Applying Smart User Interaction to a Log Analysis Tool

Semini, K.A.H. and Wijegunasekara, M.C.

Department of Statistics & Computer Science, University of Kelaniya

Department of Software Engineering, University of Kelaniya

Email: hsemini@gmail.com

Abstract

A log file is a text file. People analyze these log files to monitor system health, detect undesired behaviors in the system and recognize power failures and so on. In general these log files are analyzed manually using log analysis tools such as Splunk, Loggly, LogRhythm…etc. All of these tools analyze log file and then generate reports, graphs which represents the analyzed log data.

Analyzing log files can be divided into two categories namely, analyzing history log files and analyzing online log files. In this work analyzing history log files is only considered for an existing log file analysis framework. Most of the log analysis tools have the feature for analyzing history log files. To analyze a log file using any of the mentioned tools, it is necessary to select a time period. For example, if a user analyze a log file according to the system health, the analyzed log file specifies the system health as 'good' or 'bad' only for the given time period. In general these analysis tools provide average system health for a given time period.

This analysis is good but sometimes it may not be sufficient as people may need to know exactly what happens in the system each second to predict the future behavior of the system or to get some decisions. But using these tools, it is not possible to identify the log file in detail. To do such analysis a user has to go through the log file line by line manually. As a solution for this problem, this paper describes a new smart concept for log file analysis tools. The concept works through a set of widget and it can replay executed log files.

First the existing log file analysis framework was analyzed. This helped to understand the data structure of receiving log files. Next the log file analysis tools were studied to identify the main components and features that people most like. The new smart concept was designed by using replayable widget and graph widgets. The replayable widget uses to replay the inputted log file and the graph widgets graphically represent the analyzed log data.

The replayable widget is the main part of the project and holds the new concept. This is a simple widget that acts just as a player. It has two components; a window and a button panel. Window shows the inputted log file and the button panel contains play, forward, backward, stop and pause buttons. The log lines which is shown in the window of the replayable widget, holds a tree structure (Figure 1: Left most widget). The button panel contains an extra button to view the log lines. These buttons are used to play the log lines, go to requested log line, view the log line and control playing log lines.

It was important to select suitable chart library to design the graph widgets. A number of chart libraries were analyzed and finally D3.js was selected because it provided chart source, free version without watermarks and it also had more than 200 chart types. It has a number of chart features and also it supports to HTML5 based implementations. The following charts were implemented using D3.js chart library.

- Bar chart according to the pass/failure count
- Time line according to the time of pass/fail occurs
- Donut chart according to the total execute count
- Donut chart for Total Pass/Fail Count

Every graph widgets are bind with replayable widget, so that updates are done according to the each action. The replayable widget and the graph widgets are implemented by using D3.js, JavaScript, JQuery, CSS and HTML5. The replayable widget is successfully tested and the implemented interface successfully runs in Google Chrome web browser.

Figure 1 shows a sample interface of the design which is generated using a sample log file that had about 100 of log lines. Left most widget is the replayable widget that holds considered log file as a tree structure. Top right most widget is one of the graph widget represented as a bar chart shows pass/failure count and the bottom right most widget is another graph widget represented as a time line shows the time of pass/fail that occurred for the given log file. In addition the analyzed log file can also be visualized using donut charts.

Figure 1-sample interface of replayable and graph widget

This paper described the new smart concept for log file analysis tools. The existing analysis tools that were mentioned did not contain this new concept. Most of the log file analysis tools use graphs for data visualization. This system was successfully implemented and it was evaluated by a number of users who work with log files.

This new concept will help log analysts, system analysts, data security teams as well as top level management to extract decisions about the system by analyzing the widgets to make predictions. Furthermore, analyzed data would be useful to collect non-trivial data for data mining and machine learning procedures.

As future work, the system could be enhanced to add features such as zooming and drill down method to customize graphs and identify a mechanism to filter data according to user requirements.

Keywords: *log file, log file analysis tools, D3.js, Replayable widget, Graph widget*