White Paper

Fundamentals of Performance Testing

The Increasing Need for Proper Performance Testing due to Increasing Software Complexity in the Enterprise

There have been two significant changes in software development that have forced the industry to rethink performance testing and how it is conducted. This paper discusses emerging testing strategies and how they fit into new development approaches. These include agile development, iterative models, and leveraging Service Oriented Architectures to achieve the highest levels of success for projects.

pureIntegration

INTRODUCTION

In order to stay competitive, retain current customers, and increase their customer base, multi-system operators (MSOs) must continually roll out new applications, features, and services. There is a growing need to integrate quality Performance testing into the Systems Development Life Cycle (SDLC). Managers, Directors, VPs, and CTOs are being constantly challenged to deliver quality products and features to their customers.

MSOs are in a full sprint deploying and updating:

- New Billing Systems
- New Provisioning Capabilities
- New VOIP and Wireless Capabilities
- Customer Facing Portals
- Value Add Technologies

Executives in charge of these efforts face many challenges. Software engineering and the software development process have changed significantly in recent years. Applications have become increasingly large and more complex. Additionally, applications have become more customer-centric and may service hundreds of thousands to millions of requests per day. It is important to provide the customer with a consistent and quality experience.

A performance strategy spanning the SDLC is a critical factor in achieving a positive customer experience, yet it is often inadequate or missing. This paper will discuss two critical roles that make up a successful performance strategy; Performance Engineering and Performance Testing.

Performance Engineering is conducted during the planning, requirements, and design, phases of a project. Performance engineers bring significant enterprise design, architecture, development, as well as performance testing experience. Their role is to understand the planned application and the environment it will run in. If the application will utilize common resources that are already deployed, the performance engineer will coordinate and work with the appropriate groups to understand available resources, capacity, and how the new application will impact that.

Software performance testing is testing performed against a defined portion of a software system. This “defined portion” is commonly referred to as the Application Under Test (AUT). The goal of performance testing is to evaluate the performance

characteristics of the AUT as well as the systems with which it interacts. Performance testing takes several factors into account such as concurrency, throughput, system response, resource utilization, capacity and growth, as well as many other factors discussed later.

As the applications continue to increase in complexity and become more dependent on external resources, traditional performance testing is no longer sufficient to adequately evaluate the performance characteristics of enterprise applications. This paper will discuss the skills, tools, and best practices required to performance test modern applications. It takes a critical look at traditional performance testing and highlights its shortcomings to adequately performance test modern applications. It describes a more holistic, mature, and consistent engineering discipline applied to performance testing.

This white paper will highlight many of the performance challenges faced by MSOs as they roll out new products and features. It will identify some of the traditional performance testing models and why they fail to address the new software development landscape. Finally, it will highlight emerging performance test best practices and processes. A modern performance strategy will provide confidence to stakeholders that is has:

- Tested the AUT and dependent system components with an appropriate level of test coverage.
- Provided useful, meaningful, and relevant data while ensuring the accuracy of the data reported.
- Provided quality data for performing accurate capacity planning.

These enable stakeholders to confidently assess risk and take the guesswork out of deployment decisions.

NEW CHALLENGES IN PERFORMANCE TESTING

There have been two significant changes in software development that have forced the industry to rethink performance testing and how it is performed. First, applications are becoming more complex as they have moved from isolated "stove pipe" designs to large-scale enterprise applications. Second, "traditional" software development lifecycles have been turned on their head as more companies adopt agile methodologies and iterative development processes.

There have been two significant changes in software development that have forced the industry to rethink performance testing and how it is performed.

INCREASING APPLICATION COMPLEXITY AND DEPENDENCIES

This paper will use the term enterprise application to refer to the framework the AUT runs within and the services it utilizes. The introduction of the enterprise application has added an additional layer of complexity and work effort to performance testing. The enterprise application provides a set of services to the application being developed so the development team doesn't have to develop these services themselves. For example, most enterprise application frameworks provide services such as database connectivity pooling, message queuing, object lookup services, session management, clustering, and fault tolerance. While these services provide a great deal of power and flexibility, each one can adversely impact the performance of a system. Each service needs to be taken into consideration, monitored, and tuned based on how it is being utilized by the application being developed. This is often an overlooked area, but can drastically increase the performance with proper tuning.

Enterprise applications also distribute components of the application across multiple physical tiers. This is referred to as an N-tier architecture, the most common being a 3-tier architecture. The application's functionality is presented to the end user in the *presentation tier*. Interaction with this tier provides the end user experience. It should be responsive, intuitive to use, and meet the needs of end users. Next is the *application tier*. An application's business logic is implemented within this tier. It receives requests from users by way of the presentation tier. It is responsible for receiving user requests, processing them according to the application's business rules, and returning appropriate responses to the users. Third is the *data tier*. This tier provides any long term data storage the application requires. With the recent uptake of Service Oriented Architecture (SOA), there are now additional layers with which applications may need to interact. Some of these

services may be offered by external vendors, while others may be offered in house.

Enterprise applications introduce a unique challenge for performance testing. The performance testing team needs a holistic understanding of the technologies that make up the AUT as well as the architecture, services, and components it relies on. Simply running a performance test tool against an application provides little insight into where performance issues are lurking and in turn how to improve system performance. If a tester submits a form in a web-based application and it takes one minute to return the result, where is the delay occurring?

- Is it in the AUT or is the application performance being constrained by an external system or component it relies on? For example, are calls to the database introducing unexpected delays?
- Is the performance bottleneck in a call to a web service? Is the web service available? Is it unable to keep up with the load being placed on it?

MOVING TOWARD AGILE DEVELOPMENT AND ITERATIVE LIFECYCLE

As more companies move toward agile development and an iterative lifecycle, they often find their testing teams have a hard time adopting these new approaches. Agile development is a paradigm shift in the way software is developed. To achieve success, testing teams need to adopt new agile testing approaches and understand the team's role in the agile development process. A key to the success of any agile project is open and ongoing collaboration between all teams, throughout the entire lifecycle and iterations.

Traditionally, testing teams and software developers would take on an "us against them" mindset. To be successful in the new era of software development, teams need to work more closely throughout the entire lifecycle. There needs to be a spirit of collaboration instilled in each individual across all teams. Each team must embrace the concept that they are all working toward a common goal: driving the release of the highest quality product possible.

THE AGILE PERFORMANCE TEST TEAM

In support of core performance testing, there are fundamental characteristics and skills required to ensure the overall success of the project.

- *Continuous Collaboration:* First and foremost, the team needs to continuously collaborate and communicate with the developers and stakeholders. This close collaboration is fundamental to a successful agile project. The agile testing team has more responsibilities, to a broader audience, than traditional performance testing teams.
- *Timely Reporting and Feedback:* The testing team needs to be able to provide timely feedback to the development team and other stakeholders. This requires knowledge of and proven experience in agile performance testing, its processes, tools, and best practices that enable timely reporting and feedback.
- *Vigilant Identification of Performance Risks:* The testing team needs to seek clarity for unknown, overlooked, and unaddressed issues that could impact system performance. The team should proactively evaluate, escalate, and track risks. As a central component between all of the teams, the performance test team can assist with risk assessment and mitigation.

These are critical to successfully integrating the performance testing into an agile process.

"It is commonly believed that the earlier a defect is found the cheaper it is to fix it. For example, if a problem in the requirements is found only post-release, then it would cost 10–100 times more to fix than if it had already been found by the requirements review." - Code Complete (2nd ed), Steve McConnell

TRENDS AND THE CHANGING LANDSCAPE

Performance testing is becoming more critical to businesses as they continue to roll out new applications. Appropriate performance engineering and performance testing spanning the entire SDLC significantly reduces the number of performance issues identified in production. Performance engineering and testing identify performance issues earlier in the lifecycle. This provides three major benefits to the project. First, identifying performance issues early in the lifecycle reduces schedule slips. Second, there is significant cost reduction if performance issues are discovered early in the lifecycle. Third, it allows enterprises to more accurately estimate and plan for growth and scalability through accurate management and interpretation of capacity data.

In *T.E.S.T.* magazine's December 2009 issue, the editors interviewed prominent testers to gain their insight on what to expect for 2010. Some key thoughts from these leaders:

"We have to change our mindset to accept the changes in the way we do our work … we have to use our creativity to deal with the increasing complexity of software." – Andreas Prins

"Defect prevention at every phase, more so right at the design phase, is emphasized as defects found in the final stages of the SDLC always prove costlier, 'Don't make your users your testers!'" – MakarandTeje

To achieve the greatest level of success, application performance must be considered at every stage the SDLC.

*"When there was little or no performance testing carried out, **100% of performance defects needed to be resolved in the production environment**…*

*In performance driven companies, where performance considerations are taken into account at every stage of the life cycle, **there was only a 5% performance defect rate post deployment**."*

A PERFORMANCE DRIVEN LIFECYCLE

In 2006 Forrester conducted research on resolution of performance defects based on performance testing maturity [Table 1]. When there was little or no performance testing carried out, 100% of performance defects needed to be resolved in the production environment. For companies that set aside time for performance testing late in the application lifecycle, there remained a 30% performance defect rate in production. In performance driven companies, where performance considerations are taken into account at every stage of the life cycle, there was only a 5% performance defect rate post deployment.

Resolving Performance Defects (Forrester)

Approach	% of Defects Resolved in Production
Firefighting	100%
Performance Validation	30%
Performance Driver	5%

Table 1 - Source: Forrester Research

LEVERAGING PROVEN METHODOLOGIES AND EXPERIENCE

Due to the new challenges in performance testing, it is critical to engage a team who understands the performance requirements of the AUT as well as the applications with which the AUT

interacts. Performance test teams need the correct mix of skills, experience, and technical expertise. As performance testing continues to evolve into an engineering discipline, performance teams must transition from traditional "point and click" testers to performance test practitioners who understand the overall architecture in which the AUT runs. This encompasses technologies the AUT utilizes, such as WebLogic, WebSphere, Relational Database Management Systems (RDMS), LDAP, networking, enterprise servers. They also need to understand the available tools and possess the skills to monitor these components to make performance tuning recommendations. The performance tester also needs to stay engaged in emerging trends, testing tools, performance testing best practices, and how to best utilize them to achieve the performance testing needs of the AUT.

Performance issues that go uncaught by inexperienced performance testers typically result from:

- Failing to consider performance needs during the application design phase. Performance engineers bring significant enterprise design, architecture, development, as well as performance testing experience. They are a valuable asset during the planning and design phases.
- Failing to perform adequate or realistic capacity planning.
- Failing to use the most appropriate tools and testing techniques that best meet the needs of the project.
- Failing to have experience and tools capable of testing next generation web technologies such as AJAX, Flash, Rich Internet Application frameworks, and similar technologies. This can leave major gaps in performance testing.
- Failing to allocate adequate time for performance testing or waiting until the last minute to execute performance testing.

KEY FACTORS FOR SUCCESSFUL PERFORMANCE TESTING

To ensure the highest level of success the performance test team needs to work with the stakeholders to come up with a test plan that appropriately addresses the following key points to success:

- Understand the performance testing environment, production environment, and scaling in relation to production. This is critical in determining an accurate load model.

- Work with stakeholders to understand and assist with determining an accurate load model.
- Identify the type of test data and amount needed to perform an accurate and representative performance test.
- Identify key load injection points.
- Identify additional load injection points used to capture performance characteristics of external components.
- Identify application logging needs to assist with performance analysis. Work with the appropriate teams to identify logging strategies based on the applications and environment.
- Identify the most appropriate set of tools to inject a load against the AUT and collect and correlate the Key Performance Indicators (KPIs) for all systems utilized by the AUT.

These key factors assist the performance test team in solving the fundamental performance challenge: *identifying the root cause of performance issues and coordinating the appropriate resources to resolve the issue*. The experience and planning of the performance testing team provides the ability to take ownership of and facilitate the efficient resolution of issues by providing a holistic view into the application's performance and the environment in which it runs.

Performance test teams must take the next step in performance testing, by applying an engineering discipline to performance testing. Modern test teams provide the following valuable services:

- *Detailed and Meaningful Performance Reporting:* Performance test reports containing a top down view that brings clarity across the Customer Experience, Application, Database, and Backend Services to drive collaboration and communication.
- *Facilitated Technical Review Sessions:* Based on the results of each test, quickly pull together the right resources from Engineering, Network, and Operations to perform a technical review. Lead the dialogue to uncover the root cause of the issue and recommend resolution.
- *Tuning Recommendations and Resolution:* Take ownership of the resolution of the performance issues from identification to resolution and re-test verification. In many cases, recommend the resolution through application design reviews, architecture discussions, and performance analysis.

It is important to evaluate the current utilization of the service which the new application will depend on, how much capacity is available, and how much additional load the new application will place on the system.

- *Creditability in the Performance Test Solution:* Provides the client with the results and confidence to make the decision to deploy an application to production.
- *Pre- and Post-Production Deployment:* Allows the performance engineering team to be involved in production rollout planning and assist the operations team in putting performance monitoring in place in the production environment. The results of production monitoring are used as a feedback mechanism for improving performance testing for the next iteration.

KEY TESTING ARTIFACTS DRIVING DIRECT BUSINESS BENEFITS

The following activities are critical in achieving optimal performance across each component of the application. Often the AUT depends on an established service that is shared across multiple applications. It is important to evaluate the current utilization of the service which the new application will depend on, how much capacity is available, and how much additional load the new application will place on the system. The following activities, artifacts, and services are utilized to achieve a well-balanced application environment while significantly driving down cost.

Customer Experience Focus:

- Average response times
- Transactions per second
- Concurrent transaction performance analysis
- Multi-transaction breakdown analysis
- Transaction response time consistency analysis
- Performance characteristic analysis under multiple load models (best, average, and worse case load)

Stakeholder Reporting:

- Facilitated technical reviews
- Approach strategy reviews to facilitate stakeholder acceptance at key points in the lifecycle
- Production metrics analysis to validate performance test strategy and results

Key Performance Indicator Collection and Analysis:

- Application transaction logging
- Server performance metrics (KPIs)
- Database and Query Performance analysis

Impact Analysis of Backend Services:

- Storage Area Network (SAN) analysis
- Network architecture analysis (firewalls, load balancers, routers, proxies)
- Web Service performance characteristics analysis, modeling, and emulation.
- Network latency analysis across multiple network architecture profiles (LAN, WAN, wireless broadband, Handset devices, etc.)

Performance Testing Solutions:

- Business transaction performance analysis
- Application and hardware diagnostics and monitoring implementation
- Application tuning analysis and implementation
- Holistic and relevant correlated reporting

Performance Testing Services:

- Performance validation of existing systems
- Advanced diagnostics and reporting implementation
- Targeted strategic performance analysis to identify critical bottlenecks across application environments for pre- and post-production applications
- Prioritized improvement analysis and recommendations
- Operational and maintenance cost reduction focused performance analysis
- Revenue impacting focused performance analysis

CONCLUSION

Performance testing is an iterative and evolving process. To achieve maximum success, performance must be taken into consideration at the project's inception. Agile development increases the roles and responsibilities of the performance test team. They are leveraged to collaborate and coordinate with all project teams including the Business, Operations, Engineering, Production Support, Functional Test Group, and Software Engineering to identify, escalate, and track performance risks.

Projects which utilize a performance-driven approach from the project onset achieve a high level of success as shown in the Forrester report. Without a performance-driven approach, performance tuning is unpredictable, expensive, and not guaranteed.

Performance testing is an iterative and evolving process. To achieve maximum success, performance must be taken into consideration at the project's inception.

pureIntegration is committed to achieving the highest level of success for your projects. Our experienced performance testers, industry experience, and passion for continually improving the services we provide through industry involvement and thought leadership help ensure this high level of success

We are excited to work with your organization and help solve any performance engineering challenges you are facing.

About pureIntegration

pureIntegration is a systems integration company with a track record of achieving measurable business results. With our deep industry expertise, we give our customers a competitive edge with bold solutions to their critical business problems. *pureIntegration's* differentiated approach strives to exceed service and delivery expectations through our proven methodologies, focus on innovation, partnership with client teams, and unmatched resources. *pureIntegration* is an HP Elite Partner.

For more information, visit *www.pureintegration.com* or call 703-707-9680.