A Practical Approach to Performance Testing

Leela A. Putten
The way to get started is to quit talking and begin doing.

Walt Disney
Agenda

- Why Performance Matters?
- Basics of Performance Testing
- Practical Roadmap to Success
The WHY?

The relevance of Performance Testing in Today’s Digital Evolution
Why Does Performance Matter?

The core aim is to proactively manage and mitigate Business and IT risks such as reputational, legal or commercial. For instance, one can avoid the risk of operating a slow or broken platform and mitigate the cost of IT failure, which will result in loss of revenue for a business.

**Impact on Revenue**
A 1 second delay results in drastic loss of revenue over time. (Aberdeen Group)

**Scalability**
Ensuring that as a business grows, the IT infrastructure can scale accordingly.

**Data Analytics**
Usable and relevant customer analytics tied to goals and strategies. Enables businesses to get to a more prescriptive and AI Driven approach.

**User Experience and Retention**
If a page takes too long to load (>3s), +/- 40% will abandon it and 80% users will not return.

**Availability**
Ensuring that online platform and apps are available 24/7 on different devices from different geo-locations in the IoT world.

**Disaster Planning**
Enables businesses to be better prepared for unforeseen events and test the resilience of the IT processes and infrastructure.
Customers now expect their experiences to be continuous, constant, customized, and cross-channel—the key for companies is to deliver a continuous customer experience.

Continuous Customer Experience & Continuous Improvement
Trigger Points/Conversation Starters

Are we ready to go live?

- Will our system handle the unexpected?
- How reliable is our system?
- Will our failover work correctly?

Why is our system so slow?

- Is it our hardware or our software – or both?
- What are the bottlenecks in our system?
- Will our failover work correctly?

Does our system meet performance requirements?

- Can we handle Black Friday sales?
- Does this release perform as well as the last one?
- Do our mobile users abandon our app due to poor performance?
- Can our Cloud Platform handle our system?

What will happen if our business grows?

- How does our system scale?
- What is our maximum throughput?
- What is our maximum user load?

What is the overall Customer Experience?

- Is the usability as per industry standards?
- Is the usability as per customer expectations?
Performance Testing Basics

Deconstructing the science behind performance testing.
Performance Engineering Vs Performance Testing?

These two terms are often used interchangeably. Below is how I position it to stakeholders:

<table>
<thead>
<tr>
<th>Typical Performance Testing</th>
<th>Typical Performance Engineering</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Why</strong></td>
<td>Often driven by business process optimisation requirements to provide better business value to customers and employees through a culture of performance engineering across the organisation.</td>
</tr>
<tr>
<td>Often driven by technical requirements to simulate or test a system under different production loads and behaves as per expected service level agreements.</td>
<td></td>
</tr>
<tr>
<td><strong>What</strong></td>
<td>Optimise the application for performance from the earliest software delivery phases and the way down the lifecycle.</td>
</tr>
<tr>
<td>Simulate performance loads to identify potential bottlenecks and assist in remediation through verification and validation activities.</td>
<td>Is an ongoing process that occurs through all phases of the software delivery cycles, from requirements to design, to development, to QA, to Production.</td>
</tr>
<tr>
<td><strong>When</strong></td>
<td></td>
</tr>
<tr>
<td>Distinctive testing process that occurs iteratively once a first round of development is completed.</td>
<td></td>
</tr>
<tr>
<td><strong>Who</strong></td>
<td>Everyone takes part from the beginning and starts from software designers, system architects, developers to QA &amp; Testing and Operations. You can still have dedicated Performance Engineer(s) driving the adoption across the organisation. This produces less rework and better ROI as performance is an integral part of the design and delivery.</td>
</tr>
</tbody>
</table>
Performance Testing Criteria – The WHAT

- Robustness
- Usability
- Functionality
- Capacity
- Security/Safety
- Availability
- Interoperability
- Scalability
Performance Testing in the SDLC – The WHEN

- Check the Design of the solution
- Check the Source Code
- Check Every Build
- Check Every Deployment
- Controlled Testing and Monitoring in Production

SHIFT LEFT

- Plan
- Design
- Construct
- Build
- Package & Install
- QA / Evaluate

SHIFT RIGHT

- Deploy
- Operate

Continuous Performance Practices

Performance Testing

Performance Monitoring

Business domain
Application domain
Database domain
Infrastructure domain
Network domain
Common Practices – THE HOW
Typical Performance World

Core

- Performance requirements
- Business process aligned
- Performance Strategy
- Detect, Diagnose and resolve issues
- Performance Testing Team

Benefits

- Stable and Improved customer experience
- Apps meets SLA expectations
- Ensures Business Resilience
- Supports a scalable business model

Performance Tests

- Baseline testing
- Isolation testing
- Load testing
- Soak testing
- Scalability testing
- Volume testing
- Network Latency
- Batch Processing
- Spike Testing
- Concurrency Testing
- Synthetic Monitoring
- Real User Monitoring
- Mobile App Monitoring
- Deep-Dive Monitoring
- Transaction Monitoring
- Infrastructure Monitoring
- Network Monitoring

Supports a scalable business model

Ensures Business Resilience

Detect, Diagnose and resolve issues

Performance Testing Team

Performance Strategy

Performance Testing

Typical Performance World

LoadUI

JMeter

Micro Focus

LoadRunner & Performance Center

BlazeMeter

dynatrace

iOCo
Build Your Practical Roadmap

Finding the best adoption practices that is fit for your organization and context.
Common Challenges for Enabling Performance Testing

Quick Fix
Business has a habit of buying more infrastructure as a quick fix.

NFRs are missed
Non functional requirements are still problematic and inadequately captured.

Complex & Costly
Test leadership shy away from this discipline and often prefer to avoid it.

3rd Party Liabilities
Clearance and scope of performance testing approval needed.

Skills Gap
Not enough performance testers and engineers on the global market.

Accuracy of Analysis
To add value, performance testing should factor the architecture differences between PROD and NON-PROD.
Strategic Roadmap

Decentralization of ownership of Application Performance

Debunk performance through education and compliance

Relook at your Software Delivery Strategy and Test Strategy

Define a maturity roadmap to introducing performance testing.

Drive enterprise buy-in of performance engineering as part of the corporate identity.

Change mindset from Performance Avoidance to Performance Resilience
Building the Practical Roadmap

→ Define a practical maturity adoption plan and strategy fit for your purpose.

**Practical adoption ideas:**
1. Drive a community of practice with core teams to discuss collaboration.
2. Identify key performance SLAs that can be tackled as part of a performance testing strategy.
3. Review your IT inventory and prioritise key applications to be tested.
4. Enforce application performance quality gates in project sign-off.
Building the Practical Roadmap

→ Leverage your existing testing disciplines to drive performance testing.

**Practical adoption ideas:**

1. Create a checklist of testing case studies through workshops with the functional, automation, performance teams.
2. Train testers on use of tools such as Jmeter
3. Set up performance awareness workshops to enable testing community.
Building the Practical Roadmap

- Drive automation of application performance testing and monitoring for wider test coverage and regression testing.

**Practical adoption ideas:**
1. Identify performance testing tools at unit, API and UI level.
2. Empower your developers with tools for early detection for static and dynamic testing.
3. Define performance testing metrics and reports to drive transparency.
Where to Aim for Success?

✓ Ambassadors, Evangelists & Early-adopters
✓ Automation !> Processes && People
✓ Start small with basic practical roadmap.
✓ Geo-location, Organization, Team & Application Context.
✓ Fit for purpose for Tooling and Processes
Thank You!

Contact Us for more details:
Leela.putten@eoh.com or
Leela.putten@qualiblaze.com

Twitter: @Leela_Putten