Performance Testing Mobile Devices

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. What are we testing
2. What do we need to consider
3. Mobile application case study
4. Different approaches
5. Lessons learned

© Planit Software Testing

Planit

Common current perception:

“If you haven’t performance tested a mobile app before, you wont know how to do it!”

© Planit Software Testing

Planit

What are we testing?

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

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

What do we need to consider?

What type of app are we testing?

Does it matter?

© Planit Software Testing

Planit

Script Creation

Three 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

Load Generation

Be aware of the impact with different options of load generation

© Planit Software Testing

Planit

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

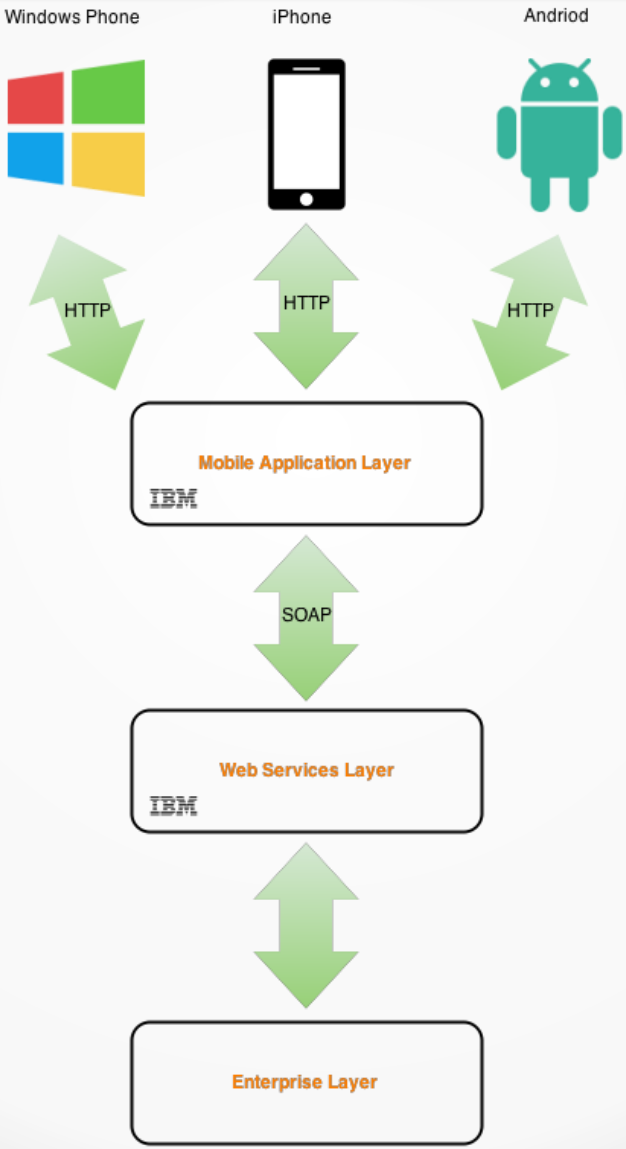
Mobile App Case Study

Project Overview

- Hybrid financial transaction app
- Used across multiple devices/platforms/networks
- Integrated to use an ESB
- All IBM components
- Large amount of backend requests with user interactions
- Financial and legal requirement for up to date information

© Planit Software Testing

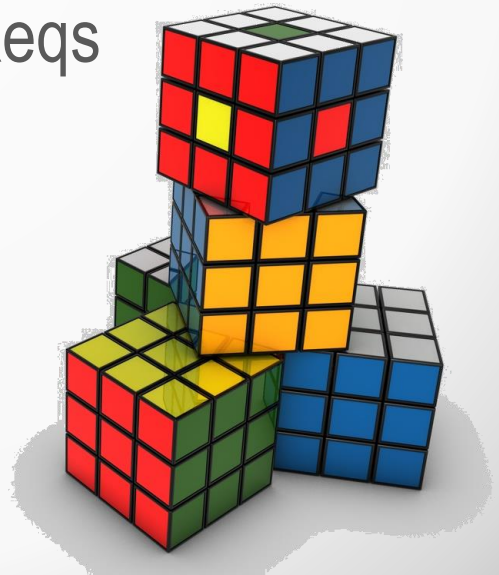
Infrastructure



© Planit Software Testing

Challenges

- Lack of end to end transaction knowledge
- Multiple vendors
- Lack of understanding on app usage
- Difficulty recording/building transactions
- Lack of sufficient Non Functional Reqs
- Changes of scope
- No enterprise monitoring solution



© Planit Software Testing

Planit

Approach

- Built transactions based off information provided by developers
- Transactions re-built after recording traffic
- Testing from the HTTP layer
- Load injected from as close to the test environment as possible
- SNMP monitoring across as many servers as possible

© Planit Software Testing

Issues

Similar issues identified as a normal performance testing engagement!

© 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

What approaches are there?

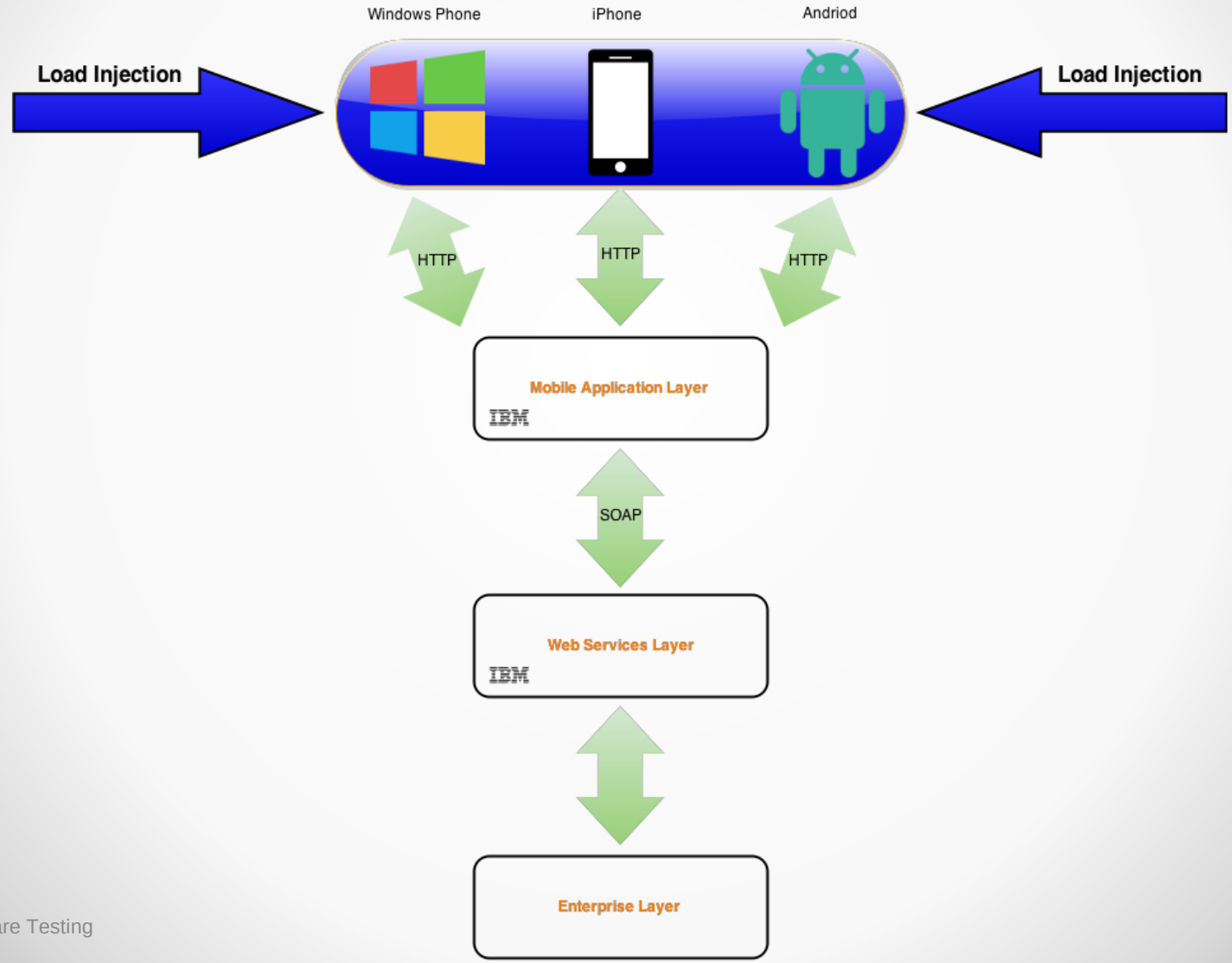
Three main approaches:

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

© Planit Software Testing

Planit

UI Performance



© Planit Software Testing

What approaches are there?

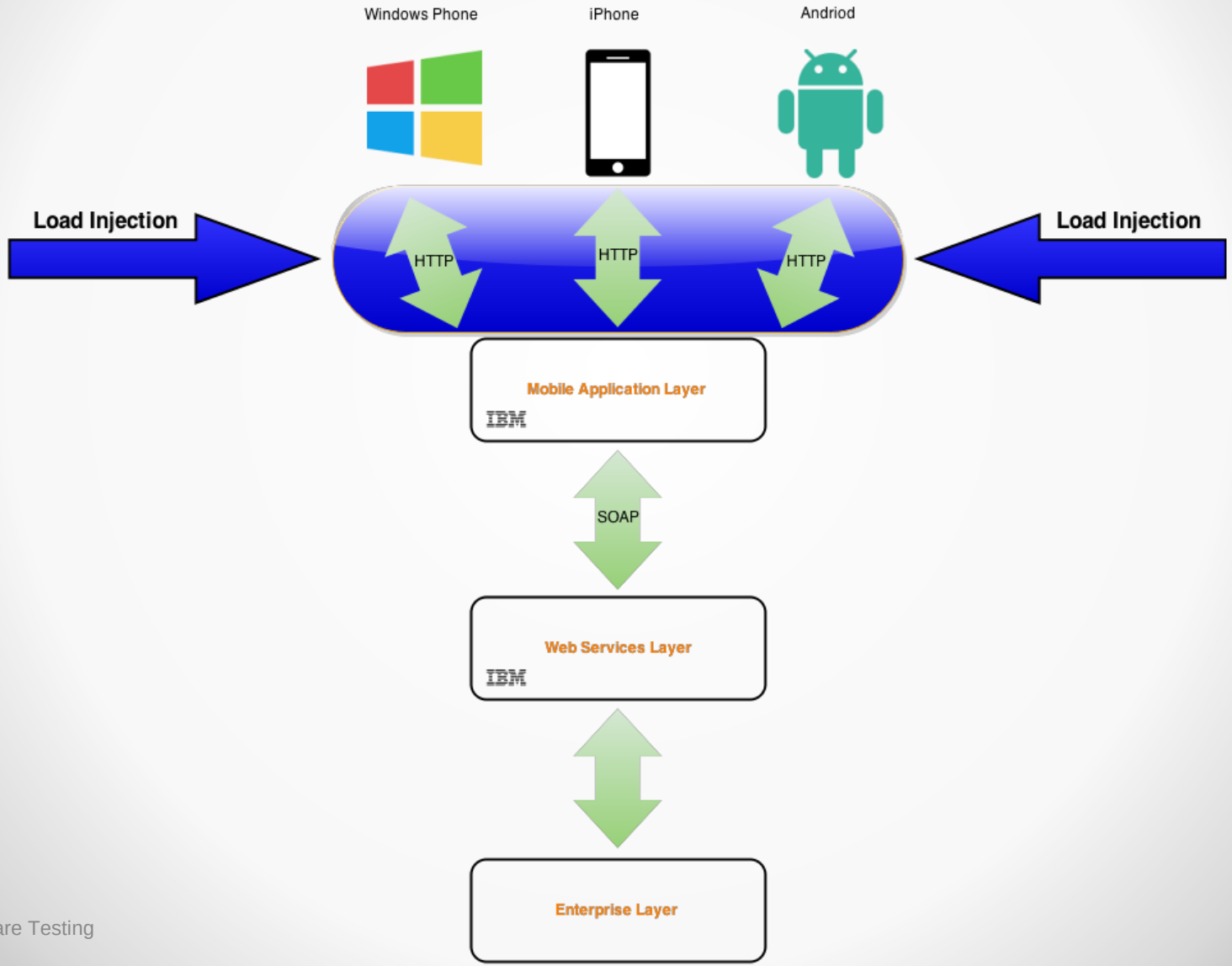
Three main approaches:

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

© Planit Software Testing

Planit

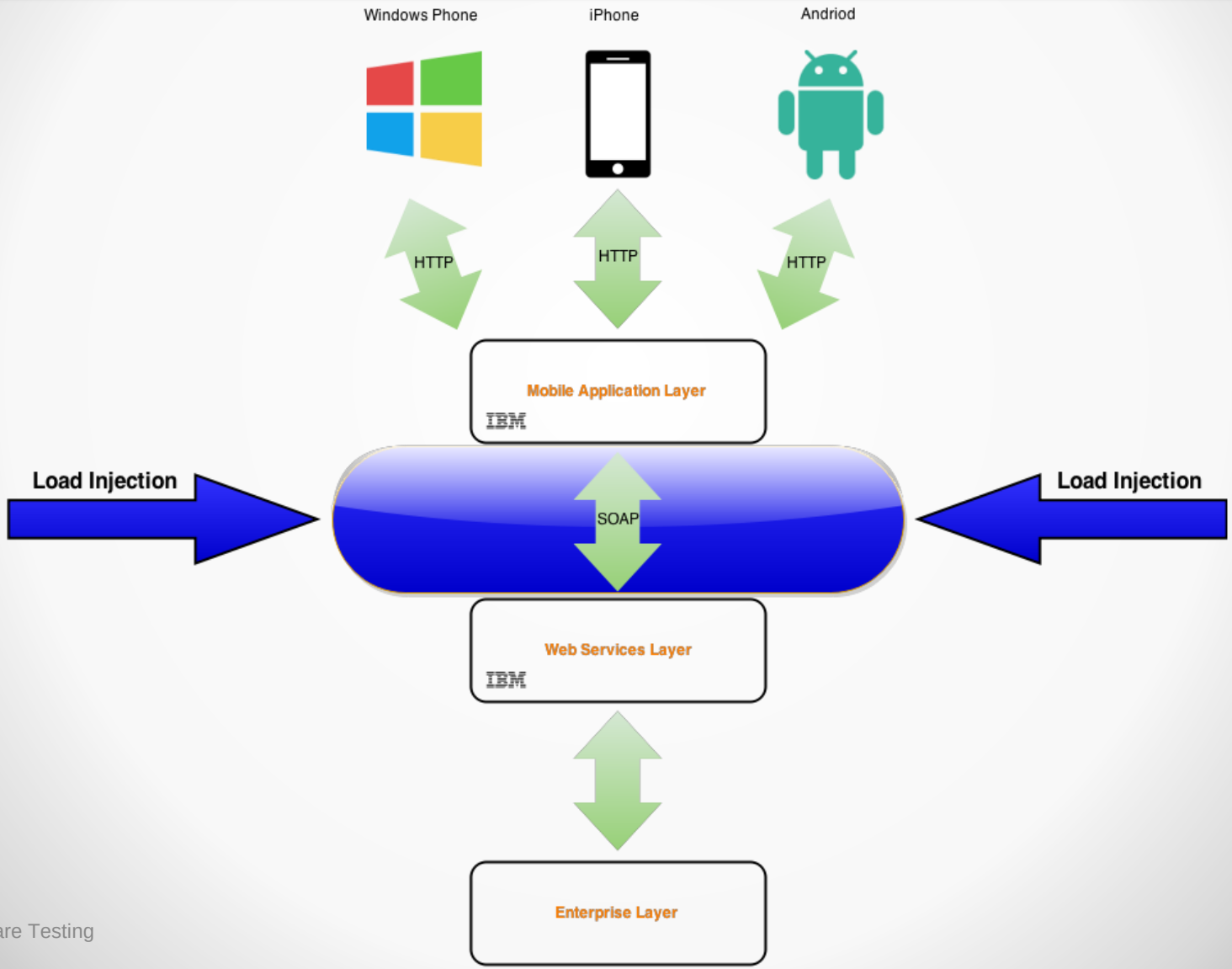
Infrastructure Performance



© Planit Software Testing

Planit

Infrastructure Performance



© Planit Software Testing

Planit

What approaches are there?

Three main approaches:

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

© Planit Software Testing

Planit

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

Lessons Learned

Lessons Learned

- More companies looking at mobile apps
- Higher turnaround getting to market
- Higher turnaround pushing out updates
- Can’t guarantee performance for every single user
- Similar approach to traditional performance testing
- Keep things light, no one wants an app that drains battery or mobile data

© Planit Software Testing

Planit

Tips

- Get in as early as possible (single user/request tests)
- Define exactly what is being tested as early as possible
- Keep an eye on message sizes while scripting
- Physically use the mobile device
- Education is often the key to success
- Monitor as much as possible

© Planit Software Testing

Summary

Performance testing mobile devices IS similar to traditional performance testing – with a few tweaks!

© Planit Software Testing

Plan it

Performance Testing Mobile Devices

Contact Planit

www.planittesting.co.nz

+64 4 815 8190

Jaron Marsh

Technical Test Consultant

jmarsh@planittesting.co.nz

+64 27 526 9793

© Planit Software Testing

Planit