

Performance Testing & Performance Engineering

CQE provides highly technical performance test and engineering solutions to assure the performance, capacity and stability of applications and platforms. Taking an engineering approach throughout the delivery lifecycle, services include early performance validation through to continuous performance testing via the CI/CD pipeline and full end-to-end performance and load testing.

Service features

- On site and offsite, delivery
- Flexible and scalable resourcing from individuals to teams
- Focus on Quality Engineering
- Operational Acceptance Testing
- Early Performance Testing as part of the CI/CD process
- Performance Testing- Load, Stress, Volume, Soak, Scalability and Risk Testing
- Resilience and Chaos Testing
- Infrastructure verification
- Performance test tool selection and application
- Performance test capacity planning and management

Service benefits

- Expertise in Performance strategy, planning and execution of cloud applications
- Ensure systems perform against specific performance needs.
- Performance issues identified and rectified prior to peak periods.
- Avoid reputation damage associated with system failure under load.
- Confirm cloud solutions stability and performance prior to public launch.
- Advanced Performance-testing services ensure realistic performance test.
- Early performance tests built into the continuous delivery process
- Automated Performance Tests and Customizable Frameworks
- Complex Performance Testing Solutions, Generating Varying Loads On Cloud
- Performance Bottleneck Analysis of cloud applications

Delivery Methodology

Performance testing cannot be just an afterthought or performed by just any person who has been given access to a performance testing tool. It requires strategic planning and experienced testers to execute correctly. The results also need to be reviewed and interpreted by seasoned testers and engineers.

A comprehensive, successful performance test starts with a well-defined plan. CQE lay the groundwork, executes the plan, and reviews the results.

Scope

CQE believe that the determination of success requires a plan or a goal. We first work with you and your team to identify the problem you are trying to solve. Are you planning on moving your application to new servers and need to validate their performance? Are you getting pages of timeout errors on your access logs and want to find out why? Do you want to offload server SSL and compression? Or, are you building a new system from scratch? Whatever the problem or goal is, we work with you to lay it out in simple terms. Success criteria is outlined and all of this is independent of any vendor specification.

Plan

Once we know what we need to do, then we can devise the plan to accomplish it. This step happens automatically after the goals have been determined. The plan should be as simple as possible, in clear and concise terms. It should include all of the specific tests necessary for achieving the goals, any configuration settings that will need to be checked or modified, and all of the tools that will be used for the test.

Execute

Once we have planned the execution, next we execute the plan. On a system being newly built, the performance testing is phased in conjunction with the development.

Phase I – Unit Test – Generally this phase of performance testing is completed by developers after they complete components and before integrating them.

Performance testing at this phase uses tools such as a memory profiler, code profiler, and coverage profiler. Each of these tools can help identify an area at risk for significant issues when many users are accessing it at the same time. Testing and fixing at this stage is much easier and less expensive.

Phase II – Application Integration Load Test – After the functional integration test has been completed and any functional issues are fixed, the integrated load test can begin. This phase is a full load test using the planned number of users to simulate an expected production load. For best results, the test is performed twice. The first time through, the test is executed with minimal monitoring. At this point, we are aiming to make sure it can hold up to an expected load. The second time through we add detailed monitoring. At this point we should be able to identify any bottlenecks and begin performance tuning.

Phase III – Production Staging Load Test – At this point, the team has moved the application to an environment that mimics production and performed functional testing. This phase of load testing is where we really see how the application performs in a shared environment where resources are in competition. We turn up the load in this phase and make sure the system can handle the expected production traffic.

Phase IV – Capacity Assessment – Once we've come to this point, we know that the application or system meets expectations, but we want to go one step further. This phase determines the system's breaking point. By assessing the capacity early, you can set points in the future when you know new hardware will need to be added in the future, or plan for new architecture. During this test we steadily increase the load on the system until resources are saturated, response times slow, and data transfer rates drop. The information obtained during this test will aid in capacity planning for the future of the application.