# Web Based Log Analyzer Tool

Rohinee Shinde Priyanka Sutar Ankita Kurdukar Nilopher Sawar

Pimpri Chinchwad College of Engineering,Nigdi 44

## Abstract

*Now a day's huge data is generated in applications logs and it's tedious to go through the entire log file to check for unexpected events. Also it is cumbersome to filter the required data from thousands of superfluous lines. It will be easier for administrator to monitor the logs, if important events (like Errors, Exceptions etc) in log files are presented in graphical format. Also important issues can be notified by an email. The CPU health can also be analyzed to check CPU cycles, memory usage and bandwidth usage. The main objective of the system is to scan the systems and applications log files to graphical representation to administrator.*

## Keywords

Log File, Log Analysis and CPU health

## 1. Introduction

IT infrastructure generate huge amount of logs every day and these machine generated logs have vital information that can provide powerful insights and network security intelligence into user behaviors, network anomalies, Exceptions, type of requests and frequency of requests etc. Generally this information is used to keep track of different events of the applications by analyzing the log file. The problem is that with so many events, identifying the crucial ones is very difficult. In a network there are number of systems and each system has number of applications. To analyze log and to extract meaningful information of each application on an individual system in the network is quite difficult. So there is requirement of a system that will provide automatic analysis of log files. So we have proposed a system WEB BASED LOG ANALYZER TOOL that will analyze the log files from the various applications in the network and gives graphical representation of required application to user. This application can also provide facility to send email alter whenever system goes in critical condition.

The main objectives are as follows:

- Analyze different types of exceptions generated.
- Keep track of different types of request, errors and their frequency.
- Provide alerts to the admin by email.
- To check health of system like CPU cycles, memory usage and bandwidth etc.
-

## 2. Related Work

A Toolkit for Event Analysis and Logging [1], this toolkit is designed as a pluggable processing pipeline that allows different components to use connectors to log events, have analyzers evaluate the events to get closer to the root cause, report and log these as alerts, and deliver the alerts to interested parties via filters and listeners.

Advanced Techniques for Analyzing Web Server Logs [2] describes the DAPHNE (Distributed Authoring and Publishing in a Hypertext and Network Environments) which provide solutions to analyze critical aspects of log files present in the server.

## 3. Architechture

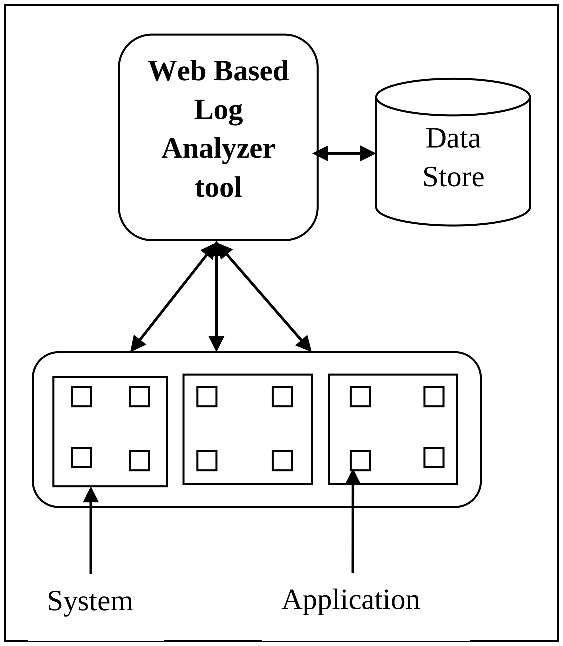
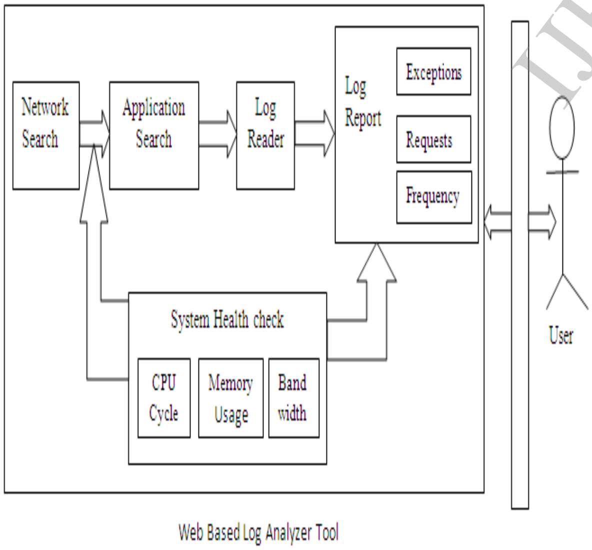


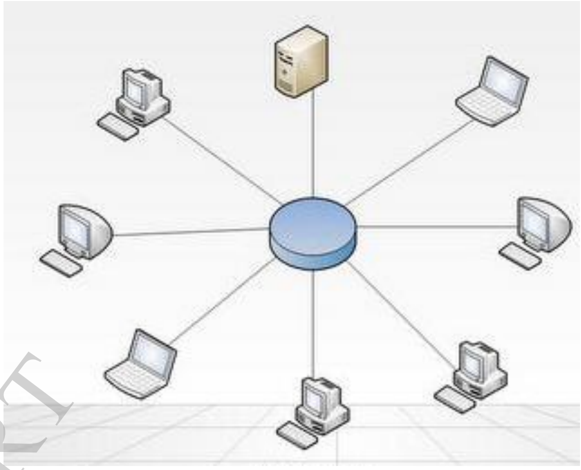
Fig 1.System Architecture



Web based log analyzer tool will scan the systems in the network. The system in turn Scan all the applications in the system. This information about system and application will get stored in the database. The log files from the applications will get scanned and vital information like exceptions occurred, type of requests and frequency of requests etc.will get converted in graphical from as per user request.

### 3.1 Scan Systems

Web based log analyzer tool will scan the systems in the network. Admin can add the systems or remove the systems from the network.



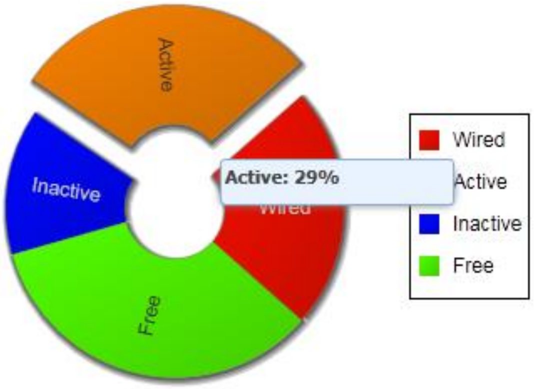
In this case the list of scanned systems will get updated. In addition administrator can search systems in the network. Each system will scan all the applications (web applications and desktop applications) running currently on the system.

### 3.2 Data Store

All the information related to system and its applications are stored in the database. Information about systems such as system id, system name, IP address etc and Information about applications such as application id, application name, type of application and date & time etc will get stored in database.

### 3.3 Analyze System Information

System information like CPU cycles, memory usage and bandwidth usage will get analyzed. System health will get analyzed in terms of CPU cycles, memory usage and bandwidth etc .System health information will get represented in graphical form.



### 3.4 Analyze Log Files

Log files in an application provide very crucial information such as date, time, IP address ,Location and exception related information like log levels of a particular application.

Following are the log levels.

**1. error**: the system is in distress, customers are probably being affected (or will soon be) and the fix probably requires human intervention. The "2AM rule" applies here- if you're on call, do you want to be woken up at 2AM if this condition happens? If yes, then log it as "error".

**2.warn**: an unexpected technical or business event happened, customers may be affected, but probably no immediate human intervention is required. On call people won't be called immediately, but support personnel will want to review these issues asap to understand what the impact is. Basically any issue that needs to be tracked but may not require immediate intervention.

**3.info**: things we want to see at high volume in case we need to forensically analyze an issue. System lifecycle events (system start, stop) go here. "Session" lifecycle events (login, logout, etc.) go here. Significant boundary events should be considered as well (e.g. database calls, remote API calls). Typical business exceptions can go here (e.g. login failed due to bad credentials). Any other event you think you'll need to see in production at high volume goes here.

**4.debug**: just about everything that doesn't make the "info" cut... any message that is helpful in tracking the flow through the system and isolating issues, especially during the development and QA phases. We use "debug" level logs for entry/exit of most non-trivial methods and marking interesting events and decision points inside methods.

**5. trace**: we don't use this often, but this would be for extremely detailed and potentially high volume logs that you don't typically want enabled even during normal development. Examples include dumping a full object hierarchy, logging some state during every iteration of a large loop, etc.

When a system is selected all the applications (desktop and web applications) will get scan for log files. Information from log files such as IP address, date & time, file path, exceptions, type of request and frequency of requests is extracted and analyzed.

### 3.5 Email Alert

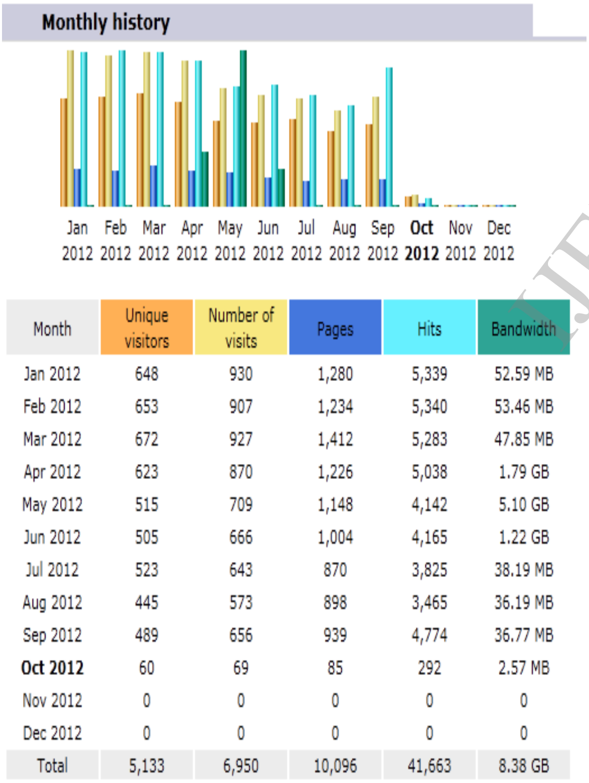
After analyzing the log files values extracted from the log files are compared with the predefined values and if result indicates any critical situation then an email alert is send to administrator.

### 3.6 Graphical Representation

If Administrator wants to check performance of an application he can select the required system and application instead of checking log files of each & every system. The information extracted from the log file will be converted into graphical representation. Information related to system health such as CPU cycles, memory usage and bandwidth is also represented in graphical form.

### 3.7 Report Generation

The information of log files can stored in form report such as pdf file for future use and user convenience.



| Month | Unique visitors | Number of visits | Pages | Hits | Bandwidth |
|---|---|---|---|---|---|
| Jan 2012 | 648 | 930 | 1,280 | 5,339 | 52.59 MB |
| Feb 2012 | 653 | 907 | 1,234 | 5,340 | 53.46 MB |
| Mar 2012 | 672 | 927 | 1,412 | 5,283 | 47.85 MB |
| Apr 2012 | 623 | 870 | 1,226 | 5,038 | 1.79 GB |
| May 2012 | 515 | 709 | 1,148 | 4,142 | 5.10 GB |
| Jun 2012 | 505 | 666 | 1,004 | 4,165 | 1.22 GB |
| Jul 2012 | 523 | 643 | 870 | 3,825 | 38.19 MB |
| Aug 2012 | 445 | 573 | 898 | 3,465 | 36.19 MB |
| Sep 2012 | 489 | 656 | 939 | 4,774 | 36.77 MB |
| **Oct 2012** | 60 | 69 | 85 | 292 | 2.57 MB |
| Nov 2012 | 0 | 0 | 0 | 0 | 0 |
| Dec 2012 | 0 | 0 | 0 | 0 | 0 |
| Total | 5,133 | 6,950 | 10,096 | 41,663 | 8.38 GB |

The above figure shows a sample report generation which is generated monthly but we can provide report on daily basis.

## 4. Advantages

1 .It is cumbersome to filter the required data from thousands of superfluous lines.

2. It will be easier for administrator to monitor the logs, if important events (like Errors, Exceptions etc) in log files are presented in graphical format.

3. Important issues can be notified by an email.

## 5. Applications

1. Administrator of network, server, system etc. can use this tool to have continuous monitoring of the system applications.

2. The applications can be prevented from being crashed.

## 6. Future Enhancement

The application can be further extended to be used in marketing domain such as to it can be used in business area to find the regular customer from different areas by analyzing the logs of the servers.

## 7. Conclusion

A log file is a file that contains a list of events, which have been "logged" by a computer. Whenever an application get stuck in between then it is required to check previous data related to the application in log files which is present in huge amount**.** The proposed system Web based log analyzer tool will automatically analyze the log file and give a graphical representation of different exception, type of requests and frequency of request etc present in the log file. As well as an email alert will be provided when a critical situation in the application occurs.

## 8. References

**1.** James Carey IBM Philip Sanders
IBM *A Toolkit for Event Analysis and Logging*
November 12-18, 2011

2. Ernst-Georg Haffner, Uwe Roth, Andreas Heuer, Thomas Engel, Christoph Meinel
*Advanced Techniques for Analyzing Web Server Logs*
November , 2000

3. http://awstats.sourceforge.net