Semantic Interactions in the Internet of Things

Paul E. Estrada-Martinez

Department of Computer Science,
CICESE Research Centre,
Ensenada, BC 22860, Mexico
E-mail: pestrada@cicese.mx

J. Antonio Garcia-Macias*

Department of Computer Science,
CICESE Research Centre,
Ensenada, BC 22860, Mexico
E-mail: jagm@cicese.mx
*Corresponding author

Abstract: The long-term vision of the Internet of Things (IoT) is to extend the reach of the Internet to any object we can imagine. This new level of connectivity and the presence of smart objects in our environments will make possible systems and applications requiring new kind of interactions. In this paper we present our proposal of semantic interactions for addressing interaction problems in the IoT. We describe some possible scenarios of the IoT where users interact with smart objects in smart spaces. We explain how semantic web technologies can be used for creating richer interactions and also present the evaluation of our proposal by applying the concepts of this work in a system prototype.

Keywords: semantic interactions; smart objects; smart spaces; internet of things.

Reference to this paper should be made as follows: Estrada-Martinez, P. E. and Garcia-Macias, J. A. (2012) 'Semantic Interactions in the Internet of Things', *Int. J. Ad Hoc and Ubiquitous Computing*, Vol. 9, No. 3/4, pp.1–7.

Biographical notes: Paul E. Estrada-Martinez holds a M.Sc. in Computer Science from CICESE Research Centre. His research interests are in semantic communications, new user interfaces for the Internet of Things and human-computer interaction. Presently, he is preparing a proposal for his Ph.D.

J. Antonio Garcia-Macias is a full-time researcher at CICESE's Department of Computer Science. He holds a Ph.D. in Computer Science from Institut National Polytechnique de Grenoble, France. His research interests are ubiquitous computing, human-computer interaction and computer networks.

1 Introduction

The Internet of Things (IoT) has been envisioned as a future extension of the Internet in which all kind of physical objects will be interconnected through Internet communication protocols (Gershenfeld et al., 2004). This extension relies in creating a new wave of computing devices and consumer products by augmenting everyday common objects (e.g., books, home appliances, food products, medicines, furniture) with various technologies such as sensors, actuators, processors and communication units. This way, objects augmented with technology become "smart objects" capable of sensing, processing information, providing services and communicating with other objects.

The presence of smart objects in our environment will increase and diversify over time. As a result, there will be smart spaces where people can interact with heterogeneous smart objects and use their services.

Nevertheless, creating an IoT poses a variety of challenges that are shaping three visions of the IoT as a paradigm (Atzori et al., 2010):

1. Things-oriented: It addresses the design and creation of smart objects, exploring new forms of interaction in smart spaces as well as the development of IoT applications.
2. Internet-oriented: This vision is related with communication protocols, architectures, network technologies for smart objects and services that harness this technological infrastructure.
3. Semantic-oriented: It involves research about representation, storage, interconnection, searching

and organization of information generated by smart objects and smart spaces aided by semantic technologies.

The main contribution we present in this paper is to address interaction problems in the IoT by using semantic web technologies. With this approach, ontologies can be used to model information about objects and their relationships. Moreover, ontologies are a powerful tool for contextual information modelling, encompassing not only modelling but also information reasoning.

The interaction problems we seek to address can be stated in the following questions: when a person enters a smart space, how to automatically know which smart objects are present and which ones are they? How to know the type and format of information provided by smart objects? How to determine what type of interactions are possible with a smart object? Beyond identifying certain characteristics of individual objects, our work also addresses physical spaces where systems are formed with collections of smart objects. With these kinds of smart spaces it is possible to have context-aware and proactive systems that anticipate user needs. Hence, we are exploring how semantic information can enable richer interactions for IoT applications and systems.

The remainder of this paper is organised as follows: In section 2 we exemplify possible interactions in smart spaces through some scenarios. In section 3 we describe semantic interactions as a means to address interaction problems in smart spaces. In section 4 we present the experimental infrastructure we have developed, and use this to describe, in section 5, some tests of semantic interactions with an experimental setup. Section 6 discusses some related work. Finally, in section 7 we make some conclusions and talk about future work.

2 Scenarios

We now present some scenarios involving smart spaces and illustrate the interaction problems we address in this paper (see Figure 1). Scenarios 2.1 and 2.2 describe interaction with smart objects by means of direct manipulation and implicit interaction. Scenario 2.3 exemplifies a device mediated interaction.

2.1 Smart gym

Tom is a 30 year old man who likes to stay physically fit, thus he trains at the gym every day after finishing his work activities. However, the gym where he goes is not a common one. It is a smart gym where all the weight bars and exercise equipment are smart objects. Also, the walls of the gym are mirrors that act as displays.

When Tom went to the smart gym for the first time, he had to register his profile data in the gym control system. This was easily done by talking with a virtual avatar in a mirror. Since then, Tom just enters the gym and begins using any equipment. The frontal mirror detects the presence of Tom and act as a display, showing a virtual avatar that teaches the execution of the exercise if the user wants to. Furthermore, the mirror shows information about Tom's weight, applied force and the exact movement of his body. After Tom finishes the exercise, the mirror shows recommendations about what to do next.

This behaviour is possible because the gym control system acts according to information gathered from user profile and sensory data collected from weight bars and other equipment utilised by the user.

2.2 Smart kitchen

Ana is a woman who loves to cook. Recently, she bought a smart kitchen module at a home automation store. Once the technicians finished mounting the kitchen at Ana's house and configured her profile, she was eager to begin cooking. The smart kitchen included a stove, a refrigerator, a microwave oven, a cutlery set and some casserole pots, all of which are smart objects with sensing and communication capabilities.

Ana decides to prepare some food, thus she takes a piece of meat from the refrigerator. Automatically, the refrigerator identifies the item taken away from its interior and advertises the event in the kitchen network. This way, interested smart objects are free to react accordingly to events. In this case, a knife identifies the event and waits for user input (i.e., the user takes the knife and puts it in position to cut). To prevent accidents with kids, the knife normally has its edge hidden inside, but once it detects the expected user input, the knife pulls out its edge to permit Ana begin cutting the meat.

Meanwhile, all the smart objects in the kitchen are waiting for stimuli to react. Then, Ana lays the knife on the table, takes a casserole pot and puts the meat pieces inside. This action triggers the activation of the stove and the microwave (i.e., turning on their display). Ana wants to use the microwave, thus she walks towards it carrying the casserole. Her movements cause a proximity interaction that provokes the opening of the microwave door. At the same time, the stove turns off its own display and Ana is now ready to cook the meat.

2.3 Smart garden

Ms Green is a 65 year old retired woman who enjoys watering the garden and taking care of her flowerpots in the mornings. Ms Green often forgets the assigned day for watering a particular flowerpot because now she has many of them. Aware of the situation, her son gave her some flowers in a special pot as a gift for her birthday. These flowers require special care since sunlight overexposure and humidity loss are very harmful for them.

This is not a problem because she has a new mobile phone with an IoT browser, and the special flowerpot comes equipped with humidity and temperature sensors.

Ms Green utilises her mobile phone while being in the garden and points with the camera towards a flowerpot. If the flower has an optimal level of humidity the mobile phone displays a picture of a happy face. Also, if a flower needs water, then the mobile phone displays a picture of a sad face. Furthermore, Ms Green can verify the exact level of humidity and temperature in the flowers with just pressing a button.

During the summer, Ms Green tries to stay away from the sunlight and heat by staying less time in the garden. Therefore, she utilises her mobile phone to monitor her flowers from inside the house. Ms Green activates a function on her mobile phone and it starts a connection to her smart garden system. Now she can verify the condition of her flowers without going outside.

3 Semantic interactions

As we exemplified with the previous scenarios, the IoT is not only about individual heterogeneous objects but also about smart spaces conformed by proactive systems. This heterogeneity represents a main challenge for enabling interactions in the IoT because smart objects can provide different information in different formats that can be useful for different purposes.

To face this problem we are exploring semantic web technologies[1] applied to interactions, which we refer as "semantic interactions". We now explain the use of semantic information below.

3.1 Ontologies

Ontology is an explicit specification of a conceptualization that defines concepts, relationships and other elements that are relevant for modelling a domain (Gruber, 1993). This specification takes the form of definitions in a vocabulary (e.g., classes, relationships, attributes), which provide a meaning for the vocabulary and formal restrictions for its usage. RDF Schema, SKOS, OWL and RIF are W3C standards for describing different forms of vocabularies[2]. In particular, OWL is based on description logics and provides different levels of expressiveness that are not possible to achieve with other languages.

Ontologies are important elements to handle semantic information because they serve as a common schema for representing information from different domains. For instance, they can be used for modelling individual objects and services of smart spaces. Using ontologies, smart objects and IoT applications may be capable of determining the type and characteristics of entities available in a smart space. In Figure 2 we show the representation of a smart object and its services with an ontology graph.

3.2 Semantic models

In order to exploit the expressive power of ontologies, we need to semantically annotate smart objects data to create linked data with a meaning. This is where semantic models are needed.

A semantic model is a document aimed at describing web resources in a conceptual manner. The W3C provides technologies such as RDF, GRDDL, RDFa and POWDER for representing semantic models and get access to their data[3]. We can use RDF semantic models to associate ontology concepts with isolated data in smart spaces. In Figure 3 we show the semantic model for a smart object.

We must consider that the IoT will provide a platform for new kinds of systems and applications. Therefore, semantic models can be a good alternative for sharing data between smart objects, allowing heterogeneous data generated in smart spaces to be collected and processed by interested entities.

3.3 Inferences and semantic queries

It is possible to identify the ontologies used in a semantic model by analysing the header of the model. Hence, an application using semantic technologies is able to read a model and derive a conclusion from the relationships and properties defined in the ontology. Moreover, ontologies can be augmented with inference rules to provide more powerful reasoning capabilities. SWRL (Horrocks et al., 2004) is a language that allows the definition of rules with OWL concepts. In Table 1 we show some rules that could be applied in the smart garden scenario.

Another possibility is to perform semantic queries using information derived from ontologies and semantic models. For instance, SQWRL (O'Connor and Das, 2009) is an SWRL-based language for retrieving knowledge from ontology, and SPARQL[4] is a W3C standard for executing queries in RDF semantic models. In Table 2 we present some semantic queries suitable for the scenarios described in this work.

3.4 Interactions

From previous sections it can be observed that the IoT is a mix of heterogeneous smart objects and systems that can be augmented with semantics, adding a layer of linked data with the purpose of facilitating interoperability. We believe it is possible to take advantage of this semantic layer for exploring novel interactions in smart spaces.

Smart objects could infer what type of interaction is more suitable for a certain user or for a specific activity. For instance, the smart gym scenario demonstrates the use of a mirror as a display that also receives input from users through a conversation with a virtual avatar. The mirror can display information or ask for user attention. This behaviour is triggered by rules such as: "if it is a new user then the mirror is a virtual avatar, otherwise it is a display". The gym control system activates and manipulates the avatar by continuously analysing the mirror's semantic model and executing the required services.

On the other hand, the smart kitchen scenario is a more proactive smart space, anticipating user needs. The smart objects react when they detect changes in the semantic model of the user (e.g. the proximity within a range). Also, some objects such as the refrigerator and the casserole pot, advertise changes on their semantic models by sending a message on the kitchen network. In this scenario, all the smart objects cooperate to fulfil the user goals.

Other interactions can be explored with semantic queries. For instance, a user has a mobile phone with a special browser for the IoT in the smart garden scenario. When pointing the mobile phone towards a flowerpot, the browser receives a semantic model from the flower as a response. As the user selects options in the graphical interface, the browser determines what information has to display by performing semantic queries in the model.

4 Experimental infrastructure

In order to materialise our proposals we have designed and implemented: a) UbiSOA, a platform for building smart environments using IoT technologies and b) sentient visors, which are systems comprised of user devices and specialized services that together allow users to interact with their environments.

4.1 UbiSOA

Our platform for building smart environments based on the IoT is called UbiSOA (from Ubiquitous Service-Oriented Architecture), and it provides 3 basic mechanisms: discovery of services, for the detection and identification of components at runtime; common messaging for all the participants, to support the interchange of information and cooperation; and event notification, to allow an application to respond to changes in the environment.

We are following a mixed hybrid model for UbiSOA. The framework is mainly based on service orientation as its basic entities are services providing RESTful interfaces, and the use of the service discovery mechanism as a directory to find and consume the needed resources. But, we are also employing the context-driven model to integrate context acquisition capabilities in the form of near-real-time notifications to serve and manage new context data from, and to, service consumers and providers. As in-depth discussions about UbiSOA, and the systems that have been built using this platform, have been provided elsewhere (Aviles-Lopez, 2012), here we give a global overview of the main components (see Figure 4).

A service in our platform consists of a RESTful Web service, which exposes its functionality by describing what common interfaces it implements and notifies its availability within the environment by using the discovery mechanism provided by the platform. We have designed a number of common interfaces for those services, including context acquisition, behaviour analysis, indoor localisation, meta data management, and others.

Once the services are registered in the platform, a developer can create compositions by defining which services are involved and the application flow between them. The composition files are sent to one of the execution engines available in the environment, which are in charge of handling the requests and data involved on the execution of the described functionality. The developer can alternatively use our platform framework to directly interact with the discovery and description mechanisms to handle its own composition implementation.

Some services in our platform have to notify data as it is available, i.e., they follow an event-based communication. To support this, we have included the publish/subscribe model. For instance, if you want to be notified about new tag readings of an RFID service, you must first subscribe to it and specify the parameters of how and when you will be alerted. As the components of our platform are mostly RESTful Web services, a notification of data usually consists of a POST request to a target subscribed URI. The notification can be configured with aggregation operators.

4.2 Sentient visors

During our research work we realised that semantic interactions would be useful to design a generic system intended as a universal platform for interacting in different smart spaces. Hence, we proposed the ”sentient visor” as a browser for the IoT (Garcia-Macias et al., 2011). This browser is meant to reside in mobile phones, tablets and other mobile devices. For the sake of brevity, we have omitted more specific details about the sentient visor in this paper. Nonetheless, technical aspects and a detailed implementation appear elsewhere (Estrada-Martinez et al., 2011).

The concept of the sentient visor relies on having a distributed system for browsing the IoT. This system is comprised of a user device and multiple service providers. For example, we could utilise a smartphone or another device with a camera and displaying capabilities. Then, the user can point with the device towards a particular object and see relevant information superimposed on the screen of the device. This functionality resembles augmented reality (AR) systems. However, the sentient visor does not necessarily need to use AR or a screen for user interaction.

The information can be alternatively provided by auditory means. Besides, user interaction with the sentient visor can trigger the execution of services in particular objects or smart spaces. In this manner, it is not always necessary to have a screen in the user device, as the sentient visor could use ambient displays or activate a specific function in a smart object as a response to users.

The functionality of the sentient visor is possible with the aid of multiple service providers. For instance, we have used our platform UbiSOA to be a provider of contextual data acquisition (e.g., RFID readings, indoor localisation, sensor reports) and other services that can be registered in the platform. Based on the sentient visor concept we have constructed a prototype system called UbiVisor (from Ubiquitous Visor). This way we have used UbiSOA and UbiVisor to create smart spaces where users can interact with the smart objects within in a more intelligent manner (Estrada-Martinez et al., 2011).

5 Experimental tests

We decided to test the concepts of our proposal by recreating some of the scenarios in an experimental setup. Thus, after defining the possible semantic interactions, we considered that the smart garden scenario would be appropriate as it would not only use the core mechanisms of semantic interactions, but would also provide insights about device-mediated interactions with our sentient visor system.

Therefore, in this section we show the semantic interactions in the smart garden scenario. For this purpose, we use the sentient visor with a tablet computer.

5.1 Semantic zoom

Since smart objects can generate variable data over time and provide different amount of services, one of our concerns was how to handle information levels. The semantic zoom (Perlin and Fox, 1993) has been explored before as an interaction metaphor intended to reveal information gradually. This metaphor relies in that users can find more useful to see details on demand, revealing different information by zooming in and out.

We applied the semantic zoom metaphor with the aid of semantic models as follows (see Figure 5). When the user points the tablet towards a flowerpot, the browser identifies the object (e.g. it obtains the URI of the flowerpot by decoding a QR code). With this URI the browser sends an HTTP GET request to the flowerpot and the flower responds with a semantic model (see Figure 6).

The browser determines the semantic zoom level of each particular service by performing semantic queries in the semantic model. While the user selects zooming in or out through the user interface, the browser retrieves the service associated with the semantic zoom level and displays the information provided by the service.

5.2 Semantic service discovery

In the smart garden scenario, the user utilises the browser for visualising the condition of the flowerpots. This requires searching for smart objects and services in the smart space.

The browser has a function to perform a semantic service discovery. This function executes a semantic query in the ontology of the smart space. For instance, the browser can search for services of smart objects in a specified location (see Figure 7). We implemented a RESTful web service that hosts the ontology and performs the queries requested by the browser.

When the user selects searching for services in the user interface, the browser sends an HTTP GET request to the ontology service and it receives a semantic model as a response. This model contains a list of smart objects and their associated services available in the smart space.

Then, the browser analyses the model for displaying the discovered services and the user can select executing particular services of the smart object (see Figure 8).

6 Related work

As we have seen, our work focuses on the possible interactions that can emerge by adding a semantic layer in the IoT. Nevertheless, we have to consider that interactions can occur under different conditions (e.g., in the morning, within a certain time span or while accomplishing an activity). For these situations, context-awareness is a key element to perform tasks on behalf of users. Several works (Ejigu et al., 2007; Wang et al., 2004; Strang and Linnhoff-Popien, 2004; Baldauf et al., 2007) have addressed the use of ontologies for modeling context and reasoning over contextual data in pervasive environments.

In (Greenberg et al., 2011), the authors propose ”proxemic interactions” as a new way of interacting with computing devices. This concept suggests that people expect increasing engagement and intimacy as they approach others, so they could have similar expectations with computers too, allowing computing devices to react according to people's expectations of interaction. This type of interactions could be supported by semantic technologies. For instance, there could be an enhanced semantic zoom that considers the proximity between the user and the smart object. Moreover, semantic information could be used with proximity interactions, as exemplified in the smart kitchen scenario, for enabling proactive behaviour in smart spaces.

An important consideration for creating smart spaces is the connectivity between smart objects. This aspect has been addressed through high-level connections between individual heterogeneous objects using semantic technologies (Van Der Vlist et al., 2010; Niezen et al., 2010). Nevertheless, the focus of those works is on facilitating connectivity whereas our approach is concerned with the interaction problems described in this paper.

On the other hand, there have been proposed semantic devices (Vazquez and Lopez de Ipina, 2007), as a way of enabling a distributed ambient intelligence in smart spaces. These devices are capable of sensing, reasoning and interpreting information at a semantic level. However, semantic information was only used for creating more autonomous smart objects without focusing on user interaction.

Home automation has been commonly utilised to exemplify digital homes enabled by the Web (Kamilaris et al., 2011) and interactions with home devices by means of speech commands (Rouillard and Tarby, 2011). These works share the vision of the IoT paradigm by exploring the creation of smart spaces.

It is also worth noting that we could expect a higher network traffic with the advent of the IoT, due to greater interconnection levels between smart objects and different systems. In this context, we have to be aware of potential malicious attacks to the smart objects and the IoT infrastructure, that could affect the interaction mechanisms. In (Liang Zhou and Han-Chieh Chao, 2011), it has been proposed an architecture for securing the communication, computation and service traffic in the IoT.

7 Conclusions and future work

We have explained how semantic technologies can be applied in the IoT domain for enabling richer interactions. Also, we have developed and utilised our experimental infrastructure, comprised of UbiSOA and sentient visors, for deploying semantic interactions in a smart space.

The experimental setup enabled us to obtain first insights to validate our proposal. As a result, we are considering some aspects to further advance with our approach of semantic interactions.

For example, in the case of smart spaces the semantic interaction mechanisms proposed in this paper are needed components, but they should be complemented with smart systems that provide more sophisticated reasoning capabilities. We are planning to extend our technological infrastructure in order to perform inferences with swarms of smart objects and explore a multi-object interaction.

We also have to consider aspects of the underlying IoT infrastructure such as search engines (Zhang et al., 2011) to enable real-time interaction with smart objects.

From a pragmatic point of view, we should consider that the problems for adopting the semantic web (e.g., complexity for content publishers, scarcity of semantic web services) could be transferred to the IoT. Although, the information cloud of the IoT will comprise not only web documents, but also physical objects data. For this situation, it could be supposed that manufacturers and distributors of smart objects should assume the costs involved in adding a semantic layer to their products, since their goal would be to sell them in the market.

Finally, we should consider that certain types of smart objects would be moving around in a smart space rather than being confined to a static location. This mobility can be caused by people moving an object or by the smart objects themselves (e.g., a portable fan with wheels). In these situations, we have to think in new possibilities of interaction since smart objects could also be seen as some type of mini-robot.

References

Atzori, L., Iera, A. and Morabito, G. (2010) 'The internet of things: a survey', *Computer Networks*, Vol. 54, No. 15, pp. 2787–2805.

Aviles-Lopez, E. and Garcia-Macias, J. A. (2012) 'Mashing up the Internet of Things: a framework for smart environments', *EURASIP Journal of Wireless Communications and Networking*, Vol. 2012, No. 79.

Baldauf, M., Dustdar, S. and Rosenberg, F. (2007) 'A survey on context-aware systems', *Int. J. Ad Hoc and Ubiquitous Computing*, Vol. 2, No. 4, pp.263–277.

Ejigu, D., Scuturici, M. and Brunie, L. (2007) 'An ontology-based approach to context modeling and reasoning in pervasive computing', in *Proceedings of 5th IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOMW 07)*, pp. 14–19.

Estrada-Martinez, P. E., Alvarez-Lozano, J., Garcia-Macias, J. A. and Favela, J. (2011) 'The sentient visor: towards a browser for the internet of things', in *Proceedings of 5th International Symposium on Ubiquitous Computing and Ambient Intelligence (UCAmI 11)*, in press.

Garcia-Macias, J. A., Alvarez-Lozano, J., Estrada-Martinez, P. and Aviles-Lopez, E. (2011) 'Browsing the internet of things with sentient visors', *IEEE Computer*, Vol. 44, No. 5, pp.46–52.

Gershenfeld, N., Krikorian, R. and Cohen, D. (2004) 'The internet of things', *Scientific American*, Vol. 291, No. 4, pp.76–81.

Greenberg, S., Marquardt, N., Ballendat, T., Diaz-Marino, R. and Wang, M. (2011) 'Proxemic interactions: the new ubicomp', *Interactions*, Vol. 18, pp.42–50.

Gruber, T. R. (1993) 'A translational approach to portable ontology specifications', *Knowledge Acquisition*, Vol. 5, No. 2, pp.199–220.

Horrocks, I., Patel, P. F., Boley, H., Tabet, S., Grosof, B. and Dean, M. (2004) 'SWRL: A semantic web rule language combining OWL and RuleML', [online] Available at: http://www.w3.org/Submission/SWRL/ (Accessed: 22 February 2012).

Kamilaris, A., Pitsillides, A. and Trifa, V. (2011) 'The smart home meets the Web of things', *Int. J. Ad Hoc and Ubiquitous Computing*, Vol. 7, No. 3., pp.145-154.

Liang Zhou and Han-Chieh Chao (2011) 'Multimedia traffic security architecture for internet of things', *IEEE Network*, Vol. 25, No. 3, pp.29–34.

Niezen, G., Van Der Vlist, B. J. J., Hun, J. and Feijs, L. M. G. (2010) 'From events to goals: supporting semantic interaction in smart environments', in *Proceedings of IEEE Symposium on Computers and communications (ISCC 10)*, pp.1029–1034.

O'Connor, M. J. and Das, A. K. (2009) 'SQWRL: a query language for OWL', in *6th International Workshop in OWL: Experiences and Directions (OWLED)*, Chantilly, Virginia, USA. Available at: http://bmir.stanford.edu/file_asset/index.php/1723/BMIR-2009-1395.pdf (Accessed 22 February 2012).

Perlin, K. and Fox, D. (1993) 'Pad: an alternative approach to the computer interface', in *Proceedings of 20th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH 93)*, pp.57-64.

Rouillard, J. and Tarby, J. C. (2011) 'How to communicate smartly with your house?', *Int. J. Ad Hoc and Ubiquitous Computing*, Vol. 7, No. 3, pp.155-162.

Strang, T. and Linnhoff-Popien, C. (2004) 'A context modeling survey', in *Workshop on Advanced Context Modelling, Reasoning and Management, UbiComp 2004 - 6th International Conference on Ubiquitous Computing.*

Van Der Vlist, B. J. J., Niezen, G., Hu, J. and Feijs, L. M. G. (2010) 'Semantic connections: exploring and manipulating connections in smart spaces', in *Proceedings of IEEE Symposium on Computers and Communications (ISCC 10)*, pp. 1–4.

Vazquez, J. I. and Lopez de Ipina, D. (2007) 'Principles and experiences on creating semantic devices', in *Proceedings of 2nd Symposium on Ubiquitous Computing and Ambient Intelligence (UCAmI 07).*

Wang, X. H., Zhang, D. Q., Gu, T. and Pung, H. K. (2004) 'Ontology based context modeling and reasoning using OWL', in *Proceedings of 2nd IEEE Annual Conference on Pervasive Computing and Communications Workshops (PERCOMW 04)*, pp. 18–22.

Zhang, D., Yang, L. T. and Huang, H. (2011) 'Searching in internet of things: vision and challenges', in *Proceedings of IEEE 9th International Symposium on Parallel and Distributed Processing with Applications (ISPA 11)*, pp.201–206.

Note

[1] http://www.w3.org/standards/semanticweb/

[2] http://www.w3.org/standards/semanticweb/ontology/

[3] http://www.w3.org/standards/semanticweb/data/

[4] http://www.w3.org/TR/rdf-sparql-query/

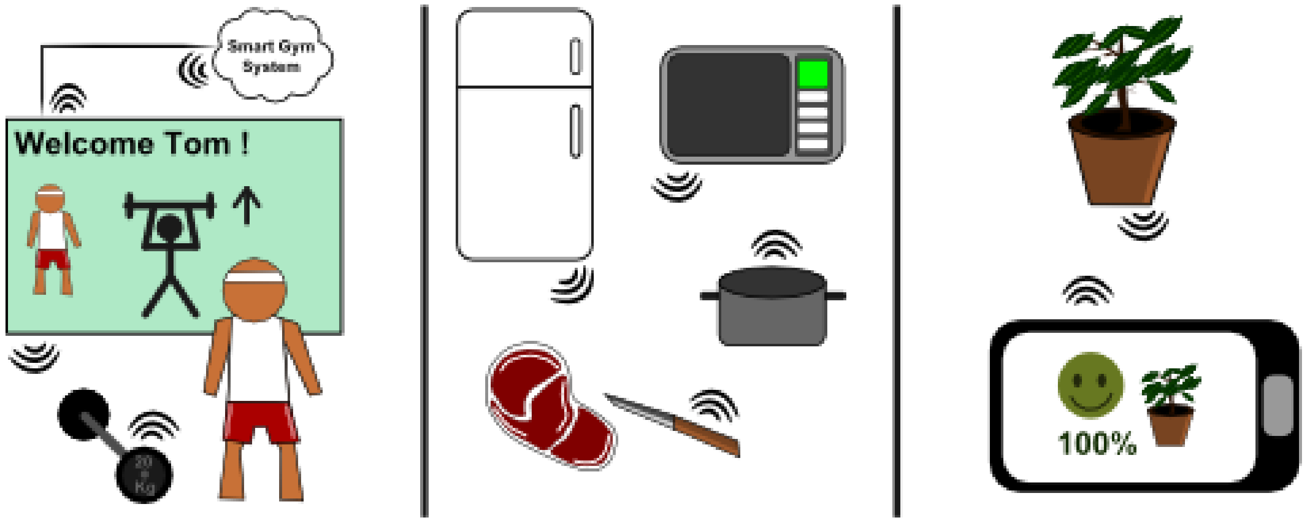


Figure 1 Examples of scenarios for semantic interactions. From left to right. a) smart gym, b) smart kitchen, c) smart garden

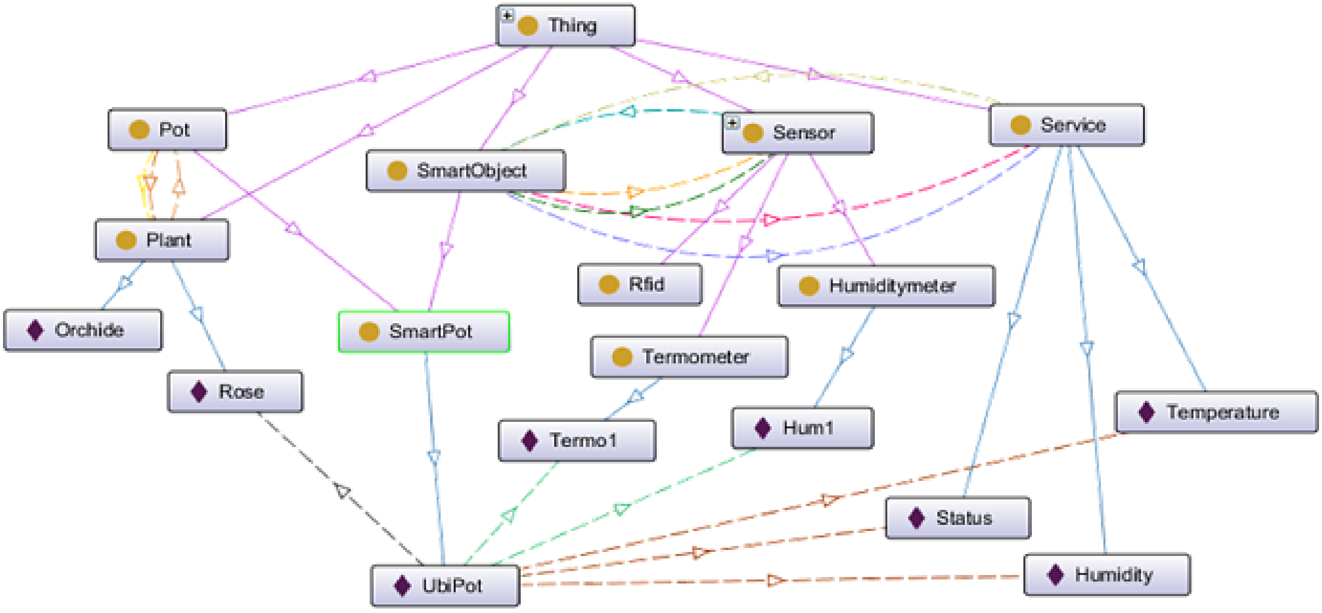


Figure 2 A graph representing the relationships between different concepts in the ontology. In this graph an instance of SmartPot (i.e., UbiPot) is defined as a subtype of SmartObject, conformed by: a Pot, a Plant, some Sensors and Services.

Table 1 Examples of SWRL rules for a smart garden

Rule	*Meaning*
SmartObject(?so) ∧ hasURI (?so, ?uri) → hasService(?so, ?s) ∧ Service (?s)	If it is a smart object and it has a URI, then it has a service
Pot(?p) ∧ hasSensor(?p, ?s) → SmartPot(?p)	If it is a flowerpot and it has a sensor, then it is a smart pot
Plant(?p) ∧ hasTemperature(?p, ?t) ∧ swrlb:greaterThan(?t, 30) → HotPlant (?p)	If it is a plant and its temperature is higher than 30° C, then the plant is hot

Table 2 Examples of semantic queries

Type	*Query*	*Meaning*
SQWRL	Person (?p) ∧ hasContentPresentation (?p, ?cp) ∧ hasType (?cp, ?type) → sqwrl: select (?type)	Select the preferred type of visualisation defined for the person
SQWRL	ThirstyPlant (?p) → sqwrl:select (?p)	Select the thirsty plant (i.e., the plant that needs watering)

SPARQL	PREFIX sv:http://www.semanticweb.org/ontologies/
	2010/10/30/sentientVisor♯
	PREFIX rdf:<http://www.w3.org/1999/02/22-
	rdf-syntax-ns♯ >
	SELECT ?name ?value
	WHERE {
	?x rdf:type sv:Sensor .
	?x sv:hasName ?name .
	?x sv:hasValue ?value .
	}
		Select the name and reading values of all sensors available

```
<?xml version=”1.0” encoding=”UTF-8”
standalone=”no”?>
<rdf:RDF xmlns:
rdf=http://www.w3.org/1999/02/22-rdf-syntax-ns♯
xmlns: sv=”http://www.semanticweb.org/ontologies/
2010/10/30/sentientVisor♯”>
<rdf:Description rdf:
about=”http://localhost:8182/ubicomp/plants/1”>
<sv:humidity>40</sv:humidity>
<sv:temperature>25C</sv:temperature>
<sv:status>good</sv:status>
<sv:name>orchide</sv:name>
</rdf:Description>
</rdf:RDF>
```

Figure 3 RDF Semantic model with information about a smart object. In this example the ontology concepts are linked to sensory data from a flowerpot.

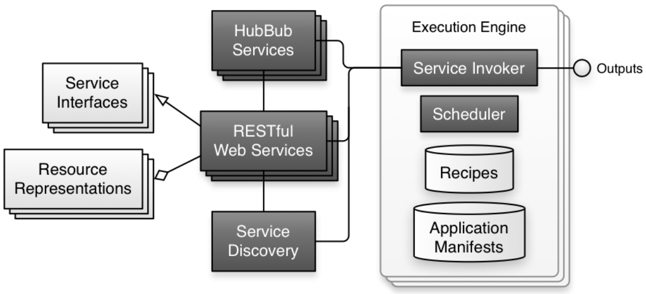


Figure 4 An architectural overview of UbiSOA.

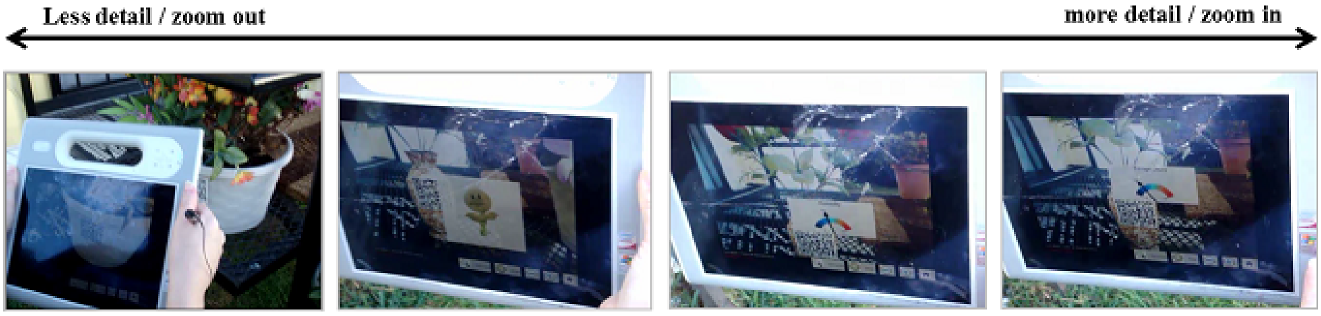


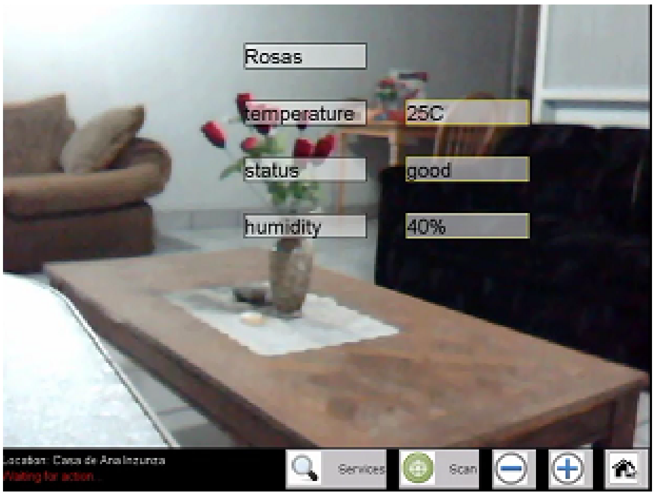
Figure 5 Applying the semantic zoom metaphor in a smart garden. From left to right: a) the user points towards a smart flowerpot, b) the browser displays a picture of a sad face, c) the flowerpot notifies its temperature, d) the flowerpot notifies its level of humidity.

```
<?xml version=”1.0” encoding=”UTF-8”
standalone=”no”?>
<rdf:RDF
xmlns:rdf=”http://www.w3.org/1999/02/22-rdf-
syntax-ns♯” xmlns:
sv=”http://www.semanticweb.org/ontologies/
2010/10/30/sentientVisor♯”>
<rdf:Description rdf:about
=”http://localhost:8182/ubicomp/plants/2”>
<sv:service>
<rdf:Description rdf:about
=”http://localhost:8182/ubicomp/plants/2/status”>
  <sv:zoom>0</sv:zoom>
  <sv:value>sad</sv:value>
  <sv:name>Status</sv:name>
</rdf:Description>
</sv:service>
<sv:service>
<rdf:Description rdf:about
=”http://localhost:8182/ubicomp/plants/2/
temperature”>
  <sv:zoom>1</sv:zoom>
  <sv:value>27C</sv:value>
  <sv:name>Temperature</sv:name>
</rdf:Description>
</sv:service>
<sv:service>
<rdf:Description rdf:about
=”http://localhost:8182/ubicomp/plants/2/
humidity”>
  <sv:zoom>2</sv:zoom>
  <sv:value>40</sv:value>
  <sv:name>Humidity</sv:name>
</rdf:Description>
</sv:service>
</rdf:Description>
</rdf:RDF>
```

Figure 6 RDF semantic model with the services of a flowerpot. The services are organized by their semantic zoom level.

```
PREFIX rdf:
<http://www.w3.org/1999/02/22-rdf-syntax-ns♯ >
PREFIX sv:
<http://www.semanticweb.org/ontologies/2010/
10/30/sentientVisor.owl♯ >
SELECT ?uri
WHERE {
?x  rdf:type         sv:SmartObject .
?x  sv:hasService    ?s .
?s  sv:hasUri        ?uri .
?x  sv:hasLocation   ?y .
?y  sv:hasUri
'http://localhost:8182/ubicomp/locations/1' .
}
```

Figure 7 SPARQL Semantic query for searching smart objects. The query is performed in the ”sv” ontology. Location is referenced by the variable ”?y”.



Rosas

temperature 25C

status good

humidity 40%

Location: Casa de Ana Inzunza

Waiting for action...

Figure 8 Displaying the current condition, temperature and humidity of a smart flowerpot.