Performance Testing of ASP.Net Application

Performance Testing to Identify Bottlenecks of Current Production System and Recommend Performance Enhancements

System Overview

System is a web based distributed application catering services to employees of organization to avail insurance benefits over internet.

System is developed using Microsoft ASP.Net 1.1 technology. System uses three layers like ASP.Net, .Net Enterprise Services and SQL Server 2005 at the backend.

Communication between ASP.Net and .Net Enterprise Services happens through ".Net Remoting Mechanism".

Performance testing engagement had following challenges:

1. Product developed using Microsoft ASP.Net 1.1 and hence limitation to choose Performance Tools that Support ASP.Net 1.1

2. Organization could not find performance test tool that will support heavy view state.

Architecture
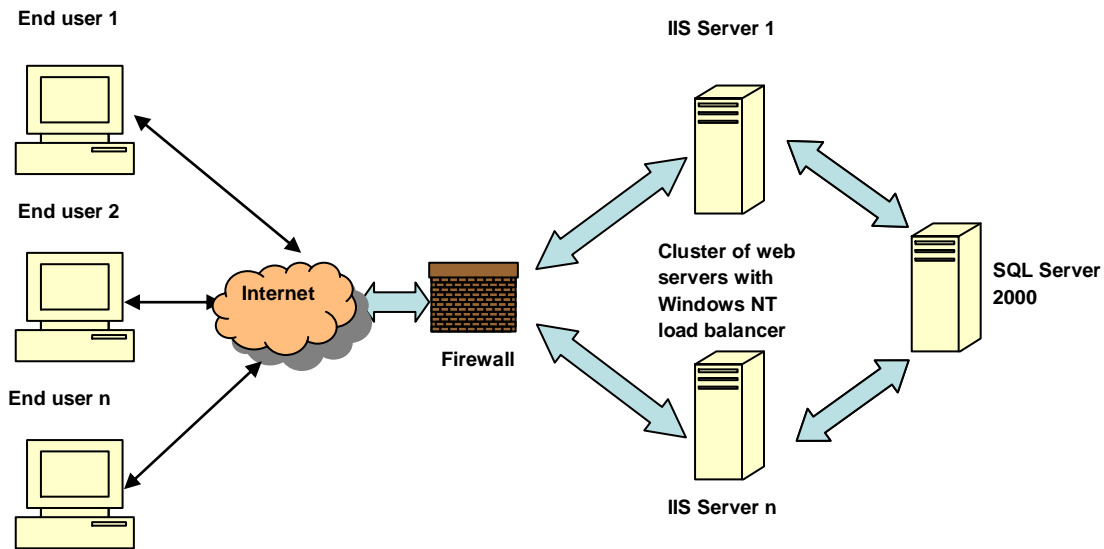
Goals

- Select tool for performance testing

- Present performance test cycle plan

- Execute performance test plan

- Identify bottlenecks in the system performance at various levels

- Publish new performance benchmarks

- Build in-house performance testing practice

Solution and Approach

1. Understood performance pain areas of the system in production

2. Defined Performance Strategy for the application and presented in Performance Plan document

3. Based on the technology and architecture, identified the performance tool (VSTS 2008) after trying multiple tools (ACT, WAPT, WebPerformer, AppPerfect)

4. Understood application key user scenarios

---

5. Documented key scenarios

6. Generated sufficient data to generate considerable load

7. Developed and parameterized performance scripts

8. Executed performance scripts with different load to identify bottlenecks using response times and server counters

9. Did code profiling of the code to identify pain areas in the code using dotTrace

10. Did static code analysis of the code to align code with Microsoft standards using FxCop

11. Performed profiling and analysis of database queries (with SQL Profiler and SQL Query Analyzer) to identify bottlenecks at database side

12. Repeated performance tests and profiling activities to publish new bench marks and trends of load vs. response times, throughput.

Skills Exhibited

- Deployed ASP.Net applications over IIS web server cluster

- Configure Windows NT load balancing

- Workload modeling

- Configured and used VSTS Load Tester for developing load tests

- Used Windows Perfmon for collecting metrics

- RDBMS / T-SQL, Database Indexes

- Used SQL Profiler for understanding performance of various SQL queries and stored procedures

- Used SQL Query Analyzer for evaluating performance of individual queries and stored procedures

Achievements / Value-Adds

- The system has not faced issues in production after the performance testing exercise was carried out

- The system is able to handle predicted peak load

- Client has in-house performance testing team for taking on further such engagements independently