A Remote Monitoring and Control of Home Appliances on Ubiquitous Smart Homes

Seong Joon Lee, Yong Hwan Kim, Sung Soo Kim, Kwang Seon Ahn
Department of Computer Engineering, Kyungpook National University
1370 Sankyuk-dong, Buk-gu Daegu, Korea
imggaibi@hotmail.com, {hypnus, ninny, gsahn}@knu.ac.kr

ABSTRACT

The goal in this paper is to design and implementation the service agent society mechanism of home network community by using Jini platform. On Ubiquitous smart homes environment, this mechanism can make various services to be used among service agents. Based on the mobility of service agent on the Jini platform and instant messaging system (IMS), the mobile device can receive the real time notify from its federation leader in order to download the necessary service, to remove the unnecessary service, or to update the newly service without reconfiguring the system. Moreover, using the Natural Language Processing, the client is possible to express smoothly the order. Therefore, our proposed agent-based management system for ubiquitous smart homes provides a user-friendly interface, and can make use of add-on functionalities of the IMS such as voice chatting, SMS, and multimedia.

Categories and Subject Descriptors

H.5.2 [**Information Interfaces and Presentation**]: User Interfaces – *User-centered design.*

General Terms

Design.

Keywords

Ubiquitous computing, interaction techniques, user participation, universal remote control, Information appliances.

1. INTRODUCTION

Smart Homes (SH) has been considered as one of the main challenges of Ubiquitous computing environment [23, 10]. SH is living space combining different technologies to provide services to residents and a variety of devices to consist of home net-work. In SH, these devices, known as networked appliances (NA) or information appliances (IA), ought to have the computational power to perform predefined functions and the networking ability

to share information with other devices. Therefore, if end-users can have the remote controller meaning any device that is installed the UI which provide the interaction between different providers and various versions, he or she is able to more efficiently control a device. The remote control means both to monitor the appliances and to change their status on the network. However, because the respective devices have their heterogeneous characteristics and limitations, not all devices can be possessed of the same user interface (UI) [19]. It is also difficult to keep the consistency between various versions because developers add novel faculties and drop unnecessary faculties at the device. For this reason, the more various digital devices will be popularized, the more integrated efficient universal user interfaces (UUI) will be necessary.

Digital Information for monitoring and controlling IAs is becoming increasingly important to households and consumer electronics. Recently, this display and management is that something computers can do very well. However, because homeowners have to go to the computer to get anything that they wish, it is not easy to notify residents of status information of home appliances to homeowner, which occur in real-time. So, one of the key issues in the implementation of SHs has been how to efficiently control home appliances through the same method both inside and outside the home [17]. The remote control means both to monitor the appliances and to change their status on the network. To building the SH environment which is accessible to the device which a resident wish, anytime, anywhere, and any device, house-holds and consumer electronics have to provide UUI that represent the interconnectivity and interoperability between different platforms and environments. For solving this problem, various UIs have been continuously developed for SHs.

Advances in Internet technologies have prompted the development of various Internet-based SH remote controls, most of which adopt a web interface between people and machines. Although the use of this web-based user interface (UI) has many ad-vantages such as scalability, visibility and a uniform UI, it has the following problems. First, it cannot provide push functionality for supplying the homeowner with status information, because the HTTP protocol is defined for one-way continuous transmission. Push technology enables authorized users who are subscribers to a server to automatically receive up-to-date information via the home server. In a web-based SH, each client may first download all state information on the smart devices to be controlled, which consists of CGI scripts, Java applets, and HTML documents. This downloaded information may include information for a variety of devices, even though the user may be interested in only one particular device. These methods are slow and tedious because when the IA has many functions users have to recursively choose

between many items displayed on the screen, which is difficult for beginners and individuals not accustomed to operating such systems. Further, users have usually already decided what to do before they interact with the system. The other problem is that, because of the TCP delay, it cannot be used to send data in real-time, such as audio or time-dependent information [21, 14]. Finally, the web-based application systems have need of the fixed IP. This problem can solve by using the Dynamic DNS (DDNS) and the demilitarized zone (DMZ) in the network address translation (NAT), which is the useful function that has access to Internet directly. However, the server that is in the DMZ is exposed into the defenseless status to the external attacks.

HASMuIM [13] have proposed a novel method using an IM system that is lightweight process, and supports real-time processing. This method can receive modified state information to occur at an IA without reconnecting to Home Server. One of possible disadvantages of this approach is the production of many data packets that transfer the state information changed around the home network. Also, when homeowners reconnect to the server, they may receive many such packets. These packets can be reduced by instructing the server to send the metadata list of updated IAs, i.e., the modified IA state information, only when the homeowner requires it. In the generality of cases, when homeowners interact with their home network, most are only interested in controlling one device. Therefore, downloading the state information that takes no interest is both costly and time consuming. For that reason, we have developed new, reuseable, and robust method that can install to different devices without the reconfiguration.

In this paper, the proposed system can supply any service or information to the homeowner through a portable device for communication. We present an efficient real-time solution for home automation systems by using Jini network, IM, and natural language processing (NLP) technology. The proposed system provides a uniform graphic UI (GUI) for end users. It immediately sends state information from the home network to the homeowner via the Internet. Homeowners with the appropriate IM installed on their mobile phones can remotely control and monitor their home appliances, anytime and anywhere. Furthermore, the system is lightweight, flexible, and extensible. The proposed system is convenient because it can sends a completion or alarm message to a manager via push functionality (NOTIFY message), even if the manager does not reconnect to the home network after it finishes setting the device. The proposed system makes use of a simple sentence for any intention of homeowner, while a web-based interface require hierarchically clicked menus.

2. RELATE WORK

2.1 Natural Language Processing

The Natural Language Processing (NLP) is that computer process human language and returns a reasonable response that people expect. Typically, these methods accept a sentence, generate appropriate parsing trees, and translate it into defined structures.

Question Answering (QA) is defined to be an information retrieval (IR) paradigm to map users Natural Language (NL) questions into the most reliable answers. Recently, many other research, including studies of humanoid robot, embedded this paradigm have been progress. When different technology is designed together with NLP, user interface can be simple, provide powerful uniformed UI tools, and easy way to give the operation that user wants. Many researchers have been studied the techniques with NLP or voice recognition and are used in QA [1]. With the progress of the ubiquitous computing environment, the adoption of NLP including voice processing will be continuously increased.

Our proposed system is similar to QA system. For example, users order or query, written in NL, will arrive at the home server. Then, the home server will analyze the delivered message through shallow NLP technology that does not have semantic analysis, look for the most reliable service name in the service table of the local data-base, and create a new delegate object to carry out the order received from homeowner. This object will connect to Jini network. As using this object, users can control all Jini-enabled devices that he / she wish.

2.2 Instant Messenger

An Instant Messenger (IM) have long been one of the most popular applications on the Internet; they al-low real-time presence and exchange of messages as to keep track of the availability of friends, independent of locale [2]. Also, most IMs merges email, SMS function of mobile phones, multimedia communication such as voice or video chats, and file transfer in an integrated environment. In the case of method relayed by IM servers, even if IM client is shielded by a firewall or NAT router, the clients are able to exchange messages. Also, there is the potential to have one user interface for many types of service, such as universal remote control. For this reason, the popularity of IM is expected to continue increasing and to disseminate throughout various devices in the foreseeable future.

There are two features that we select IM for UI of SHs: near real-time asynchronous bi-directional communications, and lightweight software. Specially, IMs can be implemented on small Internet-enabled devices, such as mobile phones, PDAs, and set-top-boxes, and have the potential that can be used for many types of services. These characteristics of IM will provide the optimized UI in the ubiquitous computing environment.

In the paper [4], they proposed the instant messaging as a platform for the realization of a true ubiquitous computing environment (IMPRUVE) platform and Instant Messenger Information Server (IMIS) that, over heterogeneous IM infrastructure, hypermedia content can be rapidly and efficiently transferred to a designated resident. They build the IM Connectivity API providing the integrated environment for different messengers, control disparate appliances or devices registered through MAPS (Mobile and Pervasive System) in IMIS. MAPS are the subsystem with the ability to automatically register services and devices coming online on a network. They showed that IM have the potential as UI for ubiquitous computing environment.

2.3 Jini Network Technology

Jini consists of a set of Java APIs and high level network protocols for distributed systems. It provides a simple way to perform tasks in home networks such as discovering, registering, and removing devices and services. Jini creates software infra-structure, called a federation of services, which shares access to the services and en-gages in interactions without the prior knowledge of the other systems or any need for human intervention [11].

In the ubiquitous computing environment, one of the various challenges is the ability to automatically register services and devices coming online on a network, and to rapidly retrieve the most reliable contextual information. To build this environment, Jini provides spontaneous networking with self-healing and administration-freeness using various key protocols, although Jini have a limitation on memory spaces. Specially, because Jini can allow downloading services, when the client invokes, from the local or the specific lookup server, it is able to provide the best solution to be suitable for any user environment.

3. PROPOSED TECHNIQUE

In this paper, we present an efficient real-time method for smart homes, which uses Jini network technology and IM system technology to provide a uniform graphic UI (GUI) for end users. In this approach, state information is immediately sent from the home network to the homeowner via the Internet. There are three main agents in this implementation: the Mobile Messenger Agent (MMA) for tools installed in the portable devices, the Home Messenger Agent (HMA) that provides an interface between the MMA, and the Information Appliance Manager Agent (IAMA) which manages the IAs. The overview of the REmote Monitoring and CONtrol method of home appliances (REMOCON) is shown in the figure 1.
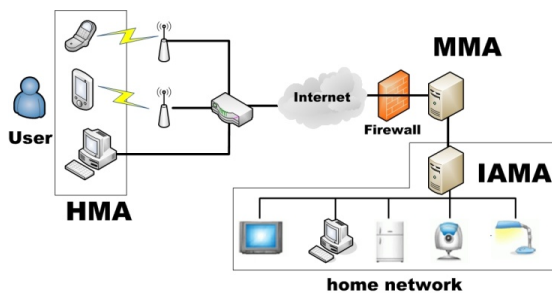

Figure 1. The overall architecture of the REMOCON.

3.1 Assumptions

We assume that the home network uses the network address translation (NAT), and that the residential gateway uses ADSL with dynamic IP addressing. Recently, in the world, almost household Internet environment make use of Ipv4 and DHCP. NAT is quite well known for protecting the home server from a trespasser using the firewall provided naturally, and for providing a practical solution to the impending exhaustion of Ipv4 address space. IM is able to avoid the problem of the DMZ by using the private IP

We assume that the proposed system use sentences for controlling IAs or services. However, NLP is difficult to determine the semantic meaning of a sentence, and to make the analysis of the context to determine the true meaning and comparing that with other text, For reducing some of these problems, each sentence is assumed to have only one service object (or IA) and various attribute values. i.e., we do not consider instructions that involve multiple services.

3.2 Definition of Methods for IAs

In the ubiquitous computing environment, devices shall be operated without authenticated user's intervention, and replaced and added ones should not be harmful the whole system. Therefore, we need to control devices using the same method. However, almost all service and method names are defined by programmers. In the result, without any agreement, it is impossible to change and insert a product. To solve this problem, we presented the easy and efficient method. These methods are divided into three types: direct control, indirect control, and request command.

   1) Set the temperature of the boiler at 45 degrees.

   2) Set the boiler at 45 degrees.

   3) How much is the temperature of the boiler?

   4) Turn the boiler on.

In the above examples, In case of 2), when the system can recognize that the do-main of '45 degrees' is the temperature, 1) and 2) is the same command. It is direct control. In case of 3), it is request command for receiving the state information of an IA. Finally, sentence 4) is direct control to represent the action of devices. The Table 1. is the interface classes of example appliances.

Table 1. The list of interface classes.

Service	Method
Lamp	turnOn()
	turnoff()
Boiler	switchOn()
	switchOff()
	setTime(Calendar dateTime)
	Calendar getTime()
	setTemperature(float degrees)
	float getTemperature

3.3 Device Description Database

When devices connect to Jini network, HMA extract the name of the service and the methods from a service proxy, and then save these in device description database (DD Database). The DD Database consist of following four tables: the *Service table* to save the name of the service, the *Method table* to save the name of the method, *VariableType* table to save the type of a variable, Service Method table to save the relation between tables. The following figure 2. shows the overall structure of DD database.
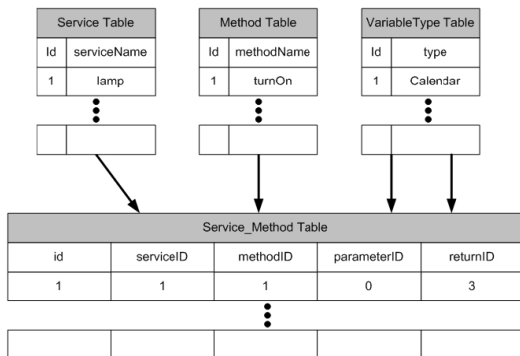

Figure 2. The device description (DD) database.

3.4 Mobile Messenger Agent

The Mobile Messenger Agent (MMA) is the agent that is executed on a personal computer or the homeowner's portable devices, such as a PDA, mobile phone, or smart phone, and is used to indirectly control home appliances connected to the home network. Because the proposed method is based on sentences, the MMA can use a GUI of a general IM system However, in legacy systems, there is no cryptographic algorithm for protecting text messages. To solve this problem, the proposed system is designed to encode messages with the Advanced Encryption Standard (AES) method announced by National Institute of Standards and Technology (NIST) as U.S.

This key is delivered from the home server once the client's messenger is started. AES is one of the most popular algorithms used in symmetric key cryptography. And it is fast in both software and hardware, is relatively easy to implement, and requires little memory. As a new encryption standard, it is currently being deployed on a large scale.

3.5 Home Messenger Agent

The Home Messenger Agent (HMA) is the agent connected in home network. The role of this agent services which the MMA and the IAMA in home network are connected. Also, it manages the homeowners including any additions to the homeowner group.

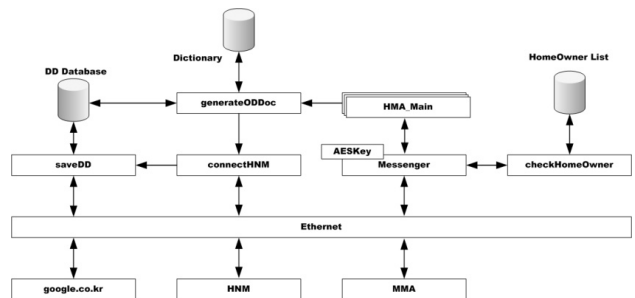When the administrator is connected by wired network device in home, manager function is activated.

Figure 3. The architecture of the home messenger agent.

When the message is delivered to HMA, the messenger function deciphers the message, transfer the plaintext with a tag to *HMA_Main*. *HMA_Main* preserve the connection with the MMA, deliver one to the *generateODDoc*. The *generateODDoc* create the parse tree from this text using the lexicalized probabilistic parser [14], and then generates the OD XML code. The following figure 4. is the translated process.
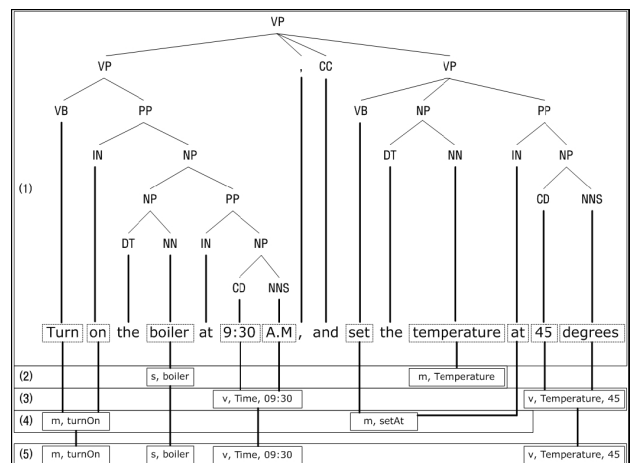

Figure 4. The parse tree for the plaintext

For example, if the plaintext is "turns on the boiler at 9:30 AM, and set the temperature at 45 degrees," then generate the following step (1) in figure 4. In step (2), a NP+PP structure is detected, and finds 'boiler' as a service name and temperature as a method name. The service name is stored in the variable, and the method name is added to the method temporary. From the PP that has CD, *generateODDoc* decides that the domain of '9:30 AM' is a time method, and '45 degrees' is a temperature method, respectively. Using these detected methods, the method temporary is updated. To eliminate overlap in method names, it is deleted the record that has the repetitive method name, and attribute 'm' (step (3)). The *generateODDoc* finds VB tags, and put the IN in the adjacent PP tag together, then checks whether VB belongs to any of these methods. In other hands, join on to Turn, search method table for 'turnOn'. If detected, this word is added to the method temporary (step (4)). The other way, this method is disregarded

because 'setAt' is not detected. Finally, *generateODDoc* is filled out the OD document (step (5)). The figure 5. shows OD XML document created by the example sentence.

```
<?xml version="1.0" encoding="UTF-8"?>
<orderDiscription              xsi:noNamespaceSchemaLocation="orderDocument.xsd"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
        <who>
                <id>mggaibi@......</id>
                <ip>55.230........</ip>
        </who>
        <where>house</where>
        <what>Boiler</what>
        <when>sch</when>
        <how>
                <method>
                        <methodName>setTime</methodName>
                        <parameter>
                                <pName>"2006. 10. 19 9:30:00"</pName>
                        </parameter>
                </method>
                <method>
                        <methodName>turnOn</methodName>
                </method>
                <method>
                        <methodName>setTemperature</methodName>
                        <parameter>
                                <pName>45</pName>
                        </parameter>
                </method>
        </how>
</orderDiscription>
```

Figure 5. The OD XML document created by the example sentence.

3.6 Information Appliance Manager Agent

The information appliance manager agent (IAMA) is to create the new object necessary for connecting with middleware based on OD document transmitted from the HMA, and to monitor the appliance. Because Jini is a centralized System, to download a certain service in the different de-vice must be connected to Lookup server. To complete the proposed system, the IAMA is designed so as to include the Lookup server of Jini to directly manage the Lookup table. So, all IAs, when firstly connecting to Lookup server, is made into OD documents by the *createODD* method called by the modified register function, the *registerAppliance*, in the IAMA, and store local storage device. For checking a message, these documents should also send to HMA.
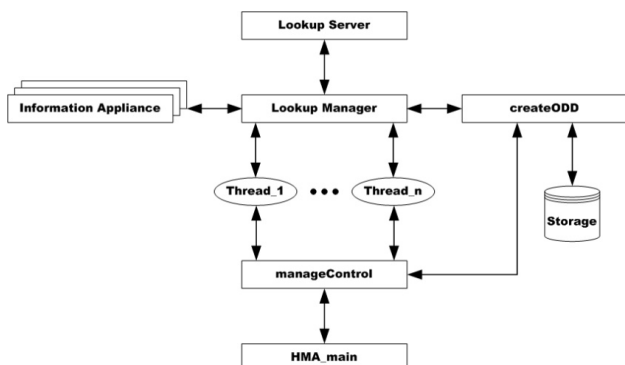
Figure 6. The architecture of the information appliance manager agent (IAMA).

If a problem is encountered at the object created by the *manageControl* method in IAMA, it informs the homeowner of an urgent error message. Otherwise, if the task is completed, it must send a result message to the homeowner. These processes must be performed as quickly as possible. The figure 6. shows the architecture of IAMA.

4. IMPLEMENTATION

The aim of this research is to develop a system for controlling and monitoring home appliances on a smart home onto which an IM was installed. To achieve this environment, we have need of each agent which must operate well even if the internal structure or the programming environment of the other two agents is changed. Therefore, we did not use technologies based on specific programming environments except JINI, so as to operate in the heterogeneous programming environments as well as platforms.

The process of the proposed system is shown in the figure 7. To implement this system, we used Jini middleware to establish an efficient home network and MSN Messenger to connect to the home server from an outdoor area. The proposed system is implemented using Java for HMA and IAMA, C sharp for MMA, XML to write OD document, and MySQL for the DD Database. The HMA is porting in the Embedded Board with Intel Xscale PXA270 and embedded Linux (kernel 2.6.11), and IAMA is running on Windows XP with Java. The MMA is PDA, smart phone, portable phone, or PC.
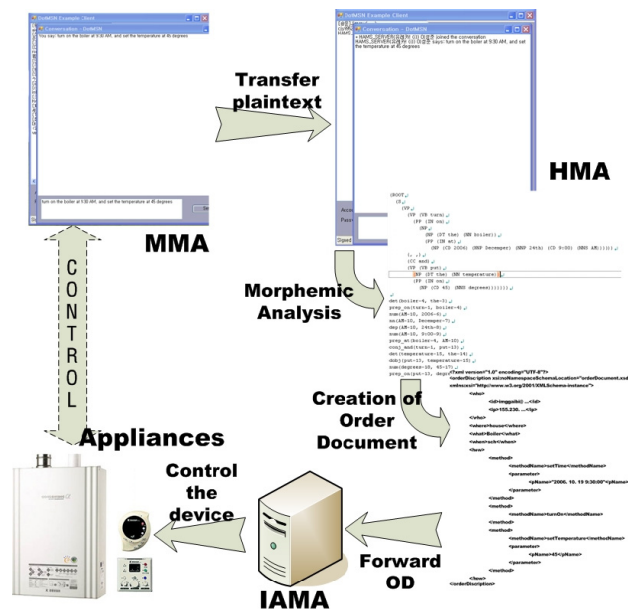
Figure 7. The prototype system setup and the experimental result.

5. CONCLUSIONS

Here we have proposed an efficient method for controlling IAs in SHs. The proposed system is based on an IMS and natural language processing. The method is characterized by five features:

1) User-friendly interface,

2) A notification function,

3) The ability to make use of dynamic IP addresses,

4) Real time processing, and

5) Using input commands written in natural language by the user.

Moreover, input commands could potentially be given verbally by using a voice recognition interface. Compared to the SIP, the proposed method is more suitable for IA control due to its flexibility. Specifically, the proposed method would require little modification if the protocol for communication between the HMA and MMA were changed to a new advanced protocol, because the proposed method makes use of higher-level protocols.

Finally, we believe that, in addition to being used as the UI for smart homes, the pro-posed method has potential applications with other middleware technologies such as UPnP, HAVi, SCP, and so on.

6. REFERENCES

[1] A. Andrenucci and E. Sneiders. Automated question answering: Review of the main approaches. In ICITA (1), pages 514–519, 2005.

[2] C. Dewes, A. Wichmann, and A. Feldmann. An analysis of internet chat systems. In IMC, pages 51–64, New York, NY, USA, 2003. ACM Press.

[3] P. Dobrev, D. Famolari, C. Kurzke, and B. A. Miller. Device and service discovery in home networks with osgi. Communications Magazine, IEEE, 40(8):86–92, 2002.

[4] C. Foley, B. Downes, E. de Leastar, and S. vander Meer. Instant messaging as a platform for the realization of a true ubiquitous computing environment. In eChallenges, 2005.

[5] T. Ha, J. Jung, and S. Oh. Method to analyze user behavior in home environment. Personal and Ubiquitous Computing, 10(2-3):110–121, 2006.

[6] HAVi Organization. The HAVi specification (version 1.1). http://www.havi.org/, 2001.

[7] Hypothetic.org. MSN protocol documents. http://www.hypothetic.org/docs/msn/index.php, 2003.

[8] R. S. Ivanov. Controller for mobile control and monitoring via short message services. In TELSIKS, pages 108–111, 2003.

[9] M. Jeronimo and J. Weast. UPnP Design by Example: A Software Developer's Guide to Universal Plug and Play. Intel Press, 2003.

[10] T. Koskela and K. V¨a¨an¨anen-Vainio-Mattila. Evolution towards smart home environments: empirical evaluation of three user interfaces. Personal and Ubiquitous Computing, 8(3-4):234–240, 2004.

[11] I. Kumaran and S. I. Kumaran. Jini Technology: An Overview. Prentice Hall PTR, Upper Saddle River, NJ, USA, 2001.

[12] R. Lea, S. Gibbs, A. DaraAbrams, and E. Eytchison. Networking home entertainment devices with havi. IEEE Computer, 33(9):35–43, 2000.

[13] S. J. Lee and K. S. Ahn. Control of information appliances using instant messaging. In EUC, pages 976–986, 2006.

[14] H. Y. LI. Web-based remote monitoring and control for process plants. In ICMLC, pages 936–941, 2005.

[15] N. S. Liang, L. C. Fu, and C. L. Wu. An integrated, flexible, and internet-based control architecture for home automation system in the internet era. In ICRA, pages 1101–1106, 2002.

[16] lsmail Cogkun and H. Ardam. A remote controller for home and office appliances by telephone. Consumer Electronics, IEEE Transactions on, 44(4):1291–1297, 1998.

[17] J. Lucenius, J. suomalainen, and P. Ventola. Implementing mobile access to heterogeneous home environment. In HOIT, 2003.

[18] B. A. Miller, T. Nixon, C. Tai, and M. D. Wood. Home networking with universal plug and play. Communications Magazine, IEEE, 39(12):104–109, 2001.

[19] M. Nikolova, F. Meijs, and P. Voorwinden. Remote mobile control of home appliances. Consumer Electronics, IEEE Transactions on, 49(1):123–127, 2003.

[20] O. Omojokun and P. Dewan. Experiments with mobile computing middleware for deploying appliances. In ICDCS Workshops, pages 375–340, 2003.

[21] OSGi Alliance. Osgi Service Platform (release 4). http://www2.osgi.org/Specifications/HomePage, 2007.

[22] M. Rahman and P. Bhattacharya. Remote access and networked appliance control using biometrics features. Consumer Electronics, IEEE Transactions on, 49(2):348–353, 2003.

[23] The Stanford Natural Language Processing Group. The Stanford Parser: A statistical parser (Version 1.5.1), 2006.

[24] M. Yasumura, R. Yoshida, and M. Yoshida. Prototyping and evaluation of new remote controls for people with visual impairment. In ICCHP, pages 461–468, 2006.