



May 14, 2008

Whitepaper

Evaluation of Load/Stress tools for Web Applications testing

CONTACT INFORMATION:

phone: +1.301.527.1629

fax: +1.301.527.1690

email: whitepaper@hsc.com

web: www.hsc.com



PROPRIETARY NOTICE

All rights reserved. This publication and its contents are proprietary to Hughes Systique Corporation. No part of this publication may be reproduced in any form or by any means without the written permission of Hughes Systique Corporation, 15245 Shady Grove Road, Suite 330, Rockville, MD 20850.

Copyright © 2008 Hughes Systique Coporation

TABLE OF CONTENTS

<u>SECTION</u>	<u>PAGE</u>
1.0 PREFACE	4
2.0 EVALUATION CONSIDERATIONS – KEY PARAMETERS USED FOR THE EVALUATION OF LOAD/STRESS TESTING TOOLS	5
3.0 TOOLS WORTH LOOKING AT – OPEN SOURCE TOOLS	7
3.1 TOOL NAME: JMETER.....	7
3.2 TOOL NAME: TESTMAKER.....	8
3.3 TOOL NAME: GRINDER.....	9
3.4 TOOL NAME: OPENSTA.....	10
4.0 EVALUATION OF IDENTIFIED TOOLS W.R.T. KEY PARAMETERS VS. COMMERCIAL TOOL (LOADRUNNER).....	11
5.0 CONCLUSIONS.....	12
6.0 ABOUT THE AUTHOR.....	13

1.0 PREFACE

With the continued tightening of expenses, IT organizations are forced to do more with less. The emergence of open source software has given IT organizations the opportunity to reduce their development cost without sacrificing quality. Software applications such as Apache web server and Open Office productivity package are examples of open source products used by businesses to reduce the total cost of ownership (TCO) within their organizations.

The high cost of performance test automation software is requiring QA professionals to begin to evaluate open source alternatives. The goal of this paper is to explain how to select open source tools, implement a sound performance test strategy, and gain the same benefits as the commercial performance test tools at a fraction of the commercial tools cost.

Although there are several open source performance test tools available, the details in following document is as a result of Evaluation Study. Document explains the most popular open source performance test tools OpenSTA, TestMaker, JMeter and Grinder. Each of these tools provides the functionality necessary to complete a performance, load or stress test.

2.0 EVALUATION CONSIDERATIONS – KEY PARAMETERS USED FOR THE EVALUATION OF LOAD/STRESS TESTING TOOLS

Key Parameter	Description
Scripting language	The medium used to represent the captured protocol data and manipulate the data for play-back.
Playback functions & Correlation	Playback function means replaying of the script and script debugging facilities. Correlation is the task of substituting values in dynamic data to enable successful playback.
Scripting Interface	The interfaces supplied by the tool application for the purpose of script editing.
Cookie Management	Detection, recording and playback of HTTP cookies. Tools need additional code to manage JavaScript generated cookies.
Parameterization	Automatically changing dynamic data values for more accurate emulation of real users. Often essential for session management and used in data driven testing.
Protocols support	The communication protocols that can be captured, manipulated and replayed by the tool.
Extensibility	The ability to increase the functionality of the tool.
Resource Monitoring	Resource usage information is captured during execution. It can be shown during execution and used to build performance reports.
Distributed tests and load generation	The ability to distribute the generation of load across multiple load-injector machines. To generate large loads additional load generator machines are required. These machines must be centrally controlled.
Cross-browser Support	Cross-browser refers to the ability for a website, web application, HTML construct or client-side script to support all web browsers.
Caching	The ability to emulate the caching of pages as performed by a web browser.
Reporting and analysis	The facilities to examine and investigate the results of a test including timers, monitored resources and display of results in graphical format.
Scalability on Virtual users	The ability of the tool to generate numbers of virtual users and the corresponding resource usage. Actual resource use depends on the number, size and complexity of the scripts.
Initial Costs and Costs of Virtual users	Purchase costs of software and licenses, excluding upgrades or support. Most commercial tools charge on the basis of the number of virtual user available. Most commercial tools charge on the basis of the number of virtual user available. Extra hardware is an additional cost.

Support and consultancy	The support services available for the tool and some costs.
Training	The training services available for the tool. This impact the learning curve of team involved.
System requirements	The operating systems required to host the tool. (Not the OS of the system under test).
Access to source code	Availability of the source code of the tool itself.

3.0 TOOLS WORTH LOOKING AT – OPEN SOURCE TOOLS

3.1 Tool Name: JMeter

Organization:

APACHE SOFTWARE FOUNDATION
<http://jakarta.apache.org/jmeter/>

Requirement:

Java 1.4 or higher virtual machine on Windows and Unix (Solaris, Linux etc)

Description:

Java desktop application from the Apache Software Foundation designed to load test functional behavior and measure performance. Originally designed for testing Web Applications but has since expanded to other test functions; may be used to test performance both on static and dynamic resources (files, Servlets, P-erl scripts, Java Objects, Data Bases and Queries, FTP Servers and more).

Scripting Language:

It has support for Java requests and BeanShell scripts.

Feature pro/cons:

Strengths	Weakness
1. Freeware 2. The distributed testing 3. Various target systems 4. Extensibility: Pluggable samplers allow unlimited testing capabilities	1. Getting "correct" response times from JMeter forces you to understand how to make a realistic client/server setup. If you're not careful you might end up measuring some incidental bottleneck, like your own workstation's capacity. 2. Its always advisable not to run JMeter on the same machine running the application to be tested. JMeter may use extensive resources that might affect the other application's performance. 3. Chart representation looks bit confusing 4. Terminology not very clear 5. Remote machines must be declared in a property file before starting application

3.2 Tool Name: TestMaker

Organization:

PUSHTOTEST

<http://www.pushtotest.com/>

Requirement:

Java 1.4 or higher virtual machine on Windows, Linux, Solaris, and Macintosh.

Description:

TestMaker from PushToTest.com delivers a rich environment for building and running intelligent test agents that test Web-enabled applications for scalability, functionality, and performance. It comes with a friendly graphical user environment, an object-oriented scripting language (Jython – simplified version of Python) to build intelligent test agents, an extensible library of protocol handlers (HTTP, HTTPS, SOAP, XML-RPC, SMTP, POP3, IMAP), a new agent wizard featuring an Agent Recorder to write scripts for you, a library of fully-functional sample test agents, and shell scripts to run test agents from the command line and from unit test utilities.

Scripting Language:

It uses Jython – an open source object-oriented scripting language with a friendly graphical user environment

Feature pro/cons:

Strengths	Weakness
<ol style="list-style-type: none"> 1. The possibility to build any kind of test agent. 2. The power of Java programming with some Python simplifications. 3. Multi-environment 4. Open source 	<ol style="list-style-type: none"> 1. To build a test plan it is necessary to get familiar with the Jython scripting language, Java language and to write it from scratch. 2. The monitoring tools are very basic, since it is limited to the response analysis. 3. The complexity to quickly build some analysis results. 4. For distributed testing, one has to pay

3.3 Tool Name: GRINDER

Organization:

A DEVELOPER GROUP
<http://grinder.sourceforge.net/>

Requirement:

Java 1.4 or higher virtual machine on any Operating System

Description:

Grinder is a GUI desktop application designed to load test functional behavior and measure performance. Grinder uses Jython for defining test. The Grinder is a Java load-testing framework making it easy to orchestrate the activities of a test script in many processes across many machines, using a graphical console application. Three types of processes make up The Grinder:

- Agent processes: A single agent process runs on each test-client machine and is responsible for managing the worker processes on that machine.
- Worker processes: Created by The Grinder agent processes, they are responsible for performing the tests.
- The console: Coordinates the other processes and collates statistics.

Scripting Language:

Latest version, Grinder 3, uses the powerful scripting language Jython, and allows any Java code to be tested without the need to write a plug-in.

Feature pro/cons:

Strengths	Weakness
<ol style="list-style-type: none"> 1. Grinder is open source software. 2. Grinder tests are written in Jython(a Java Python implementation) which is very powerful. 3. Grinder can be setup very quickly and is easy to use. 4. Grinder have a Java swing user friendly interface 	<ol style="list-style-type: none"> 1. There is no support available for Grinder. 2. There is no GUI based interface for defining the scripts 3. The reports created by grinder do not contain graphs, chart. 4. Monitoring system resource usage on the target system. 5. Making each simulated session look like it's coming from a different IP address (important if you have a load balancer). 6. Identifying itself as a real browser when contacting the web server, and being able to simulate more than one browser.

3.4 Tool Name: OpenSTA

Organization:

OPEN SYSTEM TESTING ARCHITECTURE

<http://OpenSTA.org/>

<http://portal.opensta.org/>

Requirement:

Java 1.3 or higher virtual machine on Windows NT4 only

Description:

A distributed software testing architecture based on CORBA. Using OpenSTA (Open System Testing Architecture) a user can generate realistic heavy loads simulating the activity of hundreds to thousands of virtual users.

OpenSTA graphs both virtual user response times and resource utilization information from all Web Servers, Application Servers, Database Servers and Operating Platforms under test, so that precise performance measurements can be gathered during load tests and analysis on these measurements can be performed.

Scripting Language:

Script Control Language (SCL), uses a proprietary, "BASIC" like language that has special automation scripting facilities.

Feature pro/cons:

Strengths	Weakness
1. A user-friendly graphical interface. 2. The script captures all information from the browser. 3. The monitoring functionality 4. You can make very complete test with the script language	1. Only designed for Windows 2. Only for HTTP/HTTPS

4.0 EVALUATION OF IDENTIFIED TOOLS W.R.T. KEY PARAMETERS VS. COMMERCIAL TOOL (LOADRUNNER)

Different open source tools were considered for evaluation and based on Key parameters criteria, many tools were omitted from the list. Only most suitable tools were considered for the detailed study.

Different sections of sheet include evaluator's description for the ratings for the tools. Report of the Evaluation Study is attached here for reference.



Evaluation Study
Report

Following table shows the comparison with rating as per above evaluation sheet. Weighted scores are shown below based on rankings is done on a scale of 1-5 (5 being Best)

Table: Weighted scores based on the Evaluation Study Report

Open Source tools vs. Key Parameters(Weight)	JMeter	TestMaker	Grinder	OpenSTA
Scripting language (5)	20	25	25	15
Playback functions & Correlation (4)	16	20	16	8
Scripting Interface (4)	16	20	16	12
Cookie Management (4)	16	16	16	12
Parameterization (4)	16	16	16	12
Protocols support (3)	12	12	12	9
Extensibility (5)	25	25	25	20
Resource Monitoring (3)	9	12	6	6
Distributed tests/Distributed load generation (3)	12	9	12	9
Browser Support (3)	12	12	9	9
Caching (3)	12	12	9	6
Reporting and analysis (4)	16	16	8	8
Scalability on Virtual users (4)	16	16	16	12
Initial Costs and Costs of Virtual users (3)	15	12	15	15
Support and consultancy (5)	20	20	20	20
Training (3)	12	15	9	12
System requirements (1)	5	5	5	3
Access to source code (2)	10	10	10	10
Total Score	260	273	245	198

5.0 CONCLUSIONS

Jmeter offers some good features and can perform tests on several systems, but it is a little more difficult to use than the other software. It is possible to realize a wide variety of load tests, even to distribute the scenario. However, the application is weakened by the management of Java objects and memory concerns which affect global performance. The major advantage is the possibility to add new features very simply in the application; it is also a good example of modularity and customization.

TestMaker is a powerful tool since the scenario is created with scripts in Jython which allow to use all the Java features. This framework permits to build a customized scenario and very precise test plan, with specific classes. However, its average on the monitoring part and distributed testing is not possible. But it seems interesting to have a look at the classes structure and hierarchy.

Grinder is similar to TestMaker, except that it allows distributed tests, so it can simulate heavy loads.

OpenSTA provides some good performances for HTTP testing, with simple charts and distributed tests. It is a good application for simple and reliable HTTP tests.

To summarize the evaluation study, JMeter and Grinder are of Programmer's tool category and scores more over OpenSTA. TestMaker is an equivalently good tool for Tester's use.

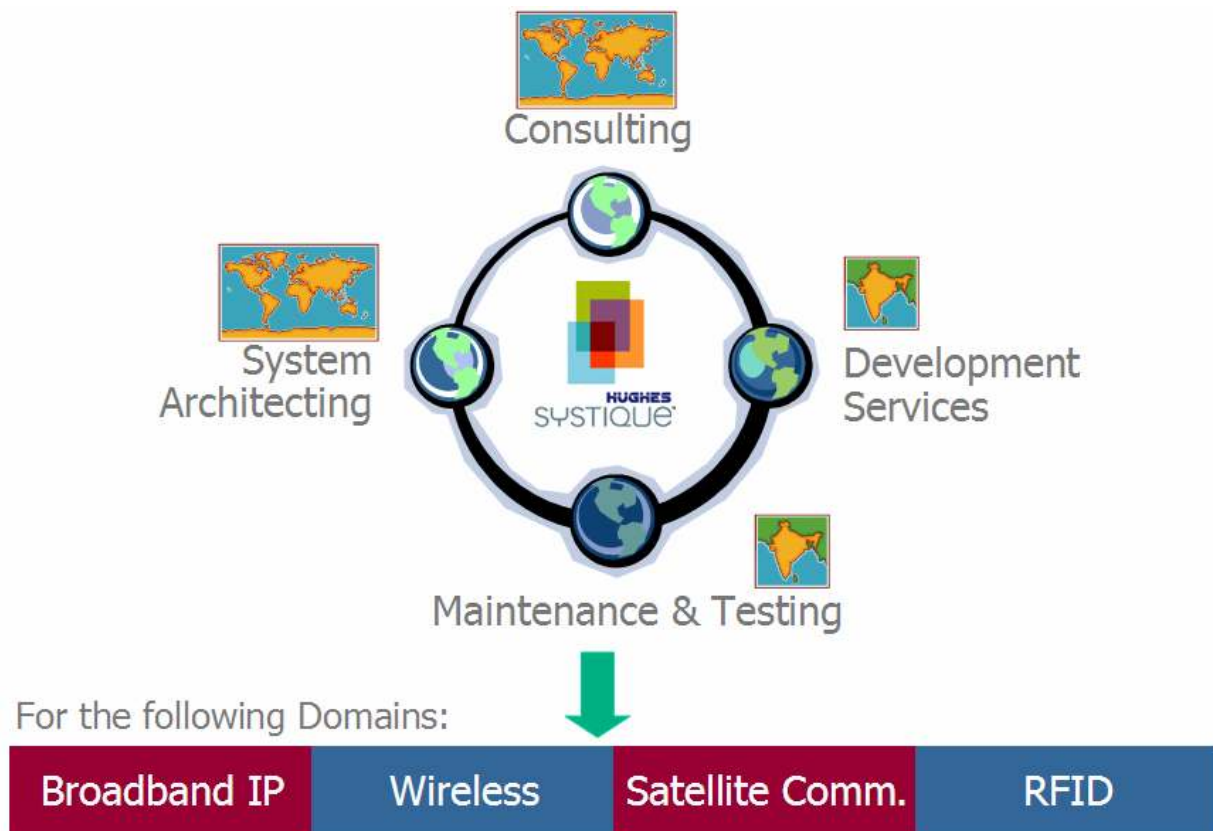
6.0 ABOUT THE AUTHOR

Praveen Arora is part of HSC team, serving for HNS client and managing Testing Projects. He is having more than 6years of experience in Software Testing Projects on Telecom domain. His educational background is MCA, MBA and holds multiple certifications. Prior to joining HSC, he was associated with Ness Technologies, Bangalore (NETCOOL and NMS based projects) and Quark Media House, Mohali (Media Portal projects). His key strengths include Testing, Testing Tools, Project Management and Quality Processes.

APPENDIX A ABOUT HUGHES SYSTIQUE CORPORATION

HUGHES Systique Corporation, part of the HUGHES group of companies, is a leading communications Consulting and Software company. We provide Consulting, Systems Architecture, and Software Engineering services to complement our client's in-house capabilities. Our "Best Shore" model coupled with an experienced management and technical team team is capable of delivering a total solution to our clients, from development to deployment of complex systems, thus reducing time, risk and cost

HSC Solution Space:



CONTACT INFORMATION:

phone: +1.301.527.1629

fax: +1.301.527.1690

email: whitepaper@hsc.com

web: www.hsc.com

HSC Expertise Areas in Brief:



CONTACT INFORMATION:

phone: +1.301.527.1629

fax: +1.301.527.1690

email: whitepaper@hsc.com

web: www.hsc.com