

Modeling of Sensor Data and Context for the Real World Internet

Claudia Villalonga^{*†}, Martin Bauer[‡], Vincent Huang[§], Jesús Bernat[¶] and Payam Barnaghi^{||}

^{*}*SAP Research, CEC Zürich, Switzerland. Email: claudia.villalonga@sap.com*

[†]*Wearable Computing Lab., ETH Zürich, Switzerland*

[‡]*NEC Laboratories Europe, NEC Europe Ltd., Germany. Email: martin.bauer@nw.neclab.eu*

[§]*Ericsson Research, Ericsson AB, Sweden. Email: vincent.a.huang@ericsson.com*

[¶]*Telefónica Investigación y Desarrollo, Spain. Email: bernat@tid.es*

^{||}*Centre for Communication Systems Research, University of Surrey, UK. Email: p.barnaghi@surrey.ac.uk*

Abstract—The Internet is expanding to reach the real world, integrating the physical world into the digital world in what is called the Real World Internet (RWI). Sensor and actuator networks deployed all over the Internet will play the role of collecting sensor data and context information from the physical world and integrating it into the future RWI. In this paper we present the SENSEI architecture approach for the RWI; a layered architecture composed of one or several context frameworks on top of a sensor framework, which allows the collection of sensor data as well as context information from the real world. We focus our discussion on how the modeling of information is done for different levels (sensor and context data), present a multi-layered information model, its representation and the mapping between its layers.

Keywords—sensor data modeling; context modeling; Real World Internet (RWI); Sensor and Actuator Networks (SANs);

I. INTRODUCTION

The Internet is evolving towards a content-centric Future Internet [1] where users not only consume data but also create data, e.g. text, audio and video. The current Internet does not provide a global solution to access and sense the physical world. However, future applications and services will require information from the real world and interact with it. More natural and unobtrusive human-machine interactions and true real world awareness can only be achieved, if information concerning the physical world can be captured and shared in an automatic fashion.

Sensor and actuator networks (SANs) [2] will play an important role at the edges of the Future Internet in order to add this real world dimension to the current Internet [3]. SANs represent an inexhaustible resource for real world information; ubiquitously deployed, they can capture a diverse set of physical phenomena and real world events characterized by multiple sensing modalities and they can also enable localized interactions.

The realization of the vision of the Real World Internet (RWI) requires: 1) digitising observed real world phenomena and feeding this information into global network infrastructure, i.e. the Internet, 2) modeling of this information at different levels of granularity to make it interpretable inside the RWI infrastructure and by interested users, 3) processing and interpretation of this measured or observed information

in order to derive real world context that can be delivered to interested users.

In this paper we present SENSEI, an architecture for the RWI (Section II), describe in detail the modeling of the information which is collected, processed and transmitted inside the SENSEI framework (Section III) and draw some conclusions about the challenges of modeling information in such a layered RWI architecture (Section IV).

II. SENSEI: AN ARCHITECTURE FOR THE RWI

As part of the European ICT-FP7 SENSEI project¹, an architecture for the Real World Internet has been developed. This architecture is based on the current and future Internet principles and integrates heterogeneous SANs into a large scale framework enabling an open marketplace for real world context and actuation applications. One of the design principles of the SENSEI architecture is decoupling context from the sensor data (see Section II-A) but integrating both dimensions in a single architecture (see Section II-B).

A. SENSEI Principle

Bringing real world information to the Internet has been studied from two different perspectives: Sensor and Actuator Networks (SANs) and Context-awareness [4]. These two different paradigms instead of producing a complementary approach have ended up creating divergent architectures to provide real world information services.

The SAN paradigm has been studied at two levels. First, the so called middleware for sensor networks [5] focused on providing rich functionalities at the sensor network plane that allow applications to be developed using a particular sensor network. And second the sensor frameworks that manage the open issues of deploying sensor based services on multiple heterogeneous and geographically dispersed sensor networks. Some examples of sensor frameworks are IrisNet [6], jWebDust [7], SenseWeb [8], and Global Sensor Networks (GSN) [9]. Each of them provides a different solution to address part of the problems and requirements of a wide scale SAN.

¹<http://www.sensei-project.eu/>

A substantial advance over the sensor frameworks is provided by the context frameworks, which collect, gather, process and deliver context information. Context can be defined as information about users (e.g. bio-physiological conditions, emotions, habits, social interaction, activity) [10], but also as the attributes that characterize the situation of an entity, such as location, status or activity and time [11]. Some examples of context frameworks are Context Toolkit [11], Gaia [12], MAGNET Beyond [13] or MobiLife [14].

The SENSEI project had to face the decision on what to offer to services, context information or sensor data. It is easier from an application point of view to ask for a property of an entity than to ask for the value provided by a sensor, i.e. temperature of the Room_987 than temperature of sensor number 10. However, context is more application specific and it cannot be assumed that a single context model will be able to provide information for all possible application domains. In order to maximise the advantages of both approaches, the SENSEI architecture provides both context and sensor information by demonstrating a decoupled context framework on top of the services provided by a single sensor framework, in such a way that multiple context frameworks could be built on top of the sensor framework. The main advantages of this approach are that it facilitates horizontalisation of the underlying SENSEI resources and caters for the needs of heterogeneous resource users, while enabling the evolvability of the system.

B. SENSEI Architecture

Resources are the core concept of the SENSEI architecture shown in Figure 1 (for further specification see [3]). Sensors, processors, or actuators that provide information about the real world or allow interaction with it are *Resources*.

Resources are made available to the SENSEI Framework through *Resource Endpoints (REP)* that implement one or several *Resource Access Interfaces (RAI)*. The RAI for different resources is not homogeneous, each resource provider might offer different kinds of interfaces, using different technologies (e.g. REST or WS) and implementing different interaction patterns (synchronous and asynchronous, publish-subscribe or simple query). An essential point for the scalability of the SENSEI architecture is that sensor and processing resources only communicates real world information if this is requested via their RAIs.

The Resource Descriptions, which describe the resource, i.e. the information it provides, the task it performs and how to access it (its RAIs and its REPs descriptions), are stored in the *Resource Directory (RD)* which implements a simple rendezvous functionality. The Resource concept and the unified Resource Descriptions provide the basis for homogeneous discovery and access to a heterogeneous substrate of SANs. Together the *Resources*, *Resource Users* and the *RD* are the core components of the SENSEI architecture and form the SENSEI sensor framework. This

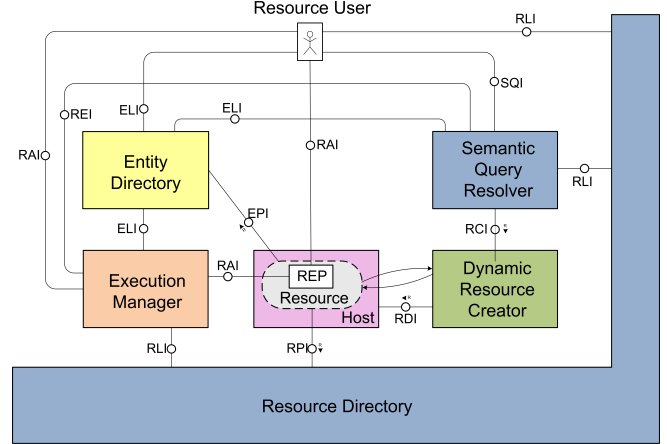


Figure 1. Resource Layer Architecture.

core architecture only provides a simple lookup functionality to applications. The burden of having to implement the interaction with heterogeneous resources is on the *Resource Users*. Moreover, *Resources* typically provide sensor level information, i.e., observations and measurements, which are not contextualized. This does not correspond to the abstraction level and functionality expected from a context framework. Therefore, we have defined the SENSEI advanced architecture components, *Semantic Query Resolver (SQR)*, *Execution Manager (EM)*, *Dynamic Resource Creator (DRC)*, and *Entity Directory (ED)*, that provide the additional functionalities of a context framework.

The *ED* provides the link between entities of the real world as specified in the context model and *Resources* providing information about aspects of these entities. This forms the basis for first finding the relevant *Resources* and then contextualizing the sensor-level information these *Resources* may provide. The *SQR* analyzes requests and creates execution plans. It utilizes information from the *RD* and *ED* to find the appropriate *Resources*. If *Resources* do not exist yet, it may be possible to dynamically create them, using the *DRC*. Optionally, execution plans can be passed to the *EM* for setting up the execution on behalf of the *Resource User*, so that the *Resource User* directly gets the requested information.

III. INFORMATION MODELING IN SENSEI

A Real World Internet infrastructure needs to model the measured or observed information about physical phenomena and the context information about entities of the real world in order to enable the interpretation of this information in the framework as well as by consumer applications and services. The SENSEI Information Model (see Section III-A) follows the basic SENSEI design principle, is based on a layered approach for sensor data and context information, and facilitates advantages like horizontalisation, heterogeneity and evolvability as described in Section II-A.

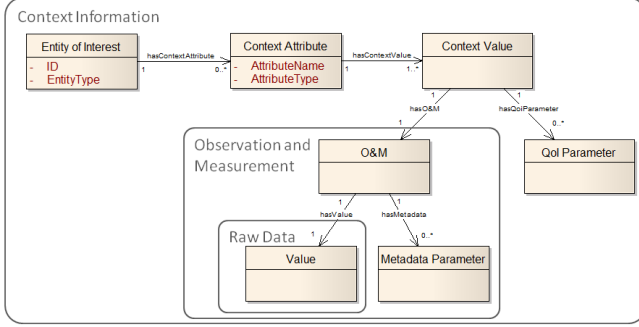


Figure 2. SENSEI Information Model.

Such a model can be implemented using several representations, e.g. a semantic representation based on RDF (see Section III-B). Having a layered model requires the definition of the mapping between the context and the sensor layers (see Section III-C).

A. SENSEI Information Model

Information in the SENSEI framework ranges from the raw data obtained from sensor nodes, the observation and measurement containing not only a value but also associated meta information about the data required by simple applications, to the high-level context required by more sophisticated applications and services. The coherent modeling of information on these three different abstraction levels is one of the main challenges of the SENSEI Information Model. The same way that the SENSEI architecture implements a context framework on top of a sensor framework, the SENSEI Information Model also follows a layered approach: on top of the raw data an observation and measurement can be built and this can be used in the upper layer as part of the context information. Figure 2 shows the three layers of the Information Model: Raw Data, Observation and Measurement (O&M) Data and the Context Information.

1) *Raw Data*: The lowest layer of the Information Model contains the value that has been observed or measured by a resource, i.e. the raw data delivered by a sensor node.

2) *Observation and Measurement (O&M)*: Sensor frameworks need to interpret the data sensed from the physical world, i.e., it is not enough that a temperature sensor returns 25.5, if the application does not know that this value is measured in degrees Celsius and that it represents a temperature. Therefore, raw sensor data can be enhanced with metadata, e.g. information about the units, the resource that observed or measured the data, the quality associated to the data or any other relevant information. The O&M layer of the Information Model takes this role allowing resources in the SENSEI framework to deliver not only the raw data but also other relevant information attached to it. Therefore, an O&M contains the *Value* and zero or more *Metadata Parameters* which define the value. The SENSEI O&M is similar to the Sensor Web Enablement (SWE) data model

[15] and the Semantic Sensor Web model [16], but the main difference is that O&M in SENSEI allows the addition of any type of metadata to the observed or measured value, making it more flexible and extensible.

3) *Context Information*: The higher level of the Information Model deals with the modeling of context information about real world entities required by context frameworks, e.g. the temperature of Room_987 is 25.5°C. Based on Dey's definition of context [11], the SENSEI context model represents any physical or virtual entity of the world, e.g. people, places or objects, as an *Entity of Interest* (EoI) that has associated several context attributes. For example an EoI of type room could have attributes like location or temperature. Since several resources can provide information about the same attribute of an EoI, each context attribute can take several values which leads to information redundancy but also to conflicting context values [17]. The model captures this context redundancy and adds quality information about the context to allow methods to resolve context conflicts.

The context model contains three main concepts: an *Entity of Interest*, a list of *Context Attributes* describing the properties of an *Entity of Interest* and one or more *Context Values* associated to a *Context Attribute* of an *Entity of Interest*. The *Context Value* contains the O&M provided by the resource plus some optional *QoS Parameters* that describe to what extent the context corresponds to the real world. The context model links the two different layers of the Information Model, context and O&M, and adds some quality measure to the *Context Value*, the *QoS Parameters*, which are different from the ones associated to the O&M, if the resource is not the EoI itself or if there is some uncertainty between the association of the resource providing the O&M and the EoI. The SENSEI context model, like other available context models [12], [17], [18], [19], follows an entity-centric approach allowing the efficient context search based on the entity type or identifier. However, if other requirements should be fulfilled, another context framework on top of the SENSEI sensor framework could define a different context model e.g. an observation-centric model, which further extends the O&M adding the EoI and attribute as metadata.

B. Information Representation

The SENSEI Information Model could be implemented using different representations, e.g. XML or ontology-based data. Whereas, XML descriptions do not provide the full potential for machines to acquire and interpret the emerging semantics from data, an ontology-based data representation solves these problems and enables advanced analysis and enhanced data processing for heterogeneous sensor network applications. Taking into consideration the advantages of an ontology-based data representation, we use in SENSEI and present in this paper an RDF encoding of the Information Model, however this is not the only option and other types of representation such as OWL would also be possible.

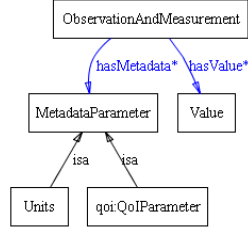


Figure 3. Ontology-based representation of O&M data

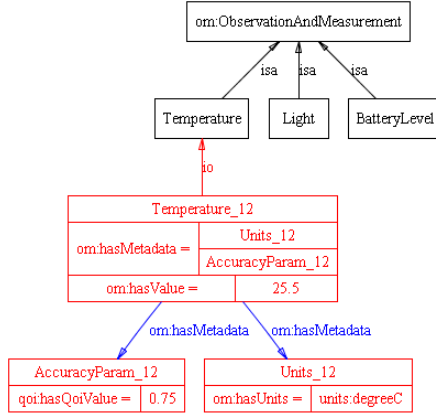


Figure 4. Example of a domain ontology for O&M data and example of an instance of an O&M temperature of 25.5°C with accuracy of the 0.75

1) *Observation and Measurement (O&M)*: An ontology-based representation of the O&M is shown in Figure 3. The *ObservationAndMeasurement* class has associated two properties *hasMetadata* and *hasValue*. The *hasValue* property has as range a class *Value* which is a subclass of *rdfs:Literal* and contains the raw data delivered by the sensor. The *hasMetadata* property has as range the *MetadataParameter* which contains the metadata associated to the value and has two important subclasses the *Units* and the *qoi:QoiParameter*. The *Units* class specifies the measurement units and for reusability purposes links through the *hasUnits* property to the class *units:Unit* of NASA's SWEET ontology². The *qoi:QoiParameter* class is part of the Quality of Information ontology which is still under development and which will most likely contain several subclasses of the *qoi:QoiParameter* for three different dimensions, time, space and reliability. For each subclass of *qoi:QoiParameter* the metrics to measure the quality will be described, however, for the purpose of this paper, we only define the *hasQoiValue* property which has *qoi:QoiParameter* as domain and *rdfs:Literal* as range and the *qoi:Accuracy* subclass of *qoi:QoiParameter*.

The O&M ontology is generic and has to be further specified for each domain. Figure 4 shows a domain ontology used in SENSEI to describe the *Temperature* and the *Light* level

measured by a sensor node and the *BatteryLevel* of a sensor node as subclasses of the *ObservationAndMeasurement*. Further O&M are described in the SensorData Ontology³.

2) *Context Information*: The ontological representation of the context model is shown in Figure 5. An EoI is represented through the *EntityOfInterest* class and its Context Attributes through the *hasContext* property which has as domain the *EntityOfInterest* class and as range the *ContextValue* class. The *ContextValue* class, which represents a Context Value, is associated with two properties *hasOandM* and *hasQoiParameter*. The *hasOandM* property has as domain the *ContextValue* class and as range the *om:ObservationAndMeasurement* class and is the link between the context model and the O&M data. The *hasQoiParameter* property has as domain the *ContextValue* class and as range the *qoi:QoiParameter* class and describes the quality of the Context Value for an Attribute of an EoI.

For each domain, the context ontology has to be further specified to describe the concepts of the real world that are supported by the framework; a simple example is shown in Figure 6. The specializations of Entities of Interest are modeled in a hierarchical way as subclasses of the *EntityOfInterest* class, e.g. a *Room* is a subclass of the *Place* class which is a subclass of *EntityOfInterest*, and the *Object* class and the *Person* class are also subclasses of *EntityOfInterest*. Context Values define concrete types of information by subclassing *ContextValue* and have to map to their equivalent *om:ObservationAndMeasurement* type. This mapping is done by creating a subproperty of *hasOandM* which links the class in the context ontology with the class in the O&M ontology; e.g. the *hasOMTemperature* subproperty links the *CtxTemperature* class in the context ontology with the *om_domain:Temperature* class in the O&M ontology. Context Attributes are further specified by defining subproperties of the *hasContext* property and setting the appropriate ranges and domains to these subproperties; e.g. *hasTemperature* is a subproperty of *hasContext* which has the *CtxTemperature* class as range and the *Room* class as part of its domain.

Figure 6 shows a context instance for the temperature of Room_987. Supposing that the resource of the previous section, the temperature sensor, has provided the instance of temperature *om_domain:Temperature_12*, this O&M needs to be transformed into context information. In this process the *om_domain:Temperature_12* is wrapped into *CtxTemperature_12*, an instance of the *ContextValue* of type temperature. The new *QoiParameters* for the piece of context information are calculated, e.g. the accuracy is calculated as a function of the accuracy of the O&M and the reliability that the temperature sensor provides information about the temperature of Room_987. Once the accuracy has been calculated, the *AccuracyParamCtx_12* is created and added to the *CtxTemperature_12* instance through the

²Available at: <http://sweet.jpl.nasa.gov/ontology/>

³Available at: <http://purl.oclc.org/net/unis/ontology/sensordata.owl>

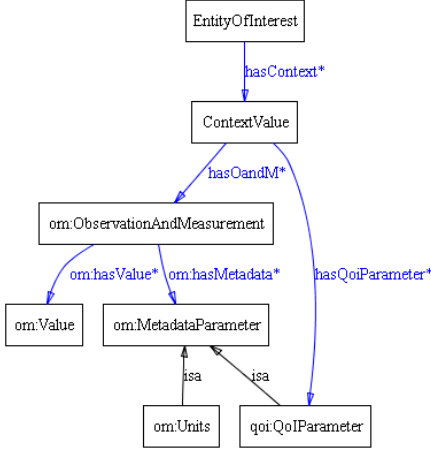


Figure 5. Ontology-based representation of Context Information

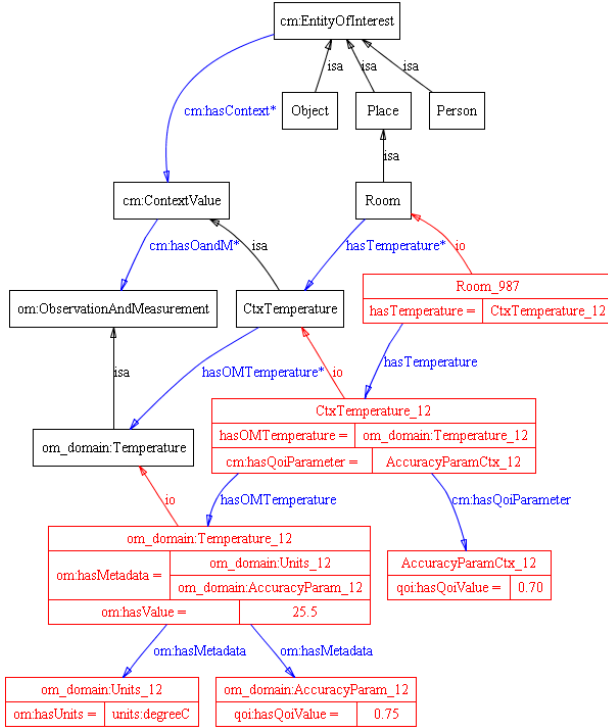


Figure 6. Example of a domain ontology for Context Information and example of a contextualized instance of the temperature of Room_987

hasQoiParameter property. And finally to complete the context representation, an instance of *Room_987*, an EoI of type *Room*, is created and the association between this instance and the Context Value *CtxTemperature_12* is done through the *hasTemperature* property which represents the temperature Context Attribute.

C. Mapping of the submodels

The SENSEI Information Model provides different layers of abstraction and it is possible to convert information from

one to another through mappings and transformations.

Data from sensor resources follows the O&M model and the contextualized information is not generally part of the O&M data since a resource might not be associated to any EoI or the relation between the resource and the EoI is dynamic. The mapping of O&M data into context information is enforced by the SENSEI framework at information delivery time by assigning to the context attribute value the O&M provided by either sensor resources or processing resources. For example, the average temperature is an attribute of a room and its value comes from averaging all sensor readings in the room, a processing resource performs the averaging task and provides this value to a framework resource which links it to the EoI. Therefore, the links between EoI attributes and resources which can provide the attribute values have to be known. Identifying and storing these links is the job of *Entity Directory* in the SENSEI architecture. For each context request, the SENSEI framework instantiates the relevant context instances making use of the O&M data delivered by resources. The framework uses the *Entity Directory* to find out which resources can provide information for the requested entity attributes. The O&M data is fetched from the resources and mapped to the entity-based context model. In the case that no resource is associated to an entity attribute, the framework first tries to use available context information and resource descriptions to establish new associations, and if this step fails, the framework dynamically creates a new resource which can provide the requested information.

Some sensor resources might not provide information in the O&M form and therefore a gateway resource has to transform the Raw Data delivered by the sensor into an O&M. In this case, sensor nodes send the data as raw binary data and the gateway resource, which has more processing and power capabilities, adds the meta-data. This addresses the limitations of the sensor nodes in terms of processing, bandwidth and air interface that is required to send large amount of annotated data.

IV. CONCLUSIONS

This paper describes the modeling of sensor data and context for a Real World Internet architecture in the SENSEI project. We use high-level models to make information provided by heterogeneous resources, e.g. global SANs, easily and efficiently accessible to the different stakeholders of the Real World Internet. Flexible and interoperable information communication using semantically annotated information over the proposed architecture will support various services and applications to automatically interpret and process status of real world entities. The context descriptions and annotated information can be utilized to acquire knowledge for intelligent applications in different domains.

The presented information model describes the O&M emerging from sensors and which uses metadata to enhance

the sensor data. The information model extends the specifications to higher-level entities and context descriptions. The context model utilizes the O&M in order to describe the situation and state of attributes of the entities of interest in different Real World Internet application scenarios. The model employs semantic descriptions of O&M information and uses context attributes to associate the O&M information to the domain concepts. The model is described and serialized in RDF format. However, the underlying design supports other types of representations and different domain ontologies can be also plugged to the structure.

The novel layered approach for context and sensor information modeling presented in this paper, as well as the architecture supporting the search, collection, processing and delivery of this information, provides the benefits of both sensor frameworks and context frameworks. As such the information model builds upon a common sensor information model which serves as underpinning to support different context models. This model is easily extensible and evolvable, therefore it does not only tackle the SENSEI scope but the Real World Internet in general, and could be applied to other domains required by future needs.

The future work will focus on integrating the model to the current SENSEI prototype and evaluating the information exchange between different resources and interpretation of the represented context information in a SENSEI framework application scenario.

ACKNOWLEDGMENT

This paper describes work undertaken in the context of the SENSEI project, Integrating the Physical with the Digital World of the Network of the Future (www.sensei-project.eu). SENSEI is a Large Scale Collaborative Project supported by the European 7th Framework Programme, contract number: 215923.

REFERENCES

- [1] P. Daras, D. Williams, C. Guerrero, I. Kegel, I. Laso, J. Bouwen, J.-D. Meunier, N. Niebert, and T. Zahariadis, "Why do we need a Content-Centric Future Internet? Proposals Towards Content-Centric Internet Architectures," European Commission, Networked Media Unit, Information Society and Media, May 2009.
- [2] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey," *Computer Networks*, vol. 38, no. 4, pp. 393–422, 2002.
- [3] A. Gluhak, M. Bauer, M. Johansson, J. Bernat, V. Stirbu, F. Montagut, and M. Presser, *Towards an Architecture for a Real World Internet, Towards the Future Internet A European Research perspective*. IOS Press, 2009.
- [4] D. Raz, A. T. Juhola, J. Serrat-Fernandez, and A. Galis, *Fast and Efficient Context-Aware Services*. John Wiley & Sons, 2006.
- [5] L. J. Wang M., Cao J. and D. S. K., "Middleware for wireless sensor networks: A survey," *Journal of Computer Science and Technology*, vol. 23, no. 3, pp. 305–326, May 2008.
- [6] P. Gibbons, B. Karp, Y. Ke, S. Nath, and S. Seshan, "IrisNet: An architecture for a Worldwide Sensor Web," *IEEE Pervasive Computing*, vol. 2, no. 4, pp. 22–33, Oct.-Dec. 2003.
- [7] I. Chatzigiannakis, G. Mylonas, and S. Nikolettseas, "The design of an environment for monitoring and controlling remote sensor networks," *International Journal of Distributed Sensor Networks*, vol. 5, no. 3, pp. 262–282, 2009.
- [8] A. Kansal, S. Nath, J. Liu, and F. Zhao, "SenseWeb: an infrastructure for shared sensing," *IEEE MultiMedia*, vol. 14, no. 4, pp. 8–13, 2007.
- [9] K. Aberer, M. Hauswirth, and A. Salehi, "Infrastructure for data processing in large-scale interconnected sensor networks," in *Mobile Data Management*, 2007, pp. 198–205.
- [10] A. Schmidt, M. Beigl, and H.-W. Gellersen, "There is more to context than location," *Computers & Graphics*, vol. 23, no. 6, pp. 893–901, 1999.
- [11] A. K. Dey, G. D. Abowd, and D. Salber, "A conceptual framework and a toolkit for supporting the rapid prototyping of context-aware applications," *Human-Computer Interaction*, vol. 16, no. 2, pp. 97–166, 2001.
- [12] M. Román, C. K. Hess, R. Cerqueira, A. Ranganathan, R. H. Campbell, and K. Nahrstedt, "A middleware infrastructure for active spaces," *IEEE Pervasive Computing*, vol. Vol. 1, No. 4, pp. 74–83, 2002.
- [13] M. Jacobssen (Editor), "Specifications of PN networking and security components," MAGNET Beyond, Public Deliverable D2.3.1, 2006.
- [14] M. Klemettinen, *Enabling Technologies for Mobile Services: The MobiLife Book*. Wiley Publishing, 2007.
- [15] M. Botts, G. Percivall, C. Reed, and J. Davidson, "OGC Sensor Web Enablement: Overview and High Level Architecture," OGC, Tech. Rep., December 2007.
- [16] A. Sheth, C. Henson, and S. S. Sahoo, "Semantic Sensor Web," *IEEE Internet Computing*, vol. 12, no. 4, pp. 78–83, 2008.
- [17] K. Henriksen and J. Indulska, "Modelling and using imperfect context information," in *PERCOMW '04: Proceedings of the Second IEEE Annual Conference on Pervasive Computing and Communications Workshops*, 2004, p. 33.
- [18] X. H. Wang, D. Q. Zhang, T. Gu, and H. K. Pung, "Ontology Based Context Modeling and Reasoning using OWL," in *PERCOMW '04: Proceedings of the Second IEEE Annual Conference on Pervasive Computing and Communications Workshops*, 2004, p. 18.
- [19] Y. Bu, T. Gu, X. Tao, J. Li, S. Chen, and J. Lu, "Managing quality of context in pervasive computing," in *QSIC '06: Proceedings of the Sixth International Conference on Quality Software*, 2006, pp. 193–200.