THE DESIGN OF ENTERPRISE SENSOR NETWORK FOR REAL-TIME, LARGE-SCALE AND MASSIVE SENSOR WITH A CASE OF USING MOBILE PHONE AS SENSOR NODE

Apichon Witayangkurn*[1] and Ryosuke Shibasaki[2]

[1]PhD Candidate, Department of Civil Engineering, the University of Tokyo, 4-6-1, Komaba, Meguro-ku, Tokyo 153-8505, JAPAN; Tel: +81-3-5452-6412 Fax: +81-3-5452-6414;
E-mail: apichon@iis.u-tokyo.ac.jp

[2]Professor, Center for Spatial Information Science, the University of Tokyo, 5-1-5 Kashiwanoha, Kashiwa-shi, Chiba 277-8568, JAPAN; Tel: +81-4-7136-4291 Fax: +81-4-7136-4292;
E-mail: shiba@csis.u-toyko.ac.jp

KEY WORDS: Real-time monitoring, Sensor network, Mobile sensing, Spatial information

ABSTRACT: Sensor technology is very famous and used in many fields of research and application such as environment monitoring, agriculture field monitoring, traffic monitoring and even human flow estimation. However, most of applications are developed based on its application and sensor specification. Hence, it is difficult to apply those systems for using in other purpose or with different type of sensor. Moreover, sharing sensor information among sensor network is problematic due to lack of standardization. In this research, we listed all main issues and problems in the development of sensor network and proposed a design and system called “Enterprise Sensor Network” that addressed most of issues such as real-time, large-scale sensor node, massive sensor data, high-frequency, multi-dimension data, flexible data communication support and international standard support. We used messaging service technique to make the system scalable and extensible in performance of data retrieving side and overcome the problems in data communication allowed sensor node to work in any type of network. Furthermore, this method allows 2-way control and synchronization of sensor data. Spatial database and cloud technique are also considered in the design to handle large-scale sensor data and support spatial query for spatio-temporal data analysis. To achieve standardization, we designed the system to support Sensor Web Enable (SWE) defined by OGC (Open Geospatial Consortium) especially in section of Sensor Observation Service (SOS). For test-bed data and devices, we focused on using mobile phone as sensor node because currently every smartphone is built-in necessary sensors for being a sensor node such as internet connection and position sensor. Also, there are many sensor boards that can directly connect to mobile phone or via Bluetooth.

1. INTRODUCTION

Sensor technology is widely used in various fields of applications nowadays due to advent communication technology and electronics technology. Weather and environment sensors such as temperature, humidity and solar radiation are the examples of sensors for measuring natural phenomenal. Sensors can be both passive and active sensors; however, both types of sensors require additional equipments to collect and store data from sensors as well as transfer data to central unit for further analysis and development. It is normally called a sensor station which consisted of control unit, sensors and communication media. Many of sensor stations are deployed in rural areas or very remote areas so collecting data as well as maintenance become difficulty and limitation. According to the development of the Internet, it enables sensor stations to connect each other and also communication with central server or main unit located in far-remote areas. Center server will receive data from sensor station, then process data and provide virtualization for users or interface for other applications. A concept of sensor network has been initiated to describe connectivity among sensor stations. There are several organizations and working group work on sensor standard such as The Open Geospatial Consortium (OGC) that play important role in promoting sensor standard as well as sensor network. Sensor Web Enables (SWE) is one of the standards introduced by OGC despite that it was defined only data format for communication among sensor systems and did not focus on a concrete detail of system development. Hence, sensor applications and sensor networks are developed using different platform, architecture and programing language. It is developed based on their expert and application requirement. Apart from those difficulties, since sensors are mostly designed and use for always-on and real-time purpose, it raised many issues such as real-time, large number of sensor nodes, massive volume of sensor data and high-frequency. How to handle and manipulate large and massive data is very challenge topic. Moreover, flexible data communication is one of importance issues because it is usually deployed and used in various conditions such as unstable internet connection. So, to archive and overcome those problem and difficulties, we designed new sensor network system call “Enterprise Sensor Network” that trying addressed main issues by introduced new method to utilize cloud technology in sensor network for robustness and scalable. In addition to Cloud technology, messaging service method was also introduced as play important role as communication media from sharing sensor data and control sensor along the network.

2. LITERATURE AND RELATED WORK

2.1 Sensor Network

Sensor Network is a group or system of heterogeneous sensors connected together using communication infrastructure to exchange information between sensor stations or sensor nodes so that all sensor nodes are able to link or synchronize data among each other or main station and finally act as network. It consists of multiple sensor stations called sensor nodes. It is deployed in the designated areas that intend to measure and collect sensor data such as weather and environment. Rural areas or disaster sites are usually installed the station. A sensor node commonly equipped with main control system, sensors, power module and communication media. The control system is a computer that has high functionality but consumes less power as well as have more tolerance than normal computer. Common deployed sensors are weather and environment sensors such as temperature, pressure, speed, wind direction, humidity and voltage. Communication media is any service that can link node together. It can be wired or wireless network, for example: Ethernet, ASDL and GPRS. The power is used from electric utility, battery and solar cell. Basically, the sensor will obtain physical information according to types of sensors and send data to main control so that it can then exchange sensor data with other stations. Potential applications comprise of weather and environment monitoring, traffic monitoring and video surveillance. Apart from Sensor node, it has another system called Central Sensor Services. It is a sensor data middleware that provides users with a platform to receive data from remote field sensor networks including data interface and virtualization. It consisted of 4 main components: Sensor Synchronization Service, Sensor Virtualization Service, Sensor Interface Service and Sensor Data Manipulation Service. Synchronization Service handles data transfer from remote side to center server to ensure data consistency and sequence of data. It also controls sensor station to check status and change configuration from remote. Sensor Virtualization Service provides user interface for users to observe sensor data. Interactive web interface with graph is usually a choice for virtualization nowadays due to portable concept that no needs to install software on client or user machine. Sensor Interface Service deals with standard interface to provide other application to communication with the system and normally, it is developed as Application Programming Interface (API). Sensor data Manipulation Service handles data processing function such as data interpolation, data assimilation and noise removing. Since Central sensor services need to be always running so that it can retrieve data from remote station any time, it typically characterized by the features: High performance, Scalability, Reliability and Open architecture. Sensor Service GIRD (SSG) is an example of the development in Sensor Network (Honda et al., 2009) as well as facade SOS (F-SOS) that proposed new method to integrative various SOS (Ahn et al., 2010.)

2.2 Issues in development of sensor network

Normally, sensor station and sensor network are designed for full-time operation that have to run 24 hours basis. It required special mechanism and technique to handle data transferred from sensor node to sensor network. So, the challenge is that how to deal and handle large number of sensor node as well as large size of sensor data. Moreover, since it will run for long time, it needs to be easy and minimize effort for scaling from small size to large size of network both in term of computation processor and storage resources. Heterogeneous and vender-specific sensors are also an issue because it is difficult to connect with sensor network due to different format of data connection and protocol. There are several of data communication for connecting with sensor such as HTML, Serial, TCP and file-based data. A standard model for sensor data feeder must be defined so that cost of developing new feeder is minimized. Different sensor data type is also need to be taking into account. Sensor data can be number, text, time and multimedia value such as video, audio, image and even streaming data (McFerren et al., 2009). Apart from data type, frequency and dimension of data are also need to be taken care of. Laser Scan is one example of high-frequency data and multi-dimensional data. To enable data communication among sensor, network connection is very important. Sensor network must be able to operate under any network both stable and unstable network because mostly station will deploy rural areas and those areas have high possibility of unstable network in case of using mobile network or satellite connection. Besides, it must provide channel or interface for 3rd party application or other systems to inquiry sensor data. Standardization interface can be used to enhance compatibility with other software. Finally, rapid installation and ease of use are concerned since sensor node usually deploy over wide-areas so it should be easy enough to use and setup to allow fast deployment.

2.3 Cloud Technology

Today, we are surrounded with a lot of data as seen in many social network systems. They have to handle such huge data generated and keeping more by users participating in social and finally go to terabytes and petabytes of data. The question raised how to handle those data and provide response within concerned time. Cloud technology provides computation, software, data access and storage services that do not required to know physical location of the resources. It involved with dynamically scalable and virtualized resources. It normally expresses the following characteristic: Cost, Location dependence, Reliability, Scalability, Performance and Security (Yang et al., 2010.)

From sensor network perspective, it can enhance sensor network to archive large-scale support and massive data processing. In this paper, we focus on using a cloud computing technology called Hadoop. Hadoop is open source large-scale distributed data processing that mainly designs to work on commodity hardware (Hadoop, 2008.) That mean it does not required high performance server-type hardware with special storage system. It can work with normal computer without special storage. It is also scalable. Increasing system performance and storage, it simply adds new node to Hadoop with required no code modification and that result in why Google, Yahoo and Facebook used it as backend system (Lam, 2011.)

3. ENTERPRISE SENSOR NETWORK

In this paper, we focus on designing and developing a prototype of sensor network system to support various heterogeneous sensors, support any network topology and easily scale from small to large size network with minimizing efforts and human operation. The key features are Large-scale support with cloud, Massive data and real-time data processing, Flexible data communication, Easy installation and ease of use, High-frequency and multi-dimension support, Open standard and integrating support, Spatial data support.

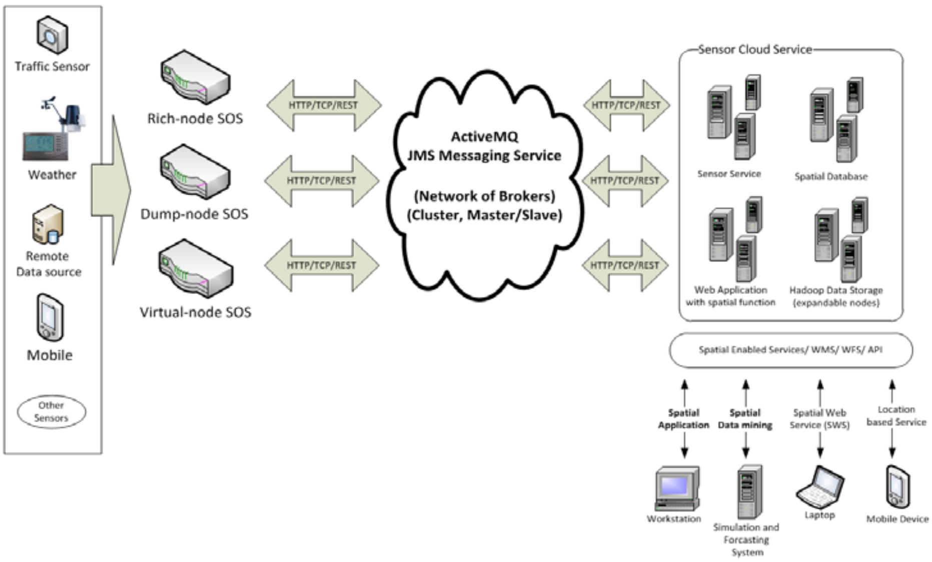

Figure 1 Enterprise Sensor Network Diagram

Enterprise Sensor Network consists of 3 main components: Sensor station, Messaging Service and Sensor Cloud Service. Sensor station is the station installed and deployed at remote or field site. It handles feeding data from sensors. Sensor station is divided into 3 types. Rich-node station is fully functional sensor station that means it cover all features such as 2-way control, web interface. Dump-node station is data feeder only station. It does not have web interface for virtualization and configuration. This aims to reduce storage usage and processing time that suitable for low power consumption device. However, data virtualization and configuration can be done via Sensor Cloud Service. Virtual-node station has the meaning as its name. It does not require physical hardware for installation and operation. A dedicated server will be used for running plenty of virtual-nodes to feed and collect data from other sensor systems aimed to increase integration and communication. Messaging Service is data communication media that allow 2-way communication between sensor station and sensor cloud service. It play important role in making system support any network topology. Lastly, Sensor Cloud Service, it is central sensor service. In our architecture, we used cloud technique called Hadoop as backend system to handle scalable. Besides, it is also enhanced with spatial database to enable spatial support. The detail for each component will be described in the following section, however; we will discuss only 2 techniques for handling large-scale and flexible data communication.

3.1 Sensor Station

The Sensor Station is a combination of SOS Service and Web Server. It is designed and implemented based a standard named Sensor Observation Service (SOS), and the sensor information can be obtained in SensorML format which describes in Observation and Measurement (O&M) Encodings. Station is capable of controlling more than one sensor and cameras. It also has the capability to collect data from several types of weather stations and data loggers. Once deployed in the field, the Sensor Station can register sensors to Sensor Cloud Services. The sensor set can be added or changed easily with a user-friendly interface; the calibration equation and other parameters will be selected appropriately to obtain the correct sensor output. Once registered, the Sensor Station can be controlled and configured remotely from the Sensor Cloud Service. The Sensor Station has been made resilient to overcome firewalls and NATs, so that sensor data can be sent from any kind of Internet connection. The SOS Station has its own web server, which is capable of displaying sensor data and images in the local network even if connection to cloud service is lost. As shown in Figure 2, it shows common setup diagram for one station and architecture diagram. It is separated into 2 parts: web server and SOS service. As described in previous section, some type of sensor station does not have web server installed in the station to reduce processing time and storage. Web server provide web interface for user and engineer to operate with the station such as data manipulation, changing configuration and seeing virtualization. Once configuration changed, it will communicate with SOS service using RMI service to reactivate function according to new configuration. Apart from web server, SOS service is a main component handled all necessary functions for acting as sensor node. It consists of Data feeder service, OGC SOS interface, Command Service, Scheduler Service and Data Synchronization Service. Data feeder has been designed and developed as service layer which is separately from the core system. The feeder templates including HTTP connection, Database connection, Serial connection and file-based reader are provided to allow custom feeder developed by other parties such as sensor manufacturers. Data feeder also support both pull and push mode. Command Service is used for receiving and sending command to Sensor Cloud Service. All configurations changed in both Sensor Station and Sensor Cloud Service will be handled by this service. Data Synchronization Service handles the process of transferring measurement data from station to cloud. It has done by using messaging service. OGC SOS interface is developed to allow OGC enable application to communicate with the system. To clear data or schedule data feeder to work on specific time, schedule service is used for that purpose.

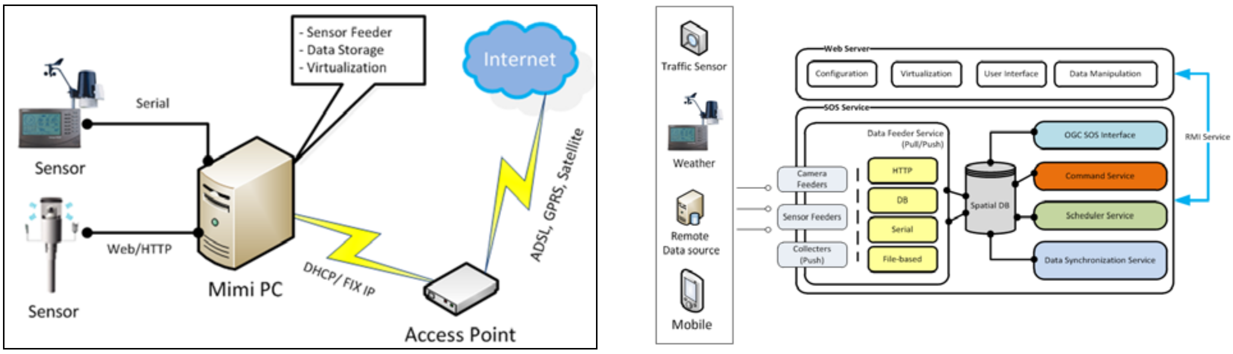

Figure 2 Common setup diagram (left) and Sensor Station diagram (right)

3.2 Messaging Service

To enable 2-ways control between sensor station and sensor cloud services, we used JMS messaging service named ActiveMQ, open source messaging under apache project. It provides an asynchronous communication protocol meant that sender and receiver do not need to interact at same time as usual TCP connection. It also supports several of protocol such as TCP, UDP, SSL, REST, and HTTP allowed service to operate and bypass security in any network system. Moreover, since it is an asynchronous protocol, it does not require public address to interact each other. For storing messages, it support file-based with journal and database which can select regarding purpose of usage. Load balance and cluster support is one of the important features in message queue service that enhanced sensor network for scalable and large-scale support. It has a concept of network of brokers meant brokers are linked together to from a network or cluster of brokers. A network of brokers can also use various network topologies such as hub-and-spoke, daisy chain, or mesh.

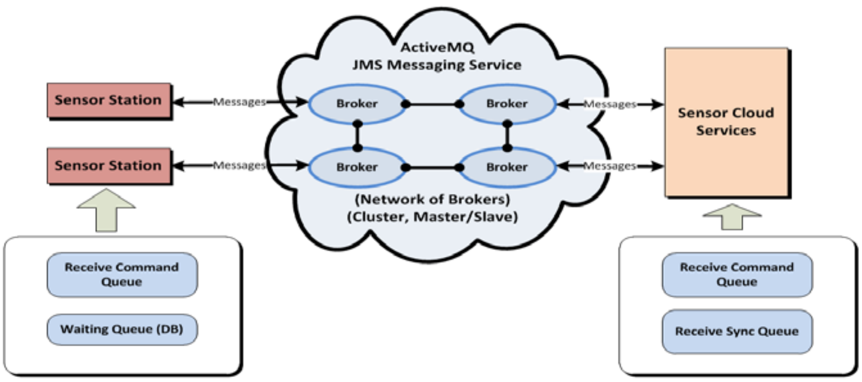

Figure 3 Messaging Service diagram with network of brokers

As shown in Figure 3, multiple brokers connect together and share messages among each other. Once a message is sent to a broker, that broker will distribute message to other brokers based on routing policy. On sensor station side, it has a Receive Command Queue to receive command from cloud service such as changing configuration, polling status or restart local service. Waiting queue is a service to collect failed sent message in case of internet connection lost or cannot connect to messaging service. It will automatically resume process and retransmit messages after connection available. On sensor cloud services, A Receive Sync Queue is listened by thread-enabled services that can increase number of consumer for scalable purpose. To receive command result and configuration change, A Receive Command Queue is set to handle it.

3.3 Sensor Cloud Services

It is a central sensor service which aim to provide users with a system to accommodate sensor data sending from remote station as well as control stations. However, for Sensor Cloud Service, it was designed and divided into many services according to its functionality and operation. At lowest level, the system was designed to use cloud services, Hadoop, instead of normal database server as backend and storage. Hadoop is an open source software framework for data intensive and distributed application. There are many services and framework under Hadoop umbrella; however, in this research, we focused on Hadoop Distributed File System (HDFS) and Hive. To setup and use Hadoop for full operation mode, it required to run 5 components: NameNode, DataNodes, Secondary NameNode (SNN), JobTracker and TaskTrackers. NameNode is the bookkeeper of HDFS; it keeps track of how your files are broken down into file blocks, which nodes store those blocks, and the overall health of the distributed filesystem. DataNodes are the workhorses of the filesystem. They store and retrieve blocks when they are told to (by clients or the namenode), and they report back to the namenode periodically with lists of blocks that they are storing. Secondary NameNode (SNN) is an assistant daemon for monitoring the state of the cluster HDFS and the SNN help snapshots NameNode to help minimize the downtime and loss of data. JobTracker is the liaison between your application and Hadoop. Once you submit your code to your cluster, the JobTracker determines the execution plan by determining which files to process, assigns nodes to different tasks, and monitors all tasks as they are running. TaskTrackers is responsible for executing the individual tasks that the JobTracker assigns and manage the execution of individual tasks on each slave node. Hive is a data warehousing package built on top of Hadoop. It target users who are familiar and comfortable with SQL to do ad hoc quires, summarization and data analysis. Web GUI and JDBC are provided for interacting with Hive by issuing queries in a SQL-like language called HiveQL (Lam, 2011.) In our designed, upper level services, web services, synchronization service and command service interact with cloud services using JDBC. All data are stored in HDFS and Hive handles write and read data from HDFS. At this level, Synchronization services handle data receiving from messaging service. It is multi-thread service that can increase number of queue consumer for scalable. Command Services is responsible for checking status of every station by sending polling command to remote station in interval and control remote stations such as restart service, reload data feeder. Web services are web server that provides user interface and virtualization to users; however, Open Standard API, proprietary API and 3rd application connectors are also run on top of this service. Apart from main services, Spatial Database (PostgreSQL with PostGIS), Arbitrary Processing Services and spatial query are provided for spatial data support. Spatial Database connects with cloud services to retrieve analyzed data or processed data. Arbitrary Processing Services is custom services that use for specific calculation and application such as interpolation process, data assimilation.

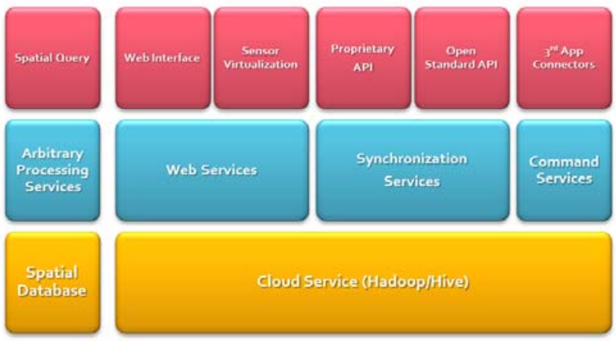

Figure 4 Sensor Cloud Services Diagram

4. CONCLUSIONS

The objective of this research is to design Enterprise Sensor Network that address current issues in development of sensor network such as handling large number of sensor node and sensor data, real-time data processing, flexible data communication, easy integration and installation. We purposed Messaging Service and Hadoop distributed platform to overcome problems in support any network infrastructure and large-scale support respectively. On sensor station side, we designed the system as services. Web server and SOS service are separated and communicate each other via RMI. SOS service is a combination of several services to handle specific operation such as SOS interface, Command Service, Scheduler Service, Data Synchronization Service and Data Feeder Service. Data Feeder Service was designed to be able to develop custom feeder for vender-specific sensor and can plug to the services. Finally, a combination of Sensor Station, Messaging Service and Sensor Cloud Service become fully functional sensor network that archive Real-time, Scalability and Robustness.

4.1 Future Work

Since it is still at early stage of research and development, we are planning to prove the concept and experiment by using simulation method. Cloud services such as Amazon EC2 will be used for simulating mobile sensor nodes and use GPS position of mobile phone combined with sensors for testing. Also, we are planning to setup messaging server with cluster feature by utilize Virtual Server from Amazon which support virtual server in different region. Finally, we will setup Sensor Cloud Services at our laboratory with full-scale mode Hadoop service and test full operations.

REFERENCES

Ahn, W., Kim, K., Yoo, S., 2010. Study on robustness middleware using integrating Sensor Observation Service in Sensor Web Enablement. In: proceeding of The 12th International Conference on Advanced Communication Technology (ICACT), Korea, Vol.II, pp. 476-479.

Honda, K., Shrestha, A., Witayangkurn, A., Chinnachodteeranun R., Shimamura, H., 2009. Fieldservers and sensor service grid as real-time monitoring infrastructure for ubiquitous sensor networks. Sensors, pp. 2371-2388.

Lam, C., 2011. Hadoop in Action. Manning, Connecticut, pp. 17-19.

McFerren, G., Hohls, D., Fleming, G., Sutton, T., 2009. Evaluating Sensor Observation Service implementations. In: proceeding of Geoscience and Remote Sensing Symposium, South Africa, Vol.5, pp. 363-366.

The Hadoop Project, http://hadoop.apache.org, 2008.

Xiaoqiang, Y., Yuejin D., 2010. Exploration of cloud computing technologies for geographic information. In: proceeding of The 18th International Conference on Geoinformatics, Beijing, pp. 1-5.

Yang, J., Wu, S., 2010. Studies on Application of Cloud Computing Techniques in GIS. In: proceeding of The International Conference on Geoscience and Remote Sensing, China, Vol.2, pp. 492-495.