ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 9, September 2018

# Bottom-up Approach for Performance Testing of Software Applications or Products

Mohammed Abdul Razack Maniyar[1], Mohd Abdul Hakeem[2], Dr. Mohd Khalid MubashirUz Zafar[3]

Ph.D Research Scholar, Department of Physics, Dravidian University, Kuppam, India[1]

Team Lead, Accenture, Hyderabad, India[2]

Associate Professor, Maulana Azad National Urdu University, Hyderabad, India[3]

**ABSTRACT:** The present study gives an effective approach for Performance testing of Software applications or Products. The bottom-up approach would help in identifying the core Performance bottlenecks in software applications irrespective of technology and specifications. It covers end-to-end including client-server setup. This approach helps in fine-tuning and optimizing the system from software code level to the server level. It also helps in achieving the expectations of the end user with respect to the application interface speed, end-to-end transit times and for more consistent performance. This approach for Performance testing of software applications or products would thus be helpful in identifying the performance bottlenecks at an early stage thereby saving cost and time.

**KEYWORDS**: Performance Testing, Bottom-up approach, Performance bottlenecks, Non-functional testing, Performance testing tools, Server fine-tuning.

## I. INTRODUCTION

Performance Testing of Software applications or products is one of the important Non-Functional testing in the Software Testing Life Cycle. Performance Testing helps in gauging the performance of the software application end-to-end and determines whether or not the performance SLA's (Service Level Agreements) with respect to application Speed, Stability, Scalability requirements are met under expected workloads.

Different types of Performance testing usually conducted are Load Test, Stress Test, Stability or Endurance Test. The main objective of Load testing is to identify Performance bottlenecks of the software application under the anticipated user load. The objective of Stress testing is to identify the breaking point of the application. It involves testing the application under extreme workloads and the objective of Stability or Endurance testing is to identify the longevity issues like Memory Leaks, Garbage Collection issues and few more. It involves testing the application for the expected load over a long period of time.

This study shows a bottom up approach in evaluating the application performance for the defined SLA's. The application is evaluated for the performance, layer-wise. Presentation Layer, Business Layer and Data Layer are first targeted and layer wise optimization is done. After the initial load test execution, the code level optimization is done. The servers, network and other layers are targeted after this.

Applying the Pareto principle (80/20 rule), the core performance issues or loose ends in the system has to be identified first which is going to save a lot of time and helps in achieving a seamless, consistent performance. According to this study, it is necessary to first focus on the core ground level issues, before focussing on the High level architecture. Fine-tuning the application at server or architecture level, ignoring the lower level layers is not always helpful in achieving the consistent performance. The study thus focuses on the efficient approach of first drawing a Baseline for the existing system with layer-wise approach and then going ahead with the later tasks or activities like Scalability and Availability and Capacity planning.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 9, September 2018

## II. RELATED WORK

Sharmila and Ramadevi [1] studied on the analysis of Performance testing for web based applications. The Performance testing concepts, its objectives, goals, tools and the different types of performance testing like load, stress and capacity were included in their study. Analysis was also done on the different possible tools available for conducting and performing the load, stress and capacity tests. Both open source and vendor tools were used in their analysis. Application characteristics like Response time, Throughput, Maximum Concurrent users, Resource utilization, and application breaking point were taken into consideration for evaluating the performance of the web applications.

Sarojadevi H [2] studied on various Performance testing methodologies and its tools. The study has highlighted various challenges to mission-critical applications with the advanced technologies like .NET, J2EE and XML. Study has further suggested adhering to the Industry Compliances and Standards for various domains to maintain Quality. The key features of the load testing tools were also studied upon. The factors for successful load testing were also determined. Sources of Performance problems in application and the strategic ways to tackle these were also presented. The study has further suggested HP LoadRunner for Cloud Performance Testing. It also highlighted the challenges pertaining to Performance Test automation.

Kambhampaty S and Modali [3] studied on the Performance modelling of Web based application developed on J2EE and .NET. They suggested an approach for Performance modelling to address the Performance issues at an early stage in the Software Development life cycle. This approach helps in identifying the critical performance issues in architecture, design and implementation by providing pointers to trade-offs that may need to be made to achieve the Performance objectives. The approach involved identification of Key Use cases, Scenarios and capturing the Performance metrics like Response time, Throughput and resource utilization.

Shruti N Pardeshi [4] studied on the various strategies used for software testing. Different tools available in market for different types of testing were also listed in this study. The Non-functional testing, especially, Performance testing were discussed along with its goals and objectives. Main goals mentioned are, measuring response times, observing the system resources and measurement of network delays between Client and Server. The common mistakes which happen during performance testing were also highlighted. Performance testing were categorized into two, Load and Stress testing. The tools required for performance testing with respect to the technology specification were studied.

Sheetal S Patil, Joshi SD and Dhotre SS [5] have predicted the Performance of Web applications by Performance Modeling with Jmeter. The overview of Performance Model for web based application throughout its Software Development Lifecycle is presented in their study. The WebHelix process integrated with the Performance model was considered. The load testing of web sites was done in a systematic and repeatable manner with Jmeter considering the challenges. Studies predicted that the effective performance modelling integrated with process modelling is essential to understand whether a web-based system will meet its Performance objectives when deployed in the real time environment.

Ramakanth Kumar and KalpanBhargav [6] have surveyed on the Performance testing approaches of web application and the importance of WAN simulation in Performance testing. The survey has described various approaches used for Performance testing of web-based applications. The importance of WAN Simulation in Performance testing is also highlighted in the survey. Various approaches in Performance testing like load, stress and capacity testing were also discussed. In Stress testing, the Performance of application is measured against the gradually increasing load.

ManjuKaushik and PratibhaFageria [7] have done a comparative study on Performance testing tools. Performance of some specific testing tools were analysed and comparison was done with respect to various parameters. The results generated for the Performance parameters with the help of performance testing tools were evaluated and analysed. Various parameters like- Throughput, Response Time, Number of Hit Pages, Error Rate, Memory and CPU utilization etc., were considered. Performance testing tools like, NeoLoad, WAPT and LoadUI were used for the study and Performance analysis and comparison was done. The comparison results provide information for the selection of better tool for Performance testing of web applications as per the requirements.

Daniel A. Menasce [8] has studied on the load testing of web sites by considering the QoS factors. The QoS factors load testing addresses have been described for various business needs at several requirements level. It

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 9, September 2018

describes about the 'Availability' which measures the percentage of time customers can access a web-based application. It also describes about the 'Response Time', which has to be measured to determine how customers perceive things with respect to downloads and searches.

Shikha Dhiman and Pratibha Sharma [9] have studied on Performance testing by doing a comparative study and analysis of Web service testing tools. On the basis of response times, a comparison is done between three performance testing tools i.e., Apache Jmeter, Grinder and HttpRider. The same web services were used for Performance testing and the results obtained from these three tools were compared. The comparison results help in identifying and selecting the best tool for the technology specifications.

Sheetal S Patil and Joshi SD [10] have studied on the identification of Performance improving factors for web-based application through Performance testing. Jmeter is used as a tool for Performance testing and various factors were studied to improve the performance of web applications. It further describes how Performance modeling helps the developers to identify the performance bottlenecks of the system. Performance testing objectives as well as important constraints were also discussed in this study.

ShagunBhardwaj and Aman Kumar Sharma [11] have done a comparative analysis of Performance testing tools. A comparative study is done between three different Performance testing tools- Apache Jmeter, Http Rider &Fwptt. Performance parameters like Latency, Throughput, and Response times were taken into consideration. The metrics obtained for these parameters through these tools were compared and the best tool is chosen for Performance testing. It was concluded that that each tool has different internal structure and different algorithm to calculate the performance parameters.

Harsh Verma et al. [12] have presented an analytical approach for Performance testing. As the Performance testing tools like, JMeter, Tsung, MQTT do not provide a generic interface to mimic all real-time scenarios, a solution was provided. Solution to this is the analytic approach used to perform user journey in the application, which helps in identifying bottleneck and stress point easily. In this paper, the focus is on an analytical load testing tool and approach used for load testing various services supporting any protocol. The study demonstrated the architectural design pattern, generation and execution of test plan and load simulation by analytic Load Runner tool. The framework mimicked with the real-time user scenarios and generate specific load.

## III. PERFORMANCE TESTING APPROACH

After the Functional aspects of an application, the end users are more focussed and concerned over the Performance of the application. A good functional performing application or product with performance issues never satisfies the end user or customer. To ascertain consistent performance, an end-to-end performance testing has to be done and bottlenecks are to be identified and as required fine-tuning and optimization have to be done. For this a good approach is required. The bottom-up approach caters the performance requirement needs and thereby helps in fine-tuning or improving the performance of the application. As a part of this approach following step-wise activities are performed (Fig.1).

*(a) Initial Load Test Execution:*
After the safe load test, the initial planned test is executed and baselines are drawn for the existing specifications and set up. Different KPI's (Key Performance Indicators) like Response times, Throughput, Hits/sec, No. of Users, Transactions Per Sec, Errors/sec, Pass precent and Latency are taken into consideration.

*(b) Code Optimization:*
Based on the metrics captured from the initial load test, a layer level drill-down (Presentation layer, Business layer and Data Access layer) is done. Different code optimization tools can be used for addressing the performance bottlenecks and for its optimization.

*(c) Server Fine-tuning:*
Metrics obtained from servers while monitoring, especially for Memory, Disc, Processor and Network resources for various Web, Database and Application servers are analysed for their threshold values. Server logs are also analysed for errors, exceptions and SLA violations. A thorough drill-down is also done to identify the concerned areas or bottlenecks. These have to be addressed to improve the server's performance.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

**International Journal of Innovative Research in Computer and Communication Engineering**

*(A High Impact Factor, Monthly, Peer Reviewed Journal)*

*Website: www.ijircce.com*

**Vol. 6, Issue 9, September 2018**

*(d) Sizing and Capacity Planning:*
Once a stable system is obtained by fine-tuning the application and servers, scaling can be done. Both Horizontal and Vertical scaling has to be taken into consideration to check the scalability of the application or product. Accordingly, the sizing and capacity planning has to be done based on the software and hardware requirements and as well as the defined SLA's.

After each patch-up and fixes, comparison tests should be done to check with the baseline drawn. This can be followed with Stress and Stability tests if required and planned.

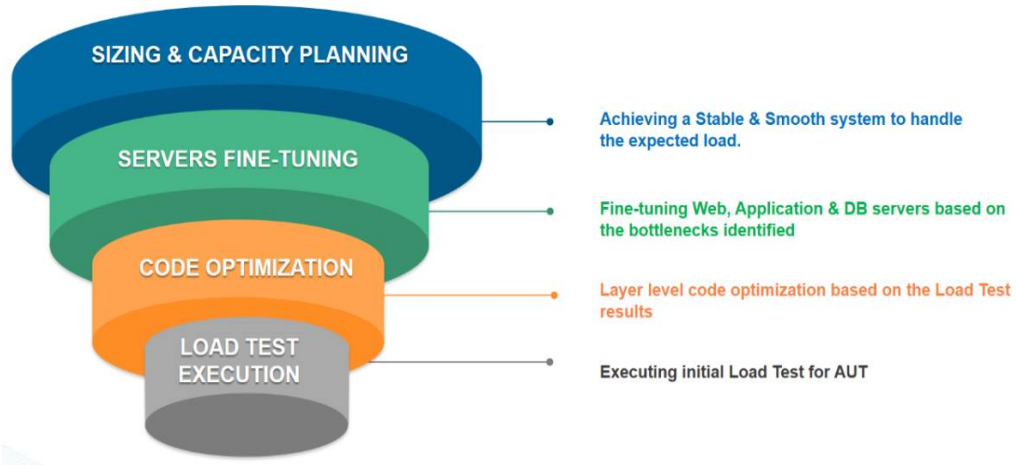


Fig.1. Bottom-up approach for Performance Testing of Software Application or Product.

**Advantages of Bottom-up approach:**
Few advantages of following bottom-up approach for Performance testing of software application or product are given below.

*(a) Early identification of Performance bottlenecks:*
This approach helps in identifying the core issues in the initial stage itself, thereby giving a complete scope of fine-tuning it and improving the performance at an early stage.

*(b) Saves a lot of time and money:*
Identifying performance issues in the initial phase saves a lot of time and money. These initial phase issues left unidentified will cost much if detected at later or final stage.

*(c) Performance efficiency improvement in software applications or products:*
Addressing issues at server level first without covering the bottom layer or code level layers would not help much in fine-tuning the application completely and in achieving a consistent performance.

*(d) Helps in understanding the complexity of the architecture:*
The layer level drill down and phase by phase approach will help in understanding the complexity of the architecture. It also identifies the limitations of the software and hardware resources at an early stage.

*(e) Helps in identifying the threshold values and to check whether the SLA's are met:*
The metrics report at each phase during the implementation of this approach helps in identifying the threshold values for the identified metrics. This approach also helps to check whether the SLA's are met. The violation of the SLA's can be taken note of in the initial phase of testing.

*(f) Helps Project Stakeholders to take important decisions:*

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 9, September 2018

The Performance reports generated at each phase helps the Stakeholders to take decisions at project milestones and on other important aspects pertaining to Quality of the software and the system.

(g) *Gives flexibility of using Hybrid Frameworks:*
A complete flexibility of usage of hybrid frameworks is possible by using this approach. Also, Performance testing Accelerators and Integrators can also be used.

**Different Tools used for Performance testing:**
Both, Open source and Vendor tools can be used for Performance testing based on the requirements and specifications. While following this bottom-up approach, different types of performance testing tools has to be used to gather performance metrics to gauge the performance at each layer or level. Different types of tools used are given below.

(a) Load test scripting tools
(b) Code optimization tools
(c) Web-page diagnostics tools
(d) Network virtualization tools
(e) Database profiling tools
(f) Server or Resource Monitoring tools
(g) Test Data Generation tools
(h) Mobile Testing tools (For Mobile applications)
(i) Analysis tools
(j) Reporting tools

| | | | | |
|---|---|---|---|---|
| Micro Focus LoadRunner | Apache Jmeter | CA Wily Introscope | SQL Profiler | AndroSensor |
| HP LoadRunner | Blazemeter | Dynatrace | SolarWinds | Trepn Profiler |
| Micro Focus Performance Centre | NeoLoad | Perfmon | Paessler PRTG | Monkey Runner |
| HP Performance Centre | LoadUI | HP Sitescope | PAL(Performance Analysis of Logs) | HPE Network Capture Express |
| Microsoft VSTS | Fiddler | Nagios | Android Studio | Mobile Wavelock Detector |
| WebLoad | Dynatrace | AWS Cloudwatch | Perfecto Mobile | HTTP Watch |

Fig.2. Commonly used Performance Testing tools

Fig.2. Few important and common tools used for Performance testing of software applications or products. These include the load generation, profiling, diagnostics, and resource monitoring and analysis tools. It also includes few commonly used Mobile testing tools. Before selecting a tool, proper evaluation has to be done. This has to be done based on the tool selection criteria. This evaluation helps in selecting the appropriate tool for the performance testing of software application.

**Best Practices:**
Some of the best practices while implementing this approach are given below.

(a) A Checklist covering end-to-end performance testing activities has to be prepared covering each phase of performance testing.
(b) The Performance SLA's (Service Level Agreements) has to be defined for the application.
(c) Tool evaluation has to be done to choose the correct and appropriate tool for Performance testing.
(d) Proper strategy and work load model has to be identified while following this bottom-up approach.
(e) The initial level performance issues have to be fixed before proceeding for the next level or phase of Performance testing.
(f) The Performance issues have to be addressed according to the severity of the risk level.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 9, September 2018

(g) Automation tools can be used for Test data management. This would help in managing the data more efficiently and would help in filtering the bad or junk data.
(h) Proper planning has to be done for logs management. Logs from different areas helps in drill-down and in test analysis for bottleneck identification.
(i) Status of the application performance at each phase or level of performance testing has to be updated or reported to the respective stakeholders.
(j) Standard thumb rules have to be used to evaluate the correctness of the performance test results at each phase.

## IV. CONCLUSION

The use of multiple technologies in the software industry for the development of software application or product is making the system more complex due to several factors. The bottom-up approach for the performance testing helps to uncover the core performance issues in architecture, design, implementation and configuration settings of the system and in fine-tuning and optimizing the system at initial phase of Performance testing life cycle with less iterations of testing and with more efficiency. With more advantageous factors, this approach would be more efficient in achieving the performance testing objectives and goals and helps in stabilizing the system with a seamless and consistent performance.

## REFERENCES

[1] Sharmila S and Ramadevi E, "Analysis of Performance testing on Web Applications", International Journal of Advanced Research in Computer and Communication Engineering, Vol.3, Issue 3, pp. 5258-5260, 2014.
[2] Sarojadevi H, "Performance Testing: Methodologies and Tools", Journal of Information Engineering and Applications, Vol.1, No.5, 2011.
[3] Kambhampaty S and Modali V.S, "Performance Modeling for Web based J2EE and .NET applications", In Proc. of world Academy of Science, Engineering and Technology, V. 8, Oct. 2005.
[4] Shruti N Pardeshi, "Study of Testing strategies and available Tools", International Journal of Scientific and Research Publications, Vol.3, Issue 3, 2013.
[5] SheetalS Patil, Joshi SD and Dhotre SS, "Prediction of Performance for web application by Performance Modeling with Jmeter", International Journal of Scientific & Engineering Research Volume 3, Issue 12, 2012.
[6] Ramakanth Kumar P and KalpanBhargav, "A survey on Performance testing approaches of web application and Importance of WAN Simulation in Performance testing", International Journal on Computer Science and Engineering, Vol. 4, No. 5, 2012.
[7] ManjuKaushik and PratibhaFageria, "Performance Testing Tools: A comparative study", International Journal of Innovative Science, Engineering & Technology, Vol.1, Issue 4, 2014.
[8] Daniel A Menasce, "Load Testing of Web Sites", IEEE Internet Computing, Vol. 6, No. 4, pp. 70-74, 2002.
[9] Shikha Dhiman and Pratibha Sharma, "Performance Testing: A comparative study and Analysis of Web Services Testing tools", International Journal of Computer Science and Mobile Computing, Vol. 5, Issue 6, pp. 507-512, 2016.
[10] Sheetal S Patil and Joshi SD [10], "Identification of Performance Improving Factors for Web Applications by Performance Testing", International Journal of Engineering Technology and Advanced Engineering, Vol. 2, Issue 8, 2012.
[11] ShagunBhardwaj and Aman Kumar Sharma, "Performance Testing Tools: A Comparative Analysis", International Journal of Engineering Technology, Management and Applied Sciences, Vol. 3, Issue 4, 2015.
[12] Harsh Verma, Arun Krishnan and Sivasubramanian, "Analytics Performance Load Test", International Journal of Computer Applications, Vol. 176, No. 2, 2017.