approach based on evolutionary algorithms.

There are many ways to describe a service in a semantical manner. Some languages like Semantic Annotations for WSDL and XML Schema (SAWSDL) allow description of additional semantics of WSDL components [21]. In summary, they specify how to associate semantic models to a service's description. Another way to describe a service is by modeling the whole service as an ontology. In this sense, OWL-S (Semantic Markup for Web Services) [22] provide a top level ontology of services as shown in Figure 2. This ontology is written in the ontology language OWL (Web Ontology Language) and it is divided into three parts:

- Service Profile: tells "what the service does". It describes the type of information, i. e., concepts involved in a service-seeking process in order to determine whether the service meets its needs.
- Service Model: tells "how the service works". It describes what happen when a service is executing, that is, specifies the details regarding to the order in which and the constraints under which steps that compose the process must be executed.
- Service Grounding: it specifies the details of how an user/agent can access a service, that is, it specifies a communication protocol, message formats, and other service-specific details such as port numbers used in contacting the service.

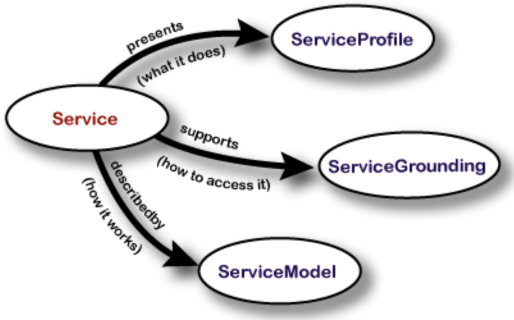

Fig. 2. Top level of the service ontology [22]

3 Applying Semantic Web Services to Wireless Sensor Networks

Sensor Networks is applied in order to provide data from a monitored area using different sources. Those data can be aggregated in order to characterize contexts of different goals. To explore the contexts it could be necessary a system that makes use of data from one or more contexts as well as data processed by other systems. One type of system that can support such processing are Web Services, since they can provide means to facilitate the interoperability with other services. More complex processing can require several service composition schemas, that can include nesting and coordination among services.

As mentioned before in Section 2.1, there are already technologies (e. g. [10,11,12]) that allow to run a minimal webserver over a node, supported by current version of ContikiOS [25]. Then using SWS and WSN together brings us advantages and disadvantages. On the one hand, we can get a flexible system with automatic service composition that could automatic integrate to others system that use same protocol to exchange information. On the other hand, use these technologies together implies an overhead over the traditional WSN. This overhead could be minimized by management an application-dependent trade-off between integration level and what WSN need to perform its goal. Some WSN performance parameter can be network lifetime, throughput, storage size, extra-local processing, and etc.

Thus, in this combined environment, each node can be seen as a service provider and consumer. It can provide a set of services for delivering data and for node's parameters management (network management, operating system, sensing devices, and other application specific values). And it can use some service provided by some node in neighborhood range. Moreover, this service set is dynamic. Nodes can provide or not some service using local or non-local information like remain battery, local data collected, neighborhood data, and so on.

There are at least four different approaches that one can use to deploy SWS in WSN. First approach is when each node has its own service description and implementation, and it does not depend on other non-local services to perform its

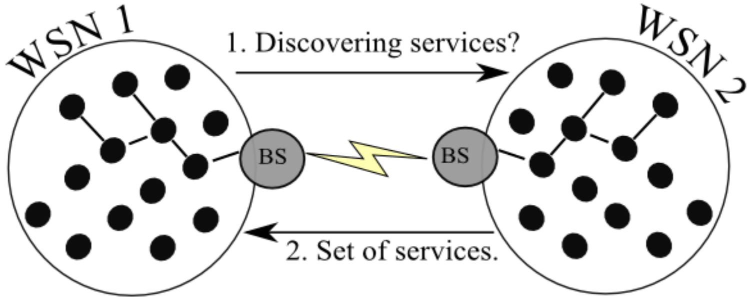

Fig. 3. An example of communication between two WSN with SWS enabled

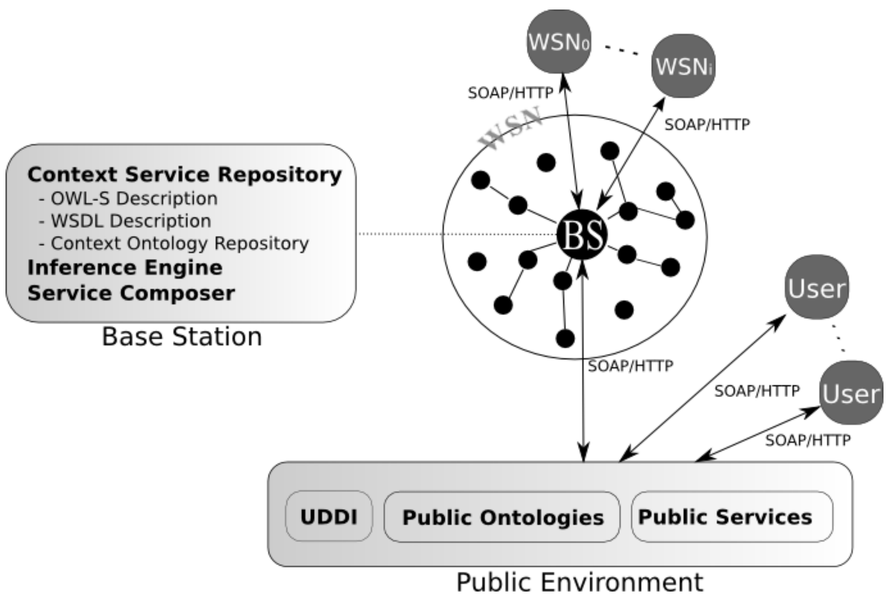

Fig. 4. An SWS option use for Wireless Sensor Network

tasks. The second one is when all related web services tasks are deployed in the base-station and the WSN only provides the data requested. The third approach has its web services deployment in a third-party system. The third-party system receives all the requested data from the base-station. And the last approach is the hybrid one, when all related parts of web services is spread over at least two of these components - nodes, base-station, and third-party system (see Figure 1). In Figure 3 it is shown a communication example between two WSN with SWS enabled in their respective base-station. And in Figure 4 it is shown a more complex scenario where it has WSN exchange information with others one and also it has users using public information through a third-party system.

Therefore, to reach such potential in technology integration a development of ontologies for WSN and SWS is need. These ontologies should meet data and service description. There are few ontologies available, e. g., [23] and [24]. Nevertheless, it is vital a more detailed ontology to help services and applications to create their derived ontology, if necessary, that can preserve the communication capability to each other.

In addition, currently there are different middleware for WSN approaches such as event-based, database-inspired, and others [26]. The middleware aim is to provide network management services, data collection, support for application development, sensing task management, etc. Moreover, they are device, operating system, communication data specific. Then for these middlewares to play an integration and interoperability roles might be difficult and might be need extra effort to achieve them. But to achieve these roles in a given SWS system can play a decisive part. Currently approaches using WSN and SWS are available, e. g., [27,28,29,30,31,32,33].

4 Discussion

Currently, some automation systems make use of wired sensors to perform factory monitoring. Nevertheless, environments that use wired sensors are less flexible for changes than ones that use Wireless Sensors. The main advantage of WSN is the capability to perform wireless communication. For this reason, WSN are more flexible and adjustable. But integration and interoperability in WSN are not consolidated yet.

Many approaches that apply Semantic Web Services to WSN have already been proposed as mentioned in Section 3. However, these approaches are intended to solve problems related only to information gathering and knowledge inference about data collected from a certain environment.

From the point of view of system integration, those approaches are not suitable, since they does not deal with important aspects such as interoperability and description of sensor devices. These are key aspects facilitate the way devices are accessed and managed. Moreover, once devices are described in a semantic manner, it is easy to adapt them according to environment changes. Typical environments are factories, buildings, and homes.

In such environment types, changes can occur on demand or by special needs. Furthermore, these environments are built to serve for different purposes during their lifetime. And for each purpose there is a distinct requirements set. Example of environment changes are re-layout of production line to make a new product or to optimize the process, factory modernization, production lines adjustment for customized products, adaptation to a new layout on a building floor, adjustment for new security and safety rules, owner changes, and so on.

As mentioned before, by using ontology-based service description it is possible to perform dynamic composition of services in way the system is adjusted within a given situation without human-involvement rather than using static and syntactic composition that are widely used in Business Process Management. Furthermore, by interfacing a sensor device, i. e,. by enabling it via (web) services, devices can communicate to each other in a interoperable manner.

Another advantage of applying SWS to WSN is the possibility to hierarchically organize service providers so that fine-grained services (sensor devices) are combined to provide more complex services, in turn are combined with other complex services in order to provide even more complex services.

On the other hand, further studies are needed to evaluate the impact of the overhead brought by high-level protocol message headers in the network performance and lifetime.

4.1 Open Issues

Using SWS with WSN still have issues to be researched, such as:

Standards: This issue plays an essential role to interoperability. It is need standards for communication (MAC, routing, topology control, etc), data representation, service description, service discovery, etc for WSN constraints. Today, we already have some efforts to address this issue, e. g., IEEE 802.15.4 [4], 6loWPAN [10], and ROLL [34].

SWS management inside WSN: SWS deal with service publishing, discovery, composition services, invoking, and monitoring. All these services must have quality of service, fault tolerance, security, dependability, etc then how to fit all of those in such constraint environment like WSN.

Software development process: Currently we have a lack of availability of software development process for WSN applications. Yet, it is need that such process must deal with WSN application development and integration to the whole third-party system.

5 Final Remarks

This paper presents a discussion about applying together wireless sensor networks (WSN) and semantic web services for integration to third-party systems and to others WSNs. This combination might give us a smooth way to deliver integration for an automation system placed with different devices and systems. This approach can facilitate an integration in an inside context for WSN and in an integration role between WSN and third-party systems.

The next step of this research will address the lack of software development process for WSN and the issues related to quality of service for automatic composition of semantic web services. This research will also deal with distinct alternatives to implement the SWS in real WSN with different constraints.

References

1. Akyildiz, I.F., Su, W., Sankarasubramaniam, Y., Cayirci, E.: A Survey on Sensor Networks. IEEE Communications Magazine 40, 102–111 (2002)
2. Hill, J., Szewczyk, R., Woo, A., Hollar, S., Culler, D., Pister, K.: System architecture directions for networked sensors. SIGPLAN Not. 35(11), 93–104 (2000)

3. Polastre, J., Culler, D.: Versatile Low Power Media Access for Wireless Sensor Networks. In: ACM Conf. Embedded Networked Sensor Sys., pp. 95–107 (2004)
4. IEEE Standard for Information technology- Telecommunications and information exchange between systems- Local and metropolitan area networks- Specific requirements Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (WPANs), IEEE Std 802.15.4-2006 (Revision of IEEE Std 802.15.4-2003) (2006)
5. Intanagonwiwat, C., Govindan, R., Estrin, D., Heidemann, J., Silva, F.: Directed Diffusion for Wireless Sensor Networking. IEEE/ACM Trans. Networking 11(1), 2–16 (2003)
6. TinyOS Network Protocol Working Group. TEP 123: The Collection Tree Protocol (2008), http://www.tinyos.net/tinyos-2.x/doc/txt/tep123.txt
7. Sabbineni, H., Chakrabarty, K.: Location-Aided Flooding: An Energy-Efficient Data Dissemination Protocol for Wireless Sensor Networks. IEEE Transactions on Computers 54(1), 36–46 (2005)
8. Ruiz, L.B., Nogueira, J.M.S., Loureiro, A.A.F.: MANNA: A Management Architecture for Wireless Sensor Networks. IEEE Communications Magazine 41(2), 116–125 (2003)
9. Madden, S., Franklin, M.J., Hellerstein, J.M., Hong, W.: TinyDB: An Acqusitional Query Processing System for Sensor Networks. In: ACM TODS (2005)
10. Montenegro, G., Kushalnagar, N., Hui, J., Culler, D.: Transmission of IPv6 Packets over IEEE 802.15.4 Networks. IETF RFC 4944 (2007), http://www.ietf.org/rfc/rfc4944.txt
11. Dunkels, A.: Full TCP/IP for 8-bit architectures. In: Proceedings of The First International Conference on Mobile Systems, Applications, and Services (ACM MobiSys 2003), San Francisco, USA (May 2003)
12. IP for Smart Objects (2008), http://www.ipso-alliance.org/
13. Estrin, D., Girod, L., Pottie, G., Srivastava, M.: Instrumenting the World with Wireless Sensor Networks. In: Proceedings of the International Conference on Acoustics, Speech and Signal Processing (May 2001)
14. Pulier, E., Taylor, H.: Understanding Enterprise SOA, Manning, Greenwich (2006)
15. Curbera, F., et al.: Business Process Execution Language for Web Services, v1.0. In: Thatte, S. (ed.) IBM (July 2001), http://www-106.ibm.com/developerworks/webservices/library/ws-bpel
16. Antoniou, G., van Harmelen, F.: A Semantic Web Primer. The MIT Press, Cambridge (2004)
17. Paolucci, M., et al.: Semantic matching of web services capabilities. In: Horrocks, I., Hendler, J. (eds.) ISWC 2002. LNCS, vol. 2342, pp. 333–347. Springer, Heidelberg (2002)
18. Kaufer, F., Klusch, M.: WSMO-MX: A Logic Programming Based Hybrid Service Matchmaker. In: Proceedings of the 4th IEEE European Conference on Web Services (ECOWS 2006). IEEE CS Press, Zurich (2004)
19. Sirin, E., et al.: Filtering and Selecting Semantic Web Services with Interactive Composition Techniques Language for Web. IEEE Intelligent Systems 19, 42–49 (2004)
20. Weise, T., et al.: Different Approaches to Semantic Web Service Composition. In: ICIW 08: Proceedings of the 2008 Third International Conference on Internet and Web Applications and Services, pp. 90–96. IEEE Computer Society, Washington (2008)
21. Farrell, J., Lausen, H. (eds.): Semantic Markup for Web Services (2008), http://www.w3.org/TR/sawsdl/

22. Martin, D., et al.: Semantic Annotations for WSDL and XML Schema (2008), http://www.w3.org/Submission/OWL-S/
23. Ota, N., Kramer, W.T.C.: TinyML: Meta-data for Wireless Networks, Technical Report, University of California, Berkeley, Computer Science Dept. (2003)
24. Eid, M., Liscano, R., EI Saddik, A.: A Novel Ontology for Sensor Networks Data. In: Proceedings of 2006 IEEE International Conference on Computational Intelligence for Measurement Systems and Applications, La Coruna-Spain, pp. 12–14 (2006)
25. Dunkels, A., Grnvall, B., Voigt, T.: Contiki - a Lightweight and Flexible Operating System for Tiny Networked Sensors. In: Proceedings of the First IEEE Workshop on Embedded Networked Sensors (Emnets-I) (2004)
26. Henricksen, K., Robinson, R.: A survey of middleware for sensor networks: state-of-the-art and future directions. In: Proceedings of the international workshop on Middleware for sensor networks, pp. 60–65 (2006)
27. Whitehouse, K., Zhao, F., Liu, J.: Semantic Streams: A Framework for Composable Semantic Interpretation of Sensor Data. In: Römer, K., Karl, H., Mattern, F. (eds.) EWSN 2006. LNCS, vol. 3868, pp. 5–20. Springer, Heidelberg (2006)
28. Aberer, K., Hauswirth, M., Salehi, A.: Infrastructure for Data Processing in Large-Scale Interconnected Sensor Networks. In: Proceedings of 2007 International Conference on Mobile Data Management, May 1, pp. 198–205 (2007)
29. Lastra, J.L.M., Delamer, M.: Semantic web services in factory automation: fundamental insights and research roadmap. IEEE Transactions on Industrial Informatics 2(1), 1–11 (2006)
30. Liu, J., Zhao, F.: Towards semantic services for sensor-rich information systems. In: Proceedings of 2nd International Conference on Broadband Networks, vol. 2, pp. 967–974 (2005)
31. Gracanin, D., Eltoweissy, M., Wadaa, A., DaSilva, L.A.: A service-centric model for wireless sensor networks. IEEE Journal on Selected Areas in Communications 23(6), 1159–1166 (2005)
32. Priyantha, B., Kansal, A., Goraczko, M., Zhao, F.: Tiny Web Services for Sensor Device Interoperability. In: Proceedings of International Conference on Information Processing in Sensor Networks (IPSN 2008), pp. 567–568 (2008)
33. Goodwin, C., Russomanno, D.J.: An Ontology-Based Sensor Network Prototype Environment. In: Proceedings of the 5th International Conference on Information Processing in Sensor Networks (2006)
34. Routing Over Low power and Lossy networks (2008), http://tools.ietf.org/wg/roll/