Implementation of a Service Oriented Architecture in Smart Sensor Systems Integration Platform

Alexander K. Alexandrov and Vladimir V. Monov
Institute of Information and Communication Technologies, Bulgarian Academy of Sciences
Acad. G. Bonchev str., bl. 2, 1113 Sofia, Bulgaria
akalexandrov@gmail.com, vmonov@iit.bas.bg

Keywords: SOA, Sensor networks, WSN, SSN, Data integration.

Abstract: The paper describes a new developed platform for smart sensor network (SSN) data integration based on Service Oriented Architecture (SOA). SOA is a software design and software architecture design pattern based on discrete pieces of software providing application functionality as services to other applications. This is known as service oriented technology which is independent of any vendor, product or technology. Our platform is layer based and defines Network Layer, Data Integration layer and Application layer. The proposed architecture improves the information flow, has ability to expose internal functionality, improves reliability and scale operations to meet different demand levels. The platform supports mechanisms for cooperative data mining, self organization, networking, and energy optimization to build higher level service structures. The development of applications is improved and simplified by the use of optional administration tools and components.

1 INTRODUCTION

Service oriented architecture (SOA) is a software design based on discrete pieces of software providing application functionality as services to other applications. This is known as Service orientation. It is independent of any vendor, product or technology (WWW Consortium, 2001).

A service is a self contained unit of functionality, such as retrieving an online bank statement (The Open Group, 2007). Services can be combined by other software applications to provide the complete functionality of a large software application (Kodali, 2005).

Each service implements one action, such as submitting an online application for an account, retrieving an online bank statement or modifying an online booking or airline ticket order. Within a SOA, services use defined protocols that describe how services pass and parse messages. The purpose of SOA is to allow users to combine together fairly large amount of functionality to form *ad hoc* applications.

SOA as an architecture relies on service orientation as its fundamental design principle. If a service presents a simple interface that abstracts away its underlying complexity, then users can access independent services without knowledge of the service's platform implementation .(Svetle, 2010).

The main benefit of SOA is to allow simultaneous use and easy mutual data exchange between applications of different vendors without additional programming or making changes to the services. These services are also reusable, resulting in lower development and maintenance costs and providing more value once the service is developed and tested. Having reusable services readily available also results in quicker time to market (Bacchanalina, Toggle, 2003).

Depending on the adopted approach, each SOA service is designed to perform one or more activities by implementing one or more service operations. As a result, each service is built as a discrete piece of code.

SOA also defines how to integrate widely disparate applications for a Web based environment and uses multiple implementation platforms. Rather than defining an API, SOA defines the interface in terms of protocols and functionality. An *endpoint* is the entry point for such a SOA implementation. For somedevelopers, SOA can be seen as modular programming, software as a service (SaaS), and

Alexandrov A. and V. Monov V.
Implementation of a Service Oriented Architecture in Smart Sensor Systems Integration Platform.
DOI: 10.5220/0005422101140120
In *Proceedings of the Third International Conference on Telecommunications and Remote Sensing (ICTRS 2014)*, pages 114-120
ISBN: 978-989-758-033-8

cloud computing (which some authors (Erl, 2007) see as the offspring of SOA).

One of the core characteristics of services developed using service orientation design paradigm is that they are composition centric. Services with this characteristic can potentially address novel requirements by recomposing the same services in different configurations. Service composition architecture is itself a composition of the individual architectures of the participating services.

In our study SOA makes it easy for smart sensors connected over a network or within a smart sensor system to succefully cooperate. Smart Sensor System (SSS) is a term describing system which integrate two or more smart sensor networks or part of them. Every smart sensor can run an arbitrary number of services, and each service is built in a way that ensures that the service can exchange information with any other service in the network without human interaction and without the need to make changes to the underlying program itself.

In this paper we propose a service-oriented, flexible and adaptable platform for sensor systems integration. Our approach allows high-level applications to easily configure the data-gathering level and exploit the available functionalities. In the remainder of the paper, some related work is briefly analyzed in Section 2, whilst Section 3 describes the architecture of the developed platform. Concluding remarks are given in Section 4.

2 RELATED WORK

The main goal of the proposed platform for sensor systems integration is the effective and seamless integration of pervasive technologies into the information system of networked enterprises. This issue has already been tackled in the literature, for example in (Samaras *et al.*, 2009) and (Delicato *et al.*, 2003). However, those two proposals are aimed at implementing a service oriented middleware directly on sensor nodes, by forcing SOA compatible protocol stacks (like (OASIS, 2009)) in resource constrained devices.

In our opinion, this approach has the major drawback of imposing too much complexity in devices that are not enough powerful to transmit and elaborate XML messages. To overcome such constraints, the authors propose to adopt *a priori* knowledge in XML message definition, thus losing middleware flexibility. Moreover, the usage of web services in resource constrained devices imposes a certain energy and latency overhead (as an example, cost for such implementations has been quantified in (Priyantha *et al.*, 2008)).

By contrast, our approach concentrates the logic of the Integrated Smart Sensor Networks(ISSN) on a powerful platform, to which the various heterogeneous sensor systems are connected. Such solution is not new, for example it has been used in (Gil-Martinez-Abarca *et al.*, 2006) for enabling management of remote bootstrap of network nodes through the Internet; and in (Grosky *et al.*, 2007) for building a peer to peer infrastructure for sharing sensors through the Internet. However, such works address a wide area of pertinence and they do not explicitly address typical WSN issues, like the energy management of nodes and the QoS support for applications.

A gateway based solution has been also proposed in (Moeller, Sleman, 2008), aiming at integrating WSNs into other existing IP based networks. However, as their work is addressed to the ambient intelligence at home, they only abstract the functionality of single sensors, i.e. applications are aware of the network deployment and request services directly to a node. Our approach instead abstracts functionalities of the various numbers of sensor networks, i.e. applications request services for a geographic area without the need to know how many nodes are deployed there or how they communicate to each other.

It is worth to note that our approach completely differs from that of querying systems like TinyDB (Madden *et al.*, 2011). In fact, such systems permits to extract data from a WSN but they do not generally provide high level interfaces for QoS configuration and management. Moreover, such systems usually exploit low level techniques for gathering data and can thus be considered as tight extensions of a particular WSN technology. For this reason, they could in turn be used for developing a WSN whose configuration and management are provided by our architecture, that is, as explained in the next section, independent by design of the underlying WSN technology.

3 SOA BASED PLATFORM FOR SMART SENSOR SYSTEMS INTEGRATION.

The services of the proposed integrated sensor systems platform are Apache and WSDL based and implement service oriented architecture. They have some functional building blocks accessible over standard Internet protocols especially SOAP. These services can represent either new applications or just wrappers around existing sensor systems to make them network enabled.

Each SOA building block in the platform can play one or both of two roles:

Service provider: The service provider creates a web service and possibly publishes its interface and access information to the service registry. Each service provider must decide which services to expose, how to make tradeoffs between security and easy availability, how to price the services, or (if no charges apply) how/whether to exploit them for other value. The provider also has to decide what category the service should be listed in for a given service. It registers what services are available within it, and lists all the potential service recipients. Furthermore, the amount of the offered information has to be decided. Depending on the data information requests, the service provider can attempt to maximize lookup requests, number of listings or accuracy of the listings. The Universal Description Discovery and Integration (UDDI) specification defines a way to publish and discover sensor data information about Web services.

Service consumer: The service consumer or web service client locates entries in the service registry using various find operations and then binds to the service provider in order to invoke one of its web services. Whichever service the service consumers need, they have to take it into the brokers, bind it with respective service and then use it. They can access multiple services if the service provides multiple services.

In the proposed sensor system integration platform we build SOAs using web services standards especially SOAP and RPC that have gained broad industry acceptance after recommendation of Version 1.2 from the W3C (World Wide Web Consortium) in 2003 (WWW Consortium, 2001). These standards (also referred to as web service specifications) also provide greater interoperability One can, however, implement SOA using any service based technology, such as Jini, CORBA or REST.

SOAP, originally defined as Simple Object Access Protocol, is a protocol specification for exchanging structured information in the implementation of Web Services in computer networks. It relies on XML Information Set for its message format, and usually relies on other Application Layer protocols, most notably Hypertext Transfer Protocol (HTTP) or Simple Mail Transfer Protocol (SMTP), for message negotiation and transmission.

Remote procedure call (RPC) is an interprocess communication that allows a computer program to cause a subroutine or procedure to execute in another address space (commonly on another computer on a shared network) without the programmer explicitly coding the details for this remote interaction. When the software in question uses object oriented principles, RPC is called remote invocation or remote method invocation.

The main purpose of the new developed SOA based platform is to integrate various heterogeneous sensor networks based on different hardware and using different communication technologies in one Integrated Smart Sensor System (ISSS). This conception enables us full integration of the sensor data and the possibility for data interchange between the sensor networks included in the platform and/or between the smart sensors from different networks too. The implemented in the platform 6LoWPan based technology allows direct access to every sensor unit and sensor node and direct sensor data interchange. Based on this technology, the developed platform provides the unique possibility of ad-hoc Virtual Sensor System (VSS) building and exploration.

The proposed custom design platform is based on the WSO2 Carbon SOA framework. OSGi-based WCO2 framework includes common capabilities shared by all WSO2 products, such as built-in registry, user management, transports, security, logging, clustering, caching and throttling services, co-ordination, and a GUI framework.

In the current version of the platform we defined and released 3 layers: **1. Application layer, 2. Data Integration layer and 3. Network layer** (Figure 1).

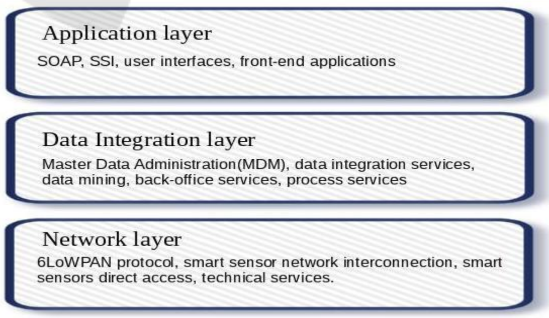

Figure 1: Integrated sensor system SOA platform

Application layer. The Application layer of the current platform is based mainly on SOAP. It relies on XML Information Set for its message format. Additionally we include in the current platform Application layer SSI and RPC protocols too. The

SSI ("Simple Sensor Interface") protocol is a communications protocol designed for data transfer between computers or user terminals and smart sensors.

Data Integration Layer. The Integration Layer marks the transition from raw sensor data to integrated data (Figure 2). This is the data that has been consolidated and rationalized. This layer represents the passage of the data through the process of integration, rather than the storage area for the data. For data with multiple sensor sources, a mastering and tuning process is required.

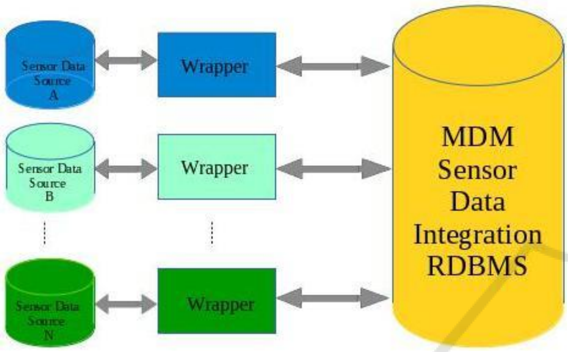

Figure 2: Data Integration layer

The core functionality of the Data Integration Layer is the Master Data Management.

Master Data Management (MDM) is the process by which data from different sensor networks or sub systems included in the platform is matched and processed to realize a single copy of data.

The MDM process should attach all relevant attribution related to the core entity. It is possible to have the MDM only process those attributes that are sourced from multiple systems. For those with a single source, no conflict exists to be resolved. The MDM system has its own internal data structures. These structures are oriented towards the structure of the source files or the target System of Record. For data with multiple sources, a mastering process is required.

Network layer. The main task of the network layer is to provide functional and procedural means of transferring variable-length data sequences from a source to a destination host via one or more sensor systems. Currently the network layer of the proposed platform for smart sensor systems integration supports the following two main protocols:

- **IPv6/6LoWPAN protocols.** 6LoWPAN is an acronym of IPv6 over Low power Wireless Personal Area Networks. The 6LoWPAN concept originated from the idea that "the Internet Protocol could and should be applied even to the smallest devices," (Mulligan, 2007) and that low-power devices with limited processing capabilities should be able to participate in the Internet of Things (Shelby, Bormann, 2011), (Shelby, Bormann, 2009).

  The base specification developed by the 6LoWPAN IETF group is RFC 6282. The problem statement document is RFC 4919.
- **Internet Control Message Protocol** version 6 (ICMPv6) is the implementation of the Internet Control Message Protocol (ICMP) for Internet Protocol version 6 (IPv6) defined in RFC 4443 [14]. ICMPv6 is an integral part of IPv6 and performs error reporting and diagnostic functions (e.g., ping), and has a framework for extensions to implement future changes.

Several extensions have been published, defining new ICMPv6 message types as well as new options for existing ICMPv6 message types. Neighbour Discovery Protocol (NDP) is a node discovery protocol in IPv6 which replaces and enhances functions of ARP. Secure Neighbour Discovery Protocol (SEND) is an extension of NDP with extra security. Multicast Router Discovery (MRD) allows discovery of multicast routers.

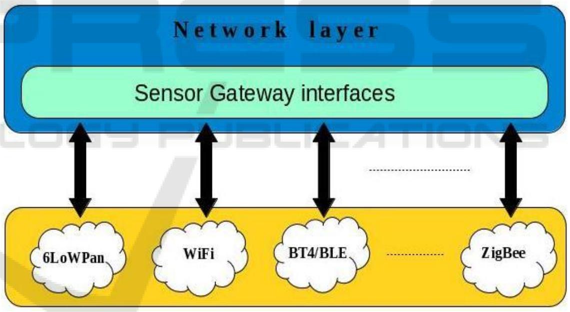

Figure 3: Network layer

The main components of the Smart Sensor System Integration Platform (SSSIP) are:

- application server running custom design modified WCO2 Carbon framework
- data base server running MySQL RDBMS
- gateway servers with related interfaces to access heterogeneous sensor networks or single addressed smart sensors.
- custom design software interfaces supporting ZigBee 802.15.4, 6LoWPan, WiFi 802.11/bgn and BT4/BLE protocols for data exchange.

The structure of the platform is open and can be easily upgraded with different functionality depending on the specific requirements.

4 CONCLUSION

The paper describes a SOA based platform developed for smart sensor systems integration. It has services to manage different heterogeneous sensor networks or group of smart sensors in one sensor system which provides the necessary interoperability. The developed services allow for easy integration of heterogeneous sensors and creation of data views for application developers. The platform is hardware independent and the developers based on the proposed services can easily access data from every sensor network or smart sensor without the need of knowledge how the sensor system work and which kind of communication protocols are running. There are services allowing data compression and messages encryption. Many smart sensor networks wireless or not can be integrated in one smart sensor system under common management and QoS. The process of integration is transparent and geographically independent. This means that sensor networks based on different countries and with different topology and functionality can act as one sensor system.

Based on developed communication interfaces, currently the platform supports ZigBee, WiFi and BT4/BLE communication technologies. The long range RoIP based communication platform is under development too. Also, there is an opportunity of building of Virtual Sensor Systems accepting specific requirements by simply developed services. The implemented 6LoWPan protocol management as service in the platform allows for direct access to smart sensor devices supporting this protocol, as for example the devices from Texas Instrument CC2538, CC1180 etc.

ACKNOWLEDGEMENTS

The research work reported in the paper is partly supported by the project AComIn “Advanced Computing for Innovation”, grant 316087, funded by the FP7 Capacity Programme (Research Potential of Convergence Regions).

REFERENCES

Bacchanalian H., Toggle, 2003. Migrating to a service oriented architecture, *IBM Developer Works*, pp. 117–121.

Delicato F., P. Pires, L. Pinnez, L. Fernando, L. da Costa, 2003. A flexible web service based architecture for wireless sensor networks. In: *Proceedings of the 23rd International Conference on, Distributed Computing Systems Workshops*, pp. 730–735.

Erl T., 2007. *SOA: Principles of Service Design*. Prentice Hall/PearsonPTR.

Gil-Martinez-Abarca J., J. F. Macia-Perez, D. Marcos-Jorquera, V. Gilart-Iglesias, 2006. Wake on LAN over internet as web service, In *Proceedings of the ETFA ’06 IEEE Conference on Emerging Technologies and Factory Automation*, pp. 1261–1268.

Grosky W., A. Kansal, S. Nath, J. Liu, F. Zhao, 2007. Senseweb: An infrastructure for shared sensing. *Multimedia, IEEE*, 14(4), pp. 8–13.

Kodali R., 2005. *What is service-oriented architecture?* http://www.javaworld.com/article/2071889/soa/what-is-service-oriented-architecture.html

Madden S. R., M. J. Franklin, J. M. Hellerstein, W. Hong, 2011. TinyDB: an acquisitional query processing system for sensor networks. *ACM Trans. Database Syst.*, 30(1), pp. 122–173.

Moeller R., A. Sleman, 2008. Wireless networking services for implementation of ambient intelligence at home. In *Proceedings of the 7th International Caribbean Conference on Devices, Circuits and Systems, ICCDCS*, pp. 1–5.

Mulligan G., 2007. The 6LoWPAN architecture, In *Proceedings of the 4th workshop on Embedded net-worked sensors, EmNets '07, ACM*.

OASIS Standard, 2009.D*evices profile for web services (DPWS)* Version 1.1.

Priyantha N. B., A. Kansal, M. Goraczko, F. Zhao, 2008. Tiny web services: design and implementation of interoperable and evaluable sensor networks. In: *Proceedings of the 6th ACM conference on Embedded network sensor systems*, pp. 253–266, ACM, New York.

Samaras I. K., J. V. Gialelis, G. D. Hassapis, 2009. Integrating wireless sensor networks into enterprise information systems by using web services. In *Proceedings of the 3rd International Conference SENSORCOMM ’09*, pp. 785–791

Shelby Z., C. Bormann, 2011. Part 1: Why 6LoWPAN? In: *6LoWPAN: The wireless embedded Internet*, John Wiley & Sons Ltd.

Shelby Z., C. Bormann, 2009. *6LoWPAN: The Embedded Internet,* Wiley Inter science.

Svelte A. T., 2010. *Cloud Computing: A Practical Approach*. McGrow Hill.

The Open Group, 2007. *Service Oriented Architecture (SOA) in the Real World*. Chapter 1. http://msdn. microsoft.com/enus/library/bb833022.aspx .

World Wide Web Consortium, 2001. *SOAP Specifications* http://www.w3.org/TR/2001/WD-soap12-20010709/.