A Brief Survey on Web Application Performance Testing Tools Literature Review

Isha Arora  
*M.Tech Scholar, Department of Computer Science and Engineering, PIET, PANIPAT, INDIA*

Vikram Bali  
*Department of Computer Science and Engineering, PIET, PANIPAT, INDIA*

Abstract - Web Testing is the complete testing of a web based system before it goes live on the internet. Due to increase in number of websites, the demand is for accurate, faster, continuous & attractive access to web content. So, before publishing it online, Testing process is required which includes basic functionalities of the site, User Interface compatibility, its accessibility, effect of traffic on server etc. This leads to the requirement of one handy and intelligent tool or application which provides these facilities of testing in an easy & automated way such that time & cost are minimized and testing is no more a headache. This paper presents an extensive Literature Survey of performance Automation Testing tools. This paper is first of two, the second is going on to consider the framework of Performance Automation tool that will carry its own new set of features which are desired by professionals and currently unavailable.

Keywords: Software testing, Automation testing, Performance testing, Testing tools, Non functional testing

I. INTRODUCTION

Software Testing is the process used to help identify the correctness, completeness, security, and quality of developed computer software. Testing is a process of technical investigation, performed on behalf of stakeholders, that is intended to reveal quality related information about the product with respect to the actual functioning of the product, it includes the process of executing a program or application with the intent of finding errors. Testing a system is harder than put it in the face of new risks introduced by the ever-increasing complexity of software and hardware because without programs that properly work, the system will never process information and produce the output for which it was designed, therefore testing procedures should be established, testing roles should be applied and testing data that can test the limits of the program should be created. But performing tests using the manual traditional way is a tedious, time consuming and costly job which has led to the development of testing tools. These tools may be the only practical way to conduct tests easily and efficiently. Also they can help in avoiding trouble, providing vital information, and they can enable an organization to take new opportunities with greater performance and strength.

There are many types of software testing which include unit testing, integration testing, system testing, acceptance testing, functional testing and non-functional testing.

Performance testing is a type of non-functional testing that includes other types of tests which are:

- **Load Testing**: Testing an application against a requested number of users (real-world load). The objective is to determine whether the site or system can sustain this requested number of users with acceptable response times.

- **Capacity Testing**: Testing to determine the maximum number of concurrent users an application can manage. The objective is to benchmark the maximum loads of concurrent users a site or a system can sustain before experiencing system failure.

- **Volume or Stress Testing**: Load testing over an extended period of time to find stress points. It includes testing beyond normal operational capacity often to a breaking point to determine the stability of a given system or application. The objective is to validate an application’s stability and reliability.
• Spike Testing: spike Testing is done by suddenly increasing the number of load generated by users by a very large amount and observing the behavior of system. Here, the goal is to determine the performance degrade, system failure.

• Endurance Testing: This type of testing is used to check that the system can withstand the load for a long or large number of transactions. For eg: Test will be performed with definite set of concurrent users for a prolonged period of time i.e 2 to 3 days.

II. OVERVIEW OF PERFORMANCE TESTING:

Application load testing is the measure of an entire web application’s ability to sustain a number of simultaneous users and/or transactions, while maintaining adequate response times. Because it is comprehensive, load testing is the only way to accurately test the end-to-end performance of a website or system prior to going live. Application stress/load testing enables developers to isolate bottlenecks and find errors in any component of the infrastructure.

Two common methods for implementing this process are manual and automated testing. Manual testing has several built-in challenges, such as determining how to:
- Emulate hundreds of thousands of manual users that will interact with the application to generate load.
- Coordinate the operations of users.
- Measure response times.
- Repeat tests in a consistent way.
- Compare results.

Because load testing is iterative in nature, the tester must identify performance problems, tune the system, and retest to ensure that tuning has had a positive impact — countless times. For this reason, manual testing is not a very practical option.

With automated stress/load testing tools, tests can be easily rerun and the results automatically measured. In this way, automated testing tools provide a more cost-effective and efficient solution than their manual counterparts. Plus, they minimize the risk of human error during testing.

Today, automated stress/load testing is the preferred choice for stress/load testing a web application or a system. Most testing tools typically use three major components to execute a test. These include:
- A control console: which organizes, drives, and manages the stress/load test.
- Virtual users: These are the processes which simulates the no of users or connections to your server application.
- Load Servers: These are used to run the virtual users request.

Using these components, automated stress/load testing tools can:
- Replace manual testers with automated virtual users.
- Simultaneously run many virtual users on a single load-generator.
- Automatically measure transaction response times.
- Easily repeat load scenarios to validate design and performance changes.
- Easily generate reports.

There is significant literature describing work on software performance modeling[10], [12], [13], software functionality testing[9], [11], software testing strategies[15], [19], software testing techniques[16] and automated testing[17], [21], [23], although all these subjects are related, this work does not directly consider how to choose the best tool among the present performance Testing tools according to the proficiency level of tool for testing the performance of software, which is the primary goal of our research. For this purpose we have performed an
extensive Literature survey which can be useful for opting the best tool among the list of present Performance Testing Tools.

III. RELATED WORK (LITERATURE REVIEW):

Software testing is any activity aimed at evaluating an attribute or capacity of a program or a system and determining that meets its required results[17]. Although crucial to software quality and widely deployed by programmers and testers, software testing still remains an art, due to limited understanding of the principles of software [15], [16], [19]. The difficulty in software testing stems from the complexity of software. The purpose of testing can be quality assurance, verification and validation, or reliability estimation [18], [20], [23]. Software performance testing is a major area of software testing. Software testing is a trade-off between budget, time and quality [14], [22].

A search of the published literature has following papers such as Eljona Proko and Ilia Ninka[2], gives a discussion that performance tools helps software developer to find out bottleneck in performance of the system as well as to choose a good platform for designing web application.

Sharmila and Dr. E.Ramadevi[3] Paper, presents performance testing concepts, objectives, goals, types and available tools for testing web applications performance. They has examined Performance testing is a used to determine the responsiveness, throughput, reliability, and scalability of a system under a given workload.

F.I. Vokolos and E.J. Weyuker [14], gives a discussion on approaches to software performance testing. A case study describes the experience that approaches used for testing the performance of a system used as a gateway in a large industrial client/server transaction processing application. An Approach and Case Study by Volkos and Weyuker[6], an examination of performance testing issues and description to an approach to address some of these issues for certain classes of software systems is addressed. This work is based on experience gathered while testing the performance of different industrial software systems. The paper classifies the objectives of performance testing and describes a technique for creating appropriate workloads needed for the testing. Also it looks at the role of requirements and specifications in successful performance testing. A case study is presented describing the results of some recent work that was done to test the performance of an existing system. The approach used earlier technique that was developed for performance testing, as described in "Deriving Workloads for Performance Testing"[7] paper. Briand, Labiche and Shousha[24], suggest that during development, some Performance Analysis should be performed to determine whether designed tasks will likely meet their deadlines or not.

Korel and Al-Yami [27], talk about automating the regression testing process which involves testing the modified program with test cases in order to establish the confidence that the program will perform according to the modified specification. During software maintenance the existing software is modified. There are three types of software maintenance: corrective maintenance is performed to correct an error that has been discovered in some part of the software. Adaptive maintenance is usually performed when the software is modified to ensure its compatibility with the new environment in which it will operate. Perfective maintenance is performed to add new features to the software or to improve performance of the software. Also Korel and Al-Yami proposed a regression test generation method to automatically generate test data that causes a modified program under test to behave incorrectly.

Yang and Pollock[25], introducing a testing tool for structural load testing which takes a program as input, and automatically determines whether that program needs to be load tested, and if so, automatically generates test data for structural load testing of the program. There exist many structural testing methods with the main goal of generating test data for executing all statements, branches, definition-use pairs, or paths of a program at least once to discover load sensitive faults, which are programming errors with which the program executes successfully if executed for a short time or with a small workload, but causes the program to fail when it is executed under a heavy load or over a long period of time.

Shams, Krishnamurthy[26] describes poor performance can adversely impact the profitability of enterprises that rely on the web applications. As a result, effective performance testing techniques are essential for understanding
whether a web-based system will meet its performance objectives when deployed in the real world. The workload of a web-based system has to be characterized in terms of sessions; a session being a sequence of inter-dependent requests submitted by a single user. Dependencies arise because some requests depend on the responses of earlier requests in a session. To exercise application functions in a representative manner, these dependencies should be reflected in the synthetic workloads used to test web-based systems. This makes performance testing a challenge for these systems. A model-based approach to address this problem is proposed. The proposed model uses an application model that captures the dependencies for a Web-based system under study.

Also the recent growth of the Internet and its increased heterogeneity has introduced new failure modes and added complexity to protocol design and testing. In addition, the advent of multicast application has introduced new challenges of qualitatively different nature than the traditional point-to-point protocols. A method for automatic synthesis of worst and best case scenarios for these new protocols was addressed by Helmy, Gupta and Estrin[28]. The algorithm uses timing semantics to handle end-to-end delays and address performance criteria.

A Monika Sharma and Rigzin Angmo[4], have discussed various web automation testing tools which will help us to understand the automation testing as well as the tools available for automation testing. A variety of web based systems and applications are tested by automation testing tools. The automation testing script is used in test automation. To choose the best tool for a task, various issues like ease of integration should be considered and weighed against the cost and performance. Also the tool needs to be compatible with the design and implementation of an application.

Pooja Ahlawat and Sanjay Tyagi[5], talk about three popular load testing tools i.e WAPT, LOADUI and LOADSTER and their comparison has been made in terms of average response time and optimal response rate. Results of the comparison will help in adoption and usage of these tools. Lalitha Shastry, Asha Gowda Karegowda and Latha Raju[1], gives a brief description of 15 non-functional tools with their year of release and platform/framework. A recent survey revealed that there is tremendous increase in development of performance testing tools because users are not tolerant of delays of more than 10 seconds.

IV. PERFORMANCE TESTING TOOLS

This section presents a list of most widely used performance testing tools for measuring web application performance and load stress capacity. These load testing tools will ensure your application performance in peak traffic and extreme stress conditions. The list includes open source as well as licensed performance testing tools. Now-a-days all licensed tools are available with free trial version so that you can estimate which is the best tool for your needs.

Httpperf : Httpperf is a high performance testing tool for measuring and analysing the performance of any web services and web applications. This is mainly used to test the HTTP servers and its performance. The main objective of this testing tool is to count the number of responses generated from this particular server. This generates the HTTP GET requests from the server which helps in summarizing the overall performance of the server. Through this tool, we will be able to conclude the rate at which the response is sent from each server and thereby the efficiency can be calculated. It runs on Windows and Linux platform.

NeoLoad : NeoLoad is a load & performance testing software designed to improve the quality of your web and mobile applications by realistically simulating users and analyzing your servers behavior. This helps you in improving and optimizing the performance of your web application. This tool analysis the performance of the web application by increasing the traffic to the website and then performance under heavy load can be determined. You can get to know the capacity of the application and the amount of users it can handle at the same time. NeoLoad is integrated with the on-demand Neotys Cloud Platform. This tool was developed by a French company named as Netosys and it was written in JAVA. It is available in two different languages; English and French.

QTest : A web load testing tool analyses the application completely and accurately. It supports all Windows platforms. Its original User interface (UI) is easy to use and understand, and used as an OnDemand hosted solution or an OnPremise application. It is supported by all Windows Platform.
OpenSTA: OpenSTA stands for Open System Testing Architecture. OpenSTA is a distributed testing architecture that enables you to create and run performance Tests to evaluate Web Application Environments (WAEs) and production systems. It can be employed at all stages of WAE development as well as being used to continuously monitor system performance once a WAE goes live. It is use to develop load Tests that include an HTTP/S load element, known as Scripts, to help assess the performance of WAEs during development, and to create Collector-only Tests that monitor and record performance data from live WAEs within a production environment. OpenSTA enables you to run Tests against the same target system within both load testing and production monitoring scenarios. This means that you can directly compare the performance of the target system within these two environments. This Tool works only on Windows platform.

LoadStorm: LoadStorm is the cheapest available performance and load testing tool. In this tool, we have the option of creating our own test plans, testing criteria and testing scenario. Through this tool, you can bring an end to all the expensive performance testing tools. The cloud infrastructure is used in this tool, which enables you to send huge amount of requests per second. There are thousands of servers available around the world for this software. They are proudly known as the lowest cloud load testing tool. This tool works with Microsoft Windows.

LoadImpact: LoadImpact is a load testing tool which is mainly used in the cloud-based services. This also helps in website optimization and improving the working of any web application. This tool generates traffic to the website by simulating users so that it can work under maximum stress and load. 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. This works well on Windows OS and Linux.

QEngine (Manage Engine): QEngine (Manage Engine) is a most common and easy-to-use automated testing tool helping in performance testing and load testing of your web applications. Many developers find it to be the most simple and easy tool to use for finding out any leakage in their web services or websites. The key important feature of this testing tool is its ability to perform remote testing of web services from any geographical location. Other than that, QEngine (Manage Engine) also offers various other testing options such as functionality testing, compatibility testing, stress testing, load testing and regression testing. This automated testing tool has the capacity to generate and simulate lot of users so that the performance can be well analysed during the maximum load. This tool works with Microsoft Windows and Linux.

LoadUI: LoadUI is another open source and load testing software used for measuring the performance of the web applications. This tool works effectively when it is integrated with the functional testing tool soapUI. LoadUI is the most flexible and interactive testing tools. This allows you to create, configure and update your tests while the application is being tested. It also gives facility to the the user with a drag and drop experience. This is not a static performance tool. The advanced analysis and report generating features allows you to examine the actual performance by introducing new data even while the application is being tested. You need not bother to restart the LoadUI each and every time you modify or change the application. It automatically gets updated in the interface. This tool is an open source application, so it is available for free and everyone can have the easy access to its full source code.

Testing Anywhere: Testing Anywhere is an automated testing tool which can be employed for testing the performance of any web sites, web applications or any other objects. Many developers and testers make use of this tool to find out any bottlenecks in their web application and rectify them accordingly. It is a powerful tool which can test any application automatically. This testing tool comes along with a built in editor which allows the users to edit the testing criteria according to their needs. This tool works well on Windows.

Loadster: Loadster is full featured solution for websites, web applications and web services. It simulates individual user state and gathers report for each virtual user separately. Load test your web pages for an increasing period of time in order to estimate your web application can able to handle thousands of concurrent users. This tool is best to
identify performance bottlenecks in your application. This tool works well with Windows 7/Vista/XP.

LoadRunner: This is a HP product which can be used as a performance testing tool. This can be bought as a HP product from its HP software division. Also, it is very much useful in understanding and determining the performance and outcome of the system when there is actual load. HP LoadRunner can simulate thousands of concurrent users to put the application through the heavy real-life user loads, while collecting information from key infrastructure components (Web servers, database servers, etc). The results can then be analyzed in detail to explore the reasons for particular behavior. This tool enables you to gather all the required information with respect to the performance and also based on the infrastructure. The LoadRunner consist of different tools such as Virtual User Generator, Controller, Load Generator and Analysis.

WAPT: WAPT is a load and stress testing tool that provides an easy-to-use and cost-effective way to test any website, including business applications, mobile sites etc. With the help of this tool you can create load tests within minutes. A few clicks are required to make thousands virtual users execute concurrent sessions against your website. It works with secure HTTPS pages and all types of authentication. Descriptive graphs and reports will let you analyze the performance characteristics of your system components under various load conditions. This is considered to be the best cost effective tool for analyzing the performance of the web services. The WAPT tools can tests the web application on its compatibility with browser and operating system. It is also used for testing the compatibility with the window application in certain cases.

TestStudio: The Telerik Test Studio load testing tool enables you to get a better understanding of how your website might perform if visited by a large number of simultaneous users. This helps you assess whether or not your web apps meet business needs for availability and user satisfaction. Test Studio makes it easy for you to get started and find the data you need for informed decisions. You are also empowered with the flexibility to create elaborate, complex load scenarios to meet your most demanding needs. This tool is compatible with all versions of browsers.

Apache JMeter: It is open source load testing tool. It is a Java platform application. It is mainly considered as a performance testing tool. In addition to the load test plan, you can also create a functional test plan. It's full multi-threading framework allows concurrent sampling by many threads and simultaneous sampling of different functions by separate thread groups. Initially, it was introduced for testing the web applications, but later its scope had widened. JMeter can load and performance test many different server types: Web - HTTP, HTTPS, SOAP, Database via JDBC, LDAP, JMS, Mail - POP3. It works under Unix and Windows OS.

WebLOAD: It is load and performance testing tool for web applications. It is useful tool in performing load and stress testing on any internet application using Ajax, Adobe Flex, .NET, Oracle Forms, HTML5 and many more technologies. We can generate load from the cloud and on-premises machines. WebLOAD’s strengths are that it is easy to use with features like DOM-based recording/playback, automatic correlation and JavaScript scripting language. This tool supports large-scale performance testing with heavy user load and complex scenarios, and provides clear results on the functionality and performance of the web application.

Rational Performance Tester: The Rational performance tester is an automated performance testing tool which can be used for a web application or a server based application where there is a process of input and output is involved. This tool creates a demo of the original transaction process between the user and the web service. By the end of it all the statistical information are gathered and they are analysed to increase the efficiency. Any leakage in the website or the server can be identified and rectified immediately with the help of this tool. This tool can be the best option in building an effective and error free cloud computing service. This tool works well on Windows and Linux platform.

CloudTest: CloudTest is a performance testing tool for the cloud computers. The users or the developers can use the cloud platform as their virtual testing lab. The developers can carry out their performance or load testing in the cloud platform in the cost effective way through this CloudTest tool. This CloudTest has the capacity to enable number of users to use the website at the same time. It also increases the traffic of the website to know the actual performance
under stress and heavy load. They provide many services for testing the websites and other web applications and now they also help in testing the mobile applications. It runs on Windows, Linux and Mac OS.

Appvance: It is a Next Generation App Performance Testing Platform. Appvance PerformanceCloud is a broad-scale platform targeted at enterprise clients which can fully exercise apps from beginning to end. The platform is the fastest way to surface performance bottlenecks in modern applications. A third party study states that QA productivity of Appvance is much higher than the LoadRunner. It is Compatible with Selenium, JMeter, Oracle Forms, Flash/Flex and more and can test up to 10M concurrent users. It runs in private or public cloud.

Table 1 provides a overview of the existing performance testing tools & summarizes their main characteristics. For more detail information regarding the tools, please refer directly to the references section.

<table>
<thead>
<tr>
<th>S.No</th>
<th>Name of the Tool</th>
<th>Trademark of the product</th>
<th>Year of Release</th>
<th>Developer(s)</th>
<th>Platform</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Httpperf</td>
<td><img src="hp.png" alt="hp" /></td>
<td>2000</td>
<td>David Mosberger at Hewlett- Packard</td>
<td>Windows and Linux</td>
</tr>
<tr>
<td>2.</td>
<td>Neoload(1.0)</td>
<td><img src="neoload.png" alt="neoload" /></td>
<td>2005</td>
<td>Developed by Neotys</td>
<td>Microsoft Windows, Linux, Solaris</td>
</tr>
<tr>
<td>3.</td>
<td>QTest</td>
<td><img src="qtest.png" alt="qtest" /></td>
<td>2005</td>
<td>Developed by Quotium Technologies</td>
<td>All Windows Platform</td>
</tr>
<tr>
<td>4.</td>
<td>OpenSTA(1.4.4)</td>
<td><img src="opensta.png" alt="opensta" /></td>
<td>2007</td>
<td>Developed by CyRANO</td>
<td>Windows</td>
</tr>
<tr>
<td>5.</td>
<td>LoadStorm(2011)</td>
<td><img src="loadstorm.png" alt="loadstorm" /></td>
<td>2008</td>
<td>Developed by LoadStorm- cloud based Testing Tool</td>
<td>Windows OS</td>
</tr>
<tr>
<td>6.</td>
<td>LoadImpact</td>
<td><img src="loadimpact.png" alt="loadimpact" /></td>
<td>2009</td>
<td>Developed by LoadImpact</td>
<td>Windows OS/ Linux</td>
</tr>
<tr>
<td>7.</td>
<td>QEngine(7.1)</td>
<td><img src="qengine.png" alt="qengine" /></td>
<td>2009</td>
<td>Manage Engine</td>
<td>Windows &amp; Linux</td>
</tr>
<tr>
<td>8.</td>
<td>LoadUI(1.0 – beta)</td>
<td><img src="loadui.png" alt="loadui" /></td>
<td>2010</td>
<td>Smartbear Software</td>
<td>Windows XP/ Vista/7</td>
</tr>
<tr>
<td></td>
<td>Tool Name</td>
<td>Version</td>
<td>Year</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>--------------------------------</td>
<td>-----------</td>
<td>------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>10.</td>
<td>Loadster</td>
<td></td>
<td>2011</td>
<td>Developed by Loadster</td>
<td></td>
</tr>
<tr>
<td>11.</td>
<td>LoadRunner(11.50)</td>
<td></td>
<td>2012</td>
<td>(Hewlett-Packard) HP Software Division</td>
<td></td>
</tr>
<tr>
<td>12.</td>
<td>WAPT(3.0)</td>
<td></td>
<td>2012</td>
<td>Developed by SoftLogic</td>
<td></td>
</tr>
<tr>
<td>13.</td>
<td>Test Studio</td>
<td></td>
<td>2012</td>
<td>Developed by Telerik Test Studio supports test execution in Internet Explorer, Firefox, Safari and Chrome.</td>
<td></td>
</tr>
<tr>
<td>14.</td>
<td>Apache JMeter(2.10)</td>
<td></td>
<td>2013</td>
<td>Stefano Mazzocchi – Developer of JMeter of the Apache Software Foundation Java 6+</td>
<td></td>
</tr>
<tr>
<td>15.</td>
<td>WebLOAD(10.1)</td>
<td></td>
<td>2013</td>
<td>Radview</td>
<td></td>
</tr>
<tr>
<td>16.</td>
<td>Rational Performance Tester(8.5.1)</td>
<td></td>
<td>2013</td>
<td>Developed by IBM Windows/ Linux</td>
<td></td>
</tr>
<tr>
<td>17.</td>
<td>CloudTest(Soasta 49)</td>
<td></td>
<td>2013</td>
<td>Developed by Soasta Windows, Linux and Mac OS</td>
<td></td>
</tr>
</tbody>
</table>

V. SUMMARY & CONCLUSION

In this paper we have done extensive literature survey of performance Testing Tools. In this research study, 18 Performance testing tools were found to be significant contributions that can be analysed to opt the right performance Testing Tool according to application requirement. This study has highlighted various main features of software testing tools which are essential for opting the right tool at right time. In future work the author is going to propose the framework of Performance Automation tool that will carry its own new set of features which are desired by professionals and currently unavailable.
REFERENCES:


[31] https://jmeter.apache.org/download_jmeter.cgi


[34] http://www.loadui.org/


[38] http://www.quotium.com/qtest