2014 IEEE 6th International Conference on Cloud Computing Technology and Science

A Cloud-based Architecture for Home Network System

Satoshi TAKATORI, Shinsuke MATSUMOTO, Sachio SAIKI, Seiki TOKUNAGA, Junho LEE, Masahide NAKAMURA
Graduate School of System Informatics, Kobe University, JAPAN
1-1 Rokkodai-cho, Nada-ku, Kobe, Hyogo 657-8501, Japan
Email:takatori, tokunaga, junho@ws.cs.kobe-u.ac.jp, {shinsuke, masa-n}@cs.kobe-u.ac.jp, sachio@carp.kobe-u.ac.jp

***Abstract*—Managing a home server within individual house is a major obstacle to practical use of home network system (HNS). Delegating the home server to a cloud is a promising approach. However, the conventional multi-tenant SaaS-based solution has the following risks among different households: security/privacy violation, failure propagation and insufficient resource demand. In this paper, we propose a novel cloud-based architecture for the home network system that achieves security isolation, fault isolation and resource isolation. Specifically, we first create a virtual home server for every household using IaaS. On top of every virtual home server, we then create additional virtual machines, each of which contains a single service or application. Finally, using dynamic scaling, we allocate resources needed for individual virtual home servers. Based on the idea, we construct the proposed architecture by three layers: HNS Lite, House Cloud and Service Provider.**

***Keywords*-home network system, cloud-based architecture, virtual machine, security isolation, scaling**

I. Introduction

Research and development of the *home network system* [1] (HNS for short, also called *smart home* in some contexts) have been actively conducted towards practical and extensive use in general households. The HNS provides value-added services by connecting household appliances (e.g., TV, air-conditioner, fan) and sensors (e.g., temperature, humidity, brightness) to a home network. The value-added services (we call *HNS services*) include integrated control of multiple appliances [2], context-aware services [3], and home energy management [4]. The HNS is a typical CPS [5] application in residential domain.

A major obstacle to practical use of the conventional HNS is that every user has to manage a *home server* within the house. Introducing the home server imposes the initial cost of purchasing and maintenance efforts upon the home user.

A straight-forward solution is *home server as a service*, which provides features of the home server as a cloud service. Delegating the home server to a cloud, individual home users do not need to manage the home server by themselves. Each house manages only physical appliances and sensors with minimal network capabilities, while appliance controls and HNS services are provided as SaaS in the cloud.

Due to the nature of the HNS, however, the typical multi-tenant SaaS-based solution has the following three problems.

Problem P1: A malicious attack or software bug may allow shared access to security-sensitive operations and data.

Problem P2: A single failure caused in a service may be propagated to other services or houses.

Problem P3: The resource sharing does not scale for various resource demands from individual houses.

To cope with the above problems, we propose a novel cloud-based architecture for the HNS in this paper. In the proposed architecture, we extensively introduce the virtualization technology to achieve three kinds of isolation:

Security Isolation: We create a *virtual home server* for every household, using IaaS [6]. All the HNS services and data for a house are managed within the dedicated virtual server. Thus, the home server is structurally isolated, so that one house cannot access the home server of another.

Fault Isolation: For every HNS service (or application) installed, we create an additional virtual machine (called *child server*) on top of the virtual home server, so that every HNS service is isolated within the virtual machine. The child server prevents a single failure of a HNS service from being propagated to other services or houses.

Resource Isolation: We measure resource demands from individual houses via the virtual home servers. For each demand, we separately allocate necessary resources for the home servers, using the dynamic scaling of the cloud.

Based on the idea, we design the proposed architecture by three layers: *HNS lite*, *house cloud* and *service provider*. The HNS lite is a light-weight HNS, where only physical devices with minimal networking capabilities are managed without the home server. The house cloud is a cloud that provides HNS services and applications using the isolated virtual home servers. The service provider is a third-party vendor that develops and publishes the HNS services.

Adopting the proposed architecture, the home users are free from the expensive management of the home server, with achieving security, fault and resource isolation. Thus, the proposed architecture can extensively promote practical use of the HNS for general households.

II. Preliminaries

A. Home Network System (HNS)

The *home network system (HNS)* provides value-added services (we call *HNS services*) by connecting household appliances and sensors to a home network. The HNS is

978-1-4799-4093-6/14 $31.00 © 2014 IEEE
DOI 10.1109/CloudCom.2014.125

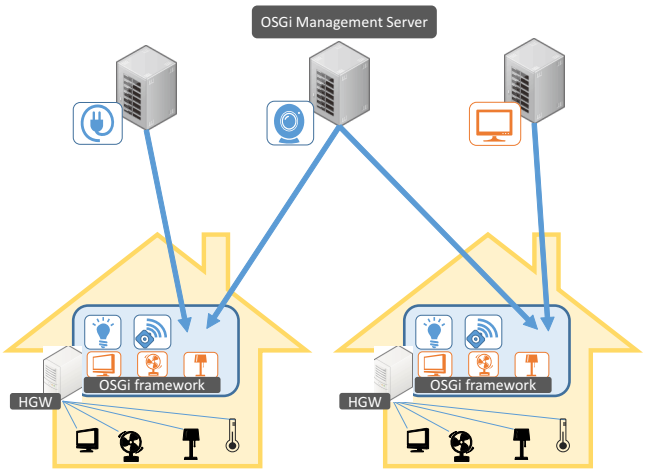


Figure 1. Conventional HNS architecture with OSGi framework

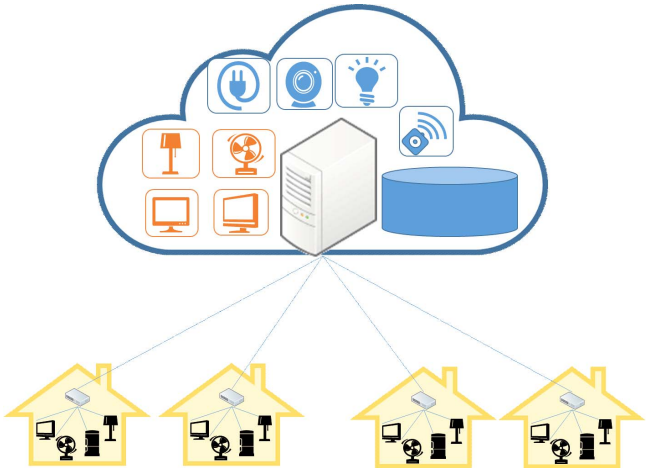


Figure 2. HNS architecture with home server as a service

promising in various domains such as smart city, energy-saving, amenity, security and medical care. Also, several HNS services have already come to the market (e.g., [7]).

Figure 1 shows a typical HNS with the OSGi framework [8]. In every house, a *home server* (also called *home gateway*, denoted by HGW) exists to manage all appliances and sensors connected. In the home server, software applications are downloaded and installed to execute HNS services using the appliances and sensors. The communication protocols for the appliances and sensors vary from a HNS vendor to another, which include infra-red, ZigBee [9], Ethernet, ECHONET Lite [10].

However, the conventional HNS architecture imposes expensive initial cost purchasing a home server upon a home user. It also requires continuous operations and maintenance of the home server. These extra cost and effort for the home server are major obstacle for general households to introduce the HNS. Another limitation is that features and performance of HNS services strictly depend on physical capability of the home server. Hence, it is difficult for home users to flexibly replace or upgrade the appliances.

B. Using Cloud for Home Server as a Service

A straight-forward approach to cope with the problem is to delegate the home server to a cloud service. Figure 2 shows a new architecture of the HNS, which we call *home server as a service*. The idea is to implement all the features of the home server within a cloud service, and share the service among multiple households. In each house, the home server is replaced with a thin terminal with minimal network capability. All intelligent tasks, such as HNS services, appliance controls, sensor reading, are provided as software as a service (SaaS). They are executed within the cloud and maintained by a HNS provider.

Since the costly home server is removed from individual home, the home users are free from the initial cost and the continuous maintenance efforts. The limitation of the service capability is also alleviated significantly, as the features of the home server are implemented using rich and elastic resources of the cloud. Thus, every home user can subscribe the latest HNS services quickly, and quit easily if the user does not like it. Thus, adopting the cloud to the HNS yields great advantages, similarly as we have seen in the cloud migration of the enterprise systems.

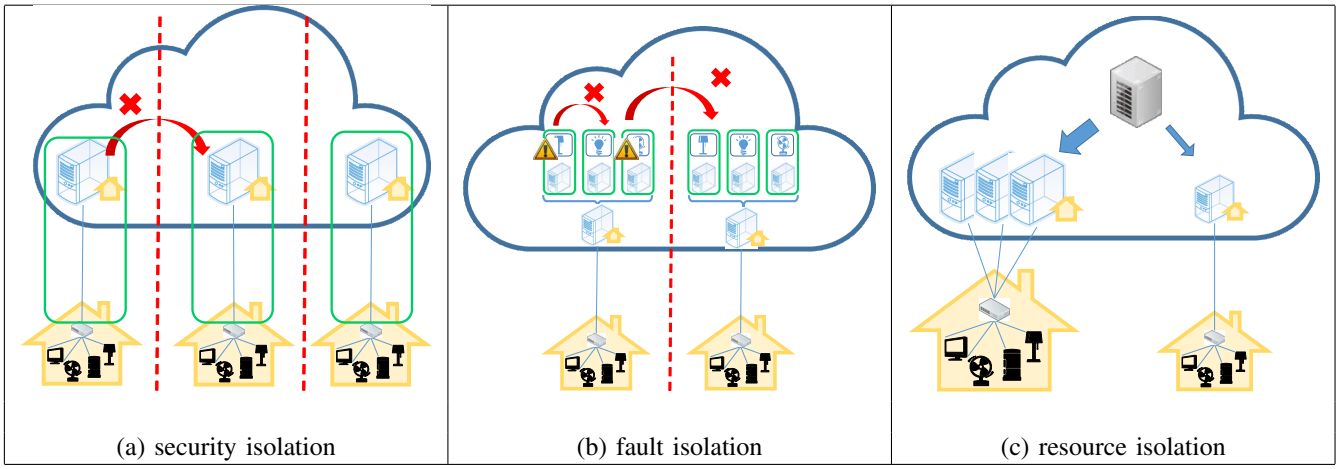
C. Problems in Home Server as a Service

Despite the above advantages, however, the home server as a service has the following three critical problems. These are domain-specific problems to the HNS.

Problem P1 (Shared access to security-sensitive operations and data): The home server as a service manages all HNS services and information within a single cloud service shared by multiple households. Due to the structure, there is a risk that security-sensitive operations and data of a household may be disclosed to other households or outsiders. This may occur by malicious attacks or software faults.

The security violation is critical especially in the HNS domain, since it may lead to damage or loss, crimes, privacy issues. For example, the operation log of appliances can imply the absence of residence, which is good information for robbers to break into the house. Also operating an air-conditioner with a heating mode in a summer will be dangerous environment for babies or elderly people.

Problem P2 (Failure propagation to multiple households): Since the home server as a service is shared by the multiple households, a malfunction of a HNS service may be propagated to all the households using the same service. Even if the failure is caused by mis-operation of a user, other users also have to share the failure. If appliance controls and HNS services become unavailable, they have difficulty in daily life.



(a) security isolation (b) fault isolation (c) resource isolation

Figure 3. Cloud-based HNS with three kinds of isolation

The high availability of the service is important factor for practical use of the HNS. However, the home server as a service structurally allows the failure propagation, which decline the total availability of the service. Moreover, when a failure occurs, it is quite difficult to identify the cause of the failure and location of responsibility.

Problem P3 (Insufficiency for individual resource demand): For a HNS service, different households consume different amount of resources. For example, the housing for two generations require more resources than a single apartment. In general, more people, more rooms and more appliances consume more resources. The lifestyle and daily rhythm also affect the variation of the resources.

The shared service architecture of the home server as a service has difficulty to satisfy the dynamically changing resource demand from individual households. Subscription of new users may decline of service response time or shortage of resources for the existing users.

III. New Cloud-based Architecture for Home Network System

The goal of this paper is to propose a novel cloud-based architecture for the HNS that can cope with Problems P1, P2 and P3. Our approach is to construct a *virtual home server* dedicated for every household using the IaaS virtualization technology, in order to achieve three kinds of *isolation*. In the following subsections, we first explain our key idea, and then propose a concrete architecture.

A. Key Idea

To cope with Problems P1, P2 and P3, in the proposed architecture, we introduce the three kinds of isolation [11]: *security isolation*, *fault isolation* and *resource isolation*.

A1: Security Isolation: *Security isolation* refers to the extent to which a system limits access to (and information about) logical objects, such as files, memory addresses, port numbers, user ids, process ids, and so on [11]. To solve Problem P1, we apply the security isolation to the home server of each household.

Figure 3(a) shows the proposed method. For each house, we construct a dedicated virtual machine of a home server (we call *virtual home server*) using an IaaS technology. We then apply *virtual private network (VPN)* between the virtual home server and a thin terminal in the house, to secure the communication channel (as depicted by a rectangle). For every house, HNS services and related data are installed in the dedicated virtual home server, and the communication is allowed only within the VPN.

As there is no channel between any pair of home servers, all the operations and data access for a household are structurally isolated from other households (as shown in dotted lines in Figure 3(a)). Thus, we achieve the security isolation of the home server, which resolves Problem P1.

A2: Fault Isolation: *Fault isolation* reflects the ability to limit a faulty system from affecting the stored state and correct operation of other systems [11]. To solve Problem P2, we apply the fault isolation to each HNS service installed on the virtual home server.

Figure 3(b) shows the proposed method. For every HNS service installed, we create an extra virtual machine (we call *child server*) on top of a virtual home server. By doing so, every HNS service has an independent memory space. Therefore, even if a failure occurs in a service, the failure is isolated within the child server, and is not propagated to other services or households. Thus, we achieve the fault isolation of the HNS services, which resolves Problem P2.

A3: Resource Isolation: *Resource isolation* corresponds to the ability to enforce the resource consumption of one system such that guarantees the existing resources preserved for other systems [11]. To solve Problem P3, we apply the resource isolation to the virtual home servers preserved for individual households.

Figure 3(c) shows the proposed method. Depending on

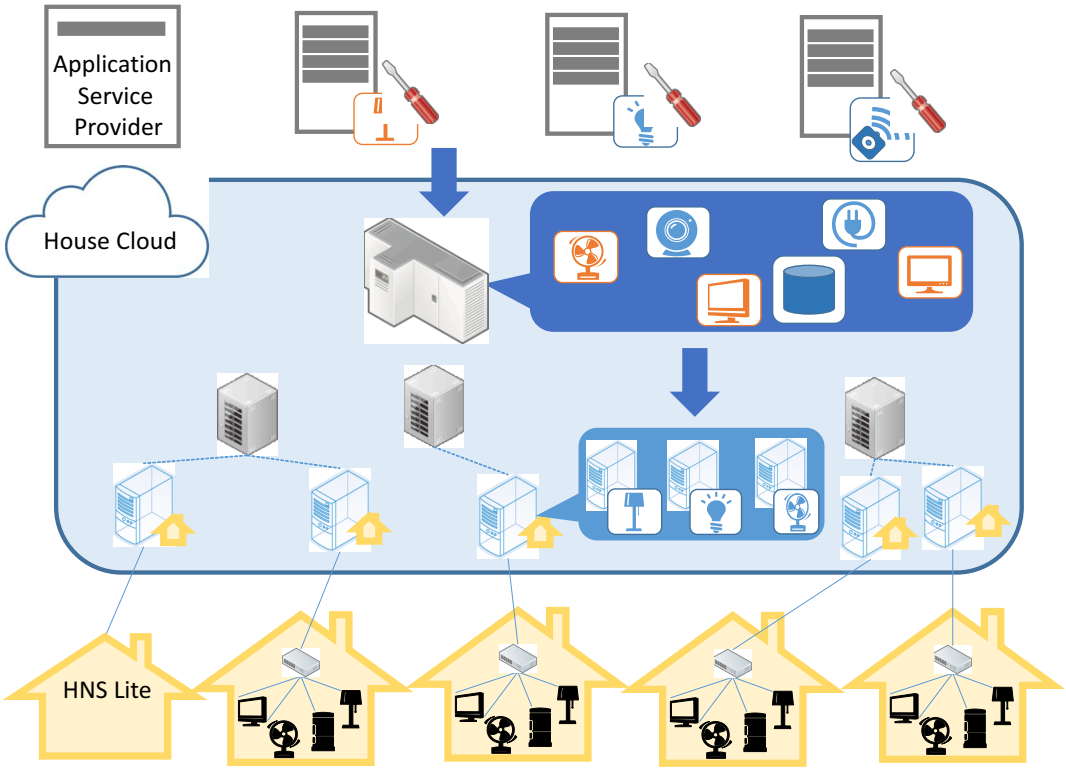


Figure 4. Proposed HNS architecture

characteristics of every household, we allocate appropriate amount of resources when deploying a virtual home server for the house. The more resources are allocated to the virtual home server of larger house with more appliances. During the operation, we measure the resource consumption for each virtual home server. Then we add or delete the resources depending on the demand, by using the dynamic scaling technology of IaaS. The resources are allocated individually to the virtual home server, without affecting resource consumption of other households.

Thus, we achieve the resource isolation for individual households, which resolves Problem P3.

B. Proposed HNS Architecture

Taking the key idea into account, we propose a new architecture of the HNS, as shown in Figure 4. The proposed architecture consists of three layers: (1) *Application Service Provider*, (2) *House Cloud*, and (3) *HNS Lite*.

Application Service Provider refers to any organization that develops and provides software for HNS services.

House Cloud refers to a cloud platform for the HNS services, in which all the HNS services and applications are executed and maintained. For every household, there is a dedicated virtual home server to achieve the security isolation. Each virtual home server contains multiple child servers, where different HNS services (software applications) are installed separately to achieve the fault isolation. The virtual home servers are managed by a cloud infrastructure for appropriate resource management, in order to achieve the resource isolation.

HNSLite is a light-weight HNS that removes a home server from the conventional HNS. Instead of the home server, HNSLite installs a thin terminal, which simply works as a minimal bridge between the home network and the virtual home server. In the following subsections, we explain each layer in details.

C. Application Service Provider (ASP)

Application Service Provider refers to any organization that develops and provides any software for HNS services. The organizations include HNS service providers, individual developers, and consumer electronics makers. The software involves HNS services, appliance drivers, controllers, network adapters, and so on.

Each HNS service is implemented by using service API of House Cloud. The appliance driver is software to connect and control a home appliance with the cloud-based HNS, which is supposed to be provided by a vendor of the appliance. All the services and drivers are registered in a management server within House Cloud. Individual home users choose and install their favorite software applications in their own virtual home servers.

D. House Cloud

House Cloud refers to a cloud platform for the HNS services, in which all HNS services and applications are executed and maintained. As shown in Figure 4, House Cloud is designed to achieve the three kinds of isolation in Section III-A. In this paper, we consider that the following F1, F2 and F3 as primary features of House Cloud.

F1: Managing HNS Services and Drivers: House Cloud manages HNS services and device drivers (for appliances and sensors) within a dedicated management server. When a home user purchases a service, the user can download related software applications from the management server, and install them in the own virtual home service. When a home user purchases a new appliance, the user installs a driver for the appliance to make the appliance available within House Cloud.

F2: Managing Virtual Home Servers: House Cloud creates and manages a virtual home server for every house. The home server is created when a home user creates an account in the initial setup of the HNS. When the user discontinues the HNS, the home server is destroyed from House Cloud for the security purpose. House Cloud also allocate appropriate resources to the virtual home servers. Specifically, depending on the number of services installed and the number of active users, the home server is scaled with respect to the processing power, memory size, storage size. When the home user and the HNS move to a new place, House Cloud migrates the virtual home server, just by redeploying the virtual machine to a near region of the new place.

F3: Managing Child Servers: House Cloud creates and manages a child server for each HNS service installed in a virtual home server. When a user subscribes to a new HNS service, House Cloud instantiates a new child server on top of the virtual home server of the user. Then, software applications and components that depends on the HNS service are installed in the child server. Every child server is equipped with API for fundamental HNS features, consisting of appliance controls, log and data collections, and configuration management of households and devices. Each HNS service is executed using the API, within the child server. Owing to the child server, it is easy to make a backup of every HNS service. Even if a failure occurs, each HNS service can be individually investigated, and smoothly restored when the fault is fixed.

Figure 5 shows an overview of a virtual home server. Every house subscribing to the HNS has a single virtual home server. Within House Cloud, every virtual home server is isolated from other servers, even if the home servers are deployed on the same physical infrastructure. Operations of HNS services and security data access are locally executed within a single virtual home server for each household. On top of each virtual home server, multiple child servers (denoted by VM) are created to isolate different HNS services.

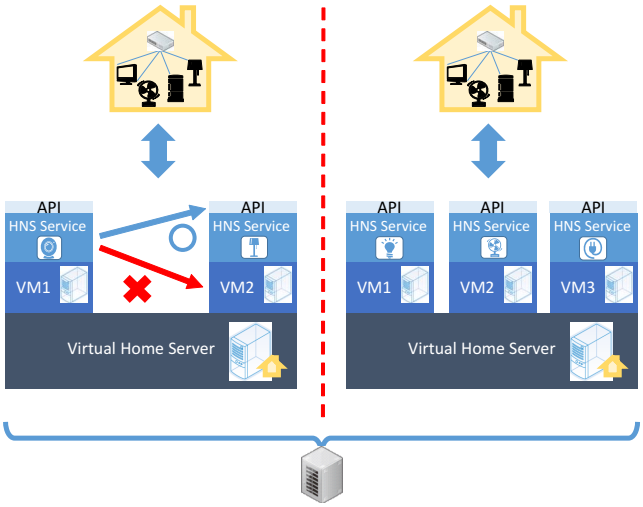


Figure 5. Virtual Home Server Overview

Different HNS services do not share any data or memory space, and communication between them are allowed only via the service level API.

E. HNS Lite

HNSLite is a light-weight HNS that removes a home server from the conventional HNS. According to the commands from House Cloud, HNS Lite operates appliances and sensors to realize the HNS service in a real space. Instead of the home server delegated to a cloud, we install a *thin terminal* in a house. It simply works as a minimal bridge between a virtual home server and appliances and sensors in the home network. Also, the thin terminal is capable to create a session of virtual private network (VPN) to the home server. No other intelligent task is executed on the thin terminal, since the virtual home server takes the responsibility of the task.

IV. Discussion

I will discuss the advantages and limitations of the proposed method.

A. Advantages

First, There is a merit vendor is able to service development easily. In the proposed method, as compared with the case of providing a HNS service in the form of SaaS, since the complexity of the service is reduced, the cost of development can be reduced. It is necessary to manage the access to user data and devices in each service in SaaS format. Therefore, As a result, the complexity of the service increases, development costs increase. In the proposed architecture, the virtual home server is provided to each household, and each service is provided on it. Therefore, access to the data or device of another user is impossible structurally. Because there is no need to implement access

control, the complexity of the service is reduced, and the development cost is also reduced. In addition, it is also possible to reuse the existing HNS services and knowledge of its development prior to use cloud. First, There is a merit vendor is able to service development easily.

Then, There is a merit that portability of the home server is high. In the proposed method, in order to create a virtual home server to each home, it is possible to place the virtual home server another location easily. For example, if there should be some failure in the cloud server of a certain region, it is possible migration to another region immediately. As events that are specific on housing, there is a case in which an address is changed in such move. In the proposed method, it is possible to change the position of the virtual home server easily to fit the place where the user lives. By placing the region of the housing close to the server, it can be expected to improve the response speed of the service HNS. Also, by setting the backup image of the VM even when there is such problems during operation, recovery is possible smoothly.

B. Limitations

As a limit of the structure of the proposed architecture, First, there is a problem of managing the virtual servers is difficult. In the proposed method, the virtual home server is created for each home and various HNS services running on it. Since all virtual servers are independent, the administrator can not be collectively manages each virtual server.

Next, there is a problem of providing services for Smart City [12] is not easy. All Data of each home such as log data is closed within virtual home server and HNS services are selected is different for each home. Therefore, to provide services for Smart City to collect data across multiple homes, performing the device operation is very difficult. As a countermeasure, the development of error detection tools can be considered. Also, it is conceivable that the development of tools to automate the management of virtual servers.

V. Conclusion

In this paper, we have proposed the architecture of distributed cloud HNS platform The proposed architecture have three features security isolation, fault isolation, and resource isolation. In particular, by making use of virtualization technology, to build a home server virtual each every house, it is possible to increase the independence of each home. In the future, first, we implement House Cloud. Then, to perform the evaluation experiment whether the user can actually be used by the system that we have created. In addition, we are planning to address the management techniques of the virtual server and error detection as described in the section IV.

Acknowledgments

This research was partially supported by the Japan Ministry of Education, Science, Sports, and Culture [Grant-in-Aid for Scientific Research (C) (No.24500079, No.24500258), Scientific Research (B) (No.26280115), Young Scientists (B) (No.26730155)] and Kawanishi Memorial ShinMaywa Education Foundation.

References

[1] M. Nakamura, A. Tanaka, H. Igaki, H. Tamada, and K. Matsumoto, "Constructing home network systems and integrated services using legacy home appliances and web services," *International Journal of Web Services Research*, vol. 5, no. 1, pp. 82–98, Jan 2008.

[2] M. Nakamura, H. Igaki, H. Tamada, and K. Matsumoto, "Implementing integrated services of networked home appliances using service oriented architecture," in *Proceedings of the 2nd International Conference on Service Oriented Computing*, 2004, pp. 269–278.

[3] N. Cohen, J. Black, P. Castro, M. Ebling, B. Leiba, A. Misra, and W. Segmuller, "Building context-aware applications with context weaver," in *IBM Research Division*, 2004, pp. 1–12.

[4] D. Niyato, L. Xiao, and P. Wang, "Machine-to-machine communications for home energy management system in smart grid," *Communications Magazine, IEEE*, vol. 49, no. 4, pp. 53–59, 2011.

[5] E. Lee, "Cyber physical systems: Design challenges," in *Proceedings of 11th IEEE International Symposium on the Object Oriented Real-Time Distributed Computing (ISORC)*, 2008, pp. 363–369.

[6] L. Vaquero, L. Rodero-Merino, and D. Morán, "Locking the sky: a survey on iaas cloud security," *Computing*, vol. 91, no. 1, pp. 93–118, 2011.

[7] I. Panasonic, "SmartHEMS," http://www2.panasonic.biz/es/densetsu/aiseg/features/index.html, accessed: 2014-01-04.

[8] S. Helal, W. Mann, H. El-Zabadani, J. King, Y. Kaddoura, and E. Jansen, "The gator tech smart house: a programmable pervasive space," *Computer*, vol. 38, no. 3, pp. 50–60, March 2005.

[9] P. Baronti, P. Pillai, V. W. Chook, S. Chessa, A. Gotta, and Y. F. Hu, "Wireless sensor networks: A survey on the state of the art and the 802.15.4 and zigbee standards," *Computer Communications*, vol. 30, no. 7, pp. 1655–1695, 2007.

[10] N. Ishikawa, "Pucc activities on overlay networking protocols and metadata for controlling and managing home networks and appliances," *Proceedings of the IEEE*, vol. 101, no. 11, pp. 2355–2366, Nov 2013.

[11] S. Soltesz, H. Pötzl, M. E. Fiuczynski, A. Bavier, and L. Peterson, "Container-based operating system virtualization: A scalable, high-performance alternative to hypervisors," *SIGOPS Oper. Syst. Rev.*, vol. 41, no. 3, pp. 275–287, 2007.

[12] R. G. Hollands, "Will the real smart city please stand up?" *City: analysis of urban trends, culture, theory, policy, action*, vol. 12, no. 3, pp. 303–320, 2008.