Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 4, Issue. 1, January 2015, pg.433 – 442

RESEARCH ARTICLE

A Comparative Study and Analysis of Web Service Testing Tools

Ravi Kumar
Research Scholar
Department of Computer Science
Himachal Pradesh University Shimla, India

A.J Singh
Professor
Department of Computer Science
Himachal Pradesh University Shimla, India

Abstract

Software testing in present era is the process of validating and verifying the correctness of software. Automated testing tool enables the developer and tester to automate the whole process of testing in software development life cycle (SDLC). Testing is very important phase of SDLC where the software is examined properly and modifications are proposed. Thus testing is necessary for quality of service provided by software. Web service is widely used concept now a days and less literature is available regarding web service performance and SOAP messaging. The objective of this paper is to conduct the comparative study of automated tool for web services as a leading tool in black box test automation. This study will help in the promotion and usage of various open source web service tool toward performance of real time network using quality of service(QOS) provided by these tools. Further in this research paper the evaluation and comparison of six automated software testing tools is done to determine their usability and effectiveness.

KEYWORDS: Software Testing, SDLC, Web Services, Automated Testing, QOS, Black Box Testing.

I. Introduction

The main purpose of software testing is to evaluate an attribute or capacity of program or product and to determine that it satisfies its quality (QOS). The testing of software also include the testing of software quality factors like usability, efficiency, reliability, security, capability, maintainability, compatibility and portability, etc [1]. Software Testing identify faults, which when removed increases the software quality which intern increases the reliability of the

software. It is the process of analyzing and evaluating components of the system or a system itself by manual or automatic means to verify that specified requirements are satisfied. In Software testing the difference between expected and actual results is analyzed [2].

There are two ways of testing that are manual or automation. In manual testing the software is tested manually by the tester for identifying the defects. To achieve this written test plan is followed that guides them through a group of important test cases. Manual testing requires great efforts and programming skills, takes much time, and some errors remain undiscovered [3]. Automation testing overcome most of the problems occurred in manual testing. The tester can perform testing with and without the knowledge of the inside details of the software module under test. In white box testing input is given to the system and it is analyzed that how the input is processed to generate the desired output. Black box testing involves testing of software based upon expected output without knowing the internal structure or coding of the program [4].

Web services can furnish various operations from simple request to a complex one. These can also be described as the software component which can be accessed through various programming interfaces. Such interfaces are specified in an eXtensible Markup Language (XML) format called Web Services Description Language (WSDL). These are accompanied with their input and output parameters can be taken as reference for testing by service interface by means of black box testing. Quality of service is a prime key for developing service-based software systems and their testing is necessary for evaluating the functional performance, reliability and correctness of services [5].

S. Hussain et al. have concluded that several studies are available which has compared various web services testing tools from functionality and features supported by them. Hence they are needed to be thoroughly tested before deployment [6]. Web services technology is heterogeneous. The performance requirement conformance is most important criteria for evaluating the system and is directly proportional to the trust of service user. Several open source as well as freeware testing tools, are available in the market that supports various features and functionality.

Quality of Service (QoS) provided by each dependent on parameters such as response time, throughput, bytes processed, etc. Response time is an interval of time between request and first response that is received by the user. Study is based on Simple Object Access Protocol that defines a protocol specification which is used to exchange structural information over a computer network. It is used for implementing the web services and relies on XML for its message format.

The software testing tools can be compared on the basis of parameters such application supported, programming language, operating support, platform independence, version detail. In this paper we have planned to run the test cases written for the temperature conversion web service.

The organization of this paper consists of following sections: Section I Lays the Basis of The Study, Section II Provides an overview of Testing Tools considered for study, In Section III

Comparative Study of the Selected Tools has been given. Section IV describes the result and discussion and Section V Concludes the study along with scope for future work.

II. Testing Tools: A Brief Overview of Selected Tools

Software testing is an important to determine the quality of the software. The main aim of testing is detection of errors, verification and validation in order to find the faults and fix them for improving the quality of the software products [7]. Quality of the product is evaluated by comparing the observed test results with expected results. Testing Tools automate the process of testing and are targeted to specific test domain. The domain may be performance, functional, security or exceptional testing etc. Tools that support functional testing are used to test the web applications that involve the GUI. Various functional testing tools are available for testing the web application GUI objects and functionality automatically [8].

Test tool enables the testers to create, execute and manage test for a particular domain, maintained for specific test for a particular application. For this research four open source web service testing tools such as Apache Jmeter, Soapui Pro, Wcf Storm, Wizdl and two freeware web service testing tools SOA Cleaner and SOAPSonar Personal have been used to evaluate and validate the testing tools.

A. Apache Jmeter

Apache Jmeter [9] is developed by Apache Software Foundation (ASF). Project that can be used as a load testing tool for analyzing and measuring the performance of a variety of services, with a focus on web applications. Apache Jmeter might be used as a test tool for HTTP, LDAP, FTP, Web services, JMS, generic TCP connections and JDBC database connections. It can also be used for some functional testing. Jmeter architecture is based on plug-in. Its other features are implemented with plug-ins. Off-site developers can easily extend Jmeter with custom plug-in.

B. Soapui Pro

Soapui Pro[10] developed by Smart Bear under the General Public License (GNU) is an open source web service testing tool based on java work. It supports Mac, windows and Unix operating system (cross platform). Its GUI is easy-to-use that makes it simple to work with Soap and Rest based web services. Soapui pro offers more usability and efficiency. It contained everything that existed in Soapui and added productivity and time saving features.

C. Wcf Storm

Wcf Storm [11] is a open-source and freely available tool for testing web services. Storm was developed by Erik Araojo. Its source code was written in F# language. Wcf Storm allows testing web services written using technologies like .Net, Java, etc. Storm supports dynamic invocation for those methods that has input parameters of complicated data types. Raw soap requests can be efficiently edited and manipulated by it. Its graphical user interface is simple and easy to use.

More than one web services can be tested concurrently for saving time and accelerating testing schedule.

D. Wizdl

Wizdl [12] is a .NET utility written in C# that allows you to quickly import and test web services within the comfort of a Windows Forms GUI. The complex web services that take arrays and nested objects as parameters can be called by it easily. The tool provides the facility of storing data in XML file format which can be later used for regression testing.

E. SOA Cleaner

SOA Cleaner [13] is an open source web service tool developed by Xyrow. It is written n dot net and provides GUI platform to enter web service description language to test web service. SOA Cleaner also supports REST testing. The main benefits of SOA Cleaner it is simple to use without the need of coding knowledge. SOA Cleaner supports .NET and Java framework. SOA Cleaner offers more efficiency and usability.

F. SOAPSonar Personal

SOAPSonar Personal [14] is developed by Crosscheck Networks. SOAPSonar Personal edition is available for free. It provides simple testing support for SOAP, XML and REST based services. SOAPSonar Personal is easy to implement and use. It requires no coding knowledge provides testing support for functional, performance and security testing. The tool provides the facility of storing data in XML file format which can be later used for regression testing. Reports can be generated efficiently.

Table 1 shows comparison of selected tools on the basis of application support, programming language, OS support, license etc.

TABLE 1
ANALYSIS OF SELECTED TOOLS ON THE BASIS OF PLATFORM, VERSION AND USAGES

Sr. No.	Tool Name	Application Support	Programming language / Framework	OS Support	License	Developer	Website
1	Apache Jmeter	Web services /Web applications	Java, JRE1.5+	Cross Platform	Apache License 2	Apache Software foundation	http://jmeter.apache.org/
2	Soapui pro	Web services	.Net, Java, JRE1.6+	Cross Platform	GNU/ LGPL 2.1	SmartBear Software	http://www.soapui.org/
3	Wftstorm	Web services	F#, .NET	MS Windows8/ 7/vista/XP/2000/NT	BSD	Eric Araojo	http://www.wcfstorm.com/wcf/home.aspx

4	Wizdl	Web services	C#, .NET, Java	MS Windows	GPLv2	---	www.wizdl.codeplex.com
5	SOA Cleaner	Web services Java,.net	C#, .NET 2.0	MS Windows	Freeware	Xyrow	http://soa-cleaner-web-service-wcf-test-tool.soft112.com/
6	SOAPSonar Personal	Web services	.NET 2.0	MS Windows	Freeware	Crosscheck Network	http://SOAPSonar-personal-edition.software.informer.com/5.5/

Table 2 shows the version detail of selected tools such as release date, version used etc.

TABLE 2
VERSION'S DETAIL OF THE SELECTED TOOLS.

Sr. No.	Tool Name	1st Release Date	1st Version	Latest Release Date	Latest Version	Used Version
1	Apache Jmeter	03/9/2001	V2.1	05/10/2014	V2.11	V2.9
2	Soap ui pro	04/10/2007	V1.7	14/1/2014	V4.6.4	V4.6.0
3	Wft storm	15/08/2012	V1.1.0	27/02/2014	V3.1.0	V2.5
4	Wizdl	10/08/2008	V1.0	01/05/2013	V5	V1.1
5	SOA Cleaner	08/01/2006	V1.3.6.0	06/11/2011	V1.3.5.0	V1.3.0.0
6	SOAPSonar Personal	28/09/2005	V 1.00.1050	25/11/2014	V 7.0.2	V6.5.10

III. Comparative Study of the Selected Tools

This section represents the comparison of four open source web service testing tools and two freeware web service testing tools along with their observed results. The observed results will help the researcher to determine the efficiency of suitable test tool for their needs. Temperature conversion web service is used to compare the selected test tools.

A. System Requirements All the test cases were run on an Intel Core i5 2.30 GHz processor machine with 4GB RAM, Microsoft Windows 8 Professional, and 2mbps Internet connection. The comparison is made between six tools with the input of same web service i.e. the temperature conversion from Celsius to Fahrenheit. Testing of the tools requires configuration which in turn involve installation, test environment setup, collection of data, analytical survey and selection of parameter. The sample web service i.e. temperature conversion is tested on the respective configure tools.

B. Approach Followed The tests were performed at the same instance of time and at same network speed. Based upon input test cases can be categorized into types that is valid test cases and invalid test cases. The critical parameters (response time and bytes processed) were

evaluated to identify the performance of the testing tools. The observed results were analyzed to determine the efficiency of the tool. Table 3 shows the response time of testing tools for valid input (Celsius,"100"), Table 4 shows the response time of testing tools for invalid input (Celsius, "abc") .

TABLE 3
RESPONSE TIME OF TESTING TOOLS FOR VALID INPUT (Celsius,"100")

Sr. No.	Tool Name	Input in Celsius	Output in Fahrenheit	Response Time	Bytes/sec
1	Apache Jmeter	100	212	902.4	409
2	Soapui pro	100	212	604.71	407
3	Wcf Storm	100	212	1369	---
4	Wizdl	100	212	1000	---
5	SOA Cleaner	100	212	678.3	---
6	SOAPSonar Personal	100	212	391.68	374.5

All the testing tools were given the same input in the format of Celsius,"100" and the results are tabulated in Table 3. The testing tools provide converted temperature from Celsius to Fahrenheit. The main important observed factor was the time taken for response. It can be seen that SOAPSonar Personal took minimum time and in open source tools Soapui pro took minimum time. However all the tools have given same result values except Wcf Storm, Wizdl and SOA Cleaner do not display bytes processed.

TABLE 4
RESPONSE TIME OF TESTING TOOLS FOR INVALID INPUT (Celsius,"abc")

Sr. No.	Tool Name	Input in Celsius	Output in Fahrenheit	Response Time	Bytes/sec
1	Apache Jmeter	Abc	---	903.25	411
2	Soapui pro	Abc	---	605.51	409
3	Wcf Storm	Abc	---	1373	---
4	Wizdl	Abc	---	1000	---
5	SOA Cleaner	Abc	---	680.2	---
6	SOAPSonar Personal	Abc	---	394.5	380

All the testing tools were given the same input in the format of Celsius,"abc" and the results are tabulated in Table 4. Since the input given was invalid input, so no results was retrieved as clear from the blank spaces in the table but gives response message. Here also it can be seen that SOAPSonar Personal took the minimum time and in open source tools Soapui pro took minimum time.

IV. Results and Discussion

From the results it is evident that each tool had its own architecture and internal processes which form the basis of comparative study of tools in terms of response time and bytes processed by test. The response time observed for various tools is shown in Table 3 & 4. Observed results, shows that Wcf Storm takes maximum time to give response to web service then all other tools. Apache Jmeter and Wizdl give similar response time which is less than Wcf Storm. Soapui pro takes less response time than rest of the tools except SOAPSonar Personal. Hence from the values observed in all tables, it's clear that SOAPSonar Personal takes minimum response time for testing selected web service. The behavior shown by SOAPSonar Personal with respect to response time clearly showed that it is the fastest tool amongst all selected tools. In open source web service tools Soapui pro took minimum time as output. Also the results of test cases are summarized to calculate average response time of each tool for web service i.e. temperature conversion. This is represented in Table 5.

TABLE 5
AVERAGE RESPONSE TIME OF TESTING TOOLS

Web Service Name	**Average response time in ms**					
	Apache Jmeter	**Soapui pro**	**Wcf storm**	**Wizdl**	**SOA Cleaner**	**SOAPSonar Personal**
Temperature Conversion	**902.83**	**605.11**	**1371**	**1000**	**679.25**	**393.09**

It can be analyzed from the table that the average response time for SOAPSonar Personal is better than other tools which are used for observation. In open source tools Soapui pro outperform rest three open source tools. The observed data can also be represented in a graph which is shown in figure 1.

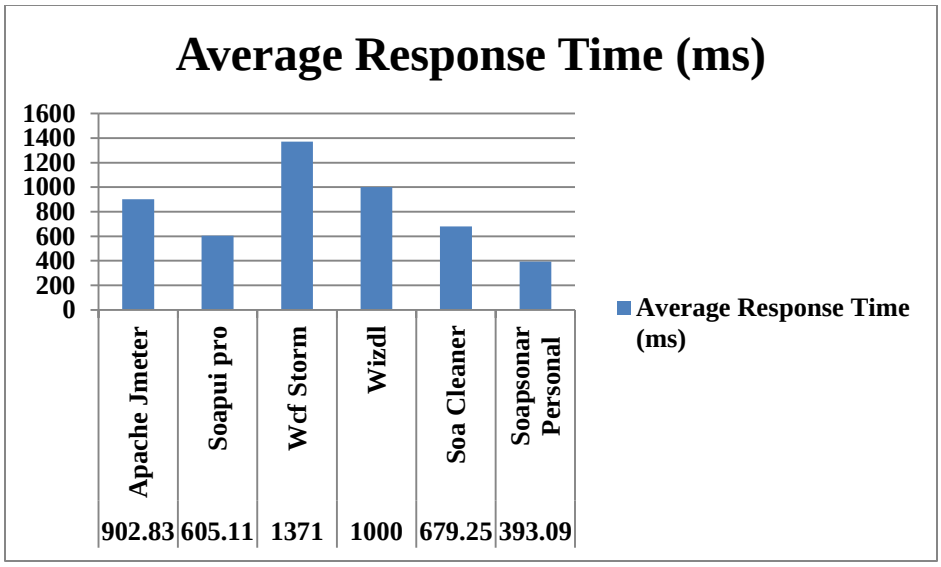

Figure 1: Average response time of testing tools

In the graph response tame is taken along y-axis and tools are taken along x-axis. It can be clearly observed from the graph that SOAPSonar Personal is fastest tool to test a web service in terms of average response time than rest of the tools considered for analysis. In open source tools Soapui pro outperform rest three open source tools.

Second parameter considered for the comparing tools is number of bytes used to process the test. From table 3 & 4 it is observed that only Apache Jmeter, Soapui pro and SOAPSonar Personal display number of bytes processed. Also the results of test cases are summarized to calculate average response time of each tool for web service i.e. temperature conversion. This is represented in Table 6.

TABLE 6
AVERAGE BYTES/SEC OF TESTING TOOLS

Web Service Name	Avg. Bytes/sec		
	Apache Jmeter	Soapui pro	Soapsonar Personal
Temperature Conversion	410	408	377.25

Apache Jmeter takes more number of byes to process the test than Soapui pro and SOAPSonar Presonal. This means that Apache Jmeter checks more options or attributes during request and response. Thus Apache Jmeter outperforms the other two tools. It can also be observed directly from the graph shown in figure 2.

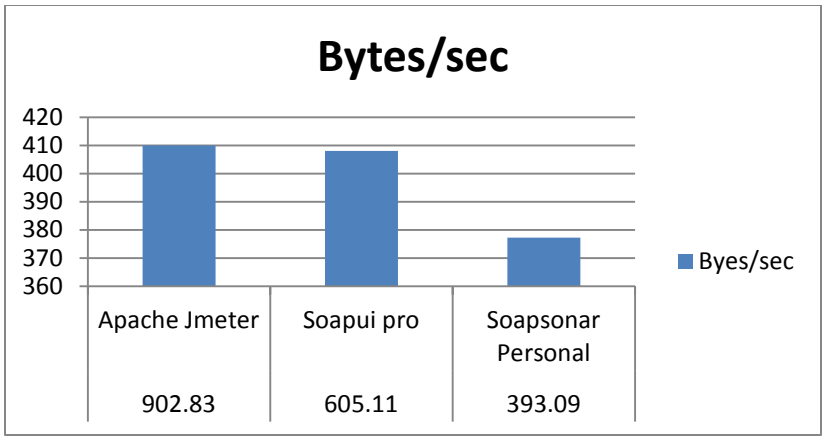

Figure 2: Bytes/sec for processing test

Here bytes/sec is taken along y-axis. It and tools are taken along x-axis. It can be clearly observed from the graph that Apache Jmeter processes more bytes than rest of the tools considered for the analysis. Hence Apache Jmeter is better in term of bytes processed than rest of the two tools.

V. Conclusion

Testing a web service is challenging activity that involves many characteristics such as response time, throughput and calculation of bytes processed etc. The experimental approach used in this paper is based on real service implementation to retrieve and store data. The parameter results of different web service testing tools such as Soapui Pro, Wcf Storm, Apache Jmeter, Wizdl, SOA Cleaner and SOAPSonar Personal have been analyzed. The same web service i.e. temperature conversion has been tested for performance with these testing tools and results has been compared. The analysis helps in the selection of the best tool. This research work can be extended to more tools, more web services and different parameters to provide more empirically realistic results.

References:

[1] Ms. Shikha Maheshwari, "A Comparative Analysis of Different types of Models in Software Development Life Cycle", International Journal of Advanced Research in Computer Science and Software Engineering Volume 2, Issue 5, May 2012.

[2] Rajendra Bathla and Shallu Bathla, " Innovative approaches of automated tools in software testing and current technology as compared to manual testing", Global Journal of Enterprise of Information System, Vol. 1, Issue 1 Jan-Jun 2009.

[3] Mohd. Ehmer Khan, "Different Forms of Software Testing Techniques for Finding Errors",IJCSI International Journal of Computer Science Issues, Vol. 7, Issue 3, No 1, May 2010.

[4] Jovanovich and Irena, "Software Testing Methods and Techniques", May 26, 2008.

[5] A. Askaruinisa and A.M. Abirami, "Test Case Reduction Technique for Semantic Based Web Services", (IJCSE) International Journal on Computer Science and Engineering, 2010.

[6] S. Hussain, Z. Wang, I. Kalil Toure and A. Diop, "Web Service Testing Tools: A Comparative Study", (IJCSI) International Journal of Computer Science, Jan. 2013.

[7] Dr.K.V.K.K.K Prasad, "Software testing tools", Dreamtech, 2006.

[8] M.G. Limaye,"Software testng Principles, Techniques and Tools", Tata MC GrawHill, 2009.

[9] http://jmeter.apache.org/, "Apache JMeter", Retrieved on September 2014 at 6 pm.

[10] http://www.soapui.org/,"Soapui–The Home of Functional Testing", Retrieved on October 2014 at 4 pm.

[11] http://storm.codeplex.com/, "Storm", Retrieved on October 2014 at 10 am.

[12] http://wizdl.codeplex.com/, "Wizdl", Retrieved on September 2014 at 11 am.

[13] http://soa-cleaner-web-service-wcf-test-tool.soft112.com/, "SOA Cleaner", Retrieved on December 2014 at 1 pm.

[14] http://soapsonarpersonal-edition.software.informer.com/,"SOAPSonarPersonal", Retrieved on December 2014 at 11 am.

