

Web Service Enhanced Home Energy Management System

Ana Rosselló-Busquet and José Soler

*Networks Technology & Service Platforms group, Department of Photonics Engineering,
Technical University of Denmark, 2800 Kgs. Lyngby, Denmark
{aros, joss}@fotonik.dtu.dk*

Abstract—In the last years energy consumption in the home environments has increased considerably. There is an interest to provide users with means to reduce their energy consumption. Introducing a Home Energy Management System (HEMS) into user residences might provide the necessary tools to reduce and optimize the energy consumption in home environments. The main element of HEMS is the home gateway. In this paper, a home gateway suitable for HEMS is presented. The home gateway proposed uses rules to implement the HEMS. The rules can be downloaded through web services from a rule server. Furthermore, web services are used to provide modularity to the home gateway by enabling the deployment of the different logical components into different devices.

Keywords—Home gateway, Home Energy Management Systems (HEMS), ontology, OSGi, web-services

I. INTRODUCTION

Despite the fact that home appliances, such as washing machines and fridges, have become more energy efficient, electricity consumption in users' residences has increased 30% over the last 30 years [1]. This growth in consumption is due to the fact that the number of appliances that can be found in households is also increasing. According to the International Energy Agency (IEA), European electricity consumption is going to increase 1.4% per year up to 2030, unless countermeasures are taken [2]. In order to avoid this, users should become more conscious of their energy consumption and try to reduce it and avoid demand peaks.

The appliances found in users' premises are usually manufactured by different producers and may use different communication technologies which can lead to interoperability issues between devices. Therefore, the main challenge in home networks is the variety of technologies, providing different communication methods, as well as the diversity of producers, providing different types of devices and services.

A Home Energy Management System (HEMS) [3] is a system from which the user can control the devices in the home network through an Graphical User Interface (GUI) and apply energy management strategies to reduce and optimize their consumption. The herein proposed home gateway is technology and device type independent, in order to offer a common "pluggable" platform to different devices in the home network, which makes them interoperable at the service level.

The home gateway is developed using OSGi and offers a GUI from where devices can be controlled and queried. A detailed description of the home gateway developed can

be found in [4]. Furthermore, the home gateway accesses a knowledge base data repository from where the capabilities of the devices can be obtained. This knowledge base data repository is implemented by an ontology, where the home devices are classified according to their functionalities and capabilities. Moreover, the ontology also contains a classification of the different commands and states a home device can support. Energy management strategies can be performed by applying a set of rules, which are based on the energy consumption of the home devices, information from the electrical grid and users' preferences. More information about the knowledge base data repository and the energy management method used can be found in [5].

The home gateway presented in [4] and [5] has been extended to include a web services interface allowing modularization of the system. In this paper, the web services incorporated into the HEMS and the required changes to the previous developed home gateway will be described. The following sections will describe in detail the developed home gateway explaining each logical component and the interaction between them. The main motivation was to create a simple home gateway which would be easily scalable and that had the necessary capabilities to create a home energy management system: offer control of devices through a GUI, run rules to apply energy management strategies and interoperability between devices even if they belong to different subnetworks. Furthermore, as our home gateway is developed using OSGi, new bundles offering other OSGi services or web services, such as communication with the electricity providers, can be easily incorporated.

The rest of this article is organized as follows: Section II introduces relevant related works. Section III briefly describes the OSGi framework. Next section, Section IV, provides an overview of the web services, while Section V describes the home gateway architecture. Section VI describes the interaction between bundles and web services, within the chosen OSGi architecture. Finally, Section VIII provides final remarks and conclusions.

II. RELATED WORK

The implemented home gateway is based on Open Service Gateway Initiative (OSGi) [6] using Equinox which is an Eclipse project that provides a certified implementation of the OSGi. To create the knowledge base data repository for the home gateway implemented, OWL ontology [7] is used.

Protégé-OWL API 3.4.4 [8] provides the necessary mechanisms to manipulated ontologies from a JAVA application.

An example of a home gateway using OSGi and ontologies is Domotic OSGi Gateway (DOG) [9] by Politecnico di Torino. DogOnt [10] ontology is used in the home gateway developed in [9] and is reused as well in the implementation herein described. This ontology provides a good classification of the devices that can be found in a home environment. However, this article will not describe this ontology as an extensive description can be found in [10]. The main difference between DOG and the home gateway herein presented is the fact that DOG is focused on domotics while the home gateway herein is mainly concerned with energy management. The user can use it to define their own energy management system by creating, modifying and deleting rules, which may reduce the total electricity consumption.

In [11], another example of a home gateway developed using OSGi can be found. As with DOG, no energy management strategies are applied, since the necessary tools, for example a rule engine, are not provided.

Furthermore, the authors in [12] describe the IntelliDomo's learning model which is an ontology-based system, able to control a home automation system and to learn users periodic patterns, when using home appliances.

In [13] a HEMS has been implemented to reduce stand-by consumption by setting a power line network. Similarly, in [14], a HEMS implementation using ZigBee and infrared communication to reduce stand-by consumption of power outlets and lights is presented. THE Hems proposed in this paper has two main advantages over [13] and [14]: 1- the energy management strategy can help reduce the consumption of all users' appliances and not only stand-by consumption; 2- the HEMS may communicate using different technologies and not only power line communication or ZigBee.

A HEMS is implemented in [15] to monitor the consumption of appliances. The implemented system does not offer any tools o apply energy management strategies unlike the HEMS presented in this paper.

In this article the home gateway presented in [4] and [5] has been extended to include web services. This web services are used to enable the deployment of the different logical components of the home gateway into separate devices. Furthermore, web services are also used to download rules for the energy management system included in the home gateway. This supposes an advantages to the users, the rules can be downloaded from a rule server instead of being introduced to the system by writing them.

III. OSGi

The OSGi Framework [6] is an open service platform for the delivery and control of different JAVA-based applications, called bundles. Bundles are JAR files containing extra metadata, making them fully self-describing. This metadata

is contained in MANIFEST.MF files. Besides the MANIFEST.MF, the bundles consist of packages containing JAVA classes. The packages of a bundle are private by default and they are not visible to other bundles. However, packages can be made public and may be imported by other bundles if the package is exported. MANIFEST.MF is used to declare which packages are imported and/or exported. Furthermore, bundles can export and import services by using the OSGi Service Registry. A service is defined using a public JAVA interface which must reside in an exported package, usually contained in the bundle exporting the service. This bundle implements the service, instantiates it and then registers the instance using the OSGi Service Registry under the service interface name and using the exported JAVA interface. Other bundles can import this service by using the JAVA interface and use its methods.

IV. WEB SERVICES

The W3C defines a web service as "a method of communication between two electronic devices over a network. A web service is a software system designed to support interoperable machine-to-machine interaction over a network. It has an interface described in a machine-processable format (specifically WSDL)." [16]

Web services are incorporated into the home gateway developed to offer modularity. The first implementation of the home gateway considered that all the bundles developed would be deployed in the same device, the home gateway. However, this may not be the case and some of the bundles may be deployed in a separated device from the home gateway.

As mentioned, the home gateway developed offers energy management strategies by using rules. Web services are used to download these rules from a rule server. In this case only one Rule Server bundle has been implemented. However, the user could use more than one rule server to obtain the rules for the energy management system from different sources.

To incorporate web services into the home gateway developed, Apache CXF Distributed OSGi [17] is used. This distribution enables an easy integration of web services into OSGi platform. Furthermore, CXF-DOSGi will auto-generate the Web Services Description Language (WSDL) from the java interface, at the deployment time. The web services are implemented in a so called server. The consumer of the service is called the client. In order for the client to use the web service, a java interface of services has to be deployed in the client side. The server bundle will contain an implementatin of this interface. The interfaces of the web services are deployed in Web Services Interface bundle which will be enclosed in the server and client sides.

V. HOME GATEWAY ARCHITECTURE

The home gateway designed is aimed for a HEMS. It provides a knowledge base data repository where information about the devices is stored and also a rule engine. The rule

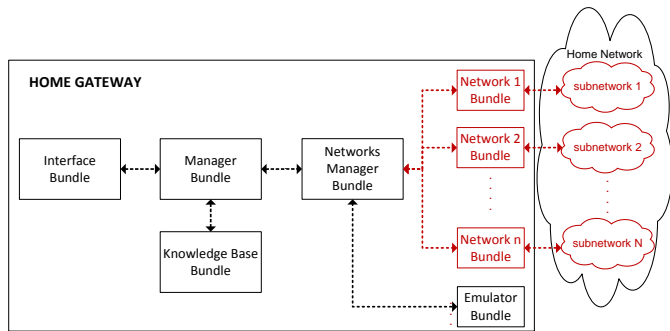


Figure 1. Home Gateway Architecture

engine is used to run rules that will help the user reduce the energy consumption at home. Furthermore, the home gateway provides a user interface from where the devices and rules can be managed.

An ontology is used as a knowledge base data repository. DogOnt ontology is used to create the initial knowledge base data repository of the home gateway. However, this article will not describe this ontology as an extensive description can be found in [10].

The first design of the home gateway implemented consists of the bundles shown in Fig. 1. This first design assumes that all the bundles will be found in the same device. However, some of the bundles could be found in different devices. The interaction between bundles found in different devices can be handled by using web services. In the second implementation, which is presented in this paper, it is assumed that there are two GUI: the Administrator GUI and the User GUI, as shown in Fig. 2. The Administrator GUI is contained within the home gateway. On the other hand, the User GUI could be contained within a separate device, for instance the users' computer. Furthermore, the Network n Bundles, which are equivalent to Network Emulator bundle, could be deployed separately from the home gateway implementation. This is a likely situation as home appliances use different communication technologies and bridges could be used, together with the home gateway, to enable communication to all the devices in the home network. The presented HEMS offers interoperability at the service level between its components, which interact through web services. In order to integrate web services into the new design, three new bundles, called Home Gateway Web Services Interface bundle, Rule Server Implementation bundle and Rule Server Interface bundle, are added into the implementation. A short overview of each bundle is provided in the following subsections.

A. Knowledge Base Bundle

This bundle handles the interactions with the knowledge base data repository and rule engine. To implement the home gateway knowledge base data repository DogOnt ontology is used. Instances of OWLIndividuals are used to represent these home devices, their functionalities and their status based on DOGont OWLClasses. The information about the devices, including their status, features and functionalities,

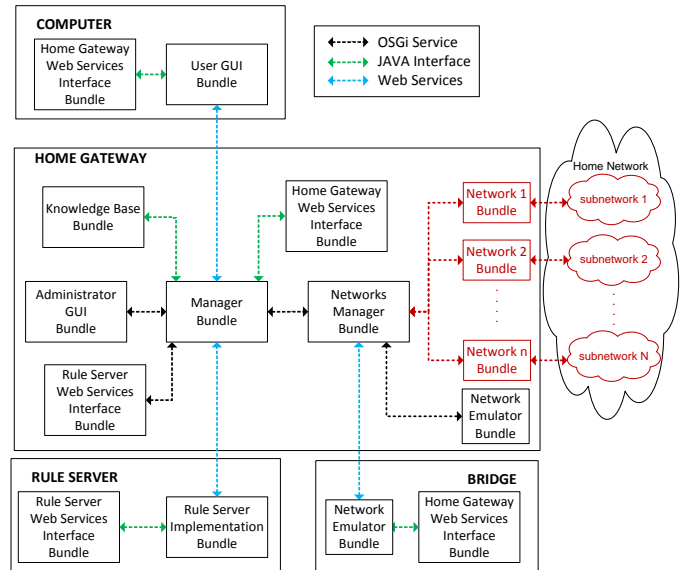


Figure 2. HEMS Architecture

can be obtained from the knowledge base data repository by using queries. SQWRL (Semantic Query-Enhanced Web Rule Language) [18] is used to write the queries. SQWRL is based on SWRL language for querying OWL ontologies and provides SQL-like operations to retrieve knowledge from them. Furthermore, SWRL (Semantic Web Rule Language) [19] is based on a combination of the OWL-DL and OWL-Lite, which are sublanguages of the OWL.

In addition this bundle contains the means to apply energy management strategies by using rules. Rules are written in SWRL, which are used to reason about OWLIndividuals by using OWLClasses and properties terms. Jess Rule Engine [20] is used to run the rules and queries. Protégé-OWL API 3.4.4 [8] is used to manipulate the ontology through the implemented home gateway. This API also provides methods to handle the interaction between Jess Rule Engine, SQWRL and SWRL. Further information details about the Knowledge Base Bundle can be found in [5].

B. Administrator GUI Bundle

The home gateway developed provides a Graphical User Interface (GUI) which is contained in this bundle. This GUI is used to communicate to the devices, obtain information about them and manage the energy management rules. This GUI is aimed at users with knowledge about the ontology and the capabilities of devices. The user can use this GUI to send commands to the devices in the home network, get their status, get the valid commands for a device, incorporate a new device to the knowledge base data repository, manage rules to apply energy savings and save the knowledge base data repository containing the information about the devices in the network. The Administrator GUI provides the list of devices in the home network and also the list of possible commands. Not all commands can be sent to all devices, so when the user tries to send an invalid command to a device an error will be prompted.

C. User GUI Bundle

This bundle is contained within a different device of home gateway, for example the users' computer. This GUI is less advanced than Administrator GUI. The User GUI is a simple GUI with limited capabilities which is aimed at users with limited knowledge about the devices and energy management strategies. This GUI is used to send commands to the devices and get their status. The energy management rules cannot be managed from this GUI. However, this functionality could be incorporated in this GUI if desired.

This bundle obtains the necessary information about the devices by using web services to communicate to the Manager bundle. For instance, the list of devices in the home network and also the list of possible commands can be queried. Web services are also used to enable this separated user GUI to send commands and receive notifications, for instance status updates.

D. Network n Bundle

The home network found in the users' premises can contain devices using different communication technologies, for example power line or wireless. Each of these Network n bundles will handle the communication with the devices in the subnetwork n in the home network. These bundles will send messages and forward notification messages to/from the devices contained in the n subnetwork.

The home gateway could provide some of these communication technologies. However, the home gateway may not provide all the communication technologies found in the home network and the possibility of using bridges to communicate to some of the home appliances is a likely scenario. For this reason these bundles could be found in the home gateway or in a different device such a bridge as shown in Fig. 2.

E. Network Emulator Bundle

The focus of this paper is not on the enabling technologies in the physical layer and their interoperation, but on the software mechanisms that allow use of the different elements, regardless of the connectivity mechanisms towards the home gateway. The home network is therefore emulated and an GUI is provided to emulate changes in the devices status. This bundle functionalities and interaction with the rest of the architecture are equivalent to Network n bundle. Two Network Emulator bundles are used, one contained in the home gateway and another contained in a bridge separated from the home gateway device. The Network Emulator bundle contained in the bridge interacts with the Manager bundle by using web services. On the other hand, the Network Emulator bundle contained within the home gateway uses the OSGi services directly provided by the Manager bundle. Both of this Network Emulators provide a user interface from where the devices status can be changed, for instance, the status of a switch or sensor.

F. Networks Manager Bundle

Due to the fact that various Network n Bundles may exist, inside the home gateway and also in bridges, this bundle is created to handle the communication with these Network n bundles. This bundle will receive all the messages that have to be send to the home network devices and forward it to the corresponding Network n bundle of the subnetwork hosting the target. In order for this bundle to function properly there has to be a mapping between the the Network n bundle and the information contained in the knowledge base data repository so the message can be forwarded to the proper Network n bundle. At the time of writing this bundle only interacts with Network Emulator bundles. However, a Network bundle to support communication with KNX networks is under construction.

G. Manager Bundle

This is the central bundle and handles the interaction between the different bundles and contains the web services implementation and therefore acts as the server. To handle this interaction, this bundle offers different services, one for each functionality. The services used in the home gateway will be explained in more detail in section VI. Some of this services are exported as web services and others are simple OSGi services.

H. Home Gateway Web Services Interface Bundle

In order to implement the web services for the home gateway this bundle is needed. It provides the JAVA interfaces needed by the client and the server of the web service to make the exchange of information possible. This bundle is used to define the methods that can be used through the provided web services. This bundle must be included in both components, the client and the server, as the interfaces are need in both sides to successfully implement the web services.

This bundle contains different packages, each containing the necessary interfaces for each web service provided by the home gateway which will be explained in section VI.

I. Rules Server Implementation Bundle

This bundle is used as a rules provider for the energy management system of the home gateway implemented. This bundle emulates a rules server that can be found in the internet. This server mainly contains a data base where SWRL rules are stored and offers access to them through web services. The user is able to connect to it using the GUI and downloading the rules (s)he selects.

J. Rules Server Web Services Interface Bundle

Similar to the Home Gateway Web Services Interface Bundle bundle, this bundle provides the JAVA interface needed by the client and the server of the web service to make the exchange of information possible. However in this case the exchange of information is between the Manager bundle and the Rules bundle An overview of how the rules are imported to the home gateway is given in section VII.

Table I
HOME GATEWAY SERVICES

Service	Provided by bundle	Used by bundles
<i>Message Listener</i>	Manager	Networks Manager
<i>Message Sender i</i>	Network i	Networks Manager
<i>GUI Printer Listener</i>	Manager	GUI
<i>Rules Provider</i>	Rules Server	Manager
<i>Knowledge Base Operator</i>	Knowledge Base	Manager
<i>Manager Functions</i>	Manager	GUI
		Network Manager
		Network i

VI. BUNDLES INTERACTION

In this section the services offered by the different bundles will be explained briefly and are summarized in Table I. A more detailed description of the services can be found in [4]. Not all the services are made available through web services as some are used only inside the home gateway. The details of the web services is given in the next section.

The following services can be found in the developed home gateway:

- **Message Listener Service:** This service is provided by the Manager bundle. It is used by the Networks Manager to receive all the messages that have to be sent to the devices in the home network. After receiving a message the Networks Manager will find which subnetwork is hosting the target and forward the message only to the corresponding Network bundle. The messages send to the devices can be generated by the user through the provided GUIs or by Knowledge Base bundle when a rule has been fired.
- **Message Sender i Service:** This service is used by the Network Manager bundle to forward the message to the corresponding Network bundle hosting the targeted device. Each Network bundle provides its own Message Sender service which the Network Manager has to import. One may argue that the Networks Manager bundle could be removed from the implementation and all Network bundles could subscribe to Message Listener and Message Sender service could be omitted. However, this would mean that all Network bundles would receive the message even though the targeted device is not found in its subnetwork. Furthermore, considering that some of the Network bundles may be found outside the home gateway device, Networks Manager bundle, Message Listener Service and Message Sender services are necessary. Due to this fact this service is offered as a web service so bridge devices can use it.
- **GUI Printer Listener Service:** This service is imported by all the user GUIs, in this case the Administrator GUI bundle and the User GUI bundle, which imports it through web services. This service is used to receive notification messages, for instance when a device changes status a message will be received by the subscribed bundles and printed in the GUI to notify the user.

- **Rules Server Service:** This service provides access to the rules stored in the Rule Server bundle. It is implemented as a web service where the Rule Server bundle is the server of the web service and the Manager bundle is the client. This service will provide methods to download the list of rules, the rules' names and brief description. It is also used to download the SWRL rule to be deployed in the home gateway energy management system. An example of how the user can download rules from a rule server is provided in section VII.

- **Knowledge Base Operator Service:** This service is used to access the knowledge base data repository, containing the information about the home network devices and rules. It is used to query knowledge base data repository and obtain information about the devices and energy management rules. The only bundle importing this service is Manager bundle. The Manager bundle will offer the functions provided by this service to the rest of the bundles. In this way there is a separation between the knowledge data base repository and the rest of the architecture in case it is modified.

- **Manager Functions Service:** This service offers different functionalities. Firstly, this service can be seen as a mid-step to query the knowledge base data repository. This is done to facilitate the modularity of the system. The Knowledge Base Operator service is used to directly access and query the knowledge base data repository. The Manager Functions service is used by bundles that need to obtain information contained in the knowledge base data repository. The bundles inside and outside the home gateway used this service to obtain information about the home network and its devices. For example, the GUIs bundles use this service to obtain the list of the home network devices and commands for those devices. The Network Manager bundle uses this service network information of the devices, for instance the subnetwork hosting the device. The Manager Functions service will use the Knowledge Base Operator service to obtain the information from the knowledge base data repository.

In addition, the Manager Functions service offers two other functionalities related to the Message Listener and the GUI Printer Listener. This service offers a method to forward the messages for a home device to Message Listener subscribers, in this case only the Network Manager bundle. In a similar way, when a message has to be delivered to the GUIs, the Manager Functions service is used to send the notification to all the bundles subscribed to GUI Printer Listener.

The last functionality of this service is to provide access to the rules servers by using Rules Provider service. Manager Functions service includes the necessary methods to bridge the GUI and the Rules Provider

service.

As the Manager Functions service is used by the GUI bundles and Network bundles it has been offered through web services.

VII. ACCESSING RULES SERVER

The Administrator GUI bundle offers a GUI from where the user can manage the energy management system rules. These rules are designed to decrease the energy consumption by, for example, turning off appliances when they are not need. For instance, the home gateway can have a rule which turns off the light in a room when there is enough natural light coming through the window or when there is no one in the room. These rules can be introduced in the HEMS through the Administrator GUI, by introducing them manually or downloading them from a Rules Server using web services. The rules in a Rule Server are presented to the user through a GUI, which displays the names and a brief description of each rule in the Rule Server. This information is obtained by the Manager Functions using the Rules Provider Service accessed through web services. If the user is interested in incorporating any rules into his/her energy management system, the rule is downloaded and loaded into the rule engine in the Knowledge Base Bundle. The rule is loaded by using Knowledge Base Operator service through Manager Functions service.

Using rules servers and web services the user is not forced to write his/her own rules, (s)he can obtain them from a rule server or even from other users.

VIII. CONCLUSION

In this paper, a home gateway using OSGi framework, ontologies for the knowledge base data repository and web services has been presented. Protégé-OWL API 3.4.4 has been used to handle the interaction between the OSGi home gateway, SQWRL queries, SWRL rules and Jess Rule Engine.

The home gateway presented can support different technologies as long as the corresponding Network bundle is created. These Network bundles can be found outside the home gateway device. For this reason web services have been incorporated into the home gateway. Through web services the remote Network bundle can easily consume the home gateway services. In a similar way, the GUI could be also found outside the home gateway, for instance a computer or mobile phone. By offering the home gateway services through web services freedom to choose between different display devices is provided. Furthermore, web services are also used to download rules for the energy management system provided in the home gateway from a rules server.

The main motivation was to create a simple home gateway which would be easily scalable, new bundles can be incorporated to the HEMS to offer new functionalities and the bridges to communicate with all home devices can be

deployed when necessary. The system developed has the necessary capabilities to create a home energy management system: offer control of devices through a GUI, run rules to apply energy management strategies and interoperability between devices even if they belong to different subnetworks. The home gateway presented in this paper is therefore suitable for Home Energy Management System (HEMS).

REFERENCES

- [1] B. Consortium, "D2.1: Service requirement specification," 2009.
- [2] E. Commission, *European Technology Platform SmartGrids - Vision and Strategy for Europe's Electricity Networks of the Future*. Office for Official Publications of the European Communities, 2006.
- [3] A. Rosselló-Busquet, G. Kardaras, J. Soler, and L. Dittmann, "Towards Efficient Energy Management: Defining HEMS, AMI and Smart Grid objectives," in *The Tenth International Conference on Networks*, Jan. 2011.
- [4] A. Rosselló-Busquet, J. Soler, and L. Dittmann, "A Novel Home Energy Management System Architecture," in *13th International Conference on Computer Modelling and Simulation (UKSim)*, 2011.
- [5] A. Rosselló-Busquet, L. J. Brewka, J. Soler, and L. Dittmann, "OWL Ontologies and SWRL Rules Applied to Energy Management," in *13th International Conference on Computer Modelling and Simulation (UKSim)*, April 2011.
- [6] OSGi Alliance, "OSGi Service Platform Core Specification Release 4," Accessed Dec. 2010. [Online]. Available: <http://www.osgi.org>
- [7] W3C OWL Working Group, "OWL 2 Web Ontology Language Document Overview," Accessed Dec. 2010. [Online]. Available: <http://www.w3.org/TR/owl2-overview/>
- [8] H. Knublauch, "Protege-OWL API Programmer's Guide," Accessed Dec. 2010. [Online]. Available: http://protegewiki.stanford.edu/wiki/ProtegeOWL_API_Programmers_Guide
- [9] D. Bonino, E. Castellina, and F. Corno, "The DOG gateway: enabling ontology-based intelligent domestic environments," *Consumer Electronics, IEEE Transactions on*, november 2008.
- [10] D. Bonino and F. Corno, "DogOnt - Ontology Modeling for Intelligent Domestic Environments," *The Semantic Web - ISWC 2008*, 2008.
- [11] S. Ok and H. Park, "Implementation of initial provisioning function for home gateway based on open service gateway initiative platform," in *Advanced Communication Technology, 2006. ICAC 2006. The 8th International Conference*, feb. 2006.
- [12] P. Valiente-Rocha and A. Lozano-Tello, "Ontology and swrl-based learning model for home automation controlling," in *Ambient Intelligence and Future Trends-International Symposium on Ambient Intelligence (ISAml)*, ser. Advances in Soft Computing. Springer Berlin / Heidelberg, 2010.
- [13] S. H. Ju, Y. H. Lim, M. S. Choi, J.-M. Baek, and S.-Y. Lee, "An efficient home energy management system based on automatic meter reading," in *Power Line Communications and Its Applications (ISPLC), 2011 IEEE International Symposium on*, april 2011.
- [14] J. Han, C.-S. Choi, and I. Lee, "More efficient home energy management system based on zigbee communication and infrared remote controls," *Consumer Electronics, IEEE Transactions on*, 2011.
- [15] C.-S. Choi, W.-K. Park, J.-S. Han, and I.-W. Lee, "The architecture and implementation of proactive green home energy management system," in *Information and Communication Technology Convergence (ICTC), 2010 International Conference on*, nov. 2010.
- [16] "Web Services Glossary W3C Working Group," Accessed April 2011. [Online]. Available: <http://www.w3.org/TR/ws-gloss/>
- [17] "Apache CXF Distributed OSGi," Accessed April 2011. [Online]. Available: <http://cxf.apache.org/distributed-osgi.html>
- [18] M. J. O'Connor, "SQWRL," Accessed Dec. 2010. [Online]. Available: <http://protege.cim3.net/cgi-bin/wiki.pl?SQWRL>
- [19] I. Horrocks, P. F. Patel-Schneider, H. Boley, S. Tabet, B. Groszof, and M. Dean, "Swrl: A semantic web rule language combining owl and ruleml," Accessed Dec. 2010. [Online]. Available: <http://www.daml.org/2004/04/swrl/>
- [20] Sandia National Laboratories, "Jess Rule Engine," Accessed Dec. 2010. [Online]. Available: <http://www.jessrules.com/>