IT University of Copenhagen

Evaluating Bluetooth Low Energy for IoT

Jonathan Fürst*, Kaifei Chen†, Hyung-Sin Kim†, Philippe Bonnet*
IT University of Copenhagen*, UC Berkeley†

CPSBench, Porto
April 10, 2018

jonf@itu.dk

Why bother with BLE?

IoT applications rely on BLE for **local**, **energy- efficient** data exchange between **smartphones** and **resource constrained peripherals**.

iBeacon based localization

Smart wristbands/watches

Environmental sensors and actuators

- Most **smartphones** are **equipped** with BLE
- Great **API support** on **Android** and **iOS**.

So job done?!

→ Sets BLE apart from other low power wireless technologies such as **ZigBee** or **Thread**.

Example: Smart Light Actuation Time

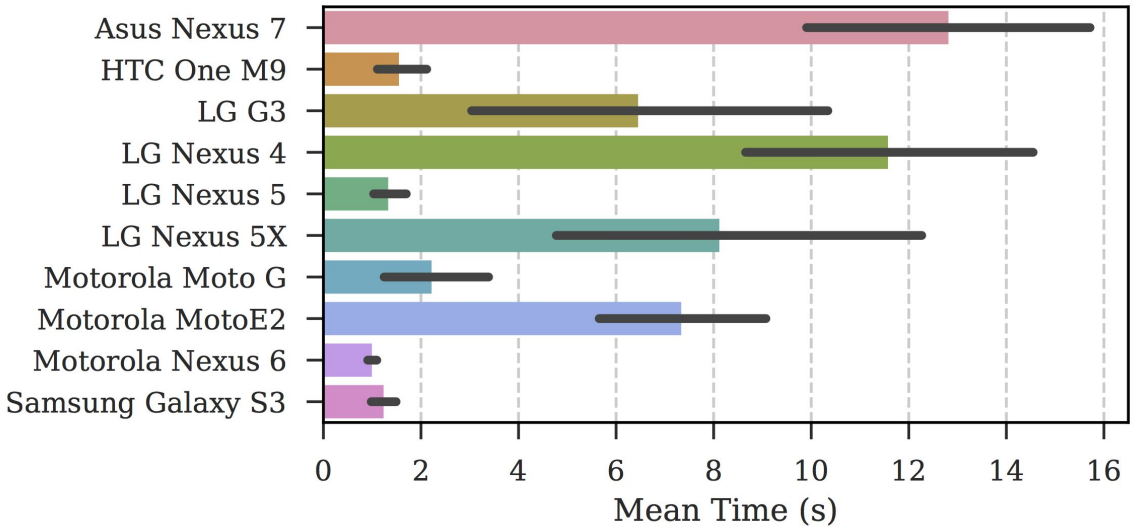
iBeacon based localization

Environmental sensors and actuators

→ Key Requirement: **Small latency of advertisements.**

Example: Smart Light Actuation Time

BLE Advertisement Latency. Advertising interval is set to 1280 ms while smartphones scan in the default balanced mode. Smartphones are placed in 2 m distance. We perform 20 repetitions.

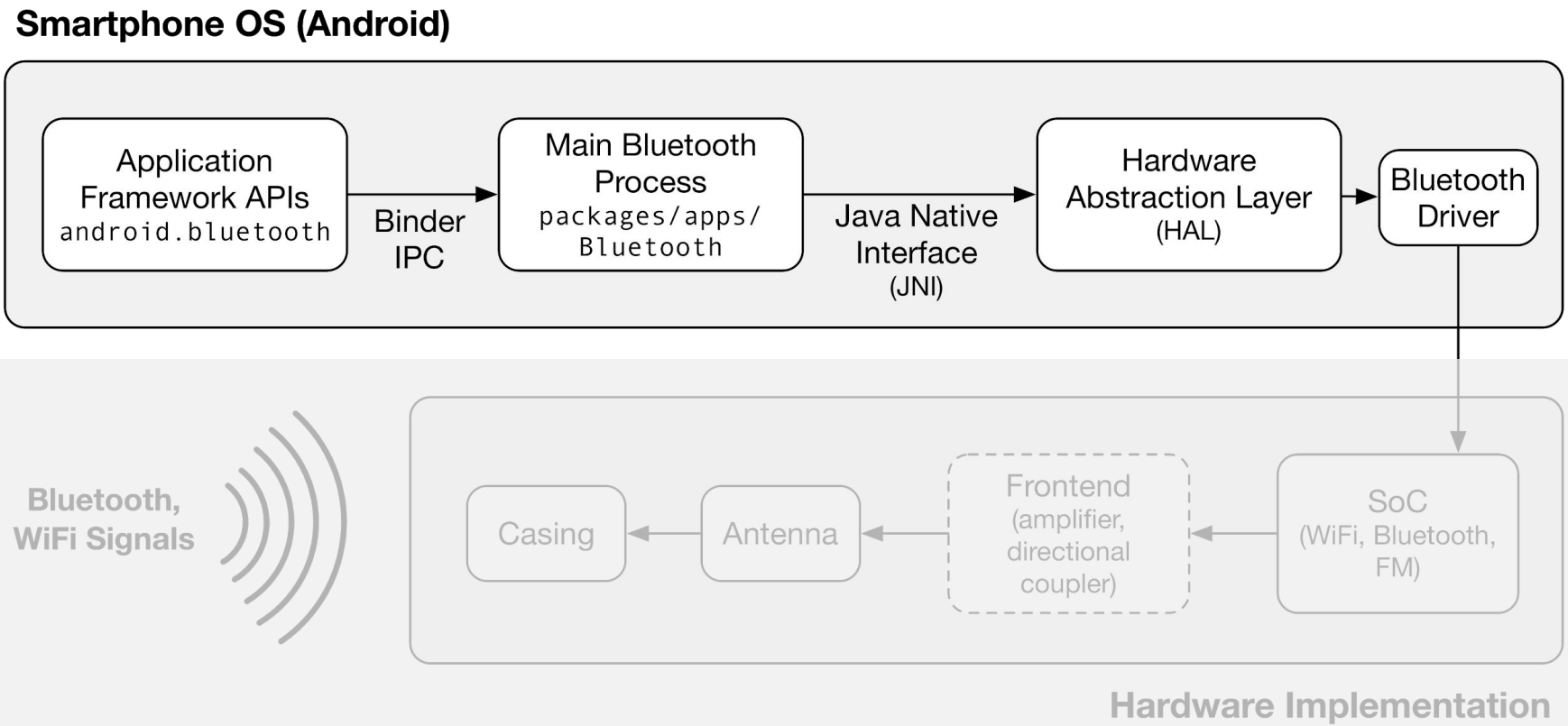


Outline

1. Introduction
2. BLE Hardware and Software Abstractions
3. Opening the Black Box
4. Conclusion and Open Questions

BLE Hardware and Software Abstractions

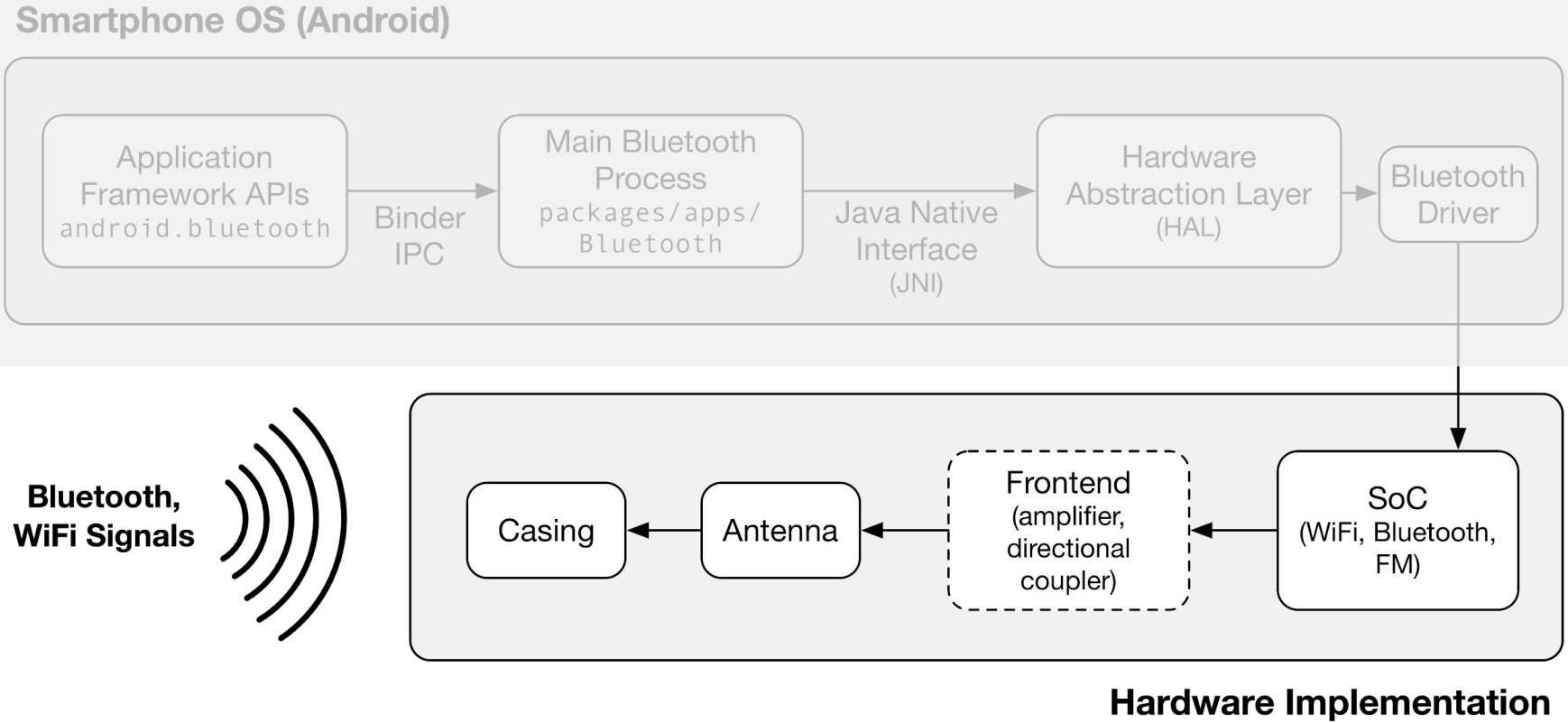
Tracing BLE down the **Software** and **Hardware** Abstractions.



Hypotheses

Hypothesis (1): BLE is accessed ‘indirectly’, through several (partly hidden) abstraction layers and proprietary drivers, which leads to unpredictable BLE behavior across smartphone models with different OS versions.

Tracing BLE down the **Software** and **Hardware** Abstractions.



Hypotheses

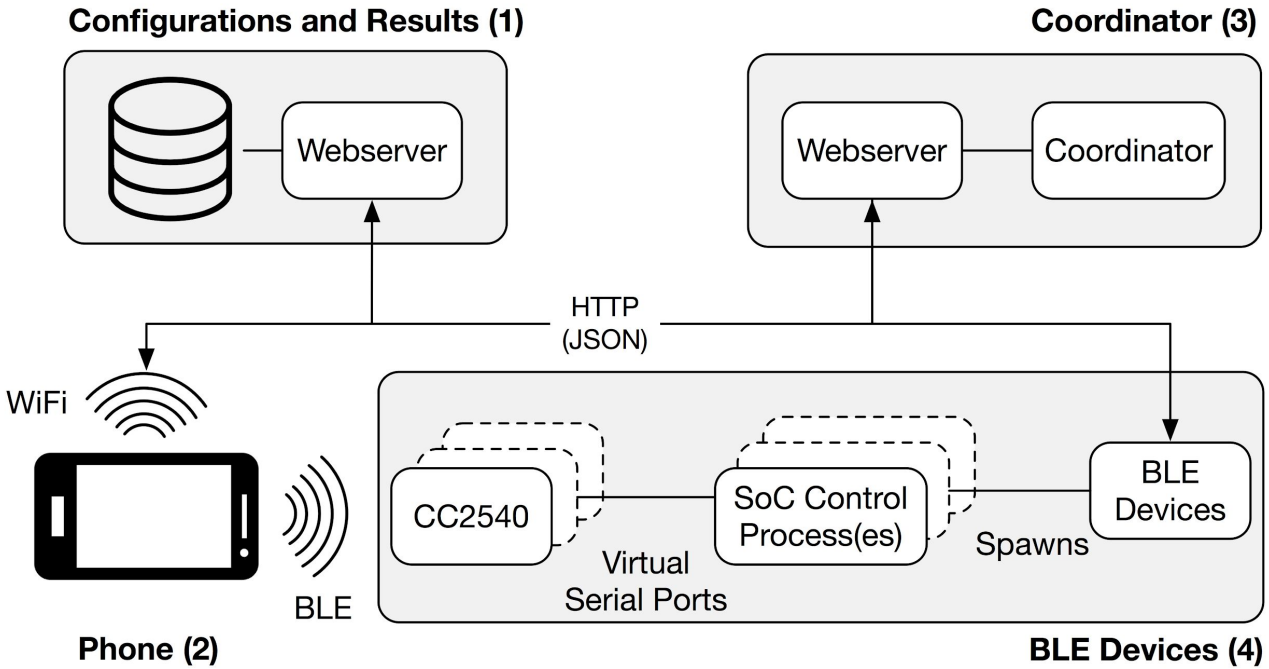
Hypothesis (1): BLE is accessed ‘indirectly’, through several (partly hidden) abstraction layers and proprietary drivers, which leads to unpredictable BLE behavior across smartphone models with different OS versions.

Hypothesis (2): SoC and BLE chip implementations are two major factors that lead to different BLE performance.

Hypothesis (3): Hardware components, such as amplifier, antenna, and cover, impact BLE performance.

Opening the Black Box

BLEva Overview

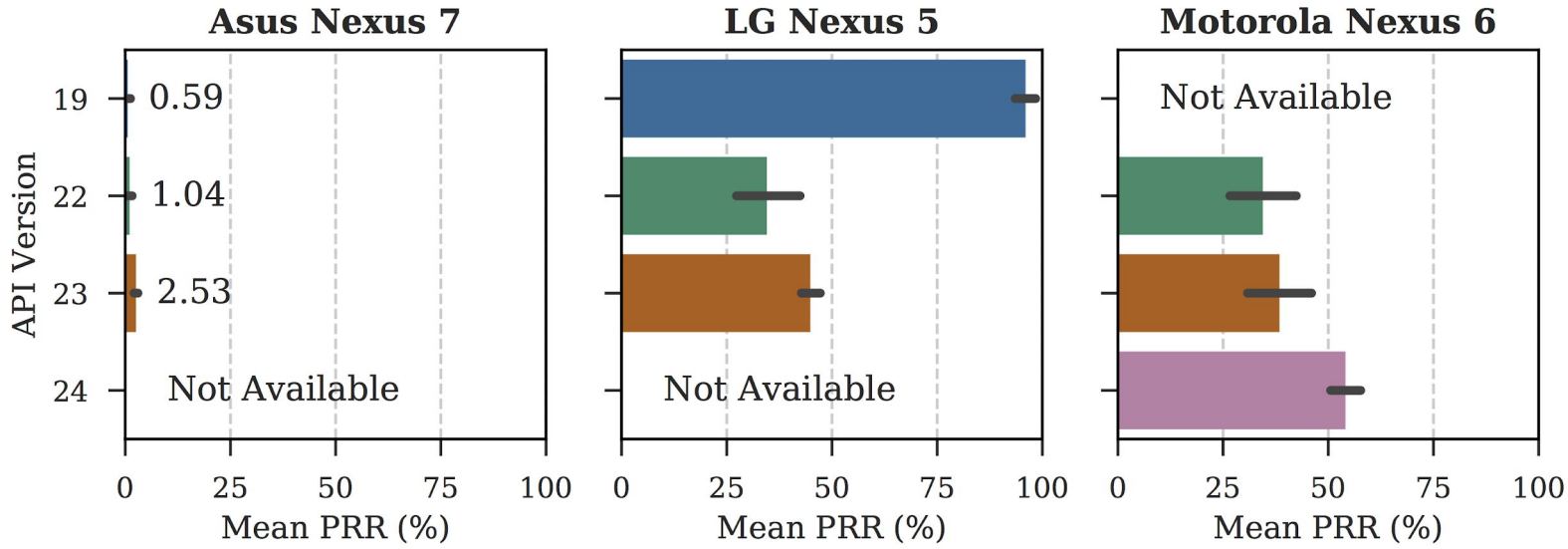


Experimental Setup

BLE Chip	Phone Models and API Versions
WCN3620	Motorola Moto G (API 22), Moto E2 (API 23)
WCN3660	LG Nexus 4 (API 22), Asus Nexus 7 (2013) (API 23)
BCM4339	LG Nexus 5 (API 23), LG G3 (API 21)
BCM4356	Motorola Nexus 6 (API 24), HTC One M9 (API 23)
QCA6174	LG Nexus 5X (API 25)
BCM4330	Samsung Galaxy S3 (API 19)

- 10 different Android models from 2013 to 2016.
- Factory state before experimentation.
- Dedicated WiFi network for communication.

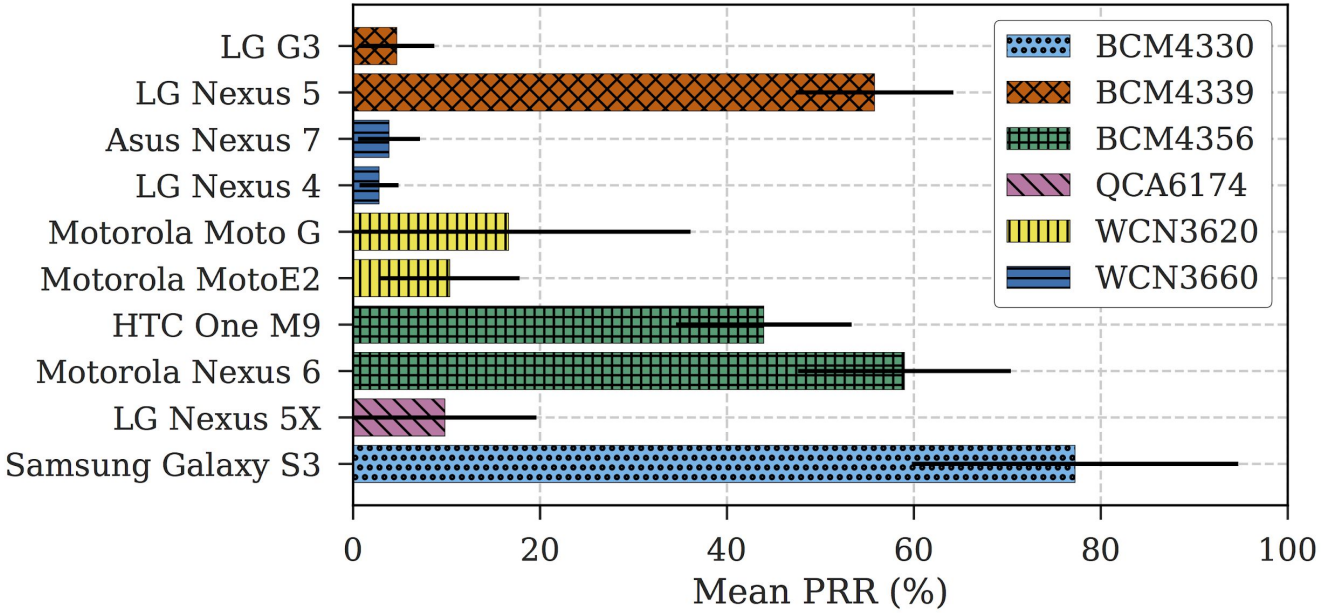
Impact of OS Implementations



Advertisement Packet Reception Ratio (PRR) for the Same Smartphone on Various OS Versions

→ **OS versions** (and BLE driver implementations) significantly **impact BLE performance**.

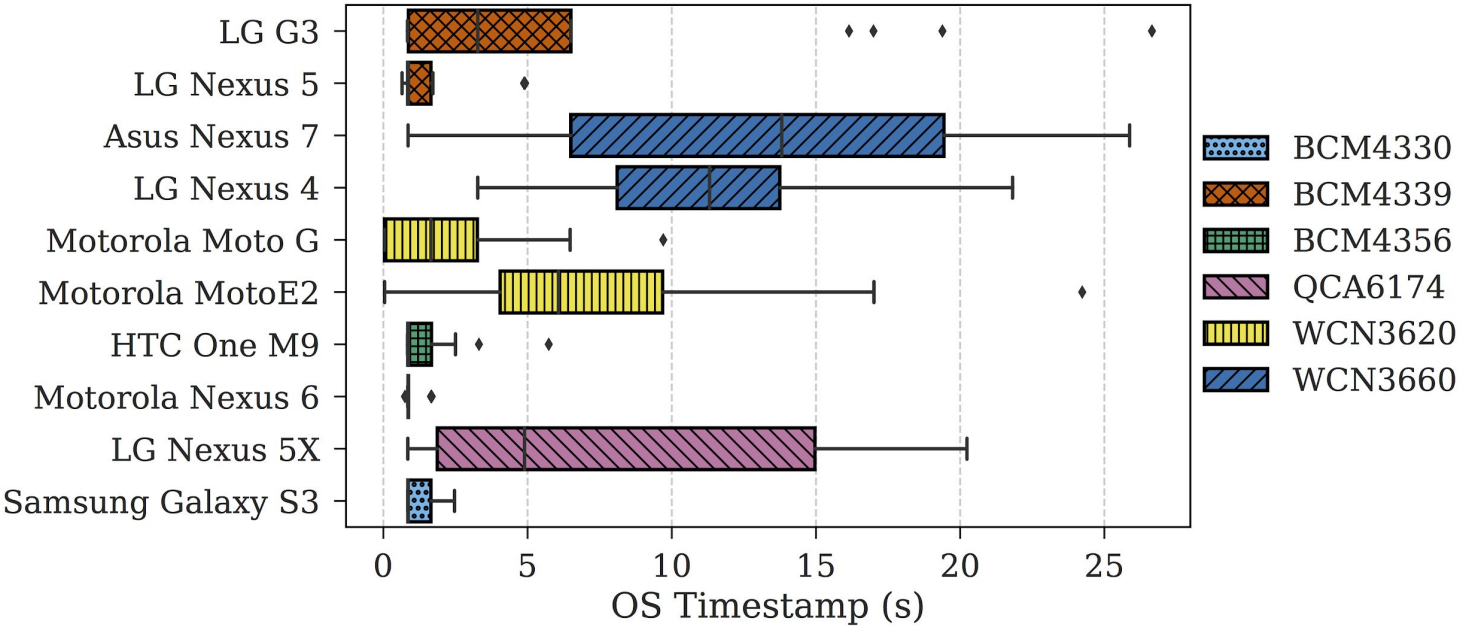
Impact of OS/SoC Implementations



Advertisement PRR of Phone Models:

→ SoC significantly impacts PRR. Phones with the same SoC exhibit similar performance, except for LG G3 because of highly customized OS.

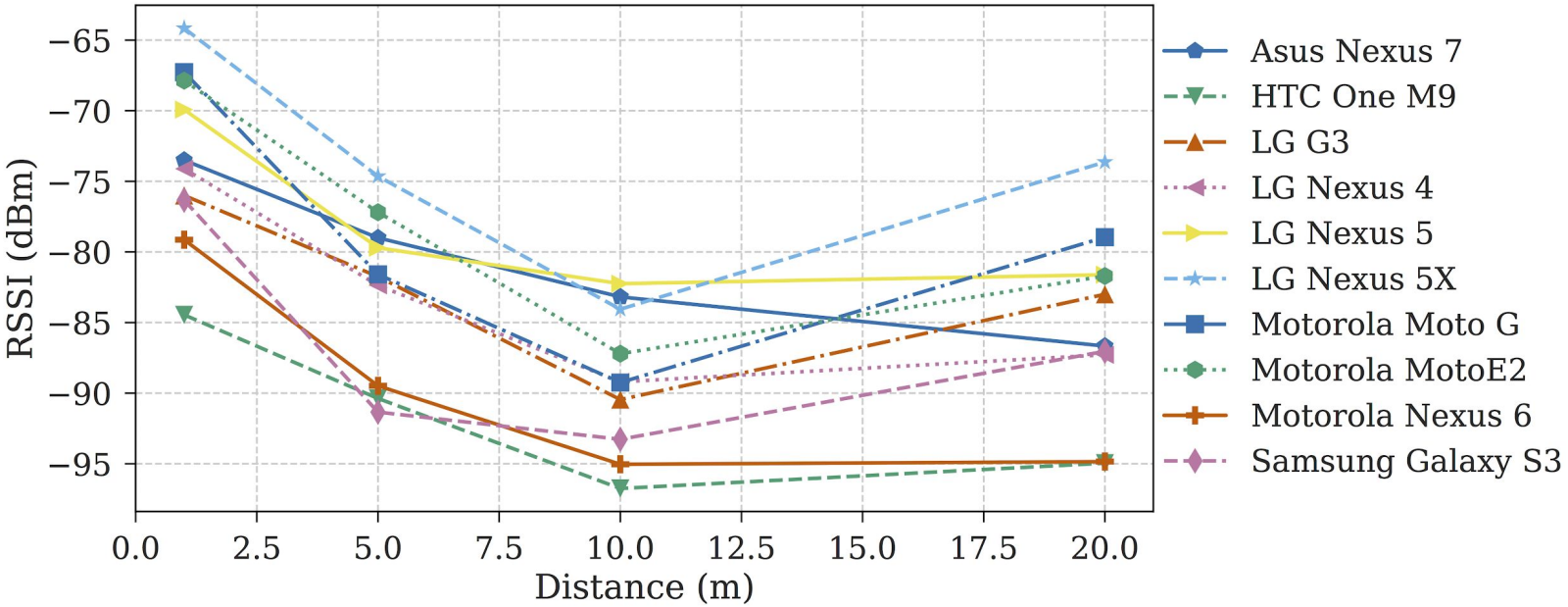
Impact of OS/SoC Implementations



Advertisement Latency of Phone Models:

→ Advertising latency varies with both SoC vendor and SoC model of a same vendor.

Impact of OS/SoC/Hardware



Distance vs. Mean RSSI of Phone Models:

→ The impact of the smartphone model nullifies the impact of distance on RSSI in many cases.

Conclusion and Open Questions

Conclusion

Hypothesis (1): BLE is accessed ‘indirectly’, through several (partly hidden) abstraction layers and proprietary drivers, which leads to unpredictable BLE behavior across smartphone models with different OS versions.

Hypothesis (2): SoC and BLE chip implementations are two major factors that lead to different BLE performance.

Hypothesis (3): Hardware components, such as amplifier, antenna, and cover, impact BLE performance.

→ **Experiments support the two first hypotheses.**

Open Questions

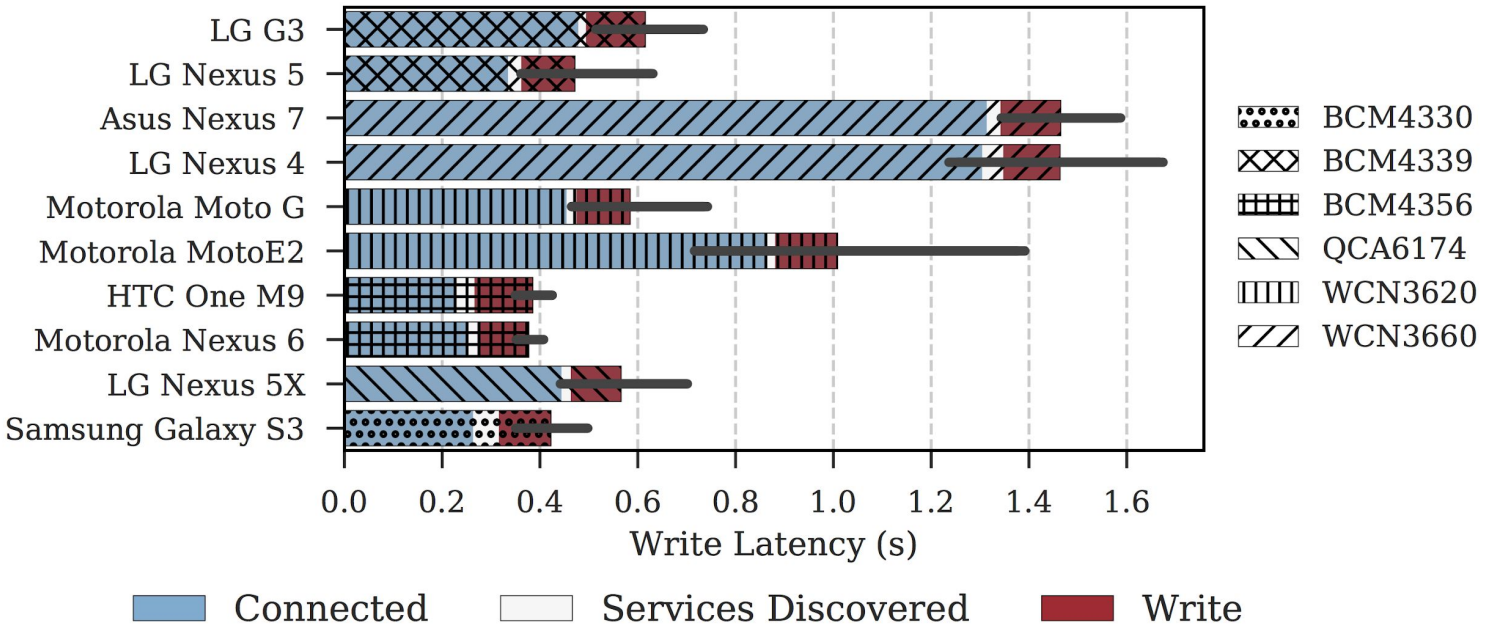
- What should be the API abstraction level on Android/iOS?
- Impact on design and implementation of embedded BLE systems?
- What does it take for applications to become “BLE- conscious”?

BLEva Repository: https://github.com/jf87/BLEva

Thank you!

Additional Slides

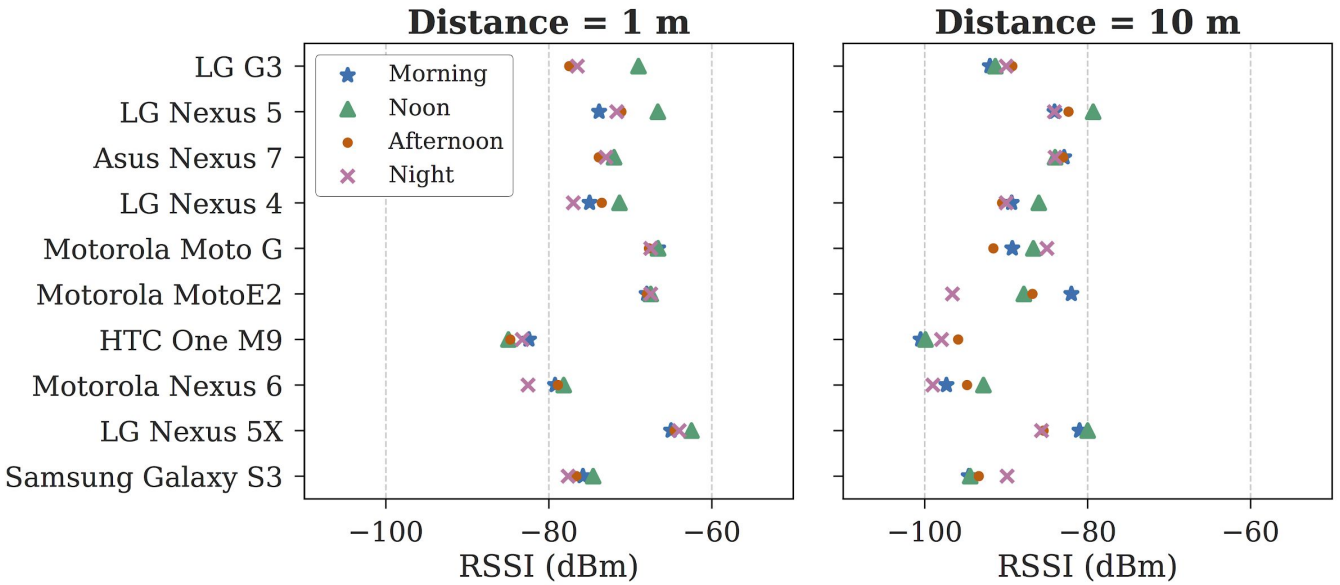
Impact of OS/SoC Implementations



Write Latency of Phone Models:

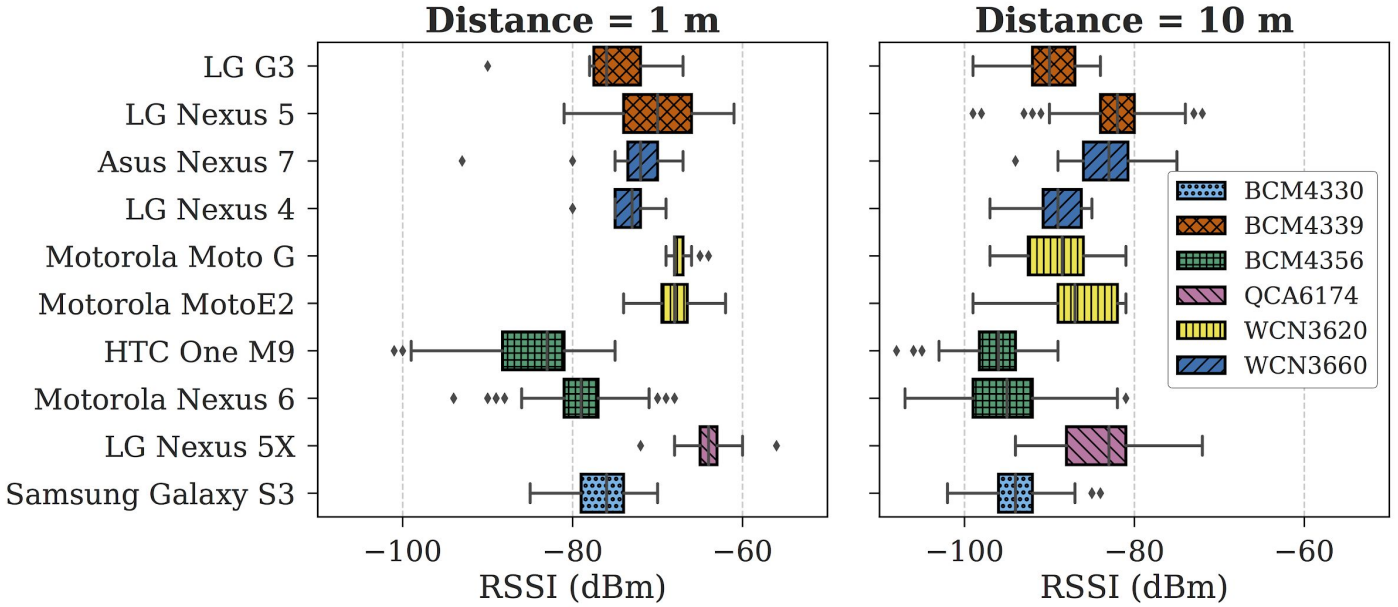
→ Write latency is SoC-bound, but varies significantly across SoCs.

Impact of OS/SoC/Hardware



Mean RSSI value at each of **four measurement times**

Impact of OS/SoC/Hardware



Every **RSSI** sample value during the **whole experiment period**.