2015 29th International Conference on Advanced Information Networking and Applications Workshops

Automatic Generation of Social Relationships between Internet of Things in Smart Home using SDN-based Home Cloud

Younggi Kim
Dept. of Computer Science
KAIST
Daejeon, Republic of Korea
Email: younggi521@kaist.ac.kr

Younghee Lee
Dept. of Computer Science
KAIST
Daejeon, Republic of Korea
Email: yhlee@cs.kaist.ac.kr

***Abstract*—The Internet of Things (IoT) idea is having a significant impact on our daily lives these days. As the number of IoT devices is growing fast, many researchers have declared that the usage of IoT and the impact of IoT will make people always use IoT devices whatever they do or wherever they are. At the same time, IoT devices at home are receiving a lot of attention because they contribute to a comfortable home environment. Thus, IoT devices at home are increasing and diversifying. In addition, it is obvious that people will use various home services using various IoT devices. In this situation, it can be extremely difficult for both users and service providers to solve a problem if there is a fault in the smart home environment. To find the position of a fault easily, we define four social relationships between IoTs: IoT-IoT, IoT-Network, IoT-Service and IoT-Physical space relationship. The relationships can be used discover IoT devices, services and resources because it provides a distributed solution that is effective, efficient and reduces the burden on people. In addition, the idea of social relationships guarantees the network navigability of searching. To prevent needless burdens on users, we propose that the four relationships are generated automatically by an SDN-based home cloud. We simulated the automatic generation mechanism with 307 switches and 2007 device nodes. The results confirmed that the relationships are generated properly with high accuracy. The created relationships are stored as a RDF/XML format. The RDF/XML format could be used semantically for such tasks as answering a semantic query or smart home service recommendation. We anticipate that the proposed mechanism will bring huge benefits not only to users but also to home service providers in the purpose of smart home management.**

***Keywords*-Social IoT; SDN; automatic relationship generation;**

I. Introduction

In recent years, not only the number of IoT devices but the number of kinds of IoT devices has been growing fast. The number of IoT devices is predicted to reach 50 billion by 2022. Unquestionably, the IoT idea is having a high impact on several aspects of the daily lives of potential users. Some researchers have predicted that Internet nodes may reside in everyday things such as food packages, furniture, paper documents and more [1]. This prediction emphasizes that people will always use IoT whatever they do or wherever they are. In this trend, home IoT devices are recently getting a lot of attention for a smart environment because many sensors and devices in houses make our life much more comfortable. Many studies have been proposed IoT for health [2], entertainment and security [3] in the house. Similarly, from the perspective of business users, they will be equally expect to fields such as automation and industrial manufacturing.

As the number of home IoT devices is growing, people will use many home services with IoT devices to make their home more comfortable. In this situation, normal users may find it difficult to easily identify a fault point when a home service does not function. The fault could occur among various domains, such as the network or the device. Thus, it may be difficult even for service providers to find the cause of the problem. To prevent this problem, we (1) define the requisite social relationships and (2) propose automatic generation of the social relationships between IoTs for a smart home.

Social relationships make it easy to trace or probe a fault because it has a network navigability aspect. According to [4], social relationships among things can discover IoT devices, services and resources because it provides a distributed solution that is effective, efficient and reduces the burden on people. Thus, the discovery of objects and services is performed effectively and the scalability is guaranteed as in human social networks [5]. However, several proposed schemes have focused on the relationship only between IoT devices. In this respect, we define four relationships to achieve the purpose of a smart home as follows. First, the *IoT-IoT relationship* is the relationship between IoT devices. The relationship consists of two types of relationships, *parental IoT device relationship* and *co-location IoT device relationship*. If an IoT device does not work in a smart home environment, it is intuitive action to ask information to same model IoT devices. At this time, the parental IoT device relationship, which is made of the relationship between same model IoTs, will be used. The co-location IoT device relationship groups the IoT devices in each house. Second, the *IoT-Service relationship* is the relationship between IoT devices and services. When users

978-1-4799-1775-4/15 $31.00 © 2015 IEEE
DOI 10.1109/WAINA.2015.93

execute a service on their IoT device, the controller can make the relationship between them by a packet analysis. The third relationship is the *IoT-Network relationship* which is the relationship between IoT devices and the network that the devices use. When it is necessary to trace or probe to check for a fault, the IoT-Network relationship can help obtain the protocol information. Last but not least, the *IoT-Physical space relationship* is created between IoT devices and the location where the devices are. These four relationships will be useful not only for fault diagnosis but also for a recommendation system.

Although the relationships are useful for a fault diagnosis and a semantic recommendation, generating the relationships would be too complex for users. Thus, we propose an automatic generation scheme to reduce the burden to users and service providers. When an IoT device connects to a home switch, the SDN-based home cloud recognizes the information of the device, such as the model name, manufacturer, network protocol etc. The SDN controller easily captures packets that are passed through SDN switches. In this way, the controller makes a *status graph* containing the information for each home IoT device. After recognizing the IoT device, it creates four social relationships automatically based on the information. To enable a semantic query, we store the status in an RDF/XML format. Using the stored information, we suggest application examples, such as a home diagnosis system and a semantic query system dealing with a home IoT fault.

To simulate the automatic generation, we use an SDN controller and an SDN simulation tool. We made 307 switches and 2007 IoT device nodes using SDN simulation tool. According to the proposed mechanism, the SDN-based home cloud recognizes the IoT device first and creates the four social relationships according to the domain conditions. The results showed that social relastionships were created properly and automatically. We anticipate that the proposed mechanism will bring huge benefits not only to users but also to home service providers in the purpose of smart home management.

In the next section, we briefly introduce the social relationships of IoT, related smart home management studies and SDN studies for the home. In section III, we define the social relationships between IoTs and describe how the relationships are generated automatically. In section IV, we suggest sample applications using the relationships. To simulate the generation, we make each home relationship in section V. Finally, we conclude the paper with future direction in section VI.

II. Related Work

Several studies on social relationships among *things* have been proposed. The authors in [6] showed that physical objects share pictures, comments and sensor data via social network. In that paper, they made a system in which things are able to post their data on *Twitter*. People who have a social relationship with the things are notified of the things' posts. The authors of [5] proposed more specific relationships, a social Internet of Things (SIoT). The types of relationship are suggested between *parental objects*, *co-location objects*, *co-work objects*, *ownership objects* and *social friend objects*. They also suggested an architecture including a SIoT server, gateway and objects for SIoT. We were motivated by these relationships. However, the relationships they proposed are not enough to find a fault point in a smart home.

In [7], they focused on human-centric interaction. When people have a certain purpose, they can use opportunistic relationships that not only have a social relationship between IoT devices but also a human social networking. However, the relationships between IoT devices were not important in that paper. They only focused on the human-centric purpose so the relationship between IoT devices is made as a co-location relationship in [5]. We also consider recognizing the dynamic of joining and leaving IoT devices at home. In addition, our scheme creates various social relationships including the co-location relationship.

Several approaches for smart home management have been proposed. Most of them focused on one domain management. Some researchers proposed policy-based management [8]. However, that approach managed only a network resource that is useful to ISP providers. In addition, event-based management was proposed to detect the resource relations by analyzing an event correlation [9]. Another studies [10] proposed a model to describe causal and temporal relationships between network events to determine faults in network management. On the other hand, several context-aware management methods were proposed to provide smart services by changing the home situation. The studies mentioned above are limited in understanding the entire home situation because only one element among the IoTs is considered. To solve this problem, J. Son proposed a resource-aware smart home management system using resource relation graph [11]. They made a configuration of the service, device, network and physical space domain. By making a resource relation graph using those domains, they could detect the problem point in a home environment. However, the relationships are mostly created by the user. In addition, the social relationship is not considered so the social relationship benefits cannot be taken advantage of. In such a case, generating relationships would be more difficult when the number of IoT home devices grows. Thus, in this paper, we propose the automatic generation mechanism for social relationships.

Some researchers have studied an SDN applied to the home network. In [12], they proposed to provide a user interface to manage the home network easily. Using network virtualization by an SDN for the home network was also proposed in [13],[14]. However, they focused only on the

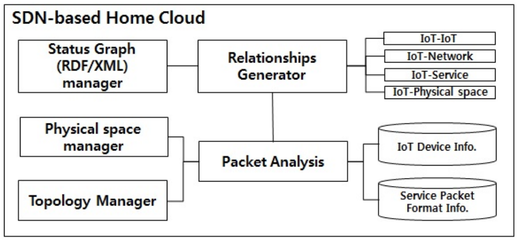

Figure 1. SDN-based home cloud modules

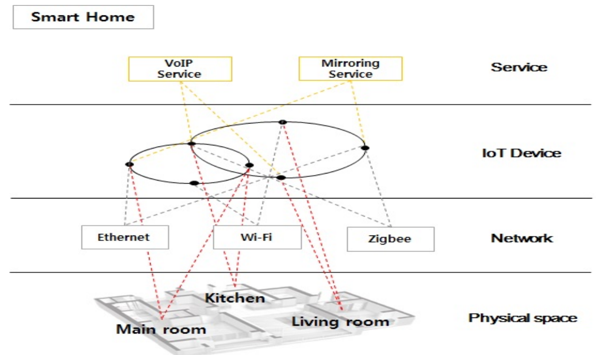

Figure 2. Example of 4 domains in smart home

network management using an SDN. We consider not only the network domain but also the service and physical space domains. We make relationships between those domains, which can be used to manage semantically the smart home. In addition, the SDN concept is useful for recognizing devices or capturing packets. Thus, we use an SDN-based home cloud to make relationships automatically.

III. Automatic generation of relationships using SDN-based home cloud

Considering a fault diagnosis in a smart home, various domains related to IoT devices should be included. According to [5], there are five relationships between IoT devices: parental object relationship, co-location object relationship, co-work object relationship, owner object relationship, social object relationship. Some kinds of relationships could be helpful for the purpose of identifying a fault in a smart home. However, only device domain relationships are not enough. It is useful only when a device itself has a problem. When the problem occurs at a service or network domain, it is difficult for both users and service providers to find the problem point. To include several domains, we propose four social relationships which are created automatically because the configuration would be complex and difficult for normal users to manage them. Thus, we use an SDN-based home cloud as depicted in Fig. 1. IoT device recognition is processed by the packet analysis module and IoT device information database. After the recognition process, relationships are created automatically by the relationships generation module. These generated relationships are stored as a *status graph* and managed by a status graph management module. In addition, the status graph follows a RDF/XML format.

We define four domains as shown in Fig. 2 which are IoT-Service, IoT-IoT, IoT-Network and IoT-Physical space relationship. To reduce users' configuration cost, these relationships are created automatically by SDN controller and SDN switch which recognize IoT information and manage a *status graph*.

A. Automatic IoT recognition

Before generating various relationships, home devices should be first recognized. When a home IoT device connects to a home switch, the SDN-based home cloud must extract the device identity. There have been several proposed standards applying to home devices, such as UPnP [15]. Basically, every home IoT device sends an ARP packet when it connects to the home switch. The SDN controller catches all first packets of the devices connecting to the switch. It is easy for the controller to capture the ARP packets. When the SDN controller captures home devices' ARP packets, it recognizes the device information by MAC address. The information is the device type, the network protocol of device, the device manufacturer and the switch that the device is connecting to. The controller stores this information and makes graphs based on the information. The MAC address is defined in the IEEE Organization Unique Identifier (OUI) [16]. The MAC address consists of 24 bits for manufacturer information and the remaining 24 bits for the host identifier. If a device uses a UPnP protocol, which is used to allow device-to-device networking, it is useful and easy to recognize the IoT device. Based on the recognized information, it will be used to make social relationships.

B. Automation relationship generation

After recognizing IoTs, the controller creates relationships. Because user-defined generation is too complex for users, the system needs to generate four social relationships automatically. The generation process is represented in Fig. 3. Based on the IoT device information, each social relationship is generated and stored in the *status graph*. We explain how each social relationship is generated respectively as follows.

1) IoT-Service relationship: When some IoT devices communicate with each other using an application, the controller recognizes the application by inspecting packets in the SDN-based home cloud. There have been several approaches for the purpose of application-awareness. In [18], they proposed an application-awareness framework by a machine learning (ML) based traffic classification technique in the SDN. They showed that most applications are detected with over 90% classification accuracy. Even if the accuracy rate would be lower, port numbers could be used to categorize applications, especially HTTP. A destination IP address also could be used to determine the service if the address is one of famous servers' addresses. This Layer 3/4-based policy is simple and fast but has limited accuracy. Thus, the controller could use the Layer 3/4-based service recognition first and the ML-based traffic classification afterwards if the simple and fast manner is failed. However, the controller cannot catch a service packet from the IoT device if the packet does not pass through a home switch, such as a bluetooth communication. For this situation, the probability of the missed service information could be reduced if the SDN-based home cloud communicates with home service application download center. When a home device downloads a service, the SDN-based home cloud catches it and makes the IoT-Service relationship. That way, the controller would be aware of the application identity and would make the relationship between the IoT device and the service.

2) IoT-IoT Relationship: The IoT-IoT relationship is the relationship between IoT devices. Following [5] we define two types of IoT-IoT relationship, the parental IoT device relationship and the co-location IoT device relationship. Intuitively, it is natural to query the same device type when a IoT device has a fault. Thus, the parental IoT device relationship is made first when the controller recognizes the joing of the IoT device. By an automatic recognition process using an IoT device information database, the SDN controller knows the model of the IoT device. After the recognition, it makes the parental relationship.

In addition, the co-location relationship of IoT devices is needed to group the home devices from the IoT-Physical space relationship, which involves IoT devices located in the same room such as kitchen, living room etc. When IoT devices have the relationship of being in the same room, they have a co-location relationship. To recognize the co-location information, packet analysis is not sufficient. There are several approaches using indoor location sensing [17]. Most of them use RFID or Zigbee to be aware of the location. We assume that the co-location relationship is generated based on a mechanism among those approaches.

3) IoT-Network relationship: In a home, there could be various kinds of network protocols. Recently, IPv4 or IPv6 have integrated all protocols for the Internet. That is, the Internet of things follows the IP protocol. However, many kinds of protocols for home networking can be used, such as Zigbee or Bluetooth. Therefore, the controller recognizes the network protocol of the IoT device and makes a networking relationship with the available IoT devices. By probing the IoT-Network relationship, network faults can be detected.

4) IoT-Physical space relationship: The physical space domain groups all IoT devices of a home. It could be in a hierarchical structure, for example, the top level is the home and the lower level is a room, such as the living room or the kitchen. By generating the relationship between the IoT device and the physical space, the controller can

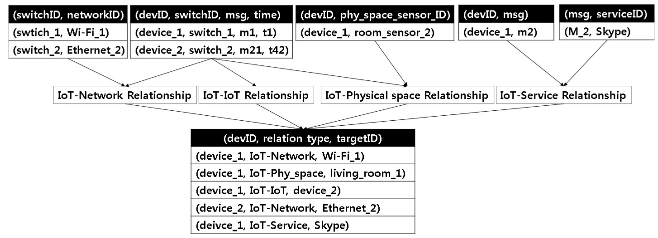

Figure 3. Automatic Relationship Generation Process.

distinguish each home and each home device.

The controller makes a logical node of the physical space according to the IoT device that can be clearly inferred. For example, it makes a 'kitchen' node when a refrigerator joins. However, the physical space of all IoT devices cannot be inferred clearly, for example, a smart phone or a wireless sensor. To create the IoT-Physical space relationship, we assume that sensors exist in each room and then determine the IoT devices that are in the same room.

As each relationship is generated, it is updated to a *status graph*. The status graph is represented by a RDF/XML format. The RDF is a standard model for data interchange on the Web [19]. Storing the information in an RDF/XML format, the graph can be used for a semantic processing. For example, users can send a semantic query and the cloud can answer it. In addition, the cloud can connect web of data, such as DBpedia which is Linked open Data set [20]. In other words, a system could gather more information by querying and receiving from many data set end points.

IV. Application Examples

A. A home IoT fault diagnosis system based on social IoT relationships

A home IoT fault diagnosis system detects a fault dynamically in the IoT environment. When a user at home tries to start a mirroring video service from a smart phone to a smart TV, a problem may occur somewhere. At this moment, the diagnosis system traces from the smart phone to the smart TV through social relationships. First, it traces the smart phone to the mirroring video service through the IoT-Service relationship. If no problem is found, the system traces the network domain by tracing from the smart phone to the smart TV. The user or service provider can determine if the device itself has a problem or it is a network problem or the service in the IoT device has a problem.

B. Semantic query-based solution service for an IoT device problem

Whether the fault is in the IoT device itself or the network or the service, the system can query the SDN-based home cloud. As the relationship is powerful enough to connect to other nodes, the system analyzes the situation and makes a query semantically. For example, a user wants to send something from one device to another device but those devices use different network protocols. The service can grasp this by the IoT-Network relationship and query the same IoT model using the IoT-IoT relationship in the same environment. As a result, some IoT devices can give information that a proper adapter application for this situation using IoT-Service relationship.

V. Evaluation

To simulate the generation of the relationships, we used *mininet* [21], which is an SDN environment simulation tool. We made 100 homes and each home had 2-4 switches. We divided three cases and each home chose one case among them.

- Ethernet + Wi-fi.
- Ethernet + Wi-fi + Zigbee.
- Ethernet + Wi-fi + Zigbee + Bluetooth.

In addition, the switches were connected to the SDN controller, *floodlight* [22]. The floodlight receives packets from home switches and sends packets which order to make flows in switches. We organized that 15-25 IoT devices are installed at each home. We modeled that every house has a main room, living room and kitchen. We made a list of home devices which consists of a PC, refrigerator, IP phone, smart TV, robot cleaner, washing machine, air conditioner, smart phone, table PC, laptop, camera, various sensors and bluetooth devices. In addition, the physical space of some IoT devices, such as a refrigerator is in a kitchen, were fixed and the others were installed randomly among the three rooms. To recognize the physical space, we assumed that there were location sensing devices at each room. Per one second, a new IoT device is connected. As a result, it created 307 switches and 2007 IoT devices nodes by mininet. The average number of IoT devices in each home is around 20.

From the original SDN controller, only the topology with MAC addresses is gathered. However, we propose additional modules to recognize device information and generate social relationships automatically. We assumed that a device can be identified by its MAC address. Thus, the IoT information database in the controller stores the mapping information of MAC address to the device model. After automatic recognition, the relationship generation module makes four relationships. The generated relationships are added to the *status graph* following a RDF/XML format. Each home relationship status is stored in an RDF/XML format.

When the IoT device was connected to a home switch, MAC address information was caught by the SDN controller. It analyzed the model and the switch that the IoT device passed through. From this information, it generated a parental IoT device relationship and an IoT-Network relationship. From the location sensing devices' data, the controller could generate a IoT-Physical space relationship

Table I
The number of created relationships

Relationship	Number of the created relationship
IoT-Service	936
IoT-IoT	1670 + 1643
IoT-Network	1943
Iot-Physical space	1943
Total	8135

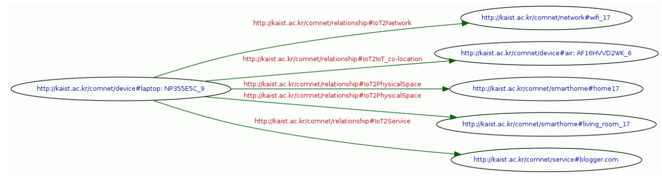

Figure 4. A visualized semantic graph of relationships.

and co-location relationship between the IoT devices. After the generation, the controller analyzed a service packet when the IoT device started to use it. We stored server IP addresses of famous web services such as *youtube*, *facebook*, *twitter*, *google maps* and so on. The controller could recognize IoT device starting to use a web service. However, it is restricted to only web services. If IoT device uses a service which does not use a networking, the controller cannot recognize it except the situation when the controller communicates with a service download center.

As a result, the SDN-based home cloud created several RDF/XML files which contains the information of IoT devices and relationships. As Table I, it showed that the number of IoT-Service relationship was 936. The parental IoT device relationship was 1670 and the co-location IoT device relationship was 1643. In addition, IoT-Network relationship was 1943 and IoT-Physical space relationship was also 1943. The reason the number of IoT-Network and IoT-Physical space relationships were 1943 respectively which is different from the number of IoT devices, 2007, is that we made bluetooth support devices. Because the way of bluetooth devices to communicate with each other does not pass through a switch, the controller could not catch the information of the devices and the network. In addition, only web services could be recognized because of the same reason. From the created RDF/XML files, we could visualize the semantic graph using W3C RDF validation service [23] as Fig. 4.

VI. Conclusion

As the number of IoT devices is growing, it can be extremely difficult for users when there is a fault in a smart home environment. To detect a fault point, we defined the relationship types between IoTs and proposed a mechanism for automatic generation of relationships in the smart home using an SDN-based home cloud. The relationships are expected to be effective, efficient and used to make people comfortable. In addition, the idea of social relationships guarantees the network navigability of searching. Thus, we defined four relationships: IoT-IoT, IoT-Network, IoT-Service and IoT-Physical space relationships. These four relationships were generated automatically by an SDN-based home cloud. In addition, we simulated the automatic mechanism using SDN-based home cloud with 307 switches and 2007 device nodes. As a result, each smart home configuration is determined in an SDN-based home cloud by connecting relationships. We expect that the automatic relationship generation will bring huge benefits not only to users but also home service providers in terms of ease of smart home management.

As further studies, probing and tracing a fault diagnosis system would make a reliable smart home environment based on proposed social relationship between IoTs. As we suggested in section IV, diagnosis could be much easier when it uses the social relationships. After finding a fault point, the categorization of a fault would be needed to solve a problem in a smart home environment. A solution mechanism according to a fault type would be another topic to investigate.

Acknowledgment

This work was supported by ICT R&D program of MSIP/IITP. [B0101-14-0334, Development of IoT-based Trustworthy and Smart Home Community Framework]

References

[1] L. Atzori, A. Iera, and G. Morabito. ”The internet of things: A survey”, *Computer networks*, Vol. 54, No. 15, pp. 2787-2805, 2010.

[2] L. Haiyan, C. Song, W. Dalei, N. Stergiou, and S. Ka-Chun, ”A remote markerless human gait tracking for e-healthcare based on content-aware wireless multimedia communications,” *IEEE Wireless Communications*, Vol. 17, pp. 4450, 2010.

[3] X. Li, R.X. Lu, X.H. Liang, X.M. Shen, J.M. Chen, and X.D. Lin, ”Smart Community: An Internet of Things Application”, *IEEE Communications Magazine*, Vol. 49, pp. 6875, 2011.

[4] L. Atzori , A. Iera, and G. Morabito, ”Siot: Giving a social structure to the internet of things”, *Communications Letters, IEEE*, Vol. 15, No. 11, pp. 1193-1195, 2011.

[5] L. Atzori, A. Iera, G. Morabito, and M. Nitti, ”The social internet of things (siot)–when social networks meet the internet of things: Concept, architecture and network characterization”, *Computer Networks*, Vol. 56, No. 16, pp. 3594-3608, 2012.

[6] M. Kranz, Luis Roalter, and Florian Michahelles. ”Things that twitter: social networks and the internet of things”, *What can the Internet of Things do for the Citizen (CIoT) Workshop at The Eighth International Conference on Pervasive Computing (Pervasive 2010)*, 2010.

[7] B. Guo, D. Zhang, Z. Wang, Z. Yu, and X. Zhou, ”Opportunistic IoT: exploring the harmonious interaction between human and the internet of things”, *Journal of Network and Computer Applications*, Vol. 36, No. 6, pp. 1531-1539, 2013.

[8] A. I. Rana, and M. O. Foghlu, ”New role of policy-based management in home area networks-concepts, constraints and challenges”, *New Technologies, Mobility and Security (NTMS), 2009 3rd International Conference on. IEEE*, 2009.

[9] J. P. Martin-Flatin, G. Jakobson, and L. Lewis, ”Event Correlation in Integrated Management: Lessons Learned and Outlook”, *Journal of Network and Systems Management*, Vol. 15, No. 4, 481-502, 2007.

[10] C. C. Hsu, and L. Z. Wang, ”A smart home resource management system for multiple inhabitants by agent conceding negotiation”, *Systems, IEEE International Conference on Man and Cybernetics*, pp.607-612, 2008.

[11] J. Y. Son, J. H. Park, K. D. Moon, and Y. Lee, ”Resource-Aware Smart Home Management System by Constructing Resource Relation Graph”, *Consumer Electronics, IEEE Transanctions on*, Vol. 57, No. 3, pp. 1112-1119, 2011.

[12] T. Fratczak, M. Broadbent, P. Georgopoulos, and N. Race, ”HomeVisor: adapting home network environments”, *Software Defined Networks (EWSDN), 2013 Second European Workshop on IEEE*, 2013.

[13] R. Mortier, et al., ”Supporting novel home network management interfaces with OpenFlow and NOX”, *ACM SIGCOMM Computer Communication Review*, Vol. 41, No. 4, 2011.

[14] Y. Yiakoumis, et al., ”Slicing home networks”, *Proceedings of the 2nd ACM SIGCOMM workshop on Home networks*, 2011.

[15] ISO/IEC 29341-1: ”Information technology – UPnP Device Architecture – Part 1:UPnP Device Architecture Version 1.0”, 2011.

[16] IEEE Standard Association. ”IEEE OUI and company id assignments.” (2006).

[17] A. Montaser, and O. Moselhi, ”RFID indoor location identification for construction projects”, *Automation in Construction*, Vol. 39, 167-179, 2014.

[18] Z. Qazi, et al., ”Application-awareness in SDN”, *Proceedings of the ACM SIGCOMM 2013 conference on SIGCOMM*, 2013.

[19] Klyne, Graham, and Jeremy J. Carroll. ”Resource description framework (RDF): Concepts and abstract syntax”, 2006.

[20] S. Auer, et al. ”Dbpedia: A nucleus for a web of open data”, *Springer Berlin Heidelberg*, 2007.

[21] B. Lantz, B. Heller, and N. McKeown. ”A network in a laptop: rapid prototyping for software-defined networks”, *Proceedings of the 9th ACM SIGCOMM Workshop on Hot Topics in Networks*, 2010.

[22] Floodlight, http://www.projectfloodlight.org/.

[23] W3C RDF Validation Service, http://www.w3.org/RDF/Validator/.