# Experience: Android Resists Liberation from Its Primary Use Case

**Noah Klugman**[†], Veronica Jacome[†], Meghan Clark[†], Matthew Podolsky[†], Pat Pannuto[†], Neal Jackson[†], Aley Soud Nassor[‡], Catherine Wolfram[†], Duncan Callaway[†], Jay Taneja[*], and Prabal Dutta[†]

ACM MobiCom, Nov 1st, 2018
New Delhi, India

† * ‡

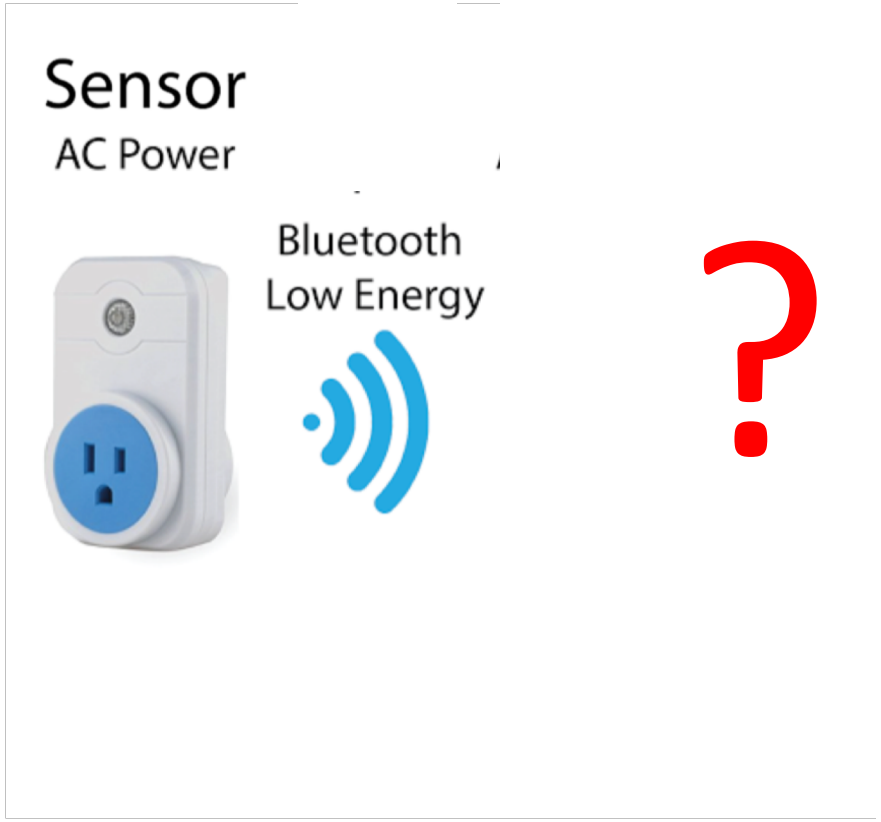
# Application: Sensing AC grid reliability in Zanzibar

Michamvi, one of our two deployment sites in Zanzibar

Photo by Veronica Jacome

# Challenge: get data from sensor to cloud

- Sensor: WiTenergy E110
  - Voltage, current
  - Bluetooth Low Energy
- Backend
  - node.js + influx
- Gateway: ?



# Insight: Android smartphones should make the gateway as easy as writing an app

- Sensor: WiTenergy E110
  - Voltage, current
  - Bluetooth Low Energy
- Backend
  - node.js + influx
- Gateway
  - Android smartphone!
  - Mature ecosystem
  - Background services
  - BLE radio, GSM radio, SD card
  - OTA updates

# Insight: Android smartphones should make the gateway as easy as writing an app …right?

- Sensor: WiTenergy E110
  - Voltage, current
  - Bluetooth Low Energy
- Backend
  - node.js + influx
- Gateway
  - Android smartphone!
  - Mature ecosystem
  - Background services
  - BLE radio, GSM radio, SD card
  - OTA updates

# Let’s start putting together a deployment…

- Pick a phone: Samsung J110H
  - Mid-range hardware
  - Low-cost ($90)
  - Widely available in Tanzania
  - Upgraded to Android 4.4.4
    - (and eventually rooted….)

# Write an app…

Bluetooth

Listen for Bluetooth packets with power data

Every 1 second, POST latest data

Cellular

Backend

Save copy of data to local storage

# And send it out into the field!

- Deployment of 16 systems in two villages in Zanzibar
- Plugged in at an outlet inside a household
- Phone was placed in a plastic box and screwed shut
- PlugWatch designed to run continually

# PlugWatch was supposed to be easy…

- Simple system based on commercially available components
- We assumed Android would trivially act as a gateway

# Phones on their own?

- Most smartphone use cases are interactive, a human holding a phone
- Tight Association[1] :
  - Assumption of frequent interactions between the human and the phone
- Loose Association:
  - Apps are continuously-running, unsupervised, and characterized by machine-to-machine interactions



[1] Ivan Beschastnikh, Yuan Zhang, Zhengping Qian, and Lidong Zhou. 2012. Liberating Mobile Phones from their Primary Use Case.

# Latent Hypothesis

- Smartphones enable unattended long-running networked sensing applications through their rich supporting environments, wireless peripheral connectivity, and physical reliability.

# Latent Hypothesis / Outline

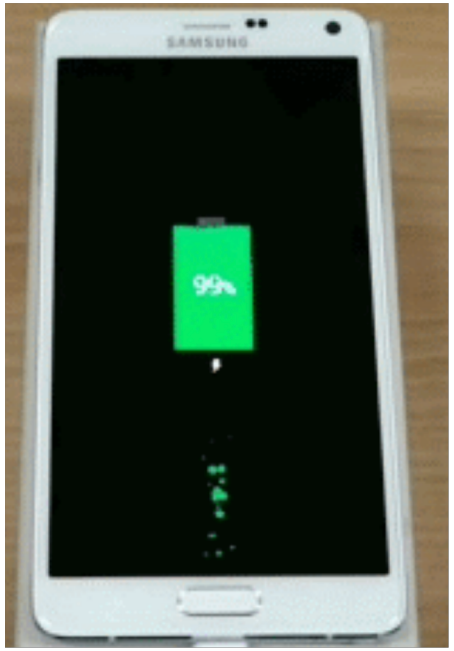
- Smartphones enable unattended long-running networked sensing applications through their rich supporting environments, wireless peripheral connectivity, and physical reliability.

# Latent Hypothesis / Outline

- Smartphones enable unattended long-running networked sensing applications through their rich supporting environments, wireless peripheral connectivity, and physical reliability.

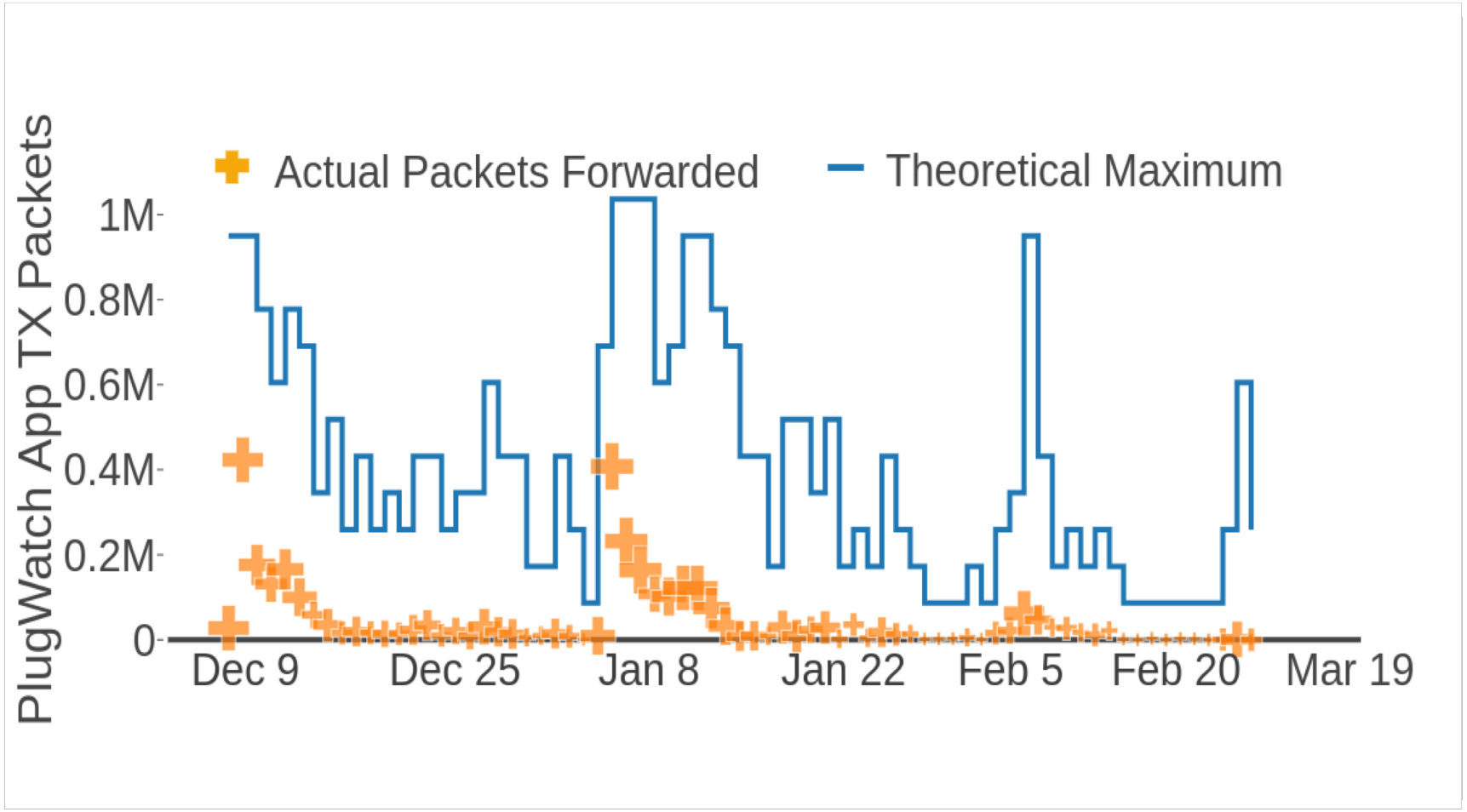
# It is difficult to keep apps running without a human in the loop.

- OS garbage collection makes it hard to run an app for long periods
- Tricks to keep a loose association app alive:
  - Force the UI open on the screen
  - Generate notifications
  - Catch crashes and reboots to restart app
  - Close modal windows
  - Multiple watchdog processes
  - Overwrite the charging image to auto start the phone on charge restored



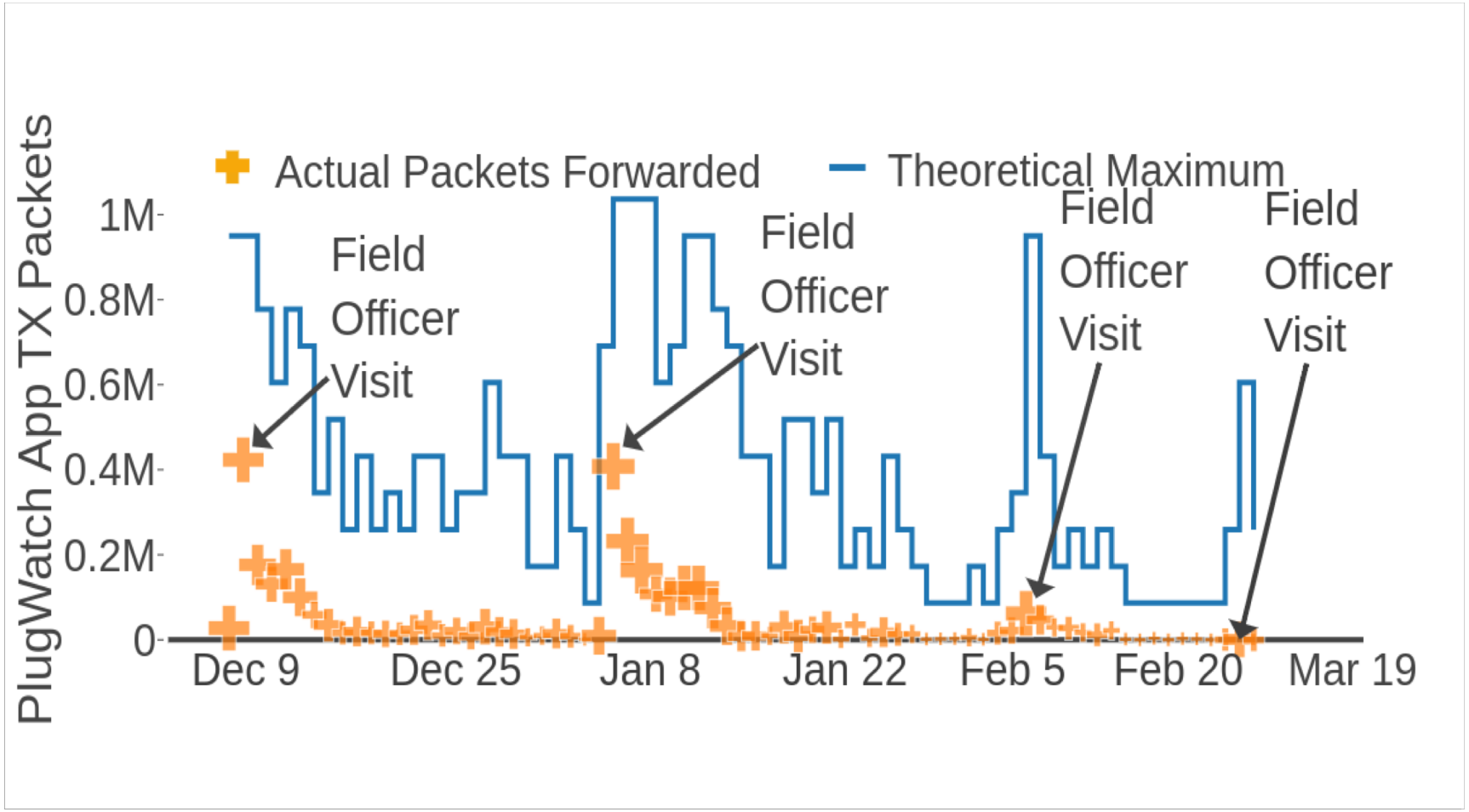

https://news.samsung.com/global/this-is-an-unofficial-review-of-the-galaxy-note-4-apps-ultra-power-saving-mode-fast-charging-and-more

# And still… the system drastically underperformed




1 Hz data *x* number of seconds in day *x* number of phones reporting at all that day

# Over long timespans phone build up residue that humans need to clear out.




1 Hz data *x* number of seconds in day *x* number of phones reporting at all that day

# Hypothesis / Outline

- Smartphones enable unattended long-running networked sensing applications through their rich supporting environments, wireless peripheral connectivity, and physical reliability.

# Hypothesis / Outline

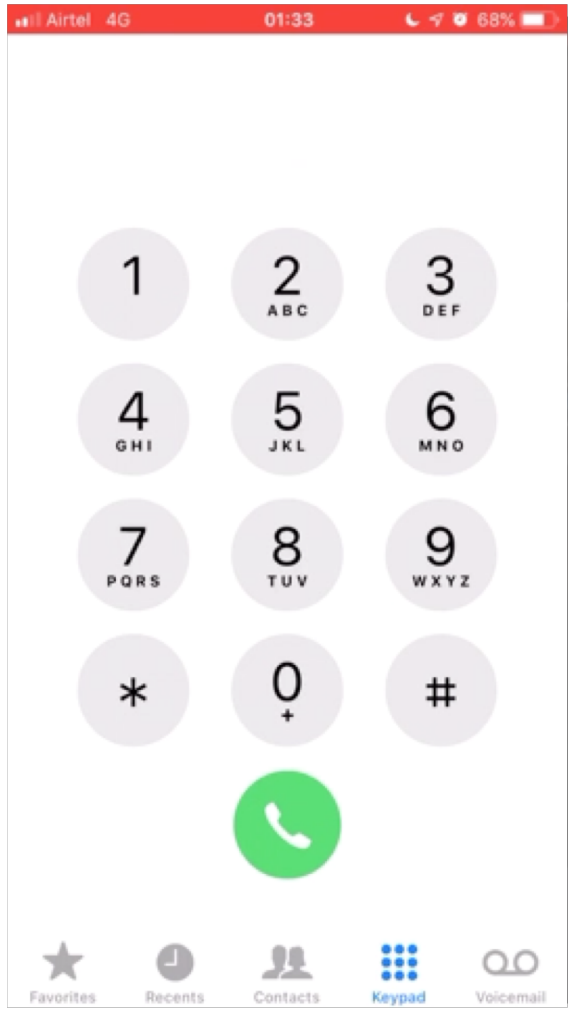
- Smartphones enable ~~unattended long-running~~ networked sensing applications through their rich supporting environments, wireless peripheral connectivity, and physical reliability.

# Hypothesis / Outline

- Smartphones enable ~~unattended long-running~~ networked sensing applications through their rich supporting environments, wireless peripheral connectivity, and physical reliability.

# Keeping multiple phones connected remotely was difficult

- SIM card data plans were non-trivial.
- Hard to diagnose connectivity problems.
  - Adding data done via scratch-off cards
  - Data had to be topped up every 30 days in-country
  - Hard to diagnose connectivity issues
- Fleet management is not well supported.


Check Balance?

# Hypothesis / Outline

- Smartphones enable ~~unattended long-running~~ networked sensing applications through their rich supporting environments, wireless peripheral connectivity, and physical reliability.

# Hypothesis / Outline

- Smartphones enable ~~unattended long-running networked sensing applications~~ through their rich supporting environments, wireless peripheral connectivity, and physical reliability.

# Hypothesis / Outline

- Smartphones enable ~~unattended long-running networked sensing applications~~ through their rich supporting environments, wireless peripheral connectivity, and physical reliability.

# Android ecosystem necessitates careful thinking

- Not all Android is created equal
  - Ecosystem does not equally support all APIs
- Automatic App Updates?
  - ~~Android Profiles~~ (req 5.0+)
  - Non-deterministic Google account logouts make orphans
- Bug fixes don’t reach down into old API’s
  - And OS updates always require human intervention

# Hypothesis / Outline

- Smartphones enable ~~unattended long-running networked sensing applications~~ through their rich supporting environments, wireless peripheral connectivity, and physical reliability.

# Hypothesis / Outline

- Smartphones enable ~~unattended long-running networked sensing applications~~ through their ~~rich supporting environments~~, wireless peripheral connectivity, and physical reliability.

# Hypothesis / Outline

- Smartphones enable ~~unattended long-running networked sensing applications~~ through their ~~rich supporting environments~~, wireless peripheral connectivity, and physical reliability.

# We encountered significant bugs in the Bluetooth Low Energy stack

| Row | Error Message | Reports |
|---|---|---|
| 1 | watchdog2 rebooting due to dead process | 257,547 |
| 2 | gridwatch.plugwatch.wit. PlugWatchService:bluetooth stack died | 108,653 |
| 3 | gridwatch.plugwatch.wit. PlugWatchService:unable to start scanning | 40714 |
| 4 | restarting due to timeout | 30,898 |
| 5 | service disconnected | 28,211 |
| 6 | An error occurred while executing doInBackground() | 18,643 |
| 7 | watchdog rebooting due to dead process | 3,981 |
| 8 | gridwatch.plugwatch.wit. ConnectionCheckService:restart rebooting due to max timeout | 3,836 |
| 9 | Exception thrown on Scheduler.Worker thread. Add ‘onError‘ handling. | 1,680 |
| 10 | [memory exhausted] | 398 |

”stack died”

”failed to scan”

“max timeout”

# Android's design makes recovery from subsystem bugs a high-cost operation

- When the bug is in the OS, applications don't have a great recovery strategy
- Other computing platforms can power cycle peripherals
- Android cannot reboot just the BLE stack
  - Had to restart app
  - Had to reboot phone
  - And the app doesn't always come back…

# Hypothesis / Outline

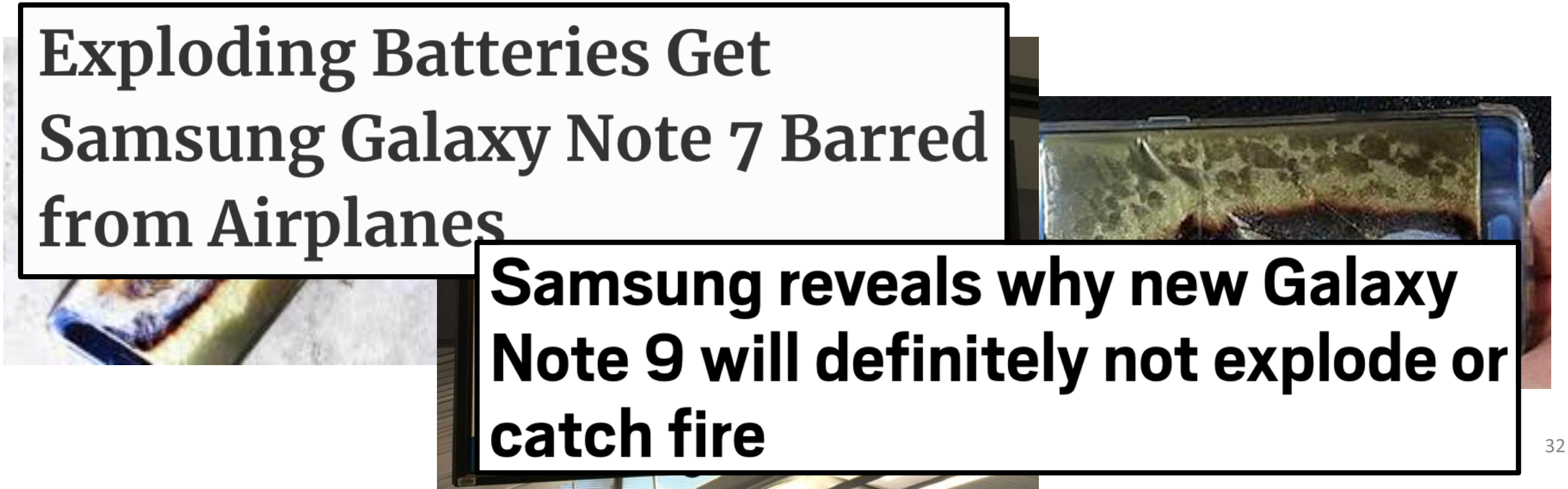
- Smartphones enable ~~unattended long-running networked sensing applications~~ through their ~~rich supporting environments~~, wireless peripheral connectivity, and physical reliability.

# Hypothesis / Outline

- Smartphones enable ~~unattended long-running networked sensing applications~~ through their ~~rich supporting environments~~, ~~wireless peripheral connectivity~~, and physical reliability.

# Hypothesis / Outline

- Smartphones enable ~~unattended long-running networked sensing applications~~ through their ~~rich supporting environments~~, ~~wireless peripheral connectivity~~, and physical reliability.



# Batteries failed catastrophically


Not much data available on the effects of unsupervised long-running applications on batteries

- We baked phone running app in an oven at 120°F for 18 hours, did not reveal issues

# Hypothesis / Outline

- Smartphones enable ~~unattended~~ ~~long-running~~ ~~networked sensing~~ ~~applications~~ through their ~~rich supporting environments~~, ~~wireless peripheral connectivity~~, and physical reliability.

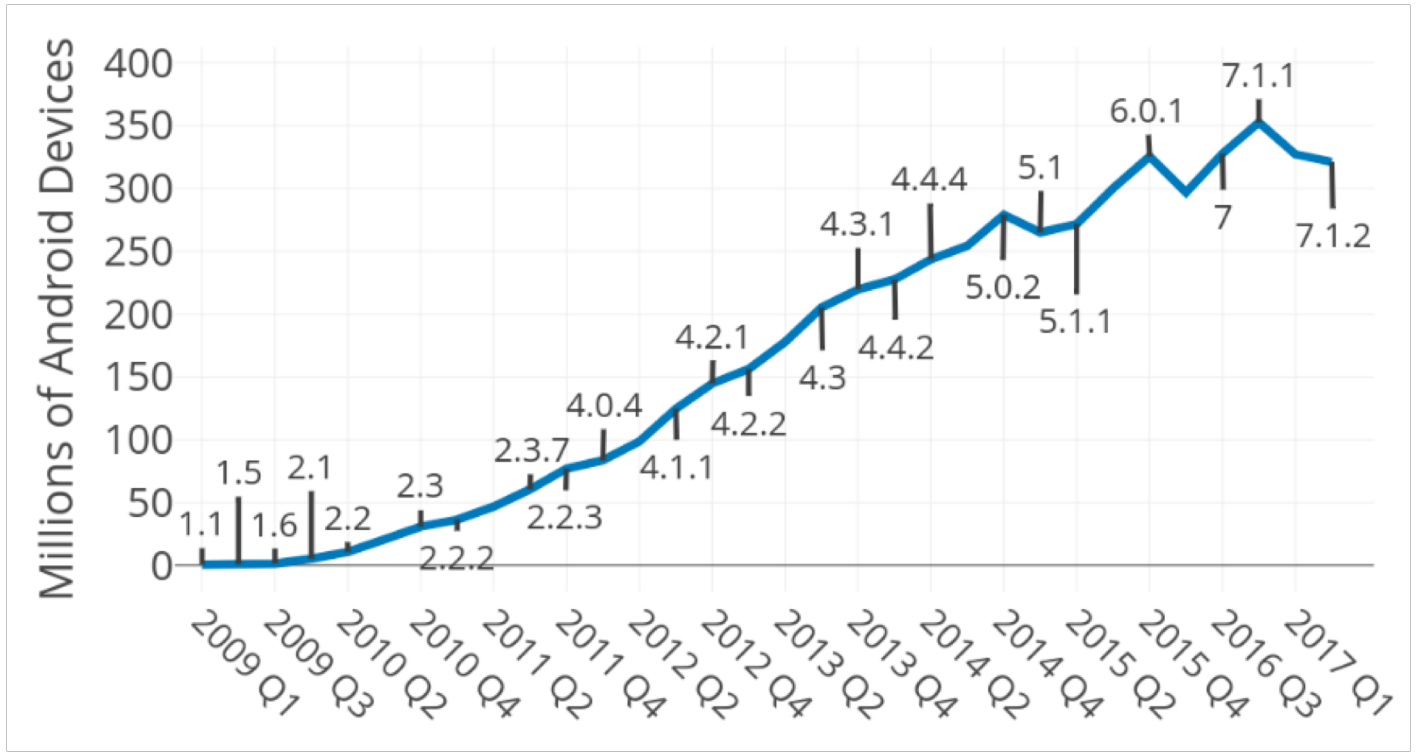
# Hypothesis / Outline

- Smartphones enable ~~unattended long-running networked sensing applications~~ through their ~~rich supporting environments~~, ~~wireless peripheral connectivity~~, and ~~physical reliability~~.

# Hypothesis / Outline

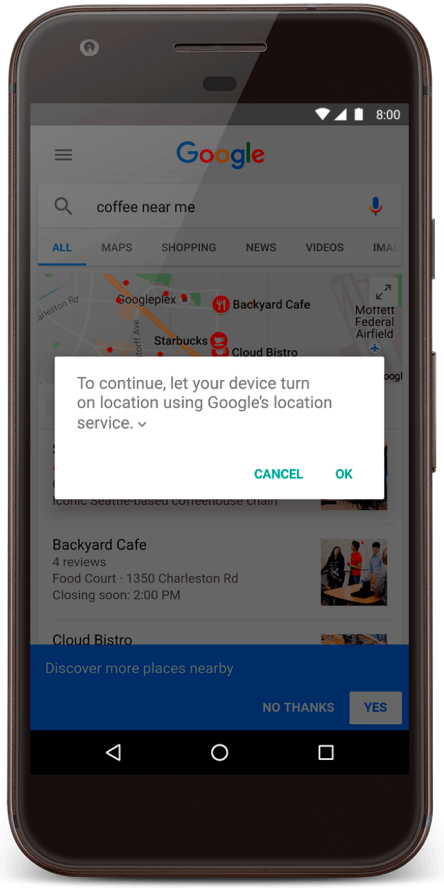
- Smartphones enable ~~unattended long-running networked sensing applications~~ through their ~~rich supporting environments~~, ~~wireless peripheral connectivity~~, and ~~physical reliability~~.

# Phone reuse is a huge opportunity to enable compute and reclaim e-waste.




# Loose association requires future work

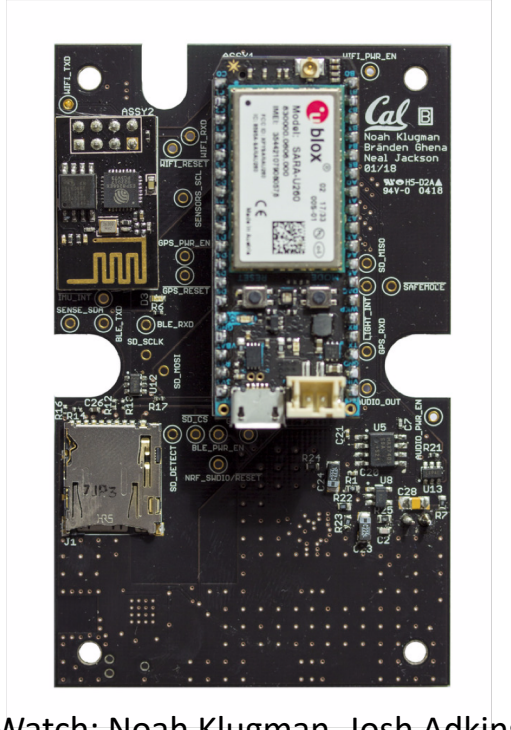
- Android is doubling down on tight association (as they should)
- A loose association version of Android
- A solution for heterogeneity
- Recycling centers




https://developer.android.com/distribute/best-practices/develop/runtime-permissions

# The PlugWatch is dead. Long live the PowerWatch.

- Custom hardware solution developed and deployed in Ghana
- Hundreds of sensors and millions of data points




PowerWatch: Noah Klugman, Josh Adkins, Pat Pannuto, Matt Podolsky, Neal Jackson, Branden Ghena, Jay Taneja, Prabal Dutta

PowerWatch devices being prepared for deployment

# Thank you!

## Experience: Android Resists Liberation from Its Primary Use Case

**Noah Klugman**[†], Veronica Jacome[†], Meghan Clark[†], Matthew Podolsky[†],Pat Pannuto[†], Neal Jackson[†], Aley Soud Nassor[‡], Catherine Wolfram[†], Duncan Callaway[†], Jay Taneja[*], and Prabal Dutta[†]

[†] **University of California, Berkeley**
Department of Electrical Engineering
Department of Energy and Resources
Energy Institute at the Haas Business School

[*] **University of Massachusetts, Amherst**
Department of Electrical Engineering

[‡] **The State University of Zanzibar**
Department of Development Studies