

Design and Implementation of Sensor Middleware Based on IETF CoAP Protocol for Multi-Sensor Networks

Songai Xuan¹ and Do-Hyeun Kim^{2*}

^{1,2}*Department of Computer Engineering, Jeju National University, South Korea*
¹*xuansongai@foxmail.com, ²kimdh@jejunu.ac.kr*

Abstract

Recently, the development of middleware is very useful in sensor networks, especially between sensor devices and sensor middleware. In this paper, we propose a hierarchical real-time context aggregation architecture for heterogeneous sensor networks. Also, we design and implement the sensor middleware based on IETF CoAP protocol for multi sensor networks. Sensor middleware in IoT system is able to get sensing data from the sensors and send it to the sensor service provider to save into database. Through this middleware, users are able to get environment huge context data in multi sensor networks.

Keywords: *Wireless Sensor Network; Sensors; sensor middleware; IoT Device*

1. Introduction

Internet of Things (IoT), a new paradigm, which is rapidly gaining momentum in the scenario of modern wireless telecommunications. The basic idea of this concept is the ubiquitous presence around us on a variety of things or objects - such as radio frequency identification tags (RFID), sensors, actuators, mobile phones, *etc.*, - which, by using unique addressing scheme capable of interacting with each other and cooperate with its neighbors to achieve the overall goals [1] [2]. IOP as an inexpensive mass deployment trillion wireless Internet Protocol (IP) sensor assembly for the detection and monitoring of each object or the things around us. A pile of a variety of sensitive people to bring a new perspective to the way we interact with the environment around us, on several levels, including social, environmental, cultural, business, *etc.*, [3].

Sensor networks are a distributed system of sensor nodes that are interconnected. Sensors gather information about the physical world, *e.g.*, the environment or physical systems. In many situations, sensor nodes are stationary whereas actuator nodes are mobile. Sensor nodes are usually low-cost, low-powered, small devices equipped with limited sensing, data processing and wireless communication capabilities, while actuator nodes typically have stronger computation and communication powers and more energy budget that allows longer battery life [4].

Smart home makes people's life more convenient and interesting, it is necessary to develop smart home systems. This paper [5] presents an approach to provide reliable security for the deployment of Internet of Things (IoT) for a smart home system, with due regard for the convenience of the users, when working with the system. Another paper [6] introduce a smart home system installed in a subject's home, sensing data can be collected in a real environment. Machine learning has been used to analyze the data generated by the sensor nodes in the overall environment. According to these papers [7-9], wireless sensor networks are indispensable in smart homes, and it is also necessary to develop the middleware between sensors and the smart home system. This paper [10] propose a

Received (January 3, 2018), Review Result (March 5, 2018), Accepted (March 12, 2018)

* Corresponding Author

Service-Oriented Context-Aware Middleware (SOCAM) architecture for building and rapid prototyping of context-sensitive services. It provides effective support for acquiring, discovering, interpreting, and accessing various contexts for creating context-sensitive services.

Middleware supports the complexity and heterogeneity of the underlying hardware and network platforms, and application. Sensor network middleware provides the sensing-based pervasive computing applications using a sink node and the sensor nodes. [11-13].

Recently, a lot of sensor networks are increased for huge context data aggregation. And, we need a new architecture to combine multiple sensor networks. In this paper, we propose hierarchical context aggregation architecture for multiple sensor networks. And we design and implement of sensor middleware based on sensor devices using CoAP protocol for multiple sensor networks. Figure 1 shows the proposed conceptual architecture for multi sensor networks. The architecture consists of application client, sensor service provider, sensor middleware and multi sensor network, sensor middleware. We use to communicate sensor networks, other modules communicate via RESTful API. Application client show the results to the users via services supported by other modules. Sensor middleware saves sensing data into database via service provided by Sensor Service Provider. Sensors collect sensing data and send them to sensor middleware.

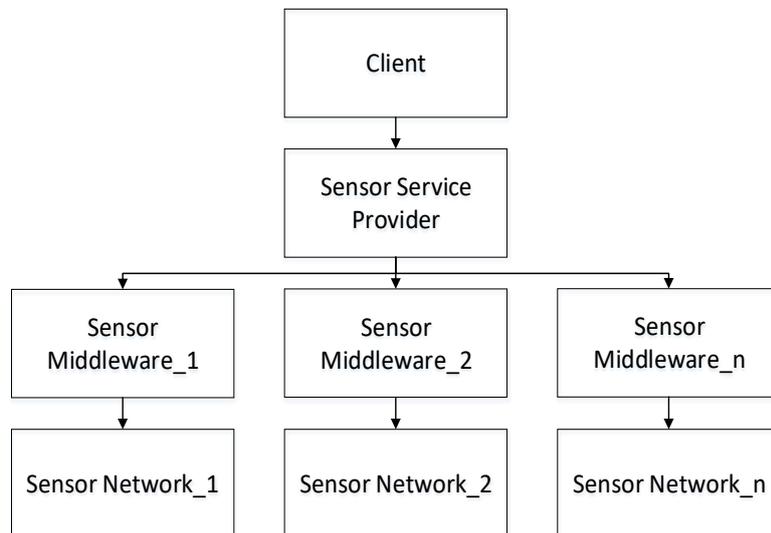


Figure 1. Conceptual Hierarchical Context Aggregation Architecture for Multi Sensor Networks

In this paper, we implement proposed sensor middleware based on IETF CoAP protocol to use Edison board as IoT devices, sensors will connect with Edison boards, and Edison boards will communicate with sensor middleware.

The organization of the structure of the paper is carried as: Section 2 represents the related work, Section 3 explains the proposed sensor middleware for context aggregation in multi sensor networks. We illuminate implementation and results of sensor middleware based on IETF CoAP protocol in Section 4 and Section 5 represents conclusion.

2. Related Work

A sensor middleware interpose between the sensor networks and the applications. Sensor middleware supports the message transmissions for context aggregation of the environment or physical environments. Until now, a lot of middleware for sensor network has been developed. A lot of Middleware for sensor network has been classified

depending on their approaches, which can be database oriented, tuple space approaches, event based and service discovery based approaches.

During the implementation, we have used RESTful API and CoAP protocol for the communication between the modules. REST (Representational State Transfer) is the architectural style that is commonly used to develop Web services. REST is a popular architectural style based on the cloud API. RESTful API means using RESTful Web services. The REST pattern involves specifying a page that contains an XML file that describes the content and contains the required content. REST usually runs through HTTP (Hypertext Transfer Protocol), which is typically used for mobile applications, social networking sites, mashup tools, and automated business processes. REST uses a limited number of operations (GET, POST, PUT, and DELETE) to enhance the interaction between the client and the service. And it is flexible because of assigning resources their own URIs (Universal Resource Identifiers) [14].

REST (State State State Transfer) is an architectural style that is often used when developing Web services. REST is a popular building style for cloud APIs. The RESTful API means that Web services use the REST architecture. The REST architecture includes reading a specific web page that contains an XML file that describes and includes the required content. REST usually works over the HTTP (Hypertext Transfer Protocol) protocol and is often used in mobile applications, mashup tools, social networking sites and automated business processes. REST uses a limited number of operations (GET, POST, PUT, and DELETE) to improve the interaction between clients and services. And it's flexible, because it assigns resources to its own URI (Universal Resource Identifiers) [14].

IETF CoAP (Constrained Application Protocol) is a software protocol designed for simple electronic devices that allow them to interact interactively over the Internet. It is designed for small, low power sensors, switches, valves and similar components, need to be remotely monitored or monitored via standard Internet. CoAP is an application-level protocol designed for resource-based Internet devices such as WSN nodes. CoAP is designed to easily translate HTTP, easily integrate with the network, and meet specific requirements such as multicast support, very low overhead and simplicity. Multicast, low overhead, and simplicity are important for the Internet, which is typically embedded in equipment and machine-to-machine (M2M) devices, and has less storage and power than traditional Internet devices [15].

We used Edison boards [16] as IoT devices during the implementation. And about the development of sensor middleware, we also referenced from other systems. The paper [14] proposed the use of custom components that can be used simultaneously in several compositions and allow to be changed the quality properties of the data in each composition, and in this way, adding new components or new clients does not affect existing dependencies, and users are able to get an efficient and flexible parametrization.

3. Proposed Sensor Middleware for Context Aggregation in Multi Sensor Networks

Figure 2 shows the detailed architecture of the proposed system. Sensors connect to Edison boards, sensors also collect sensing data and send it to the Edison boards. Edison boards communicate with sensor middleware by CoAP protocol, sensor middleware and sensor service provider communicate by RESTful API in sensor service provider. Sensor service provider is able to provide sensor information and sensing data to other modules. This figure shows that sensor service provider provides sensing data and sensor

information to the client. Sensor middleware saves sensing data into database via service by sensor service provider.

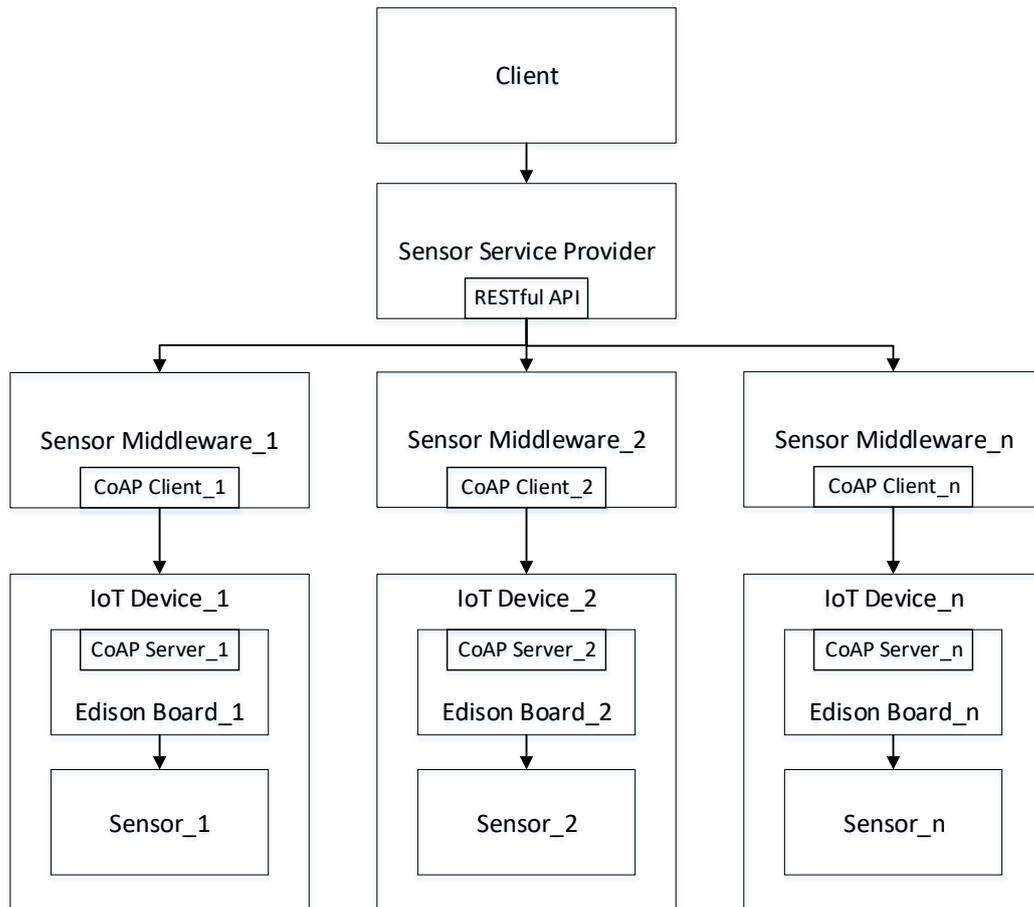


Figure 2. Implementation Model of Sensor Middleware for Multi Sensor Networks

Figure 3 shows the sequence diagram of saving sensing data. The CoAP client in sensor middleware sends request to CoAP server in Edison board to get sensing data, CoAP server gets sensing data from sensor and then returns sensing data to CoAP client. CoAP client in sensor middleware sends sensing data to the sensor service provider and saves it into the database. Then sensor service provider is able to provider sensing data to other modules.

Figure 4 shows the configuration diagram of sensor middleware. Sensor middleware take a role in collecting sensing data, which get from the sensors via Edison Board by CoAP protocol. Sensor middleware is also able to save the sensing data into database by the service provided by Sensor Service Provider. Configurator request sensor information (ID, Type) from Sensor Service Provider, and also verify pertinent IP address and service provider access privileges. Sensing Driver takes various sensors' sensing data format information and parse processing through received sensing data. CoAP Client is able to request sensing data from CoAP Server in Edison Board and get the sensing data. Sensing Data Receiver accesses sensing data sent from CoAP Client and saves at memory through sensing data Parser. Sensing Data Transporter read Memory-saved sensing data and send to Sensor Web provider.

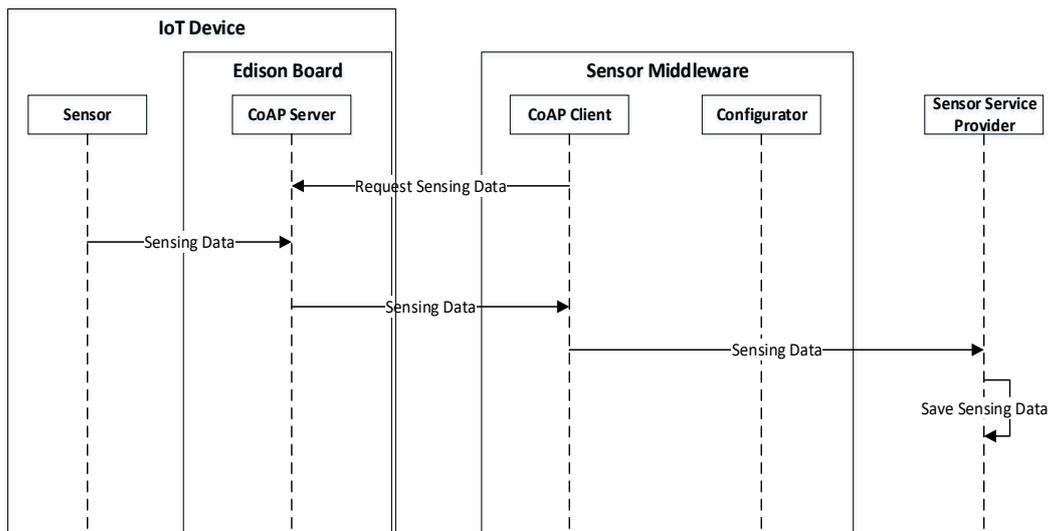


Figure 3. Sequence Diagram of IoT Devices, Sensor Middleware, and Sensor Service Provider

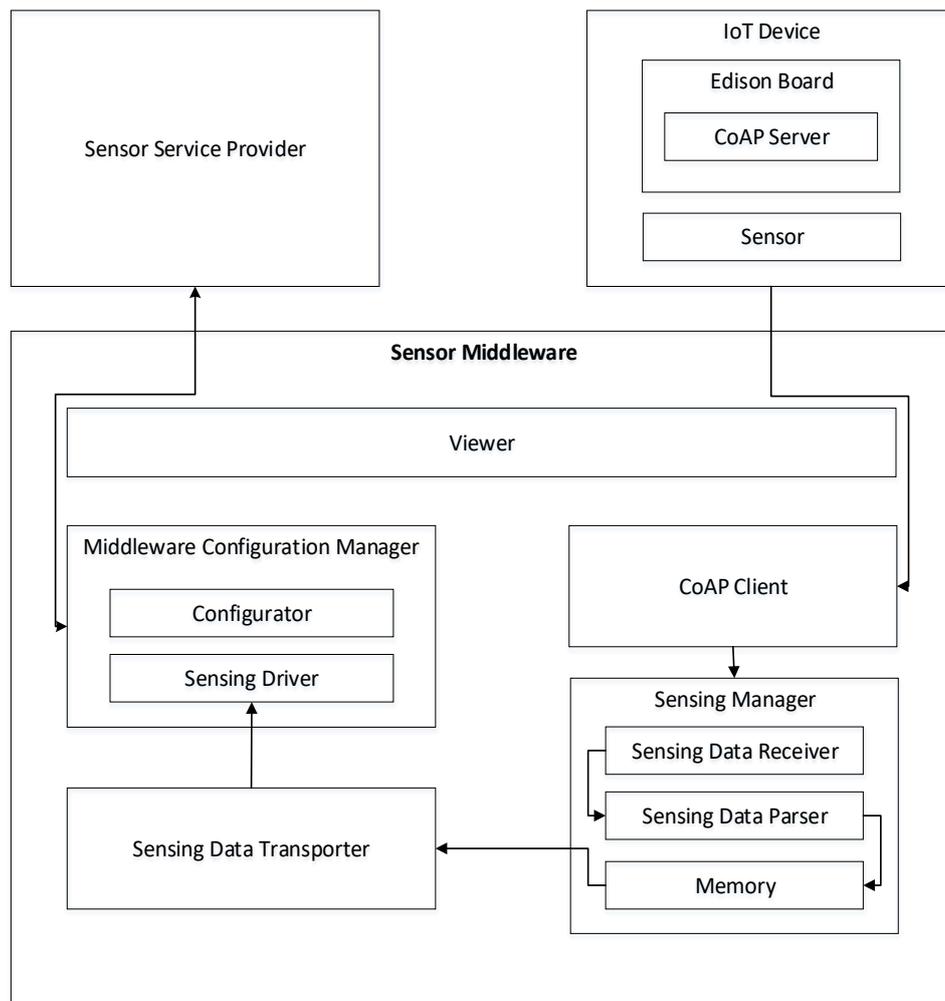


Figure 4. Configuration Diagram of Sensor Middleware

4. Implementation and Results of Sensor Middleware Based on IETF CoAP Protocol

The implementation environment of our middleware is described in Table 1. We used Windows 10 (X64) as the operating system, Visual Studio 2015 and Intel System Studio IoT Edition as development tool and used Microsoft SQL Server 2016 as database management system.

Table 1. Table Label

Components	Version
Operating System	Windows 10 – 64 bit
Microsoft Visual Studio	2015
Intel System Studio IoT Edison	4.5.2
Microsoft SQL Server	2016
CoAP.NET	1.1.0

Figure 5 shows the Edison board, we used the Intel Edison compute module and the Arduino expansion board. Figure 6 shows the temperature sensor. During the implementation, it is connected to Edison board.



Figure 5. Edison Board for IoT Devices

Figure 7 shows the implementation result of the sensor middleware. "Sensing Data" shows the sensor information and sensing data. Click "Edison Board" button to start the connection between sensor middleware and Edison Board and send sensing data to sensor service provider to save data into the database.



Figure 6. Temperature Sensor

Figure 8 shows the implementation result of CoAP Server. It shows the sensing data received from temperature sensor, the sensor is in a stable environment which the temperature is 54.66°C.

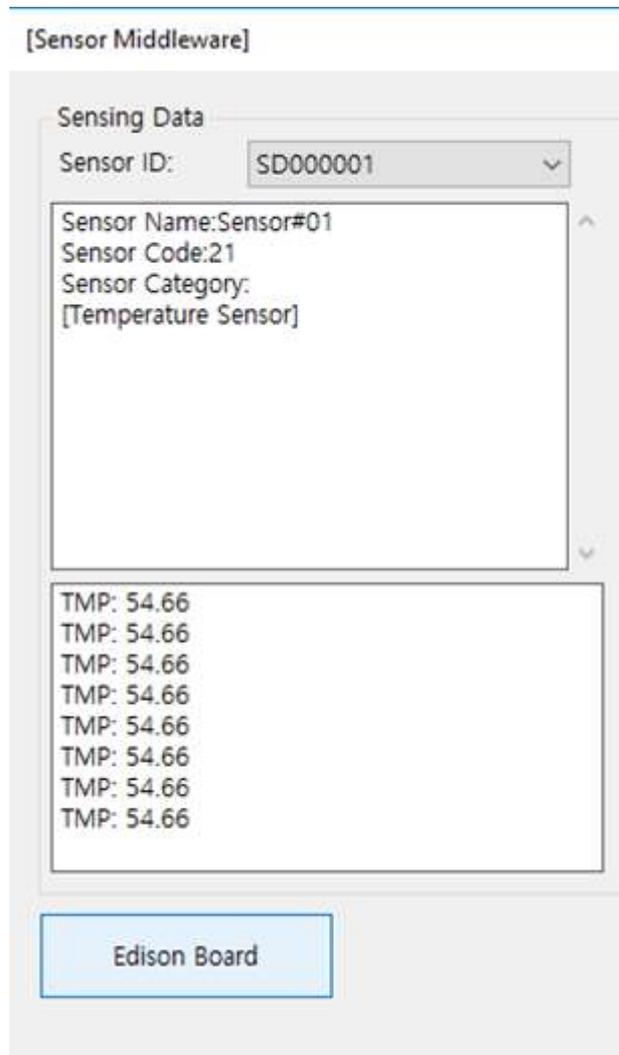


Figure 7. Implementation Result of Sensor Middleware

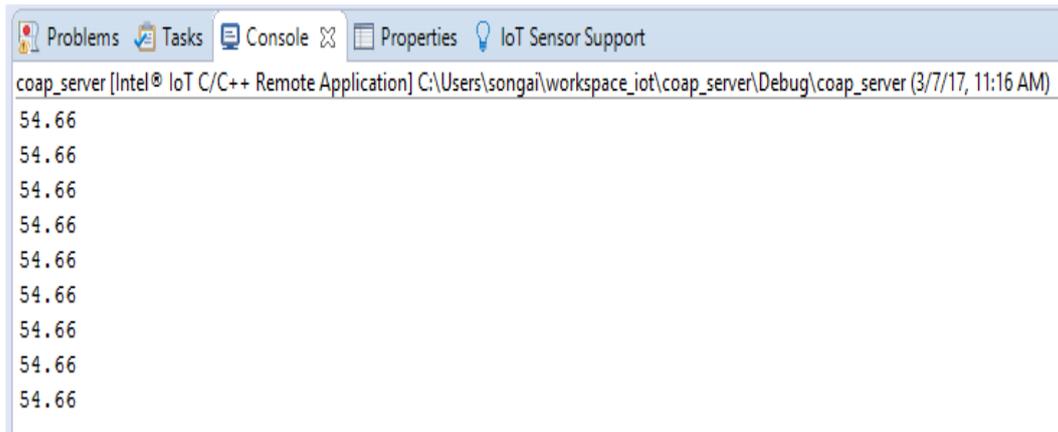


Figure 8. Implementation Result of CoAP Server

Figure 9 shows the saved temperature data in the database. This is an integrated environment database which is able to save the date (accurate to millisecond) of getting sensing data, the temperature data, the luminance value, and the humidity data. In this simulation test, we used the sensor which is only able to get temperature data, so the luminance value and the humidity value is 0.

	DATA_CODE	SENSING_TIME	TEMP_VALUE	LUX_VALUE	HUM_VALUE
7616	8268	2017-03-07 11:19:31.110	54.66	0	0
7617	8269	2017-03-07 11:19:32.130	54.66	0	0
7618	8270	2017-03-07 11:19:33.160	54.66	0	0
7619	8271	2017-03-07 11:19:34.270	54.66	0	0
7620	8272	2017-03-07 11:19:35.300	54.66	0	0
7621	8273	2017-03-07 11:19:36.330	54.66	0	0
7622	8274	2017-03-07 11:19:37.360	54.66	0	0
7623	8275	2017-03-07 11:19:38.380	54.66	0	0
7624	8276	2017-03-07 11:19:39.400	54.66	0	0
7625	8277	2017-03-07 11:19:40.430	54.66	0	0
7626	8278	2017-03-07 11:19:41.460	54.66	0	0
7627	8279	2017-03-07 11:19:42.490	54.66	0	0
7628	8280	2017-03-07 11:19:43.510	54.66	0	0
7629	8281	2017-03-07 11:19:44.610	54.66	0	0
7630	8282	2017-03-07 11:19:45.640	54.66	0	0
7631	8283	2017-03-07 11:19:46.670	54.66	0	0
7632	8284	2017-03-07 11:19:47.700	54.66	0	0
7633	8285	2017-03-07 11:19:48.730	54.66	0	0

Figure 9. Saved Data in Database

Figure 10 shows the data view of application client. A screen appears when a sensor node on the floor map is selected (right-clicked). The screen shows sensor node detailed information and sensing data. The temperature sensing data is shown in the data view.

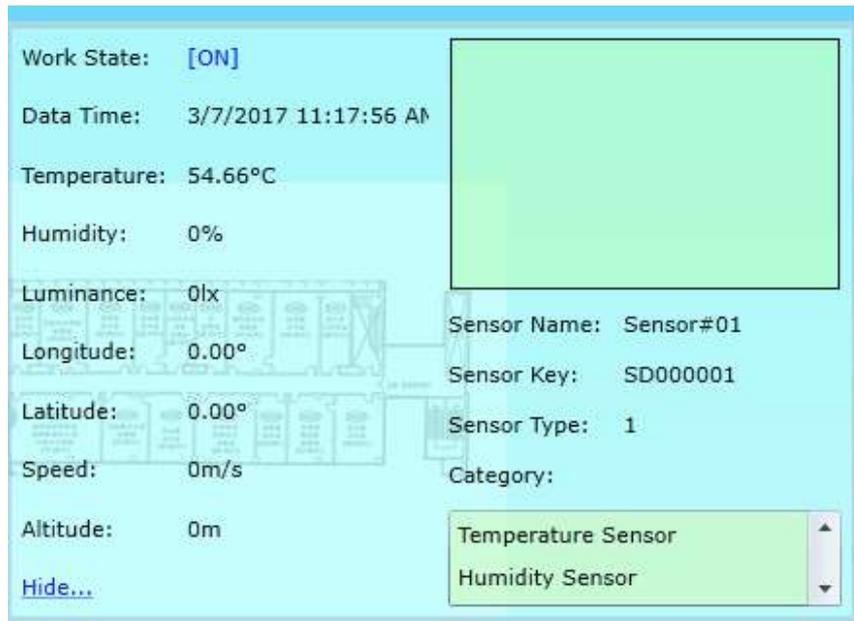


Figure 10. Result in Client

5. Conclusion

In this paper, we design and implement of a sensor middleware based on CoAP protocol for multi sensor networks. This sensor middleware enables users to get environment data from multiple devices. During this study, we developed a deep understanding about Ediosn Board and CoAP. In future, we will try to extend this system by implementing a detailed IoT system, through which users can control smart devices based on the environment data supplied by the wireless sensor network.

Acknowledgments

This research was supported by the MSIT(Ministry of Science and ICT), Korea, under the ITRC(Information Technology Research Center) support program(2014-1-00743) supervised by the IITP(Institute for Information & communications Technology Promotion), and this This work was supported by Institute for Information & communications Technology Promotion(IITP) grant funded by the Korea government(MSIT) (No.2017-0-00756, Development of interoperability and management technology of IoT system with heterogeneous ID mechanism). Any correspondence related to this paper should be addressed to DoHyeun Kim; kimdh@jejunu.ac.kr.

References

- [1] D. Giusto, "The internet of things: 20th Tyrrhenian workshop on digital communications", Springer Science & Business Media, (2010).
- [2] L. Atzori, A. Iera and G. Morabito, "The internet of things: A survey", Computer networks, vol. 54, no. 15, (2010), pp. 2787-2805.
- [3] A. E. Kouche, "Towards a wireless sensor network platform for the Internet of Things: Sprouts WSN platform", Communications (ICC), 2012 IEEE International Conference on IEEE, (2012).
- [4] G. Baliga, S. Graham and P. R. Kumar, "Middleware and Abstractions in the Convergence of Control with Communication and Computation", Proceedings of the 44th IEEE Conference on Decision and Control, and the European Control Conference 2005 Seville, Spain, (2005) December 12-15.
- [5] F. K. Santoso and N. CH Vun, "Securing IoT for smart home system", Consumer Electronics (ISCE), 2015 IEEE International Symposium on IEEE, (2015).
- [6] D. Basu, "Wireless sensor network based smart home: Sensor selection, deployment and monitoring", Sensors Applications Symposium (SAS), 2013 IEEE, (2013).

- [7] N. Xu, "A survey of sensor network applications", IEEE communications magazine, vol. 40, no. 8, (2002), pp. 102-114.
- [8] A. Maleki Tabar, A. Keshavarz and H. Aghajan, "Smart home care network using sensor fusion and distributed vision-based reasoning", Proceedings of the 4th ACM international workshop on Video surveillance and sensor networks. ACM, (2006).
- [9] M. Erol-Kantarci and H. T. Mouftah, "Wireless sensor networks for cost-efficient residential energy management in the smart grid", IEEE Transactions on Smart Grid, vol. 2, no. 2, (2011), pp. 314-325.
- [10] T. Gu, H. Keng Pung and D. Qing Zhang, "A service - oriented middleware for building context - aware services", Journal of Network and computer applications, vol. 28, no. 1, (2005), pp. 1-18.
- [11] S. Zarghami, "Middleware for Internet of Things", University of Twente, (2013).
- [12] J. Schneider, A. Klein, C. Mannweiler and H. D. Schotten, "An efficient architecture for the integration of sensor and actuator networks into the future internet", Adv. Radio Sci., vol. 9, (2011), pp. 231-235.
- [13] T. Melodia, D. Pompili, V. C. Gungor and I. F. Akyildiz, "Communication and Coordination in Wireless Sensor and Actor Networks", IEEE Transactions on Mobile Computing, vol. 6, no. 10, (2007), pp. 1116-1129.
- [14] Representational state transfer – Wikipedia: https://en.wikipedia.org/wiki/Representational_state_transfer.
- [15] CoAP - Constrained Application Protocol: <http://coap.technology/>.
- [16] Overview of the Intel® Edison Module | Intel® Software: <https://software.intel.com/en-us/articles/what-is-the-intel-edison-module>.

Authors



Do-Hyeun Kim, he received the B.S. degree in electronics engineering from the Kyungpook National University, Korea, in 1988, and the M.S. and Ph.D. degrees in information telecommunication the Kyungpook National University, Korea, in 1990 and 2000, respectively. He joined the Agency of Defense Development (ADD), from March 1990 to April 1995. Since 2004, he has been with the Jeju National University, Korea, where he is currently a Professor of Department of Computer Engineering. From 2008 to 2009, he has been at the Queensland University of Technology, Australia, as a visiting researcher. His research interests include sensor networks, M2M/IOT, energy optimization and prediction, intelligent service, and mobile computing.



Songai Xuan, is currently perusing M.S. in Department of Computer Engineering, Jeju National University, Republic of Korea. She received her B.S. degree in Computer Science from YanBian University, China in 2016.