Data Fusion Mechanism based on a Service Composition Model for the Internet of Things

Sandra Rodríguez-Valenzuela, Juan A. Holgado-Terriza, Jesús L. Muros-Cobos, José M. Gutiérrez-Guerrero
Software Engineering Department - University of Granada
C/ Periodista Daniel Saucedo Aranda s/n, 18071, Granada, Spain
{sandra, jholgado, jesusmuros, jmgutierrez}@ugr.es

Abstract-- The Internet of Things (IoT) enables communication between smart objects promoting the pervasive presence around us of a variety of things or objects which are able to interact with each other and cooperate to reach common goals. The IoT objects can obtain data from their context, such as home, office, industry, body, etc. This data can be fused and combined to obtain new and more complex information by a data fusion process. There are several algorithms of data fusion, but instead in this paper a novel method is presented for managing the data acquisition and fusion based on the service composition mechanism of a SOA middleware. This approach can simplify the data provision to data fusion mechanism, making easier the treatment of the information in a pervasive environment.

Keywords-- SOA, Internet of Things, Pervasive Computing, Services Composition, Data fusion

I. Introduction

The main idea of the Internet of Things (IoT) is the pervasive presence around us of several things or objects, such us sensors or actuators, different between them, but which are able to interact with each other and cooperate with their neighbors to reach common goals [1].

From the point of view of working and home-automation fields the IoT has direct applications and it involves new paradigms to the application development [2] [3] [4]. All the objects of the real world, at home, at office, etc., are interconnected providing high level functionalities with autonomous, smart collaborative behavior. Also, the IoT has new development problems concerning the networking and composition aspects. The objects on the IoT will be resource-constrained devices in terms of both computation and energy capacity and the proposed solutions must resolve lightweight applications which uses efficiency the resources and managing the scalability and interoperability of the devices [5].

The big heterogeneity of the IoT devices makes suitable use a middleware layer to guarantee the scalability and the interoperability in this kind of systems. Service Oriented Architecture (SOA) is becoming the most widespread approach for the implementation of distributed systems middleware [6]. The idea of assembling application components into a network of loosely coupled services to create flexible and dynamic processes with agile applications spanning different computing platforms reinforces the role of the service as the main abstraction to support the development of distributed applications [7].

The success of the IoT application development will be strongly linked with the cooperation or collaboration between the heterogeneous networked embedded devices such as services [8]. In the next generation Internet, real-world devices will be able to share their functionality and to cooperate with other components dynamically. In a real scenario, such as home-automation or chronic disease management, the information provided by sensors as heterogeneous data can be combined and fused to obtain new information more useful to the final user [9] [10]. The data fusion process has been traditionally carried out in a centralized fashion using powerful servers which analyze, apply reasoning and do inference of new knowledge with the data [11]. In this paper we propose a new paradigm of data fusion based in the services composition model of DOHA (Dynamic Open Home-Automation), a SOA distributed middleware. We use distributed and dynamic services which are responsible of their own behavior and which can collaborate between them interchanging context data information and carrying out complex functionality.

The rest of the paper is organized as follows. Section 2 introduces the research background related with this work. Section 3 presents the basics of services platform on which data fusion is implemented. Section 4 details the main aspects of data fusion mechanism and how it is implemented over the service platform based on the IoT principles. Section 5 explains three examples of how it is applied on different application domains. Finally, Section 6 shows the future research lines and the conclusions of the work.

II. Background

The pervasive computing and the IoT are changing the way we live. There is a virtual world of objects with which we interact at our work, home, and even in our relationship with other people. The application of the IoT goes from ambient assisted living to healthcare domain, including sensing, data collection, and monitoring of personal, health and environmental parameters.

Fused and combined information from sensors embedded in mobile devices can be used by IoT applications to determine the user’s situation and to build adaptive context-aware services. Bernardos et al. uses a SOA-based architecture implemented in mOSGi for the development of context-aware applications over a framework called CASanDRA [12].

An information model is frequently used to structure the information, to define the abstractions and concepts underlied into IoT domain objects and finally to describe the relationships between them. De et al. presents a semantic and well-structured model to establish the relation between the real-world heterogeneous physical objects with the digital world services which represent them [13].

Others very popular scenarios related with the IoT are Body Sensor Networks (BSNs). BSNs come with the promise to improve quality of life and healthcare of disabled and elderly people. However, the distributed and changeable character of this kind of networks has new challenges to resolve. The research of Calhoun et al. describes an approach that addresses all aspects of an IoT system in the area of BSNs, from low-level hardware design, to higher level communication and data fusion algorithms, up to top-level applications [14].

From a point of view of collaborative services which interact between them to carry out their functionality, the data fusion algorithms applied to obtain context-awareness, semantic representation and healthcare applications can be seeing as composite operations distributed in the services of a network. We have focused our research work in the IoT on sensor data modeling and composing operations to carry out complex task without the user intervention. From the perspective of a very near approach with us, the work of Guinard et al. proposes a process and a suitable system architecture that enables developers to dynamically query, select and use running instances of real-world services in the context of composite, real-world applications [8].

In the scope of the interconnected embedded devices OSGi (Open Services Gateway initiative) alliance has created several specifications to make easier the development of applications. This specification defines a system which allows design compatibles platforms to share services [15]. An important aspect to take account if you decide to use OSGi is its centralized character; all services must to be executed in the same machine.

III. DOHA: SOA-Based Middleware

DOHA (Dynamic Open Home-Automation) is a services platform for the access, control and management of home-automation systems that facilitates the construction of dynamic, scalable and pervasive applications, based on a set of lightweight and independent services.

DOHA abstracts the physical distribution of devices and its management by a set of high-level collaborative services, as shown in Fig. 1. The platform promotes the collaboration between services which involve communication between different kind of nodes at a lower level, and the interconnection between devices across different networks placed on diverse subnets at the lowest level.

The DOHA platform takes into account another important aspect in pervasive systems, the large number of heterogeneous hardware devices that may be part of a network, and how the hardware interaction is managed from the service level (e.g. HVAC, temperature sensor, alarm clock). DOHA makes use of JavaES (Java Embedded System) for the management of different types of physical devices (e.g. appliances, sensors, actuators) [16]. The access to the hardware is carried out in a standardized fashion, since JavaES abstracts the specifics hardware capabilities of each embedded device [17].

A service in the context of DOHA is an autonomous self-contained component capable of performing specific activities or functions independently, that accepts one or more requests and returns one or more responses through a well-defined, standard interface. There are two special types of services in DOHA: the *Device Service* and the *Pure Consumer Service*. The *Device Service* can interact with the physical devices of the environment and it provides physical device control. The *Pure Consumer Service* does not provide access to other services; it often provides access to end users applications, usually with a graphical user interface, and it does not offer functionality to the rest of services.

A DOHA service has an internal structure organized in a set of software layers: interface, application and interaction. The multilayer structure facilitates the

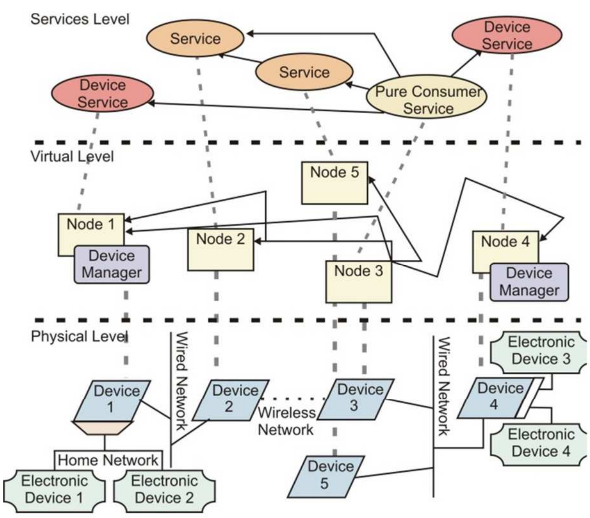

Figure 1. Levels of abstraction of DOHA platform.

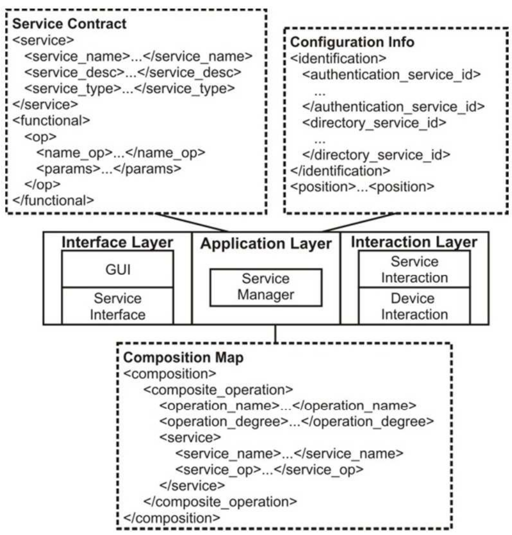

Figure 2. Anatomy of a DOHA service.

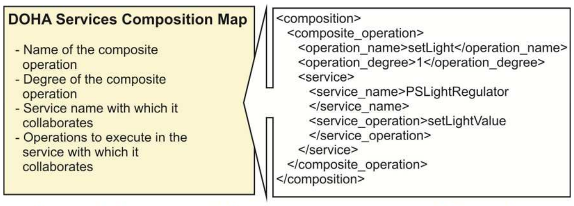

Figure 3. Structure of the Service Composition Map and an example represented in XML.

decoupling of tasks performed by a service in cohesive components; e.g., separating the task of requesting services from the task of providing services. Fig. 2 shows the anatomy of a DOHA service which provides a model to design and implement services and also allows managing the behavior flow of the service in each step of its execution.

The deployment, start-up and execution of the service require knowing additional information that is managed by the service during its life-cycle at runtime. This information is enclosed into three descriptive documents which form the base of the service specification. These ones are the *Service Contract*, the *Service Composition Map* and the *Service Configuration*. Each of them abstracts a fundamental aspect of the service within the platform and contextualizes his collaborative behavior with other services. The *Service Contract* is a public resource exchangeable between services, containing a description of the requirements, restrictions and functionality of a specific service. This information will be exploited by the rest of services in order to be aware of the functionality offered by the service and later make use of it. The *Service Composition Map* establishes the relations among services and the operations they are to perform, in order to carry out composite operations. This information is private and only accessible by the service itself, which is the only one who knows with which other services it is supposed to interact, in order to carry out a composite operation. Finally, the *Service Configuration* is needed to initiate the execution of the service, and it contains configuration parameters related to the software infrastructure that provide support to the service such as JavaES.

IV. DATA FUSION

The next generation Internet comes with the promise of integrating several technologies and communications systems with the aim of building an intelligent abstraction of information around the world. On the other hand, Machine-to-Machine communication systems combine multiple sensors interactions and merge the information obtained by the sensors applying several data fusion algorithms. The distributed architecture of DOHA and especially its dynamic services composition model can provide the base for a distributed data fusion mechanism.

In data fusion we must distinguish the mechanism used for acquiring data to be fused from the specific algorithm applied to produce new data. In this paper we have investigated the first one, i.e., the mechanism for data provision. But, in general, papers are more focused on the algorithms, its complexity and the provision of estimators and classifiers [18].

Traditionally, data fusion is processed by a hardware or software component with a direct access with external data sources by invoking an API function or by using system buses. Thereby, we can impose a sustained data provision and also a satisfaction of real-time restrictions by the implementation of a time-out facility which controls both issues.

Instead of implementing in-house or legacy components, the application of a distributed technology for data fusion has some advantages with respect to traditional approaches. A distributed paradigm supplies a well-established communication model (e.g. message passing primitives) with a control of networks fails, giving a reliable binding mechanism with external data sources. In addition, the computing processing is shared between devices instead of performing all on a single device or server.

The use of a SOA paradigm benefits the overall building of a system, since the system can be seen as a group of interconnected low coupled services, each one in charge of its part of functionality and the maintenance of data produced by them. Besides, the services composition model of DOHA gives a natural way to implement a distributed data fusion mechanism. Each service handles autonomously the management of specific functions and data. Some of them can need to collaborate with other services by requesting data by means of the execution of composite operations. This kind of service may then combine or fuse applying a specific data fusion algorithm to the data acquired from external services in order to build new information or data, which in turn is shared to any other services that claim for it. Using this method the services can be run in different nodes. This system allows data fusion without a main server due to information will be processed in the embedded devices.

A. *Service Composition Model*

The data fusion mechanism depends on the service composition model afforded by the services platform. DOHA provides a method based on the activities that a service should perform when it needs to collaborate with other services to complete a requested operation. The types of operations that services can perform are simple or composite. A *simple operation* is a single transaction that the service can perform by itself, i.e., the service has all the resources necessary to carry it out and it does not require interaction with other services. In contrast, a *composite operation* involves the invocation of one or more operations in one or more services. The service that owns the composite operation is responsible of its execution and it must interact with the services involved in the operation, the *requested services*, to get the necessary functionality in order to complete the whole operation. A service that only has simple operations is called a basic service; whereas a service that implements composite operations invoking other services is called a composite service. A *Device Service* is an example of a basic service that is composed only of simple operations. The composite operations are the base of the collaboration model of the platform and are listed by the *Service Composition Map*.

In the XML code of Fig. 3 we can observe the structure of a DOHA *Service Composition Map* that lists the requested services (using meta-data enclosed into *<service_name>* tags) and the corresponding invoked operations required for the execution of a composite operation available on the service. In this case, the *Service Composition Map* belongs to the *PSLight* service, which make use of the *PSLightRegulator* service to manage the composite operation *setLight*().

The service composition model presented is verifiable and correct by construction [19]. Each service has its own composition graph based on a directed acyclic graph which models its composite operations. The set of all composition graphs in an application forms the total collaborative execution flow of the application based on composite operations. By construction, we can define the system composition graph of an application as the union of individual composite operation graphs of the services of the application. The system composition graph can be built dynamically in order to know the *Service Composition Map* of the running services at any time in contrast with other works that require a static definition of the system composition graph [20].

From the point of view of the owner of the composite operation, the service composition process starts when it receives a request for this operation from a service consumer. After receiving the composite operation request the service searches for the services involved in the operation. When the communication between services can be carried out, the service owner of the composite operation acts as customer of the rest of the services. The users that request for a composite operation do not know that it involves the collaboration between services and the composition of their operations fusing the data obtained in the interaction process. The users only make the request and receive a response from the service owner of the composite operation. The service is only aware of the requested services, allowing the collaboration with them to be carried out without user intervention, creating autonomously collaborative applications at service level. However, the service is not aware of the rest of the services available in the platform

V. An Example of Application to a Collaborative Services System

Summing up the previously concepts presented in the paper, the data fusion process proposed is based on the composition model established by the DOHA service platform. The composition model determines the execution of the composite operations of the services. In a distributed and absolutely independent manner determined by the composition map, the services interact between them to obtain the information from the rest of the services and they combine the information achieving the data fusion by themselves.

To illustrate the model of service composition in DOHA platform we are going to show several example scenarios applied to some relevant application areas of the IoT, such as home-automation, ambient assisted living and industry. We are going to see how different services implement their functionality using composite operations which involve the collaboration between services and which implies to fuse the data obtained.

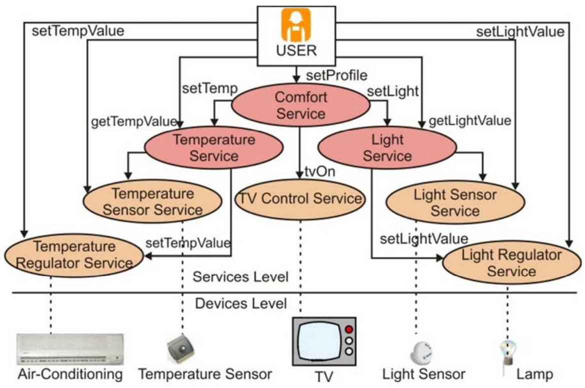

Figure 4. Example of home-automation services composition.

A. Home-Automation Study Case

The goal of the example application in the home-automation scope is to develop services which users need to cover the main home requirements. E.g., we want to measure the comfort level of the user home or room. To do it, we need to create several device services which interact with the environment and which obtain, e.g., light and temperature levels. Other services will have higher level and they interact with the first one. With data obtained of this interaction, the composite services will do data fusion and they will inference new, more complex information. The implemented services are *Comfort Service*, *Light Service*, *Light Sensor Service*, *Light Regulator Service*, *TV Control Service*, *Temperature Service*, *Temperature Sensor Service* and *Temperature Regulator Service.* As we can see in Fig. 4 the sensor and actuator services of light and temperature, and the television service, are device services (simple services) which do not have composite operations, which means that do not interact with other services of the platform and they do not fuse the data obtained from other services. Instead, the comfort service, the temperature service and the light service have composite operations, i.e. they will do data fusion with the information obtained after interact with other services of the platform.

B. Ambient Assisted Living Study Case

Actually several public organizations are researching in systems which can monitor state of people which lives alone, especially older people, to send help if it is necessary. In order to detect the person state is not enough with only a measure. To monitor a person, the system needs to reason with data acquired from several sensors, cameras, and so on. It is necessary to fusion data when this is acquired. These systems include programs which detect when a person has suffered a health problem and to advise a telecare center. In these centers there are people working which can try to contact with user and send help if it is necessary [21].

With DOHA it is not necessary a big center to process data. All nodes can be used to fuse data. Supposing a case where network have several nodes, being one node a mobile phone with accelerometer and other nodes are connected to different IP cameras. When mobile phone accelerometer detects an abrupt change, mobile device can execute a service to check if person is fallen. This service will use nodes with cameras to check if person is in the floor. The camera nodes will communicate with each other to determine the probability that the person is fall merging their images. If probability is over threshold a service to call for help will be executed.

This method can save money to corporation dedicated to telecare because infrastructure is more reduced than a main server. At same time the privacy is protected (e.g. camera images) and the presence of false positive is reduced, by sending data to telecare center only when a risk situation is detected.

C. *Industrial Study Case*

At big industries such as automotive, chemical, steel, and so on, there are a lot of embedded devices involved in complex, cyclic processes on huge industrial plants. These processes are direct examples of data fusion processes. Sensors distributed in the plant collect information such as temperature, humidity, pressure, etc., and then these data is fused and processed, usually, in a centralized PLC (Programmable Logic Controller). By means of the DOHA services composition model, the information captured by sensors of the plant is managed directly by services which interact between them to fuse the data and to start new operations if it will be necessary.

E.g., a process that involves data fusion in an industrial plant is the control of a water tank. As we can see in Fig. 5, in this industrial process there are involved two sensors, temperature and pressure, and several actuators to control the valves of input and output water of the tank, with an impulsion pump which makes circulate the water around the circuit.

For the sensor and actuators we define device services, i.e., simple services without composite operations. However, we need three composite services which use the simple operations defined before to hold the water tank in a correct state. The high level services will be the *Instrumentation Service,* the *Actuation Service* and the *Tank Service.* The *Instrumentation Service* will interact with the sensors services of the water circuit checking its correct state. The *Actuation Service* will interact with the regulators service of the water circuit to modify the state of the system. Finally, the higher level service will be the *Control Tank Service*

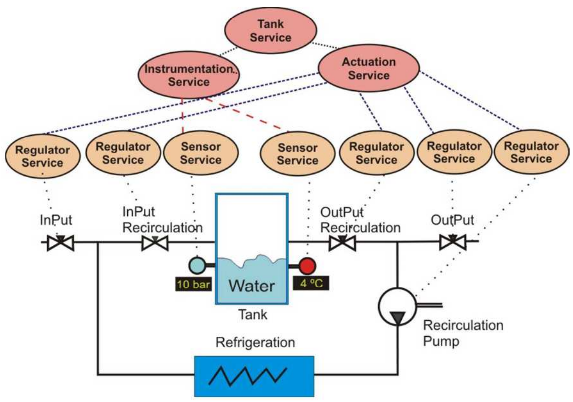

Figure 5. Example of industrial plant services composition.

which will fuse the data obtained from the *Instrumentation* and *Actuation* services to keep correct the state of the water bank in the industrial plant.

VI. Implementation Notes

DOHA proposes an architectural model based on services which can be implemented over several technologies [22]. To test the applicability of DOHA for the distributed data fusion mechanism based on SOA, the DOHA implementation in DPWS (Device Profile Web Service) is selected for several reasons, e.g. DPWS is a framework to develop lightweight web services especially for embedded devices.

It adds some restrictions to the standard specification of web services, such as the messages sizes. These restrictions ensure the web services run properly on devices with limited resources [23]. DWPS is focused to develop application on devices with IP protocol. The main objective of DPWS is to implement web services in small devices, which will be able to do basic functions, such as send and receive messages safely, dynamically discover web services, describe a web service and subscribe/receive events from another web service.

To deploy DOHA with DPWS we must respect the DOHA anatomy of services and implement the class model proposed by the platform. We have implemented the DOHA anatomy of services using the concepts of Hosted Device and Hosted Service of DPWS. There are aspects of DOHA with direct correspondence in DPWS; e.g. the DOHA interface layer can be implemented using the concept of Hosted Device. Unfortunately there are many aspects of DOHA that DPWS ignores, as the interaction layer, the discovery services to allow the service discovery in remote networks, etc., but the implementation of these aspects is out of the scope of this paper.

The DOHA services Implemented with DPWS have been deployed in three different kind of embedded devices with different properties and resources. We can see in Table 1 a resume of the characteristics of the embedded devices used.

The role of mobile devices has an increasing importance in pervasive systems, since as Mattern pointed mobile phone, tablets, etc. can be used as a mediator between system and person [24]. Actually

TABLE I

Hardware and Software characteristics of the Embedded Devices Tested

Embedded Device	Hardware	Software
SNAP	2 Mbyte flash memory 8 Mbyte DRAM	J2ME CLDC, certified by Sun Micro Systems TCP/IP stack
IM3910	RISC processors SDRAM interface, 4 GB address range, auto detected configuration	J2ME CLDC, certified by Sun Micro Systems
pcm9374 advantech	C3-LP 1GHz processor 2nd Cache Memory 64 KB on the Eden processor	J2ME CLDC, certified by Sun Micro Systems

most of all people have, at least, a mobile phone (Android, IPhone, etc.). Mobile devices are becoming the preferred device for users, and probably the way to interact with any object or device in IoT. Besides, the increasing computing power of next generation smartphones and tablets makes possible to profit its computing resources to fuse data obtained from other Device services over resource-constrained embedded devices. Also even it can collaborate with other sharing its computing power to solve heavy operations on embedded system with scarce resources. In our case, several kinds of clients applications is performed on Android.

The test of the DOHA services over these embedded devices was successful, but we have detected a relevant issue which must be resolved in next implementations: DPWS only support communication in local networks and the interaction over the Internet is not possible. We are studying how to develop a special service which allows establishing communication between nodes located in different sub-networks.

VII. Conclusions

Nowadays the relevance of the IoT in the scope of the pervasive computing is growing up. The number of embedded devices existents in daily life is increasing in an exponential way. At home, at work and even in our relationship with other people we use continuously multiple interconnected embedded devices. The information provided by these kinds of objects and devices must be interpreted and, sometimes, merged to obtain richer information. A lot of research works pose data fusion process using complex algorithms. In this paper we propose a novel method to implement a distributed data fusion using the service composition model of DOHA, a SOA-based middleware. This method allows working with data in a distributed, decentralized manner. Each service is responsible of acquiring data from external services by means of its composite operations in order to manage, combine, fuse or build new information, which in turn is shared to any other services that claim for it through the services network. The distributed character of the proposed approach for data fusion makes the service model very dynamic and scalable, really important aspects in the development of applications which will be displayed in embedded devices.

References

[1] L. Atzori, A. Iera, and G. Morabito, "The internet of things: A survey," *Computer Networks*, vol. 54, no. 15, pp. 2787 – 2805, 2010.

[2] D. Zuehlke, "Smartfactory: Towards a factory-of-things," *Annual Reviews in Control*, vol. 34, no. 1, pp. 129 – 138, 2010.

[3] D. Markovic, D. Cvetkovic, D. Zivkovic, and R. Popovic, "Challenges of information and communication technology in energy efficient smart homes," *Renewable and Sustainable Energy Reviews*, vol. 16, no. 2, pp. 1210 – 1216, 2012.

[4] M. Kovatsch, M. Weiss, and D. Guinard, "Embedding internet technology for home automation," in *Proceedings of the 15th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA 2010), Bilbao, Spain*, Sep. 2010.

[5] D. Havlik, S. Schade, Z. A. Sabeur, P. Mazzetti, K. Watson, A. J. Berre, and J. L. Mon, "From sensor to observation web with environmental enablers in the future internet," *Sensors*, vol. 11, no. 4, pp. 3874–3907, 2011.

[6] M. Papazoglou, P. Traverso, S. Dustdar, and F. Leymann, "Service-oriented computing: State of the art and research challenges," *Computer*, vol. 40, no. 11, pp. 38 –45, nov. 2007.

[7] J. Holgado-Terriza and S. Rodríguez-Valenzuela, "Service oriented middleware for home-automation," 2011, submitted to Journal of Network and Computer Applications.

[8] D. Guinard, V. Trifa, S. Karnouskos, P. Spiess, and D. Savio, "Interacting with the soa-based internet of things: Discovery, query, selection, and on-demand provisioning of web services," *IEEE Trans. Serv. Comput.*, vol. 3, no. 3, pp. 223–235, Jul. 2010.

[9] A. Bernardos, E. Madrazo, and J. Casar, "An embeddable fusion framework to manage context information in mobile devices," in *Hybrid Artificial Intelligence Systems*, ser. Lecture Notes in Computer Science, E. Corchado, M. Graña Romay, and A. Manhaes Savio, Eds. Springer Berlin / Heidelberg, 2010, vol. 6077, pp. 468–477.

[10] C. Bellos, A. Papadopoulos, R. Rosso, and D. Fotiadis, "Heterogeneous data fusion and intelligent techniques embedded in a mobile application for real-time chronic disease management," in *Engineering in Medicine and Biology Society,EMBC, 2011 Annual International Conference of the IEEE*, 30 2011-sept. 3 2011, pp. 8303 –8306.

[11] D. Han, J. Zhang, Y. Zhang, and W. Gu, "Convergence of sensor networks/internet of things and power grid information network at aggregation layer," in *Power System Technology (POWERCON), 2010 International Conference on*, oct. 2010, pp. 1 –6.

[12] A. Bernardos, P. Tarrio, and J. Casar, "Casandra: A framework to provide context acquisition services and reasoning algorithms for ambient intelligence applications," in *Parallel and Distributed Computing, Applications and Technologies, 2009 International Conference on*, dec. 2009, pp. 372 –377.

[13] S. De, P. Barnaghi, M. Bauer, and S. Meissner, "Service modelling for the internet of things," in *Computer Science and Information Systems (FedCSIS), 2011 Federated Conference on*, sept. 2011, pp. 949 –955.

[14] B. Calhoun, J. Lach, J. Stankovic, D. Wentzloff, K. Whitehouse, A. Barth, J. Brown, Q. Li, S. Oh, N. Roberts, and Y. Zhang, "Body sensor networks: A holistic approach from silicon to users," *Proceedings of the IEEE*, vol. 100, no. 1, pp. 91 –106, jan. 2012.

[15] O. Dohndorf, J. Kruger, H. Krumm, C. Fiehe, A. Litvina, I. Luck, and F.-J. Stewing, "Towards the web of things: Using dpws to bridge isolated osgi platforms," in *Pervasive Computing and Communications Workshops (PERCOM Workshops), 2010 8th IEEE International Conference on*, 29 2010-april 2 2010, pp. 720 –725.

[16] J. A. Holgado-Terriza and J. Viúdez-Aivar, "A flexible java framework for embedded systems," in *ACM International Conference Proceeding Series*, 2009, pp. 21–30.

[17] J. A. Holgado-Terriza and J. Viúdez-Aivar, "Distributed, embedded and real-time java systems: Javaes, a flexible java framework for embedded systems," 2012, pp. 321-350, Edited by Andy Wellings and M. Teresa Higuera-Toledano.

[18] D. Smith and S. Singh, "Approaches to multisensor data fusion in target tracking: A survey," *Knowledge and Data Engineering, IEEE Transactions on*, vol. 18, no. 12, pp. 1696 –1710, dec. 2006.

[19] J. Holgado-Terriza and S. Rodríguez-Valenzuela, "Services composition model for home-automation peer-to-peer pervasive computing," in *Proceedings of the Federated Conference on Computer Science and Information Systems*, 2011, p. 529–536.

[20] M. Nakamura, A. Tanaka, H. Igaki, H. Tamada, and K. Matsumoto, "Constructing home network systems and integrated services using legacy home appliances and web services," *International Journal of Web Services Research*, vol. 5, no. 1, pp. 82–98, 2008.

[21] A. Dohr, R. Modre-Opsrian, M. Drobics, D. Hayn, and G. Schreier, "The internet of things for ambient assisted living," in *Information Technology: New Generations (ITNG), 2010 Seventh International Conference on*, april 2010, pp. 804 –809.

[22] S. Rodríguez-Valenzuela, J. M. Gutierrez-Guerrero, and J. A. Holgado-Terriza, "A service platform towards the interoperability between jxta and web services," 2012, unpublished.

[23] OASIS, *Devices Profile for Web Services Version 1.1*, OASIS Standard Std., Rev. 1.1, 2009.

[24] F. Mattern and C. Floerkemeier, *From the Internet of Computers to the Internet of Things*, ser. LNCS. Springer, 2010, vol. 6462, pp. 242–259.