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...

1. Bluetooth
   - Listen for Bluetooth packets with power data

2. Save copy of data to local storage

3. Every 1 second, POST latest data

4. Cellular
   - Backend
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\textsuperscript{[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

\textsuperscript{[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

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.
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.
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

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

"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.

Exploding Batteries Get Samsung Galaxy Note 7 Barred from Airplanes

Samsung reveals why new Galaxy Note 9 will definitely not explode or catch fire
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
• 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

The PlugWatch is dead. Long live the PowerWatch.

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

‡

The State University of Zanzibar
Department of Development Studies