Journal of Engineering and Applied Sciences 12 (Special Issue 2): 6304-6307, 2017
ISSN: 1816-949X

Implementation of an Efficient Big Data Collection Platform for Smart Manufacturing

[1]Hyeopgeon Lee, [1]Young-Woon Kim and [2]Ki-Young Kim
[1]Department of Data Analysis, Seoul Gangseo Campus, Korea Polytechnic University, Seoul, Korea
[2]Department of Computer Software, Seoul University, Seoul, Republic of Korea

Abstract: Smart manufacturing uses various equipment ranging from information systems to small sensors with network functions and the type of data collected is also diverse. As a result, there are many studies on data collection techniques that can collect data from each type of equipment. Thus, this study will implement an effective big data collection platform for smart manufacturing. The proposed big data collection platform was designed for collecting various types of data from different types of equipment. The proposed big data collection platform was implemented using Apache Flume and Sqoop to improve data collection and analysis performance compared with existing big data integrated collection platforms.

Key words: Big data, big data collection platform, smart manufacturing, analysis, performance, effective

INTRODUCTION

Smart manufacturing (Apache Flume) is a paradigm that promotes the strategic innovation of existing manufacturing techniques by integrating people, technology and information and is the embodiment of a fusion between advanced ICT and manufacturing techniques. Advanced manufacturing companies in countries such as Germany and the United States are proactively conducting various studies for the development of advanced technology through connecting people, government and schools to practice industry 4.0 and smart manufacturing (ASF, 2017).

To implement this smart manufacturing, there are many studies that have applied big data technology which is one of the representative technologies of the Fourth Industrial Revolution. To apply big data technology in smart manufacturing, data collection must first be performed. Equipment that collects data in smart manufacturing are diverse from information systems to small sensors with network functions and the types of data that are collected are also varied. Therefore, the data collection technique used to collect data from this equipment is important (Apache Sqoop) (Azhari and Loh, 2016).

Big data collection platforms (Lee *et al*., 2012) that collect various types of data are called "big data integrated collection platforms" and existing big data integrated collection platforms build big data collection platforms according to the types of data that are collected. The big data collection platform acts as an agent and stores data that were collected into the big data integrated collection platform. As a result, because big data integrated collection platforms take a long time to collect and analyze data and the process of managing the stored data is so complex, difficulties arise in operating these big data platforms.

Thus, this study implemented an effective big data collection platform for smart manufacturing. The proposed big data collection platform was designed to collect various types of data from information systems to small sensors with network functions. The proposed big data collection platform was implemented using Apache Flume and Sqoop to improve data collection and data analysis performance compared with existing big data integrated collection platforms.

MATERIALS AND METHODS

Big data collection platform: Big data collection is a basic technique of big data technology and there are various big data collection platforms. This study examines the most frequently used big data collection platform based on Apache Flume, the Apache Sqoop-based big data collection platform and existing big data integrated collection platforms.

Apache flume-based collection platform: The Apache Flume-based big data collection platform by PCAST AMP Steering Committee Report-Accelerating US Advanced Manufacturing in 2014, can collect sensing data from small sensors with server, network equipment log and network functions. The Flume is a distributed, reliable and available service for efficiently collecting, aggregating and

Corresponding Author: Hyeopgeon Lee, Department of Data Analysis, Seoul Gangseo Campus, Korea Polytechnic University, Seoul, Korea

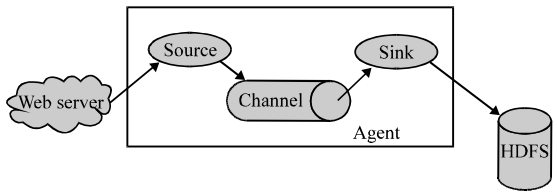

Fig. 1: The example of apache flume-based collection platform model

moving large amounts of log data. It has a simple and flexible architecture based on streaming data flows. It is robust and fault tolerant with tunable reliability mechanisms and many failover and recovery mechanisms. It uses a simple extensible data model that allows for online analytic application. Figure 1 shows the example of Apache Flume-based collection platform model.

A Flume source consumes events delivered to it by an external source like a web server. The external source sends events to Flume in a format that is recognized by the target Flume source. For example, an Avro Flume source can be used to receive Avro events from Avro clients or other Flume agents in the flow that send events from an Avro sink. A similar flow can be defined using a Thrift Flume source to receive events from a Thrift sink or a Flume Thrift RPC Client or Thrift clients written in any language generated from the Flume thrift protocol. When a Flume source receives an event, it stores it into one or more channels. The channel is a passive store that keeps the event until it's consumed by a Flume sink. The file channel is one example it is backed by the local file system. The sink removes the event from the channel and puts it into an external repository like HDFS (via. Flume HDFS sink) or forwards it to the Flume source of the next Flume agent (next hop) in the flow. The source and sink within the given agent run asynchronously with the events staged in the channel (Lee *et al*., 2012).

Apache Sqoop-based collection platform: The Apache Sqoop-based big data collection platform (Ranjan, 2014) is used to collect Relational Database Management System (RDBMS) data that are used in information systems. The Sqoop is a tool designed to transfer data between Hadoop and relational databases or mainframes. The sqoop automates most of this process, relying on the database to describe the schema for the data to be imported. Sqoop uses MapReduce to import and export the data which provides parallel operation as well as fault tolerance. Figure 2 shows the example of Apache Sqoop-based collection platform model.

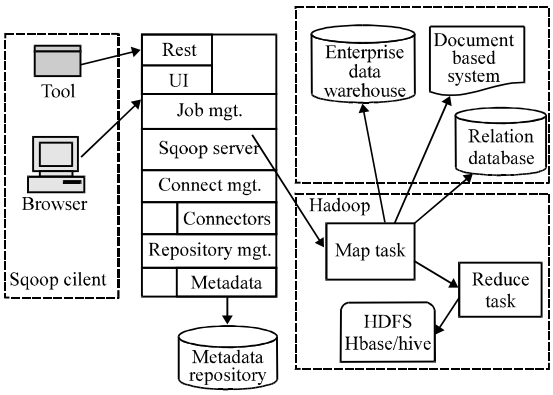

Fig. 2: The example of apache Sqoop-based collection platform model

Existing big data integrated collection platforms: Existing big data integrated collection platforms (Yan *et al*., 2016) build big data collection platforms according to the types of data that are collected and the big data collection platform that was built acts as a data collection agent and performs other similar roles. The RDBMS data and sensing data that are collected from each big data collection platform that was built are converted into a Comma-Separated Values (CSV) file and this generated file is stored in the storage of the big data integrated collection platform. The CSV file conversion procedure of the collected data requires an extremely high consumption of input and output resources. Therefore, if there is a high volume of data collection as in smart manufacturing, the amount of resource consumption for the input and output files will be significant. Moreover, to deliver the collected data to a big data integrated collection platform, there is additional network load and more resources for the input and output files are consumed. Because existing big data integrated collection platforms store the collected data without categorizing them according to type, the performance of big data analysis suffers and it is difficult to use a big data analysis query language.

Proposed big data collection platform: The proposed big data collection platform was created for effective data collection from various data collection targets from smart manufacturing such as sensing data and information systems. Figure 3 shows the proposed big data collection platform.

This proposed platform is composed of a big data collection agent, big data infrastructure layer, big data processing layer and big data storage layer. The big data collection agent collects data from various collection targets from the information systems of smart manufacturing to small sensors with network functions.

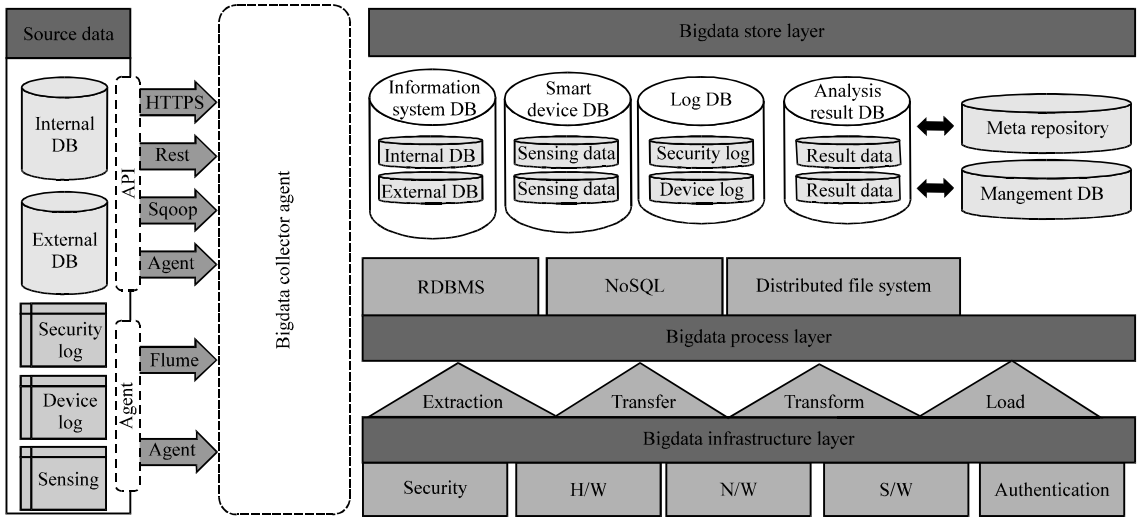

Fig. 3: The proposed big data collection platform

The internal and external databases use an Application Programming Interface (API) to collect data and apply the Representational State Transfer (REST) method based on secure HTTPS and uses Sqoop for connection. The security log, equipment log and sensing data are collected by the agent and Flume is used to collect data. The big data collection agent only collects data. It does not extract, transmit, convert or store data.

The big data infrastructure layer creates the software, hardware, network, security and authentication-related infrastructures to implement the big data collection platform. RDMS, NoSQL and distributed file system are software that use infra implementation and the hardware is installed to fit the software that are installed. Security manages authentication and other security keys.

The big data processing layer extracts, transmits, converts and stores data so that the data delivered from the big data collection agent can be systematically stored in the big data storage layer. The big data storage layer stores data according the following clusters: the information system database stores data from the internal and external information system; the equipment database stores sensor data; the log database stores security logs and equipment logs and the analysis results database stores the big data analysis results. It also uses a meta repository and an operation and management database to index the stored big data and manage cluster.

RESULTS AND DISCUSSION

Performance evaluation: This study shows that the proposed big data collection platform is more efficient than existing big data integrated collection platforms. The per-second data collection count and per-second data analysis count served as indices to analyze the performance of the proposed big data collection platform and the data collected were database data and log data with a volume of 10 kb or less.

Per-second data collection count: The per-second data collection count was measured by creating each platform and measuring the time it took from collecting to storing the data. Time was measured from 0-10 sec. Figure 4 shows the per-second data collection count.

The per-second data collection count significantly increased after 0.5 sec for both comparison targets and there was no significant increase after approximately 3.5 sec. After 2.5 sec, there was a gradual difference between the proposed big data collection platform and the existing big data integrated collection platform in terms of the data collection count. The data collection count for the existing big data integrated collection platform was 725.103 while the data collection count for the proposed big data collection platform was 784.476 which shows an 8.19% increase. The reason for this is that in the existing big data integrated collection platform, each of the big data collection platforms that were built according to data type collected data, then sent the collected results back to the storage of the big data integrated collection platform, resulting in two rounds of data collection and storage that created additional network load. There is also the additional process of converting the form of the collected data. However, the proposed big data collection platform collected data regardless of data type and stored them in the storage clusters that were categorized according to data type right away without having to convert the data.

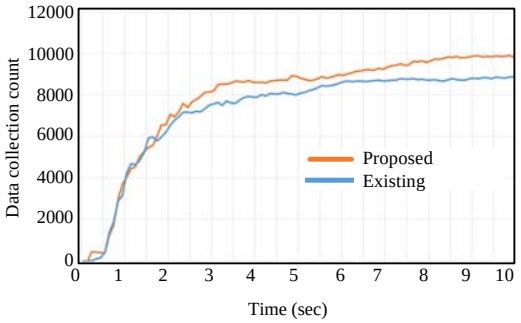

Fig. 4: The per-second data collection count

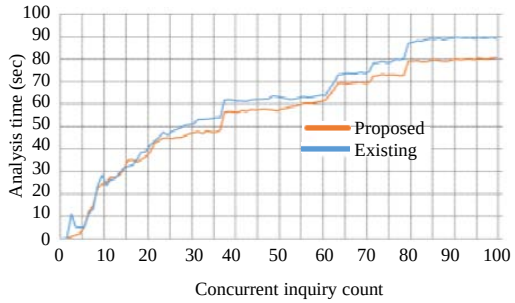

Fig. 5: The per-second data analysis count

Thus, the data collection count had increased compared with the existing big data integrated collection platform.

Per-second data analysis count: For the per-second data analysis count, Apache Spark which is one of the main techniques for real-time big data analysis was used to measure the time from data analysis inquiry to result output. Data inquiry was set from 1-100 simultaneously. Figure 5 shows the per-second data analysis count.

There was no significant difference in per-second data analysis time at 25 inquiries for the comparison targets, but there was a gradual difference between them after that point. The total data analysis time for the existing big data integrated collection platform was 59,988.41 while the proposed big data platform took a total of 5,518.81 sec which is 8.51% decrease. The reason for this result is that in the existing big data integrated collection platform, different data types were converted into the same data format then stored in one location which added the process of searching for and converting data. However, because the proposed big data collection platform stored data in categorized storage clusters according to data type, there was no need to search for data according to data type or convert data which decreased the data analysis time.

CONCLUSION

This study implemented an effective big data collection platform for smart manufacturing. The proposed big data collection platform was designed to collect various types of data from information systems to small sensors with network functions. The proposed big data collection platform was implemented using Apache Flume and Sqoop to improve data collection and data analysis performance compared with existing big data integrated collection platforms.

RECOMMENDATIONS

In the future, we plan to perform a more detailed performance analysis according to the various data types in diverse smart manufacturing environments. We also plan to provide a more realistic verification of the proposed big data collection platform by applying it to a real environment.

REFERENCES

ASF., 2017. Apache sqoop. Apache Software Foundation, Forest Hill, Maryland. http://sqoop.apache.org/

Azhari, F. and K.J. Loh, 2016. Dissolved oxygen sensors for scour monitoring. IEEE. Sens. J., 16: 8357-8358.

Lee, H., A. Kim and Y. Shin, 2012. Retransmission algorithm for channel allocation in IEEE 802.15. 4 LR-WPAN. Proceedings of the 6th International Conference on Convergence and Hybrid Information Technology (ICHIT'12), August 23-25, 2012, Springer, Daejeon, Korea, pp: 657-664.

Ranjan, R., 2014. Modeling and simulation in performance optimization of big data processing frameworks. IEEE. Cloud Comput., 1: 14-19.

Yan, L., K. Morris and F. Simon, 2016. Current standards landscape for smart manufacturing systems. Master Thesis, National Institute of Standards and Technology, Gaithersburg, Maryland.