ICIC Express Letters
Part B: Applications
Volume 8, Number 1, January 2017

ICIC International ©2017 ISSN 2185-2766
pp. 43–49

A LOG HANDLING APPLICATION FRAMEWORK FOR PROCESS MINING OF STEEL INDUSTRY

Jong Ik Jang[1], Keun Hee Kim[1], Nayeon Kim[2]
Gyusung Cho[3] and Minsoo Kim[1,*]

[1]Graduate School of Management of Technology
Pukyong National University
Yongdang Campus, 365 Sinseon-ro, Nam-gu, Busan 48547, Korea
svisor@naver.com; jfranco@pukyong.ac.kr; *Corresponding author: minsky@pknu.ac.kr

[2]Division of Systems Management and Engineering
Pukyong National University
45 Yongso-ro, Nam-gu, Busan 48513, Korea
nakim@pukyong.ac.kr

[3]Department of Port Logistics System
Tongmyong University
428 Sinseon-ro, Nam-gu, Busan 608-711, Korea
gscho@tu.ac.kr

Received June 2016; accepted September 2016

Abstract. *Many companies have been investing in various information systems such as enterprise resource planning (ERP), manufacturing execution system (MES), advanced planning & scheduling (APS) and supply chain management (SCM) for their competitiveness. Over the year, data volume in business and the types of log created by enterprise applications have been rapidly increasing with all those information systems. Under this IT environment, process mining techniques draw global attention to find business values from these huge log data. As the world enters the era of big data, tools and methodologies are actively being researched for storing and analyzing massive volume of data. This is also applied to the process mining area that actively searches an application framework to incorporate big data related technologies into log handling methodologies. In this paper, authors categorize multiple types of log, and organize the procedure of process mining from the viewpoint of big data technology such as Hadoop and Hive. Authors also propose a method to archive extracted logs into Hive, and to produce information from Hive by retrieving and refining. By demonstrating the proposed approach for log data from steel industry, authors show the usability of the proposed approach in accordance with commercial process mining tools like DISCO.*

Keywords: Process mining, Log analysis, Log mining methodology, Steel industry

1. **Introduction.** With the progress of information technology and its long history of enterprise application, lots of enterprise IT systems have been deployed to the business environment. Companies nowadays run so many IT systems like ERP (Enterprise Resource Planning), MES (Manufacturing Execution System), APS (Advanced Planning & Scheduling), SCM (Supply Chain Management), just to name a few. The log data that such enterprise applications generate have already been accumulated in volume and are still even more explosively expanding. Such high-volume of data that increases with high-velocity and high-variety is now commonly called big data [1-3]. It is not difficult to find applications in distributed processing area that incorporate big data technology such as Hadoop, MapReduce and NoSQL. However, utilizing those big data technologies is not common and not easy either in the enterprise business application. According to survey data, 85 percent of organizations plan to use big data, but only 17 percent believe they have adequate capabilities to make full use of the technology [4,5]. This is also applied

to process mining applications in enterprise environment. This situation, however, will change soon with active studies on process mining applications over big data platform.

As regards process mining research in enterprise environment, there have been many studies on analytics of weblog or security log, but not mainly on log analytics of business applications. Most of previous researches are done on a single type of log or done with predetermined types of log [6-8]. Under big data enterprise environment where various types of log are intensively generated, process mining application should be able to process various types of log or even properly handle dynamically changing types of log. In this paper, authors introduce multiple types of log into big data enabled process mining application. Different types of log generated by MES in the steel industry are classified and stored in the Hive that is a data warehouse infrastructure built on top of Hadoop for big data handling. Authors also propose a regular procedure to extract archived log instances from Hive and feed them to process mining application for subsequent analysis.

The organization of this paper is as follows. After the introduction of Section 1, Section 2 briefly overviews related technologies and works on big data and process mining. In Section 3, current requirement and system analysis for log handling of big data enabled processing mining are presented. Detailed design of the proposed log handling framework that meets identified requirements is explained in Section 4. Finally in Section 5, conclusions and future research issues are described.

2. **Related Technology and Works.**

Big data goes with lots of new technologies and concepts such as Hadoop, NoSQL and Hive for handling exploding volumes of structured and unstructured data. To enable effective and efficient process mining in the real industry application, such big data technologies should be appropriately applied over existing application framework. In this section, some big data technologies and process mining concepts are briefly introduced together with glue application framework.

2.1. **Hadoop and NoSQL.**

Hadoop is an open-source software framework for storing data over distributed environment and for running applications on clusters of commodity hardware. It provides massive storage for any kind of data, enormous processing power and the ability to handle virtually limitless concurrent tasks or jobs. The differences from other distributed computing are its high accessibility, robustness, scalability, and simplicity [9,10]. HDFS (Hadoop Distributed File System) is highly fault-tolerant and provides high throughput access to application data. HDFS is designed to be deployed on low-cost hardware and suitable for applications that have large data sets. MapReduce is a software framework for easily writing applications which process vast amounts of data in-parallel on large clusters of hardware in a reliable and fault-tolerant manner, typically over HDFS.

NoSQL database which stands for ‘Not only SQL’ is a scalable and distributed database that is designed to provide highly reliable, flexible and available data management to overcome conventional relational databases. Because relational databases were not designed to cope with the scale and agility challenges that modern applications face, NoSQL database is highly requested to handle huge amount of data in a distributed environment. There are currently more than 200 NoSQL databases, and Hive is one of such NoSQL databases [11,12]. Hive was initially developed by Facebook and is now a data warehouse package built on top of Hadoop [9]. Hive provides HiveQL (Hive Query Language) that looks very much like traditional SQL database query instead of letting the developers learn MapReduce programming again to use Hadoop, so it reduces learning cost and time for accessing data on Hadoop.

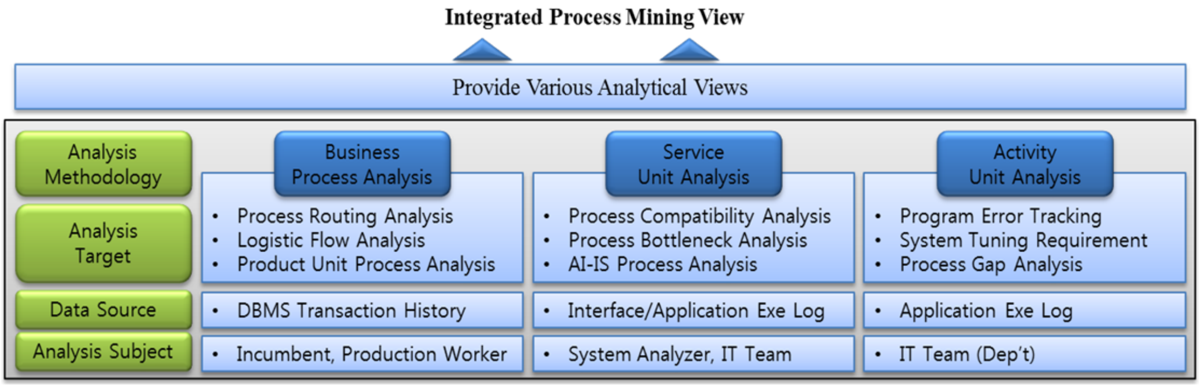
2.2. **Process mining and glue framework.**

The purpose of process mining is analyzing event log of big data generated by enterprise applications such as MES, ERP, BPM, and CRM. By analyzing event log, it can find meaningful information and optimize their business process. Process mining is useful in all stages of the business process, but most are utilized in ‘diagnostic’ and ‘perform’ stages. Prerequisite for process mining is that event logs have been sorted in chronological order, and these should be separately stored in each case [13,14]. The core technology in process mining is a technique for deriving the process model from the event log. It can be useful to find a process that has not yet been made visible in a company formally. Even if the process is not standardized, such as hospital care process that is highly flexible and varying, it can be visualized through a process mining [13,15]. The biggest thing that a company or organization can expect is improving management level and performance of their business process. It also secures flexibility, agility and transparency of the process to respond in the rapidly changing environment.

Glue framework is an integrated application framework based on standard J2EE framework. It is a platform and vendor independent framework that is possible to handle both UI (User Interface) and Non-UI (message and data) processing. It has been applied to many applications of various industries such as steel, electronics, chemicals, and shipbuilding. Glue framework is also widely used in large steel companies of South Korea including POSCO ICT which is the target application company of this paper. Key features of glue framework are its visual design capability for business activity, and high reusability of component services. It is quite efficient for batch job processing as well as for normal business operation. Compared to other frameworks like full-stack framework, glue framework allows user’s control for codes and library preference. These features of glue framework are very useful to integrate big data technologies with process mining application. If glue framework is used to implement proposed log handling framework, it is possible to manage data acquisition and processing rapidly, and analyze big data with automated processing of log data. It also presents several analytical perspectives and analysis methods, and provides a tool to improve efficiency for both process owners and information systems personnel.

3. **Current Requirement and System Analysis.** Since process mining is an iterative task that is performed periodically and continuously by the users from different administrative levels, log handling framework should be able to analyze subjects in accordance with different types of user aspects. The proposed framework provides three aspects of process mining which are business process view, service unit view and activity unit view as described in Figure 1.

A business process can be composed of multiple service units, and each service unit can be subsequently composed of multiple activity units. For example, a business process called ‘item production’ consists of service units such as ‘allocating material’, ‘material feeding’, ‘result inspection’, and ‘outcome task’. The service unit of ‘allocating material’ also consists of ‘searching material location’ and ‘setting material allocation position’


Integrated Process Mining View
Provide Various Analytical Views

Analysis Methodology	Business Process Analysis	Service Unit Analysis	Activity Unit Analysis
Analysis Target	• Process Routing Analysis • Logistic Flow Analysis • Product Unit Process Analysis	• Process Compatibility Analysis • Process Bottleneck Analysis • AI-IS Process Analysis	• Program Error Tracking • System Tuning Requirement • Process Gap Analysis
Data Source	• DBMS Transaction History	• Interface/Application Exe Log	• Application Exe Log
Analysis Subject	• Incumbent, Production Worker	• System Analyzer, IT Team	• IT Team (Dep’t)

Figure 1. Properties of the proposed framework

activity units. The data from business process level are usually archived in the DBMS, and service unit level data are stored between information systems as a separate text file or stored in the application execution log. Finally, activity unit level data are typically archived within the application execution log.

A log handling framework for big data enabled process mining should be able to provide following functionalities. Firstly, it should be scalable following the volumes of log data. Secondly, it should be flexible and extensible to handle multiple types of log and even to allow the introduction of new types of log rather than being fixed to a specific log format. Finally, fast data extraction is required while achieving storage space efficiency.

To explain the proposed log handling framework for the steel industry, three identified categories of log in the target application area are briefly summarized in Table 1. All logs in this case study are generated by MES system of POSCO ICT, and are related to individual works of that company. Level 2 transmission history log is for records that are generated by communication between different levels of systems. Level 3 stands for MES system layer, and level 1 stands for PLC (Programmable Logic Controller) or sensor layer. Level 2 layer bridges the communication between those two levels, and stores transmission history as a log data. Data transaction log is for records that are stored in DBMS as a result of service execution by information system. Finally, the application execution log is for records that are generated by specific activities or batch job tasks while executing UI or Non-UI programs. These are typically generated as a separate file by using Log4j library in the program. Several examples of application execution log are listed in Table 2. With this category of log, the start and end of service, transaction, and activity are recorded as event as well as query execution and query parameters.

Table 1. Log types of target company

Category	**Description**	**Types**
Level 2 transmission history log	Transmission history of system which manages processing facilities	{ Input | Work | Product | Pause | Partition } record
Data transaction log	Record generated by data processing or output result	Transmitted record to and from other system, work production record
Application execution log	All service and activity records generated by program execution	Log4j output record

All above three categories of log are processed by the proposed log handling framework. Since all categories of log come with different formats in different companies, the exact archiving format for Hive can vary between companies.

4. **Proposed Framework Design.**

The overall outline of the proposed log handling framework is briefly described in Figure 2. The proposed framework consists of three phases: 'Log Collection', 'Hive Repository', and finally 'Data Extraction/Analysis' phase.

At log collection phase, all three categories of log are periodically collected from the data sources via various types of transfer protocols or query processing. During this collection phase, necessary preprocessing such as formatting, sorting and summarizing is conducted. At the Hive repository phase, collected log is archived to Hive database in a format that is customized to specific mining application like DISCO. This is to increase the performance of process mining tasks for large volumes of data. Authors also introduce log partitioning scheme to further enhance the performance of analysis by applying sliding window concept for monthly or weekly time period data. Since most of process mining tasks are usually

Table 2. Examples of application execution log

Types	Sample Log	Meaning
Audit	22th Sep. 07:24:00 Debug [[ACTIVE] ExecuteThread: '6' for queue: 'weblogic.kernel.Default(self-tuning)'] – PosAuditAttributes.a(246) | Default Audit info: [ObjectType=A][ObjectID=E07942] [ProgramId=M471010070-service]	Date Written: 22th Sep. 07:24:00 WAS Execution Thread: 6 EmpID: E07942 Service: M471010070
Service Start	22th Sep. 01:43:00 INFO [[ACTIVE] ExecuteThread: '6' for queue: 'weblogic.kernel.Default (self-tuning)'] – PosBizController.do.Action(212) | ServiceName: [M472020090tab01-service] StartTime[Mon Sep 22 13:43:00 KST 2014]	Date Written: 22th Sep. 01:43:00 WAS Execution Thread: 6 Service Exe: PosBizController.doAction Service: M47202009tab01 Servic Start Date: Mon Sep 22 13:43:00 KST 2014
Service End	22th Sep. 01:45:00 INFO [[ACTIVE] ExecuteThread: '6' for queue: 'weblogic.kernel.Default (self-tuning)'] – PosBizController.do.Action(212) | ServiceName: [M472020090tab01-service] EndTime[Mon Sep 22 13:43:00 KST 2014]	Date Written: 22th Sep. 01:43:00 WAS Execution Thread: 6 Service Exe: PosBizController.doAction Service: M472020090tab01 Servic Start Date: Mon Sep 22 13:43:00 KST 2014
Transaction Start	22th Sep. 01:43:00 INFO [[ACTIVE] ExecuteThread: '6' for queue: 'weblogic.kernel.Default (selftuning)'] – PosDataSourceTransactionManager. startTransaction(122) | Glue:TransactionStart	Date Written: 22th Sep. 01:43:00 WAS Execution Thread: 6 Transaction Exe:startTransaction
Activity Start	22th Sep. 01:43:00 INFO [[ACTIVE] ExecuteThread: '6' for queue: 'weblogic.kernel.Default (selftuning)'] – PosActivityHandler.runActivity(145) | ActivityName: [M472020090tab01service] StartTime [Mon Sep 22 13:43:00 KST 2014]	Date Written: 22th Sep. 01:43:0 WAS Execution Thread: 6 Activity Exe:runActivity Service: M472020090 tab01 Activity: Activity Start Date: Mon Sep 22 13:43:00 KST 2014

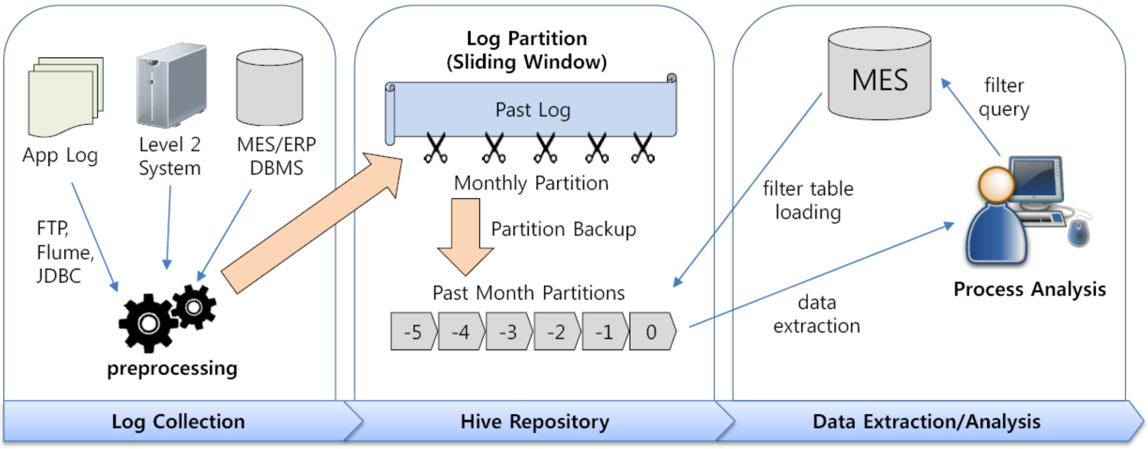


Figure 2. Overview of the proposed framework

conducted on recent log data, current data that is within the time window is kept intact and active, while old data that has passed time window is archived in a compressed format for economic storage usage. To achieve best performance for process mining task while minimizing overhead for compressing and decompressing old data, proper size of time window is very important.

Descriptions of the proposed framework, compared to the conventional method are shown in Table 3. Considering the proposed framework's big data awareness, performance enhancement via sliding window concept, and flexibility for multiple types of log, it will be quite promising to deploy suggested log handling framework into real world applications. Previous process mining researches in South Korea are mainly conducted for office and service processes, so researches on analysis or improvement of manufacturing sector are relatively scarce. In this study, authors have applied the proposed log handling framework to process mining application for manufacturing sector of steel industry as a demonstration of the proposed framework's feasibility. Though current implementation is not the full creation of the proposed framework including real-time processing of log data, it successfully has proved the operation of big data enabled process mining application.

Table 3. Conventional process mining vs. the proposed framework

	Conventional Approach	Proposed Framework
Process	Data Extraction ➔ Data Cleansing ➔ Data Transformation ➔ Data Editing ➔ Data Analysis	Preprocessing ➔ Hive Archiving ➔ Data Extraction ➔ Data Analysis
Analysis View	· No particular views provided	· Business process unit · Service unit · Activity unit
Analysis Log Type	· Simple type log · Same types of logs	· Simple type log · Complex logs, need lexical analysis · Business system history log
Data Filtering	· Filter at first log extraction · Filter archived data item	· Filter archived data item · Any types of filters with MES/DW feature
Loading/Extraction Method	· File/DBMS archiving in business system · Log Extraction/Transformation/Loading from business applications · Extract data from all logs to analyze	· Integrated Hive archiving for file/DBMS · Select only target logs · Fast log access via sliding time window · Old logs' are compressed while archiving
Analysis Goal	· Business process identification & improve · Recovery of application design document	· Business process identification & improve · Recovery of application design document · Information system performance enhance · Application error tracking

In this paper, not only application execution logs generated by MES but also work histories executed in the MES are included to process mining. When applying the result of this study to real world business, following improvements should be considered, namely, (1) event data quality (reliability, completeness, clarity, and information diversity), (2) process mining framework functionality (storage, reprocessing and data collection efficiency, analytical data efficiency), (3) organization and business regulations (operator training, reorganization, and business rules).

5. **Conclusions and Future Research.** In this paper, a log handling framework for the process mining of steel industry is proposed. Using a variety of log analysis, authors identify 3 distinct types of mining view, and suggest 3 categories of log for steel industry application. The contributions of this study and some benefits of the proposed framework are as follows. First, a working prototype of process mining for steel industry that classifies various types of log is presented. It also provides different analytical points of view based on subjects, which is customizable to each application area. Second, using sliding window concept based on the creation time of data, frequently used recent data is stored uncompressed, while passed time window data is archived in compression. This makes more efficient use of storage spaces while considering analysis performance. Third, by

using the business data in the MES/DW, a way to filter required data in a short time is presented compared to traditional mining processes.

Since current implementation relies on batch-processing over FTP (File Transfer Protocol) for collecting and processing log data, it is not a real-time implementation. It thus requires further enhancement to provide real-time handling of log data. In addition, some queries or configuration files still require manual job, so it needs ongoing management from the user. In order to minimize those administrative tasks, the use of automated tools is strongly needed. The analytic view and log types can be different in other industries. It thus is needed to organize further studies for specific industrial applications.

Acknowledgment. This work was supported by a Research Grant of Pukyong National University (2016 year).

REFERENCES

[1] D. Laney, 3D data management: Controlling data volume, velocity, and variety, *META Group Original Research Note*, 2001.

[2] M. Schroeck, R. Smart, D. Romero-Morales and P. Tufano, Analytics: The real-world use of big data: How innovative enterprises in the midmarket extract value from uncertain data, *IBM Institute for Business Value*, 2012.

[3] W. Fan and A. Bifet, Mining big data: Current status, and forecast to the future, *ACM SIGKDD Explorations*, vol.14, no.2, pp.1-5, 2013.

[4] P. Barth and R. Bean, Who's really using big data, *Harvard Business Review*, 2012.

[5] M. Chui, J. Manyika, J. Bughin, B. Brown, R. Roberts, J. Danielson and S. Gupta, *Ten IT-enabled Business Trends for the Decade Ahead (Updated Research)*, McKinsey Global Institute, 2013.

[6] D. Wang, J. Ge, H. Hu and B. Luo, A new process mining algorithm based on event types, *IEEE the 9th International Conference on Dependable, Autonomic and Secure Computing*, pp.1144-1151, 2011.

[7] I. Ari, E. Olmezogullari and O. F. Celebi, Data stream analytics and mining in the cloud, *IEEE the 4th International Conference on Cloud Computing Technology and Science*, pp.857-862, 2012.

[8] M. A. Chaves and E. R. Cordoba, Deciphering event logs in SharePoint server: A methodology based on process mining, *Latin American Computing Conference*, 2014.

[9] A. Holmes, *Hadoop in Practice*, 2nd Edition, Manning Publications, 2015.

[10] T. White, *Hadoop: The Definitive Guide*, 4th Edition, 2015.

[11] NoSQL Website, *Your Ultimate Guide to the Non-Relational Universe!* http://nosql-database.org, 2016.

[12] T. Hills, *NoSQL and SQL Data Modeling: Bringing Together Data, Semantics, and Software*, 1st Edition, Technics Publications, 2016.

[13] W. V. D. Aalst, *Process Mining: Data Science in Action*, 2nd Edition, Springer, 2016.

[14] W. V. D. Aalst and A. J. M. M. Weijters, Process mining: A research agenda, *Computers in Industry*, vol.53, no.3, pp.231-244, 2004.

[15] W. V. D. Aalst et al., Process mining manifesto, *Lecture Notes in Business Information Processing*, vol.99, pp.169-194, 2011.