

In Depth Analysis of Web Performance Testing Tools

Reenu Bhatia

Department of Computer Science
Himachal Pradesh University
Shimla, India.

Anita Ganpati

Department of Computer Science
Himachal Pradesh University
Shimla, India.

Abstract— Today, there is a huge shift towards web applications; therefore it is necessary to evaluate the performance of web applications. To evaluate a web application performance properly, significant knowledge of the web application on multiple levels is essential. Performance evaluation of a web application requires knowledge of its users and their aggregated behavior - how many users use the application, when, and which operations they perform, at what frequency. Load testing, is the most common kind performance testing which is sometimes referred as reliability test. The point of load testing is to prove that a system can sustain full operations under the peak load defined in the requirements. The peak load level is defined as some percentage of the breaking point assessed by an initial stress test. Load testing is usually performed with a constant virtual user count to further observe its reaction to increasing load. However this study evaluates performance testing tools namely Neoload, LoadImpact, Loadster, LoadUI and Webload. The basic objective of this study is to provide basic information about the tools on the basis of their properties and characteristics. An empirical study is also carried out on the basis of parameters namely response time, throughput, memory utilization, CPU utilization and hits per sec. From this study, it is concluded that these tools shows different behavior under different parameters. Webload is better in terms of response time and throughput. Neoload is better in terms of both memory utilization and CPU utilization. Therefore from the above observation we can conclude the Webload has better performance from LoadUI, LoadImpact, Loadster and Neoload.

Keywords- *LoadImpact, Loadster, LoadUI, Neoload, Performance Testing, Web Application Testing, Webload.*

I. INTRODUCTION

Performance testing can be defined as the process of determining the throughput, response time, resource utilization and other measurable attributes of a software system under a particular workload. Performance of web applications can be determined in terms of Availability, Response Time, Throughput, Utilization and Latency [9]. Performance testing can be carried out as a benchmark test or load test, or it can be used for capacity planning. Benchmark testing is performed with applications running on target systems (hardware and standard software) to get information about the highest load of users working with the application it can handle, to check whether it reaches the limits that were set in a contract, or it is used during the application design phase. This is a short-term test and can last up to a couple of hours [11].

Load testing the most common kind of performance test sometimes referred to as reliability test. Load tests (sometimes called soak tests) identify possible performance issues in a longer time period. Most common are memory issues that are often not detected during short-term testing. During this type of performance test, the application must be able to handle the generated load and its performance should not degrade over time [11].

In this paper, section I gives the introduction about load testing and section II describes the literature survey related to the study. Section III describes the introduction about the selected tools for the study. Section IV describes the objectives and scope of the study. Section V describes the research methodology. Section VI describes the comparative analysis of selected testing tools and section VII describes the conclusion and future scope of the work.

II. LITERATURE SURVEY

Monika Sharma et al. [10] gave a comparative study of JMeter, HP LoadRunner, WebLOAD and Grinder on the basis of parameters like Server Monitoring, Unlimited Load generation, ease of use, cost, etc. After comparison it is concluded that JMeter is best tool as it is free, having great load generation and easy user interface.

Manju Kaushik and Pratibha Fageria [5] conducted a comparative study on performance analysis of Neoload, WAPT, LoadUI on the basis of parameters like throughput, response time, number of hit pages, error rate, memory and CPU utilization etc.

Rigzin Angmo and Monika Sharma [1] gave a Performance Evaluation of Web Based Automation Testing Tools namely selenium webdriver and watir webdriver. Here the performance of these testing tools is evaluated and compared, and concluded that watir webddriver is suitable under specific situation, but selenium webdriver is better choice in various conditions like using domain specific language.

Cheng-hui Huang, and Huo Yan Chen [3] gave a Tool to Support Automated Testing for Web Application Scenario named WASATT (Web Application Scenario Automated Testing Tool), the tool support the automated testing for scenario of web-based applications.

Vandana Chandel et al. [2] has done a comparative study of testing tools: Apache JMeter and Load Runner which compare

these tools based on the criteria such as performance, speed, throughput and efficiency and concluded that JMeter is better tool to go forward with.

Harpreet Kaur and Gagan Gupta [4] conducted a comparative study of automated testing Tools: Selenium, Quick Test Professional and Testcomplete on the basis of their usability and effectiveness and concluded that one can select a testing tool based on the type of application need to be tested, budget and the efficiency required.

Vinita Malik and Mamta Gahlan [8] has given a comparative study of automated web testing tools of automated testing namely Quick Test Professional, Selenium, Watir and Sahi based on the criteria such as efforts involved with generating test scripts, capability to play back the scripts, result reports, speed and cost and concluded that QTP is the best tool among them all.

Rifa Nizam Khan and Shobhit Gupta [7] has given a comparative study of automated testing tools: Rational Functional Tester, Quick Test Professional, Silk Test and Loadrunner and determine their usability and efficiency and concluded QTP is a good tool.

Dipika Kelkar and Kavita Kandalgaonkar [6] has given an analysis and comparison of performance testing tools namely LoadRunner and JMeter and determine their accuracy of responses and recommend going ahead with HP LoadRunner as it is very stable and robust.

III. OVERVIEW OF PERFORMANCE TESTING TOOLS

A. Neoload

NeoLoad is a load and performance testing tool that realistically simulates user activity and monitors infrastructure behavior so you can eliminate bottlenecks in all your web and mobile applications [17]. This helps in improving and optimizing the performance of web application. This tool analysis the performance of the web application by increasing the traffic to the website and the performance under heavy load can be determined. The capacity of the application and the amount of users it can handle can be determined at the same time [14].

B. LoadImpact

LoadImpact is a load testing tool which is mainly used in the cloud-based services. This also helps in website optimization and improvising the working of any web application. This tool generates traffic to the website by simulating users so as to find the stress and maximum load it can work. This LoadImpact comprises of two main parts; the load testing tool and the page analyzer. The load testing can be divided into three types such as Fixed, Ramp up and Timeout. The page analyzer works similar to a browser and it gives information regarding the working and statistics of the website [14].

C. Loadster

Loadster is a full-featured load testing solution for websites, web apps, and web services. It's built for real web apps and

effortlessly handles cookies, user sessions, custom headers, dynamic form data and more. Loadster simulates individual user state and gathers stats for each virtual user separately [16].

D. LoadUI

LoadUI is hands down the easiest way to run a quick API load test, either against a single web service endpoint or based off of an existing functional API test created in SoapUI NG. LoadUI is used for test the speed and scalability of new changes to your APIs in minutes, not days, preview API performance behaviors before releasing to production environments and shift performance insights more to the left so developers build more reliable code [15].

E. Webload

Webload supports a wide range of web, mobile, and enterprise protocols and technologies. For example, HTTP/HTTPS, WebSocket, PUSH, AJAX, SOAP, HTML5 and many others [13]. WebLOAD's strengths are its ease of use with features like DOM-based recording/playback, automatic correlation and JavaScript scripting language. The tool supports large-scale performance testing with heavy user load and complex scenarios, and provides clear analysis on the functionality and performance of the web application [14].

IV. OBJECTIVE

This study evaluates performance testing tools namely Neoload, LoadImpact, Loadster, LoadUI, Webload. The basic objective of this study is to provides basic information about the tools on the basis of their properties and characteristics. An empirical study is carried out on the basis of parameters namely response time, throughput, memory utilization, CPU utilization and hits per sec.

V. RESEARCH METHODOLOGY

The research methodology follows theoretical approach that comprises literature survey, articles, books, research papers and internet. An experimental approach is also followed which evaluate the performance of the tools on the basis of set of parameters i. e. response time, throughput, memory utilization, CPU utilization and hits per sec.

VI. COMPARATIVE ANALYSIS OF PERFORMANCE TESTING TOOLS

This section represents a comparative analysis of the selected tools along with their observed results. The tests were conducted at the same instant of time at same network speed. In this study namely Neoload, LoadImpact, Loadster, LoadUI, Webload are evaluated. Table I represents basic information about selected load testing tools. The environment in which experiment is performed in Microsoft Windows 7, 32-bit with memory 4GB, processor Intel core i5 CPU 650 @3.20 GHz X 4 and disk 320 GB.

TABLE I. BASIC INFORMATION OF PERFORMANCE TESTING TOOLS

Sr.No.	Parameter	Neoload	LoadImpact	Loadster	LoadUI	Webload
--------	-----------	---------	------------	----------	--------	---------

1	Language Used	JAVA	JAVA	HTML	JAVA	-
2	Operating System	Microsoft Windows, Linux and Solaris	Microsoft Windows and Linux	Microsoft Windows	Cross Platform	Microsoft Windows and Linux
3	Protocol	JSON and SPDY	TLS, SSL	HTTP	REST, AMF, JMS, JDBC	HTTP / HTTPS
4	Development Year	1.0/first version in 2005	2008	-	2010	2010
5	Developer	French Company Netosys	Ragnar Lonn	-	Smartbear	RadView Software
6	Language Support	AJAX, .NET, J2EE, FLEX, SOAP.	XML, JAVA	HTML	JAVA, Groovy	Ajax, Adobe Flex, .NET, Oracle Forms, HTML5
7	Browser Support	Multi Browser	Firefox, IE, Chrome, Safari	Multi Browser	Multi Browser	Multi Browser
8	Tool Architecture	Controller and Load generator	Load testing tool and the page analyzer	Loadster Cloud Engine	Generator and Runner	Integrated development Environment and Webload Console

In Table I, basic information of selected performance testing tools is given. The table describes the language used, operating system, protocols, development year, developer, language support, browser support and tool architecture of these tools.

An experimental approach was carried out which evaluated the performance of the tools on the basis of Client side parameters and Server side parameters. Client side parameters include response time and throughput whereas Server side parameters include memory utilization, CPU utilization and hits per sec.

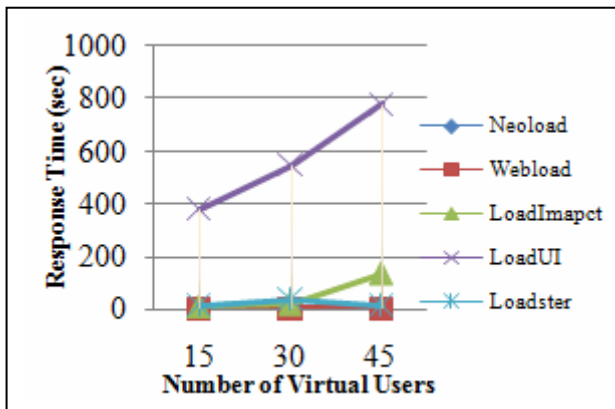


Figure 1. Response Time of Testing Tools

From Figure 1, it is depicted that the tool with the highest response time appears to be LoadUI. The tool having second highest response time appears to be LoadImpact. LoadUI shows the great difference in response time with varying number of users. Instead of LoadUI other tools i. e. Loadster, Neoload, Webload, LoadImpact shows little difference in response time. The tool with lowest response time is Webload. Therefore Webload has better performance from other tools in term of response time.

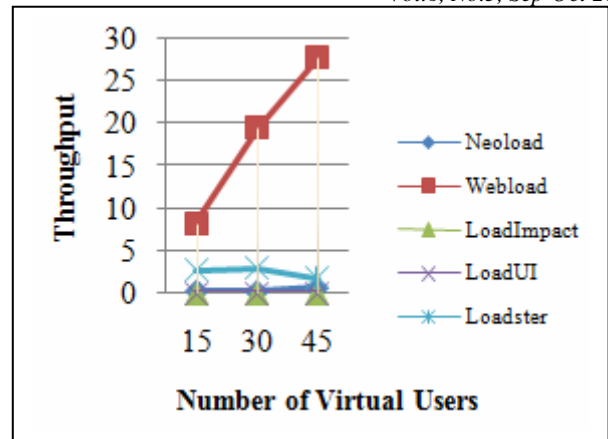


Figure 2. Throughput of Testing Tools

Throughput determines the number of requests per minute the server has processed. From Figure 2, it is depicted that the tool with the maximum throughput appears to be Webload. With the increase in number of users the throughput also increases. The tool having second highest throughput appears to be Loadster. The tool with lowest throughput is LoadImpact., therefore here Webload shows the better performance in terms of throughput.

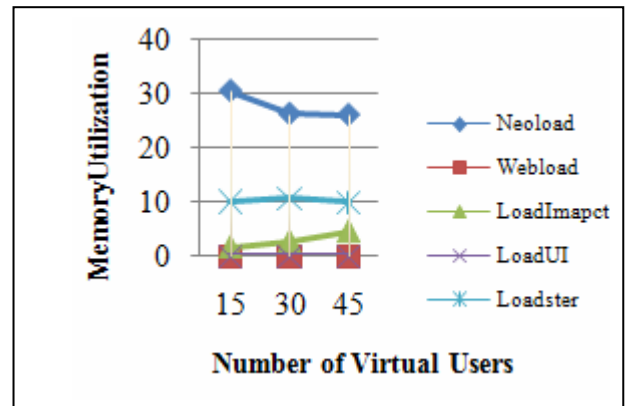


Figure 3. Memory Utilization of Testing Tools

From Figure 3, it is depicted that the tool with the maximum memory utilization appears to be Neoload. The tool having second highest memory utilization appears to be Loadster. The tool with lowest memory utilization is LoadImpact. Therefore here Neoload shows the better performance in terms of memory utilization.

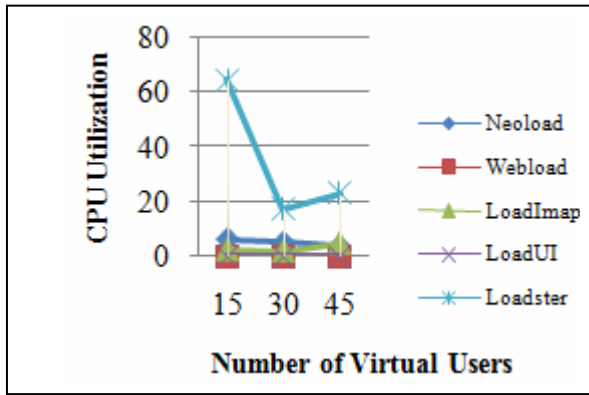


Figure 4. CPU Utilization of Testing Tools

From Figure 4, it is depicted that the tool with the maximum CPU utilization appears to be Loadster, but as the number of users increases from 20 virtual users to 50 virtual users the CPU utilization decreases. The tool having second highest memory utilization appears to be Neolaod. The tool with lowest memory utilization is LoadImpact. Therefore here Loadster shows the better performance in terms of memory utilization.

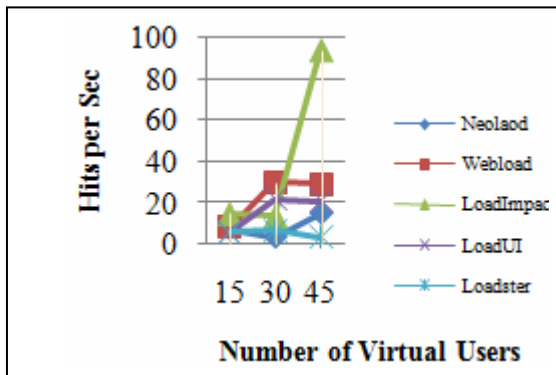


Figure 5. Hits per sec of Testing Tools

From Figure V, it is depicted that the tool with the highest hits per second appears to be LoadImpact. The tool having second highest hits per second appears to be Webload. The tool with lowest hits per second is Loadster. Therefore here Loadster shows the better performance in terms of hits per second.

From the above results it is concluded that these tools shows different behavior under different parameters. Webload is better in terms of response time and throughput. Neoload is better in terms of both memory utilization and CPU utilization. Therefore from the above observation we can conclude the Webload has better performance from LoadUI, LoadImpact, Loadster and Neoload.

VII. CONCLUSION AND FUTURE SCOPE

Performance evaluation of a web application requires knowledge of its users and their aggregated behavior - how

many users use the application, when, and which operations they perform, at what frequency In this paper performance testing tools namely Neoload, LoadImpact, Loadster, LoadUI, Webload are evaluated on the basis of properties and characteristics of the tools individually, since the tools only partially overlap in feature sets, and each one offers something the other ones don't. The tools are compared side by side according to the expectations and requirements set out for an ideal performance testing tool. From this observation it is concluded that these tools shows different behavior under different parameters. Webload is better in terms of response time and throughput. Neoload is better in terms of both memory utilization and CPU utilization. Therefore from the above observation we can conclude the Webload has better performance from LoadUI, LoadImpact, Loadster and Neoload. In future more work can be performed by developing other performance tests such as stress testing, spike testing, stability testing, etc.

REFERENCES

- [1] Angmo Rigzin, Sharma Monika," Performance Evaluation of Web Based Automation Testing Tools" IEEE 2014.
- [2] Chandel Vandana, Patial Shilpa and Guleria Sonal, "Comparative study of Testing Tools : Apache Jmeter and Load Runner" International Journal of Computing and Corporate research(IJCCR) Volume 3, Issue 3, 3May 2013.
- [3] Huang Cheng-hui, Chan Huo Yan," A Tool to Support Automated Testing for Web Application Scenario" IEEE International Conference on System, Man and Cybernetics, October 2006.
- [4] Kaur Harpreet, Gupta Gagan, "Comparative Study for Automated Testing Tools : Selenium, Quick Test Professional and TestComplete" International Journal of Engineering Research and Application (IJERA) Volume 3, Issue 5, Sep-Oct 2013.
- [5] Kaushik Manju, Fageria Pratibha," Performance Testing Tools : A Comparative Study " International Journal of Innovative Science Engineering & Technology (IJSET) Volume 1, Issue 4, June 2014.
- [6] Kelkar Dipika, Kandalgaoonkar Kavita, "Analysis and Comparison of Performance Testing Tools" International Journal of Advanced Research in Computer Engineering & Technology (IJARCE) Volume 4, Issue 5,May 2015.
- [7] Khan Rifa Nizam, Gupta Shobhit, "Comparative Study of Automated Testing Tools : Rational Functional Tester, Quick Test Professional, Silk Test and Load Runner" International Journal of Advanced Technology in Engineering and Science (IJATES) Volume 3, Issue 1, Feb 2015.

- [8] Malik Vinita, Gahlan Mamta, "Comparative Study of Automated Web Testing Tools" International Journal of Latest Trends in Engineering and Technology (IJLTET) Volume 6, Issue 3, January 2016.
- [9] Molyneaus Ian, "The Art of Application Performance Testing" O'REILLY Second Edition.
- [10] Sharma Monika, Iyer Vaishnavi S., Subramanian Sugandhi, Shetty Abhinandhan, "A Comparative Study on Load Testing Tools" International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE) Volume 4, Issue 2, February 2016.
- [11] Wartiak Rastislav, "Performance Testing Tools for Web Applications" Charles University in Prague.
- [12] <http://www.qatestingtools.com/loadster/loadsterperformance> Accessed On: 02 June, 2016.
- [13] <http://www.radview.com/webload-download/> Accessed On: 26 May, 2016.
- [14] <http://www.softwaretestinghelp.com/performance-testing-tools-load-testing-tools/> Accessed On: 16 July, 2016.
- [15] <https://smartbear.com/product/ready-api/loadui/overview/> Accessed On: 12 June, 2016.
- [16] <https://www.loadsterperformance.com/> Accessed On: 30 July, 2016.
- [17] <https://www.neotys.com/neoload/overview> Accessed On: 10 August, 2016.