Pervasive semantic representation and integration of context-aware homes in context sensitive cities

Aggeliki Vlachostergiou
aggelikivl@image.ntua.gr

Georgios Stratogiannis
stratogian@islab.ntua.gr

Georgios Siolas
gsiolas@islab.ntua.gr

George Caridakis*
gcari@image.ntua.gr

Phivos Mylonas†
fmylonas@image.ntua.gr

Intelligent Systems, Content and Interaction Laboratory
National Technical University of Athens
Iroon Polytexneiou 9, 15780 Zografou, Greece

ABSTRACT

This paper presents an overview on how an ecosystem, consisted of users, appliances and environmental context interacts. The ecosystem is modeled by using Semantic Web technologies from the Internet of Things (IoT) perspective. The IoT is made of users, appliances, sensors and houses. Users are modeled as fuzzy personas and these models are semantically related. Semantic Web technologies enhance the system with adaptability and assist the incorporation of environmental context, user and usage information. Context information consists of temperature, humidity and luminosity and information about infrastructure reconfiguration and user location. This information is collected from various IoT sensors, in the pervasive and urban environment and stored into a repository for rule triggering and system adaptation. Experiments were conducted in order to validate the effectiveness of the proposed system both in the restricted scale of a smart home as well as in a larger scale using input from the SmartSantander Smart City project, collected using the FIWARE framework.

Keywords

Pervasive Computing; Ontologies; SWRLs; Sensors; Smart Homes; Smart Cities; Context Awareness

1. INTRODUCTION

Emerging ubiquitous or pervasive computing technologies offer “anytime, anywhere, anyone” computing by decoupling users from devices [7].To provide adequate complex service for the users, applications and services should be aware of their contexts and automatically adapt to their changing contexts known as context-awareness [7] .Context is very important, since it provides information about the present status of people, places, things and devices in the environment. Context is any information that can characterize the situation of an entity. An entity could be a person, place or object that is considered relevant to the interaction between a user and an application, including location, time, activities and the entity’s preferences [7]. Context-awareness refers to the ability of using context information. Thus, a system is context-aware if it can extract, interpret and use context information to adapt its functionality to the current context of use [3]. In the same view, the term context computing is commonly understood by those working in context-aware area, where it is felt that context is a key in their efforts to disperse and transparently weave computer technology into our lives.

Currently, there is a period of beginning activity on context-aware pervasive computing practices that rely on the use of Internet, where objects are identified as Internet resources and can be accessed and utilized as such (Internet of Things approach). The Internet of Things (IoT) is made of sensors and other components that connect our world made of i.e., humans, our devices, etc., with other objects and appliances. This enables homes and cities respectively to be self-aware and dynamically reconfigurable in real- or near-real-time, based on changes that are continuously monitored and captured by sensors. Data collected by various IoT sensors and processed via appropriate analytics can also help predict the immediate future with reasonable accuracy, which enables better planned responses and actions. Homes and cities can thus become more adaptable to the humans’ needs resulting in the formation of “smart(er) communities” that are socially connected in new ways.Thus, using a complete ecosystem of users, context sensors and smart home appliances that interact following a ubiquitous computing paradigm, would help to adapt and enhance the everyday user-appliance interaction. As a result, for Smart Home environments to further fulfill the users’ needs, they have to contextualize their large-scale data. Contextualization is crucial in transforming senseless data into real information, information that can be used as actionable insights that enable intelligent corporate decision-making.

What context comprises is a widely debated and controversial issue. An extended body of literature exists about the nature of “context in interaction” [8]. In addition, attempts to create a standardized definition of use context have been made [9]. However, some researchers consider the present definitions of context too vague and general to be adapted to any specific design processes. The following objection

*George Caridakis is also affiliated with the Department of Cultural Technology and Communication, University of the Aegean, University Hill, 811 00 Mytilene, Lesvos, Greece, E-mail:gcari@aegean.gr

†Phivos Mylonas is also affiliated with the Department of Informatics, Ionian University, Plateia Tsirigoti 7, 49 100, Corfu, Greece, E-mail: fmylonas@ionio.gr

is common [8]: because context is tightly intertwined with users' internal and continuously changing interpretations, it seems difficult to capture context in any general sense that would support design. Thus, the demand for a new, empirical approach has been noted. For example, Dourish [8] distinguishes between two strands of empirical context-aware computing research: the physical based interaction and the development of interactive systems around understandings of the generally operative everyday social interaction. The majority of empirical research falling under the first category has mainly been concerned with fixed indoor contexts (e.g. offices, meeting rooms, and lecture halls), due to the fact that such settings appear to be static. However, the present study falls under the second line of empirical research, in which we try to understand particular user and home appliances pervasive interactional processes in a sensor driven smart home environment.

Particularly, the novelty of our proposed approach is to provide a common context-aware architecture system in which the user ("eahouker" in SandS) is able to control his household appliances in a collective way via the SNS (Social Network Service) and in an intelligent way via the adaptive social network intelligence. As our system is human-centered the UM (User Modeling) is related to the user's activity inside the ESN (Eahoukers Social Network), while the context aware environment refers to the contextual information that characterize the situation and conditions of the system's entities. The modeling of the contextual information is completed through the capture of the semantics of the relationships between the user and the various entities of the ecosystem (other users, appliances, recipes) to further improve the overall user experience. The semantic description framework of our proposed approach, is based on a number of home rules that are defined for a specific household and eahouker. Since the SandS architecture consists of two layers, high and low respectively, we have on the one hand recipes for common household tasks produced and exchanged in the SandS Social Network that are described in near natural language. Additionally, on the other hand we have every user's context which consists of the actual appliances that the user has in house with their particular characteristics (type, model, brand, etc.). Finally, to ensure with the executability and compatibility of a recipe and to deal as well with any uncertainly and vagueness in modeling the contextual information, a number of some axioms, to enforce constraints to all objects (things in IoT paradigm) of the ecosystem have been introduced in the proposed Web Ontology Language (OWL) that was adopted. To conclude with, the experimental results for the above framework are presented that have been conducted inside the "Social & Smart"(SandS)[1] FP7 European Project which aims to highlight the potential of IoT technologies in a concrete user-centric domestic pervasive environment.

The remainder of this paper is structured as follows: we discuss context in ubiquitous interaction in section 2 and we further introduce context extraction along with context semantic representation in section 3. The experimental validation and the results are analysed in section 4, with related conclusions derived and future work presented within section 5.

[1]http://www.sands-project.eu/, Accessed: 2015-02-19

2. CONTEXT IN UBIQUITOUS INTERACTION

2.1 Users' Context Awareness

As the interaction between the humans and the systems becomes increasingly important for the systems, the user modeling becomes very crucial.User modeling (UM) is the progress through which systems gather information and knowledge about the users and their personal characteristics [11].One of the areas of significant growth for UM is ubiquitous computing.The community's modeling, until recently focused mostly on context: the user's location, physical environment, and social environment. The emphasis was stronger on modeling context than on modeling the user. In the UM community, on the other hand, because of the influence of ubiquitous computing, there has been an increasing concern for the inclusion of context in UMs.

More specifically, the field of pervasive computing has been benefited with the use of ontologies [4].Particularly, using ontologies to include context in UMs assists in overcoming some major issues such as the discovery of the new entities, their current availability, the interoperability between the different entities as well as their adaptability to rapidly changing situations (context-awareness) [14]. W.r.t. these requirements Stavropoulos et al.[17] proposed the BonSAI (Smart Building Ontology for Ambient Intelligence) which benefits from existing ontologies and also adds more classes to model concepts (i.e, services, resources, users, context, actuators, etc.). Other context driven composition approaches suggest the implementation of customized services using components such as building blocks by using context information to deal with the variability of pervasive computing devices and user personalization [13]. In a similar way, an agent-based framework that is more adaptable to context changes has been proposed to better fulfill more complex services [15].

2.2 Pervasive Context

Another important issue that should be pointed out is the interplay between user's preferences and situational context within a pervasive contextual interaction.The view here is that "user models" may have certain main effects, but it is often the contextualized behavioral interaction we are interested in.

So far, most users write rules by hand to interpret sensor data and to control devices. For example, home owners install home automation equipment must write their own rules for when their lights turn on and off. Artificial intelligence (AI) plays a pivotal role in automating this process.AI technologies seek useful information on the contextualized residents' behavior and the state of the home.Computer algorithms have been designed to predict and identify activities performed in the home and to recognize emotions and gestures. These capabilities, as well as the abilities to recognize activities, identify trends, are becoming more available and robust, but are not commonly found in actual homes.

As a result, the goal is to enable devices to interact with their peers and the networking infrastructure without explicit human control. The smart home must also be imbued with an awareness of the resident context (location, preferences, activities), physical context (lighting, temperature) and time context (hour of day, day of week, season, year). Providing this type of context-aware technology makes it possible to design environments that provide, for example customized lighting and temperature settings based on learned users' preferences.

3. CONTEXT EXTRACTION AND SEMANTIC REPRESENTATION

Besides cross-referencing internal data with a plethora of other sources, we need algorithms to extract real human meaning, from the data. To accomplish this, the context-awareness of the changes in the users' context and between the user and the application should be expressed in an intelligent way through a number of executing rules. A recent attempt has been described in [6]. The authors developed the architectural pattern Event-Control-Action (ECA) to collect context information (i.e. facts) and to formulate rules through the use of the ECA-DL expressive domain language. Stavropoulos et al. [16] have also adopted the ECA-DL language to build their systems, with a focus on energy savings. Finally, other works' suggestions range from the use of context-aware rule-based notification services to provide notification depending on the users' context, to the adoption of Multi-Context Systems [2] to collect, process, change and share the available context information hosted by ambient agents.

On the contrary, our approach uses Semantic Technologies to represent the knowledge of the ecosystem. The latter, consists of cities and is converted to its corresponding semantic representation. The cities are comprised of the houses, which are located in them, the appliances and their parts which exist in the houses, as well as the people and the sensors. The sensors are located both in each house and the cities, providing the environmental context of each house and each city respectively.The environmental context potentially could be: (i) the temperature, (ii) the humidity, (iii) the luminosity, (iv) the power consumption, (v) the noise levels and (vi) the human presence. All the data received from the sensors, are stored in a database and are updated every time a new value is received from the sensors.

To define the conditions under which the appliances should be switched on or off, a number of rules are introduced, known as "home rules". Some examples of such rules might be "do not operate the Air-conditioner when the outside temperature is greater than 15°C", or, "do not operate the washing machine when the power consumption of the house in greater than 20kW".

3.1 Ecosystem Modeling

To semantically model the ecosystem,we use the OWL 2 Web Ontology Language (OWL 2)[5], which is an ontology language for the Semantic Web with formally defined meaning. OWL 2 ontologies provide classes (or concepts), relations (or properties), individuals and data values (or literals) which are stored as Semantic Web entities.

Ontology Hierarchy: To represent the ecosystem as an ontology, we choose to model as classes the following parts of the ecosystem: (i) the Users, (ii) the Locations, (iii) the Appliances, (iv) their parts and (v) the Sensors.

Due to the fact that our ontology is hierarchically structured, the descriptive classes such as:"Refrigerator", "Washing Machine" and "Air-conditioner", should be subclasses of the class "Appliance". Figure 1 illustrates an example of the hierarchical structure of the ontology used to semantically model our ecosystem.

Individuals: Each ecosystem is comprised of a number of appliances, sensors, people and locations. All these entities of interest are modeled as individuals. In a more detailed way, every single appliance, such as a sensor and a person is represented in the ontology as an individual of the Appliance, Sensor or Person class respectively. Finally, every unique individual, has a different ID.

Relations: To describe the way in which individuals interact with each other, we use relations. Relations can normally be expressed directly between individuals or between classes. These relations also can be called "object properties", while the relation between an individual, or a class, with a data value is called "data properties".

The object properties of our ontology are mainly used to relate the sensors with a specific location, (i.e., the inhabitants with the house they live, the appliances with the house in which they are, the appliance parts with the appliances they are parts etc.). On the other hand, data properties are similar to object properties with the sole difference that their domains are typed words. In our ontology, they relate the actual sensor values with a sensor, power on or off status of the appliances, and the user properties with the data values.

The relations are also used for the user modeling. Data properties with names such as "age", "gender" and "status" are modeling the information of the users about their age, gender and their social status, respectively. On the other hand, object properties such as "marriedWith", "livesIn" and "belongsTo" provide additional information about the user context and relate the users with other users and the objects of the ecosystem.

The ontology of the ecosystem has been created using the open source Protégé 4.2 platform. Figure 2 illustrates the graph of the ontology produced based on the Protégé platform. The edges between the nodes represent the relations between the ontology concepts and individuals.

3.2 Rules and Consistency Check

In this section we present our novel semantic representation of the ecosystem's home rules. In the Semantic Web, the home rules are expressed using the Semantic Web Rule Language (SWRL) [10]. SWRL has the full power of OWL DL, only at the price of decidability and practical implementations. Despite that, the decidability can be regained by restricting the form of admissible rules, typically by imposing a suitable safety condition. Rules have the form of an implication between an antecedent (body) and a consequent (head). This meaning can be read as: 'whenever the conditions that are specified in the antecedent may hold, the conditions that are specified in the consequent must also hold'. The critical thing of the ontology is that the ontology should always be consistent, after the reasoning. This condition is verified by using of a Pellet reasoner [12]. Every time a home rule is violated, an inconsistency is always detected.Taking this into account and whenever the conditions that are specified in the antecedent's hold, the conditions specified in the consequent must also hold, hence the home rule's violation is transformed to the respective antecedent of the SWRL.

To prohibit the appliances from switching on, the creation of a data restriction is needed. A data property is created with the name "restriction", whose domain is an ap-

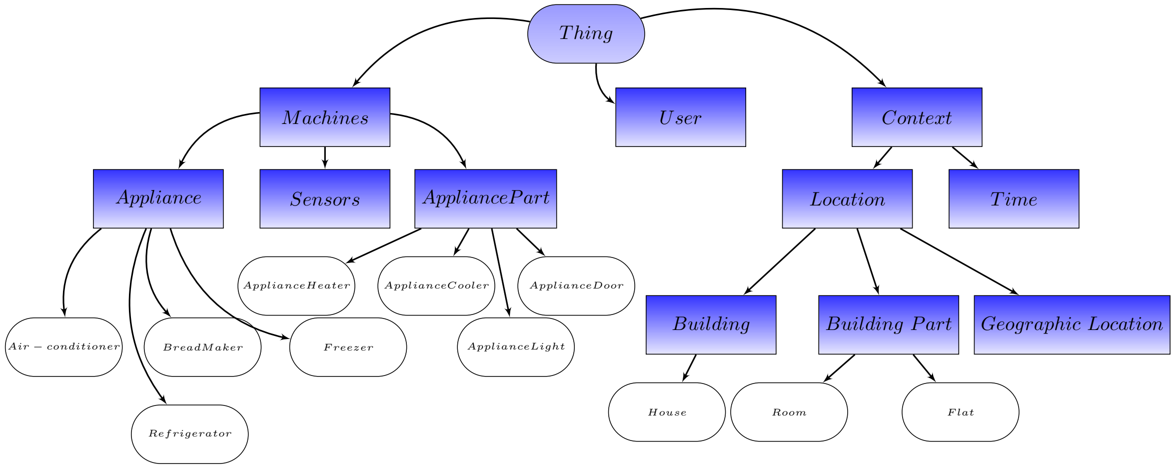


Figure 1: Hierarchical structure of the semantically modeled ecosystem.

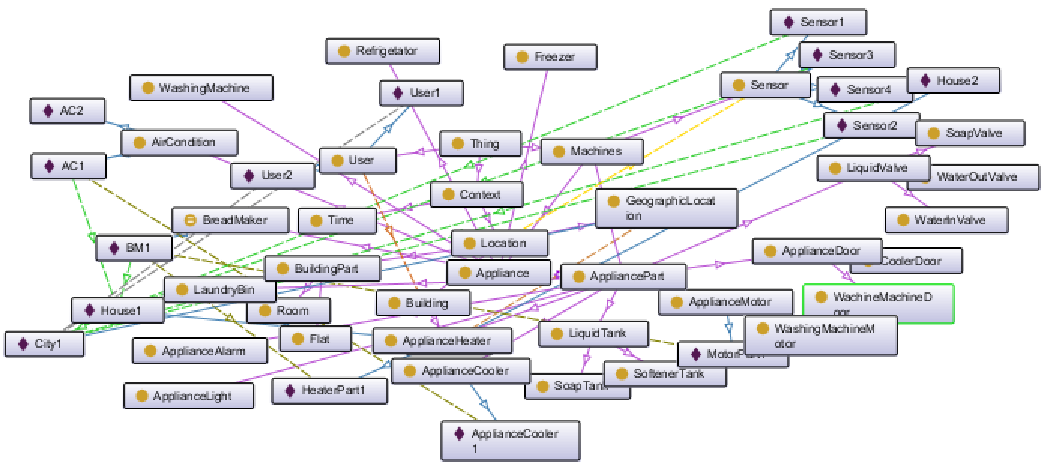


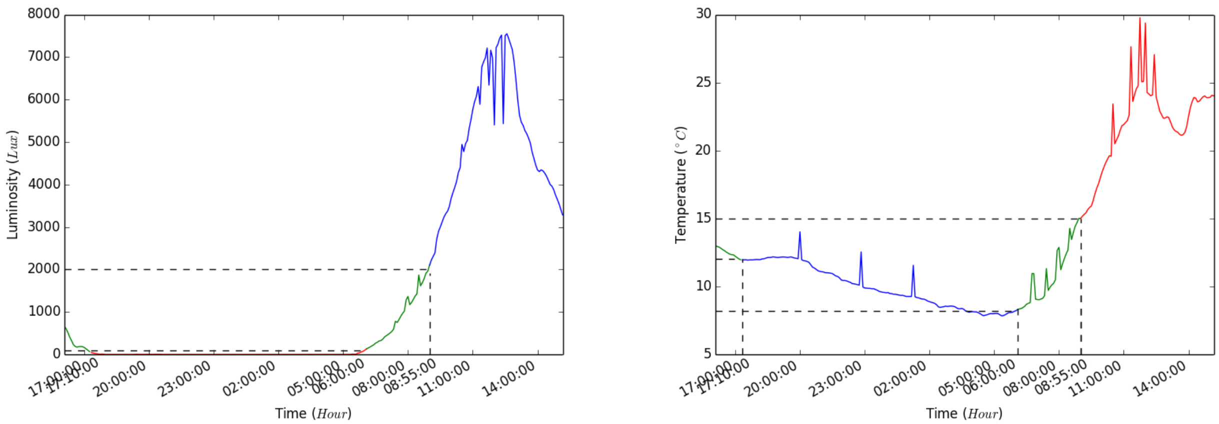
Figure 2: Ontology graph implemented in Protégé platform.

pliance and its value is a Boolean.This restriction is added to the "Appliance" class stating that this data property should never get the true value. Then, every home rule must be created from a text in natural language to its corresponding SWRL. Its form has two parts. If the left side of the rule is satisfied, then after applying the Pellet reasoner, the "restriction" property is created for the appliance with its value being true. Each one of the appliances is switched on until at least one of the home rules is triggered. When a home rule is triggered a restriction is created, leading to an inconsistency. Every time new data is gathered from a sensor the database is queried to check if any of the new sensor values are causing any inconsistency in the ontology. Subsequently, using the Pellet reasoner the system checks for any possible existence of inconsistency. Finally this inconsistency is forcing the appliance to switch off or switch on.

In Table 1 we present some home rules, expressed using the SWRL with the help of the Protégé. The first home rule expresses that the Washing Machine should not be operated when the power consumption is too high and specifically, greater than 1000W per hour. The value of the power consumption is gathered through a sensor in the house and is stored in the database.To check the consistency of this home rule the database must be queried. Similarly, the three following home rules express that the Air-conditioner should not be switched on, when the temperature is higher than 15°C , the luminosity is lower than 100 lux and no human presence is detected in the house. It should be emphasized that the data gathered from the sensors are not stored into the ontology but into the database.To further check any home rule triggering, the database is queried to check whether any of the home rules is triggered. Even though the SWRL does not have negation, that should not be considered as a problem due to the fact that, if any negation is needed, it can be represented to the left part of the rule using a property that could be either true or false (e.g. isOn, notEqual).

4. EXPERIMENTAL VALIDATION

For the large-scale tests of the unified user in the smart



(a) Luminosity value during a twenty four hour period.

(b) Temperature value during a twenty four hour period.

Figure 3: Environmental values, collected by sensors, of in-house and city sensors during a twenty four hour period.

home in the smart city, the SandS approach uses the context sensor data that has been gathered within the European Project SmartSantander[2], which is turning into a living experimental laboratory as part of the EU's Future Internet initiative. Major companies have been involved in the project including Telefonica Digital, the company's R&D wing, along with other smaller suppliers as well as utility and service companies.

4.1 Sensors

In the presented ecosystem, as it has been already mentioned, there are sensors both in every house and for the whole city, which are called 'in-house' and 'city' sensors respectively. The sensors send periodically information about the temperature, the luminosity and other environmental context information. Both the in-house and the city sensors send the values of the sensors periodically to the ecosystem. These values are stored to a specific table of a database overwritting the previous record that was stored. The in-house sensors send information about the environmental context inside the house, such as the noise levels, the house temperature, etc. At the same time, the city sensors gather environmental information about the city, such as the temperature, the humidity, the air pollution, etc. Every time a new sensor value is stored into the database, its timestamp is also stored. Once again, all the sensors send their values periodically to the ecosystem.These values are stored to a specific table of a database overwriting the previous record that was stored. The collection of the city sensor values is accomplished using the FIWARE Ops tools [1]. Adding all this sensors' information in a database, it is every time feasible for the system to identify the exact condition inside and outside the house, by just making a simple query in the database. Then, due to the home rules' structure it is possible in a very short time for the ecosystem to know if any home rule is triggered and if any appliance in a house should be switched on or off.

4.2 Experimental Results

For our experiments we simulated an ecosystem in which the home rules, presented in Table 1 hold. Figure 3 presents the environmental values, collected through the "in-house' and "out-house' sensors during a twenty four hour period. More specifically, we suppose that the human presence sensor detects a human in the house for the whole day. In the same house there exist an Air-conditioner. Furthermore, Figures 3b and 3a present the temperature and the luminosity values during 24 hours, gathered from the city sensor. During the 24 hours the Air-conditioner is switched of whenever any of the home rules, related with the Air-conditioner, is inconsistent. At 17:10 the Air-conditioner of the house is switched off because an inconsistency was appeared since the luminosity of the city was measured lower than 100 lux. At 6:00 the Air-conditioner is switched on as the ontology inconsistency disappears, but at 8:55 the Air-conditioner is switched off once again, due to a different home rule which triggers the ontology inconsistency whenever the city temperature is greater than 15°C.

5. CONCLUSIONS AND FUTURE WORK

It is observed that applications out there can gather and analyze large scale-data, detect human-based meaning from it, and visualize it all, but any application is limiting itself if it is only useful once you open the application and enter a query. Attempting to go beyond this, we decided to formalize and build a complete ecosystem of users, context sensors and smart home appliances that interact based on the ubiquitous computing paradigm to adapt and enhance the everyday user-appliance interaction. Our preliminary experimental results that have been carried in a small scale Smart Home setting and in a larger one using the FIWARE[3] framework, confirm that such contextual computing technology could form a new generation of personalized technology that knows us better than our closest friends. Armed with that knowledge our personal devices can anticipate what we will need next and serve us even better. Although the system has been tested only with six appliances and the SmartSantander sensors it can also work well for large scale

[2] http://www.smartsantander.eu/

[3] http://www.fiware.org

Table 1: Home rules transformed to the respective SWRLs

Rules
House(?house)∧Sensor(?sens)∧hasSensor(?house, ?sens)∧hasPowerConsumption(?sens, ?power)∧WashingMachine(?wm)∧isLocatedIn(?wm, ?house) ∧isOn(?wm, true)∧greaterThan(?power, 1000)→restriction(?wm, true)
City(?city)∧House(?house)∧builtIn(?house, ?city)∧Sensor(?sens)∧hasSensor(?city, ?sens)∧Air-conditioner(?air)∧isLocatedIn(?air, ?house) ∧hasTemperature(?sens, ?temp)∧isOn(?air, true)∧greaterThan(?temp, 15)→restriction(?air, true)
City(?city)∧House(?house)∧builtIn(?house, ?city)∧Sensor(?sens)∧hasSensor(?city, ?sens)∧Air-conditioner(?air)∧isLocatedIn(?air, ?house) ∧hasLuminosity(?sens, ?lum)∧isOn(?air, true)∧lessThan(?lum, 100)→restriction(?air, true)
House(?house)∧Sensor(?sens)∧hasSensor(?house, ?sens)∧Air-conditioner(?air)∧isLocatedIn(?air, ?house) ∧humanPresence(?sens, ?hum)∧isOn(?air, true)∧notEqual(?hum, 1)→restriction(?air, true)

experiments. Any complexity can be solved by splitting the ontology to smaller with less complexity.

Future work consists of the exploration of other rule based paradigms or the combination of the OWL 2 RL ontology with the SPIN rule based reasoner to enhance the adaptability to rapidly changing situations. Additionally, it would be interesting to incorporate in the future the user, usage and the context information through a unified semantic representation, driving an adaptation mechanism aiming to provide a personalised service and optimizing the user experience. Among the aspects of the architecture that will be stressed through experimental validation is the computational cost and the scaling of SandS to a wider user group. Based on the SandS architecture the cloud infrastructure ensures the optimal handling of the computational load since the intermediate processes are not computationally demanding. On the other hand, issues that may arise from the scaling of the platform application are part of the experimental validation since the load is directly correlated with the user activity. The large scale validation at SmartSantander will provide us with useful insights about the latter.

6. REFERENCES

[1] https://www.fi-xifi.eu/fiware-ops/service-offer-management.html. Accessed: 2015-04-17.

[2] A. Bikakis, G. Antoniou, and P. Hasapis. Strategies for contextual reasoning with conflicts in ambient intelligence. *Knowledge and Information Systems*, 27(1):45–84, 2011.

[3] H. E. Byun and K. Cheverst. Utilizing context history to provide dynamic adaptations. *Applied Artificial Intelligence*, 2010.

[4] M. Compton, P. Barnaghi, L. Bermudez, R. García Castro, O. Corcho, S. Cox, J. Graybeal, L. Lefort, M. Hauswirth, C. Henson, et al. The ssn ontology of the semantic sensor network incubator group.

[5] W. W. W. Consortium et al. Owl 2 web ontology language document overview. 2012.

[6] L. Daniele, P. D. Costa, and L. F. Pires. Towards a rule-based approach for context-aware applications. In *Dependable and Adaptable Networks and Services*, pages 33–43. Springer, 2007.

[7] A. K. Dey. Understanding and using context. *Personal and ubiquitous computing*, 5(1):4–7, 2001.

[8] P. Dourish. Seeking a foundation for context-aware computing. *Human–Computer Interaction*, 16(2-4):229–241, 2001.

[9] I. O. for Standardization. *Human-centred Design Processes for Interactive Systems*. International Organization for Standardization, 1999.

[10] I. Horrocks, P. F. Patel-Schneider, H. Boley, S. Tabet, B. Grosof, M. Dean, et al. Swrl: A semantic web rule language combining owl and ruleml. *W3C Member submission*, 21:79, 2004.

[11] A. Jaimes. Data mining for user modeling and personalization in ubiquitous spaces. In *Handbook of ambient intelligence and smart environments*, pages 1015–1038. Springer, 2010.

[12] B. Parsia and E. Sirin. Pellet: An owl dl reasoner. In *Third International Semantic Web Conference-Poster*, volume 18, 2004.

[13] D. Preuveneers and Y. Berbers. Automated context-driven composition of pervasive services to alleviate non-functional concerns. *International Journal of Computing and Information Sciences*, 3(2):19–28, 2005.

[14] A. Ranganathan, R. E. McGrath, R. H. Campbell, and M. D. Mickunas. Use of ontologies in a pervasive computing environment. *The Knowledge Engineering Review*, 18(03):209–220, 2003.

[15] M. J. Santofimia, F. Moya, F. J. Villanueva, D. Villa, and J. C. Lopez. An agent-based approach towards automatic service composition in ambient intelligence. *Artificial Intelligence Review*, 29(3-4):265–276, 2008.

[16] T. G. Stavropoulos, E. Kontopoulos, N. Bassiliades, J. Argyriou, A. Bikakis, D. Vrakas, and I. Vlahavas. Rule-based approaches for energy savings in an ambient intelligence environment. *Pervasive and Mobile Computing*, 19:1–23, 2015.

[17] T. G. Stavropoulos, D. Vrakas, D. Vlachava, and N. Bassiliades. Bonsai: a smart building ontology for ambient intelligence. In *Proceedings of the 2nd international conference on web intelligence, mining and semantics*, page 30. ACM, 2012.