INTEGRATION OF RESTFUL WEB OF THINGS FOR SMART HOME ENERGY MANAGEMENT

[1]BHAGYA NATHALI SILVA, [2]MURAD KHAN, [3]CHANGSU JUNG, [4]JIHUN SEO, [5]YONGTAK YOON, [6]JEONGYUN KANG, [7]KIJUN HAN

[1,2,3,4,5,6,7]School of Computer Science and Engineering, Kyungpook National University, Daegu, Korea
E-mail: [7]kjhan@knu.ac.kr

Abstract - Internet of Thing (IoT) concept has matured with the evolution of smart things. Smart home notion is highly favored with the extensive attention on IoT. However, realistic implementation of energy efficient smart home architectures is hindered by platform incompatibilities among heterogeneous smart devices. Web of Things (WoT) expose these connected devices over the web in a more transparent and effective way. Thus, we proposed a WoT based sensor network to collect and share data of a smart home employing representational state transfer (REST) architecture. The smart home scenario is tested through computer simulation. The simulation results revealed the performance enhancement in terms of energy efficiency of domestic appliances, device discovery performance, response time, and failed transmissions.

Keywords - Smart Home; Energy Management, Web of Things, Sensor Networks

I. INTRODUCTION

Internet of Things (IoT) is a collection of heterogeneous devices that are uniquely addressable and capable of collecting and sharing data, while making intelligent decisions with less or without human interaction [1]. IoT concept is expanded to various interest areas i.e. smart home, smart city, smart grid, smart healthcare, etc. [2, 3] connecting various smart things. However, seamless communication among these devices is hindered by the platform incompatibilities. Thus, it creates an essential demand for a universal platform, which is tedious to build from the scratch. Owing to ubiquitous cross-platform interoperability, web is chosen as the ideal candidate to satisfy the universal platform demands. As a result, the experts suggest to exploit well formed web standards and protocols to build Web of Things (WoT), which acts as the universal platform for heterogeneous devices in IoT applications.

Home automation is widely used in security locks, heating and air conditioning control, and central controlling of lights. In the recent past, home automation has deviated from its conventional concept to a domestic energy conservational concept. The drastic increase in domestic energy consumption resulted rapid deterioration of non-renewable energy sources, while increasing the monetary value of energy. Currently, both experts in industry and academia are focused on improving energy efficiency of domestic appliances. Nevertheless, lack of uniformity reduces the ubiquitous accessibility and controllability of heterogeneous smart devices, which are crucial for effective domestic energy management. With the emergence of WoT notion, it allows to integrate smart things into the web. Consequently, these devices become accessible over the web anytime and anywhere. Thus, we proposed a smart home architecture integrated with RESTful WoT, which aims to uplift the energy efficiency with its seamless and ubiquitous accessibility.

The rest of the paper is organized as follows: Section 2 presents a brief overview of the similar architectures and methods used in the field of WoT. Section 3 illustrates the proposed scheme in detail. The simulation setup and results are discussed in Section 4. Finally, the conclusion is given in Section 5.

II. STATE OF THE ART

Integration of heterogeneous wireless networks enable WoT to provide cost effective and fast services with high-speed connectivity.

Energy efficient communication protocols are highly favored for smart device communication, since a majority of the smart devices is controlled through battery-operated sensor [4, 5]. In [6], the authors addressed home automation in an energy efficient and cost effective manner by introducing z-wave short-range wireless technology. However, the z-wave technology does not specify the interoperability with the Internet protocols. Therefore, a dedicated gateway is required to translate the z-wave protocol format into an acceptable format for the Internet protocols [6]. As stated before, the utmost goal of WoT is to design a generic and well-established architecture to provide ubiquitous connectivity between the objects and the Web [7]. Fielding et al. integrated smart objects with the Web using REST architecture with the HTTP protocol [8]. However, the topology and localization of sensor nodes were not considered in [8] when integrating the smart things with the web. Even though HTTP is considered as the de facto protocol for RESTful architectures, TCP communication hinders its applicability on resources constrained smart devices.

In order to address the existing issues, internet engineering task force (IETF) developed a constrained RESTful group (CORE). It developed constrained

application protocol (CoAP), which transfers data of resource constrained smart objects with a minimal power and processing in a WoT environment [9]. Moreover, CoAP supports resource caching and device discovery via a limited set of requests such as GET, POST, and DELETE. Since CoAP operates on user datagram protocol (UDP), the reliability is obtained through a built-in retransmission mechanism. Along with the promising features of CoAP, it has few disadvantages. The standardization of CoAP is not finalized and the interoperability is restricted for CoAP.

Integrating web services with the smart devices became prominent and promising in the recent past. Various architectures were proposed to enable the interaction between web services and smart devices. The RESTful Web services are becoming popular for the Web connectivity of heterogeneous smart devices [10]. Similarly, RESTful provide application programming interfaces (API) that support the REST style. Thus, RESTful API is considered to be a convenient and simple solution for the next generations of constraint oriented applications in WoT environments.

III. PROPOSED FRAMEWORK

In this work, we propose a WoT architecture that incorporates a smart gateway to expose the services over the web. The architecture is explained in two folds 1) smart home energy management with the RESTful WoT integration and 2) service representation over the web. The appliances of the smart home are fixed at specific places and uniquely identifiable via the IDs that are assigned to the sensors, which controls each appliance.

A. Home Energy Management

Energy management is in the spotlight of modern smart home applications due to its importance in cost reduction and non-renewable energy conservation. The integration of smart home with RESTful WoT allows users to view the status of their smart home using any web browser. All domestic appliances are represented via a hyperlink on the respective web page. The entire web oriented smart home architecture is presented in Figure 1.

The smart home consists with a controller unit (CU) and a management station (MS). The CU consists with non-IP appliances and Zigbee sensors, which are deployed in the house to gather data corresponding to energy consumption and ambient environment. The ZigBee sensor attached with each appliance controls the operation of the appliance. In order to expedite the device discovery, each ZigBee sensor is connected with a ZigBee coordinator that is further connected with the MS. ZigBee coordinators are placed to reduce the packet loss due to coexistence interference and the distance between sensors and MS. Moreover, coordinator placement reduces the energy consumption for data transmission of the nodes, since it reduces unnecessary hops between sensors and MS. The residents of the smart home turn on the appliances manually and the CU turn off the devices upon receiving commands from MS. Therefore, CU periodically checks for user existence and notifies MS. The event manager of MS handles the communication between CU and users. MS is capable of real-time event generation to enforce energy efficiency of the smart home. The MS coordinates self-generated controls and user-generated controls with the event manager.

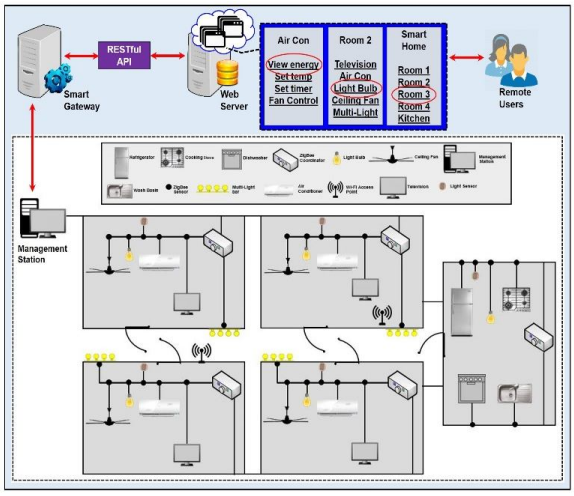


Fig. 1 Proposed web oriented smart home energy management system

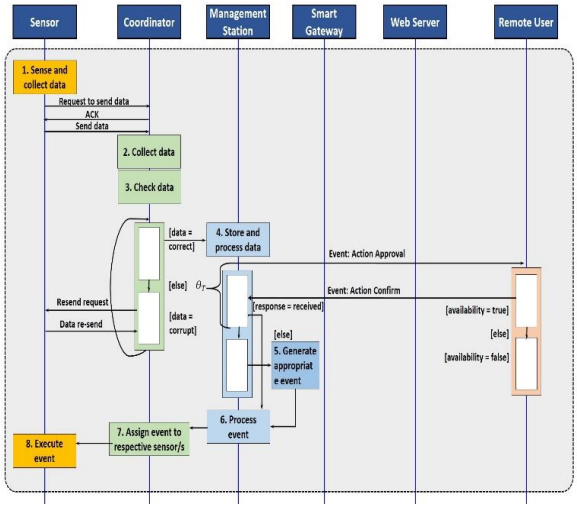


Fig. 2 Communication between smart home and remote user for self-generated controls

Upon noticing user absence, self-generated controls are generated by the MS to turn off unnecessary appliances. However, the MS notifies the users and waits for the user response. Once the waiting time (α_t) is expired, MS make decisions that align with previous user responses in the decision module. Hence, the decision module ensures that user unavailability to respond for a notification does not cause a negative impact on the smart home operations. Figure 2 represents the communication

flow between a remote user and smart home architecture for self-generated controls.

On the other hand, user-generated controls are generated externally and received by the MS via the smart gateway. The RESTful API translation enables the users to access the smart home remotely via the web. The APIs expose the smart home services i.e. turn on and turn off appliances, view energy consumption level, and device specific controls. Worthy to note that user-generated controls do not involve α_t waiting time, since the user has initiated the event. Upon receiving the user request, MS checks for requested information and if it is available on MS the response is immediately sent to the user. However, if the requested information are not available at the MS, it identifies the corresponding ZigBee sensor and request is forwarded via the ZigBee coordinator. The identification has become easier with the self-describing HTTP messages. The communication flow of user-generated controls are illustrated in Figure 3.

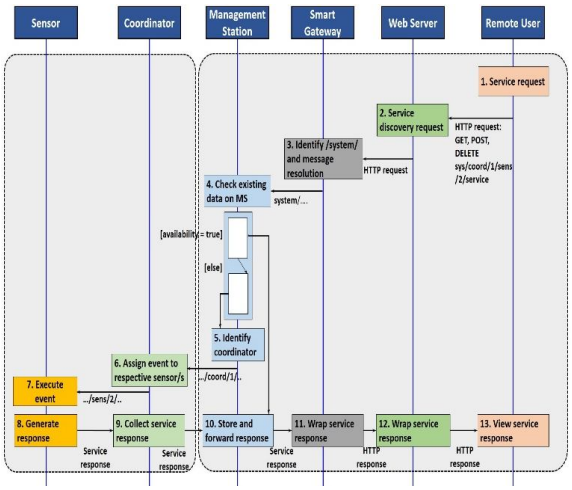


Fig. 3 Communication between smart home and remote user for user-generated controls

B. Service Representation over the Web

The web facilitates ubiquitous accessibility to domestic appliances via the web. Consequently, smart devices act as any other web resource in the web. Generally, the devices in WoT environments comply with implementing TCP/IP protocol over existing network and implementing a web server complying with HTTP 1.1 web server. Nevertheless, not all smart devices adhere to these requirements due to memory, computational, and bandwidth limitations. Hence, we proposed a smart gateway to compensate the HTTP and TCP/IP requirements of heterogeneous smart devices in the smart home. In other words, smart gateway bridges the incompatibility gap between the web and smart devices, while hiding internal details from the smart home users.

In order to enable seamless communication between devices, sensors, coordinators, and remote users, the smart gateway resolves proprietary protocols e.g. ZigBee. Smart gateway shares sensor data related to environmental parameters or smart home appliances via the RESTful APIs embedded to it. The RESTful architecture binds uniform resource identifier (URI) to the functionalities of virtual smart things on the web. As a result, remote smart home users can request home services with a self-describing HTTP request message.

For example, the remote smart home user makes an HTTP request to get current temperature of room 3. The web server passes the HTTP request: /sys/MS/coord/3/sens/3/get_temp(). Sequentially, the appropriate sensor is located from the self-describing HTTP request. After identifying the corresponding sensor, it is passed to the smart gateway, which translates the HTTP request into the ZigBee format (service translation for proprietory protocols). Sequentially, smart gateway captures the translated message, passes it to the MS of the smart home to process the service request as mentioned in the previous section. Similarly, the MS returns the service response and smart gateway converts the ZigBee response into a HTTP response with the aid of the RESTful architecture in order to convey the response to the user.

IV. PERFORMANCE EVALUATION

The simulation was designed to evaluate total energy consumption of the smart home and the network performance in terms of response time and transmission failure.

A. Simulation Environment

As shown in Figure 1, we simulated the smart home scenario along with the MS, smart gateway, and web server. The smart gateway and MS are installed on a Core I3 CPU 3.60 GHz computer. The RESTful framework and web server are installed on Windows Server 2008 operating system in another computer. The smart home scenario is tested in C# programming language. Each household appliance is attached with a ZigBee sensor. ZigBee sensors follow the IEEE 802.15.4 protocol architecture for communication. The coordinator nodes act as Full-Function Device (FFD), which coordinates a personal area network. The rest of the nodes are considered as Reduced-Function Devices (RFD) that are with limited resources and communication capabilities. A star topology is followed by considering one FDD node in each room and kitchen attached to multiple sensors.

The IEEE 802.15.4 does not define a network layer, therefore we made the modification for routing the data following a multi-hop environment. Sensor nodes immediately connect with the coordinator and the coordinators collect and send data to the MS. Similarly, the MS is enabled to follow a multi-hop routing strategy of IEEE 802.11n standard. Further, an automatic topology adaptation is introduced to take care of the changes made in the network. For example, if a new node joins the network, the topology will

change automatically to adapt to the arrival or depart of a node. Thus, it makes the system more dynamic and adaptive and it is able to handle a large number of nodes joining and leaving the network at once.

B. Energy and Network Performance Evaluation

The proposed WoT-ESN architecture was tested under a heavy load that are generated by multiple home users to check the details of a household appliance through the Web. The request generation frequency was limited to one request per one minute for a single user. In a typical household, number of users are less than the number of appliances. However, we simulated a scenario where number of users are greater than the number of appliances. Such that, we can evaluate the absolute performance level of the proposed architecture.

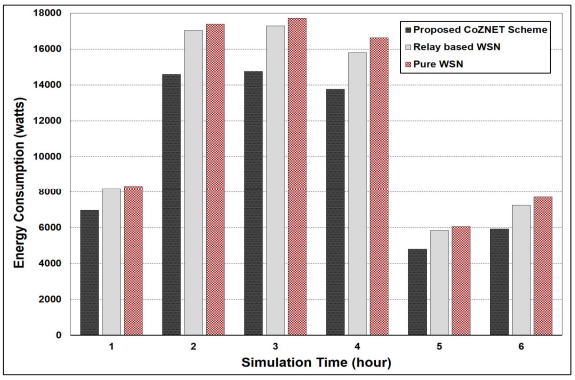


Fig. 4 Energy consumption of entire smart home architecture

For a duration of six hours, the operational energy consumption of all domestic appliances were obtained for the proposed scheme, relay based WSN, and pure WSN. Consequently, the results were compared with each other to analyze the energy consumption behavior. The user activities are random. As an example, a user can go to any room or kitchen and switch on any household appliance he prefers. Unlike pure WSN, the household appliance is switched off following an approval from the user through the Web, in order to conserve energy while preserving user comfort. However, if a user is not available, the user's previous context is utilized to determine the corresponding action. The results revealed that an average of 15% energy is saved in the case of proposed scheme as shown in Figure 4.

The discovery time corresponding to all the sensors available in the smart home architecture is illustrated in Figure 5. All sensors attached with devices are registered in the web server via the coordinator and the MS. Thus, it requires significantly a lesser time to discover with less number of sensors. However, with a large number of sensors it requires more time to load all the sensors into the framework. We experimented to discover 30 sensors that are present in the smart home architecture. The experimented results revealed that if 20 sensors are turned on and discovered at once, it requires around 19 s to fully connect with the framework. However, the discovery time rapidly increases if more than 20 sensors are turned on at once.

Another experiment is performed for 10 minutes to evaluate the response time and fail transmissions of the proposed scheme. We placed the webpage on a separate computer, while the MS and sensors are presented in a different computer. In addition, both the webpage and sensor and MS are present on a different network to check the performance of the multi-hop concept in the proposed scheme. As shown in Figure 6, the users randomly sent requests to three, five, and seven household appliances.

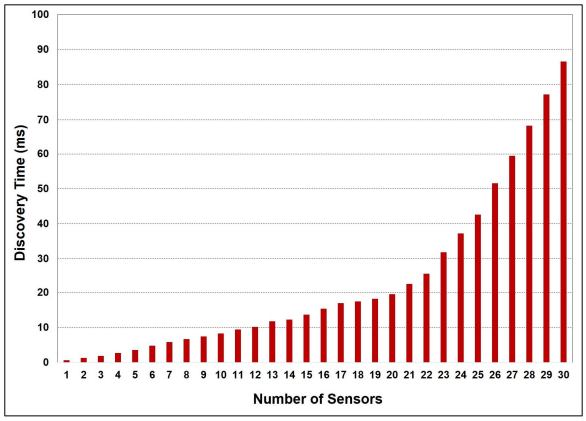


Fig. 5 Device discovery analysis

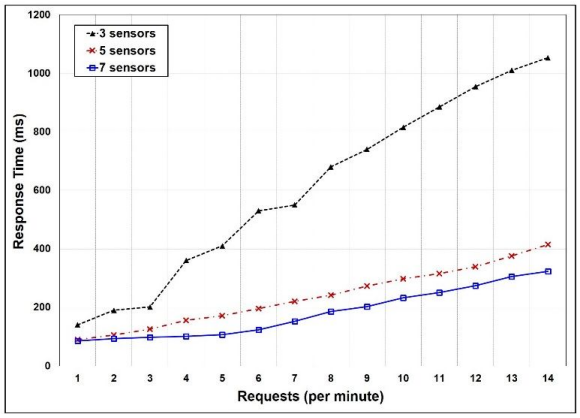


Fig. 6 Response time analysis

According to the results, the response time increases with the number of requests. Nevertheless, the response time significantly decreases with increasing number of household appliances. The division of the requests into processes remarkably reduced the response time even if the requests are for the same household appliance.

In fact, the probability to interfere with each other increases with the proximity to each other. Thus, it can increase the number of failed attempts to send requested data to the Web. In order to simulate a similar scenario, the sensor nodes available in the smart home scenario are divided into four groups. The groups 1, 2, 3, and 4 are available at 3, 6, 9, and 12 meters apart from each other. The group of sensors available 3 meters are the ones that are highly

affected from the interference. Hence, it was expected to have the highest fail transmission attempts. The results clearly revealed that the number of failed attempts are more than 3% of 120 requests/minute. However, these are the worst conditions, which is far better in real time scenarios by implanting the proposed scheme. Figure 7 illustrates the worst case scenario, which we directed the requests to sensor groups in 3 meter interference range.

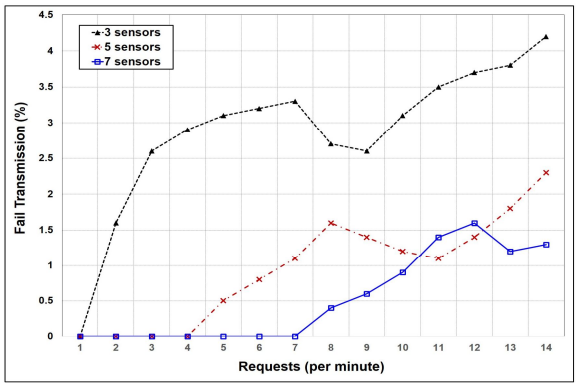


Fig. 7: Failure Transmission Analysis

CONCLUSIONS

Providing remote accessibility to smart home services is hindered by heterogeneous technologies. Thus, web has been chosen as the ideal candidate to unify these technologies at the application layer. Thus, we proposed a WoT based architecture to access smart home services via the web, in order to achieve energy conservation demands.

The WoT integrated smart home scenario is tested and simulated to evaluate various performance characteristics. The energy consumption of household appliances in the proposed web integrated architecture is compared with pure WSN and relay-based WSN scenarios. The results show that adding user initiated semi-automated control of the appliances has optimized the energy consumption of household appliances. The proposed web architecture improved its performance with respect to response time, failure transmission, and device discovery. This work can be further extend to various smart buildings that aim to enhance the energy efficiency of appliances and the network.

REFERENCES

[1] B. N. Silva, M. Khan and K. Han, "Internet of Things: A Comprehensive Review of Enabling Technologies, Architecture, and Challenges," IETE Technical Review, 2017.

[2] M. Khan, B. N. Silva and K. Han, "Internet of Things Based Energy Aware Smart Home Control System," IEEE Access, vol. 4, pp. 7556-7566, 2016.

[3] B. N. Silva, M. Khan and K. Han, "Big Data Analytics Embedded Smart City Architecture for Performance Enhancement through Real-Time Data Processing and Decision-Making," Wireless Communications and Mobile Computing, pp. 1-12, 2017.

[4] D. Yi and H. Yang, "HEER – A delay-aware and energy-efficient routing protocol for wireless sensor networks," Computer Networks, vol. 104, pp. 155-173, 2016.

[5] B. Zeng and Y. Dong, "An improved harmony search based energy-efficient routing algorithm for wireless sensor networks," Applied Soft Computing, vol. 41, pp. 135-147, 2016.

[6] C. Gomez and J. Paradells, "Wireless Home Automation Networks: A Survey of Architectures and Technologies," IEEE Communications Magazine, vol. 48, pp. 92-101, 2010.

[7] D. Guinard, V. Trifa, F. Mattern and E. Wilde, "Architecting the Internet of Things," in From the Internet of Things to the Web of Things Resource-oriented Architecture and Best Practices, Berlin, Springer, 2011, pp. 97 - 129.

[8] R. T. Fielding and R. N. Taylor, "Principled design of the modern web architrcture," ACM Transactions on Internet Technology (TOIT) , vol. 2, no. 2, pp. 115-150, 2002.

[9] Z. Shelby, K. Hartke and C. Bormann, "The Constrained Application Protocol (CoAP)," Internet Engineering Task Force (IETF) , June 2014. [Online]. Available: https://tools.ietf.org/html/rfc7252. [Accessed June 2016].

[10] C. Pautasso, O. Zimmermann and F. Leymann, "RESTful Web Services vs. "big" web services: making the Right Architectural Decision," in Proceedings of the 17th International World Wide Web Conference, Beijing, 2008.

★★★