



# 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 lost 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

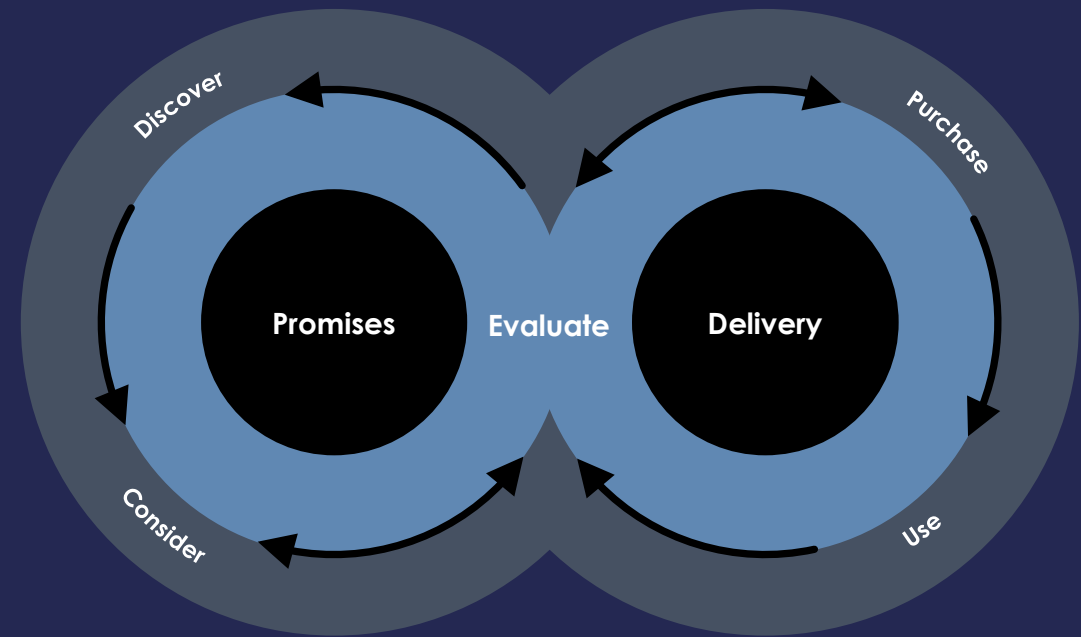
Ensuring that online platform and apps are available 24/7 on different devices from different geo-locations in the IOT world.

## Disaster Planning

Enables businesses to be better prepared for unforeseen events and test the resilience of the IT processes and Infrastructure

# Continuous Customer Experience & Continuous Improvement

- Customers now expect their experiences to be continuous, constant, customized, and cross-channel—the key for companies is to deliver a continuous customer experience.



Open content and channels

Branded content and channels

# 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

A network diagram consisting of white circular nodes connected by thin white lines, set against a dark blue background with a subtle bokeh effect of light blue circles.

Deconstructing the science behind  
performance testing.

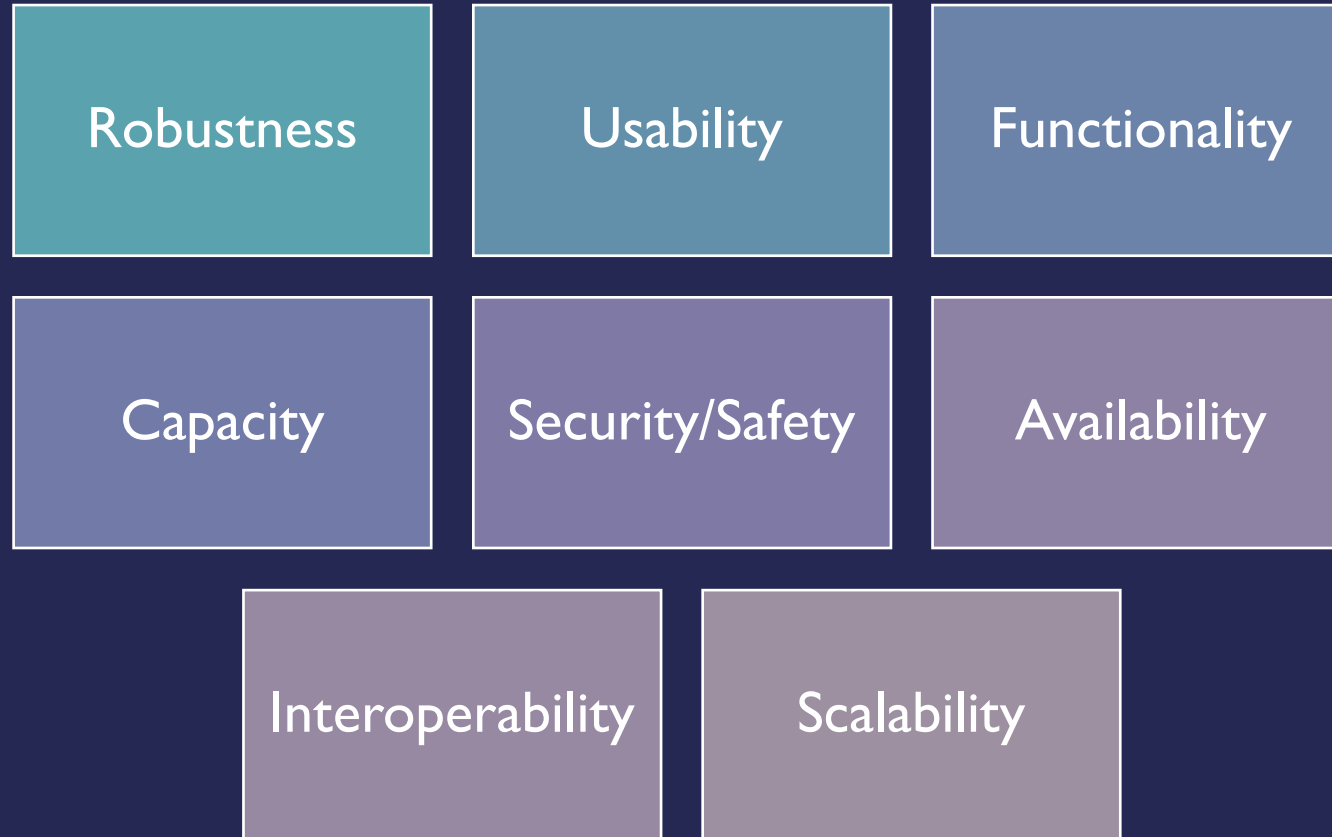


# Performance Engineering Vs Performance Testing?

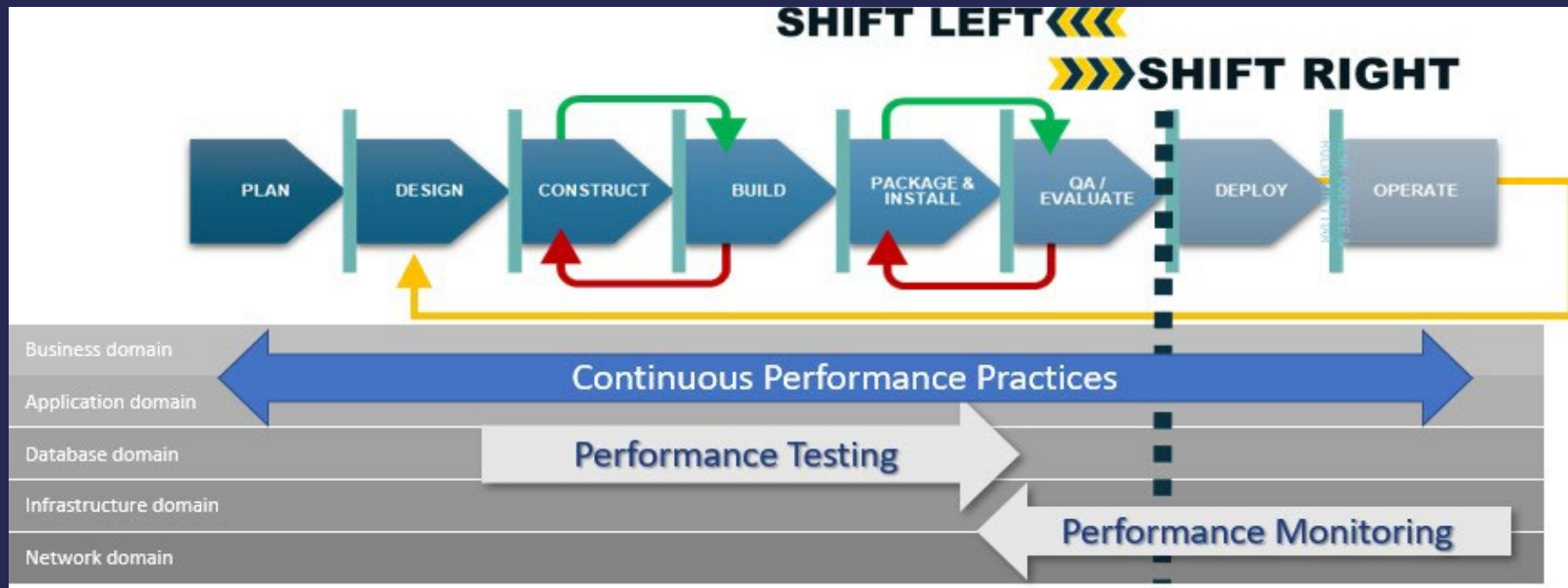
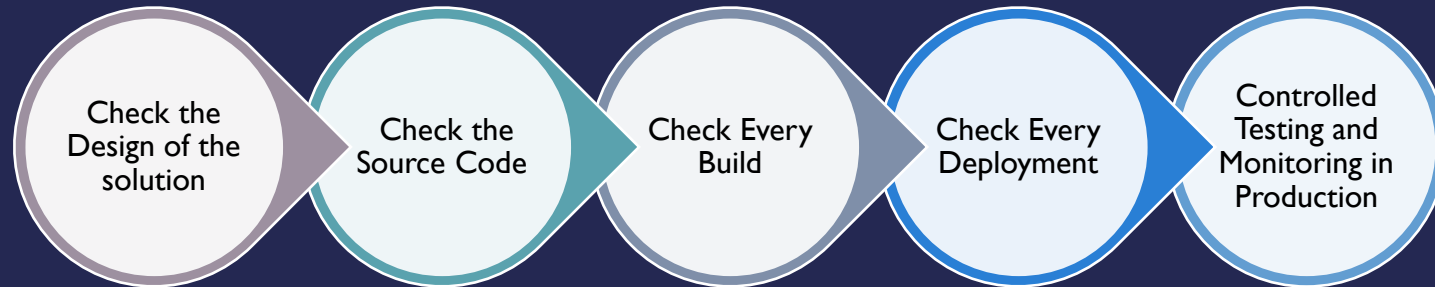
These two terms are often used interchangeably. Below is how I position it to stakeholders:

	Typical Performance Testing	Typical Performance Engineering
<b>Why</b>	Often driven by technical requirements to simulate or test a system under different production loads and behaves as per expected service level agreements.	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.
<b>What</b>	Simulate performance loads to identify potential bottlenecks and assist in remediation through verification and validation activities.	Optimise the application for performance from the earliest software delivery phases and the way down the lifecycle.
<b>When</b>	Distinctive testing process that occurs iteratively once a first round of development is completed.	Is an ongoing process that occurs through all phases of the software delivery cycles, from requirements to design, to development, to QA, to Production.
<b>Who</b>	Dedicated Performance Engineers and Testers execute performance testing and remediation. Functional testers, Test Automators, Security Testers, and Exploratory testers can help greatly in the performance testing of apps. Operations and development teams get involved in the remediation of performance issues.	Everyone takes part from the beginning and starts from software designers, system architects, developers to QA & 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.

# Performance Testing Criteria – The WHAT



# Performance Testing in the SDLC – The WHEN



# Common Practices – THE HOW



# Typical Performance World

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

Core

Stable and Improved customer experience

Apps meets SLA expectations

Ensures Business Resilience

Supports a scalable business model

Benefits

Baseline testing

Isolation testing

Load testing

Soak testing

Scalability testing

Volume testing

Network Latency

Batch Processing

Spike Testing

Concurrency Testing

Performance Tests

Synthetic Monitoring

Real User Monitoring

Mobile App Monitoring

Deep-Dive Monitoring

Transaction Monitoring

Infrastructure Monitoring

Network Monitoring

Performance Monitoring



A network diagram consisting of white circular nodes connected by thin white lines, set against a dark blue background with a subtle bokeh effect. The nodes are arranged in a complex, interconnected pattern, suggesting a network or data structure.

# Build Your Practical Roadmap

Finding the best adoption practices that is fit for your organization and context.

# Common Challenges for Enabling Performance Testing

2023



## 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.



## 3<sup>rd</sup> 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.

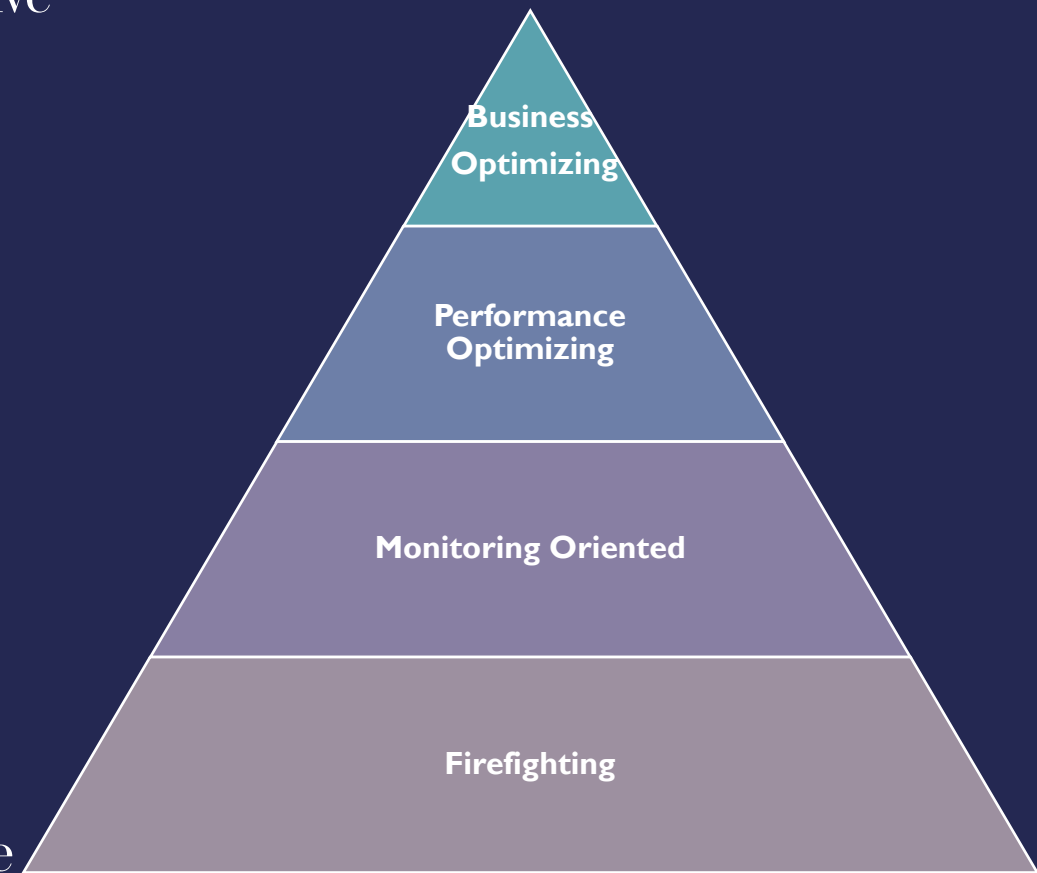
## **Practical adoption ideas:**

- 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.*

Proactive



Reactive



# 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.*

## **Gear Up Your Application Performance**

A practical approach to setting up a performance engineering and testing strategy for optimal application deliveries

**Leela Putten**

IOCO & QualiBlaze Consulting

### Performance Testing Types

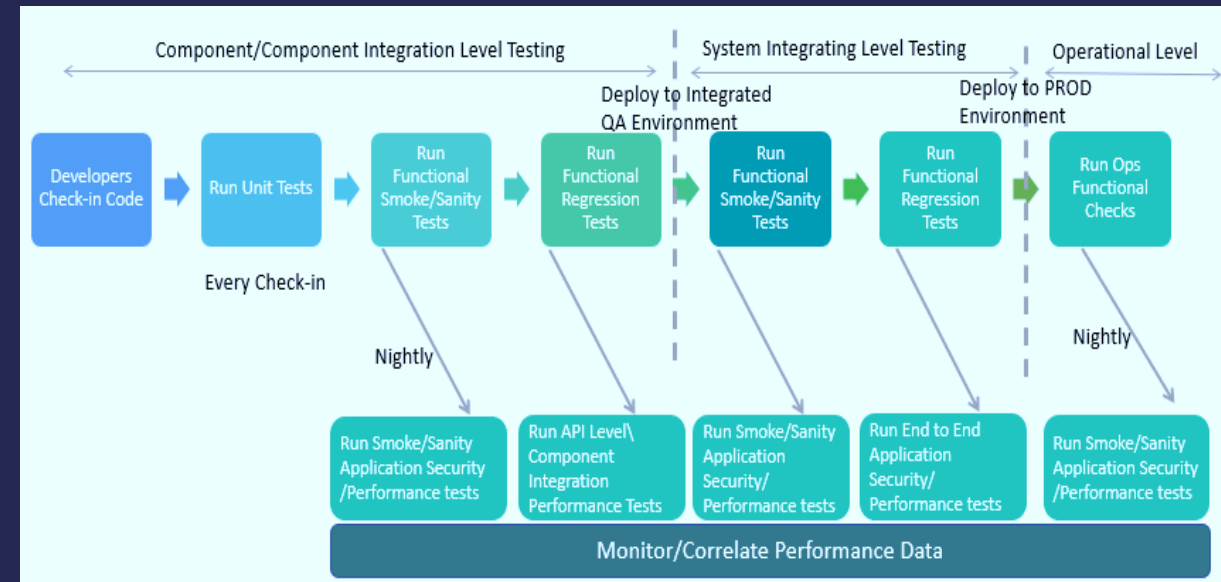
Types	Purpose
Baseline tests and Isolation testing	Test to establish performance baselines - Baseline tests are often mentioned but also ignored. However, they hold far more value than just establishing performance baselines and are one of the most important steps in this methodology. With some effort and time taken to examine details, up to 85% of performance problems can be identified and solved during baseline test runs. It involves executing each business process script in isolation, and validates with single user test execution, and ramps up the concurrency to see how the system responds.
Capacity Testing	Capacity testing determines how many <u>users</u> and/or transactions a given system will support and still meet the stated performance objectives. These objectives may also be stated with regards to the data volumes resulting from the transactions.
Load Testing	Load testing focuses on the ability of a system to handle increasing levels of anticipated realistic loads resulting from transaction <u>requests</u> generated by controlled numbers of concurrent users or processes.

# Building the Practical Roadmap

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

## **Practical adoption ideas:**

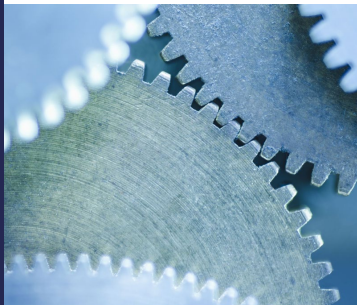
- 1. Identify performance testing tools at unit, API and UI level.*
- 2. Empower your developers with tools for early detection for static and dynamic testing.*
- 3. Define performance testing metrics and reports to drive transparency.*



# 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](mailto:Leela.putten@eoh.com) or  
[Leela.putten@qualiblaze.com](mailto:Leela.putten@qualiblaze.com)

Twitter: @Leela\_Putten

