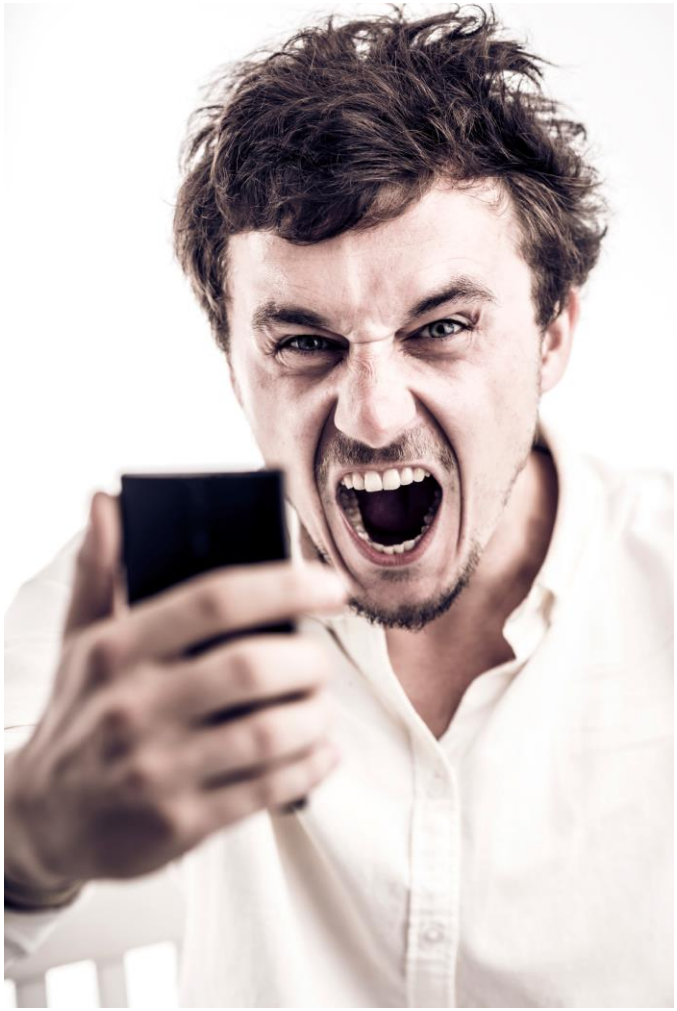
TestAWARE: A Laboratory-Oriented Testing Tool for Mobile Context-Aware Applications

Chu Luo, Miikka Kuutila, Simon Klakegg, Denzil Ferreira, Huber Flores, Jorge Goncalves, Mika Mäntylä and Vassilis Kostakos

chu.luo@unimelb.edu.au

Context-Awareness

Bugs in Mobile Apps



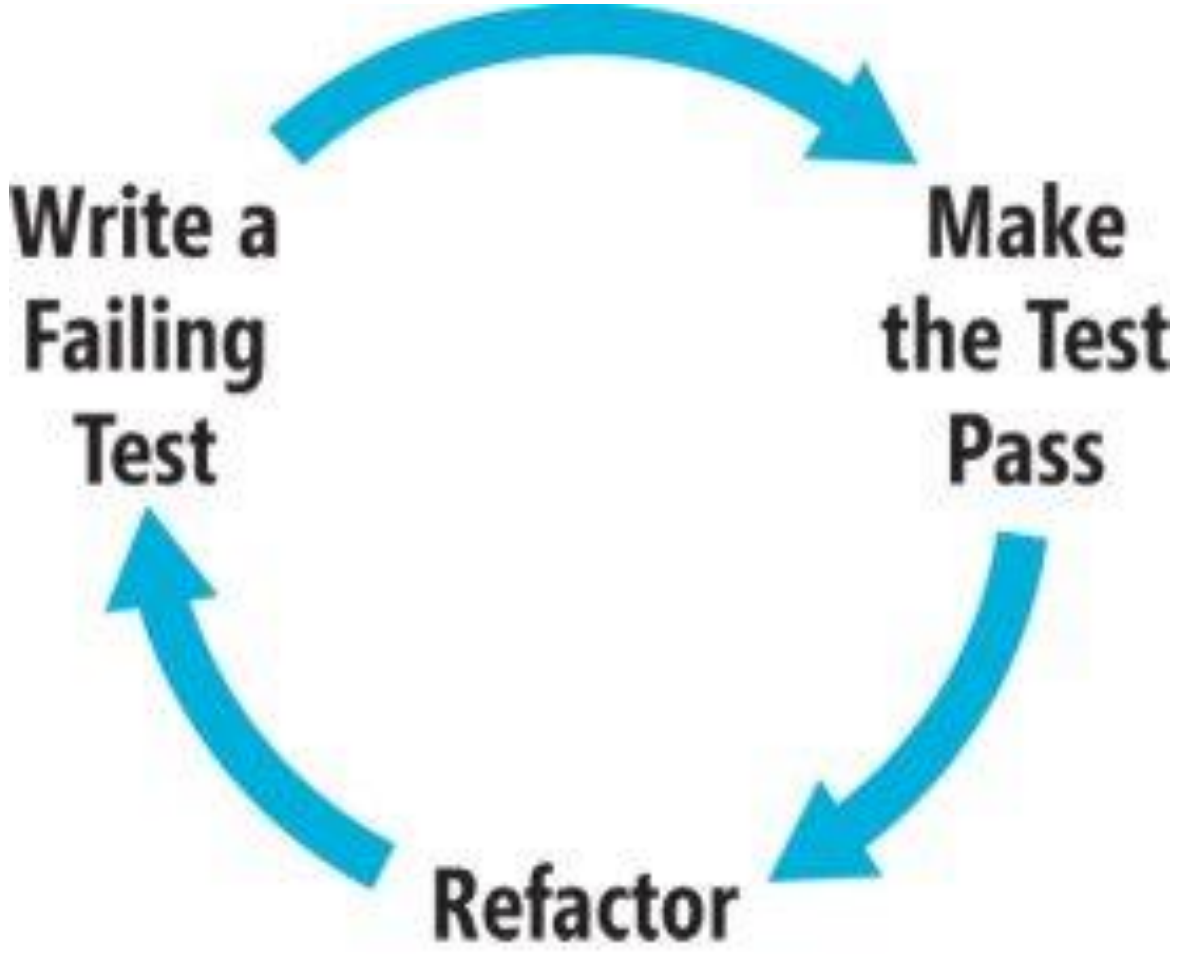
Testing and Maintenance: 50% budget

Testing Mobile Context-Aware Apps

More challenging!

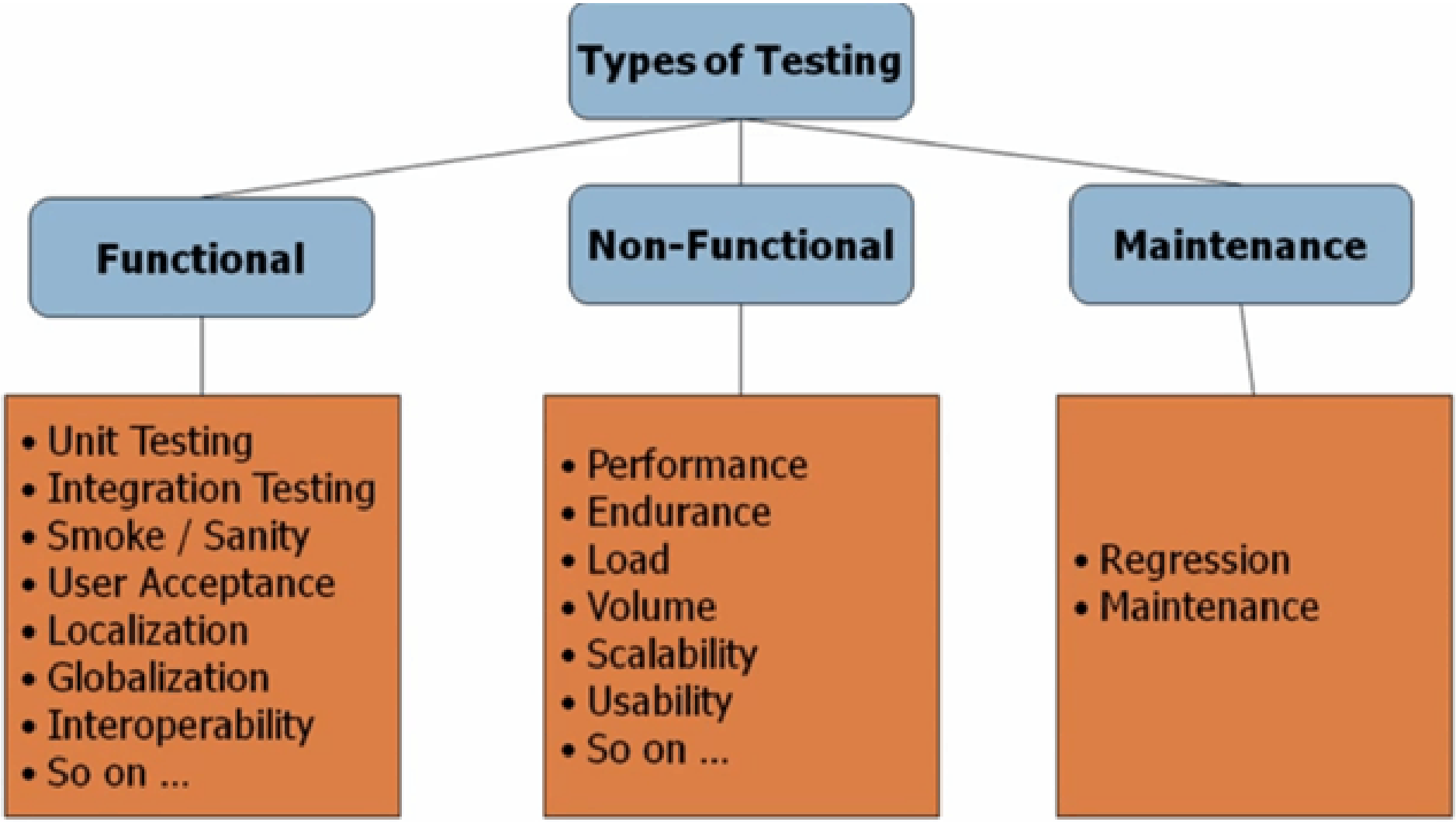
Mobile Bugs…

Testing a Car-Crash-Detection App


Write a Failing Test
Make the Test Pass
Refactor

Testing a Fall-Detection App

Testing Apps for Chronic Disease


Types of Testing
Functional
Non-Functional
Maintenance
• Unit Testing
• Integration Testing
• Smoke / Sanity
• User Acceptance
• Localization
• Globalization
• Interoperability
• So on ...
• Performance
• Endurance
• Load
• Volume
• Scalability
• Usability
• So on ...
• Regression
• Maintenance

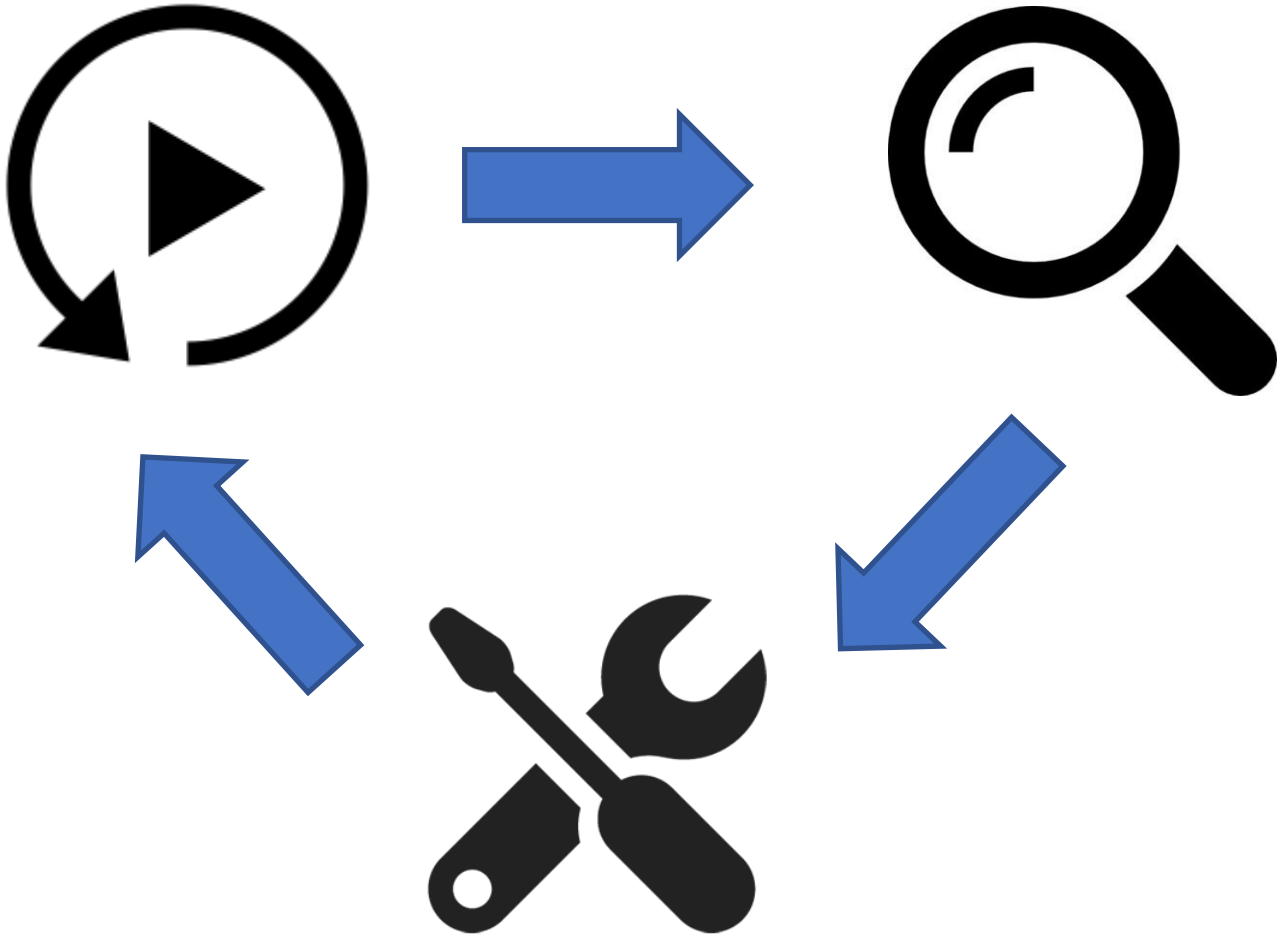
Testing Mobile Context-Aware Apps

Challenges:

1. **Acquisition of test data (test cases)**
2. **Longitudinal dataset**
3. **Diversity of testing tasks**

Laboratory Testing Needed

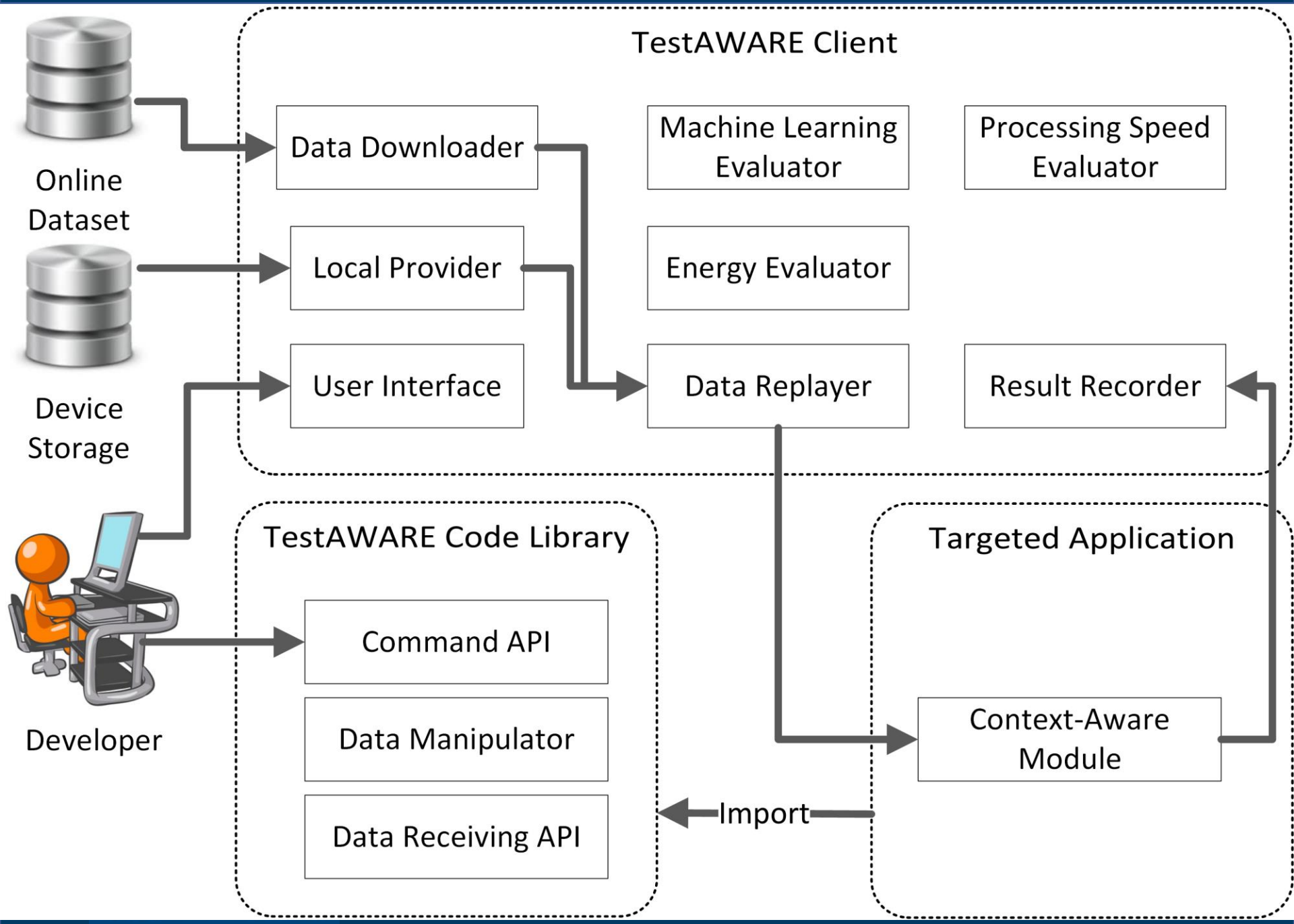
Laboratory Testing Loop



TestAWARE: A Laboratory-Oriented Testing Tool

Supporting:

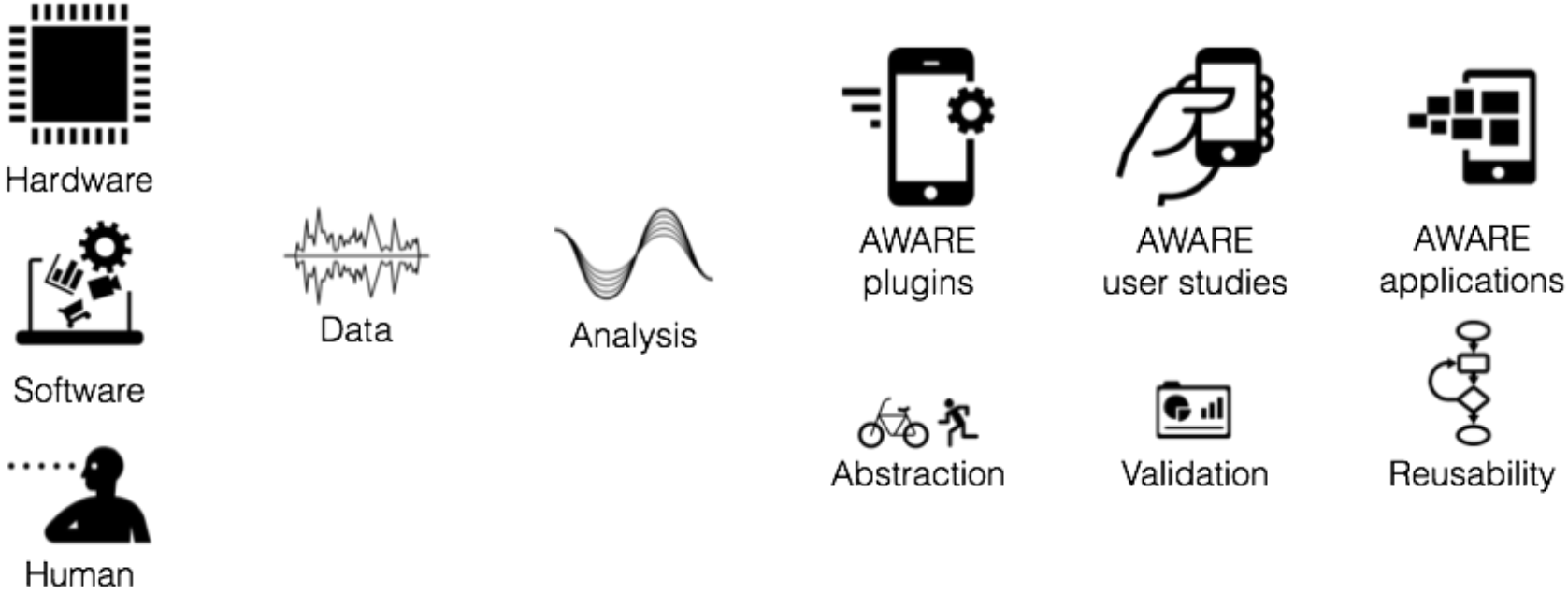
1. **Replay test data (many types/sources)**
2. **Replay speed control (faster/slower)**
3. **Black/White-box, functional/non-functional testing.**


TestAWARE Client
Online Dataset
Device Storage
Developer
Data Downloader
Local Provider
User Interface
Machine Learning Evaluator
Processing Speed Evaluator
Energy Evaluator
Data Replayer
Result Recorder
TestAWARE Code Library
Command API
Data Manipulator
Data Receiving API
Import
Targeted Application
Context-Aware Module

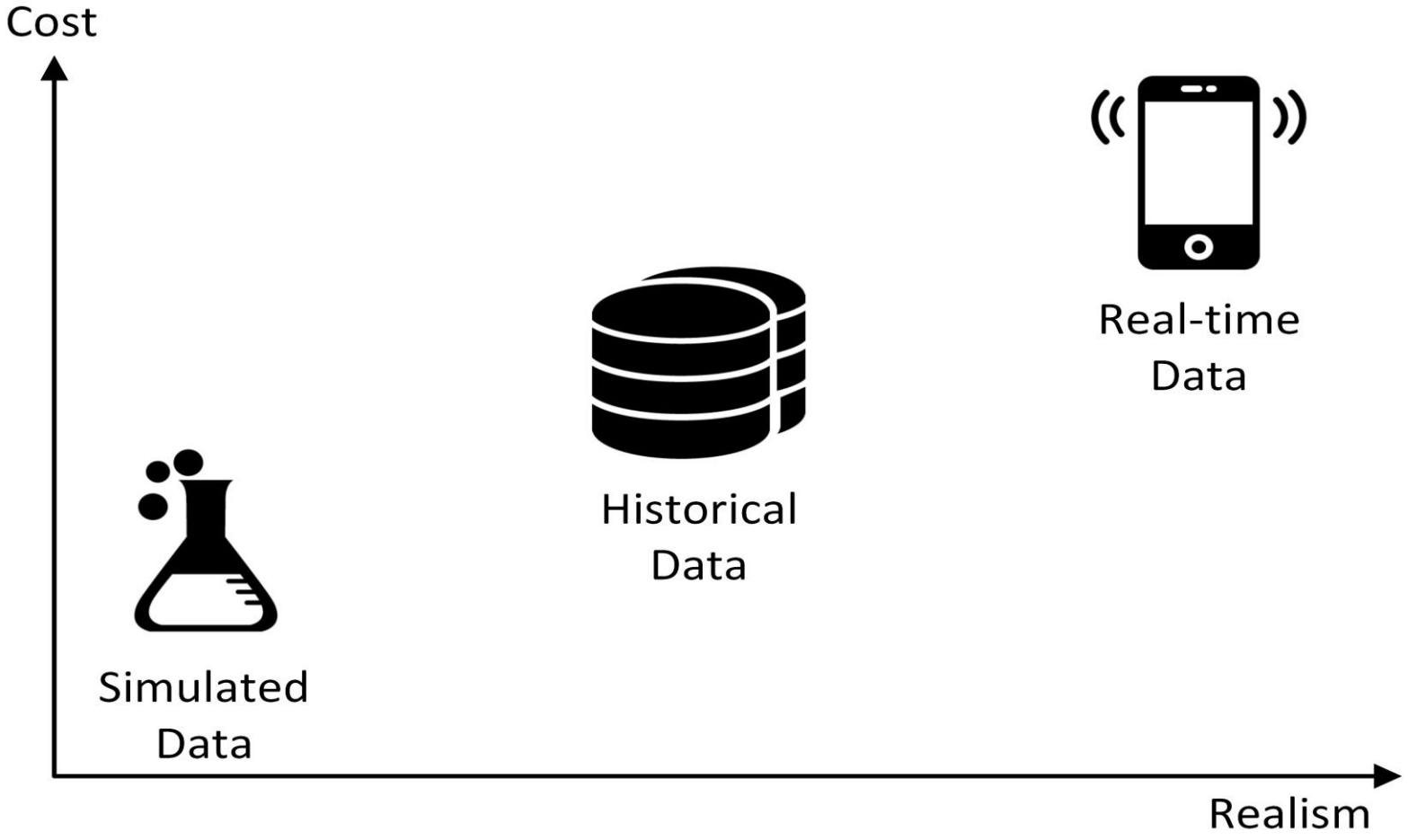
Data Types

AWARE Framework

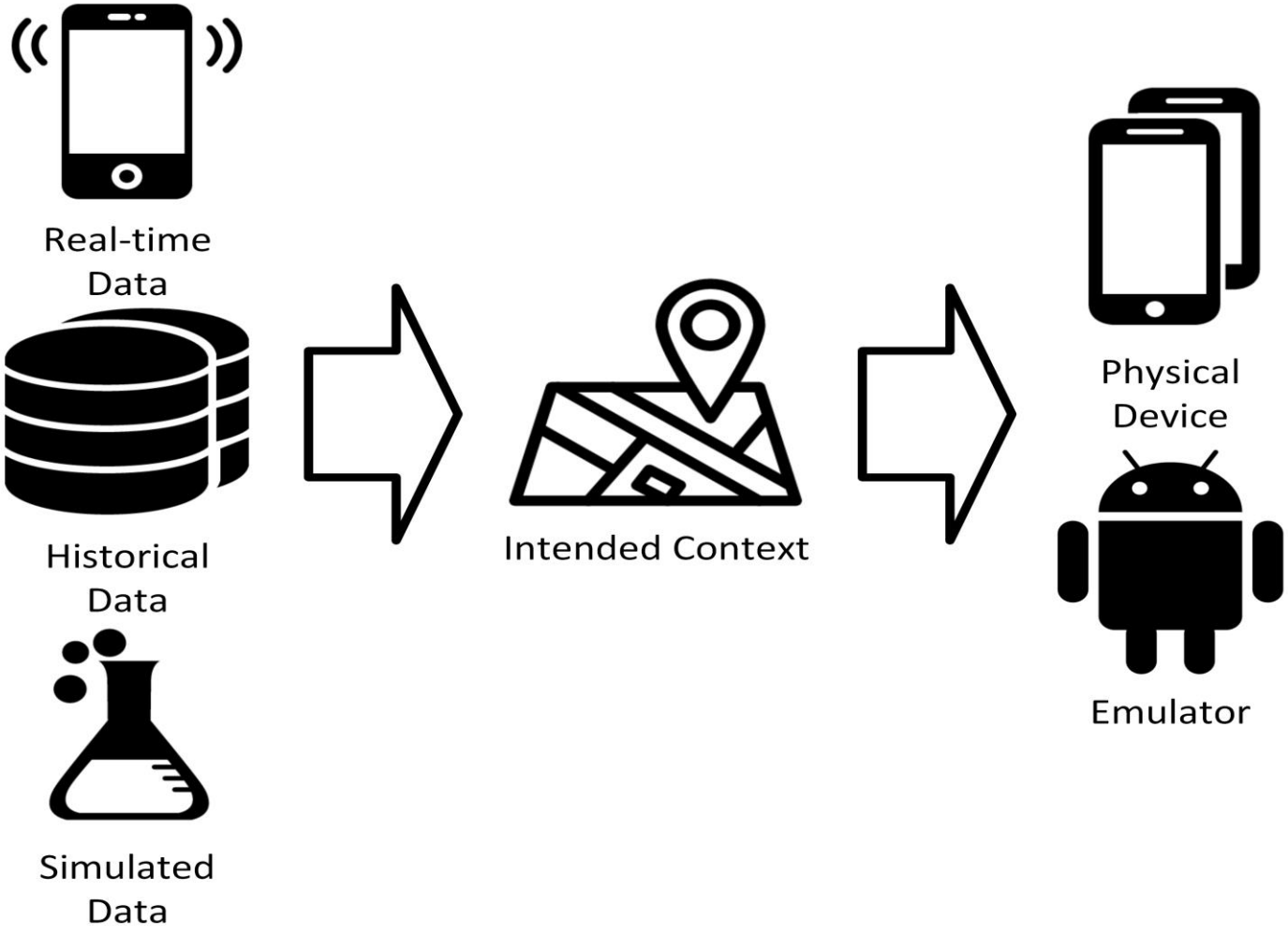
http://www.awareframework.com/



Data Sources



Replaying Data for Lab Test



Recording/Observing Results

Intended Context

App

Result Recorder

Speed Control of Replay

1. Fast replay for longitudinal datasets

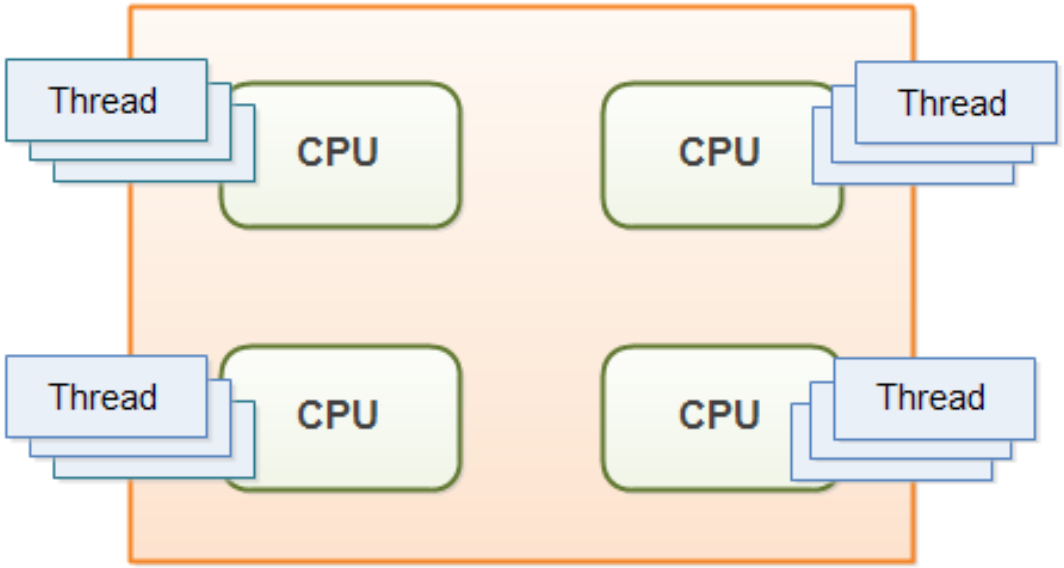
a) Using recalculated timestamps

Speed Control of Replay

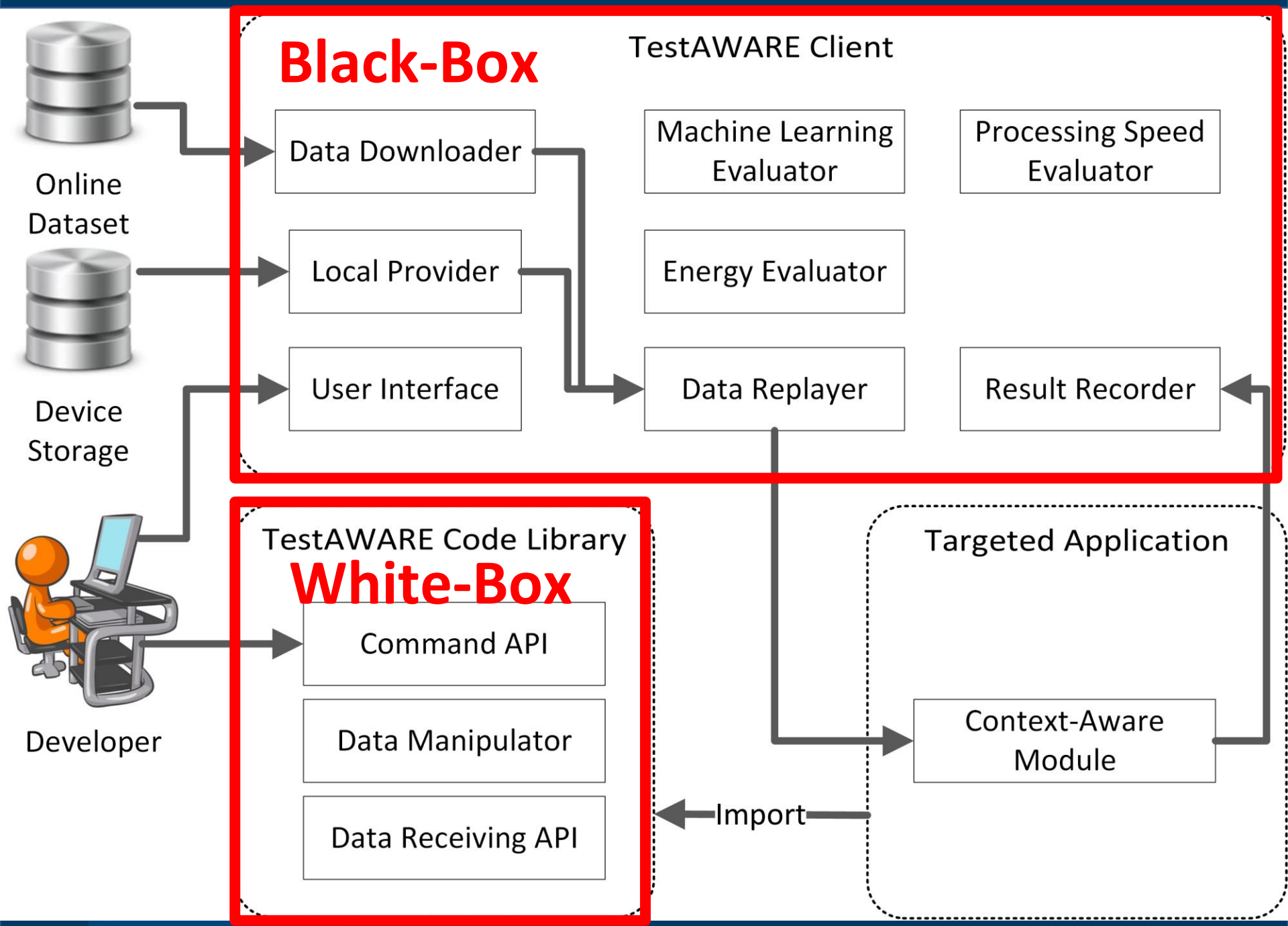
Also supports sensor + audio data

Optimisation of Replay

Concurrent data replay for multiple sensors using Java 8 concurrency package



Black + White Box Testing Support


Black-Box
TestAWARE Client
Data Downloader
Machine Learning Evaluator
Processing Speed Evaluator
Online Dataset
Local Provider
Energy Evaluator
Device Storage
User Interface
Data Replayer
Result Recorder
TestAWARE Code Library
White-Box
Targeted Application
Command API
Developer
Data Manipulator
Context-Aware Module
Import
Data Receiving API

Assertion

Task:
1+1

App

Output

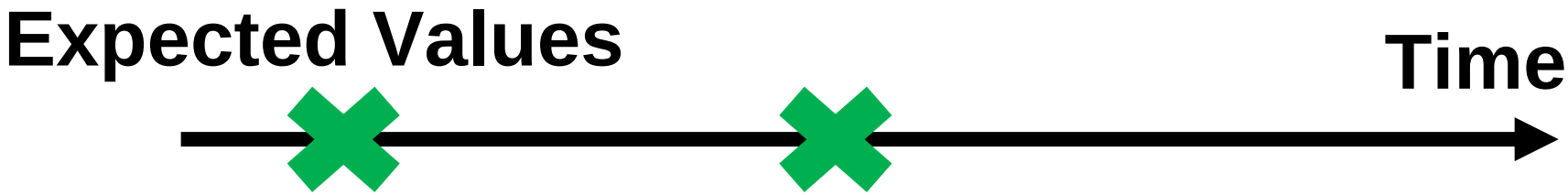
Compare

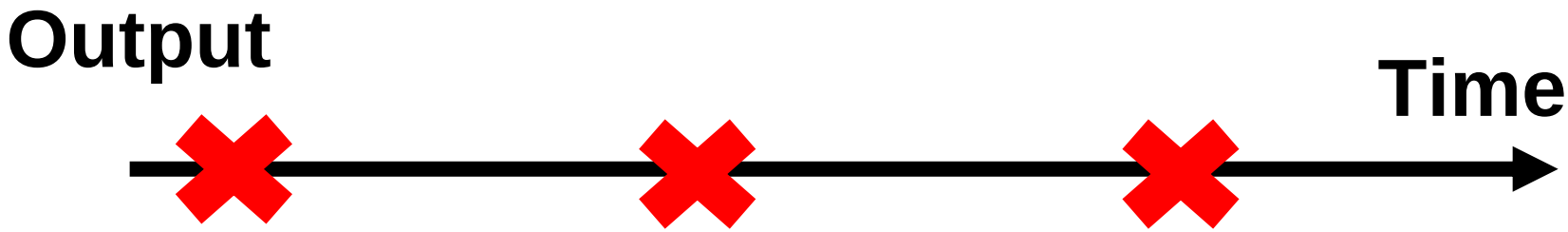
Expected
Value:
2

Non-functional Testing

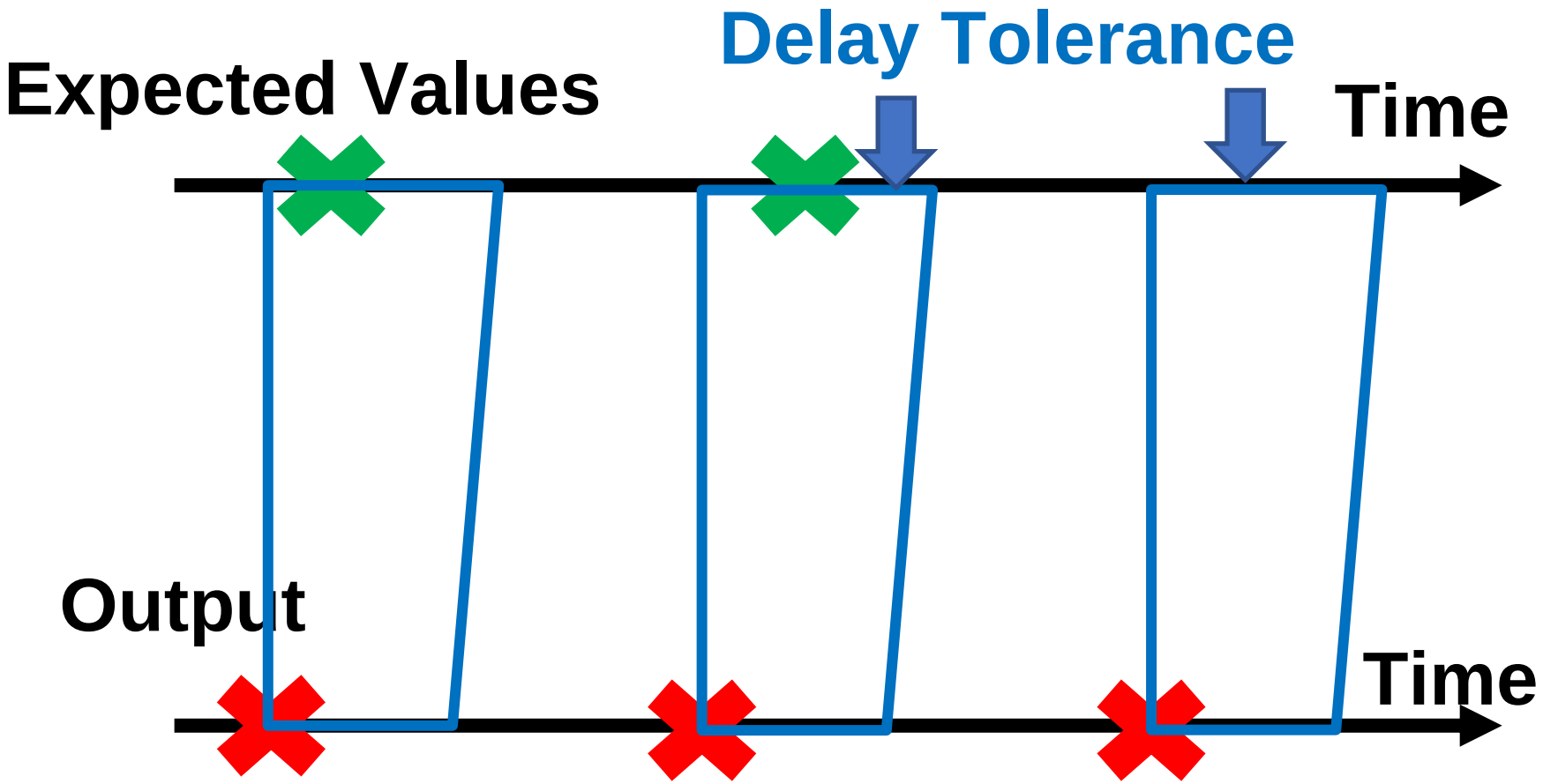
1. Machine Learning Accuracy (Classification/Regression)
2. Power Consumption Estimation
3. Processing Speed Measurement

Expected Values and Output of Machine Learning



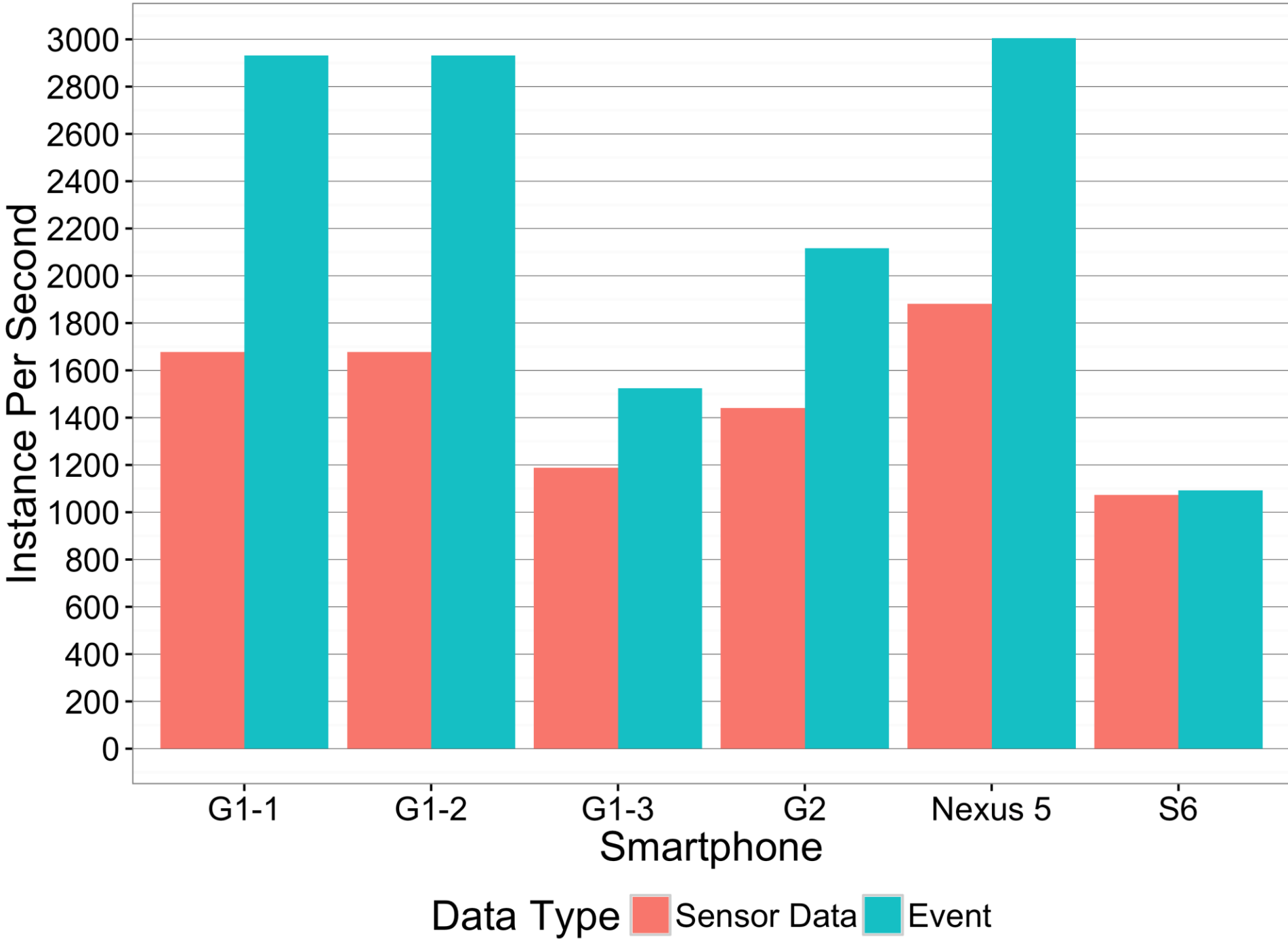


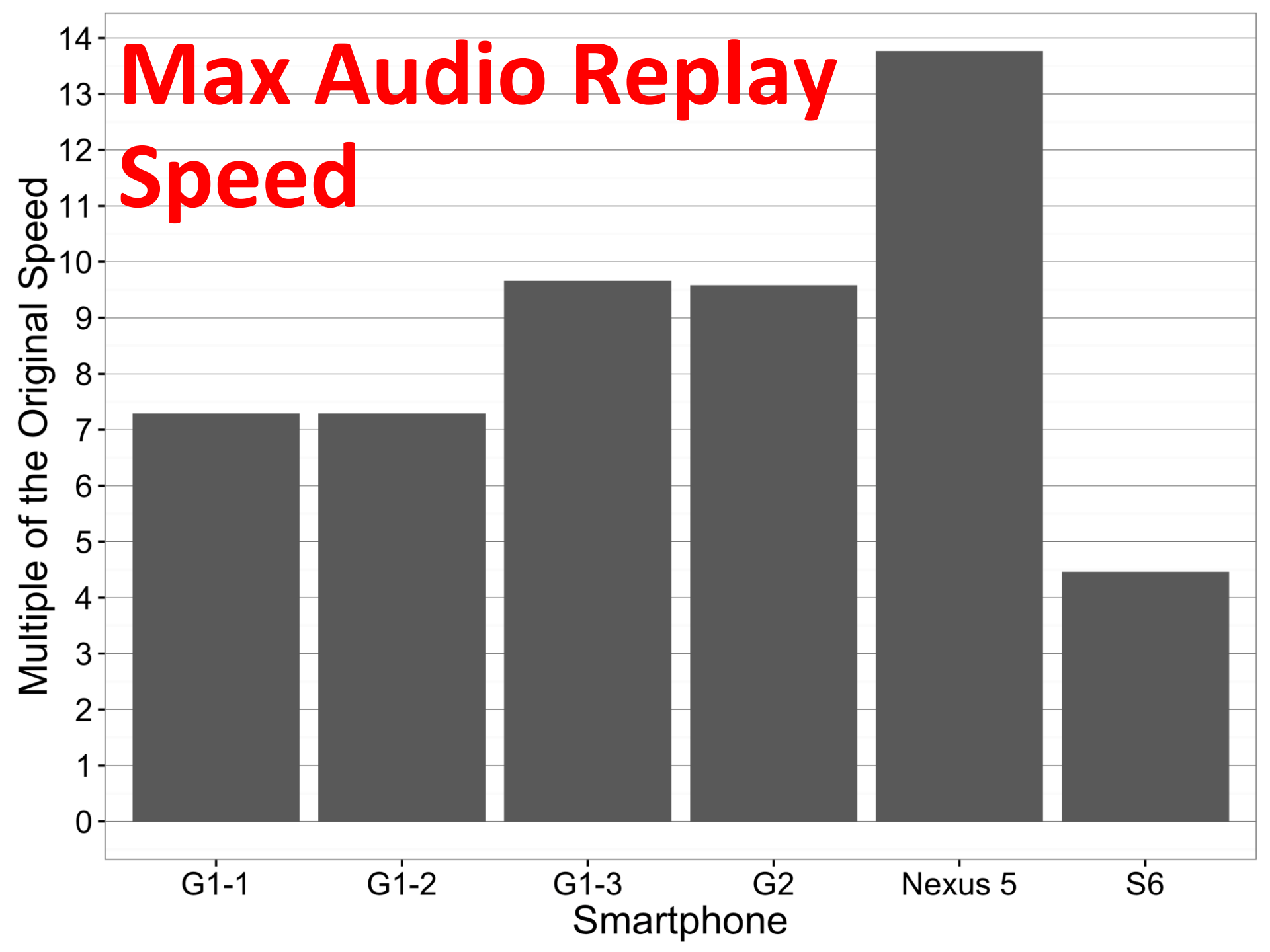
Expected Values and Output of Machine Learning

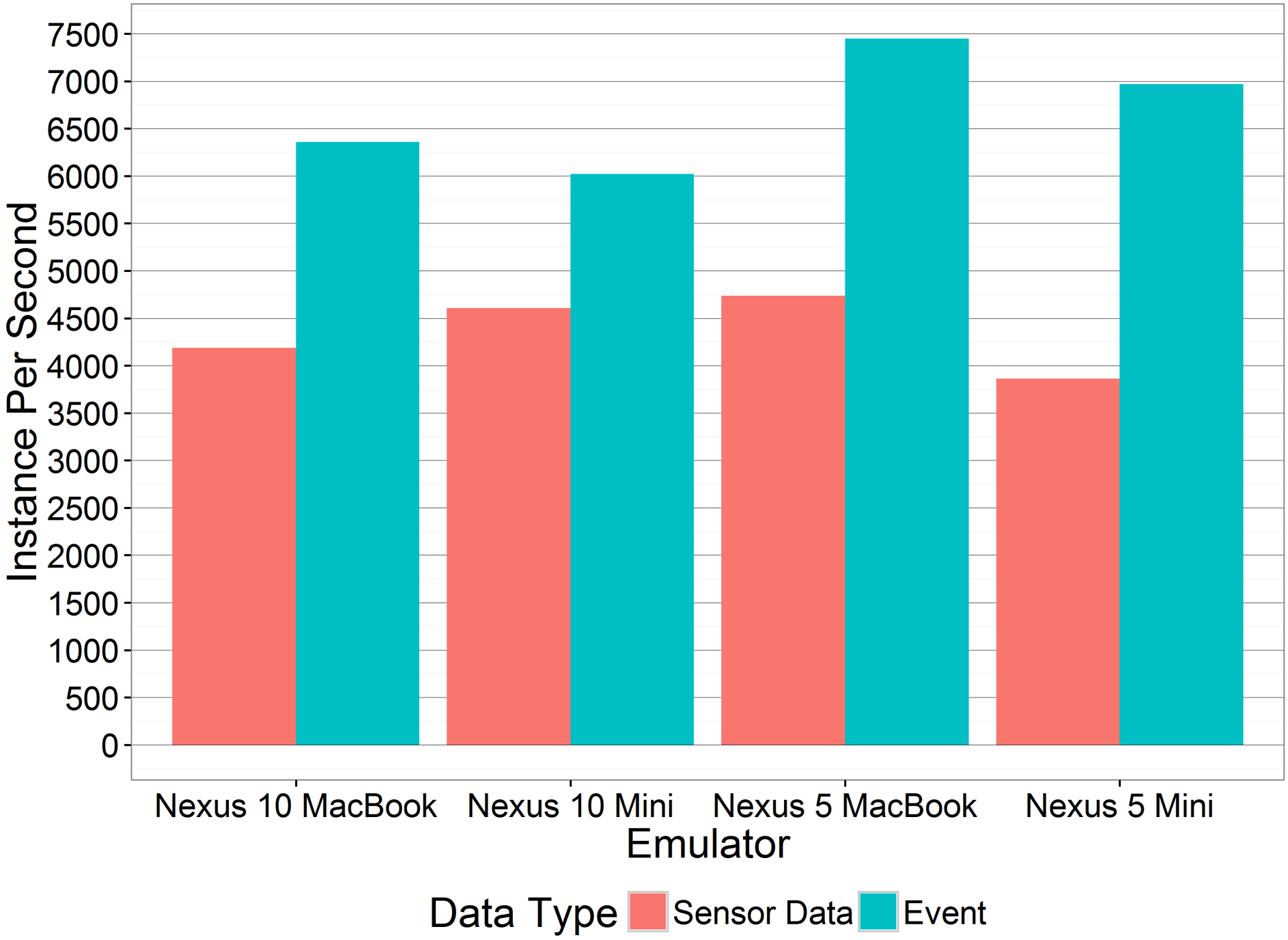


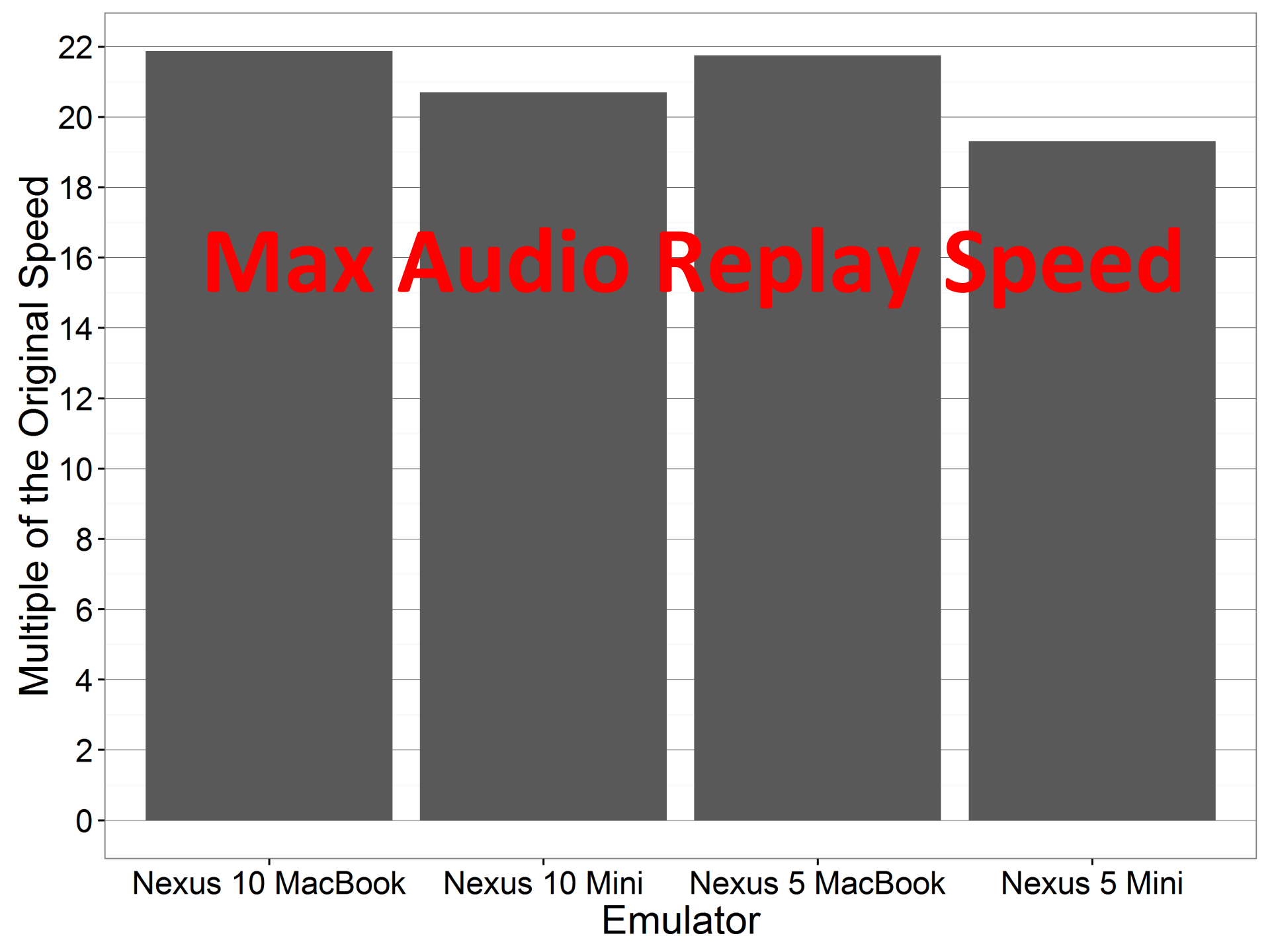
Evaluation: Maximal Replay Speed

1. Device: 6 phones, 6 tablets, 2 PCs
2. Data: Sensor readings, OS events, raw audio data
3. On a single thread


Instance Per Second
3000
2800
2600
2400
2200
2000
1800
1600
1400
1200
1000
800
600
400
200
0
G1-1
G1-2
G1-3
G2
Nexus 5
S6
Smartphone
Data Type
Sensor Data
Event


Max Audio Replay Speed
Multiple of the Original Speed
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
G1-1
G1-2
G1-3
G2
Nexus 5
S6
Smartphone


Instance Per Second
7500
7000
6500
6000
5500
5000
4500
4000
3500
3000
2500
2000
1500
1000
500
0
Nexus 10 MacBook
Nexus 10 Mini
Nexus 5 MacBook
Nexus 5 Mini
Emulator
Data Type
Sensor Data
Event


Max Audio Replay Speed
Multiple of the Original Speed
0
2
4
6
8
10
12
14
16
18
20
22
Nexus 10 MacBook
Nexus 10 Mini
Nexus 5 MacBook
Nexus 5 Mini
Emulator

Evaluation: User Study

1. Participants: 13 pro developers
2. Tasks with a buggy real-world app :
   a) Black Box Testing
   b) Functional/Non-Functional White Box Testing

User Study Findings

1. A wrong prediction may not be a bug
2. White Box Testing is very effective to find and locate a bug
3. The replay feature reproduces bugs well

Limitation and Future Work

1. The fragmentation problem for replay
2. Design and implementation for iOS
3. Testing tools for smartwatch and wearable computers

Take-away Points

1. Laboratory-oriented testing with data replay is necessary
2. We must match expected values and output
3. Replaying data on PC emulators is significantly faster

TestAWARE: A Laboratory-Oriented Testing Tool for Mobile Context-Aware Applications

Chu Luo, Miikka Kuutila, Simon Klakegg, Denzil Ferreira, Huber Flores, Jorge Goncalves, Mika Mäntylä and Vassilis Kostakos

chu.luo@unimelb.edu.au