# Performance – Mobile – Agile The Future

Presenter: Jaron Marsh

© Planit Software Testing

# Our knowledge is your power

Software Testing

Consulting, Advisory, Delivery,
Managed Services, Training

© Planit Software Testing

Planit

# Introduction

# Agenda

1. Traditional Performance Testing
2. Agile Performance Testing
3. Mobile Performance Testing
4. The Future

© Planit Software Testing

Planit

# Firstly – what is Performance Testing?

Simulating user behaviour of application

Testing against Non Functional Requirements (NFRs )

Analysing all results (test and monitoring results) to produce a view of performance behaviour

© Planit Software Testing

# Traditional Performance

- Occurs at end of project (can easily delay go-live)
- Production-like load and production-like environment
- Modelled off production stats
- Requires all functionality

© Planit Software Testing

# Traditional Performance

Steps involved:

1. Test planning (workload model, requirements etc.)
2. Test preparation (scripting, smoke testing etc.)
3. Test execution (peak load, soak etc.) – multiple iterations
4. Test reporting

© Planit Software Testing

# Traditional Performance

**Golden rule:**

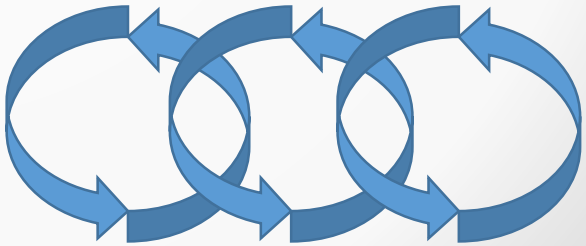
*'As production-like as possible'*

© Planit Software Testing

# Agile Performance Testing

# Agile Performance

- Occurs at end of each sprint/release
- Baseline/benchmark approach
- Not reliant on production load
- Doesn't require all functionality
- Traditional performance approach at the end



© Planit Software Testing

Planit

$$Perf(Env_{Const} + WL_{Const} + Code_{Variable}) = X$$

© Planit Software Testing

Planit

# Approach

- What are the tests?
- How are they run?
- Who runs them?
- What gets reported?

© Planit Software Testing

Planit

# Benefits

- ✓ Earlier identification of performance issues
- ✓ Final performance window smaller
- ✓ Shift left
- ✓ Continuous integration

© Planit Software Testing

Planit

# Risks

- Potential higher cost (performance resources engaged earlier)
- Developer assistance required earlier
- Convincing project of benefits

© Planit Software Testing

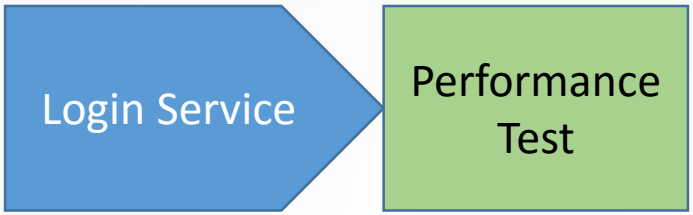
# Example

**RELEASE 1**

© Planit Software Testing

# Example

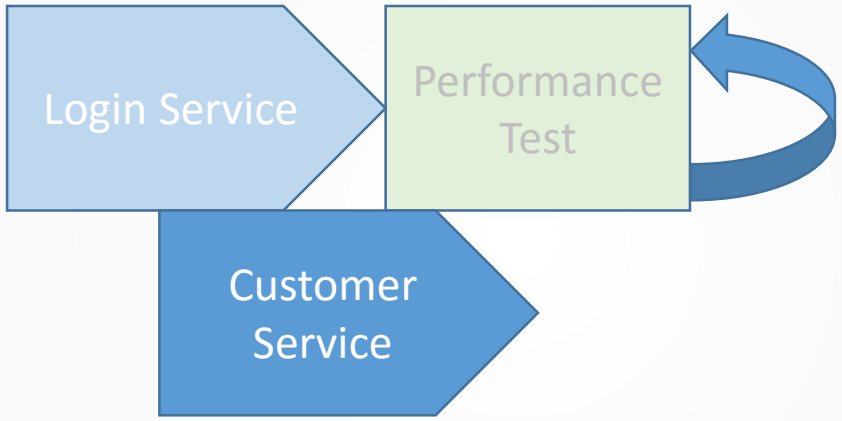
**RELEASE 1**




© Planit Software Testing

# Example

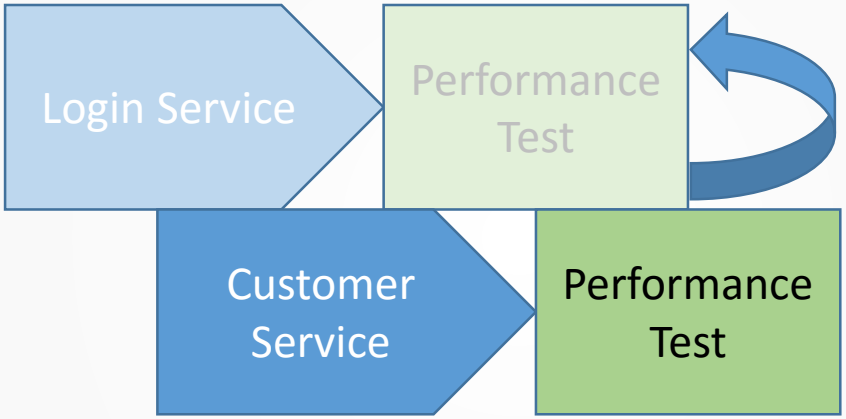
## RELEASE 2




© Planit Software Testing

# Example

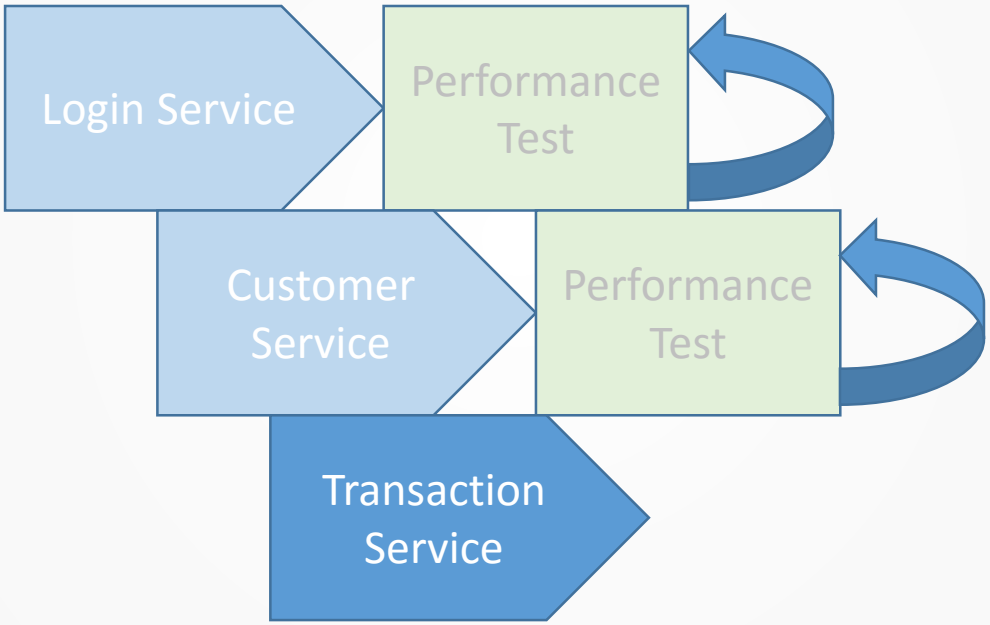
## RELEASE 2




© Planit Software Testing

# Example

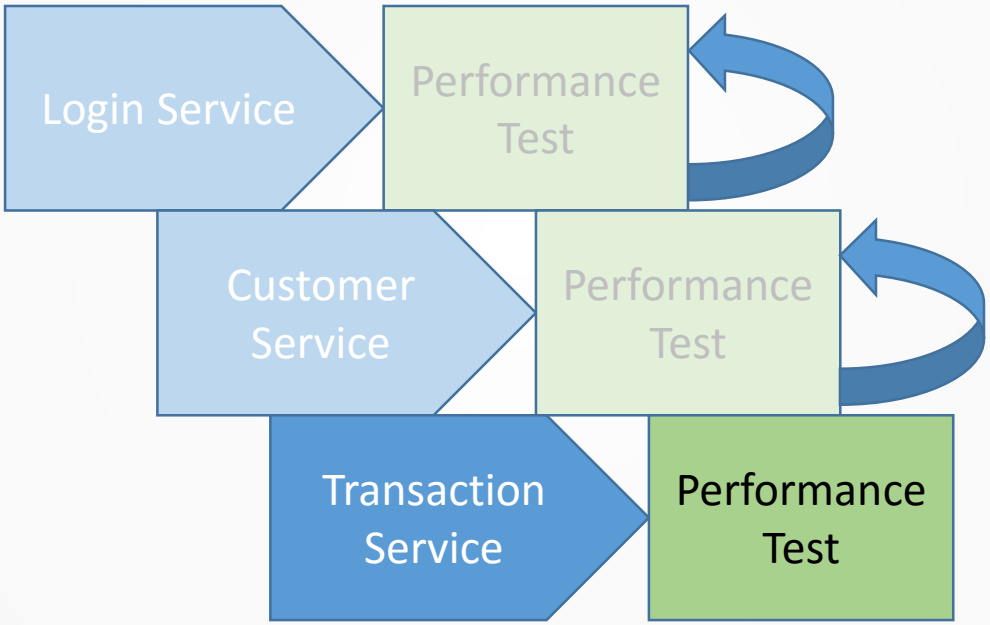
## RELEASE 3




© Planit Software Testing

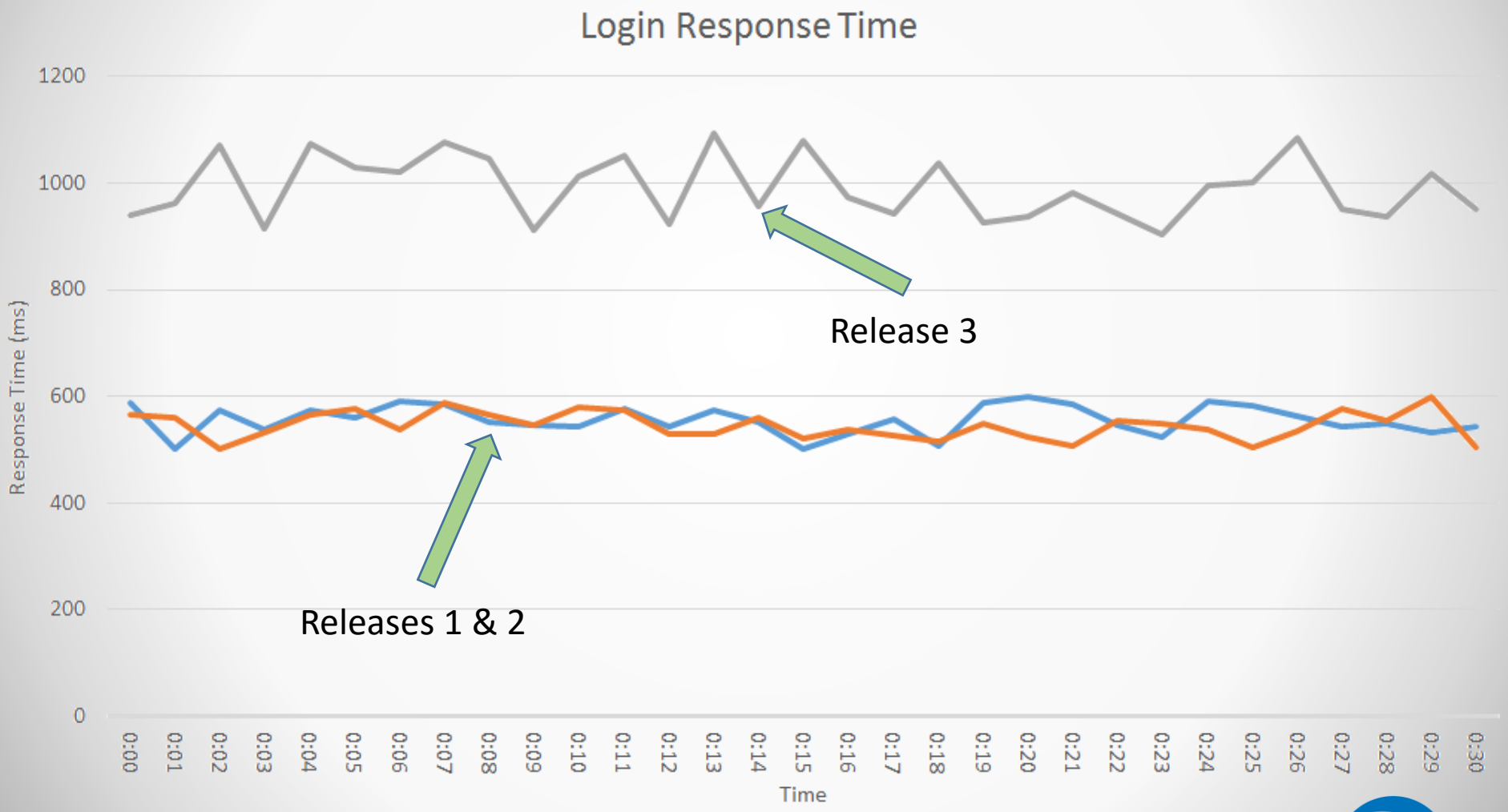
# Example

## RELEASE 3



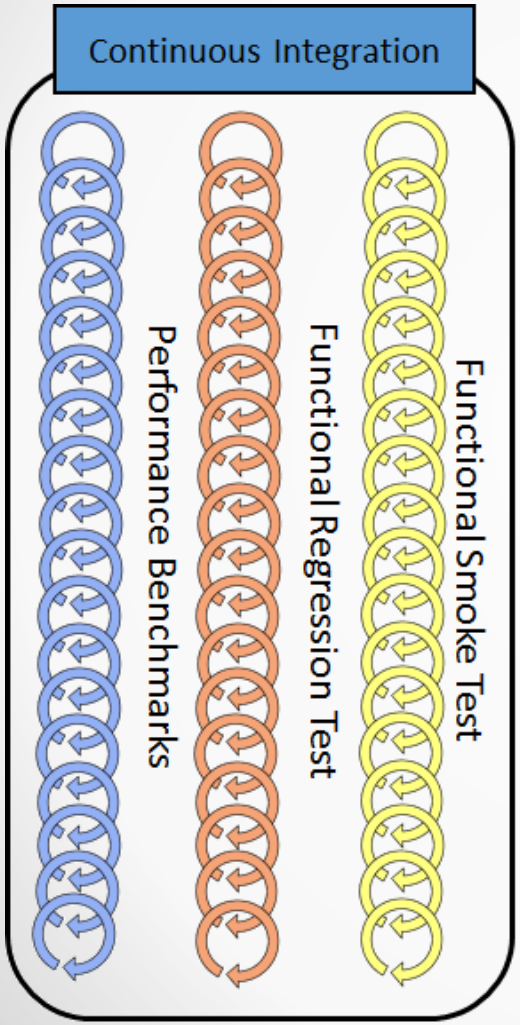

© Planit Software Testing

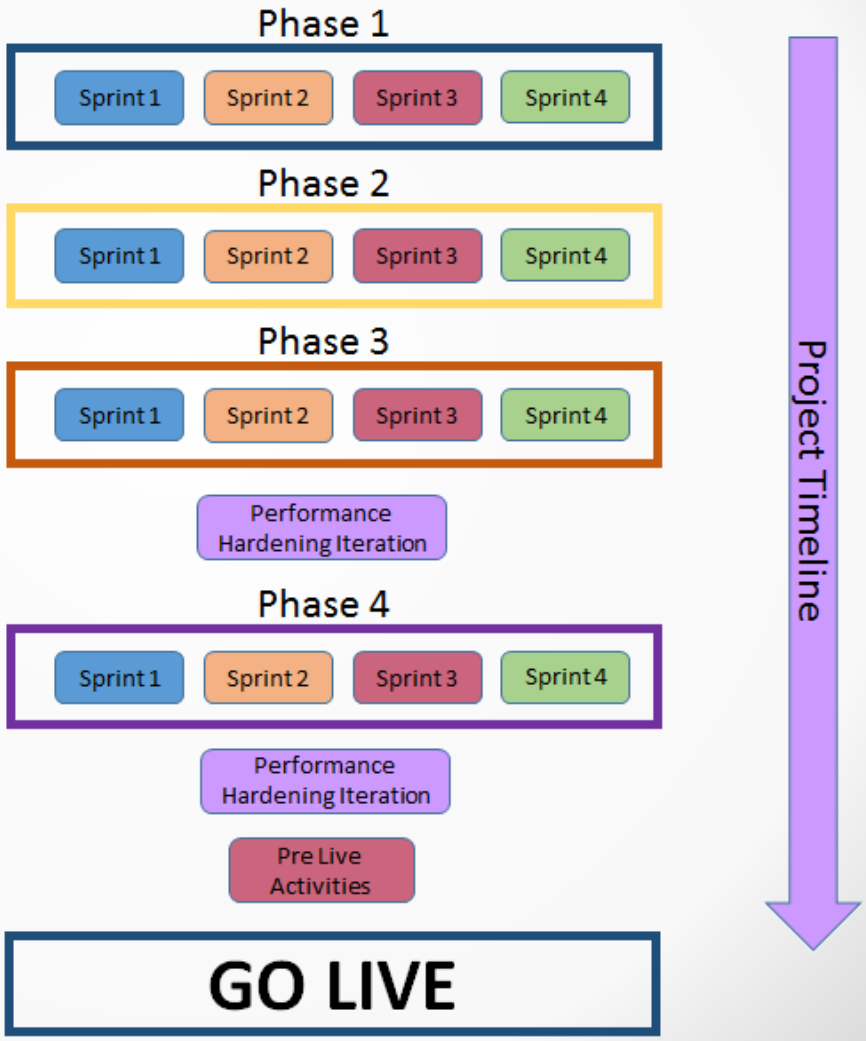
# Results




© Planit Software Testing

# Approach




© Planit Software Testing

# Mobile Performance Testing

# Different mobile applications

1. Native application (e.g. Facebook, Angry Birds)

2. Mobile Web application (e.g. Financial Times)

3. Hybrid application (e.g. Instagram mobile app)

© Planit Software Testing

Planit

# How do they work?

**Largely the same as a desktop application!**

1. User interacts with the application
2. Application may or may not send a request to the backend
3. Backend returns a response to the user, back to step 1.

© Planit Software Testing

# Why so critical?

1. Faster delivery to production
2. Continued customer satisfaction
3. Desire to be best option in the market

© Planit Software Testing

Planit

# What do we need to consider?

# Script Creation

Options:

1. Manually build each request
2. Record traffic and re-build requests
3. Record with performance tool through a proxy to create the requests

© Planit Software Testing

# Load Generation

What do you want to replicate?

Where can you generate the load from?

© Planit Software Testing

# Is there anything unique about apps?

- Data/CSS etc. often loaded during app start up
- Not as easy to ‘record and replay’
- User-Agent HTTP header is critical to simulate the correct device/browser

© Planit Software Testing

# Different Approaches

# What approaches are there?

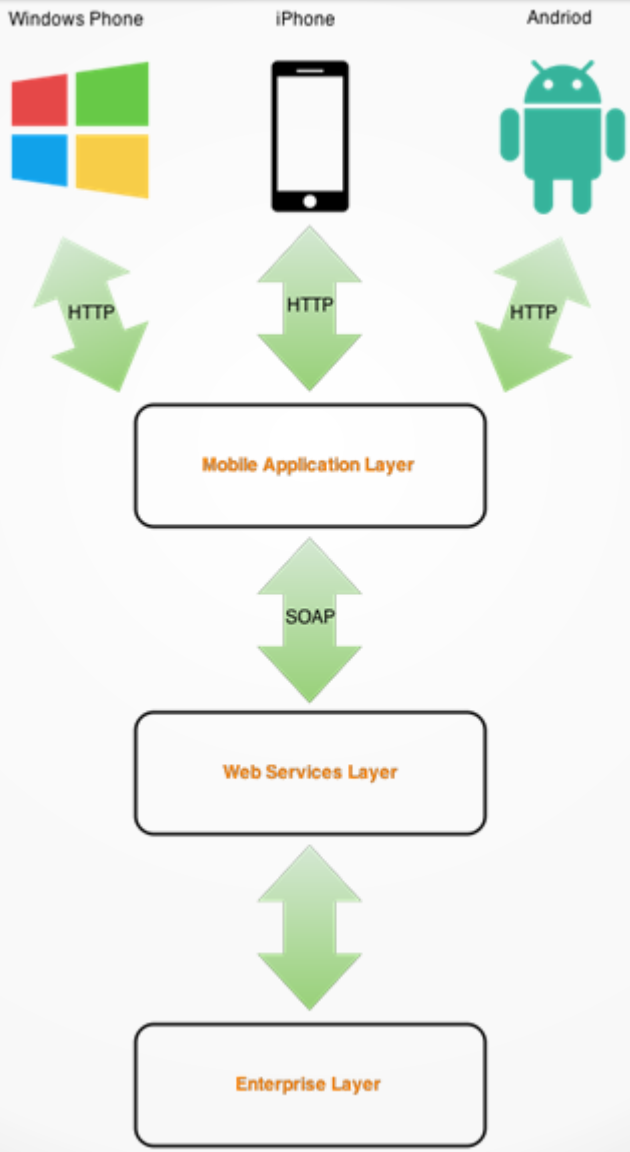
Three main approaches:

1. UI performance
2. Infrastructure performance
3. Combination of the first two approaches

© Planit Software Testing

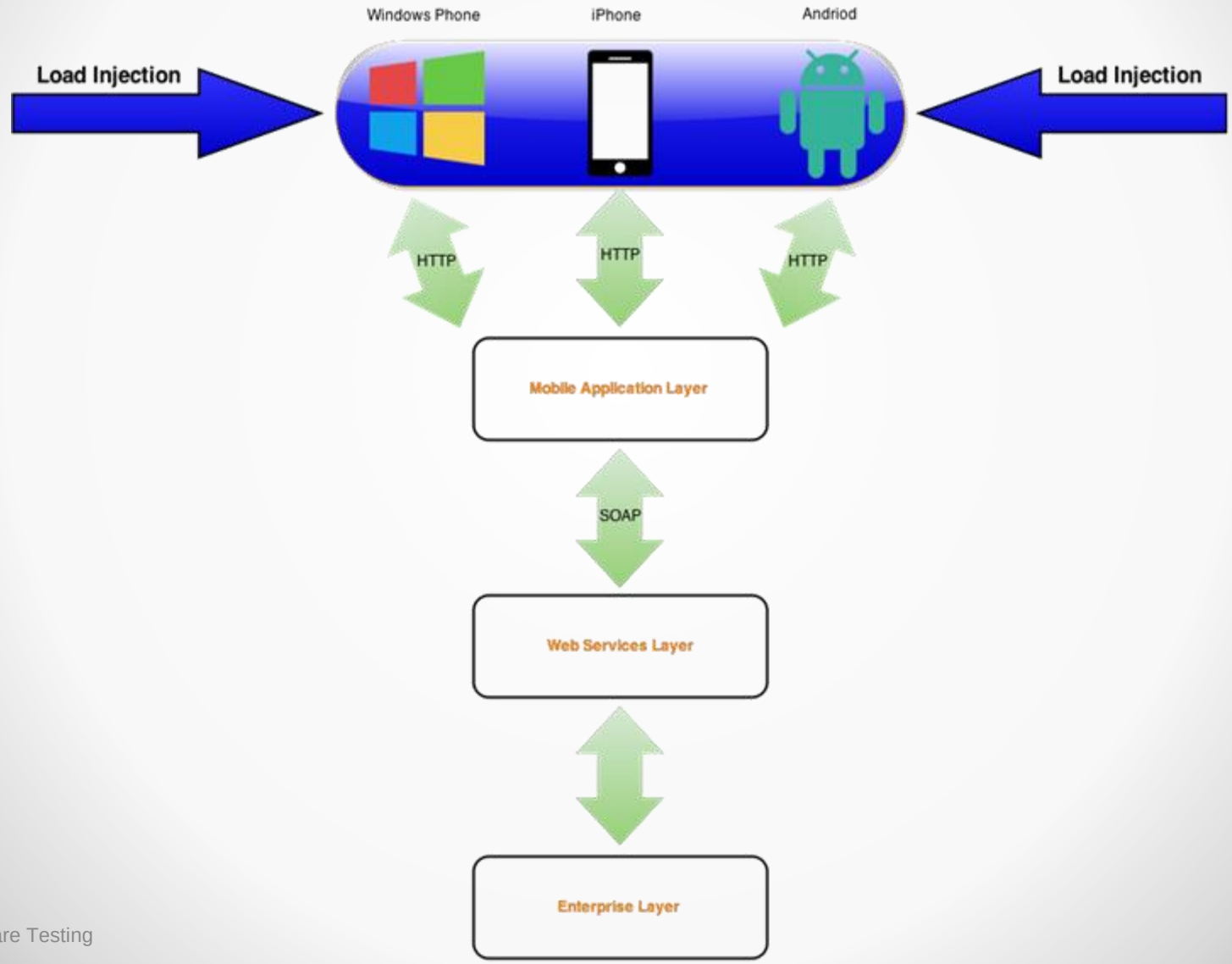
Planit

# Infrastructure



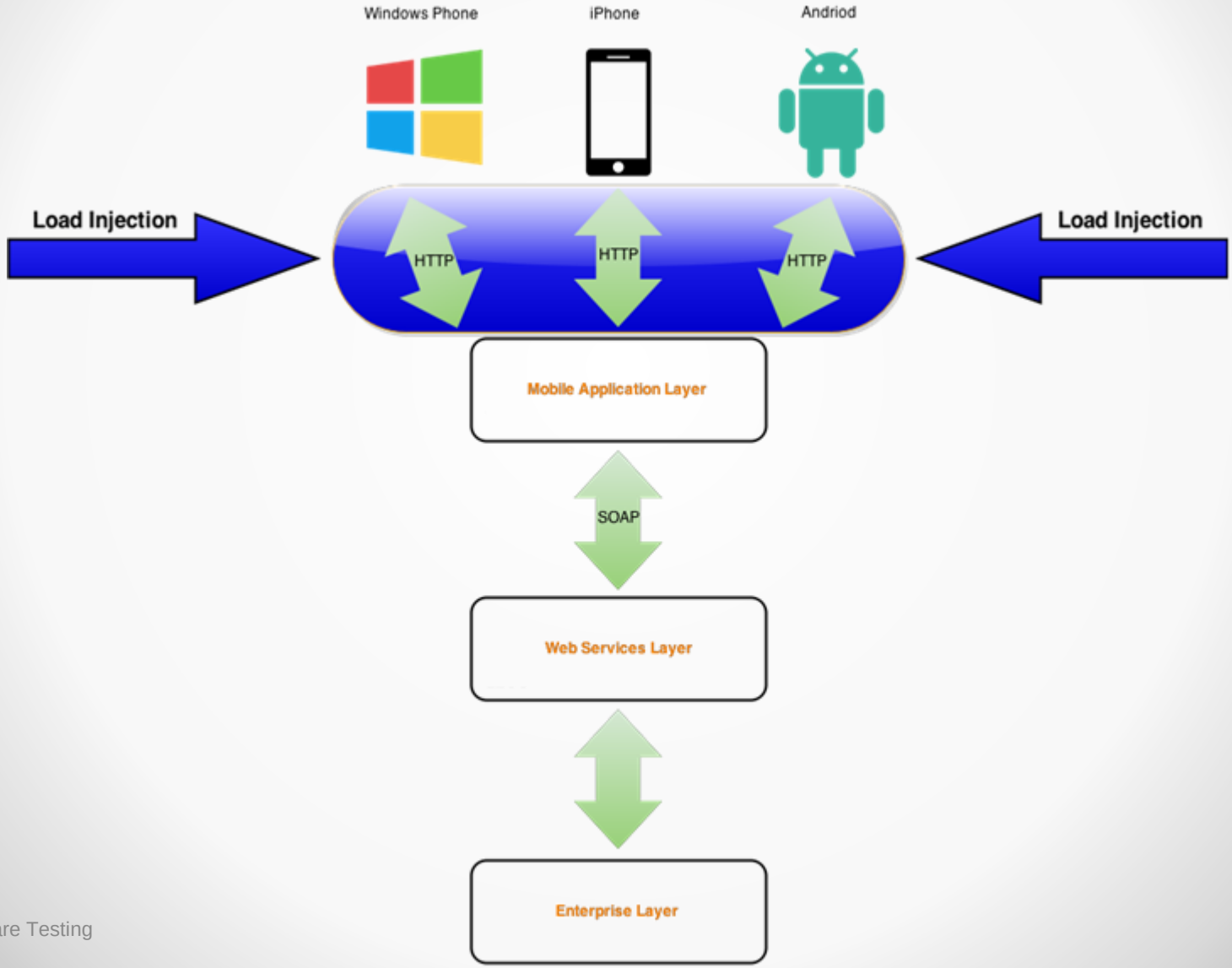

© Planit Software Testing

# UI Performance




© Planit Software Testing

# Infrastructure Performance




© Planit Software Testing

# Blended Approach

## Outcome

1. We have tested the infrastructure (main performance risk) under load
2. We have identified performance of the application on a mobile device while the infrastructure is under load

© Planit Software Testing

Planit

# The Future

# The Future

- Move to have everything ‘mobile’
- Agile approach with a traditional ending for mobile applications
- More mobile apps built agile
- Quicker turn-around

© Planit Software Testing

Planit

# The Future

- More in-house uptake
- Higher demand to be first to market
- More involvement earlier
- More integration

© Planit Software Testing

Planit

# The Future of Performance

# CHANGE

© Planit Software Testing

Planit

# Summary

The future is faster, more agile (in more ways than one) and constantly changing!

© Planit Software Testing

# Performance – Mobile – Agile – The Future

## Contact Planit

www.planittesting.com


**Jaron Marsh**

Senior Technical Test Consultant

jmarsh@planittesting.com

+64 4 815 8140


© Planit Software Testing

Planit