



## Software Performance Testing Measures

Mrs. Charmy Patel, Dr. Ravi Gulati

Shree Ramkrishna Institute of Computer Education and Applied Sciences, M.T.B College Campus,  
Athwalines, Surat, Gujarat, India.

[charmypatel@gmail.com](mailto:charmypatel@gmail.com)

Department of Computer Science, Veer Narmad South Gujarat University,  
Surat, Gujarat, India.

[rmgulati@gmail.com](mailto:rmgulati@gmail.com)

### ABSTRACT

Software developers typically measure a Web application's quality of service in terms of webpage availability, response time, and throughput. Performance testing and evaluation of software components becomes a critical task. Poor quality of software performance can lead to bad opportunities. Few research papers address the issues and systematic solutions to performance testing and measurement for modern components of software. This paper proposes a solution and environment to support performance measurement for software. The objective is to provide all kind of important measures which must be tested at the coding phase instead of after completion of software. So developers can make software that can meet performance objectives.

### Keywords

Software Performance, performance measures, Performance Analyser

---

# Council for Innovative Research

Peer Review Research Publishing System

Journal: [International Journal of Management & Information Technology](#)

Vol. 8, No. 2

[editor@cirworld.com](mailto:editor@cirworld.com)

[www.cirworld.com](http://www.cirworld.com), [member.cirworld.com](http://member.cirworld.com)

## INTRODUCTION:

Performance tuning can sometimes be more of an art than science, due to the sheer complexity of the systems involved in a modern Web application. The expected performance of the system needs to be scaled down accordingly. A performance test can help determine whether the product meets the performance goals or not.

The objective is to provide different measures to evaluate performance, including functional speed, throughput, availability, reliability and resource utilization etc. It reports the development efforts on constructing a performance evaluation environment for software based on a set of well-defined performance evaluation metrics and techniques. To achieve performance driven development, at the time of application development, developer requires an indication about the performance of a page/file created by him/her. Before application deployment quality assessment of whole application is required to get the idea about functioning and performance of the project. To meet the requirements of performance testing, different software are available in the market but they all have their own limitations and variations in different aspects.

## ANALYSIS OF SOFTWARE PERFORMANCE MEASURES:

By the pilot survey and analysis among industry people it is observed that

- To achieve performance driven development, at the time of application development (coding phase), developer requires an indication about the performance of a page/file created by him. Before application deployment quality assessment of whole application is required to get the idea about functioning and performance of project for software quality improvement.
- Different tools and techniques are available for checking performance of Web or Windows application but there is no such kind of framework available that covers all the measures to analyze and evaluate application performance in every aspect related to images, script, audio, video, multimedia, code execution, query execution, external resources integration, server response time, memory usage, CPU usage, cache utilization etc.

So, after doing the pilot survey of different testing tools and discussion with industrial people we are trying to find the common and important performance measures that lead the developers in performance driven development at coding phase.

The main focus of Performance testing of any website depends on its

- Speed – Determines whether the application responds quickly.
- Scalability – Determines maximum user load the software application can handle.
- Stability – Determines if the application is stable under varying loads.

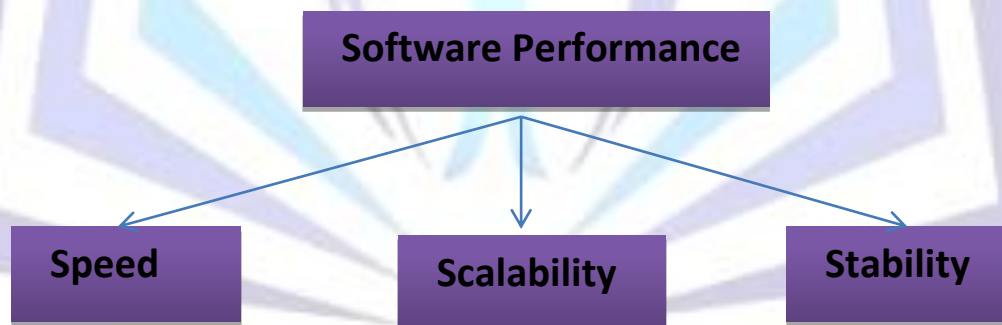


Fig 1: Measures of Software Performance Testing

Performance testing is the process to determine the speed or effectiveness of a computer, network, software program or device. [2]

### A) Speed:

- **Load time** is normally the initial time it takes an application to start. This should generally be kept to a minimum. While some applications are impossible to load within one minute, Load time should be kept under a few seconds if possible.
- **Response time** is the time it takes from the time a user inputs data into the application until the application outputs a response to that input. Generally this should be very quick. Again if a user has to wait too long, they lose interest.
- **Tolerable Waiting Time** for downloading a Web page may depend on various factors such as level of experience and age of users, individual user's characteristics means tendency to wait, task type, expected content of the Web page, expected download time, and information available about the wait.

## B) Scalability:

Scalability for website performance testing is to determine system behavior by increasing the load with a particular scaling ratio. For every scaling point all the performance attributes have to be determined. A software product suffers from poor scalability when it cannot handle the expected number of users or when it does not accommodate a wide enough range of users either simultaneously or even otherwise. Scalability is focused on understanding how an application scales as it is deployed on larger systems and/or more systems or as more load is applied to it. The goal is to understand at what point the application stops scaling and identify the reasons for this. As such scalability testing can be viewed as a kind of performance testing.

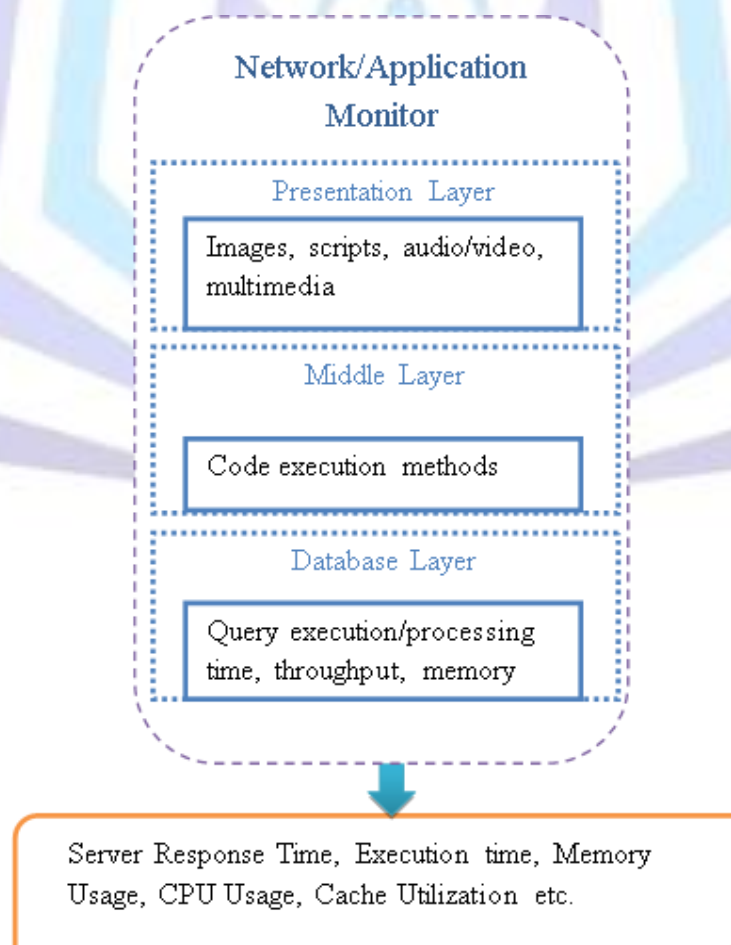
Following are the only few attributes out of many that can be considered for the scalability of website performance testing - Response Time, Throughput, Screen transition, Time (Session time, reboot time, printing time, transaction time, task execution time), Hits per second, Requests per second, Transactions per second, Performance measurement with number of users, Performance measurement under load, CPU usage Memory usage (Memory leakages, thread leakage) Bottlenecks {Memory, cache, process, processor, disk and network}, Network Usage (Bytes, packets, segments, frames received and sent per sec, Bytes Total/sec, Current Bandwidth Connection Failures, Connections Active, failures at network interface level and protocol level), Web server (request and response per second, services succeeded and failed, server problem if any) etc.[5]

## C) Stability:

Sometime web application is unable to identify when there are more requests than it can handle, it builds up memory (like a memory leak) and later it dies without any type of possible recovery. For that very simple measurements are there like system is able to identify high loads of request and server unavailable try again pages, until it recovers from the high load, a set number of users concurrent or page views that will allow us to have a clear metric of when to use scalability options like Load Balancer and Caching, Estimate your application usage behavior means number of concurrent users expected, typical user activity.

## PROPOSED FRAMEWORK FOR PERFORMANCE ANALYSIS:

With the use of all the possible measures and parameters we propose a framework which will include software performance analyzer during the coding and deployment phase itself.



**Fig 2: Proposed Framework of Performance Analysis**



Network and Application monitoring will be divided in to three different parts.

**Presentation layer** will analyze performance of all scripts and will scan different files (image, scripts, audio/video, multimedia etc.) to check for performance. **Middle layer** will analyze different code execution methods. **Database layer** will analyze query execution/processing time, throughput, memory usage.

All three layers will give detailed report about its execution time and resource utilization and Application can be scanned with this new approach which in turn will give detailed report on the status of application in terms of network/server response, memory usage, execution time, resource utilization, will display execution time and memory usage for each request/response and can also suggest for performance and quality improvement for application/software. [4]

In the phase of project maintenance and for Service Level management features of this concept can be utilized and expected outcomes can be achieved in previously deployed project. A performance driven approach will lead industry to quality and will help in stress reduction.

## CONCLUSION

Performance of many website depends on the load on the application at peak time under varying conditions. Performance testing is normally conducted in a reasonably simulated environment with the help of Performance testing tools. However Performance of Software and Web application depends on various parameters and each parameter must be tested under varying stress levels. With the help of all the mentioned measurements developer will get clear idea about performance of the application at coding phase itself and identify the actual area causing performance degradation or can cause in future and will provide possible optimum solution.

## REFERENCES:

- [1] Performance analysis of web Applications by Jagtshwar Singh Bedi & Aseem Khanna Department of Computer Science & Engineering, Guru Nanak Dev University, Amritsar, India
- [2] Website Performance Analysis Based on Component Load Testing: A Review Charu Babbar & Neha Bajpai – INTERNATIONAL JOURNAL OF COMPUTER TRENDS & TECHNOLOGY 2011
- [3] A study on tolerable waiting time: how long are Web users willing to wait? Citation: Nah, F. (2004), A study on tolerable waiting time: how long are Web users willing to wait? Behavior & Information Technology, forthcoming
- [4] Software Performance Testing Tools – A Comparative Analysis by Charmy Patel & Dr. Ravi Gulati – International Journal of Research & Development 2012
- [5] SCALABILITY TESTING APPROACH, TRACE TECHNOLOGY, Empowering Software Quality
- [6] Five Steps to Solving Software Performance Problems By Lloyd G. Williams, Ph.D., Connie U. Smith, Ph.D. June, 2002
- [7] Performance Analysis Framework for Large SoftwareIntensive Systems with a Message Passing Paradigm Christian Del Rosso
- [8] ] Lloyd G. Williams, Ph.D., Connie U. Smith, Ph.D. Five Steps to Solving Software Performance Problems June, 2002
- [9] J. D. Musa, A. Iannino, and K. Okumoto, Software Reliability Measurement, Predication, Application, McGraw Hill.
- [10] Best Practices for Software Performance Engineering, Connie U. Smith, Performance Engineering Services.