[*Seminar * et al., 5(1): January, 2018*] ISSN: 2349-5197
Impact Factor: 2.715

INTERNATIONAL JOURNAL OF RESEARCH SCIENCE & MANAGEMENT

DEVELOPMENT OF A PROTOTYPE OF EVENT STREAM PROCESSING ACTIVITY LOGGER FOR MOBILE APPLICATION

R Fiza Anwar Sharif[1], Kudang Boro Seminar[2]*, Arief Ramadhan[3]
[1]Master student of Business School, Bogor Agricultural University(IPB) Bogor, West Java, Indonesia, 16151
[2]Business School, Bogor Agricultural University(IPB) Bogor, West Java, Indonesia, 16151
[3]Business School, Bogor Agricultural University(IPB) Bogor, West Java, Indonesia, 16151

DOI: 10.5281/zenodo.1139951

Keywords: big data, event stream processing, system design, prototype.

Abstract
One of the critical requirements of Yellow Pages Mobile Applications is the ability to process and to analyse big data acquired from massive activity logs generated by Yellow Pages Mobile applications. Conventional computing and storage model is no longer compatible to handle big data processing . This research is aimed to develop a proptoype of event stream processing activity logger for mobile applications that can be implemented and aligned with current technological developments. The prototype development approach is based on Software Development Life Cycle that provides sistematic incremental development stages to deliver a final prototype of event stream processing. In the implementation stage Apache Kafka, Apache Storm and Apache Cassandra are used to support messaging system, distributed computing system, and media storage system respectively. The developed prototype was tested using two-month data logs of 50-gigabyte log activity from 48.880 transaction activity records. The transactions come from 89 registered members and 202 anonymous users. The result of this research shows that there are 1.998 times of performance improvement compared to the current existing system. This significant better performance promotes the utilization of the developed prototype of event stream processing logger technology as a potential substitute for the current existing processing system,.

Introduction
In 2017, PT Metra Digital Media (MD MEDIA) launched its new Yellow Pages Mobile Application to provide user an easy access to business directory across Indonesia. This application records all user's activities such as application menu selection, keyword search, and user location changes made in mobile apps and writes it directly into database server. However, the rapid development of this application's feature had given a significant effect to its performance due to it's growing amount of data, real-time monitoring and reporting system.

The implementation of the latest big data technology is expected to help to build an effective infrastructure prototype design for this application. Big data technology is now becoming a new trend in information technology to enable real-time data storage and a large-scale data analysis. This paper will show the importance of big data research as a solution to manage, store, and process large data.

Nowadays, companiesstart to records all of their user's system activities and it becomes a special challenge for the storage company to meet high customer demand. In 2020 the amount of data in the world would reach 40 zettabytes or equivalent to 43 trillion gigabytes [5]. This amount is 300 times larger compared to the whole data existed in 2005.

Research Objectives
This research aims to design prototype of event stream processing activity logger that can perform big data processing to support Yellow Pages Mobile Applications with better performance.

Related Work

Event Stream Processing (ESP) technology is commonly used in managing huge stream of log data. There are some of information technology solution vendors that offer ESP technologies with various business models, ranging from open source systems to licensed systems. Tibco from IBM, Azure Event Hubs from Microsoft, Kinesis from Amazon are few of the examples of paid ESP technology. While RabbitMQ, ActiveMQ, ZeraMQ, OpenMP and Kafka are open source ESP technology software.

A research developed as an event-driven infrastructure, was built on Service Oriented Architecture (SOA) for real-time detection of traffic event from twitter stream analysis [8]. Messaging Server Solution on surveillance video system using RabbitMQ is implemented on a research performed in 2014 [7]. In this research, Messaging Server can minimalize the number of server required used. A research use ActiveMQ as a solution to manage parameterized job request amongst bioinformatics analysis that results a picture [1]. This job request is deployed using ActiveMQ as message queue. When the job request is received, analysis module calls java application to download raw dataset of 5D pictures.

A research using Apache ActiveMQ as a solution for system authorization in IoT services. ActiveMQ utilization resulted in good indication with reasonable overhead [2]. Meanwhile a research comparing several paid technology platform [3]. Microsoft Azure is one of the biggest player in this platform sector, it even offers cloud computing services with Infrastructure as a Service (IaaS) and Platform as a Service (PaaS) model [3].

Various messaging system has been implemented in previous research, however, this research use Apache Kafka as ESP technology for the messaging system. Apache Kafka is used because it is an open source platform and there are many big corporation that also use Apache Kafka on their Event Stream Processing messaging system.

Apache Kafka as Event Stream Processing Messaging System

Apache Kafka is an open source distributed messaging mediation system for real-time data stream that can work stably. Apache Kafka was formerly developed by LinkedIn in January 2011 and this technology keeps developing and utilized by many corporations such as Yahoo, Twitter, Spotify and much more.

The utilization of Apache Kafka in LinkedIn during 2014 had reach 300 kafka brokers and more than 18.000 kafka topics managed by more than 140.000 partitions. This massive infrastructure is used for managing 220 billion data per day that requires 40 terra bytes input and 160 terra bytes output data [4].

Based on the research, Apache Kafka delivers better performances compared to other application such as RabbitMQ version 2.4 and ActiveMQ that uses Kahadb as a default storage [6]. This research used two Linux machine with an 8-core 2GHz processor, 16GB memory and 6 hard-drive with RAID 10. This machine is connected to a network with 1 GBPS bandwidth. The first machine serves as a broker and the second machine serves as producer or consumer.

Apache Cassandra

Apache Cassandra is a distributed decentralized database system with high availability and throughput. Because of its distributed system, this system is free from single point of failure. Apache Cassandra was formerly developed by Facebook and became an open source platform in July 2008. Together with Apache Thrift, one of Apache products, Apache Cassandra has its own query language named Cassandra Query Language (CQL) which is similar to Structured Query Language (SQL). Compared to SQL, CQL has an advantage in managing large-scale data with better performance because of its cluster technology. Apache Cassandra's Topology uses cluster system with nodes, and more nodes will result in better performance.

Distributed Computing Using Apache Storm

Apache Storm is a real-time distributed computing system that was developed by Back Type and subsequently acquired by Twitter in September 2011. Similar with Apache Kafka, Apache Storm also used by Yahoo,

Twitter, Spotify and other big corporation. Apache Storm also uses cluster system from Zookeeper, which required at least two components, namely Storm Nimbus and Storm Supervisor.

Storm Nimbus serves as the master node that is responsible for distributing codes to every cluster, assigning Storm Supervisor and monitoring system failure.

Research Method

Framework

This research begins with a literature study and the main problem is the declining performance of Yellow Pages Mobile Application. The next step is performing an investigation and problem identification. After identifying the problem, the researcher performs system requirement analysis based on previous research. Then, to develop the system, the researcher uses prototyping approach with event stream processing model. The next phase is to evaluate the performance of the existing system and compare it to the performance of the developed prototype system. If the system performance is declining or become stagnant, the researcher have to re-develop the system prototype. However, if there is an improvement in performance, then this research will move to the result analysis and conclusion. The research framework is illustrated in Figure 1 below:

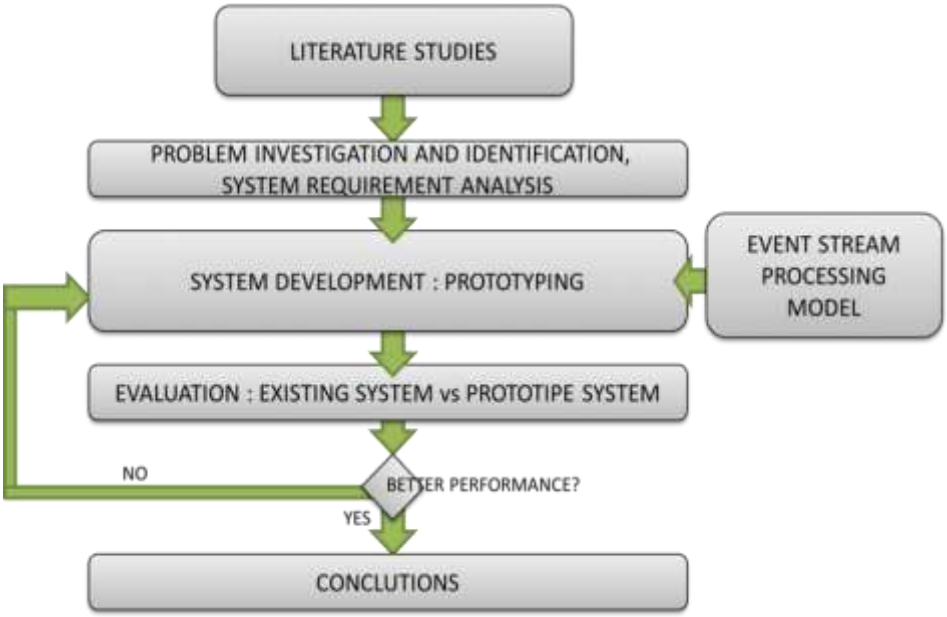

Figure 1. Research Framework

Research Design

This research uses Software Development Life Cycle (SDLC) as an approach in prototyping [10]. Prototyping begins with System Analysis and Design, Prototype System Design and Output, and System Evaluation and Testing. This research refers to the architecture design concept where it results in Data Management System with this structure [11]:

- Physical Layout And Representative Concept
- System Architecture Design And Implementation
- Taxonomy Data Model
- System Architecture Taxonomy
- Model Consistency Taxonomy

One of the most important steps of this research is designing the prototype. This research uses Sequence Diagram in prototype designing. The research phase is illustrated in Table 1 below:

Table 1. Prototype Development Design Phase

	Research Design Phase	Technical Development
1	System Analysis and Design	• Business Process Design • Infrastructure: Architectural Design • Sequence Diagram Model
2	Prototype System Design	• Basic Platform Installation and Configuration (Ionic, Apache Kafka, Apache Storm, Cassandra) • Database Installation and Configuration • User Profile Data Model Installation and Configuration
3	Prototype Output and Result	• Stream Data Processing Platform Infrastructure • Log system Database
4	Protipe Evaluation and System testing	• Comparison between Existing system and prototype associated and related to data definition and manipulation o Estimated time for table creation (Create Table) o Estimated time for concurrent data entry (Import Table) o Estimated time for data query (Query Table) o Estimated time for export data (Export Data)

Findings

Prototype Development

Activity log of event stream processing was implemented on user's activity log in detail. The design process of activity log is illustrated on Figure 2 in 7 phase using Sequence Diagram.

- Phase 1: Yellow Pages Mobile Application user serves as an actor (<<actor>>) in performing activities (ex: category searching) on Yellow Pages Mobile Application.
- Phase 2: Yellow Pages Mobile Application serves as a user interface (<<UI>>) that performs activity log of category searching and then send the log to Apache Kafka. In Apache Kafka's framework, Yellow Pages Mobile Application serves as a Kafka Producer.
- Phase 3: Apache Kafka serves as a device that manages data stream traffic in every log activity (<<controller>>).
- Phase 4: Apache Storm serves as Kafka Consumer in Apache Kafka platform. Apache Storm serves as data stream user in Apache Kafka. Data acquisition is performed by Apache storm using distributed computing process.
- Phase 5: Apache Storm performs data stream computation and delivers it to Apache Cassandra storage.
- Phase 6: Apache Storm performs data stream computation and delivers it to Apache Cassandra storage to record every log activity of category searching by the users.
- Phase 7: Apache Cassandra stores data from every log activity stream into a specific keyspace.

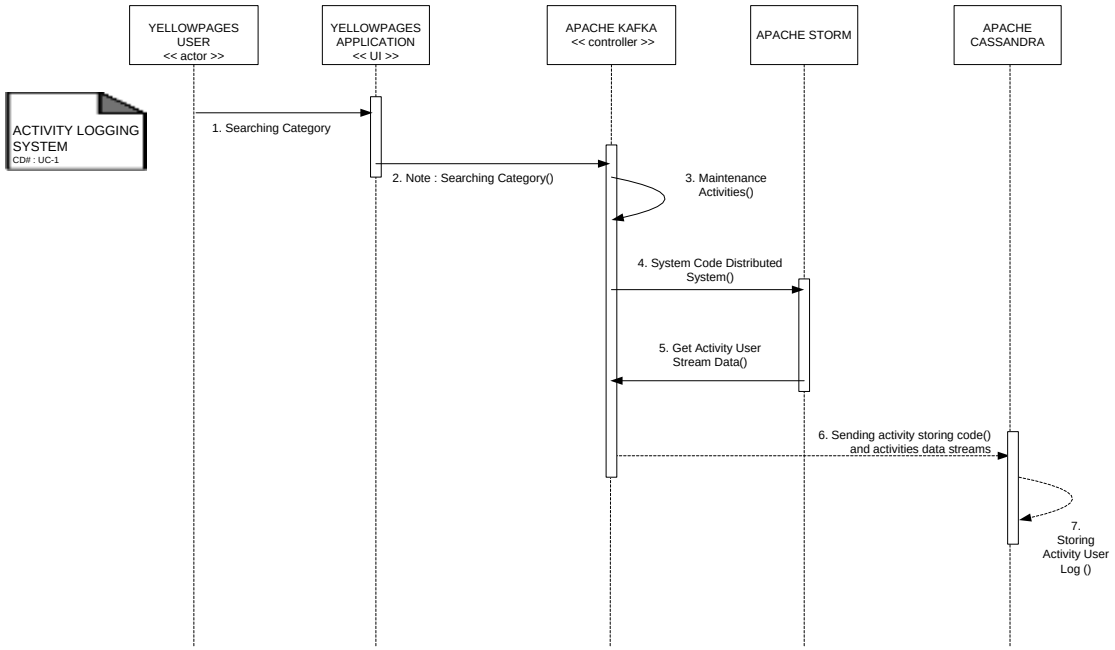

Figure 2. Sequence Diagram of Design System Business Process

The architecture design of prototype system infrastructure was developed by integrating several devices as referred to previous system design.

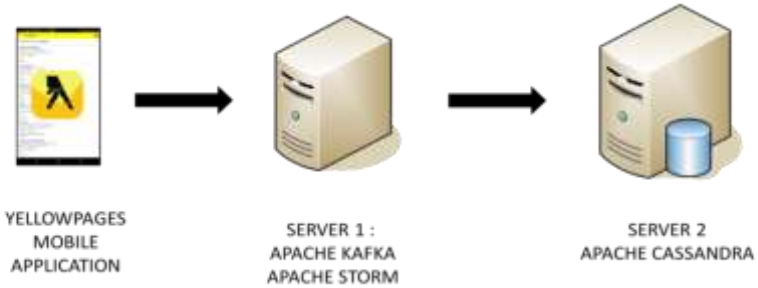

Figure 3. Architecture Infrastructure System

The architecture design is illustrated in Figure 3 where infrastructure device begins from data sources, data distribution process and ends in data storing. Yellow Pages Mobile Application that has been installed on mobile phone serves as the data source. The produced data is streamed to distribution process in Server 1. Server 1 has processor 2 core Intel Xeon X7560 2.27 GHz, RAM 8 GB Memory, with Linux Centos 6.7 with Apache Kafka and Apache Storm installed in it.

Apache Kafka will store the data as a temporary stream. This stream of data will be distributed and stored in Server 2. Apache Cassandra in Server 2 has Processor 4 Core Intel Xeon X7560 2.27 GHz, RAM 12 GB Memory with Linux Centos 6.7 to store the data completely. Apache Storms works for performing distributed data streaming process orchestration from Apache Kafka to Apache Cassandra. Figure 4 illustrates the data stream stored in Apache Kafka.

Figure 4. Apache Kafka Streaming Data

Prototype Evaluation

This two months of research has resulted in 50 gigabytes of log activity only from 48.880 transaction log. This transaction comes from 89 registered members and 202 anonymous users.

Table 2. System Performance Evaluation Test Results

	Testing	Existing System Result (in second)	Prototype System Result (in second)	Increase Ratio Parameter
1	Estimated time for table creation (Create Table)	0.02	0.01	2.000
2	Estimated time for concurrent data entry (Import Table)	17.02	7.03	2.421
3	Estimated time for data query (Query Table)	12.03	10.06	1.196
4	Estimated time for export data (Export Data)	82.954	34.954	2.373
			Average Performances	1.998

From table 2, there are performance improvements of 1.998 times faster compared to the existing system. These findings are expected to prove that event stream processing technology can be used to replace the existing technology.

During this research, the writer faced obstacle where the system performance was declining due to the needs of a bigger platform for event stream processing. The capacity of Apache Kafka storage was reaching 80% of usage and results in gradual decrease of performance. The test on Table 2 was performed when the system is in 20% of memory usage.

Conclusions

The event stream processing prototype development was developed with a simple configuration to help management, especially in Information Technology Division understand the prototype easier. This prototype design also utilizes the latest big data technology as implemented in other big corporations. System development for commercial purposes can be executed according to the next product development design. The low capacity of infrastructure prototype in existing production capacity is expected to prove that the utilization of this event stream processing technology can significantly improve performances if implemented in proper production infrastructure. This prototype can be implemented immediately and complies with the latest technology and Yellow Pages product development strategy. The discovery of obstacles in this prototype has been identified and can be improved for further development. A research of event stream processing architecture implemented

for secure advertising media solution in the distribution process to its user [9]. In future, this prototype can be utilize as an advertising media solution in the distribution process to Yellow Pages Mobile Application user.

References

[1] Bilgin CC, Fontenay G, Cheng Q, Chang H, Han J, Parvin B, "BioSig3D: High Content Screening of Three-Dimensional Cell Culture Models" in PLoS ONE, VOL. 11, NO. 3, pp. e0148379, 2016. doi:10.1371/journal.pone.0148379.

[2] Duan L, Zhang Y, Chen S, Wang S, Cheng B, Chen J, "Realizing IoT service's policy privacy over publish/subscribe-based middleware" in SpringerPlus, VOL. 5, pp.1615, 2016. DOI 10.1186/s40064-016-3250-x.

[3] Fylaktopoulos G, Goumas G, Skolarikis M, Sotiropoulos A, Maglogianis I. 2016. "An overview of platforms for cloud based development". SpringerPlus, VOL. 5, NO. 38, 2016.. DOI 10.1186/s40064-016-1688-5.

[4] Haskin C, Palino T, "LinkedIn Enterprise Kafka", http://www.slideshare.net/ToddPalino/enterprise-kafka-kafka-as-a-service?qid=e748c4cd-cbf5-46f0-a1cf-4a0018a13a86&v=&b=&from_search=1, March 2017

[5] IBM, "IBM Big Data & Analytics Hub", http://www.ibmbigdatahub.com/infographic/four-vs-big-data, April 2017

[6] Kreps J, Narkhede N, Rao J, "Kafka: a distributed messaging system for log processing", http://notes.stephenholiday.com/Kafka.pdf, March 2017

[7] Limna T, Tandayya P, "A flexible and scalable component-based system architecture for video surveillance as a service, running on infrastructure as a service" in Springer Science + Business Media, VOL. 75, pp. 1765-1791, 2014. DOI 10.1007/s11042-014-2373-8.

[8] Naidu K V G N, Sireesha P, "Twitter Analysis for Identification of Real-time Traffic" in International Journal of Research Science & Management", VOL 4. NO. 5, Pp.148-151, May 2017.

[9] Oger M, Olmez I, Erinc Inci E, Kücükbay S, Fatih Emekci F, "Privacy-Preserving Secure Online Advertising" in Procedia - Social and Behavioral Sciences, VOL. 195, pp.1840 – 1845, 2015. DOI: 10.1016/j.sbspro.2015.06.405.

[10] Seminar K B, Rizqya E M, Buono A, "Prototype Development of a Traceability System for Coconut Palm Sugar Supply Chain in Indonesia" in International Journal of Research Science & Management", VOL 4. NO. 11, Pp.69-76 November 2017.

[11] Wu L, Yuan L, You J, "Survey of large-scale data management systems for big data applications" in Journal Of Computer Science And Technology, VOL. 30, NO. 1, pp. 163–183, January 2015. DOI 10.1007/s11390-015-1511-8.

.