International Journal of Pure and Applied Mathematics
Volume 117 No. 20 2017, 473-484
ISSN: 1311-8080 (printed version); ISSN: 1314-3395 (on-line version)
url: http://www.ijpam.eu
Special Issue

ijpam.eu

Web Application-based RDBMS Data Collection Framework Modeling

[1]Hoo-Young Lee, [2]Koo-Rack Park and [3]Dong-Hyun Kim

[1]Department of Computer Engineering,

Kongju National Univercity,

Chungnam Cheonan Subuk Cheonan-Dearo,

South Korea.

Hooyoung.paul.lee@gmail.com

[2]Department of Computer Science & Engineering,

Kongju National Univercity,

Chungnam Cheonan Subuk Cheonan-Daero,

South Korea.

ecgrpark@kongju.ac.kr

[3]Department of Computer Engineering,

Kongju National Univercity,

Chungnam Cheonan Subuk Cheonan-Daero,

South Korea.

dhkim977@naver.com

Abstract

Studies have been actively implemented to collect data from diverse kinds of repositories for big data analysis. This study performed the modeling of data collection framework that connects the Relational Database Management System (RDBMS) through web applications, extracts data and transfers them to servers for big data analysis.

Install a client program that transfers to RDBMS the extracted data and data extracted into a server. In the analysis server, install a server program that stores data transmitted from multiple clients to collect data. Each step of data collection, transmission and storage is controlled and monitored by a web application.

Concerning data collection framework, open source-based programs pose difficulties in use and commercially available programs are costly. The

program proposed in this study is an open source-based web application to ensure easy system expandability and maintenance. The entire processes of the proposed system are controllable and monitored based on the web, ensuring convenience in use.

The proposed system moves the data kept in RDBMS to a data analysis server. Future study is planned to collect high variety data such as photos, videos and texts in addition to the structured data in the RDBMS and find how to efficiently store them in big data analysis systems like Hadoop.

Keywords: BigData, structured data, RDBMS, data collection, spring framework.

1. Introduction

The IDC (International Data Corporation) reported in 2011 that the volume of data in the internet exceeded 12B. Big Data becomes a recent big issue in the international IT community, attracting increasing attention to related technologies[1].

Big data analysis and use become a very important issue across the overall areas in the modern society such as politics, economy, culture and medicine. In this situation, studies are actively implemented concerning big data analysis and use in diverse different areas. With the rising interest in big data analysis and use, data collection is also researched actively in studies. Open source-based programs and other multiple commercially available programs are already utilized.

Big data analysis technology is to utilized diverse forms of data such as text, photo, and video to extract valuable information. Depending upon the degree of variety, data are classified into structured data, semi-structured data and high variety data. Semi-structured data are not any kind of fixed formed data but refer to data including schema. High variety data refer to data not stored in a fixed field such as text, photo and audio. On the other hand, structured data are those stored in a fixed table in relational data based in a column form, meaning they have a certain constant form.

Apache Sqoop is one of the main open source programs collecting data in relational database, a major sort data type. However, it requires user expertise in their use as it has text-based user interface. Splunk is one of the commercially available applications in this type. But because of its license and usage expenses, smaller firms, public agencies and non-profit organizations have difficulty in using them[2].

This paper installs a client program managing data extraction and transmission from/to each database server as well as a server data collection program in data collecting servers to deliver data through socket communication via a pre-defined port. This paper proposes a data collection framework model capable of monitoring the entire processes of such data extraction and transmission using spring, the MVC pattern-based Java web application development framework. With the proposed model in place, the target data collection method was simplified in this study. By doing so, smaller companies and agencies are expected to use it in their data collection.

2. Literature Review

Big Data

Big data varies slightly according to who defines it. Gartner said that big data had a huge volume, fast speed and diversified information assets. McKinsey

defines big data as the data set in too big scale to be treated with any typical database and said big data cannot be defined in any specific volume[3,4].The technical characteristics of big data include high data volume, high variety) and high velocity[5].

In other words, the numerous kinds and forms of data undergo the procedures of collection, storage, analysis and visualization. In the processes, valuable data are extracted. It can be said to be the purpose of big data analysis and use.

Structured Data

Big data element technologies include data volume representing media or location information, video, etc.; data input/output velocity of real-time data production; and data variety on unstructured (high variety) form[6].

Structured and unstructured types are the classification based on the level of big data variety.

Structured type refers to the data stored in a fixed field (relational database). Semi-structured type refers data including schema (XML, HTML etc.), though not the fixed field.

Structured type means data not stored in a fixed field or, that is, text, image, video, etc[7].

Spring Framework

Spring framework is an open source framework for Java platform. As Java was put under limelight, Servlet emerged and Java began to spread out into web-based applications.

In step with this trend, EJB (Enterprise JavaBeans) appeared, which provides transaction, security, etc. In the situation, Java landed as an indispensable technology in building enterprise applications[8,9].

However, its development acceleration faced limitation because of the inconvenience from diverse multiple complicated code changes and test implementation in line with appropriate container distribution. That is, to test it was difficult without an appropriate container and development velocity was undermined as well. For these reasons, many developers have difficulties in progression their development. In order to ease such an inconvenience, Spring Framework emerged[10].

Spring framework is a light-volume container based on Java and allows selective use of necessary objects only to support efficient development. It is in the MVC (Model, View, Controller) structure to allow module-specific development.

The strongest point of Spring framework is its layered form of structure which provides efficient flexibility in system expansion and maintenance[11].

3. Proposed Work

System Overview

The system proposed in this paper installs a data collection client (DCC) in multiple relational data bases (RDBMS), which can collect data and transfer the collected data to servers. Through the DCC installation, the system extracts data in the file form and transmits the extracted data files to data collection server (DCS). Its collection request and each processing steps are made by a web application-based management program. Users can access the management program via a web browser and check the server list of database with client installation.

There are two data extraction methods – a table-unit data extraction method and Structured Query Language (SQL)-based data extraction method that finds out the data meeting special sets of conditions.

The following [Figure 1] shows the proposed system structure, its connection between the client program installed in each database server and collection program installed in servers. Text formed data extracted from the client program are stored in data collection servers through socket communication.

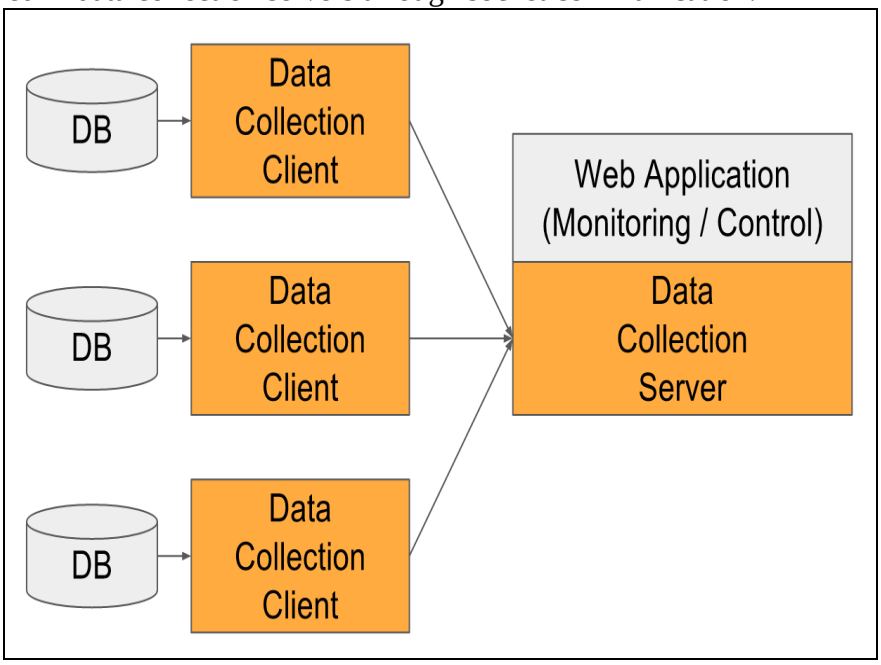

Figure 1: System Structure

Flow Chart of Data Collection and Storage Process

Data collection and storage process consists of three steps as shown in [Figure2].

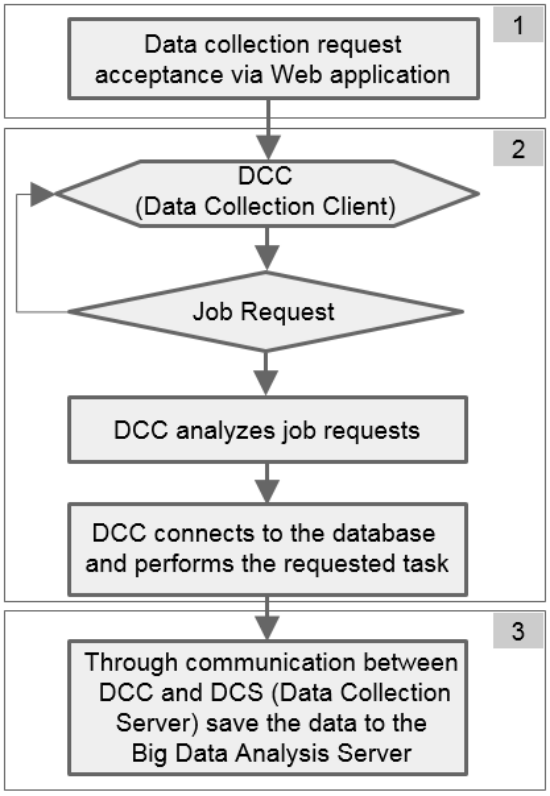

Figure 2: Flow Chart ofData Collection and Storage Process

First is the data collection request processing through the web application. Users access a target database server via the management program and extract target data by the unit of database or table or by using SQL satisfying given conditions.

Second is the data collection client stage. The collection client is installed in the demon form in server with relational database installation; communicates with server every certain time intervals; and checks if any new request is registered. In the event of finding a new request, the client program connects to the database, designates the collection target data in a repository in the route designated by the client in the UTF-8 text form, and transmits the files based on the communication with data collection server. After transmission completion, it returns information on request implementation result such as success, failure, collection data volume, etc.

Third is data collection server stage. The data collection server stays standby and checks if there is any file transmission request from the client. Upon the reception of file transmission request, the server stores corresponding file in its file system and returns its result to the management program.

Data Collection Client

The following Figure 3 is the main algorithm of data collection client. Data collection client generates thread at the designated time. Once thread is created,

it checks if there is any new job request. If there is a new data extraction work, it analyzes the job phrase and implements it. In the event of an error during implementation, it checks the Queue_SEQ request and updates the error code. After data extraction, the extracted files are sent to a server and the system records the file volume and transmission date.

```
DataCollectorClientClass {
"THREAD"(new Runnable() {
@Override
public void run() {
DATABASE Connect
IF there is a new job THEN
WHILE multiple new job
job request analysis
BEGIN
job request implementation
EXCEPTION
unexpected error while implementation
return error code
END
ENDWHILE
ENDIF
DATABASE Close

file transmission object creation
IF a file transmission request THEN
WHILE multiple file transmission
BEGIN
file transmission
EXCEPTION
unexpected error while implementation
return error code
END
ENDWHILE
ENDIF
}
}, 0, 5 , TimeUnit.SECONDS);
}
```

Figure 3: Data Collection Client

Web Application-based Management Program

Data collection request is made via web application-based management program. Users check the list of connected relational databases through the management program and choose servers to collect data from. The selected server table information is displayed. Users can choose tables to collect data from or use SQL to selectively extract data meeting a certain set of conditions.

The following Table 1 shows queue table structure. The requested work is registered with queue table of management program. Each requested work has a unique queue-SEQ value. The collection client and server implement the requested work and update the times of inspection, implementation and completion and completion code consecutively. Users can understand the work progress via queue table.

Table 1: Queue Table Structure

Column Name	Type	Description
Queue_SEQ	Integer	QueueNumber
Queue_Regist_Date	Date	Queueregistration data
Queue_Command	Varchar	users' request
Queue_Registor	Varchar	Queueregister
Queue_Flag	Char	Queuestatus
Queue_Client_Read_Date	Date	record the time when DCC reads Queue
Queue_Client_Exe_Date	Date	record the time when DCC implementsQueue
Queue_Client_Done_Date	Date	record the time when DCC completes Queue
Queue_Client_Exe_Code	Varchar	record the result of DCC's Queue implementation
Queue_Server_Read_Date	Date	record the time when DCC reads Queue
Queue_Server_Exe_Date	Date	record the time when DCC implementsQueue
Queue_Server_Done_Date	Date	record the time when DCC completes Queue
Queue_Server_Exe_Code	Varchar	record the result of DCC's Queue implementation

The data collection server stands by while opening a specific port and when a file is sensed from the client, receives the file and stores it. After storing, it checks Queue_SQE in the queue table and updates the time of reply and file size.

4. Experiment & Consideration

For the experiment of this paper, Spring (ver 3.1.1) was utilized, which is a Java-based web application development framework, in developing a web application managing data collection and transmission. For data extraction, MySQL version 5.1.41 was installed in 172.20.23.95 and client program was installed, which extracts and transmits data. For data collection, a collection server program was installed in 110.10.26.80. [Figure 4] below shows the web application structured in this study. It connected to the database installed in 172.20.23.95 server, checks the table list and selectively chosen customer table for data extraction.

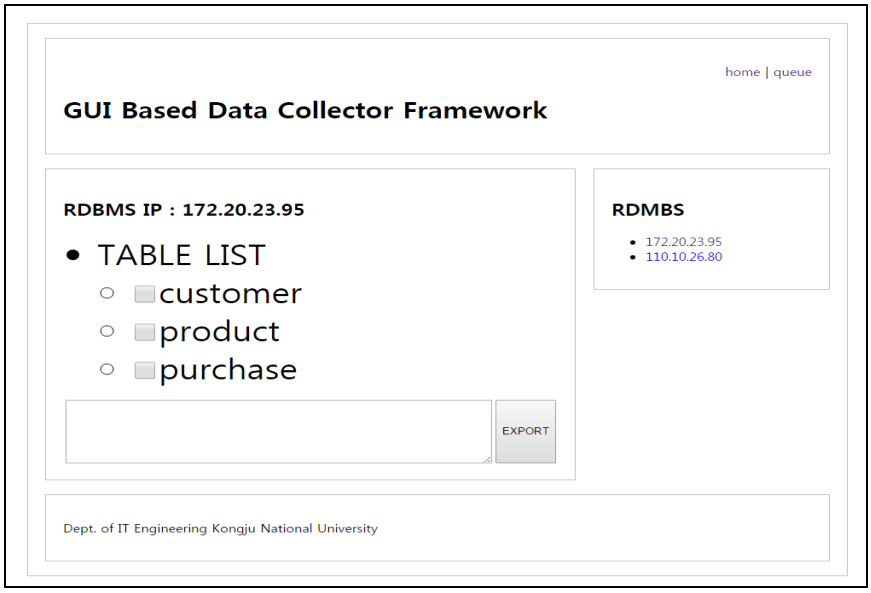

Figure 4: Accessible Relational Database Server and Its Table List

[Figure5] below is a customer table. The data extracted from table are temporarily saved in 172.20.23.95 server in the csv(comma-separated values) file form. After extraction and storage completion, it begins to transmit the file to the transmission standby server and pre-determined port and records file size and transmission time. Servers check the transmitted file from the client and records the time of transmission before the work completion.

```
1 "1";"87462024688";"Nowmer";"Sheri";"A.";"2433 Bailey
  Road";NULL;NULL;NULL;"Tlaxiaco";"Oaxaca";"15057";"Mexico";"30";"271-555-9715";"119-555-1969
  ";"1961-08-26";"M";"$30K - $50K";"F";"4";"2";"Partial High
  School";"1991-09-10";"Bronze";"Skilled Manual";"Y";"4";"Sheri Nowmer"
2 "2";"87470586299";"Whelply";"Derrick";"I.";"2219 Dewing
  Avenue";NULL;NULL;NULL;"Sooke";"BC";"17172";"Canada";"101";"211-555-7669";"807-555-9033";"1
  915-07-03";"S";"$70K - $90K";"M";"1";"0";"Partial High
  School";"1993-03-11";"Bronze";"Professional";"N";"3";"Derrick Whelply"
3 "3";"87475757600";"Derry";"Jeanne";NULL;"7640 First
  Ave.";NULL;NULL;NULL;"Issaquah";"WA";"73980";"USA";"21";"656-555-2272";"221-555-2493";"1910
  -06-21";"M";"$50K - $70K";"F";"1";"1";"Bachelors
  Degree";"1991-06-11";"Bronze";"Professional";"Y";"2";"Jeanne Derry"
4 "4";"87500482201";"Spence";"Michael";"J.";"337 Tosca
  Way";NULL;NULL;NULL;"Burnaby";"BC";"74674";"Canada";"92";"929-555-7279";"272-555-2844";"196
  9-06-20";"M";"$10K - $30K";"M";"4";"4";"Partial High School";"1994-05-21";"Normal";"Skilled
  Manual";"N";"2";"Michael Spence"
5 "5";"87514054179";"Gutierrez";"Maya";NULL;"8668 Via
  Neruda";NULL;NULL;NULL;"Novato";"CA";"57355";"USA";"42";"387-555-7172";"260-555-6936";"1951
  -05-10";"S";"$30K - $50K";"F";"3";"0";"Partial
  College";"1992-08-21";"Silver";"Manual";"N";"3";"Maya Gutierrez"
```

Figure 5: Customer Table Saved in the CSV Format

5. Conclusion

Increasing number of government organizations or enterprises are trying to utilize big data recently. In this situation, how to collect scattered data across the networks receive more and more attention. Target data to collect include unstructured data such as log, photo, and video as well as structure data stored in relational databases in the table and column forms. Understanding its significance, many enterprises and research institutions develop software products to this end and many people are using them.

However, most of the open source-based programs present text-based interfaces requiring special knowledge to users. Other easy-to-use commercial software products, on the other hand, require costly license fee to pose a barrier for more users. In this situation, the present study installed a data collection and transmission program in multiple relational database servers to collect data from and installed a server program in servers for data analysis. By doing so, communication is established between the client and server program. Based on it, the present study developed a web-application-based management program that extracts and transmits desired data and tested a data collection/transmission framework that collects and transmits data.

This study development is expected to be useful for users as they can connect to a target database, transmits data to servers and monitors the processes just with simple manipulation in a relatively friendlier UI instead of the complicated console-based commands.

Subsequent study will need to look at the unstructured data collection such as log, photo and video while exploring efficient storage method of collected data in big data distributed processing systems like Hadoop.

References

[1] Song Y.J., Policy Challenges for the Future of Data-Based Country Strategy, NIA, IT Future Strategy, 2013.

[2] Hoo-Young Lee, Koo-Rack Park, Dong-Hyun, Kim, A Study on Structured Data Collection Framework System Based on GUI, The 4th ICDPM, Danang, Vietnam, 2017.

[3] STAMFORD, Conn., Gartner Says Solving 'Big Data' Challenge Involves More Than Just Managing Volumes of Data, 2011.

[4] McKinsey Global Institute. Big Data: The next frontier for innovation, competition and productivity, McKinsey Global Institute, 2011.

[5] Cha Sang-Yook A., Study on Big Data Circumstance and Privacy Protection, IT & Law Research Institute, 2014.

[6] McKinsey, Big Data: The Next Frontier for Innovation,

Competition, and Productivity, McKinsey & Company, 2011.

[7] Man-Mo Kang, Sang-Rak Kim, Sang-Moo Park, Analysis and Utilization of Big Data. Communications of the Korean Institute of Information Scientists and Engineers, 2012.

[8] Sun Microsystems JAVA EE.

[9] The Apache Software Foundation Tomcat.

[10] Yoonyoung Park, Haecheol Park, Haewon Byun, Design and Implementation of the Sharing of Book Information Oriented On line Library System Using Spring Framework, Korean Institute of Information Scientists and Engineers, Korea Computer Congress (2009), 383-387.

[11] JiHun Cha, TaeHyeong Kim, SeungHa Lee, YangWoo Kim, Design of HDFS Interface based on Spring Framework, Korean Society for Internet Information (2009), 177-180.