Design Guidelines for Integration of Wireless Sensor Networks with Enterprise Systems

Laurent Gomez
SAP Research
805, Av. du Dr. M. Donat
06250 MOUGINS, France
laurent.gomez@sap.com

Annett Laube
SAP Research
805, Av. du Dr. M. Donat
06250 MOUGINS, France
annett.laube@sap.com

Alessandro Sorniotti
SAP Research
805, Av. du Dr. M. Donat
06250 MOUGINS, France
alessandro.sorniotti@sap.com

ABSTRACT

Deploying a large number of small wireless sensors that can gather, process, and deliver information about physical environment to external systems opens many novel application domains. Today Wireless Sensor Networks (WSN) are highly interesting for different application domains such as military or healthcare [3]. Despite this interest, the integration of WSNs with business applications still raises technical challenges. WSN and context-aware middlewares aim at addressing this issue. Nevertheless, these middleware frameworks mainly focus on application development and deployment rather than on business application requirements. In this paper, we discuss the necessity of an Enterprise Integration Component (EIC) that addresses business application requirements. Combining top-down approach of context-aware middleware and bottom-up approach of WSN middleware, we propose the design of a SOA based architecture. The latter addresses the heterogeneity of WSNs and offers a standardized, secure and trusted way to access sensor data.

Categories and Subject Descriptors

H.3 [**Information Storage And Retrieval**]

General Terms

WSN, Enterprise System, Integration, Middleware

1. INTRODUCTION

Sensor networks appear to be one of the technologies of the 21st century [5, 17]. Empowered with short wireless communication capabilities, wireless sensor networks (WSNs) leverage traditional sensor networks by enabling random and dense deployment of smart and low-resource devices in different physical areas (e.g. battlefield, body) [3]. We understand WSNs as any self-organized wireless network involving low-cost, low-power and heterogeneous sensor devices, which are capable of sensing, processing, routing and disseminating data towards special sensor nodes so-called sinks or gateways. WSNs are highly profitable for business applications from different domains (e.g. defense, healthcare, traffic control). Monitoring and controlling of physical environment is one of the main reasons of the enthusiasm for WSNs. Thus, WSNs bring physical world to business applications, which support adaptation to specific needs of customers. In addition, it supports business applications in optimization and reduction of time-latency in decision making.

Nevertheless today there is a gap between business applications and WSN middleware systems. Most WSN middleware systems, described in Section 4, focus on constraints of sensor network hardware and issues related to application development and deployment rather than on the requirements of business applications. Today's work in the sensor network community focuses on collecting and aggregating data from specific networks with an associated base station. The problem of delivering this data to enterprise systems is typically left open. There is a clear need for another type of middleware systems.

In this paper, we discuss the design of an architecture that allows business applications to use processed (e.g. by aggregation, fusion) or raw sensor data directly acquired from sensor nodes to adapt their business functionality. We consider, for example, business applications such as remote healthcare monitoring where a patient is monitored remotely at home after surgery. His pulse, body temperature, ambient temperature and his activities are monitored 24 hours per day using a WSN. The latter is connected to a Medical Emergency Response Center (MERC) through a middleware, partially hosted for example by the patient's PDA. This middleware is in charge of detecting any irregularities in patient health condition. The MERC then registers to the middleware for a set of high level information related to patient health condition. The middleware can for example trigger an alert in case of irregularities to the MERC, which contacts a physician for a home visit. Such scenarios raise few technical issues regarding (i) interoperability between WSN and business applications, (ii) heterogeneity of acquired sensor data, (iii) confidentiality and availability of patients' medical information.

In order to ease the design of such an architecture, we first express needs of business applications. Then, we analyze the requirements for a new type of integration middleware, the Enterprise Integration Component (EIC). On the base of the in-depth analysis, we sketch an architecture of a higher-level middleware, which enables the integration of WSNs into

MobilWare'08, February 12-15, 2008, Innsbruck,Austria,

business applications in a standardized, secure and trusted way. In Section 2, we first define our terminology, before we start to analyze the business application needs and the resulting requirement for an EIC. Section 3 depicts a possible SOA based architecture that will be compared with the state of the art middleware systems in Section 4.

2. WSN INTEGRATION INTO BUSINESS APPLICATIONS

2.1 Terminology

For sake of clarification, we first define few terms used in this paper such as Enterprise System, Business Application, Wireless Sensor Network, Wireless Sensor Network Application and Enterprise Integration Component.

An **Enterprise system**, sometimes called ERP system (Enterprise resource planning system), is a set of business applications that allows large enterprises to run all phases of an enterprise's operations to facilitate cooperation and coordination of work across the enterprise. The ideal enterprise system could control all major business processes in real time [9]. Enterprise systems have in general high requirements concerning availability, scalability, reliability as well as security and interoperability.

Business applications process business information to support a specific business function such as purchasing, controlling or supply chain management.

A **Wireless Sensor Network (WSN)** is a wireless network connecting small, heterogenous and autonomous devices (sensor nodes) with limited resources to cooperatively monitor physical or environmental conditions at different locations and deliver sensed data to computing systems. By correlating sensor output of multiple nodes, a WSN as a whole provides information that a single node cannot [16].

A **WSN middleware** is a software infrastructure that glues together the sensor hardware, operating system, network stacks and the application services. A complete middleware solution contains a runtime environment that supports and coordinates application services and standardized system services and provides adaptive mechanism for energy saving [11].

WSN applications are applications executed on several sensor nodes in a WSN. They combine the functionality of services provided by the sensor nodes and the WSN middleware in order to fulfill their goals. Complex examples are fire or disaster detecting applications [10] and the monitoring of storage regulation of hazardous goods [4].

An **Enterprise Integration Component (EIC)** is a generic mediation layer between enterprise systems and the WSN middleware. This infrastructure allows business applications running in an enterprise system to access and process sensor data (e.g. aggregated, fused data) in a standardized format and in a secured and trustworthy way.

2.2 Business Application Requirements

Business applications use WSNs to retrieve real-time or nearly-real-time data from objects or locations relevant for their business (data collection). That information can be used in addition to other information sources (e.g. manual entries, barcode scans, RFID readings) for the (i) adaptation of functional and security behavior of business applications, or (ii) control or monitoring of physical environment. It allows also the acquisition of detailed information, closer to current circumstances, that were not available before, e.g. regular temperature measurements of moving objects such as a container on a ship.

Beside the collection of data, a business application can use specialized WSN applications that handle functionalities, which cannot be provided by traditional applications. Here the integration of WSN application into the business process offers new and exciting possibilities. The monitoring of dangerous goods scenario in COBIS [4] or the cloning of agents in Agimone [10] are first examples of potential use cases.

The functionalities requested by business applications can be divided into three groups: functional, interoperability and security. Regarding **functional requirements**, the EIC has to facilitate the process of sensor data acquisition while providing means to access the current data including historic sensor readings. The EIC has to provide a configurable storage for sensor data with the functionality to access and process the data. In order to leverage information delivered by WSN middleware, EIC is in charge of processing sensed data.

The second requirement of business applications is **interoperability**. The EIC must first provide sensor data in a standardized format. Thus and due to the heterogeneity of sensor nodes on the market, each node manufacturer proposes its own format for sensor data. Business applications require conversion in a unified format of the delivered sensor data by the EIC. The EIC is to support connectivity to more than one WSN via Intra- or Internet together with WSN discovery to support WSN mobility. The EIC has to provide services that support the different standards used in business applications, e.g. Health Level 7 (HL7) for healthcare applications. All communication and information exchanges have to be standardized, using common protocols. The EIC also has to provide means to acquire raw or processed sensor data in pull or push mode. In push mode, business applications explicitly request information to the EIC. In pull mode, business applications follow the publish/subscribe paradigm. In this case, business applications subscribe to event of interest (e.g. increase of the ambient temperature). The EIC then notifies only the interested applications about state change in the WSN. This approach permits to minimize communication overhead between the EIC and the applications.

The third requirement concerns **security and trust** of sensor data from nodes to business applications. Processed data contain normally sensitive and private business information (e.g. patient's medical information). Business application then implement high security standards by means of security and trust policies to be enforced from the nodes to the business applications including the EIC. All communications have to be secured and the entire sensor data acquisition, processing and storage have to be secured. Security require-

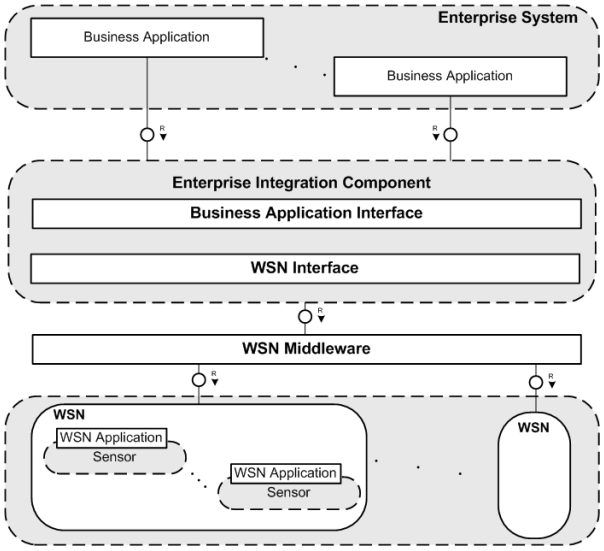

Figure 1: Architecture Overview

ments are highly dependant from the use cases. Nevertheless, availability, confidentiality and integrity of information are the most common security goals identified. Availability is mission-critical for applications controlling the business processes of big enterprises. Additionaly, business applications express trust requirements regarding sensor data. Business applications require the mapping between sensor data and confidence value. Under a given confidence threshold defined by applications, sensor data are considered as unsuitable for further use.

2.3 Enterprise Integration Component Requirements

The main objective of the EIC is the support of business applications in integrating sensor information to improve their business functionality. This integration of an enterprise system, the EIC and WSNs is depicted in Figure 1. In order to serve as a mediation layer between enterprise systems and WSNs, the EIC first exposes an interface to business applications, and provide an interface to WSN middlewares. From a functional point of view the EIC has two main interfaces, which we want to characterize more deeply in the next sections:

- **Business Application Interface**: This interface provides the functionalities to access information processed by EIC. The information can be raw or processed sensor data, acquired now or in the past. The information can be connected to events of interest or by specifying the source of information (object or location where the sensor is attached or the sensor identified by an id).
- **WSN Interface**: The second interface focuses on hiding the heterogeneity of WSN middleware and programming models. The WSN interface is in charge of collecting sensor data from WSNs and performing data conversion in a standard format. On the other hand this interface can be also used from the WSNs to perform load balancing of resource coming from restricted devices to the EIC infrastructure.

We motivate both interfaces from existing research work in Section 4. The business application interface corresponds to the functionality provided by context-aware middleware systems, while the WSN interface is partly concerned in the ongoing work to WSN middleware systems. We aim at combining the top-down approach known from context-aware systems with the bottom-up approach of the WSN middleware in one component, the EIC.

2.3.1 Business Application Interface

We distinguish three main business applications approaches to use information from WSNs: event-driven, data-driven and application-driven approaches.

In the **event-driven approach**, business applications focus on detection of state changes in the physical environment or of living or non-living objects. The business application stays passive until an event of interest occurs and reacts accordingly. A lot of examples can be found in the area of disaster detection, emergency handling and traffic control. To fulfill the needs of the event-driven use of WSNs by business applications, we can identify two main functionalities. A first service has to support the definition of events of interest. An event of interest is a change in the sensed information of a given object or location, e.g. a suddenly rising temperature of a room or a body temperature of a human higher than 38 degree Celsius. The second service has to provide the functionality to subscribe to a defined event of interest and to publish the information to the subscribers when the event occurs.

In the **data-driven approach**, business applications aim at sensing the environment in order to adapt either their functional or security behavior. Sensor information for a given location or object is actively requested by the business application. That information can be used to make decisions required by the business process which allows to view detailed information about business objects or processes during their lifetime. Examples are monitoring applications like patient monitoring or herd control, where detailed information about the health status (heart rate, body temperature, oxygen level,...) of the individual or animal is recorded for real-time analysis. Considering data-driven requirements, only a service for retrieval of sensor data is mandatory. This service should allow the retrieval of raw and processed sensor information for a given location or object reflecting the current or past state (historic sensor data). Additional services for sensor data processing are required. Examples are data aggregation, data fusion or reasoning. To keep a history of sensor data the EIC has to include a persistent storage.

In the **application-driven approach**, business applications try to integrate business functionalities or even outsource to sensor nodes. Excellent examples are the self-aware drums from COBIS [4]. The sensor nodes attached to drums and other containers used to store chemicals are responsible to detect reactive chemicals in proximity, to respect storage limits of the warehouse and to observe the ambient storage conditions. The warehouse management system receives the information and alerts from the sensors

and reacts accordingly. To support the application-driven approach the EIC has to provide a standardized service interface to call WSN applications. We suggest a web service interface described by an xml-based service description language (SDL) stored in a service repository. The service description mainly covers the service interface and service characteristics related to the resource constraints of WSN and related to security and is used to invoke the services dynamically.

These approaches are not used exclusively. On the contrary, they can be combined. For example in a healthcare application constant patient monitoring can be combined with detection of emergency situations.

In addition, EIC expresses security and trust requirements inherited from business application needs. In the example described in Section 1, confidentiality of information appears to be crucial. As in most of healthcare applications, preservation of patient privacy is the main requirement. But when patient life is concerned, requirement on confidentiality is replaced by availability of patient medical information. In addition, trustworthiness evaluation of sensor data is also considered as important requirements. Considering the fact that sensor nodes are prone to failure or corruption by attackers, trust evaluation of sensor data permits to predict fake or corrupted information.

2.3.2 WSN Interface

The WSN interface layer, sometimes named "WSN abstraction layer", of the EIC has to deal with the heterogeneity of the connected WSNs. The EIC should be able to connect to several sensor networks, which can implement different WSN middlewares and use different programming models. The EIC has to smooth the differences of the connected WSNs and provide functionalities of less powerful sensor nodes by its own infrastructure. E.g. sensor data aggregation/fusion can be done on the nodes or on the EIC infrastructure. On the other side, the EIC helps the sensors to deal with resource restriction and to balance some of their workload to the backend infrastructure. A quite interesting concept is the creation of background infrastructure representatives (BIRTs) like proposed in [19].

In order to fulfill interoperability expressed in Section 2.3.1, we target heterogeneous format of sensor data and interoperability between business applications and EIC. Capitalizing on ontologies for contextual information such as CoOL [14], we propose to integrate in the EIC a service for data mapping. The latter deals with the translation of exotic data formats, due to the heterogeneity of sensor nodes, to a standard format for sensor data. In addition, in order to address interoperability between business application, we propose to base our EIC on the web service paradigm. It enables to support a standardised interface between the business applications and the EIC.

In our architecture, we consider any services for controlling, monitoring and managing WSNs, as out of scope.

3. ENTERPRISE INTEGRATION COMPONENT ARCHITECTURE

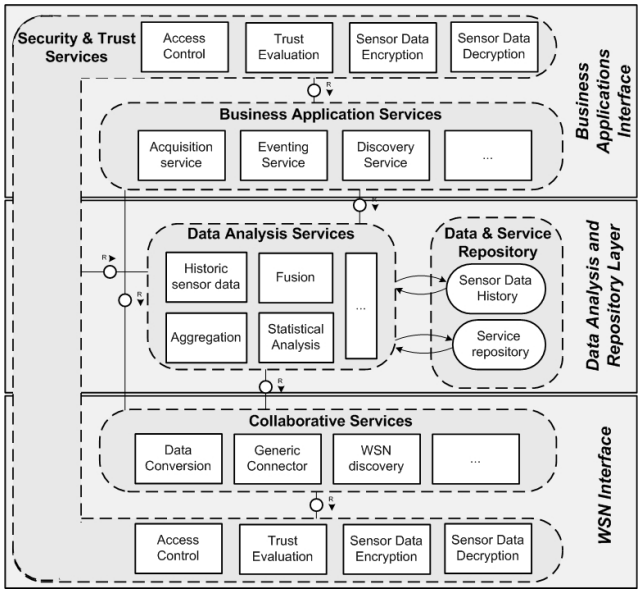

Figure 2: Enterprise Integration Component

In this section, we outline our approach driven by the previously described requirements. As depicted in Figure 2, our architecture consists of five types of services organized in the three following layers: Business Application Interface, Data Analysis and Repository and WSN Interface. Business Application Interface is a SOA based layer which contains all the web services available to business applications, as described in Section 2.3.1. The services fulfill their work by composing dynamically the underlying services. The Data Analysis and Repository layer is a WSN independent layer providing an abstraction from the WSN by providing standardized function to store and process sensor data. The WSN Interface handles the device/platform specific conversions to different operating systems and programming models. Additionally, it supports connectivity with several (mobile) WSNs and WSN discovery. Security and trust requirements of business applications are fulfilled with the security and trust services. We suggest, as example, access control and trust evaluation of sensor data. The latter supports the business application in determining the confidence in the delivered raw or processed sensor data by the EIC.

We illustrate the collaboration of the three layers with the following example: a purchasing application allows to buy cows directly from the farmer. Before an animal is bought, the buyer wants to verify the health status of the cow. In order to get information about current health status of the cow, the acquisition service uses the different services of the data analysis and repository layer to get the current and historical sensor readings and uses an aggregation service to summarize the health status. If necessary the trustworthiness of the data can be evaluated in parallel and the result can be converted in a requested data format. The request to get the current sensor data is forwarded to the WSN interface, where a WSN middleware specific service translate the request into the WSN application service request that is injected into the WSN. Figure 3 outlines this scenario.

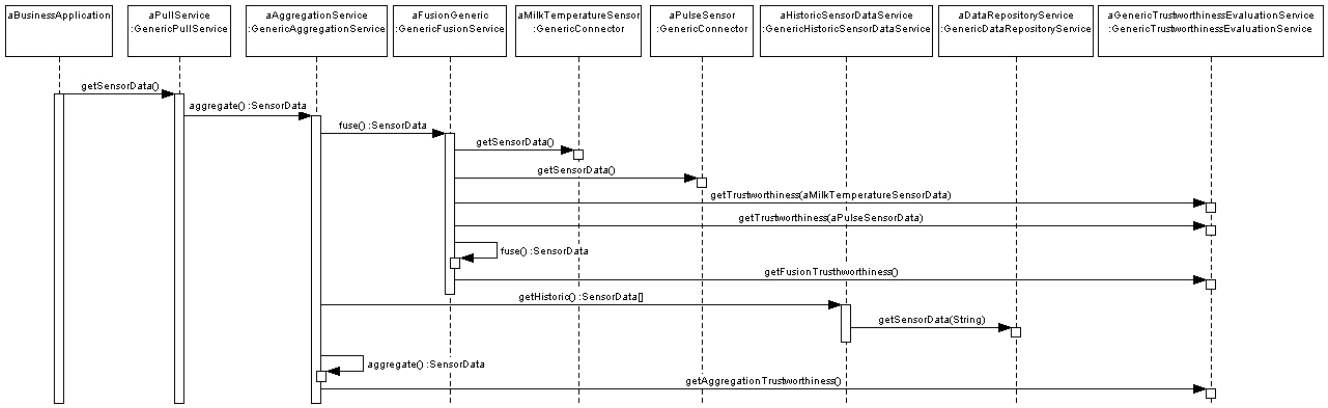

Figure 3: Herd Control Sequence Diagram

In Figure 2, we identify five different types of services: security and trust, business application, data analysis, data-and-service repository and collaboration services. Security and trust services include any services addressing confidentiality, integrity or trustworthiness of sensor data. Depending on business application or EIC requirements, these services secure sensor data going into and out of the EIC. In addition, data analysis and repository can still benefit from these services. We propose the following security and trust services.

- **Access Control**: this service aims at protecting raw and processed sensor data from unauthorised business applications by enforcing an access control policy.
- **Sensor Data Encryption and Decryption**: these services provide means for encryption or decryption of sensor data, from sensor nodes to business applications. These services target the confidentiality requirements expressed by business applications.
- **Trust Evaluation**: this service evaluates trustworthiness of sensor data from the acquisition to delivery to business applications. It enables business applications to select sensor data according to their level of confidence in the delivered information.
- **Signature**: this service permits to sign or check signature of sensor data. It provide a security mechanism to fulfill the integrity requirements on data.

Business Application Services address interoperability requirement of business applications. In order to ease the communication with enterprise systems, all these services have to implement a web service interface and to expose their functionality by means of a WSDL. We propose two basics business application services: acquisition and eventing services. The acquisition service delivers the sensor readings in a standardised format unpon business application request. Dependent on the freshness requirement, the data can also be read from the database instead directly from the sensor. The database is used as a resource cache of the WSN. The eventing service provides all functionality related to the event definition and the publish/subscribe service at the Business Application Service layer. Following the SOA paradigm of the top layer, we propose the implementation of the WS-Notification standard[13]. In addition to these services, the discovery service enables EIC to publish the available WSN application services to business applications.

Data Analysis Services implement tools for pre-processing sensor data and for enabling workload balance requirements expressed by business applications.

- **Historic sensor data**: this service delivers the historic sensor reading in a standardised format.
- **Fusion**: The fusion service performs any processing on sensor data in order to derive high level contextual information such as reasoning about patients' health condition from basic medical information (e.g. pulse, body temperature).
- **Aggregation**: This service aggregates sensor information based on object identification, the relationship of objects and its nodes and the historic sensor data. The service should be only used, if not already provided by WSN applications.
- **Statistical Analysis**: This service permits statistical analysis on sensor data on the fly. Ambient temperature average is an example of statistical analysis on the fly. Each time a new ambient temperature is received from WSN, average on this type of information is computed.

Data and Service Repositories provide a persistent storage of collected sensor data and service description in EIC. The sensor data repository is used for storage of sensor data for further data analysis or workload balancing in case of sensor nodes failure. And the service repository provides a description of the service available to business applications in EIC.

Collaborative services enable connectivity with WSN and fulfill interoperability requirements of EIC.

- **Data conversion**: This service converts on one hand exotic sensor readings in a standardized format for data processing and storage. On the other hand, the

service is also used to convert the internal data format to the format understandable by business applications. The service supports plugins, to add new conversion modules.

- **Generic connector**: This service meets the interoperability requirement with WSN middlewares, and connectivity with WSNs.
- **WSN discovery**: This service permits the EIC to discover new WSNs middlewares.

The list of these service is not exhaustive and can be extended accroding to business applications and EIC requirements.

4. RELATED WORK

We want to analyze existing WSN middleware systems and context-aware frameworks with regard to their capabilities to meet the requirements of an EIC defined in sections 2.2 and 2.3.

4.1 WSN middleware

The main purpose of middleware for sensor networks is to support the development, maintenance, deployment and executing of sensing-based applications [16]. In CoBIs [4], business applications are able to access functionalities provides by the sensor nodes via web services. The major tasks of the CoBIs middleware comprise the mediation of service requests between the application layer and the device layer. The focus lies thereby on life-cycle management of CoBIs nodes and deployment and discovery of required services. The RUNES project (Reconfigurable, Ubiquitous, Networked Embedded Systems) [18] uses a component-based middleware. Components are encapsulated units of functionality and deployment that can interact through "interfaces: and "receptables". The outside interfaces, needed to integrated into business applications, are provided by a number of so called communication services in order to support the publish/subscribe model, group communication, media streaming and RPC. The RUNES supports plug-in interaction paradigms (PIPs). Theses PIPs are described as generic APIs and can be implemented in accordance with the application needs. Agimone [10] is a higher-level middleware that supports the integration of WSNs and IP networks. It focuses on the distribution and coordination of WSN applications (Agilla mobile agents) across WSN boundaries. Agimone integrates the Agilla [8] and Limone [7] middleware platform. Despite Agimone's motivation by a cargo tracking application, Agimone is a general-purpose middleware with a uniform programming model for applications that integrates multiple WSNs and the IP network. Unfortunately integration into existing tracking or logistic applications are out of scope. The MiLAN middleware (Middleware Linking Applications and Networks)[21] focuses on sensor network management to enable proactive WSN applications, which support QoS requirements and energy constraints.

Interoperability is the main drawback of these middlewares. They first do not address the problem of heterogeneity of sensed data from WNSs. Middleware such as MILAN introduces the notion of generic connectors, which permits to gather sensor data from any type of nodes. Nevertheless, they do not consider the standardization of sensor data in a single format. Moreover, these middlewares do not propose any pre-processing of sensor data such apart from aggregation of sensor data. Fusion or reasoning about information is not addressed in these middlewares.

4.2 Context-aware middleware

Several architectures have been developed for supporting discovery, acquisition and reasoning about context information. But the technical means to deliver context-information cover only the part of the EIC requirements related to the data- and event-driven approach of delivering WSN data to business applications. The Context Toolkit Architecture[6] aims at facilitating development and deployment of context-aware applications. The basic components of this architecture are context widgets, interpreters, aggregators and a discoverer. The Context Broker Architecture (CoBrA)[2] proposes an architecture for discovery, acquisition and reasoning about context information. It includes also mechanisms for privacy protection of context information. CoBrA is based on a Web Semantic Ontology. CORTEX[20] is a context-aware middleware system that supports pervasive and ad hoc computing, similar to Gaia[15]. The flexible framework disseminates events in real-time and offers a number of diverse service discovery protocols. CORTEX is based on an anonymous and asynchronous event model that is made available through a publish/subscribe system. Context Fusion Center[1] propose a context-aware middleware which enable aggregation and inference on sensor data. They develop a peer-to-peer platform, *Solar*, which focuses on the issue of disconnection from WSN. PACE [12] is a application needs-driven middleware which address multiple business application requirements such as heterogeneity, mobility, scalability, privacy, traceability, control, tolerance to sensor nodes failure, and easy deployment and configuration of WSN. This middleware is probably the most advanced in the literature.

Even if these middlewares address partially our business application and WSN requirements, they often do not cover connectivity and interoperability with WSN. Additionally, they either propose proprietary interface to business applications, or do not support dynamic discovery of service supported by the middleware. To conclude on security and trust requirements, none of the WSN or context-aware middleware cope with the end-to-end security of sensor data from the WSN middleware to the business applications.

5. CONCLUSION

To conclude, it exists a wide variety of WSN middleware systems, which focus on different aspects of WSN application management and development. None of those middleware integrates a in-depth analysis of business applications in their design and rarely consider the integration with existing enterprise applications. As far as we know, this paper proposes the first deep analysis of the requirements of business application using WSN applications and providing guidelines for the design of such architecture.

In the WASP project, we plan to develop a prototype of the EIC which aims establish a secure bridge between WSN and business applications from healthcare, herd control and road transport. Based web service standards, such as WS-

Notification [13], for standardisation of communication with business applications or context ontology for standardisation of sensor data format, we aim at evaluating the performance of such approach. In addition, we will focus on security and trust related requirements from business applications and EIC.

6. ACKNOWLEDGEMENTS

This work is partially financed by the European Commission under the Framework 6 IST Project "Wirelessly Accessible Sensor Populations (WASP)".

7. REFERENCES

[1] G. Chen, M. Li, and D. Kotz. Design and implementation of a large-scale context fusion network. In *the Proceedings of the 1st Annual International Conference on Mobile and Ubiquitous Systems (MobiQuitous), IEEE Computer Society*, 2004.

[2] H. Chen. *An Intelligent Broker Architecture for Pervasive Context-Aware Systems*. PhD thesis, University of Maryland, Baltimore County, December 2004.

[3] C.-Y. Chong and S. P. Kumar. Sensor networks: evolution, opportunities, and challenges. *Proceedings of the IEEE*, 91(8):1247–1256, 2003.

[4] COBIS Consortium. COBIS. FP STREP Project IST 004270. *www.cobis-online.de*.

[5] P. Coy and N. Gross. 21 ideas for the 21st century. *Business Week*, pages 78–167, 1999.

[6] A. K. Dey. *Providing Architectural Support for Building Context-Aware Applications*. PhD thesis, College of Computing, Georgia Institute of Technology, December 2000.

[7] C.-L. Fok, G.-C. Roman, and G. Hackmann. A lightweight coordination middleware for mobile computing. In R. DeNicola, G. Ferrari, and G. Meredith, editors, *in the Proceedings of the 6th International Conference on Coordination Models and Languages (Coordination 2004)*, number 2949 in Lecture Notes in Computer Science, pages 135–151. Springer-Verlag, February.

[8] L. Fok, G.-C. Roman, and C. Lu. Mobile agent middleware for sensor networks: An application case study. In *in the Proceedings of the International Conference on Information Processing in Sensor Networks (IPSN'05), Special track on Platform, Tools, and Design Methods for Network Embedded Sensors (SPOTS)*, April 2005.

[9] Gartner, Inc. and/or its Affiliates. The Gartner Glossary of Information Technology Acronyms and Terms. 2004.

[10] G. Hackmann, C.-L. Fok, G.-C. Roman, and C. Lu. Agimone: Middleware support for seamless integration of sensor and ip networks. In *International Conference on Distributed Computing in Sensor Systems (DCOSS'06)*, June 2006.

[11] S. Hadim and N. Mohamed. Middleware challenges and approaches for wireless sensor networks. *IEEE Distributed Systems Online*, 7(3), 2006. art. no. 0603-o3001.

[12] K. Henricksen, J. Indulska, T. McFadden, and S. Balasubramaniam. Middleware for distributed context-aware systems. In *in the Proceedings of the International Symposium on Distributed Objects and Applications (DOA)*, 2005.

[13] OASIS. Web Service Notification TC. 2004.

[14] D. Preuveneers, J. V. den Bergh, D. Wagelaar, A. Georges, P. Rigole, T. Clerckx, Y. Berbers, K. Coninx, V. Jonckers, and K. D. Bosschere. Towards an extensible context ontology for ambient intelligence. In P. Markopoulos, B. Eggen, E. H. L. Aarts, and J. L. Crowley, editors, *EUSAI*, volume 3295 of *Lecture Notes in Computer Science*, pages 148–159. Springer, 2004.

[15] M. Roman, C. Hess, R. Cerqueira, A. Ranganathan, R. Campbell, and K. Nahrstedt. Gaia: A middleware infrastructure to enable active spaces. *IEEE Pervasive Computing*, 1(4):74–83, 2002.

[16] K. Römer, O. Kasten, and F. Mattern. Middleware challenges for wireless sensor networks. *ACM Mobile Computing and Communication Review (MC2R)*, 6(4):59–61, October 2002.

[17] W. Roush. 10 emerging technologies that will change the world. *Technology Review*, 106(2), 2003.

[18] RUNES Consortium. RUNES, IST-004536-RUNES. *www.ist-runes.org*.

[19] F. Siegemund, C. Floerkemeier, and H. Vogt. The value of handhelds in smart environments. *Personal and Ubiquitous Computing*, 9(2):69–80, March 2005.

[20] C.-F. Sørensen, M. Wu, T. Sivaharan, G. S. Blair, P. Okanda, A. Friday, and H. Duran-Limon. Context-aware middleware for applications in mobile ad hoc environments. In *ACM/IFIP/USENIX International Middleware conference 2nd Workshop on Middleware for Pervasive and Ad-Hoc Computing (online proceedings)*, Toronto, Canada, Oct. 2004.

[21] H. C. W. Heinzelman, A. Murphy and M. Perillo. Middleware to Support Sensor Network Applications. *IEEE Network Magazine Special Issue*, Jan. 2004.