International Journal of Advanced Culture Technology Vol.7 No.4 96-103 (2019)
DOI 10.17703/IJACT.2019.7.4.96

IJACT 19-12-12

Two-Stage Multichannel Architecture for Oyster Product Management System

Yeong-Yil Yang

Professor, Department of Semiconductor Engineering
GyeongSang National University, JinJu Dae Ro 501, Jinju city, GyeongNam, 52828 Korea
yyang@gnu.ac.kr

Abstract

In this paper, we propose two-stage multichannel architecture for oyster product management system, called cloud stage and agent stage. There are two communication channels at each stage. In cloud stage, the embedded system in the smart scale communicates with the server through two channels, Ethernet or 3G/LTE mobile communication. In agent stage, PCs and smart phones called agents communicate with the server also through Internet and 3G/LTE mobile communication. Compared with previous system in which the amount of the oyster produced in oyster workplaces could be monitored only at the console of only one oyster main server, developed system makes it possible to monitor the amount of produced oyster at several PCs (or smart phones). In addition to the amount of oysters produced at all oyster workplaces the environment of oyster workplaces such as temperature and humidity can be monitored on agents to judge the freshness. Two-stage architecture with multiple channels makes it possible to monitor the amount of oyster product and environment of the oyster workplace at any place in real time.

Keywords: *3G/LTE, Embedded System, Cloud System, Oyster, Monitoring System, Smart Phone*

1. INTRODUCTION

Industry 4.0 in which computers and automation come together in an entirely new way is the current trends in the various application areas.[1] The idea of Industry 4.0 was the integration of Internet of Things (IoT) in the manufacturing process in the factory.[2] IoT becomes the smart component with various sensors, actuators and IoT devices are connected tightly to get the goal.[3-7] The amount of the oyster produced in Korea is 40,000 tons every year and 22,000 workers are engaged in oyster production. In the conventional oyster workplace, workers crushe oyster shells and take oyster shells away. The weight of the oyster produced by each worker is measured by the scale and it is written on the note by hand. The system in Ref. [8] totals up the amount of the oyster produced in all oyster workplaces in real time.

The problem of the system developed in Ref. [8] is that the amount of the oyster produced in oyster workplaces could be monitored only at the console of only one oyster main server which is in the oyster cooperative union. Therefore, the system which makes it possible to monitor the amount of oyster produced at several PCs (or smart phones) at the division of several city halls in real time is required. The diversification of communication channels is inevitable trend of the development for monitoring and remote control of remote systems, home appliances, robot control [10-14].

Manuscript received: October 21, 2019/ revised: November 05, 2019/Accepted: November 10, 2019
Corresponding Author: yyang@gnu.ac.kr
Tel:+82-55-772-1733, Fax: +82-55-772-1739
Prof., Department of Semiconductor Engineering, GyeongSang Nat'l Univ.

In this paper, we propose two-stage architecture for oyster product management system with multiple channels, called *cloud stage* and *agent stage*. There are two communication channels at each stage. In *cloud stage*, the oyster management system at each workplace communicates with the oyster main server at the control center through two channels, Ethernet or 3G/LTE mobile communication. In *agent stage*, PCs and smart phones called *agents* also communicate with oyster main server through Ethernet and 3G/LTE mobile communication. Two-stage architecture with multiple channels makes it possible to monitor and control the workplaces at any PCs and smart phone in real time. We developed the embedded system, the monitoring program running on the server and the monitoring program running on the client PC and the app running on the client smart phone, which access the monitoring program at the control center.

The architecture of two-stage multichannel system for oyster product monitoring will be explained in section 2. In section 3, the implementation of the proposed system is described. Finally, conclusions are given in Section 4.

2. ARCHITECTURE OF TWO-STAGE MULTICHANNEL SYSTEM

Fig. 1 shows the architecture of two-stage multichannel system for oyster product management. It consists of two stages, *cloud stage* and *agent stage*. Two-stage multichannel system can be classified into three parts, workplace system, management server and agents which are PCs or smart phones for monitoring the amount of oyster product and environment of the oyster workplace. The circle in Fig.1, S_1, S_2, ⋯, and S_n represents the embedded system which is included in the smart scale. The standalone system of *cloud stage* is developed in Ref. [8]. The embedded system in the smart scale acquires data from sensors such as the weight sensor, the temperature sensor and NFC card reader to get information of workers at the oyster workplace. The management program *OPMP*(*Oyster Product Management Program*) runs at the server which is in oyster cooperative union. If we look at the *agent stage,* there are two kinds of agents, *PC agent*s and *phone agent*s. We developed the monitoring program which runs on *PC agent* and the app which runs on smart phone called *phone agent*. The management program *OPMP* in the management server in Fig. 1 receives the commands from *agent*s and sends the commands to the system in *cloud stage*. When a change happens in *cloud stage*, the management server sends data to *agent*s to update the display information on *agent*s. When the problem is detected at the embedded system, the management server sends received signals form the embedded system to *agent*s in real time.

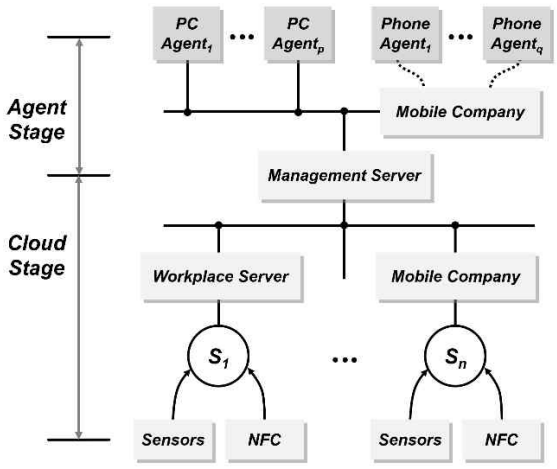

Figure 1. Architecture of two-stage multichannel system for oyster product management

Fig. 2 shows the dataflow of two-stage multichannel system. The data of the system is stored in the management server and the middleware in Fig. 2 is the gateway of the data. There are four communication ports in the middleware. Two ports are used for *cloud stage* and two ports are used for *agent stage*. In Fig. 2,

OPMP is the management program running on the management server in Fig. 1. In *cloud stage*, the management program *OPMP* sends and receives data with *WOPMP*(*Workplace Oyster Product Management Program*) running on the computer at the oyster workplace through internet and *cloud app* running on the smart phone in the oyster workplace through 3G/LTE mobile communication, respectively. The embedded system in the smart scale communicates with *WOPMP* through intranet (Wi-fi). Therefore, the embedded system can transmit the data to the management program *OPMP* through *WOPMP* and the middleware in Fig. 2. Also, the embedded system communicates with the smart phone in the workplace by Bluetooth. That is, the app running on that smart phone, *cloud app*, transmits data between the embedded system and the middle through Bluetooth and 3G/LTE mobile communication. The app server program in the middleware transmits the data between the management program *OPMP* and *cloud app*. In addition, *cloud app* provides functions which control the embedded system directly on the smart phone.

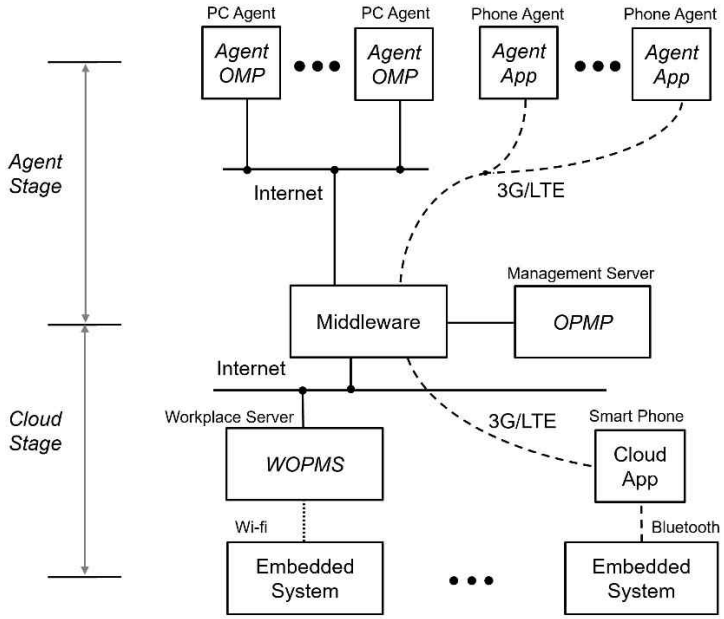

Figure 2. Dataflow of two-stage multichannel system

In *agent stage*, the management program *OPMP* communicates with *agent*s through internet and 3G/LTE mobile communication. The *PC agent* is connected to internet and communicates with the program *WOPMP* through the middleware. The monitoring program called *Agent OMP*(*Agent Oyster Monitoring Program*) running on the *PC agent*s sends commands and receives data through the middleware in Fig. 2. Also, the monitoring app called *Agent App* running on the smart phone communicates through 3G/LTE mobile communication which is represented with the dotted line between *phone agent* and the middleware in Fig. 2.

3. IMPLEMENTATION OF TWO-STAGE MULTICHANNEL MANAGEMENT SYSTEM

To develop two-stage multichannel system for oyster product management, we implemented the embedded system in the smart scale, management programs *OPMP*, *WOPMP* and monitoring program *Agent OMP*, *Agent App* and middleware which is the gateway for messages.

3.1 Middleware

Fig. 3 shows detailed block diagram of the middleware in Fig. 2. As shown in Fig. 3, there are four communication ports, two for workplace system and two for *agent*s. All the messages coming into the middleware are transferred to the management program *OPMP*. The management program *OPMP* does not

communicate with *agent*s and systems in *cloud stage* directly. The main functions of the middleware are as follows.

- Send received messages from *agent*s and systems in *cloud stage* to the management program *OPMP*.
- Send received messages from the management program *OPMP* to *agent*s and systems in *cloud stage*.

As shown in Fig. 3, the middleware consists of three blocks which are the message queue, the security module and the app server module. The message queue stores the messages received from cloud systems and *agent*s. The app server module receives messages coming from smart phone or sends messages to the smart phone. When the operator logs in the system at *PC agent*, the unique socket for that agent is generated. The data for that account is transferred through that socket. When the massage is received from *agent*s, the middleware checks the received message whether it's from authorized account. To verify the authorized account, the middleware checks *login id*, *password* and *identification number* which are included in the message. MAC address for *PC agent* and USIM number (or IEMI) for *phone agent* are used as *identification number*. If the message is unauthorized account, the middleware sends the error message to *agent*. For messages from the authorized account, the middleware inserts the received message to the message queue. The messages in the message queue are sent to the oyster product management program *OPMP* one by one. USIM number (or IEMI) for cloud phone also is used as *identification number*.

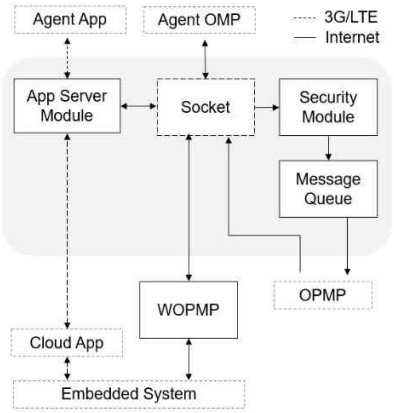

Figure 3. Detailed block diagram of the middleware

3.2 Embedded System in the smart scale

The embedded system in the smart scale acquires data from sensors such as the temperature sensor, the weight sensor, NFC card reader and so on. The main functions of the embedded system are as follows.

- Communicate with the management program *WOPMP* through Wi-fi and with the management program *OPMP* through Bluetooth.
- Acquire the data from the temperature sensor and the humidity sensor and send the acquired data.
- Measure the weight of the oyster on the smart scale and send it to the management program.
- Identify the worker who is measuring the weight of the oyster after reading the data recorded in NFC card.

Fig. 4 shows the block diagram of the embedded system we developed. As shown in Fig. 4, it consists of 4 modules.

- *Processing module*: Control the embedded system.
- *Sensor module*: Acquire the data from the sensors such as the weight sensor (load cell), the temperature sensor, the humidity sensor and read data from NFC card reader.
- *Communication module*: Send and receive data through two communication ports, intranet (Wi-fi) port or

Bluetooth port.

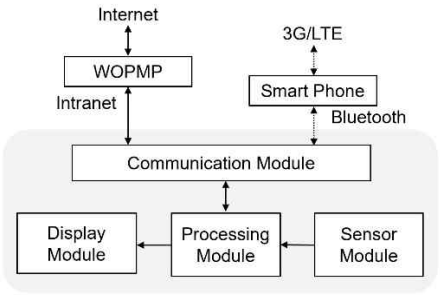

Figure 4. Block diagram of the embedded system

3.3 Workplace Output Management Program

The workplace oyster output management program, *WOPMP* running on the computer at the oyster workplace manages the output of the oyster produced at that workplace automatically in real time. The main functions of the program *WOPMP* are as follows.

- Communicate with embedded systems in the smart scale and the program *OPMP* through Wi-fi and internet.
- Total up the output produced at the oyster workplace.
- Handle the events coming from cloud system, the program *WOPMP* and *cloud app*.
- Enroll, modify or delete the information of the workers and smart scales of each workplace.
- Send the data of the oyster workplace, i.e. the information of workers, smart scales and the output of the oyster at that workplace to update the information stored on the server at the center.

3.4 Oyster Management Program

The oyster management program *OPMP* and the oyster monitoring program *Agent OMP* have similar user interface. The function for registering/unregistering *agent*s is active at the management program *OPMP* only.

The monitoring program *Agent OMP* runs at the registered PC. The operator can log in the system as system administrator account or the user account at every *PC agent*. Following functions become active in the system administrator account.

- Create/Delete the user accounts.
- Enroll, modify or delete the information of oyster workplaces.
- Register/Unregister workplaces to the specific accounts. The workplace can be registered to several accounts.

The oyster product management program, *OPMP* running on the computer at the center (oyster cooperative union) monitors the output of the oyster produced at all the oyster workplaces in real time. The main functions of the oyster output management program at the center are as follows.

- Communicate with the workplace management program *WOPMP* running on the computer at the workplace through the internet or the oyster output management app through the 3G/LTE mobile communication.
- Total up the output of the oyster produced at all the oyster workplaces.

As described in Ref. [8], Fig. 5 shows the main window of the oyster management programs, *OPMP* and *Agent OMP*. The icons which are listed on the upper side of the window provide various functions needed to set and show the information. The data received from workplaces is listed on the central part of the oyster output management program in Fig. 5. If the oyster workplace on the left side in Fig. 5 is clicked, the

information of the selected oyster workplace is displayed.

Figure 5. The main window of the programs, *OPMP* and *Agent OMP*

Fig. 6 shows the dialog box to enroll/modify oyster workplaces. The oyster workplaces are listed on the left side of the windows in Fig. 6. If one of oyster workplaces is selected, the information of the selected oyster workplace is displayed with the location of the selected oyster workplace on the map. We can enroll or modify the information of the selected oyster workplace using the dialog box shown in Fig. 6.

Figure 6. The Dialog box to enroll/modify oyster workplace

In addition to the functions described before, the management programs, *OPMP* and *Agent OMP* have many useful functions such as statistical information of each oyster worker and for each oyster workplace. The management programs, *OPMP* and *Agent OMP* are developed with Eclipse IDE for Java Developers [15-17]. The database is designed with SQL Lite.

3.5 Monitoring Program on Phone Agent

The monitoring program *Agent App* which is the app on the phone agent runs at the registered client smart phone. The monitoring program *Agent App* shows same information with the monitoring program *Agent OMP*.

3.6 Cloud App on Cloud Phone

The program *cloud App* which is the app on the cloud phone communicates with the smart scale through Bluetooth and the middleware as shown in Fig. 3. The main functions of the app running on the smart phone can be classified into two.

- Communication channel between the embedded system in the smart scale and the middleware.
- Providing the functions to monitor and set the parameter of the embedded system. The embedded system can be monitored and controlled through the app on *cloud phone*.

4. CONCLUSION

We implemented two-stage multichannel architecture for oyster output management system, *cloud stage* or *agent stage*. The proposed oyster output management system based on IoT(Internet of Things) uses two communication channels, the internet or the 3G/LTE mobile communication at each stage. In *cloud stage*, the

oyster management system at each workplace communicates with the oyster main server at the control center through two channels, Ethernet or 3G/LTE mobile communication. In *agent stage*, PCs and smart phones called *agents* also communicate with oyster main server through Ethernet and 3G/LTE mobile communication. In addition to the amount of the oysters produced at all oyster workplaces, the environment of oyster workplaces, such as temperature and humidity can be monitored on *agent*s to judge the freshness. Two stage architecture with multiple channels makes it possible to monitor the amount of oyster product and environment of the oyster workplace at any place using PC or smart phone in real time. We developed the embedded system, the monitoring program running on the server and the monitoring program running on the client PC and the app running on the client smart phone, which access the monitoring program at the control center. We are improving our system to expand the application area for various kinds of marine products.

REFERENCES

[1] R. Drath, and A. Horch, "Industrie 4.0: Hit or Hype?", IEEE Industrial Electronics Magazine, 8(2), 2014, pp. 56-58.
[2] Kagermann, H., W. Wahlster, and J. Helbig, eds., Recommendations for implementing the strategic initiative Industrie 4.0: Final report of the Industrie 4.0 Working Group, Frankfurt, 2013.
[3] Giusto, D., A. Iera, G. Morabito, and L. Atzori, eds., The Internet of Things, Springer, New York, 2010.
[4] Mario Hermann, Tobias Pentek and Boris Otto, "Design Principles for Industry 4.0 Scenarios," 2016 49 Hawaii Internal Conferencr on System Sciences, 2016. Yeong-Yil Yang, Young-Ho Choi, Young-Sik Park, and Jong-Chul Lee, "IoT Based Oyster Management System," *Int'l Journal of Circuit and Electronics*, Vol 2, pp. 83–88, 2017.
[5] Hee Jang, Ki Won Nam and Yong Gyu Jung, "Smart Building Block Toys using Internet of Things Technology,", *International Journal of Advanced Culture Technology*, Vol. 4, No. 2 pp. 34-37, 2016.
[6] Eun-Soo Choi, Min-Soo Kang, Yong Gyu Jung, and Jean Kyung Paik, "Implementation of IoT-based Automatic Inventory Management System," *International Journal of Advanced Culture Technology*, Vol. 5, No. 1 pp. 70-75, 2017.
[7] Byung-Ho Cho and Heui-Hak Ahn, "Analysis and Design of Smart Vending Machine System based on IoT," *The Journal of The Institute of Internet, Broadcasting and Communication*, Vol. 19, No. 3, pp. 121-126, 2019.
[8] Yeong-Yil Yang, Young-Ho Choi, Young-Sik Park, and Jong-Chul Lee, "IoT Based Oyster Management System," *Int'l Journal of Circuit and Electronics*, Vol. 2, pp. 83–88, 2017.
[9] Yeong-Yil Yang, Young-Sik Park, Hyun-Jong Lee, Young-Ho Choi, and Jong-Chul Lee, "Multi-Channel Housing Monitoring System," *NAUN Int'l Journal of Computers and Communication*, Vol. 10, pp. 74–77, 2016.
[10] Kaiguo Li, Zhiliang Kang, Wuweu Ding, and Shen Mao, "Design of Appliance Control System Based on TCP/IP Protocol," *Journal of Measurement and Control Technology*, vol. 30, No. 7, pp. 41–45, 2011.
[11] Wenbing Wan, Xuerui Li, and Yonghua Shi, "Design and Research of Smart Switch Control System Based on Zig," *Journal of Embedded System*, vol. 32, No. 3, pp. 57–79, 2010.
[12] Tengfei Ahang, Qinxiao Li, and Funin Ma, "Remote Control System of Smart Appliances Based on Wireless Sensor Networks," *25th Chinese Control and Decision Conference* (*CICC*), pp. 3704–3709, 2013.
[13] Rethina kumar, Gopinath Ganapathy and Jeong-Jin Kang, "A Novel Architecture for Mobile Crowd and Cloud computing for Health care," *International Journal of Advanced Culture Technology*, Vol. 6, No. 4 pp. 226-232, 2018.
[14] Se-Jun Park, Eik-Hyeon Choi, Tae-Jun Yoo, Ji-Soo Mo and Sang-Ho Lee, "Implementation of Pet Management System using Lora Module," *The Journal of The Institute of Internet, Broadcasting and Communication*, Vol. 18, No. 5, pp. 275-281, 2018.
[15] Kenneth L. Calvert and Michael J. Donahoo, TCP/IP Sockets in Java, Second Edition: Practical Guide for Programmers 2nd Edition, 2008.

[16] Fiach Reid, Network Programming in .NET: C# & Visual Basic .NET 1st Edition, 2004.
[17] Johannes Eickhold, Serial Communication in Java with Rasberry Pi and RXTX, 2012.