The performance testing helped the client identify and resolve performance bottlenecks which otherwise crippled the business. The ability to support 500 concurrent users was a performance improvement of almost 60 times when compared with the first capacity test run. Needless to say, the client was delighted by the end result!
## Contents:

- Executive Summary ................................................................. 3
- Need for Performance Testing ...................................................... 4
- ZenQ’s Performance Test Process .................................................. 7
  - Discovery & Planning .............................................................. 9
  - Design .................................................................................. 12
  - Execution ............................................................................ 13
  - Test Results Analysis ............................................................ 14
- Load Test Tools, Framework and Hardware Environment .................. 15
- Framework Components ............................................................. 17
Executive Summary

In the age of SaaS & Cloud, high Performance attribute for an Application/Website is no more a luxury; it’s a basic need for business success. Customers are turning intolerant towards slow performing web applications and shifting their loyalty towards competitive alternatives which are fast performing & enable them to complete their transactions sooner.

Bearing the above fact in mind, it is an absolute must for every product vendor to incorporate performance yielding best practices into their Application Design/Development/Deployment processes and validate the performance of application through formal methods before go-live, to be 100% sure that the client expectations & SLAs are met.

As a trusted QA partner of several ISVs & Web/SaaS Product Vendors, ZenQ has been supporting its Clients in identifying & fixing performance bottlenecks, assess capacity of their applications and judge their application’s ability to support anticipated user loads in different infrastructure configurations.

This paper describes the strategy and process followed by ZenQ for its Performance Testing service deliveries to unearth the performance issues of an application, assist developers in fixing the root causes of performance bottlenecks, and thus enabling our Clients to rollout worry-free product releases.
Need for Performance Testing

Impact of Slow Performance of an Application can be seen from different angles.

Some of the viewpoints are outlined below -

★ The Business Angle – Impact on Revenue

For B2C Applications, Good performance and better page speeds have always proved to have positive impact on increasing the revenue, and at the same time, slow performance of application causes significant revenue loss, customer satisfaction and above all, damages the brand reputation.
The following are few examples that indicate the impact of application performance on revenue (data/images courtesy: www.strangeloopnetworks.com)
The User Experience Angle – Impact on Customer Satisfaction

Enterprises use applications to make workflows faster, simpler and easier. Slow page loads & sluggish performance would have negative impact, may cause significant slowdown in the business processes. Such application behaviour may also induce users to abandon the application and resort to manual process. If application performance issues persist for longer period, your customers may look for alternatives and may switch their loyalty towards better performing applications of your competitors.

The following chart illustrates the impact of page delay on customer satisfaction (data source: Microsoft)
ZenQ’s Performance Test Process

With rich experience acquired through successful delivery of several performance testing projects and process maturity achieved, ZenQ has devised a 4-step process to deliver best results from every performance testing project. The Fig 1.0 below depicts the performance test process flow and each of the phases are described in detail in the subsequent sections below.
Fig 1.0: Performance Test Process Flow

1. Commence Project
2. Discovery & Planning
   - Gather & Analyze Performance Test Requirements
   - Perform Feasibility Study and Determine the best suitable Load Test Tool
   - Define Performance Test Strategy/Plan
3. Design
   - Develop Load Test Scripts
   - Develop Test Data
   - Design Load Test Scenarios
   - Prepare Load Test Environment
4. Execution
   - Execute Performance Tests & Collect Metrics
5. Results Analysis & Report
   - Analyze Test Results and Prepare Results Report
   - If Application Performance objectives are not met i.e. Performance Issues are identified?
     - Yes: Fix the Performance Issues identified
     - No: Re-run Performance Tests and Validate fixes/compare with benchmark test results
6. End Project
Discovery & Planning

ZenQ’s Performance CoE team will initiate review of the application, understand Client expectations from the Performance test project by gathering the Test Requirements, Analyses the requirements and Defines the strategy to be employed during the tests. Typically this phase will take 3 to 5 business days. Detailed steps & sub-phases involved in this phase are outlined below –

**Performance Test Requirements Gathering & Analysis:**

In this sub-phase, ZenQ gathers the test requirements from the important stakeholders of the Application such as Project Manager(s), Business Analyst(s), Product Architect(s) and the Developers. A formal Test Requirements questionnaire document is shared with the Client - which has several questions related to Application, its architecture, reasons behind initiative for Performance testing, User loads and work flows to simulated etc.

These requirements are very important and help us understand the expectations of the Performance Testing Project and also the context in which Performance Testing is being initiated.

**Typical requirements that we encounter in majority number of performance testing projects are –**

- Does the Application support anticipated Peak user load with the existing hardware configuration?
- What is the Capacity OR Maximum number of Users supported by the Application in the current hardware/software configuration?
- What is Application Performance (Page response times and Transaction times) at Peak user loads, Normal and Off-peak user loads?
**Feasibility Study to Determine suitable Load Test tool:**

After Analysing Test Requirements and Understanding Application Architecture/Design, ZenQ conducts a feasibility study to determine the best suitable tool for Load testing project. The different parameters taken into consideration while determining the tool are –

**Technical Parameters**

- Application Platform & Architecture
- Application UI Design & Technology used
- Communication Protocol between Client & Server

**Non-Technical Parameters**

- Project Budget Allocation and Constraints
- Technical Expertise of In-house Development/QA teams

If the Client has a preference for a specific Load test tool, and if it’s already established as the preferred tool that supports the application design & architecture, the above step may be skipped.

**Test Strategy Definition and Planning:**

This is the most important step in the Project delivery, where in, ZenQ defines the Test Strategy to be employed during the Testing. While defining the strategy, the following aspects are taken into consideration –

- Performance Test Requirements and Expectations
- Application Architecture/Design and Technology/Platform
- User profiles, User-case scenarios, Environment, and Work-loads to be simulated
- Timeline and Budget allocation for the project
Test Strategy significantly varies from project to project, based on the performance test requirements. It’s very important to review the requirements and arrive at the right kind of test strategy, as it affects the collection of relevant test data, and affect the conclusions drawn from the testing. Once the Test strategy is defined, the Performance tests are designed to meet the desired objectives of the project.

The key decisions taken during test definition process include –

- Type of tests and the number of tests to be executed
- User loads to be simulated
- Users Work-flows to be simulated and their load distribution percentages
- Duration of each test, User ramp-up/ramp-down rate, Maximum User load
- Type of networks to be simulated
- Metrics to be collected at the client side and at different tiers of the application (i.e. at the APP server, Web Server, DB server and other Servers if any exist such as Cache Server, Load balancers etc.)

Upon completion of the Strategy definition & Test Planning, a detailed Performance Test plan will be crated and sent to the Client for review & approval, and any feedback provided by Client will be incorporated into the Test plan.
Design

After the Test plan is approved, ZenQ team starts working on developing required load test scripts, test data etc. in preparation to execute the planned Load tests. This phase (with all its sub-phases described below) may take between 5 – 15 business days depending on the number of test scripts to be developed, test data to be created and application complexity.

Test Scripts Development:

In this phase, ZenQ develops the Code/Test scripts to be used in Load test tool for simulating User work-flows as planned.

Test Data Creation:

After the Test Scripts development, ZenQ creates required Test data per the requirements of the Use-cases being simulated.

Test data may include the following –

★ User credentials
★ Documents/Files (to be uploaded if any)
★ Any Specific information filled in any of the Forms within the Application

Test Scenario Development:

Once the Test scripts and Test data are created, based on the Type of Load tests to be executed and User workflow requirements - Load test scenarios are configured within the Load test tool with details of User loads, Work-flows, Test Data configuration, Network and Browsers simulation configurations etc.

Test Environment Setup:

Test Environment Setup includes any application related configuration, Load test environment on the Cloud (with required number of load generators based on the count of concurrent user load to be simulated) and Installation & Configuration of required Performance Metrics collection tools.
Execution

Performance test scripts are executed in this phase. Each load/performance test is executed per the test parameters defined during the planning phase such as test duration, user ramp-up rate and maximum user load simulated etc. While the tests are in progress, monitoring tools/agent software setup in different tiers of the application will collect General system level metrics such as CPU, Memory, Disk and Network usage and Application specific related metrics related to App, Web and DB servers. To ensure the accuracy of the metrics collected, each test is repeated 3 times and average values of metrics are collected from the 2 or 3 successful iterations of tests executed will be used to analyse the results and to identify the performance bottlenecks.
Test Results Analysis

In this step, Performance metrics collected during Load tests are analysed to identify the performance of the application and root cause of performance bottlenecks/issues (if any exist). ZenQ uses top-to-bottom digging approach of metrics analysis, starting with analysis of surface level of metrics and going deeper in to the Application architecture to pinpoint the source of performance issues.

Based on the results analysis, ZenQ submits a comprehensive Test Results report to Client that contains information & graphs related to page variation of page response times with user load, server-side system resources utilization with user load, metrics related to App/Web/DB software etc. (Additional details related to graphs and metrics included in the Report are explained in section Results Report & Metrics Collected)

If Application developers fix the performance issues identified during the tests, ZenQ re-runs the same tests and validates the fixes. We also compare the results/metrics from the current run with those of previous run, and verify if there is an improvement OR degradation in the performance of the application due to fixes. This procedure is repeated for all the incremental fixes made to the application and continued until the Application meets the desired performance goals.
Load Test Tools, Framework and Hardware Environment

Load Test Tools:
ZenQ has experience in the use of the following Load test tools -

- Microsoft's Visual Studio 2012 Ultimate
- HP's Load Runner 11.5
- Apache’s JMeter

In addition to the above commercial/open source tools, ZenQ also has an in-house developed Load test tool.

Load Test Framework
The following is the framework followed for Load test execution and metrics collection –
Fig 1.1: Framework Architecture:
Framework Components

**Load Test Controller:**
Load Test controller manages tests running on all load test agent machines and controls the count of concurrent users against AUT. It is also responsible for collecting test results from all agents and storing them in the Load test repository.

**Load Test Agent:**
Load test Agent runs as a service and generates requests against AUT simulating concurrent virtual users executing the specified use-cases. Weightage can be added to each test agent, based on the weight value user load is distributed across the agents.

**AUT (Application under Test):**
Application (and its components) against which Load test is executed.

**Load Test Results Repository:**
Load test results data collected by the Controller are stored in the Results Repository database.

**Performance Metrics Collection Agent:**
Performance Metrics Collection Agents are installed on different servers/tiers of AUT, and are responsible to collect General system performance metrics such as system resources utilization and APP/Web,DB server software related metrics. These metrics will be communicated to Performance Metrics Collection Server.

**Performance Metrics Collection Server:**
Performance Metrics Collection Server collects performance metrics from all metrics collection agents and stores them in performance metrics repository.
Load Test Hardware Environment

ZenQ uses Cloud-based Hardware for Load test execution & metrics collection. We have business relationships with few leading Cloud Infrastructure vendors for provisioning the required Load generation infrastructure as needed.

Load Test Results Analysis Report

ZenQ Load Test Results Analysis Report typically contains the following sections –

Executive Summary
This section contains a summary of load test results such as performance of the application (page response times), system resource utilization and high-level details of performance issues/bottlenecks observed during load test.

Tests Details
This section contains information about the load tests executed with complete details related to User loads and use-cases simulated, User ramp-up/ramp-down rate, Test duration etc.

Detailed Test Results
This section contains information about the load test results with exhaustive list & details of the metrics collected, graphs for metrics in comparison with user load variation and elaborative information related to performance issues identified and ZenQ analysis of performance bottlenecks/root cause of issues (analysed based on the metrics collected during load tests).
The following is the typical list of metrics and graphs included in the Load test results reports:

**Note:** The following list represents only a partial list of metrics & graphs included in the actual load test report and metrics/graphs may also vary based on the load tests requirements & tests executed.

**Metrics Collected:**

<table>
<thead>
<tr>
<th>Location/Tier where Performance Metric is Collected</th>
<th>Metrics Collected</th>
</tr>
</thead>
</table>
| Client Computer                                     | ★ Avg. Page Time (for all pages involved in use-cases)  
|                                                     | ★ Errors/Sec      
|                                                     | ★ Avg. Transaction/Use-case Time  
|                                                     | ★ Throughput      |
| Application Servers (APP/DB/Web and any Other Servers if exist) | ★ Processor % processor time  
|                                                     | ★ Physical Disk Read Latency  
|                                                     | ★ Network Interface Output Queue length  
|                                                     | ★ Physical Disk Avg. Disk Read/Sec |
| App/Web Server                                      | ★ .Net/Java Memory usage  
|                                                     | ★ App/Web Server Process CPU/Memory Usage  
|                                                     | ★ Other App/Web/Framework specific metrics |
Few e.g. Graphs included in Report:

**Graph: User Load vs. Avg. Page Time**

**Description:** Avg. Page time is the time taken to load all the individual components of page. The graph plotted against Avg. Page time and User Load provides the information about page load time of a page at various loads.
Graph: **User Load vs. Errors/Sec**  
Description: The graph plotted against User Load Vs Errors/Sec provides the information about frequency of errors observed with the increase of user load during the load test.
Graph: User Load vs. Average Test/Use-case completion Time
Description: The following graph plotted against User Load vs. Average Test/Use-case completion Time provides the information about variation of transaction/test time with increase in user load.
About ZenQ

ZenQ is a global provider of high quality Software Development & Testing Services, offering cost effective value-add outsourcing solutions to our clients. Our highly competent IT Professionals, Domain experts, combined with industry best practices & our investments in state-of-the-art technologies makes us a dependable and long-term IT service partner to all our clients.

For more information, email us at: sales@zenq.com OR Visit us at www.zenq.com