

## A Design of Web Application for Complex Event Processing Based on Hadoop and Java Servlets

<sup>1</sup>Maxim Shcherbakov, <sup>1</sup>Dmitry Kachalov, <sup>1</sup>Valery Kamaev,  
<sup>1</sup>Nataliya Shcherbakova, <sup>1</sup>Anton Tyukov and <sup>2</sup>StrekalovSergey  
<sup>1</sup>Volgograd State Technical University, Lenina av. 28, 40000 Volgograd, Russia  
<sup>2</sup>Volgograd State University of Architecture and Civil Engineering, Akademicheskaya St. 1,  
400074 Volgograd, Russia

**Abstract:** Different domains require perform complex event processing and pattern matching in large amount of text files or log files. Hadoop is the technology stack to do this kind of job. However, it is still open question how to build web based application for interaction with user as a frontend where backend components implement Map Reduce jobs over Hadoop? The contribution of the study is the design of web application for complex event processing for log analysis over Hadoop based on Java Servlet technique. The real implementation is performed to prove the performance of suggested approach.

**Key words:** Design, web application, Hadoop, Java servlets, log analysis

### INTRODUCTION

Log analysis and complex event processing in the big issue in different domains. This analysis allows to understand customers behaviour more precisely and use this knowledge for more efficient interactions. Moreover, the results of this analysis open possibilities to build proactive systems allow to prevent negative cases or reduce the damage due to its occurrence (Engel and Etzion, 2011; Owoeye *et al.*, 2013). The performance of the log analysis and complex event processing depend on the scale of data. In case of big data, this issue become very crucial for business. Existed infrastructure needs improvement in proceeding of big data and we can state the following problem: need to develop the efficient ways for log analysis complex event processing over terabytes of data. Hadoop is used as a technique for distributed data processing over large amount of data (Vavilapalli *et al.*, 2013; Shcherbakov and Tykov, 2014). However, the main restriction here is low integration of Hadoop technology stack to business application and lack of frameworks for building enterprise decision support systems. In this case, there is a question: how to build web application for complex event processing over Hadoop in efficient way? The contribution of the study is the design of web application for complex event processing over Hadoop based on Java Platform.

### DESIGN

The proposed design is based on Hadoop technology stack and Java Platform. There are three layers

could be defined here: the representation level for communications with users; business logic layer consists of components performing the log analysis and data layer (HDFS). Also model-view-controller has been chosen as a basic concept. Figure 1 shows the proposed design.

The architecture contains the view block pages represented as the JSP pages. Using these pages users set the log analysis templates (e.g., the template of events for searching as the tuples of 'key-value' pairs) and launch the task for searching. It sends the AJAX request which calls Java Servlet and the user interface displays information about the progress of analysing. This request

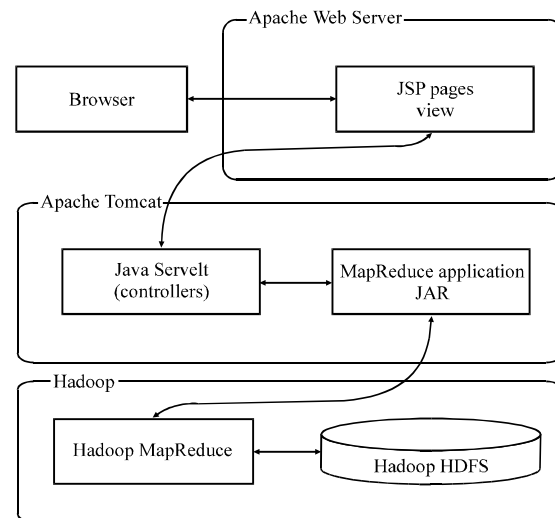


Fig. 1: The proposed design of web application

is handled by controller which is implemented as a Java Servlet. It actualizes an appeal to Hadoop. Since, search in log has been completed the servlet sends the response back to user for informing user.

The business layer consists of two applications and build based on low coupling concept. The first application is a Java Servlet that handles the requests via POST or GET methods. The second one is Mapreduce Java application performs Map Reduce tasks. The last application is compiled and archived as a JAR file and included in the project. Java Servlet receives log analysis patterns from user, verify these patterns and call the search function (over HDFS) and finally provides the results and performance time. The Java Servlet reads the results of Mapreduce task from HDFS. Map Reduce application contains 'map', 'reduce' and 'run' functions for implementation of log analysis over Hadoop. It returns true or false, depending on the success of the research. The input parameters are:

- The file name or folder in HDFS
- An array of key-value pairs where the key is the value of a field in which there is a search and the value the value of this field. Java-based application contains a function mapper and function reducer too

### IMPLEMENTATION

Log analysis is implemented in Map Reduce Java class with run, map and reduce methods. Mapper gets strings from the selected file, split the current string and generates output keys and values pairs. Reducer searching for pairs key values based on user defined template. To launch the Map Reduce the run method is called externally:

- Long time = System.currentTimeMillis()
- Loganalysis.FindTemplate
- Run(request.getParameter("filename"))
- Long time2 = System.currentTimeMillis()

In this code we called function run from namespace loganalysis and base class Findtemplate (name of Java-based application). Also, we calculate the time of performance. After process has been finished Java Servlet reads results from HDFS using the code (Algorithm).

**Algorithm: Source code snapshot for reading files from HDFS:**

```
Path pt = new Path ("hdfs://hdnode:10001/user/output/part-00000");
Configuration conf = new Configuration();
conf.set("fs.default.name", "hdfs://hdnode:10001/");
FileSystem fs = FileSystem.get(conf);
BufferedReader br = new BufferedReader(new InputStreamReader(
fs.open(pt)));
String line;
line = br.readLine();
```

### CONCLUSION

This short study presents the design of web application for log analysis over Hadoop based on Java platform. This concept could be used for implementing log analysis and complex event processing in large amount of data in different domain. The future research is:

- In implementation of more complicated algorithms in Hadoop technology stack
- Implementation of web application using spark as a backend
- Implementing the domain oriented tasks (Shcherbakova *et al.*, 2013; Owoeye *et al.*, 2013).

### ACKNOWLEDGEMENT

Researchers would like to thank RFBR for support of the research in framework of grant #14-07-00945.

### REFERENCES

- Engel, Y. and O. Etzion, 2011. Towards proactive event-driven computing. Proceedings of the 5th ACM International Conference on Distributed Event-based System, July 11-15, 2011, New York, USA., pp: 125-136.
- Owoeye, D.A., M. Shcherbakov and V. Kamaev, 2013. A photovoltaic output backcast and forecast method based on cloud cover and historical data. Proceedings of the 6th IASTED Asian Conference on Power and Energy Systems, April 10-12, 2013, Beijing, China, pp: 28-31.
- Shcherbakov, M. and A. Tyukov, 2014. Mapreduce-based algorithms for outlier detection in energy data. Proceedings of the 8th Multi Conference on Computer Science and Information Systems, June 2014, Greece.
- Shcherbakov, M., Y. Timofeev, A. Saprykin, V. Trushin and A. Tyukov *et al.*, 2013. An On-line and Off-line Pipeline-based architecture of the system for gaps and outlier detection in energy data stream. Proceedings of the 3rd Eastern European Regional Conference on the Engineering of Computer Based Systems, August 29-30, 2013, IEEE Computer Society, Washington, DC, USA., pp: 1-7.
- Vavilapalli, V.K., A.C. Murthy, C. Douglas, S. Agarwal and M. Konar *et al.*, 2013. Apache hadoop YARN: Yet another resource negotiator. Proceedings of the 4th Annual Symposium on Cloud Computing, October 1-3, 2013, Santa Clara, CA., pp: 1-16.