iGEMS: A Cloud Green Energy Management System in Data Center

Chao-Tung Yang[1,2(✉)], Yin-Zhen Yan[1], Shuo-Tsung Chen[1],
Ren-Hao Liu[1], Jean-Huei Ou[2], and Kun-Liang Chen[2]

[1] Department of Computer Science, Tunghai University,
Taichung 40704, Taiwan
{ctyang, shough33}@thu.edu.tw,
r541754175@gmail.com,
ryan125125@gmail.com

[2] Computer Center, Tunghai University, Taichung 40704, Taiwan
{randall, kala}@thu.edu.tw

Abstract. Today the growing demand for reducing the power is not limited to household electricity saving. For businesses, it is the more important issue to effectively reduce the cost of electricity and the excess consumption under the huge electricity. In order to achieve energy saving and energy requires, the development of energy monitoring systems to obtain information related to consumption is necessary. Accordingly, this work proposes a cloud green energy management system. Because of the data size and the computational efficiency of data analysis, we add the big data technology and cloud computing to upgrade the system performance. By building cloud infrastructure and distributed storage cluster, we adopt the open source, Hadoop, to implement the two main functions: storage and computation. Based on these two functions, the proposed system speeds up the analysis and processing of big data by using Hadoop MapReduce to access HBase. The systemic risk is thus reduced too. Both real-time data and historical data are analyzed to obtain electricity consumption behavior for real-time warning and early warning. Moreover, carbon reduction and environmental protection are also considered in the analysis. Finally, a virtualized user-interface is designed to show the proposed system functions and analysis results. The experimental results indicate the performance of the proposed system.

Keywords: Cloud computing · Power management · Energy saving · Big Data · Hadoop MapReduce

1 Introduction

How to save electricity costs has been a very important issue today. Not just in the family, for businesses this issue is more important, particularly in the manufacturing-related industries. Manufacturing now has high automation by using a lot of automated

This work is supported in part by the Ministry of Science and Technology, Taiwan, under grants number MOST 104-2221-E-029-010-MY3, MOST 104-2622-E-029-008-CC3, and MOST 103-2622-E-029-012-CC3.

X. Huang et al. (Eds.): GPC 2016, LNCS 9663, pp. 82–98, 2016.
DOI: 10.1007/978-3-319-39077-2_6

manufacturing machines. However, the manufacturing machine consumes energy so that the cost of electricity costs a big proportion for manufacturing industries account. To be able to control and manager electricity costs, we must clearly and completely understand electricity. Therefore, power monitoring is a very important issue. Many energy monitoring or energy management systems have been proposed in recent years. In addition to providing real-time monitoring, these systems provide data recording function.

Han and Lim [1] suggested a Smart Home Energy Management System (SHEMS) based on IEEE802.15.4 and ZigBee. The proposed smart home energy management system divides and assigns various home network tasks to appropriate components. It can integrate diversified physical sensing information and control various consumer home devices, with the support of active sensor networks having both sensor and actuator components by using Disjoint Multi Path based Routing. Park et al. [3] proposed a Smart Energy Management System (SEMS) which functions as a control using a motion sensor and setting time of power usage to reduce power consumption. The SEMS not only supplies power as the way the common power strips do but also controls sockets of the SEMS using ZigBee wireless communication. Yardi [4] designed a Home Energy Management (HEM) system comprises an HEM unit that provides monitoring and control functionalities for a homeowner, and load controllers that gather electrical consumption data from selected appliances and perform local control based on command signals from the HEM system. A gateway, such as a smart meter, can be used to provide an interface between a utility and the data base for the electrical consumption is also maintained through internet.

According to assessment of the necessary conditions energy based on Leadership in Energy and Environmental Design (LEED), using above 300RT of the central air-conditioning chiller systems in IT room, PUE value must be less than 1.52 which can meet the basic threshold of LEED. In contrast, the domestic IT room PUE average actual measurement results above about 1.9, shows that IT room with more than 30 % energy-saving space. If it can integrate the use of international technology, design techniques and related standards, the new domestic IT room should be able to achieve the carbon reduction targets of LEED standard [5, 6].

However, most systems only monitor the rough region to estimate total electricity consumption so that they do not measure the detail electricity consumption for each device and analyze the big historical data. Accordingly, this work apply the cloud infrastructure and virtualization technology to develop a cloud green energy management system. The proposed system has many useful functions including: historical electricity data analysis, single device abnormal warning, system electricity abnormal warning, estimation of pipeline failure by using tree-network electricity layer checking, power restoration, power adjustment in contract capacity of electricity, prediction for machine breakdown and renewal. In order to have these functions in the proposed cloud green energy management system, we first adopt electric sensor, namely WPM-100 Wireless Multifunction Power Meter, to collect data including electric voltage, electric current, power factor, machine status, etc. Next, we adopt the open source Hadoop [8–12] consisting of many components including MapReduce [11, 13] and HBase [9, 10, 13–16, 22] to build a decentralized architecture to efficiently store and assess the big data including real-time data and historical data. Then, both real-time data and historical data are analyzed to obtain electricity consumption behavior for real-time warning and early warning. Moreover, carbon reduction and environmental protection are also considered in the analysis. Finally, a graphical user-interface is designed by graphical presentation to facilitate data analysis including Power Usage Effectiveness (PUE), power consumption, and so on. For a demonstration, we implement the proposed system in IT room of our school. The experimental results indicate the performance of the proposed system.

The rest of this work is as follows. Section 2 gives background and some preliminaries. In Sect. 3, we introduce the proposed system design and implementation. Section 4 shows our experiment environment and results. In Sect. 5, some conclusions are given.

2 Background and Preliminaries

This section reviews some background knowledge and preliminaries [17–21] including Hadoop, HDFS, HBase, MapReduce. We will implement the proposed system based on these background knowledge and preliminaries in next section.

2.1 Hadoop

Hadoop is an open source project under the Apache Software Foundation. The initial prototype of Hadoop-Nutch was developed for web searching by Doug Cutting and Mike Cafarella. In 2006, Doug Cutting joined Yahoo and set up a professional team to continue research and development of this technology, officially named as Hadoop. Hadoop is written in java; it can provide a distributed computing environment for huge data. The Apache Hadoop software library is a framework that allows for the distributed processing of large data sets across clusters of computers using simple programming models. It is designed to scale up from single servers to thousands of machines, each offering local computation and storage. Rather than rely on hardware to deliver high-availability, the library itself is designed to detect and handle failures at the application layer, so delivering a highly-available service on top of a cluster of computers, each of which may be prone to failures. The project includes following modules:

- Hadoop Common: The common utilities that support the other Hadoop modules.
- Hadoop Distributed File System (HDFS): A distributed file system that provides high-throughput access to application data.
- Hadoop YARN: A framework for job scheduling and cluster resource management.
- Hadoop MapReduce: A YARN-based system programming model for massive data processing of large data.

2.2 HDFS

The Hadoop Distributed File System (HDFS) is a distributed file system designed to run on commodity hardware. It has many similarities with existing 1distributed file systems. However, the differences from other distributed file systems are significant. HDFS is highly fault-tolerant and is designed to be deployed on low-cost hardware. HDFS provides high throughput access to application data and is suitable for applications that have large data sets. HDFS relaxes a few POSIX requirements to enable streaming access to file system data. HDFS was originally built as infrastructure for the Apache Nutch web search engine project. HDFS is now an Apache Hadoop subproject. HDFS has a master/slave architecture.

An HDFS cluster consists of a single NameNode, a master server that manages the file system names-pace and regulates access to files by clients. In addition, there are a number of DataNodes, usually one per node in the cluster, which manage storage attached to the nodes that they run on. HDFS exposes a file system names-pace and allows user data to be stored in files. Internally, a file is split into one or more blocks and these blocks are stored in a set of DataNodes. The NameNode executes file system namespace operations like opening, closing, and renaming files and directories. It also determines the mapping of blocks to DataNodes. The DataNodes are responsible for serving read and write requests from the file systems clients. The DataNodes also perform block creation, deletion, and replication upon instruction from the NameNode. Figure 1 shows the architecture of HDFS.

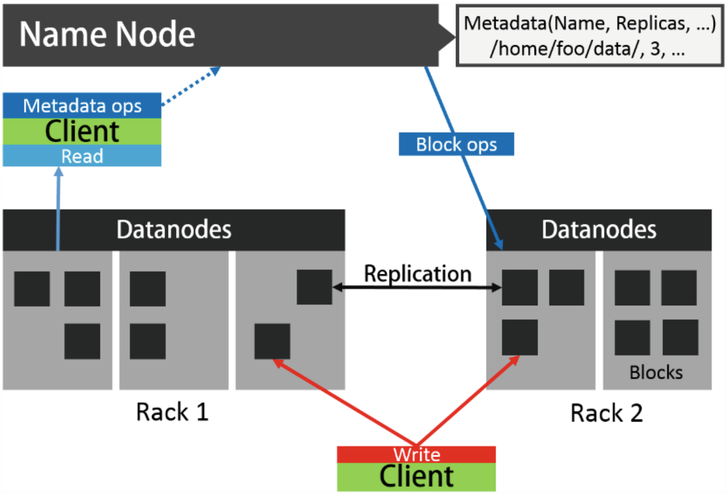

Fig. 1. HDFS architecture.

2.3 HBase

HBase, is an open source, high reliability, high performance, scalable, and not based on the relational model based on distributed repository for storing large scale unstructured data.

HBase is a distributed, versioned, non-relational database (i.e., NoSQL) modeled after Google's Bigtable: A Distributed Storage System for Structured Data. Just as Bigtable leverages the distributed data storage provided by the Google File System, Apache HBase provides Bigtable-like capabilities on top of Hadoop and HDFS. HBase data is actually stored in HDFS. Figure 2 shows the different components of HBase, and how it works with existing systems. The features of HBase are listed in the following:

- Strongly consistent reads/writes: HBase is not eventual consistent data storage; thus, it is ideal for the task of gathering, such as for high-speed counter.
- Automatic sharing: HBase tables are distributed over the cluster through regions that can be automatic separated and allocated as data grow.
- Hadoop/HDFS Integration: HBase is in support of HDFS as its distributed file system.
- MapReduce: HBase supports massively parallel processing through MapReduce by using HBase as sources and sinks.
- Java Client API: HBase supports easy-to-use Java APIs for programming.
- Thrift/REST API: HBase supports REST and Thrift as non-Java front ends.
- Block Cache and Bloom Filters: HBase supports block caches and Bloom filters for high-capacity query optimization.
- Operational Management: HBase provides insight into the operation and JMX metrics by the built-in web page.

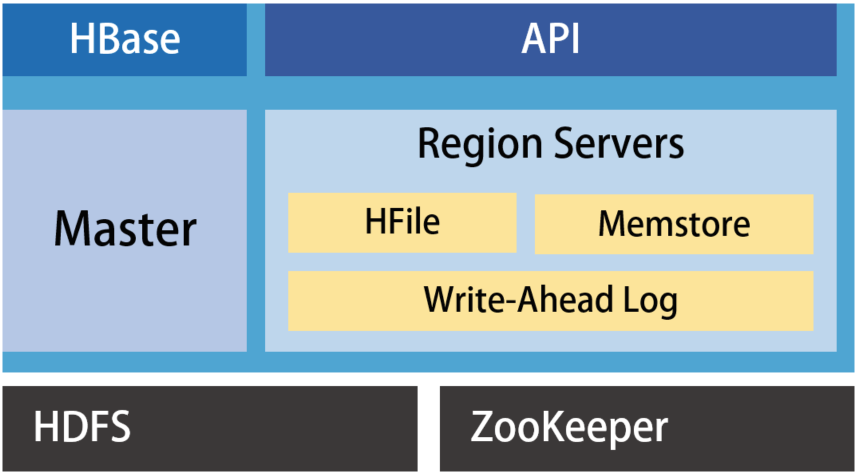

Fig. 2. The components of HBase.

3 System Design and Implementation

This section introduces the design of the proposed cloud green energy management system and its implementation. We first introduce the proposed system architecture design and then present the system implementation.

3.1 System Architecture Design

The proposed cloud green energy management system has three phases: Cloud infrastructure, Cloud platforms, Cloud services. Cloud infrastructure is the basis of the proposed system. It consists of the deployment and management of virtual machines which can be used to increase the scalability of the proposed system [23]. Based on this infrastructure, both real-time data and historical data collected from electric meters are collected and processed to obtain electricity consumption behavior for real-time warning and early warning. In addition, carbon emission and environmental data are also collected and stored. All these data will form big data. To store and process the big data, some high-availability platforms including Big Data Distributed Database Platform and Cloud Computing Platform are established based on this infrastructure. Four cloud services are provided for users in our system. They are Data Collection Service, Data Monitoring Service, Data Analysis Service, and Web Applications Service. In order to increase the feasibility of our system, we adopt open source to implement above three phase as shown in Fig. 3.

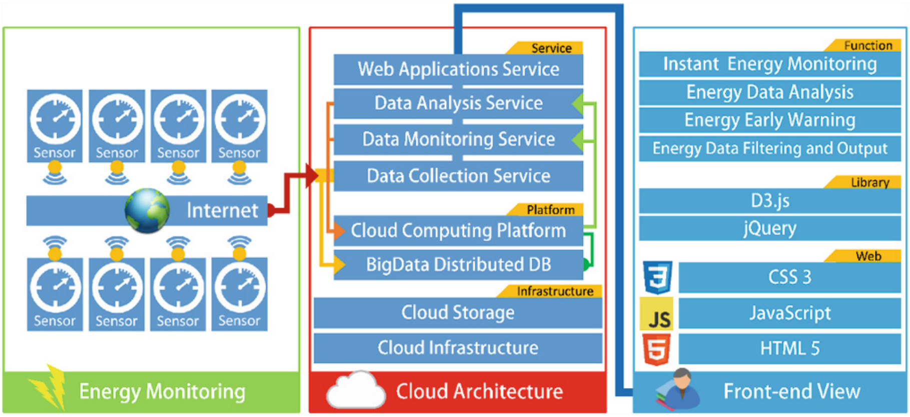

Fig. 3. Architecture design of the proposed system.

3.2 System Implementation

The first step to implement our system is the deployment of sensors. Without wiring, we adopt the electric sensor, namely WPM-100 Wireless Multifunction Power Meter, to collect data including voltage, electric current, power, etc. These collected data are then transmitted to the proposed Big Data Distributed Database in our cloud system. The format and amount of the collected data is listed in Table 1. It is expected that the data will be big data in the future since each sensor transmit 120 Byte data every minute. To store and process big data, we design a cloud infrastructure as shown in Fig. 4. In order to have better environmental data monitoring, data storage, and data

Table 1. The format and amount of the collected data

Name	Data example
meter_id	13243
location	L040
tool_id	HC705
ch1_pf	0.807
ch1_voltage	121.6
ch1_current	10.343
ch1_watt	1015
ch2_pf	0.878
ch2_voltage	122.2
ch2_current	9.786
ch2_watt	1050
ch3_pf	0.88
ch3_voltage	121.8
ch3_current	11.466
ch3_watt	1229
total_watt	3294
total_kwm	0.055998
state	PRD-PROD
time_stamp	2014-02-01 12:00
voltage12	211.2
voltage23	211.4
voltage31	210.9

processing in our system, in this cloud infrastructure we initially create several physical machines and 1 Gb/s networking speed to be a cluster. The operation system is Linux. For storing, we adopt Hbase with HDFS in Hadoop. For storage and computation, we adopt MapReduce in Hadoop.

In addition, four services, Data Collection Service, Data Monitoring Service, Data Analysis Service, and Web Applications Service are provided for users in our system. These four services are introduced as follows. As shown in Fig. 5, Data Collection Service is a Java Web Service which receives and checks the sensor data through Java. If the sensor data is from our sensor, then they will be put into Hbase by using ZooKeeper and then distributed in physical machines by HDFS. Data Monitoring Service filters and operates the collected data according to user's request while Data Analysis Service analyzes the collected data according to user's request. Figures 6 and 7 show the detail of these two services. Finally, Web Applications Service shows the results obtained from Data Monitoring Service and Data Analysis Service.

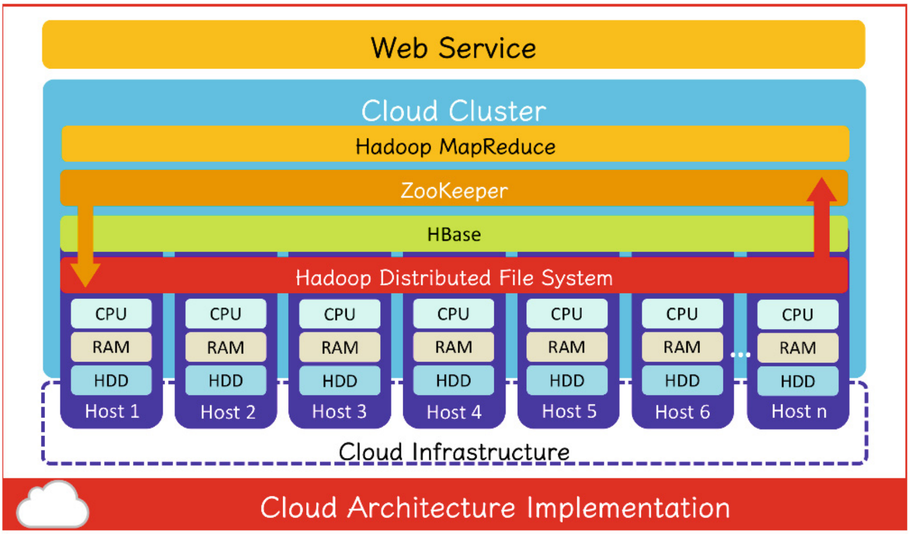

Fig. 4. Architecture of cloud.

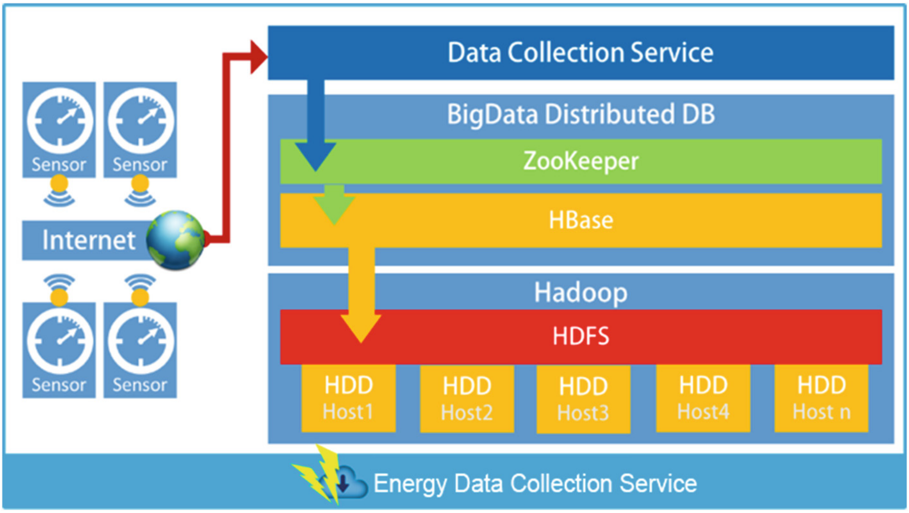

Fig. 5. Data collection service.

4 Experimental Environment and Results

This section first presents the experimental environment for the proposed system design. Next, we implement the proposed energy management system and obtain several useful experimental results based on this experimental environment.

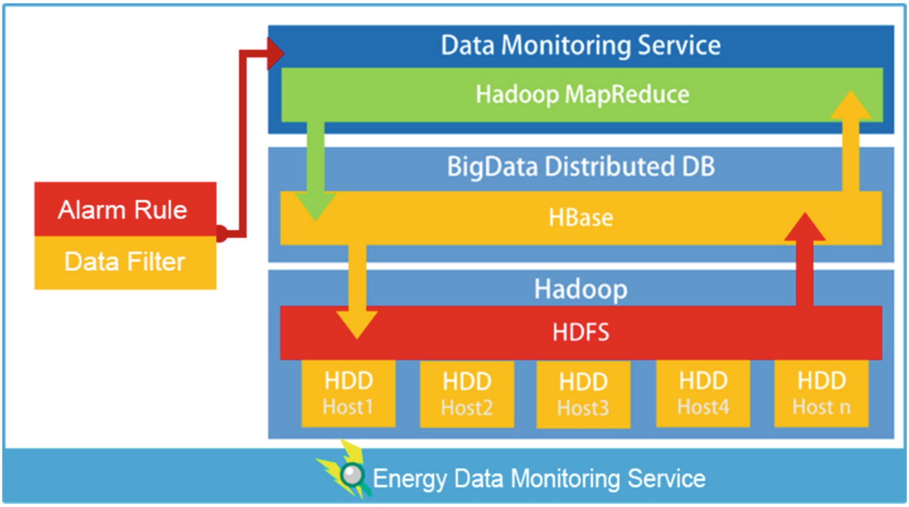

Fig. 6. Data monitoring service.

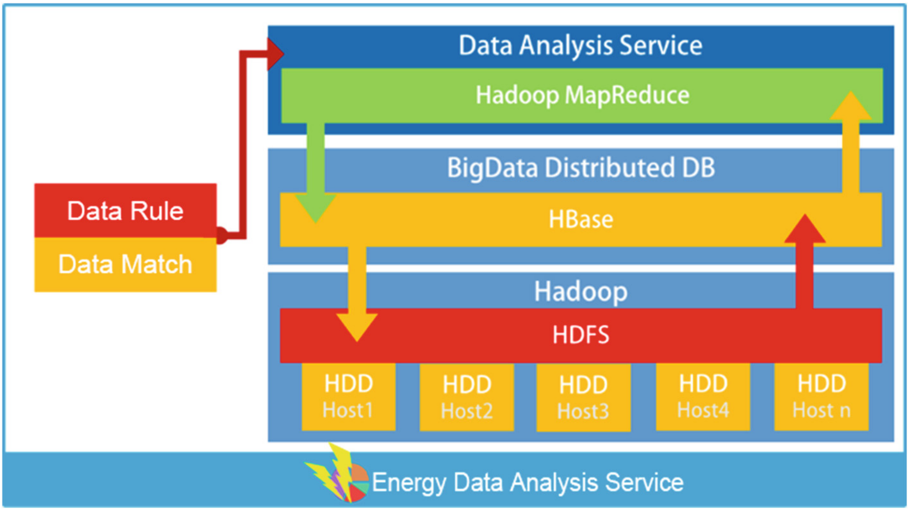

Fig. 7. Data analysis service.

4.1 Experimental Environment

To implement the proposed system, our experimental environment is introduced as follows. We adopt Intel Xeon 8-core E5-2640 V2@2.0 GHz as the CPU of the physical machine. Ubuntu 12.04 with 64 bit is adopted as our operating system. In addition, we use the electrical sensor WPM-100 Wireless Multifunction Power Meter.

4.2 Experimental Results

In this section, we present three results of our power-saving concept. First, we show the function which provides the real-time information of all machines and the air conditioning in IT room. This function is shown on an interface so that users can monitor all information we obtained. Second, PUE is calculated to help us have an energy-saving basis. Finally, we develop an interface function for historical information monitoring.

4.3 Real-Time Information of Electricity

For the proposed system performance, a monitoring platform for environmental status, air conditioning and power consumption is first established. Users can survey the real-time information related to environmental status, air conditioning, and power consumption obtained from each machine. Accessing environmental information is to make the system a better way for power saving. When user is connected to the home page, the view of the monitoring function is shown as Fig. 8. Moreover, users can observe the detail information of voltage, current, and others. In addition, the information updates automatically per second so that one can see the proportion of server power consumption and air conditioning, as shown in Fig. 9.

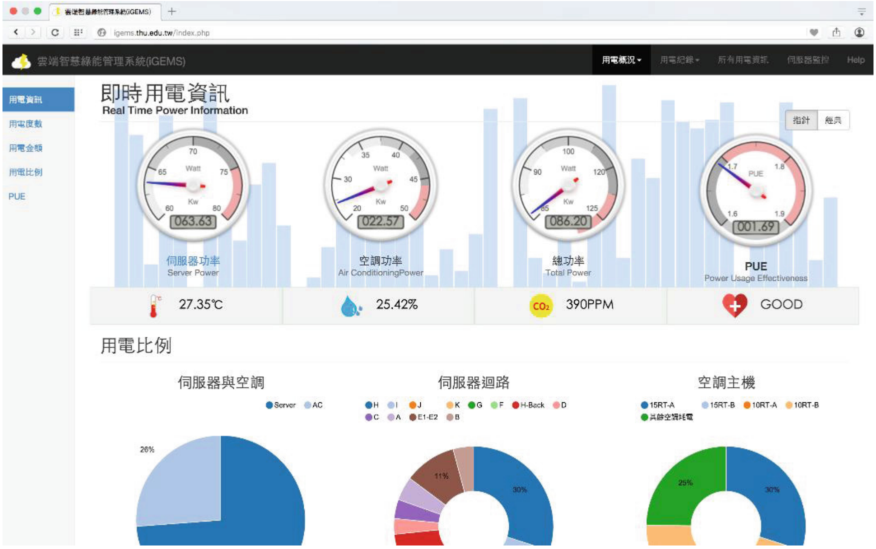

Fig. 8. Homepage of iGEMS service

4.3.1 PUE Calculation

The main server room power consumption required for the operation from the server power consumption, energy-consuming air conditioning system for cooling, power

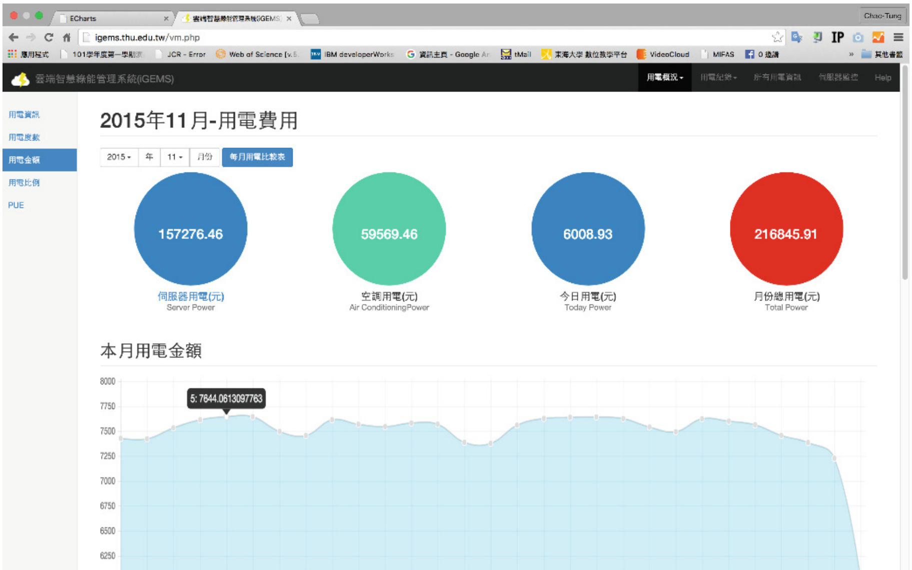

Fig. 9. The proportion of server power consumption and air conditioning

network switching equipment, and network management devices, etc., as shown in Fig. 10. For excellent energy efficiency in the server room, the energy consumption for the operation of most all servers, we calculate the PUE to help us make energy-saving basis. This study will use Power Usage Effectiveness (PUE) as energy efficiency targets of the engine room. Reduce PUE value to assist achieve energy-saving server room. In international, the average PUE is between 1.8 and 2.1.

Definition of PUE

- *Totale:* The total energy consumption room
- *ITe:* IT equipment energy consumption for each machine
- *ACs:* Air-conditioning system energy consumption for each machine
- *Ls:* Lighting system energy consumption for each machine
- *Ec:* Energy power conversion for each machine

$$PUE = \frac{Totale}{\sum ITe} = \frac{\sum ITe + \sum ACs + \sum Ls + \sum Ec}{\sum ITe}. \qquad (1)$$

4.3.2 Observation of Historical Information

We can show the interface of our power environment monitoring for historical data information. In Fig. 11, through the exploration of the historical data, we can record the amount of the monthly server power, air conditioning, environmental status, and power

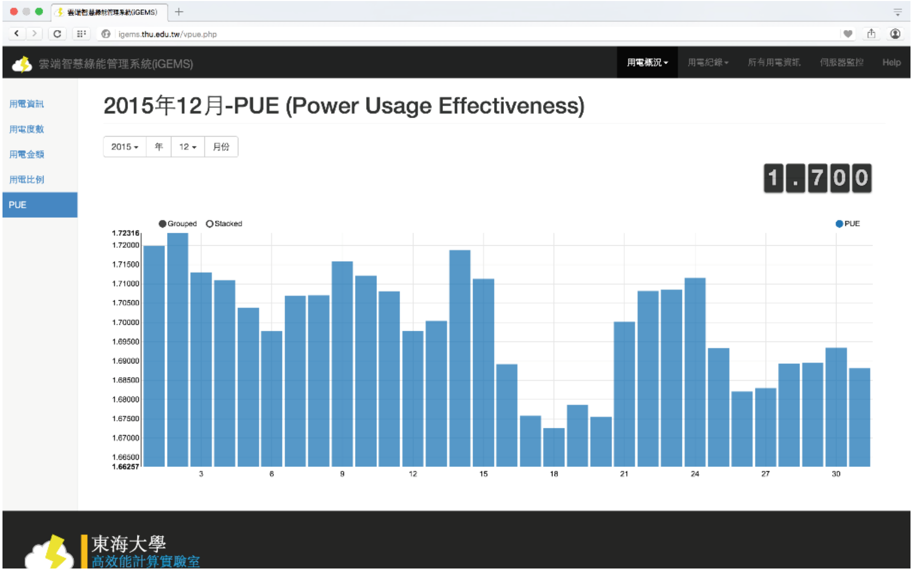

Fig. 10. PUE calculation results on interface

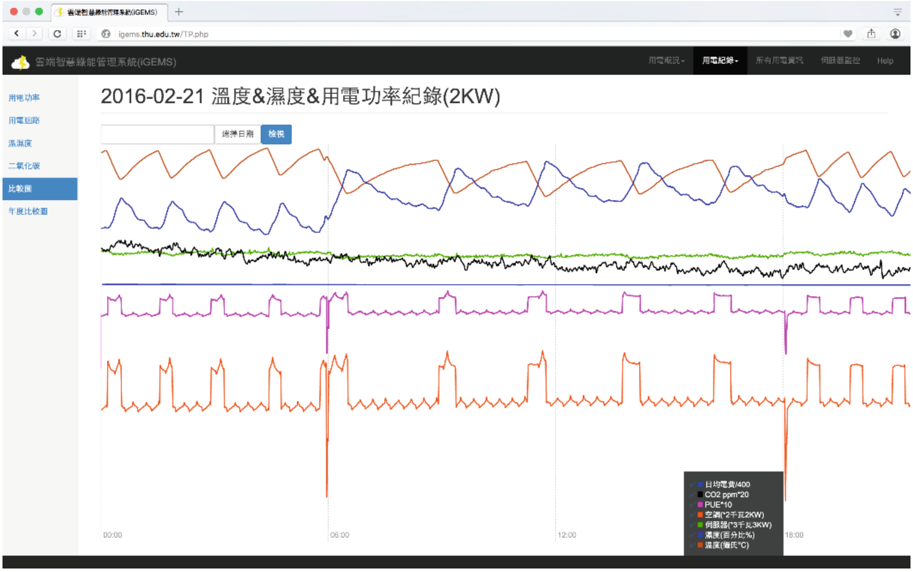

Fig. 11. Comparison of various historical data.

consumption each month. Due to the fact that our IT server is not closed all day long, we can predict the electric power changes this month. The data will update and calculate the average electricity costs each time when it gets a new data. It is shown in Fig. 12.

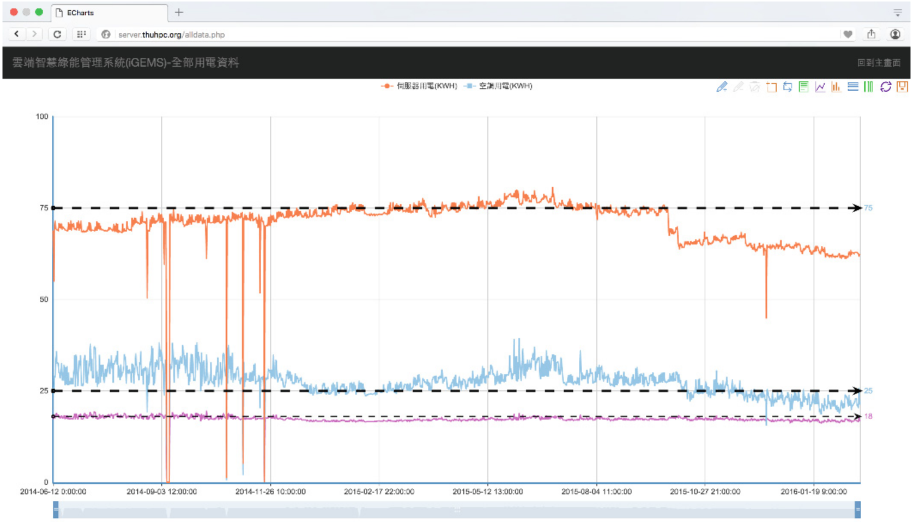

Fig. 12. Comparison of electrical data.

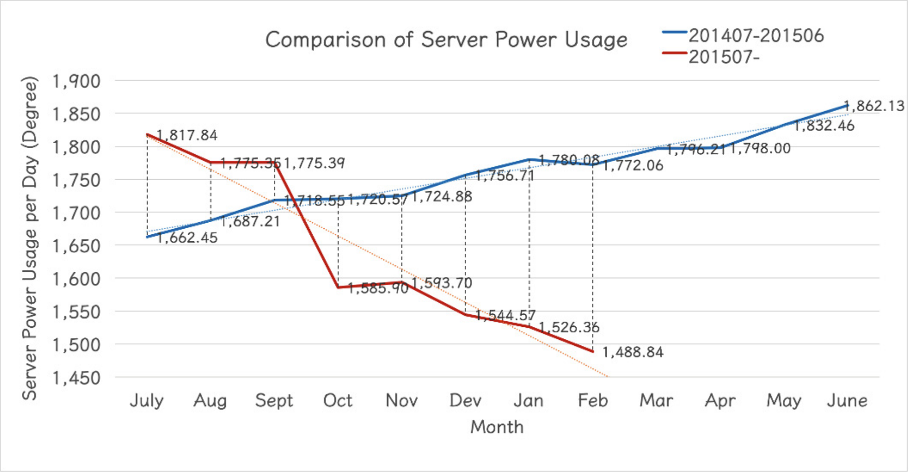

Fig. 13. Improvement of power consumption for server (Color figure online)

By presenting server location and providing each server's status including operation loading, networking loading, etc., one can observe the relation between any two factors and conclude some results to adjust system. For example, based on the analysis

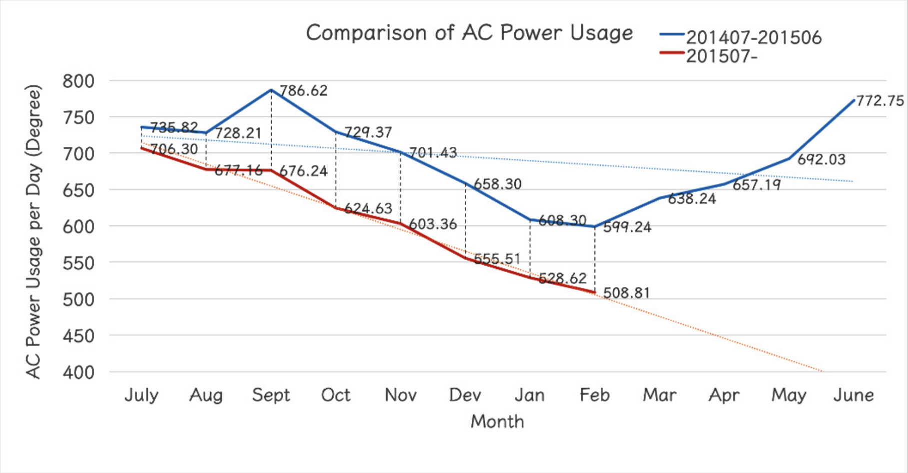

Fig. 14. Improvement of power consumption for air conditioner (Color figure online)

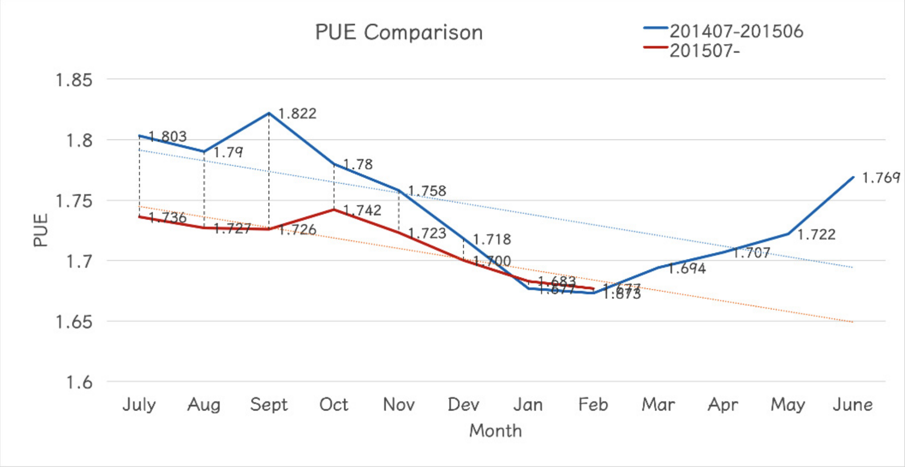

Fig. 15. Improvement of power consumption for PUE. (Color figure online)

results of the historical data, we know the proper location of air conditioners and find out the servers with high power consumption. Thus, we move the air conditioners to the location close to servers and change the servers with high power consumption to VMware virtual platforms so as to obviously reduce PUE and power consumption, as shown in Figs. 13, 14, 15, 16 and 17.

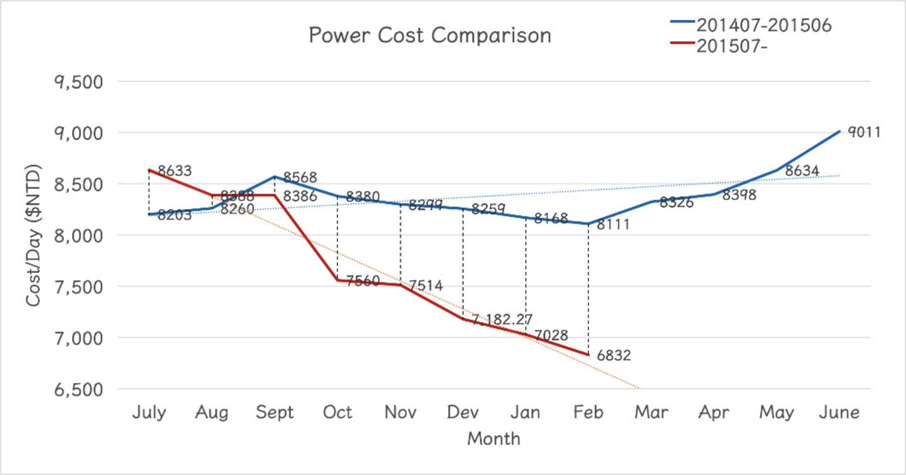

Fig. 16. Improvement of total power cost per day for each month. (Color figure online)

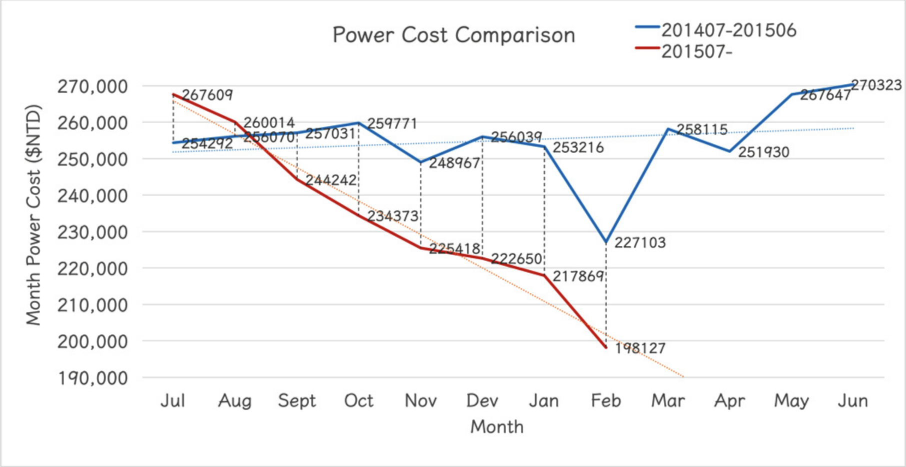

Fig. 17. Improvement of total power consumption cost for each month. (Color figure online)

5 Conclusions

This work applied the cloud infrastructure and virtualization technology to develop a cloud green energy management system. The proposed system has many useful functions including: historical electricity data analysis, single device abnormal warning, system electricity abnormal warning, estimation of pipeline failure by using tree-network electricity layer checking, power restoration, power adjustment in contract capacity of electricity, prediction for machine breakdown and renewal. The analysis of all information allows us to understand the factors relative to power consumption of IT room. Moreover, it helps us to know how to reduce power consumption.

Acknowledgements. This work is supported in part by the Ministry of Science and Technology, Taiwan ROC, under grants number MOST 104-2221-E-029-010-MY3, MOST 104-2622-E-029-008-CC3, and MOST 103-2622-E-029-012-CC3.

References

1. Han, D.-M., Lim, J.-H.: Design and implementation of smart home energy management systems based on ZigBee. IEEE Trans. Consum. Electron. **56**, 1417–1425 (2010)
2. Huang, L.-C., Chang, H.-C., Chen, C.-C., Kuo, C.-C.: A ZigBee-based monitoring and protection system for building electrical safety. Energ. Build. **43**, 1418–1426 (2011)
3. Park, S., Choi, M.-i., Kang, B., Park, S.: Design and implementation of smart energy management system for reducing power consumption using ZigBee wireless communication module. Procedia Comput. Sci. **19**, 662–668 (2013)
4. Yardi, V.S.: Design of smart home energy management system. Int. J. Innovative Res. Comput. Commun. Eng. **3**(3), 1851–1857 (2013)
5. Park, W.-K., Choi, C.-s., Jang, J.: Energy efficient multi-function home gateway in always-on home environment. IEEE Trans. Consum. Electron. **56**, 106–111 (2010)
6. Yang, I.-K., Jung, N.-J., Kim, Y-I.: Status of advanced metering infrastructure development in Korea. In: Korea Electric Power Research Institute IEEE T&D, pp. 1–3, Asia (2009)
7. Han, J., Choi, C.-S., Park, W.-K., Lee, I., Kim, S.-H.: Smart home energy management system including renewable energy based on ZigBee and PLC. IEEE Trans. Consum. Electron. **60**(2), 198–202 (2014)
8. O'Driscoll, A., Daugelaite, J., Sleator, R.D.: 'Big data', Hadoop and cloud computing in genomics. J. Biomed. Inform. **46**(5), 774–781 (2013)
9. Taylor, R.C.: An overview of the Hadoop/MapReduce/HBase framework and its current applications in bioinformatics. BMC Bioinform. **11**(Suppl. 1), S1 (2010)
10. QIU, Z., Lin, Z.-w., Ma, Y.: Research of Hadoop-based data flow management system. J. China Univ. Posts Telecommun. **18**(Suppl. 2), 164–168 (2011)
11. Dittrich, J., Quian, J.A.: Efficient big data processing in Hadoop MapReduce. Proc. VLDB Endowment **5**(12), 2014–2015 (2012)
12. O'Driscoll, A., Daugelaite, J., Sleator, R.D.: 'Big data', Hadoop and cloud computing in genomics. J. Biomed. Inform. **46**(5), 774–781 (2013)
13. Taylor, R.C.: An overview of the Hadoop/MapReduce/HBase framework and its current applications in bioinformatics. BMC Bioinform. **11**(Suppl. 12), S1 (2010)
14. Zhang, C., De Sterck, H.: Supporting multi-row distributed transactions with global snapshot isolation using bare-bones HBase. In: 2010 11th IEEE/ACM International Conference on Grid Computing, pp. 177–184 (2010)
15. Vashishtha, H., Stroulia, E.: Enhancing query support in HBase via an extended coprocessors framework. In: Abramowicz, W., Llorente, I.M., Surridge, M., Zisman, A., Vayssière, J. (eds.) ServiceWave 2011. LNCS, vol. 6994, pp. 75–87. Springer, Heidelberg (2011)
16. Sun, J., Jin, Q.: Scalable RDF store based on HBase and MapReduce. In: ICACTE (2010) 3rd International Conference on Advanced Computer Theory and Engineering, vol. 1 (2010)

17. Yang, C.-T., Liao, C.-J., Liu, J.-C., Den, W., Chou, Y.-C., Tsai, J.-J.: Construction and application of an intelligent air quality monitoring system for healthcare environment. J. Med. Syst. **38**(2), 15 (2014)
18. Yang, C.-T., Shih, W.-C., Chen, L.-T., Kuo, C.-T., Jiang, F.-C., Leu, F.-Y.: Accessing medical image file with co-allocation HDFS in cloud. Future Gener. Comput. Syst. **43–44**, 61–73 (2015)
19. The Hadoop distributed file system: architecture and design (2007). http://hadoop.apache.org/docs/r0.18.0/hdfs_design.pdf
20. Hadoop. http://hadoop.apache.org/
21. Vora, M.N.: Hadoop-HBase for large-scale data. In: 2011 International Conference on Computer Science and Network Technology (ICCSNT), vol. 1, pp. 601–605 (2011)
22. Yang, C.-T., Shih, W.-C., Huang, C.-L., Jiang, F.-C., Chu, William, C.C.: On construction of a distributed data storage system in cloud. Computing **98**(1–2), 93–118 (2016)
23. Yang, C.-T., Liu, J.-C., Huang, K.-L., Jiang, F.-C.: A method for managing green power of a virtual machine cluster in cloud. Future Gener. Comp. Syst. **37**, 26–36 (2014)