International Journal of Applied Engineering Research ISSN 0973-4562 Volume 12, Number 23 (2017) pp. 13314-13320

Comparative Analysis of Cost Modeling on Lightweight Messaging Protocols for IoT Integrated Ubiquitous Collaboration System

Minkyung Kim[1], Jai-Hoon Kim[2] and Kangseok Kim[2, 3]*

[1]*Department of Computer Engineering, Graduate School of Ajou University, 16499, Suwon, Korea.*

[2]*Department of Cyber Security, Ajou University, 16499, Suwon, Korea.*

[3]*Department of Data Science, Ajou University, 16499, Suwon, Korea.*

**Corresponding author*

[1]*Orcid: 0000-0002-8627-2482,* [2]*Orcid: 0000-0001-9738-6197,* [2, 3]**Orcid: 0000-0001-8950-7577*

Abstract

IoT is driving the evolution of ubiquitous collaboration to work together through data or activities generated from interactions among people, machines, and things. To integrate diverse smart things into a globally virtual collaboration system will become increasingly important. We address this for universal IoT integrated collaboration and access — the capability to link and access together with disparate accessible smart things on Internet and to reflect data generated from the things into collaboration system. This paper extends our prior work with Internet connected interactive IoT devices for ubiquitous collaboration. As a first step, we describe the cost modeling (mainly focusing on lightweight messaging protocols based on IoT) and comparative analysis results on the cost modeling to identify the influence of different network conditions according to the characteristics of the IoT application services. We also describe lessons learned and discuss future works.

Keyword: Internet of Things; Ubiquitous Collaboration; Cost Modeling; Lightweight Messaging Protocols

INTRODUCTION

An IoT (Internet of Things) is usually defined as things of physical smart objects or devices connected on the network [1-3]. The IoT technology is driving the evolution of ubiquitous collaboration with increasing demands in terms of the capability to collect and exchange new data or activities generated from interactions among people, machines, and things in a synchronous or asynchronous time. In IoT technologies, there are a lot of challenging issues such as consistency, heterogeneity, scalability, reliability, energy efficiency, semantic interoperability, data management, security, privacy, and so on [4-6]. As IoT environments are involved in collaboration, collaboration activities generated can be tracked among people, machines and things. Then the data collected from the activities of IoT can create collaboration services much richer than can be provided by previous isolated collaboration systems. In this scenario, collaboration services will coexist with the emerging IoT systems. We address this for universal IoT integrated collaboration and access — the capability to link and access together with the disparate accessible smart devices on Internet and to reflect data generated from the things to collaboration system. This paper extends our prior analysis models [7-9] with Internet connected interactive IoT devices for ubiquitous collaboration. As a first step, we introduce the cost modeling (mainly focusing on the performance comparison of lightweight messaging protocols) of universal IoT integrated collaboration system built on Cloud. Then we describe the comparative analysis results with the IoT specific light-weight messaging protocols for ubiquitous collaboration.

The cost modeling focuses on analyzing the basic network performance of request/reply (CoAP) and publish/subscribe (MQTT) models to know the influence of different network conditions according to the characteristics of the IoT application services. Fig. 1 shows an overview of IoT integrated collaboration and access system built on Cloud. Various heterogeneous information can be aggregated into IoT mashup framework by diverse lightweight messaging protocols such as request/reply (CoAP), publish/subscribe (MQTT) and so on. The CoAP (Constrained Application Protocol) [10] is a common protocol for request/reply communication. Also, IoT technologies employs MQTT (Message Queuing Telemetry Transport) [11] based on topic-based publish/subscribe method. The efficient integration of multiple IoT functions and resources into the framework makes it possible to create effective capabilities and services.

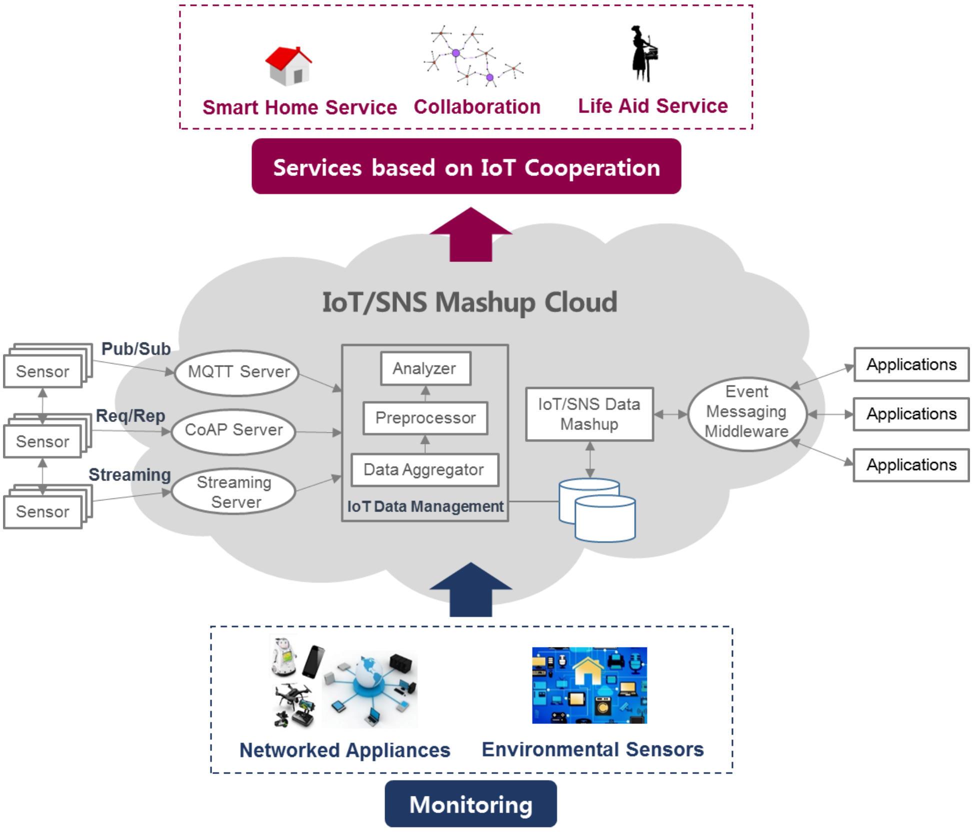

Figure 1: An overview of IoT integrated collaboration and access system built on Cloud

IoT environments require energy-efficient communication protocols for exchanging information. The using of lightweight application protocols like CoAP and MQTT etc improves the overall performance of IoT applications such as battery power consumption, bandwidth usage and communication latency. Therefore, the efficient processing of connectivity and mobility is crucial to offer reliable IoT services. In this paper, we analyze the network performance of the widely adopted communication models such as distributed request/reply (CoAP) and publish/subscribe (MQTT) in IoT involved collaboration environment. After identifying the overall characteristics of network protocols, the appropriate network protocol would be selected depending on each application service.

Firstly, a request/reply model to communicate with each other is a common in traditional client-server architecture. The client sends a query to sensors, and the sensors process the query and sends the response back to the client. The Constrained Application Protocol (CoAP) [12-17] is based on a request/reply interaction model between endpoints, allowing REST based communications based on HTTP methods such as GET, POST, PUT and DELETE. It is a web application protocol for resource-constrained and high performance devices. Therefore the CoAP could provide a lightweight reliability mechanisms for supporting tiny, low message overhead, low power embedded sensors in IoT systems. In CoAP, once the information is read, it is possible to provide a certain service by using the cached value within a certain time. Therefore, it is an appropriate protocols for providing status notification services based on status information such as empty spaces for parking and aircraft arrivals. The research of [14] proposes a system architecture for scalable IoT cloud services based on CoAP and shows that the low overhead of CoAP improves backend service scalability among a large number of low cost IoT devices.

Next, MQTT protocol [18-20] is a lightweight connectivity protocol based on a topic-based publish-subscribe messaging model. MQTT is designed to have a low message overhead and to use low battery power. So, it is suitable for resource-constraint devices such limited processor and memory resources in IoT environments. Basically, publish/subscribe model [7-9] is a data centric communications used to disseminate data, share and discovery services, and so on. A publisher publishes topic-based information asynchronously to the broker. Then, the broker disseminates specific topic-based information which registered interests on the broker to subscribers. Thus, it is useful to provide appropriate services according to the change of state because the broker automatically pushes the desired topic-based information without polling continuously. In aspects of IoT services, MQTT protocol is suitable to provide application services based on real-time event-based data communication such as automatic indoor temperature and humidity control and plant management. For example, an indoor sensor of smart-home publishes the topic-based data about temperature and humidity to the MQTT broker. And the broker pushes the topic-based data to an indoor temperature and humidity control application installed on the smartphone. And a user's application subscribes the published topic-based data. Finally, the states of indoor temperature and humidity can be automatically controlled through communicating with small single-board computers and microcontrollers. An interconnection solution for Cloud federations based on publish/subscribe services was proposed to support intercommunication among Cloud providers [21]. Also, there is a project about publish-subscribe internet routing paradigm to offer desired internetworking services [22]. In addition, the performance comparison of light weight protocols considering diverse network conditions has studied. The qualitative and quantitative comparison between CoAP and MQTT for smartphone-based sensing are proposed in [23]. Their experimental results show that MQTT is more appropriate for the advanced functionalities like persistent message transmission. On the other hand, CoAP is a more proper choice for reducing bandwidth and resource usage. Also, there is a research that the performance evaluation of MQTT and CoAP is conducted [11]. From the experimental results, they show that CoAP is more reliable for message transmission in condition of transferring small size of messages with the lower packet loss rate. In this paper, we compare the most commonly used protocol such as CoAP and MQTT in constrained IoT environments. Based on our analysis, the appropriate protocol can be adopted for certain application services.

The remainder of this paper is organized as follows. Section 2 briefly describes the overall architecture of our IoT integrated collaboration and access system built on Cloud. Section 3 presents cost modeling on IoT network protocols used in the collaboration system. Also, the parameter analysis for the cost modeling based on request/reply (CoAP) and publish/subscribe (MQTT) protocols with disparate accessible smart devices on Internet is presented. Finally we conclude with a brief discussion of future work in Section 4.

A Broad Architecture View for IoT Integrated Collaboration System

In this section we briefly describe an architecture for IoT integrated ubiquitous collaboration system built on Cloud computing environment that handles cooperation and communication among people, machines and things. A key function is to provide a generic solution for accessing and controlling IoT, aggregating various IoT data, and maximizing the use of various collaborative capabilities to collaborator. Another function is to provide a structure for development and deployment of various data integration (mashup) that supports management of heterogeneous information. How to share and integrate diverse data effectively and efficiently has become one of the key concerns of mashup. In our work, for communication service, we have used request/reply (CoAP) and publish/subscribe (MQTT) protocols for messaging middleware system built with heterogeneous networks to support group communications among people, devices, sensors and collaborative applications. With such an approach using open management system, we can build a sustainable high functionality system taking advantage of the latest system technologies with appropriate interface in a modular fashion.

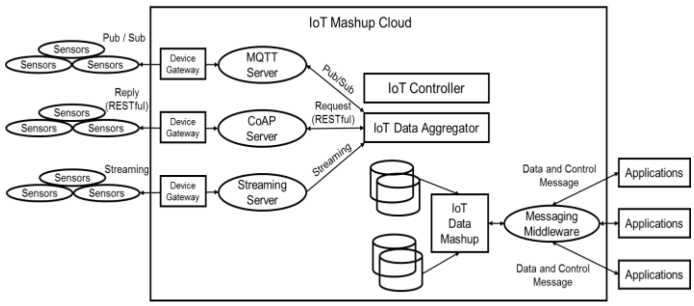

Figure 2: An overall architecture view of IoT integrated ubiquitous collaboration system

Fig. 2 shows a broad architecture view for our IoT integrated collaboration system. The system is structured as three layers and five major components. The three layers are composed of IoT layer, servers, database, and framework layer on Cloud, and application layer. It is composed of five major components – IoT data aggregator to collect and aggregate IoT data respectively, IoT controller to control IoT data mashup framework. The mashup framework in the figure is a basic structure to integrate collaboration data and activities, large-scale web contents and available sensing information from multiple sources. It also has a capability to maintain shared state consistency among things. It is important to users joining a collaboration that it seems to be in the same workspace by ensuring consistency state even when using heterogeneous computing devices at remote locations. In pull based IoT data dissemination approach (CoAP), the data can be disseminated based on requests of users (request and response paradigm). In this approach, the key advantage is well suited to things with constrained resources in using CoAP message protocol. Therefore, it is an appropriate protocols for the status notification services in the distributed environments. In push based IoT data dissemination approach (MQTT), data collected from various sensors is disseminated to collaborators through the mashup framework and communication channel (a broker for publish / subscribe). Then, the data stored in the database is periodically updated by data collected from various sensors and by activities from collaborators. Thus this protocol is used for the application services that push real-time data to support persistent connections. In next section we will present the cost modeling related on lightweight messaging protocols used in the IoT integrated ubiquitous collaboration environment.

Cost Analysis Modeling on Lightweight Messaging Protocols

In this section, we propose a cost analysis model for network performance of lightweight messaging protocols such as publish/subscribe and request/reply used in our IoT involved collaboration environment. Based on previous analysis model [7-9], we extended cost analysis model to analyze network performance of lightweight messaging protocols involved in IoT integrated ubiquitous collaboration system. This cost comparison of network protocol is executed for providing appropriate IoT application services. We assume following basic system parameters to analyze the cost model as shown in Table 1. Basically, those parameters are governed by Poisson process with average inter-arrival time. The access request arrival times form a Poisson process since the inter-arrival times of the requests are independent random variables with exponential distribution with pre-known mean inter-arrival rate.

We analyze the cost of two different models: request/reply (CoAP) and publish/subscribe (MQTT) models depending on the communication methods of the application services as shown in Table 2. We categorize two types of applications. The App-1 denotes the application services of request/reply types. When a service request occurs, the application services are provided based on state information. Also, the App-2 denotes the application services of publish/subscribe types. It is used to provide real-time data based application services for respond to changes. Then we consider the communication cost in both cases of per unit time and per access. And the delay time per access based on usage of each model according to the communication methods of application is supposed. Firstly, we analyze the communication cost per unit time for each model in Table 2. When using the application type according to service requests, the communication cost of request/reply models is calculated as βC_{rr} since it is considered the access rate during a certain unit of time. And the communication cost for each access is C_{rr}. Also, request/reply model's time delay for event access rate is supposed. Next, we suppose the communication cost of the usage of request/reply models when the application for respond to changes is used. This cost is affected by the polling interval time. And the communication cost per access is $\frac{1}{\alpha T_{poll}}$. The polling interval time is divided by the average number of the event generation per unit time to calculate the communication cost for each access. Also, the delay time per access would take half of the polling interval time.

Secondly, when publish/subscribe models are used for the application services according to service requests, we suppose that the communication cost per unit time is αC_{ps} because the communication cost is supposed whenever the event is generated. When Publish / subscribe models is used for IoT services, the communication costs per unit time for both types of application about App-1 and App-2 are the same. Also, $\frac{\alpha}{\beta}$ is the average number of events occurred before each access. Then, the communication cost for each access is $\frac{\alpha C_{ps}}{\beta}$ when the application according to service requests is used. And C_{ps} is the communication cost per access of publish/subscribe models when the application for respond to changes. To provide the application services according to service requests, there is no time delay for access after the subscriber's intention because events have already been delivered to a broker. However, the delay time is required when using the types of application responding to the changes.

Table 1: Parameter definitions

Parameters	Definitions
α	event generation rate
β	client's access rate
C_{ps}	publish and subscribe communication cost per event
C_{rr}	request and reply communication cost per generated event
T_{poll}	polling interval time according to event occurrence
t_{ps}	time delay for publish and subscribe communication
t_{rr}	time delay for request and reply communication

Table 2: Cost of selected models

Models		App-1 (Req/Rep type) : according to service requests	App-2 (Pub/Sub type) : response to changes
Usage of Request / Reply (CoAP) Model	communication cost per unit time	βC_{rr}	$\frac{C_{rr}}{T_{poll}}$
	communication cost per access	C_{rr}	$\frac{1}{\alpha T_{poll}}$
	delay time per access	t_{rr}	$\frac{T_{poll}}{2}$
Usage of Publish / Subscribe (MQTT) Model	communication cost per unit time	αC_{ps}	αC_{ps}
	communication cost per access	$\frac{\alpha C_{ps}}{\beta}$	C_{ps}
	delay time per access	0	t_{ps}

International Journal of Applied Engineering Research ISSN 0973-4562 Volume 12, Number 23 (2017) pp. 13314-13320
© Research India Publications. http://www.ripublication.com

Finally, we describe comparative analysis on cost modeling of lightweight messaging protocols with IoT integrated ubiquitous collaboration architecture. Table 3 shows system parameters defined by referencing our previous cost analysis [7-9]. Fig. 3 denotes the graph of the communication cost in the usage of request/reply (CoAP) models when β is a variable in the experiment. The graph of Fig. 3 (a) show that the communication cost per unit time of App-1 is lower than that of App-2 because App-1 needs communication cost only when a service request comes. In the next graph of Fig. 3 (b), the communication cost per access of App-1 is better than that of App-2 as before. And the communication cost per access of App-2 is affected by the polling interval time. Therefore, when the polling interval time is larger, the communication cost of App-2 would be low. Fig. 4 denotes the graph of the communication cost per access in the usage of publish/subscribe (MQTT) models when α is a variable in the experiment. The communication cost per access of App-2 is lower than that of App-1 as long as the values of α and β do not equal. In the Fig. 4, we only present the graph of the communication cost per access because both App-1 and App-2's the communication cost per unit time are the same. Although the communication cost per unit time is the same in both App-1 and App-2, the communication cost of App-2 would be better if the event generation rate is lower and the access rate is higher. Our cost analysis result would be used to provide appropriate network configurations for various application services in IoT environments. There are a lot of tradeoff among those approaches to disseminate various IoT data to collaborators in synchronous or asynchronous time for IoT integrated ubiquitous collaboration. In future work we will investigate more issues such as consistency, interoperability, scalability, security and so on with the properties of the analyzed networks and the consideration of energy efficiency of mobile smart things as well.

Table 3: Parameters for analysis

Parameters	α	β	C_{ps}	C_{rr}	T_{poll}	t_{ps}	t_{rr}
Values	0.1 ~0.4	0.1 ~0.4	1	1	0.1 ~0.4	1	1

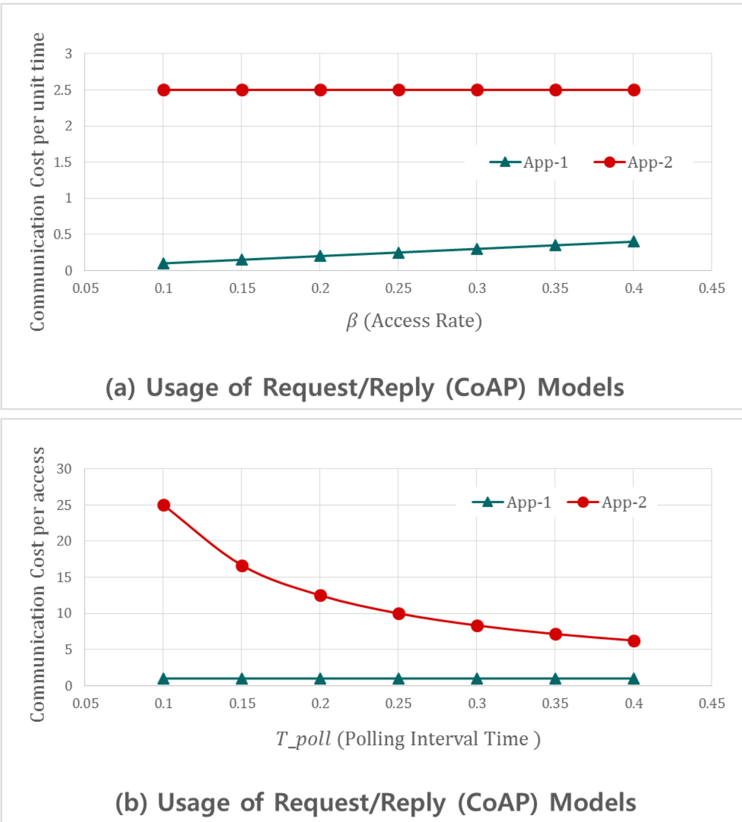

Figure 3: (**a**) communication cost per unit time in the usage of request / reply (CoAP) model
(**b**) communication cost per access in the usage of request / reply (CoAP) model

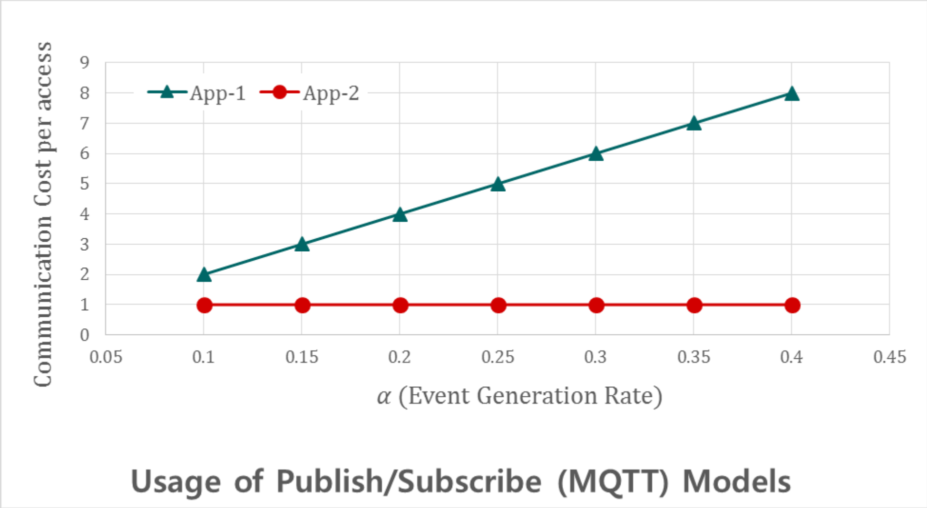

Figure 4: Communication cost per access in the usage of publish / subscribe (MQTT) model

CONCLUSIONS AND FUTURE WORK

In this paper we presented cost analysis modeling for the lightweight IoT messaging protocols and performance comparisons by parameter analysis on the modeling in IoT integrated collaboration and access system built on Cloud. Also we presented the integration architecture of IoT into ubiquitous collaboration. In our first research phase, the initial efforts are aimed at building IoT integrated collaboration system with mashup framework. Various heterogeneous information can be aggregated into the framework through diverse lightweight messaging protocols such as publish/subscribe (push based MQTT approach), request/reply (pull based CoAP approach), and streaming message fashion. Therefore in our future research we expect the infrastructure improvements of software, hardware, and networking will make IoT involved ubiquitous collaboration more prevalent. This paper extended our prior work with Internet connected interactive IoT devices for synchronous ubiquitous collaboration. In future work we will consider the consistency issue based on the results of the modeling, analysis and experiments which we had in developing universal mashup of data generated from IoT and collaboration activities. Additionally, we will have experiments with lightweight messaging protocols built in IoT involved collaboration environment to show the viability of cost analysis modeling. The main purpose of the experiment is to identify key factors that influence various IoT networks comparing latency incurred from request/response communication of lightweight messaging protocols. Also we will consider data security and personal privacy preservation issues with our IoT integrated collaboration and access system built on Cloud. Moreover, we plan to extend our research with LPWA (Low Power Wide Area) environments, focusing on reliable and energy efficient method for supporting mobile smart things-based services.

ACKNOWLEDGMENTS

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (No. NRF-2015R1D1A1A01060236 and No. NRF-2015R1D1A1A01060034).

REFERENCES

[1] L. Atzori, A. Iera, G. Morabito, "The Internet of Things: A survey", Computer Networks, Vol. 54, pp. 2787-2805, 2010.

[2] J. Gubbi, R. Buyya, S. Marusic, M. Palaniswami, "Internet of Things (IoT): A vision, architectural elements, and future directions", Future Generation Computer Systems, Vol. 29, pp. 1645-1660, 2013.

[3] Borgia, E. "The Internet of Things vision: Key features, applications and open issues", Computer Communications, Vol. 54, pp. 1-31, 2014.

[4] C. Sarkar, S. N. A. U. Nambi. V. Prasad, A. Rahim, R. Neisse, G. Baldini, "DIAT: A scalable distributed architecture for IoT", IEEE Internet of Things Journal, Vol. 2, No. 3, pp. 230-239, 2015.

[5] D. Miorandia, S. Sicarib, F. De. Pellegrinia, I. Chlamtac, "Internet of things: Vision, applications and research challenges", Ad Hoc Networks, Vol. 10, No. 7. pp. 1497-1516, 2012.

[6] J. Gubbi, R. Buyyab, S. Marusica, M. Palaniswamia, "Internet of Things (IoT): A vision, architectural elements, and future directions", Future Generation Computer Systems, Vol. 29, No. 7, pp. 1645-1660, 2013.

[7] S. Oh, S. L. Pallickara, S-H. Ko, J-H. Kim, G. Fox,

"Publish/Subscribe Systems on Node and Line Error Prone Mobile Environments", Lecture Notes in Computer Science (ICCS'05), Vol. 3515, pp. 576-584, 2005.

[8] S. Oh, S. L. Pallickara, S-H. Ko, J-H. Kim, G. Fox, "Cost Model and Adaptive Scheme for Publish/Subscribe Systems on Mobile Grid Environments", Lecture Notes in Computer Science (ICCS'05), Vol. 3516, pp. 275-278, 2005.

[9] S. Oh, J-H. Kim, G. Fox, "Real-time performance analysis for publish/subscribe systems", Future Generation Computer Systems, Vol. 26, No. 3, pp. 318-323, 2010.

[10] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, M. Ayyash, "Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications", IEEE Communications Surveys & Tutorials, Vol. 17, No. 4, pp. 2347-2376, 2015.

[11] D. Thangavel, X. Ma, A. Valera, H-X. Tan, C. K-Y. Tan, "Performance evaluation of MQTT and CoAP via a common middleware", Proceedings of IEEE 9th International Conference on Intelligent Sensors; Sensor Networks and Information Processing (ISSNIP), pp. 1-6, 2014.

[12] A.P. Castellani, M. Gheda, N. Bui, M. Rossi, M. Zorzi, "Web Services for the Internet of Things through CoAP and EXI", Proceedings of IEEE International Conference on Communications Workshops (ICC), pp. 1-6, 2011.

[13] T. A. Alghamdi, A. Lasebae, M. Aiash, "Security analysis of the constrained application protocol in the Internet of Things", Proceedings of IEEE International Conference on Future Generation Communication Technology (FGCT), pp. 163-168, 2013.

[14] M. Kovatsch, M. Lanter, Z. Shelby, "Californium: Scalable cloud services for the Internet of Things with CoAP", Proceedings of IEEE International Conference on the Internet of Things (IOT), pp. 1-6, 2014.

[15] C. Bormann, AP. Castellani, Z. Shelby, "CoAP: An application protocol for billions of tiny internet nodes", IEEE Internet Computing, Vol. 16, No. 2, pp. 62-67, 2012.

[16] B. C. Villaverde, D. Pesch, R. D. P. Alberola, S. Fedor, M. Boubekeur, "Constrained Application Protocol for Low Power Embedded Networks: A Survey", Proceedings of IEEE International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS), pp. 702-707, 2012.

[17] L. Rodrigues, J. Guerreiro, N. Correia, "RELOAD/CoAP architecture with resource aggregation/ disaggregation service", Proceedings of IEEE International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC), pp. 1-6, 2016.

[18] S. Wagle, "Semantic data extraction over MQTT for IoTcentric wireless sensor networks", Proceedings of IEEE International Conference on Internet of Things and Applications (IOTA), pp. 227-232, 2016.

[19] J. E. Luzuriaga, J. C. Cano, C. Calafate, P. Manzoni, M. Perez, P. Boronat, "Handling mobility in IoT applications using the MQTT protocol", Proceedings of IEEE International Conference on Internet Technologies and Applications (ITA), pp. 245-250, 2015.

[20] P. Fremantle, B. Aziz, J. Kopecký, P. Scott, "Federated Identity and Access Management for the Internet of Things", Proceedings of IEEE International Workshop on Secure Internet of Things (SIoT), pp. 10-17, 2014.

[21] C. Esposito, M. Ficco, F. Palmieri, A. Castiglione, "Interconnecting federated clouds by using publish-subscribe service", Cluster Computing, Vol. 16, No. 4, pp. 887-903, 2013.

[22] N. Fotiou, D. Trossen, G. C. Polyzos, "Illustrating a publish-subscribe Internet architecture", Journal of Telecommunication Systems, Vol. 51, No. 4, pp. 233-245, 2012.

[23] N. De Caro, W. Colitti, K. Steenhaut, G. Mangino, G. Reali, "Comparison of two lightweight protocols for smartphone-based sensing", Proceedings of IEEE 20th Symposium on Communications and Vehicular Technology in the Benelux (SCVT), pp. 1-6, 2013.