



A Comparative Analysis of Load Testing Tools Using Optimal Response Rate

Pooja Ahlawat

DCSA, Kurukshetra University
India.

Sanjay Tyagi

DCSA, Kurukshetra University
India.

Abstract-Web applications are widely known as the building blocks of typical service oriented applications. The performance of such an application system is mainly dependent upon the components of web applications. These web applications should be load tested to ensure that they can function correctly under high load. In recent years, various tools and techniques are developed for measurement and evaluation of quality of service of web applications. These tools facilitate in quality of service analysis and are helpful in evaluation of service performance in real-time network. In this paper, three popular load testing tools have been described 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.

Keywords: Load testing, testing tools, test script, Open-source Software, web applications.

I. INTRODUCTION

The goal of a load testing is to uncover functional and performance problems of a system under load. Performance problems refer to the situations where a system suffers from unexpectedly high response time or low throughput [5]. Many applications support concurrent access by hundreds or thousands of users. To assure the quality of these applications, load testing is a required procedure in addition to conventional functional testing procedures such as black box testing, unit testing, integration testing etc. Load testing refers to the practice of accessing the system's behaviour under load [17]. Load refers to the rate of the incoming requests to the system. A load test can last for several hours or even a few days. Load testing requires one or more load generators which mimic clients sending thousands or millions of concurrent requests to the application under test. During the procedure of a load test, the web application is monitored and the performance data results along with execution logs are stored. Performance data is recorded in respect to the resource usage information such as CPU consumption, memory usage, disk I/O and network traffic. In execution logs, the run time behaviour of the web application under test is recorded [16]. Objective of performance testing is to verify system's performances like response time, service availability, capacity. It is executed by simulating hundreds or more concurrent users access over a defined time interval. Information about accesses are recorded and then analyzed to estimate the load levels exhausting system resources. For web applications, system performances is a critical issue because web users don't like to wait too long for a response to their requests and they expect that services are always available. Load testing of Web applications should be considered as an everlasting activity to be carried out by analyzing data from access logs, in order to tune the system adequately. Load testing requires that system performance is evaluated with a predefined load level. It aims to measure the time needed to perform several tasks and functions under predefined conditions. The predefined conditions include the minimum configuration and the maximum activity levels of the running application. Also in this case, lots of simultaneous user accesses are simulated. Information is recorded and if the tasks are not complete then failure reports will be generated. Failures found by load testing are mainly due to faults in the running environment [3]. Load testing lets you measure your application's quality of service performance based on actual customer behaviour. When customers access your site, a script recorder uses their requests to create interaction scripts. A load generator then replays the scripts which are modified by test parameters against the Web site. Parameters [5] that affect Load Testing are Response Time, Think Time, Work Load and Throughput. Response time refers to the time in which the system responds for a particular transaction request. Response time is often defined as the interval from when a user initiates a request to the instant at which the first part of the response is received to user by the application. Think time refers to the time taken for selecting a new transaction after the response for the previous transaction has been received. Work Load depends on Transactions and Number of users. Throughput refers to the amount of work that can be performed by a system or component in a given period of time. Throughput depends on Transactions/sec, Web Interactions/sec, Pages/sec, Byte/sec.

II. Related work

Nianmin Yao, Feng Gao, Shaobin Cai & Wenbin Yao [11] proposed a new method to test the performance of network storage, which easily can generate requests that exceed the maximum loading of storage devices. Pu yunming and Xu mingna [13] analyzed the performance testing criteria, including response time, concurrency users, throughout and performance counter. With the LoadRunner and TestDirector testing tools, a load testing scheme based on an online

examination system was designed. Zhang Hui-li, Zhang Shu, Li Xiao-jie, Zhang Pei, Zhengzhou and Liu Shao-bo [9] implemented load testing for the e-commerce applications system by means of LoadRunner testing tool, analysed results, and improved the performance of the system based on the test results. Osama Hamed and Nedal Kafri [10] proposed a performance testing approach for web based applications early in the development process. They compared performance in terms of response time for two web application architectures. The performance goals were based on comparing the response time and memory utilization for the implemented applications. Sneha Khorja and Pragati Upadhyay [12] analyzed the performance of some specific testing tools and compared them to determine their usability and effectiveness in software testing. The paper described two testing tools: WAPT and RANOREX, which were implemented on a web application. The performance of these testing tools were evaluated and compared. Shariq Hussain, Zhaoshun Wang, Ibrahima Kalil Toure and Abdoulaye Diop [1] presented a comparative study of open source web service testing tools with technical overview and features. Comparison was made on several quality factors including response time, throughput, and usability. Tools were evaluated by collecting the sample web services and collecting the test results.

III. Proposed work

In this work, three testing tools WAPT, LOADUI and LOADSTER are taken, which are implemented on a web application. WAPT is a load, stress and performance testing tools, used for website and web applications. LOADUI is another open source load testing tool, which is used for measuring the performance of the web applications. Functions in LOADUI can be added at run time using plug-ins. LOADSTER is a desktop based advanced HTTP load testing tool. The web browser can be used to record the scripts which are easy to use and record. The performances of these testing tools have been evaluated & compared and results have been presented and discussed.

A. WAPT

WAPT is an easy to use and cost effective yet powerful testing tool for web environment. With WAPT, we can test and analyze the performance characteristics and bottlenecks of web site under various load conditions. WAPT has a number of features to test sites with dynamic content and secure HTTPs pages. Various key Features of WAPT are accurate real user simulation, dynamic test data generation, recording and playback of HTTPS/SSL Pages, flexible test volume & load definition, clear reports & graphs and full log of virtual user's activities. We take a shopping site for testing and analyze its metrics.

Response time, sec (with page elements)							
Name	Time	0:00:00-0:00:30	0:00:30-0:01:00	0:01:00-0:01:30	0:01:30-0:02:00	0:02:00-0:02:30	0:02:30-0:03:00
Profile1.page_1: http://www.flipkart.com		2	1	4	4	7	7
	Min	1,38 (2,5)	1,41 (2,56)	1,33 (2,45)	1,34 (2,53)	1,31 (2,44)	1,38 (2,48)
	Avg	1,38 (2,51)	1,41 (2,56)	1,5 (2,64)	1,41 (2,59)	1,38 (2,58)	1,4 (2,59)
	Avg90	1,39 (2,52)	1,41 (2,56)	1,61 (2,72)	1,5 (2,66)	1,46 (2,85)	1,44 (2,74)
	Max	1,39 (2,52)	1,41 (2,56)	1,61 (2,72)	1,5 (2,66)	1,5 (3,05)	1,47 (2,81)
Profile1.page_2: http://www.flipkart.com_utm.gif		1	2	3	5	5	6
	Min	0,28 (0,28)	0,25 (0,25)	0,23 (0,23)	0,25 (0,25)	0,25 (0,25)	0,25 (0,25)
	Avg	0,28 (0,28)	0,25 (0,25)	0,25 (0,25)	0,33 (0,33)	0,27 (0,27)	0,29 (0,29)
	Avg90	0,28 (0,28)	0,25 (0,25)	0,27 (0,27)	0,45 (0,45)	0,28 (0,28)	0,34 (0,34)
	Max	0,28 (0,28)	0,25 (0,25)	0,27 (0,27)	0,45 (0,45)	0,28 (0,28)	0,36 (0,36)
Profile1.page_3: http://www.flipkart.com/index.php		1	2	3	4	5	7
	Min	0,75 (8,5)	0,78 (9)	0,75 (8,31)	0,78 (8,81)	0,73 (8,05)	0,75 (8,38)
	Avg	0,70 (8,5)	0,25 (9,09)	0,32 (8,38)	0,77 (9,38)	0,28 (8,44)	0,77 (8,63)
	Avg90	0,75 (8,5)	0,88 (10,4)	0,83 (8,48)	0,92 (9,77)	0,77 (8,65)	0,81 (8,99)
	Max	0,75 (8,5)	0,88 (10,4)	0,83 (8,48)	0,92 (9,77)	0,77 (8,65)	0,83 (9,23)

Fig 1: Response Time using WAPT tool

B. LOADUI

LOADUI is an open source load testing solution that is free and cross-platform. With a special feature of visual drag and drop interface, it allows us to create, configure and redistribute load tests interactively and in real-time. In a single test environment, LOADUI provides complete test coverage and supports all the standard protocols and technologies.

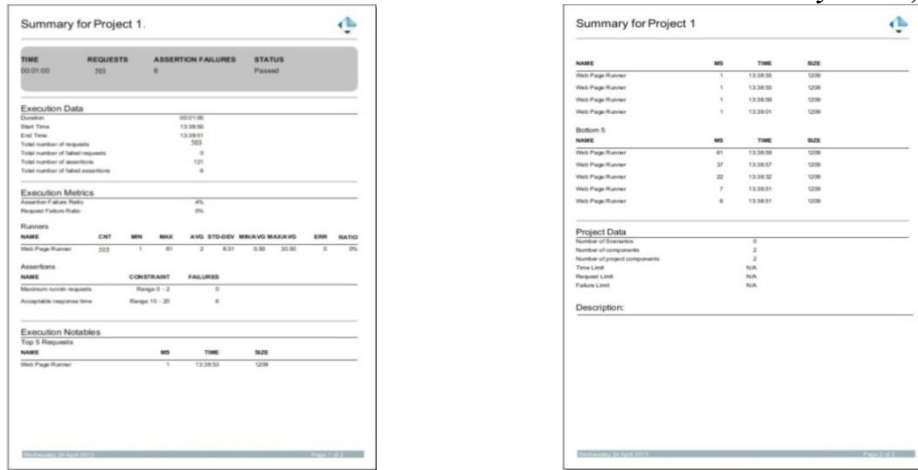


Fig 2: Summary Report of Load Testing using LOADUI tool



Fig 3: Graphs of throughput and response time

C. LOADSTER

LOADSTER is a desktop-based load and stress testing tool. Load testing with LOADSTER is a great idea when a dynamic web application is build. LOADSTER has easy to use script recorder and realistic test scenarios that simulate multiple users' behaviour at once. When the test is finished, LOADSTER generates simple HTML test reports with graphs and minimal effort.

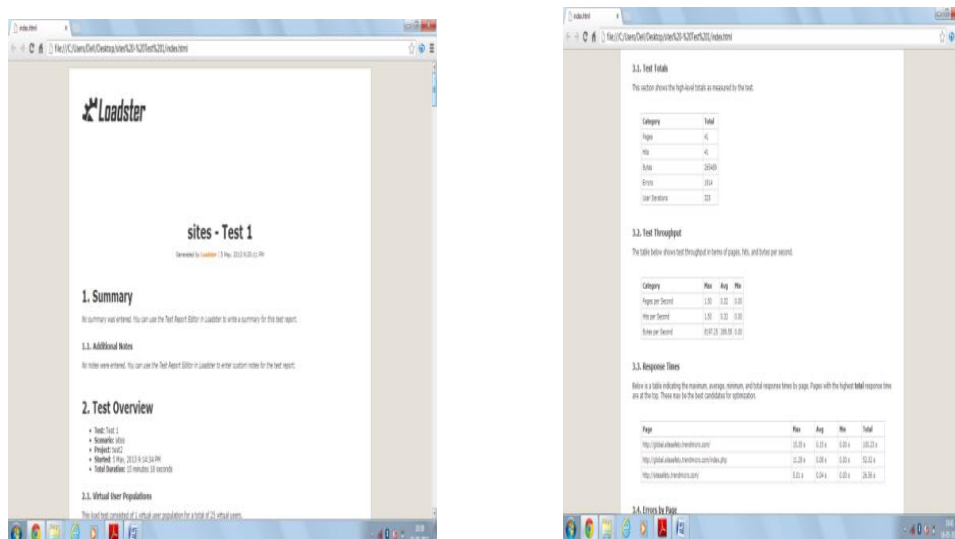


Fig 4: HTML Test Report

Comparison of Average Response Time
Table 1: Comparison of Average Response Time

Web Page under Test	WAPT	LOADUI	LOADSTER
1	0.70	0.61	0.15
2	0.25	0.37	0.08
3	0.32	0.22	0.04
4	0.71	0.70	0.65
5	0.28	0.60	0.38

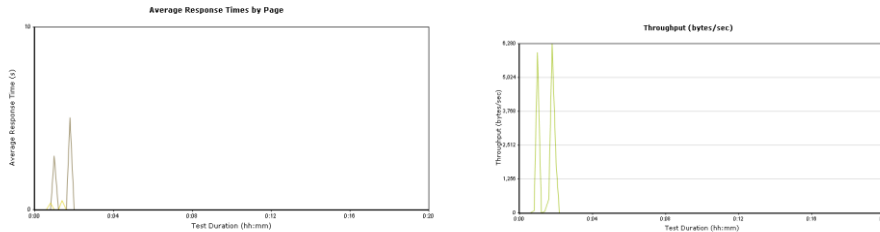


Fig 5: Graphs of average response time and throughput

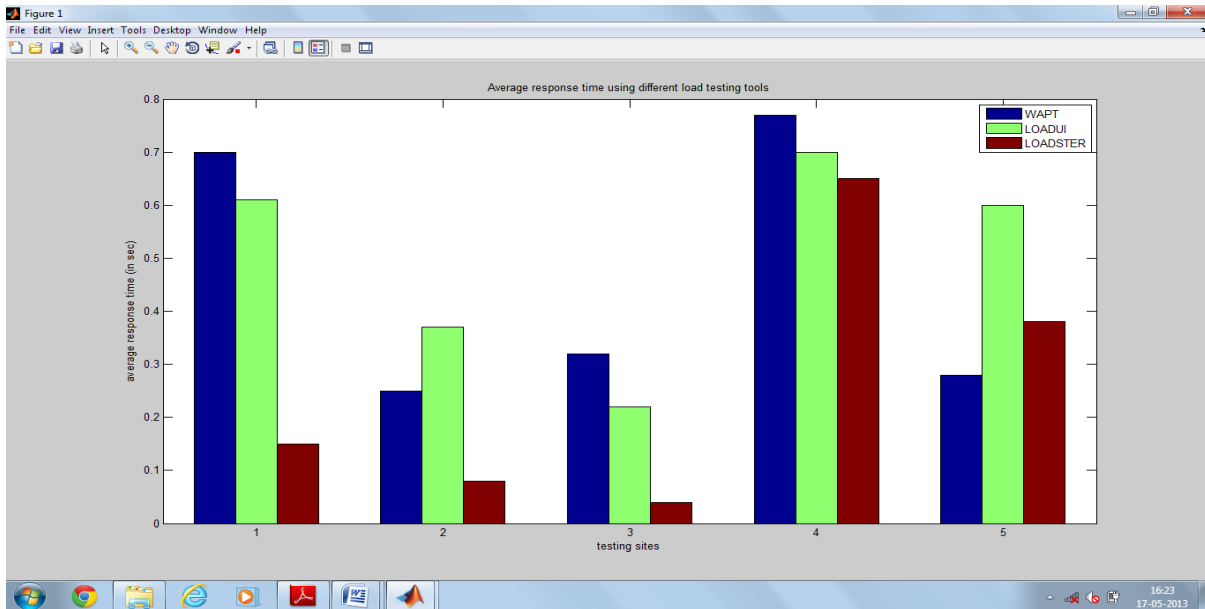


Fig 6: Graph of Comparison

IV. Observation

We implemented a metric to determine the relationship between average response time and optimal response rate with different concurrent user's transaction. Here, three different testing scenarios according to different average response time are created. They are based on the same test scripts with different testing tools. We identified how long users would wait for pages to respond and reported the following ratings:

- Optimal response time = below .3 sec
- High response time = .3 - .5 sec
- Low response time = above .5 sec

Table II: Comparison of Different Status

Web Page under Test	WAPT		LOADUI		LOADSTER	
	Response Time	Status	Response Time	Status	Response Time	Status
1	0.70	Low	0.61	low	0.15	Optimal
2	0.25	Optimal	0.37	high	0.08	Optimal

3	0.32	High	0.22	optimal	0.04	Optimal
4	0.71	Low	0.70	low	0.65	Low
5	0.28	Optimal	0.60	low	0.38	High

Optimal response rate = number of optimal response request / total number of request

Table III: Optimal Response rates

	WAPT	LOADUI	LOADSTER
Optimal response rate	40%	20%	60%

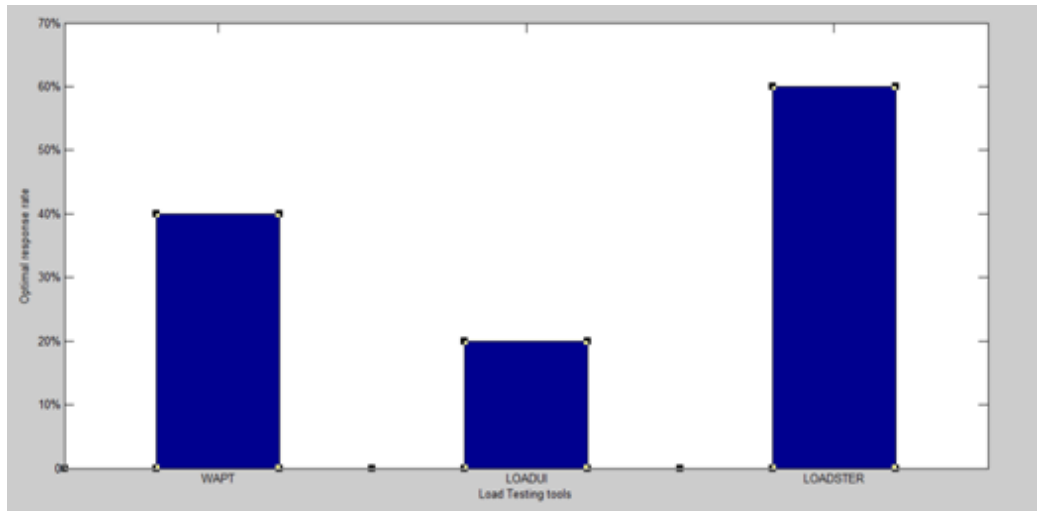


Fig 7: Graph of Optimal Response rate of Different Tools

High response rate = number of high response requests / total number of requests

Table IV: High Response rate

	WAPT	LOADUI	LOADSTER
High response rate	20%	20%	20%

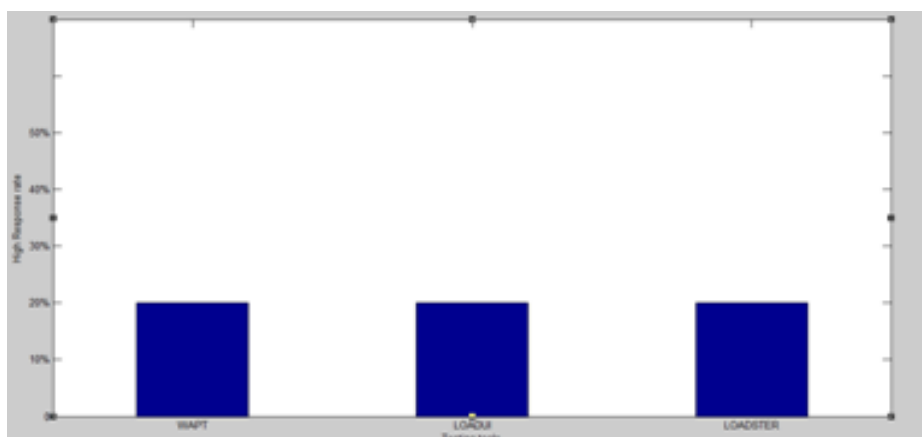


Fig 8: Graph of High Response rate of Different Tools

Low response rate = number of low response requests / total number of requests

Table V: Low Response Rate

	WAPT	LOADUI	LOADSTER
Low response rate	40%	60%	20%

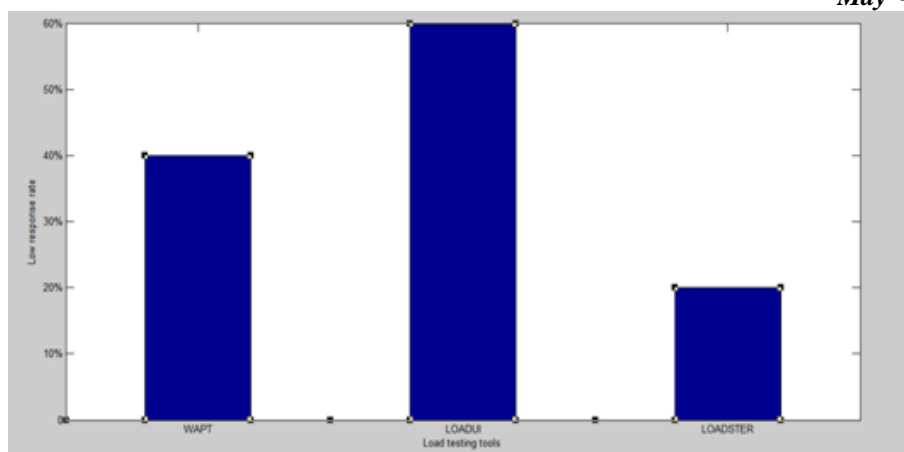


Fig 9: Graph of Low Response Rate of Different Tools

V. Conclusion and future work

Load testing is an important integral part of the performance testing. Load testing measures the application's ability to sustain a number of simultaneous users and/or transactions with adequate response times. In the paper, we presented an overview of three open source web service testing tools and a technical overview of each tool. Tools were evaluated and then test results were collected. Comparison was made on average response time and a new proposed metric optimal response rate, which may help researchers in selection of suitable tool. After analyzing the performance of these load testing tools, we conclude that LOADSTER is the best tool for web applications because of its larger percentage of optimal response rate. This research work can be further enhanced by including more testing tools for comparison so that we could find more suitable load testing tools for testing the application software. Further, some different metrics can be used for performance evaluation so that comparison could be more realistic and reliable.

References

- [1] Abdoulaye Diop, Ibrahima Kalil Toure, Shariq Hussain and Zhaoshun Wang, "Web Service Testing Tools: A Comparative Study", IJCSI International Journal of Computer Science Issues, Vol. 10, 2013.
- [2] Ahmed E. Hassan, Gilbert Hamann, Parminder Flora and Zhen Ming Jiang, "Automatic Identification of Load Testing Problems", IEEE, 2008.
- [3] Anna Rita Fasolino and Giuseppe A. Di Lucca, "Testing Web-based applications: The state of the art and future trends", Science Direct Information and Software Technology 48, 2006.
- [4] Arora A. and Sinha M., "Web Application Testing: A Review on Techniques, Tools and State of Art", International Journal of Scientific & Engineering Research, Volume 3, Issue 2, 2012.
- [5] Daniel A. Menasce, "Load Testing of Websites", <http://computer.org/internet>, IEEE Internet Computing, PP.70-74, 2002.
- [6] Guangzhu Jiang and Shujuan Jiang, "A Quick Testing Model of Web Performance Based on Testing Flow and its Application", Sixth Web Information Systems and Applications Conference (WISA 2009), 2009.
- [7] Hailong Sun, Minzhi Yan, Xu Wang and Xudong Liu, "WS-TaaS: A Testing as a Service Platform for Web", IEEE 18th International Conference on Parallel and Distributed Systems, 2012.
- [8] <http://www.softwaretestinghelp.com/performance-testing-tools-load-testing-tools/>
- [9] Li Xiao-jie, Zhang Hui-li and Zhang Shu, "Research of Load Testing and Result Application Based on LoadRunner", National Conference on Information Technology and Computer Science, 2012.
- [10] Nedat Kafri and Osama Hamed, "Performance Testing for Web Based Application Architectures", IEEE, 2009.
- [11] Nianmin Yao, Feng Gao, Shaobin Cai and Wenbin Yao, "A New Method of Performance Test for Network Storage", DOI 10.1109/IMSCCS, 2007.
- [12] Pragati Upadhyay, Sneha Khorja, "Performance Evaluation and Comparison of Testing Tools", VSRD International Journal of Computer Science & Information Technology, Vol. 2, 2012.
- [13] Pu yunming, Xu mingna, "Load Testing for Web Applications", The 1st International Conference on Information Science and Engineering IEEE, 2009.
- [14] Qinglin Wu, "Performance Testing and Optimization of J2EE-based Web Applications", Second International Workshop on Education Technology and Computer Science, IEEE, 681-683, 2010.
- [15] Steven Haines Senior, "Performance Testing Methodology", IEEE International Conference on Information management, 2005.
- [16] X. Bai, W. Dong, Y. Chen and W. T. Tsai, "WSDL-Based Automatic Test Case Generation for Web Services Testing", Proc. of IEEE International Workshop on Service-Oriented System Engineering, 2005.
- [17] Zhen Ming Jiang, Ahmed E. Hassan, Gilbert Hamann and Parminder Flora, "Automated Performance Analysis of Load Tests", Proc. ICSM 2009, Edmonton, IEEE, Canada, pp.125-134, 2009.