International Journal of Agriculture and Environmental Research

ISSN: 2455-6939

Volume:03, Issue:06 "November-December 2017"

A THREE-LAYERS SENSOR SYSTEM ARCHITECTURE IMPLEMENTED TO AUTOMATED WEATHER STATION FOR SMARTFARM

[1]JoonYong KIM, [3]ChunGu LEE, [2]Young-Joo KIM, [1,3,*]Joong-Yong RHEE

[1]Research Institute for Agriculture and Life Sciences,
Seoul National University, Seoul, 151-921, Republic of Korea

[2]Environmental Materials & Components Center,
Korea Institute of Industrial Technology, Jeonju, Jeollabukdo, Republic of Korea

[3]Department of Biosystems and Biomaterials Science & Engineering,
Seoul National University, Seoul, 151-921, Republic of Korea

*Corresponding author

ABSTRACT

Smartfarm is a technical trend to improve agricultural value chain through ICT convergence. To build up a smartfarm system, ICT components like sensors, communication modules are important. However, commercial sensor systems for local farms have some management problems such as data loss and high maintenance cost because they use personal computers as their user interface and data storage. The objectives of our research were designing the three-layers architecture that solves these problems, implementing it to an automated weather station, and evaluating it.

The three-layers architecture has the sensor layer, the data management layer, and the user interface layer. In sensor layer, a sensor node collects data and transfers it. The data is stored and managed in the data management layer. A user can analyze the data and extract information in the user interface layer.

Based on this three-layers architecture, an AWS system consisting of a weather station, a data management unit, and a monitoring application was developed. The weather station measures seven meteorological factors. The data management unit manages the weather data and communicates with the weather station and the monitoring application. The monitoring application shows weather information to users. In order to evaluate the AWS system under real conditions, two systems were installed at local farms and compared with two conventional systems. The data loss rate of new architecture system was under 1%.

As a result, the three-layers sensor system architecture could lessen the management burden of users and give advantages in terms of user interface, cost, and interconnectivity.

Keywords: Agricultural information system, automated weather station, embedded systems, precision agriculture, smart farm

1. INTRODUCTION

Smartfarm is a technical trend to improve agricultural value chain based on knowledge and data; for example, it can control the micro-climate of crops by analyzing environmental factors and plant growth and lead to an improvement in productivity. With the technological developments in information and communication domain, and agriculture, various technologies and systems for farm automation have been developed (Kaivosoja et al., 2014; Kim et al., 2014; Nikolidakis et al., 2015; Rajagopal et al., 2014). Nowadays, there are many commercial products to notify local weather, greenhouse environmental condition, etc, and to control greenhouse environment, nutrient solutions, CO2 level and so on. However, it has not been used widely. There were some reports to analyze the reason. According to Kwon et al. (2014), most farmers felt that it was inconvenient to use a system related to a personal computer. They did not have deep knowledge of computers and it was a large burden for them to manage the system. In addition, there is another management issue that is related to after-sales service. Cho et al. (2015) reported problem occurrence rate of ICT solution installed on greenhouses. The problems were occurred on sensor, actuator, environmental control system, greenhouse structure, remote control technique, plant management technique, energy saving technique, and utilization of software. In order to realize the value of smartfarm and to distribute it widely, it is important to solve those problems. Among those problems, this research focused on problems related to sensors because the sensor is the most basic component of the smartfarm system.

Among plenty of sensors using agricultural domain, the automated weather station (AWS) is a representative sensor system. Because weather affects all growth characteristics of crops, such as size, weight, color, and maturation period, and can cause crop damages, such as diseases and frost (Felland et al., 1997; Fitzell et al., 1984; Marsh et al., 1999; Stanley et al., 2000), the base technology for smart farm system is to measure meteorological factors. The ASW for measuring meteorological factors have been developed and have already been applied to the crop production system. The AWS is helpful because it can save human labor and provide the ability to measure data from remote areas. According to Kuśmierek-Tomaszewska et al. (2012), the data obtained from an AWS can be applied and used for various applications, such as irrigation

ISSN: 2455-6939

Volume:03, Issue:06 "November-December 2017"

scheduling, estimating energy requirements for grain drying, predicting stage and rate of crop development, and predicting livestock weight gains (Hubbard et al., 1983). An AWS typically consists of a weather-proof enclosure containing meteorological sensors, a power source, and, optionally, a communication device. A personal computer is usually used both ways to store data and to monitor it.

The objectives of this study were to design a new architecture for sensor system, to implement the architecture to an AWS system, and to evaluate it. In traditional architecture, a computer has two roles: as a server for a database system and as a client for the user interface. The new designed architecture separates the two roles. A personal computer or a smart device could be used only for the user interface and an embedded system is used as a server to save and manage data. The AWS system included both hardware and software.

2. MATERIALS AND METHODS

2.1 Conventional architecture of AWS system

The typical architecture of an AWS system for a local farm is shown in Fig. 1. An AWS has several sensors for measuring meteorological factors, and the measured data is transferred to a computer, which has a database and a graphical user interface. The computer is typically a desktop computer running a Windows operating system. The computer should be always on in order that meteorological measurements could be saved continually to the database.

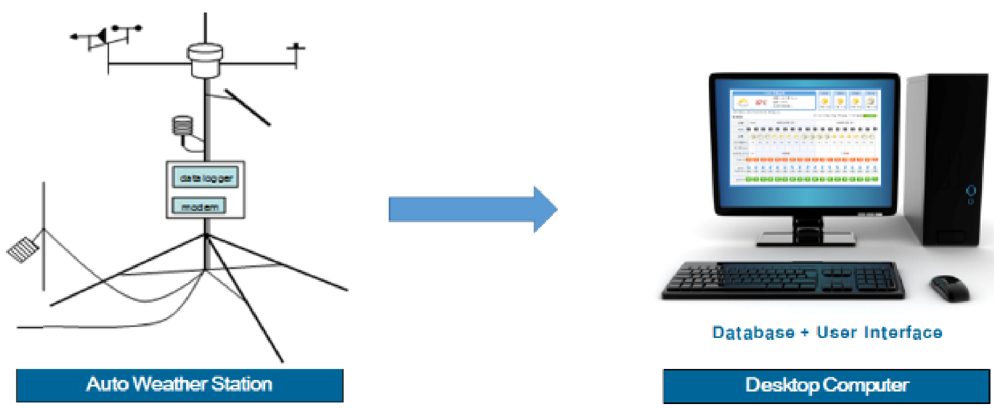

Fig 1. Typical architecture of an AWS system

This AWS architecture causes several problems. First, it can cause data loss. Many farmers use the computer only to check the weather. It means that they regard the computer as a monitoring device and not as a server. They turn the computer off frequently in order to save electricity or without meaning to and it causes data loss. Second, the architecture increases electricity costs.

ISSN: 2455-6939

Volume:03, Issue:06 "November-December 2017"

Since a desktop computer usually uses 60-300W, it is a burden for a farmer to leave the computer on every day. Using a data logger prevents data loss and reduces electricity costs, but it increases the installation cost. In addition, the data logger could connect one system usually and it does not have computing power to analyze stored data. Since the interconnectivity of sensor systems is essential for future agricultural information system, it might be a major weakness of the typical AWS system.

2.2 New architecture for AWS system

In order to overcome the problems of the typical architecture, a conceptual architecture with three layers was designed. The architecture, shown in Fig. 2, consists of a sensor layer, a data management layer, and a user interface layer. In sensor layer, a so-called sensor node which has sensors and a communication device collects data and transfer it. The data management layer is implemented on an embedded system. It stored and managed the data. A user can analyze the data and extract information in the user interface layer. Each layer has corresponding hardware. The hardware corresponding to the sensor layer is an AWS. The AWS does not have a data logger and it only contains meteorological sensors and a communication device. The hardware corresponding to the data management layer is an embedded system that manages the weather data that has been measured by the AWS. Nowadays, many embedded boards, such as the CubieBoard and the Raspberry PI, are cheap and powerful, and database systems can be executed on them. A monitoring device used to monitor weather data corresponds to the user interface layer.

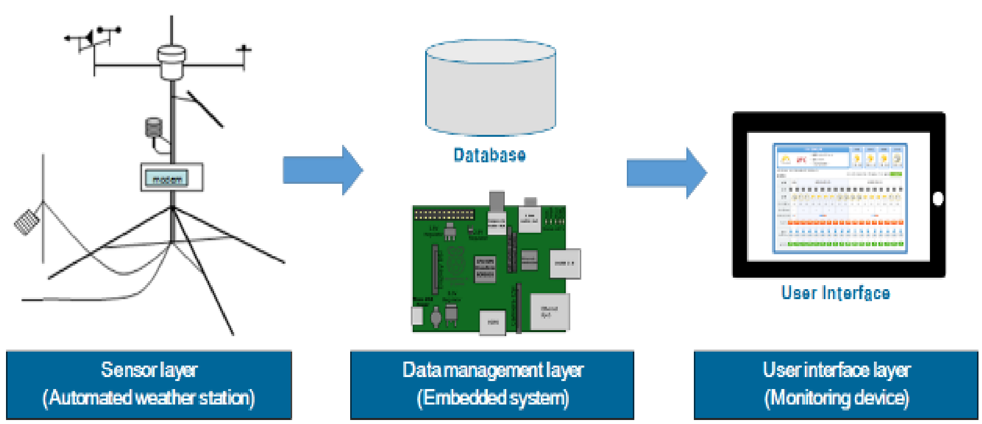

Fig 2. Concept of new architecture for an AWS system

Based on the architecture we designed, we created the AWS system shown in Fig.3 to consist of a weather station, a data management unit, and a desktop computer as a monitoring device. The weather station includes meteorological sensors, a data acquisition board, and a communication device. The meteorological sensors are capable of measuring the surrounding weather. The data acquisition board is able to read the electrical signals from the sensors and transfer them to the data management unit. Wireless communication between the weather station and the data management unit was enabled by using Bluetooth. The data management unit consists of an embedded board with a database system. It saves the observations from the weather station in the database. A user can use the monitoring application to monitor the observations. The monitoring application connects to the data management unit's database using Ethernet or WI-FI communication.

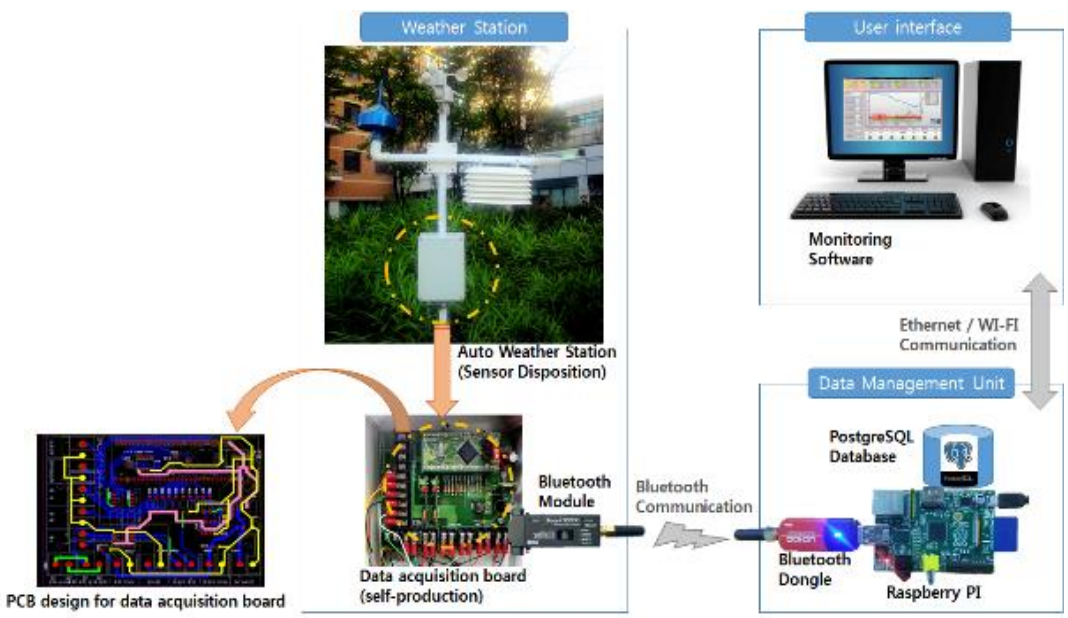

Fig 3. Structure of new AWS system

2.3 Hardware for the weather station

The weather station includes meteorological sensors, a data acquisition board, and a communication device. It was important to carefully select the meteorological factors to be measured so that the necessary information could be gathered while keeping the price of the final product in mind. We reviewed various papers and collected advice from experts. Creasy (1980) used humidity, net precipitation, and temperature to predict russet severity. Léchaudel et al. (2005) used maximum, minimum, and mean daily temperatures as well as hourly global radiation as model inputs. In 2010, the National IT Industry Promotion Agency (NIPA) of Korea published a report on guidelines for the installation and management of a Ubiquitous Sensor Network-based crop-growth environment management system. According to NIPA (2010), air temperature, humidity, and wind speed are important parameters for crop growth. Seven meteorological factors were selected: ambient temperature, ambient relative humidity, dew point, solar radiation, wind speed, wind direction, and rain status. Table 1 shows the specifications of the sensors used, all of which were selected considering their prices. Relative humidity and temperature were measured by a unique capacitive senor element and a band-gap sensor, respectively, which were embedded in a temperature/humidity sensor (SHT 10, Sensirion, Switzerland). Both sensors were seamlessly coupled to a 14-bit analog-to-digital converter and were fully calibrated. A weather sensor assembly (Weather Sensor Assembly p/n 80422, Argent Data Systems, USA) was used to measure wind speed, wind direction, and rainfall. Wind speed was measured by a cup-type anemometer, where a wind speed of 2.4 km/h causes an inner switch to contact and send one signal per second. Wind direction was computed using the angle of the anemoscope. When the angle changed, the inner resistance also changed which, in turn, led to a change in the output voltage. A pyranometer (SP-110, Apogee Instruments, Inc., USA) was used to measure the solar radiation. It generates 0.2 mV per W/m^2. A rain sensor (WRS-09R, Woosung Hitech, Co., Ltd., Korea) was used to check whether or not it rained. It had changing resistance type and a sensing area of 84 cm^2.

Table 1. Specifications of sensors

Source	Manufacturer	Model name	Specifications
Temperature	Sensirion	SHT10	Accuracy: ±4.5% Repeatability: ±0.1% Response time: 8s
Humidity			Accuracy: ±0.5°C Repeatability: ±0.1°C Response time: 5-30s
Dew point			Derived from humidity and temperature
Wind speed	Argent Data Systems	Weather Sensor Assembly p/n 80422	2.4 km/h = 1 signal / s
Wind direction			Changing resistance type
Solar radiation	Apogee Instruments, Inc.	SP-110	Photodiode type: 0.2mV per W/m^2
Rain	Woosung Hitech Co., Ltd.	WRS-09R	Changing resistance type Sensing area: 84 cm^2

The data acquisition board supported five input ports (temperature, humidity, and dew point use a single port) and Modbus protocol using RS-485 serial communication. Modbus is an application-layer protocol based on a master-slave architecture. Bluetooth communication, a wireless communication technology standard for exchanging data, was used to transmit data between the weather station and the data management unit. Bluetooth is known as a good wireless alternative to serial data communication such as RS-485. Table 2 shows the specifications of the Bluetooth modules.

Table 2. Specification of Bluetooth modules

Manufacturer	Model name	Specifications
Sena Technologies, Inc.	ParaniTM UD-100	Bluetooth 4.0 Class 1 USB2.0 Working Distance: 400 m (Dipole 3dBi)
	ParaniTM SD1100	Bluetooth 2.0 Class 1 RS485, RS422 Working Distance: 200 m (Dipole 3dBi)

The Raspberry Pi 1 Model B+ was selected as the hardware for the data management unit. Raspberry Pi is a small-sized single-board computer developed by the Raspberry Pi Foundation. It uses low power (3.0 W) compared to a desktop computer. It is able to use many Linux distributions as well as other operating systems. Because the Raspberry Pi supports Linux as its operating system, there are many types of open source software available. PostgreSQL (9.1, PostgreSQL Global Development Group) is an example of such open source software and it is an object-relational database management system. The data management unit used this software to store observations obtained from the weather station. Although the Raspberry Pi supports a variety of databases, PostgreSQL has an advantage because it handles geological data using its PostGIS extension.

2.4 Experiments for evaluation

A communication test and a data loss test were conducted in order to evaluate the developed AWS system. A communication test between the AWS and the data management unit was performed in the orchard fields of the Gyeonggi-do Agricultural Research and Extension Services, Republic of Korea. The working distances of the wireless communication devices may depend on the field conditions. It was necessary for us to know the stable communication distance. Data middleware, developed by Kim et al. (2013), was used for the field test. It sent request packets to the weather station and received back response packets. The tests were repeated ten times at each location and the middleware calculated the ratio of success communication and total elapsed time. Fig. 4 shows the location where the AWS was installed and where the communication tests occurred. There were no communication errors within 150m and it took approximately 0.98 s to process each request and response.

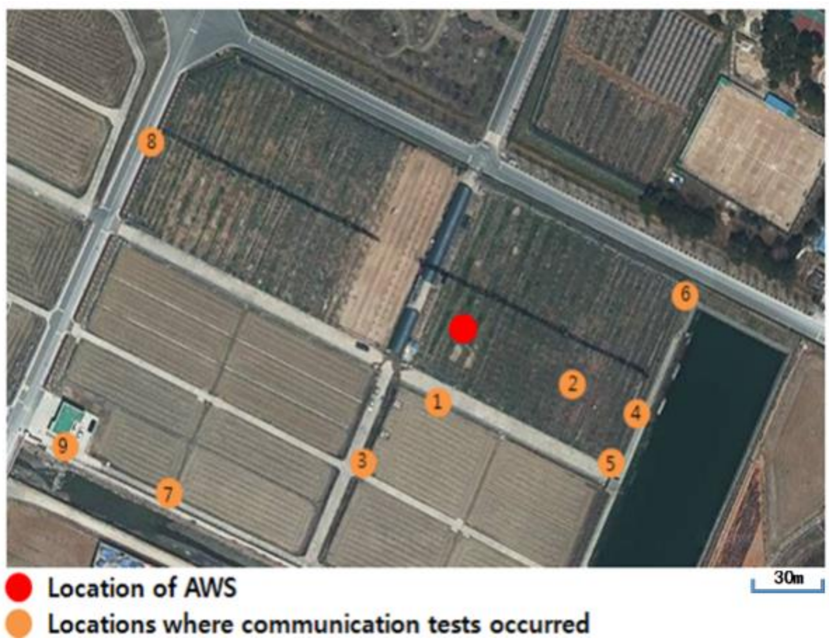

Fig 4. Locations of communication performance tests

Our AWS systems, shown in Fig. 5, were installed at two orchard farms, located in Namyangju and Ansung, Gyeonggi-do, Republic of Korea. Two orchard farms at Gongju, Chungcheongnam-do and Pyeongtaek, Gyeongggi-do, Republic of Korea were selected as a control group. These two farmswere already using commercialized AWS systems (Nongjung Cyber, Republic of Korea). After the installation of our two systems, the data loss rates were compared over a course of three months (2014.11 - 2015.01). In order to get unbiased results, the farmers were not informed about this evaluation and they were asked to use the systems as usual.

Fig 5. Views of AWS systems at Namyangju and Ansung, Gyeonggi-do, Republic of Korea

3. RESULTS AND DISCUSSION

3.1 Development of the data management unit

The data management unit has three roles. The first role was to collect observations from weather station. In order to get the observations, it sends a request to the station, receives and parses the response, and saves the parsed observations to the database. The Python programming language (2.7.6, Python Foundation) was used to develop software for data management unit. The software can encode and decode the Modbus packet which consists of an address, a function, data, and a 16-bit CRC checksum. Also, it was designed to apply other communication standards easily.

The second role of the data management unit is to manage the observations in a database. All observations were saved in PostgreSQL and the database schema used is shown in Fig. 6. There were seven tables, named “observation_#,” used to store observations. The data for each sensor was stored in a separate table. The descriptions for all seven tables’ schema were the same, containing time, a numeric value, and a string value. This means that the observations of each sensor could be abstracted into a time of measurement and a value (numeric or string). This abstraction makes it is easy to add new sensors to this system.

ISSN: 2455-6939

Volume:03, Issue:06 "November-December 2017"

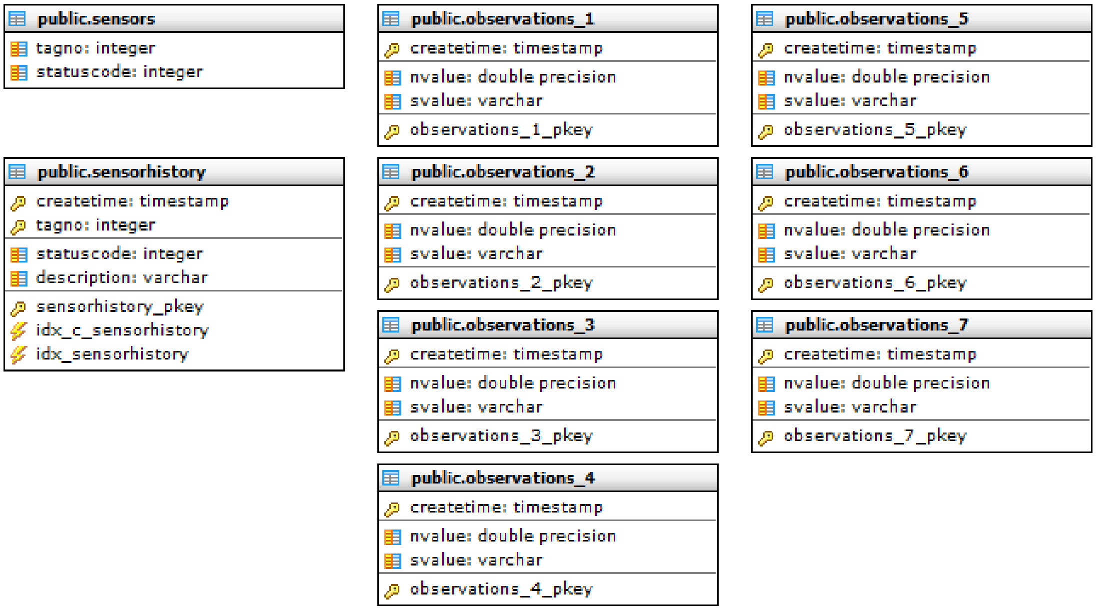

Fig 6. Database schema for the data management unit

The third role of the data management unit was to manage the statuses of the sensors. The “sensorhistory” table recorded the status of each sensor at a certain time. It can detect two types of malfunctions of the sensors. The first type is fixed-value errors. If a sensor records the same values continually, there might be a problem, since damaged sensors behave in this way. The threshold of how many same values can be recorded in a row without being flagged as a malfunction can be configured because it depends on the time interval of measurement. The second type of malfunction is abnormal-value errors, occurring when a sensor records a value outside of its normal range. For example, if relative humidity is -10% or ambient temperature is 100°C, the sensor might break down. When a malfunction is detected, the “statuscode” column of the “sensors” table is updated and a new record is appended to the “sensorhistory” table.

3.2 Development of the AWS monitoring application

The AWS monitoring application was developed to help a user easily monitor weather information. Delphi (Borland Software Corporation, 7, USA), an integrated development environment for graphical desktop applications, was used to develop the monitoring software. Delphi uses the Object Pascal language and generates native code for 32- and 64-bit Windows operating systems. It is known as a good development tool to develop graphical user interface application on Windows environment.

ISSN: 2455-6939

Volume:03, Issue:06 "November-December 2017"

The AWS monitoring application consists of three windows: the main window, the history window, and the configuration window. The main window is shown in Fig. 7 and has four functions. First, it shows the forecasted weather at the top. The forecasted weather information is retrieved from the MSN weather service. It can help a user compare the forecasted weather to the observed local weather. Because they may differ, it is important to give the user both. Second, the daily weather statistics information is shown on the left side. Although it shows the data for the current date by default, a user can select a different date to check the daily statistics of a specific day. Third, a 24-hour historical weather plot is located in the center of the window. It shows changes in all meteorological factors over the last 24 hours. Finally, it shows both the current weather and the sensor statuses on the bottom of the window. The seven different types of current meteorological observations are shown and are updated every minute. There are seven green buttons below the current weather table that represent all the sensors are normal. If a button is red, the data management unit detects a malfunction of a sensor.

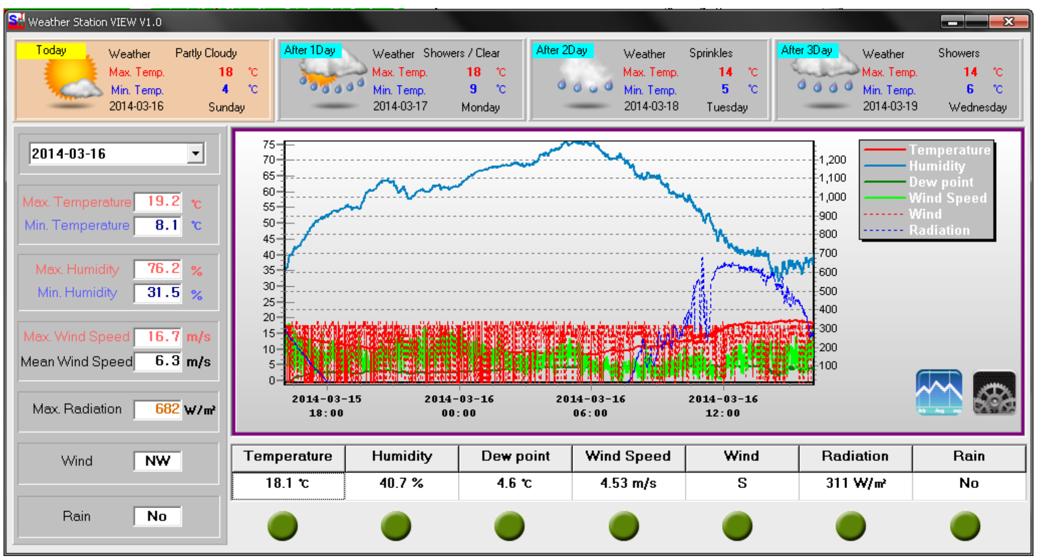

Fig 7. Main window of the monitoring software

Fig. 8 shows the history window and the configuration window. The history window shows historical weather information according to search conditions specified at the bottom of the window. The chart displayed in this window shows weather changes, and the weather at a specific time can be viewed by pointing at it with a mouse. It is possible to select specific sensor types by using the checkboxes and to search a particular period by using the radio buttons and calendar. The configuration window shows information about the database connection, the user's language, and area information. It is possible to change language in use because the user interface supports both Korean and English. In order to obtain the forecasted weather from the

MSN weather service, the software uses the area information of where the farm is located. The default city is Seoul, but the application supports 15 cities and provinces.

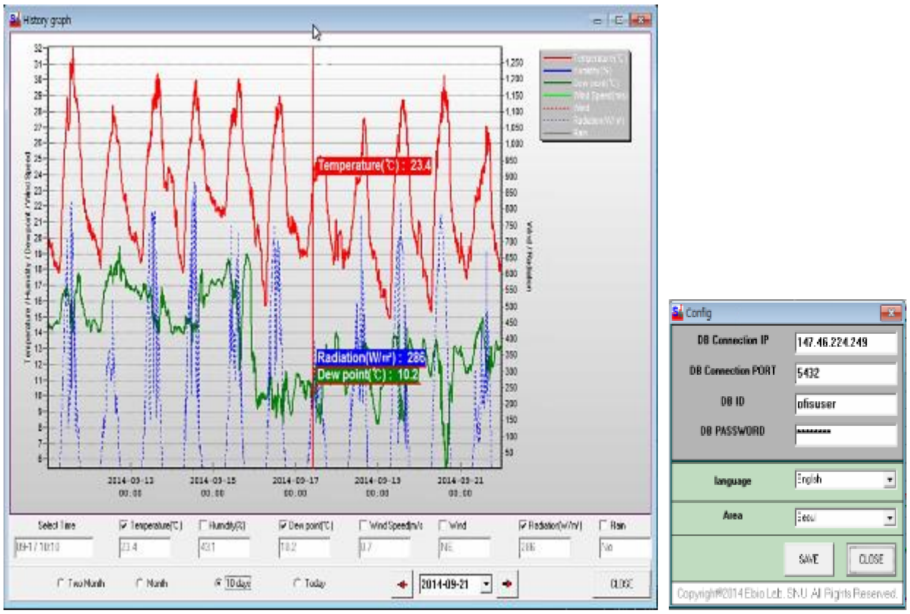

Fig 8. History window and configuration window of the monitoring software

3.3 Evaluation of developed AWS system

Table 3 shows the data loss rates of the different AWS systems over the three-month period. The data losses of our new architecture were much less than that of the typical architectures. Although the reasons for the loss were not clear, we suspect the unstable power of the rural areas might be the reason, because the losses occurred over specific time periods.

ISSN: 2455-6939

Volume:03, Issue:06 "November-December 2017"

Table 3. Field data on data loss rates of AWS systems

Architecture	Location	Data loss rate
Two-layer (Typical Architecture)	Gongju,Chungcheongnam-do (A)	19.82%
	Pyeongtaek,Gyeonggi-do (B)	45.52%
Three-layer (Suggested Architecture)	Namyangju, Gyeonggi-do (C)	0.6%
	Ansung, Gyeonggi-to (D)	0.13%

In addition to the high data loss rates, another important problem of the typical architecture was the large difference of data loss rates between two systems. It shows that the typical system depends highly on the user. After the test, the two farmers who use the typical systems were interviewed. The farmer of Farm (A) kept the computer on for the three month period except for during the year-end holidays, but the farmer of Farm (B) left the computer off for 10 days during the evaluation period and, after that, turned it on and off frequently. The first farmer made the mistake of turning off the computer over the year-end holidays and the second farmer did not know how important continual data collection is. The result clearly shows the disadvantages of the typical architecture even though our field experiments were small scale. The success of the typical system is closely related to the management ability and attention of its users. Even a farmer with good management skills can cause data loss by mistake. Therefore, it is important to develop a system that works well without human intervention.

Our new architecture detaches the management burden from the user. This was achieved by using an embedded system, housed in an enclosure, which has enough computing power to process data and to communicate with other devices. The component is a small electrical device with several LEDs and a power plug. It means the users do not have to care about the details of the embedded system. If there is a problem, they simply have to replace the component. This provides an obvious advantage as compared to desktop computer management.

The second advantage that our system provides is that a user can use the monitoring application on his desktop computer in the same way as they used the software for the typical system. The user can turn the computer off without any concerns about data loss. In addition, it is possible to implement mobile software for Android or iOS devices as a monitoring application.

The third advantage is that an embedded board has a low enough cost to replace a data logger. For example, the price of Raspberry Pi was only $25 USD for Model A and $35 USD for Model B. The total cost of our developed AWS system is lower than that of a typical AWS system. In addition, it is possible to save electricity costs since an embedded system uses less power than a personal computer.

Finally, it is possible to implement a new communication protocol on a new architecture. Our developed AWS system has the ability to communicate with other systems through the Internet. For example, the TTAK.KO-06-0288 is a set of key standards for implementing a greenhouse automation system in Korea. The TTAK.KO-06-0288 Part 1 describes communication protocols between sensor nodes like the AWS system and a greenhouse control gateway (TTA, 2012). Also, it is possible to save data on a cloud computing system. For future research, the developed AWS system could be interconnected with the Open Farm Information System, a cloud-based farm management information system (Kim et al., 2014).

4. CONCLUSIONS

The AWS has great potential for the automation of farm operations based on knowledge. However, the typical AWS system has several problems caused by its architecture, such as data loss and maintenance cost. The objectives of this study were to design a new architecture that lightened management work, to develop an AWS system that adopted this architecture, and to evaluate the performance of the system. The new architecture has three layers: the weather station layer, the data management layer, and the user interface layer. Based on this three layer architecture, an AWS system consists of three parts: a weather station, a data management unit, and a user interface device that displays a monitoring application. The weather station measures ambient temperature, ambient relative humidity, dew point, solar radiation, wind speed, wind direction, and rain status. The data management unit manages the weather data and communicates with the weather station and the monitoring application. The monitoring application shows both forecasted weather and the local weather. In order to evaluate the developed AWS system under real conditions, two developed AWS systems were installed at local farms and compared to two typical AWS systems already installed on other local farms. The data loss rates of our AWS systems were under 1% over the course of three months (2014.11 – 2015.01) and the results were similar between both of our systems. This means that our AWS system, based on the three layer architecture, successfully lessens the management burden of users and has some advantages in terms of user interface, cost, and interconnectivity.

ACKNOWLEDGEMENT

This work was supported by Korea Institute of Planning and Evaluation for Technology in Food, Agriculture, Forestry(IPET) through Advanced Production Technology Development Program, funded by Ministry of Agriculture, Food and Rural Affairs(MAFRA)(316077)

REFERENCES

[1] Creasy, L., 1980. The correlation of weather parameters with russet of 'Golden Delicious' apples under orchard conditions. Journal of the American Society for Horticultural Science 105: 735-738.

[2] Felland, C., J. Travis, J. Russo, W.Kleiner and E.Rajotte, 1997. Validation of site-specific weather data for insect phenology and disease development in Pennsylvania apple orchards. PA. Fruit News 77(2): 10-17.

[3] Fitzell, R.D., C.M. Peak and R.E. Darnell, 1984. A model for estimating infection levels of anthracnose disease of mango. Annals of Applied Biology 104: 451-458.

[4] Hubbard, K., N. Rosenberg and D. Nielsen, 1983. Automated Weather Data Network for Agriculture. Journal of Water Resources Planning and Management 109(3): 213-222.

[5] Kaivosoja, J., M.Jackenkroll, R.Linkolehto, M. Weis and R.Gerhards, 2014. Automatic control of farming operations based on spatial web services. Computers and Electronics in Agriculture 100: 110-115.

[6] Kim, J. Y., C. G. Lee, S. H. Baek and J. Y. Rhee, 2014. Design and Analysis of Requirements for Open Farm Information System (OFIS), KSAM 2014 Autumn Conference, pp. 63-64, Korea Society for Agricultural Machinery.

[7] Kim, J. Y., C. G. Lee, T. H. Kwon, G. H. Park and J. Y. Rhee, 2013. Development of an Agricultural Data Middleware to Integrate Multiple Sensor Networks for an Farm Environment Monitoring System. Journal of Biosystems Engineering 38(1): 25-32.

[8] Kuśmierek-Tomaszewska, R., J.Żarski and S.Dudek, 2012. Meteorological automated weather station data application for plant water requirements estimation. Computers and Electronics in Agriculture 88: 44-51.

[9] Kwon, T. H., J. Y. Kim, G.H. Park, C.G. Lee, A.Ashitiani-Araghi, S.H.Baek and J.Y. Rhee, 2014. Survey on Informatization Status of Farmers for Introducing Ubiquitous Agriculture

ISSN: 2455-6939

Volume:03, Issue:06 "November-December 2017"

Information System. Journal of Biosystems Engineering 39(1):57-67.

[10] Léchaudel, M., M.Génard, F.Lescourret, L. Urban and M.Jannoyer, 2005. Modeling effects of weather and source–sink relationships on mango fruit growth. Tree physiology 25(5): 583-597.

[11] Laurenson, M., 1989. ODE-Orchard Decision Environment, II International Symposium on Computer Modelling in Fruit Research and Orchard Management 276: 301-304.

[12] Marsh, K., A. Richardson and E.Macrae, 1999. Early-and mid-season temperature effects on the growth and composition of satsuma mandarins. Journal of Horticultural Science and Biotechnology 74(4): 443-451.

[13] Nikolidakis, S.A., D.Kandris, D.D.Vergados and C.Douligeris, 2015. Energy efficient automated control of irrigation in agriculture by using wireless sensor networks. Computers and Electronics in Agriculture 113: 154-163.

[14] NIPA, 2010. Installation and management of USN based crop growth environment management system. National IT industry Promotion Agency of Korea.

[15] Pierce, F. J. and T.V. Elliott, 2008. Regional and on-farm wireless sensor networks for agricultural systems in Eastern Washington. Computers and Electronics in Agriculture 61(1): 32-43.

[16] Rajagopal, G., V.M.Lodd, A.Vignesh, R. Rajesh and V.Vijayaraghavan, 2014. Low cost cloud based intelligent farm automation system using Bluetooth Low Energy, Humanitarian Technology Conference (R10-HTC), 2014 IEEE Region 10: 127-132.

[17] Stanley, C., D. Tustin, G. Lupton, S.McArtney, W. Cashmore and H. D. Silva, 2000. Towards understanding the role of temperature in apple fruit growth responses in three geographical regions within New Zealand. Journal of Horticultural Science and Biotechnology 75(4): 413-422.

[18] Stewart, T. R., R.W. Katz and A.H. Murphy, 1984. Value of weather information: A descriptive study of the fruit-frost problem. Bulletin of the American Meteorological Society 65(2): 126-137.

[19] TTA, 2012. Greenhouse Control System – Part 1: Interface for Between Sensor Nodes and Greenhouse Control Gateway, TTAK.KO-06-0288-Part1.

