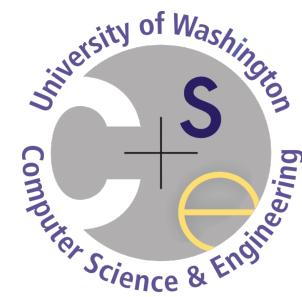


Improving Power Efficiency Using Sensor Hub Without Re-Coding Mobile Apps

Haichen Shen, Aruna Balasubramanian,
Anthony LaMarca, David Wetherall



Intel Science and Technology Center for
Pervasive Computing

Continuous sensing apps provide a wide range of services



BeWell, Acoustic: Lifestyle monitoring



Ambulation: Healthcare monitoring



MobiPerf: Participatory sensing

but are huge energy consumers...



A Google User - August 22, 2012 - Samsung Galaxy Nexus with version 3.0.120704r635 [go](#)

★ ★ ★ ★ ★ Destroys your battery

Appears to drain battery

Assumes
your batter



Locale app, \$10



A Google User - June 17, 2012 - Samsung Galaxy S with version 3.0.2 [go](#)

★ ★ ★ ★ ★ Love the app. Hate the power drain.

Like many people I installed Locale to help me control power usage. What I found lately is that Locale is causing a race condition in wakelocks that is preventing my phone from entering deep sleep. Disabled locale and all is well again. I am using CM9 on a Galaxy S. Please investigate and fix this issue so I can start using the app again.



This app does what it says, but it alone was consuming about 10% of my battery and about half my memory. As if I didn't have enough performance issues already with my phone. Uninstalled!

Why consume so much energy?

10.00s: (0.1, 0.2, 9.8)

10.02s: (-0.2, 0.1, 9.8)

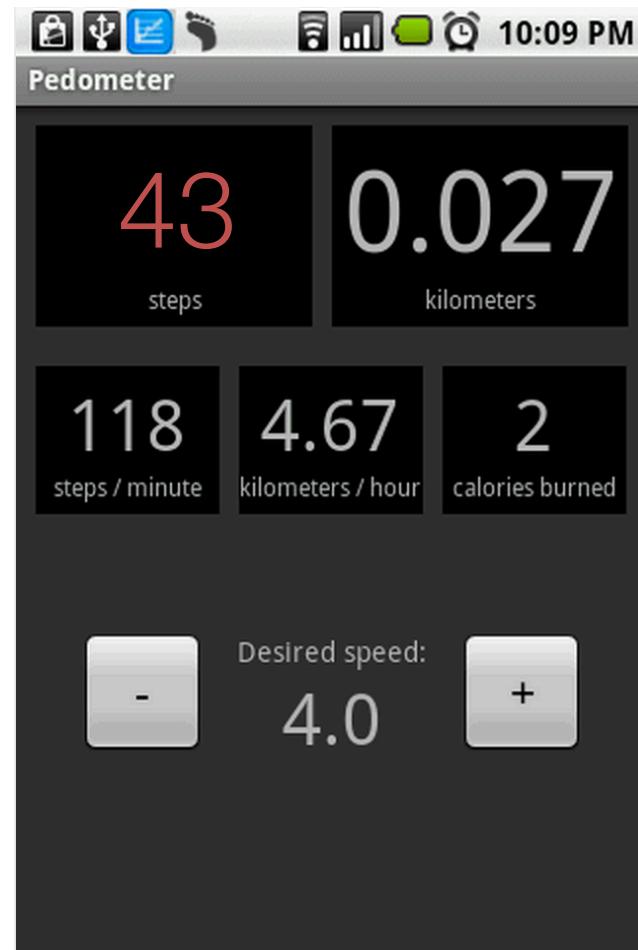
10.04s: (-0.3, -0.2, 9.7)

.....

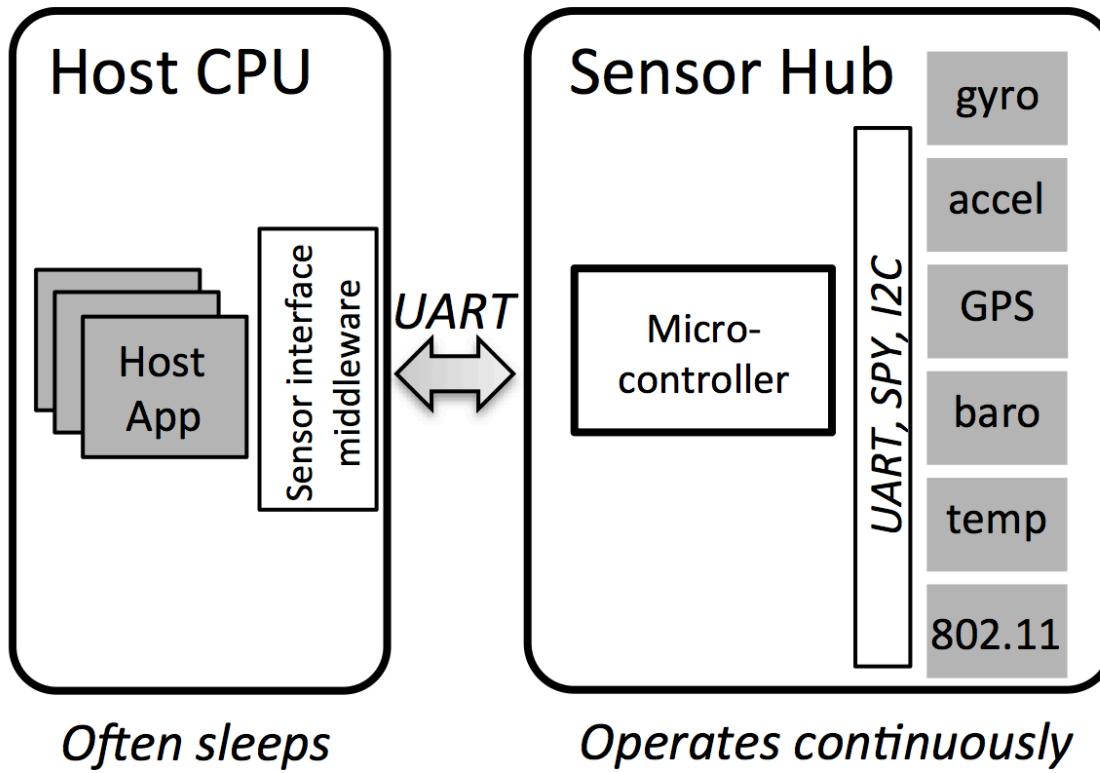
.....

.....

10.56s: (-4.2, 2.1, 9.5)



Low power dedicated sensor hub



Intel Merrifield



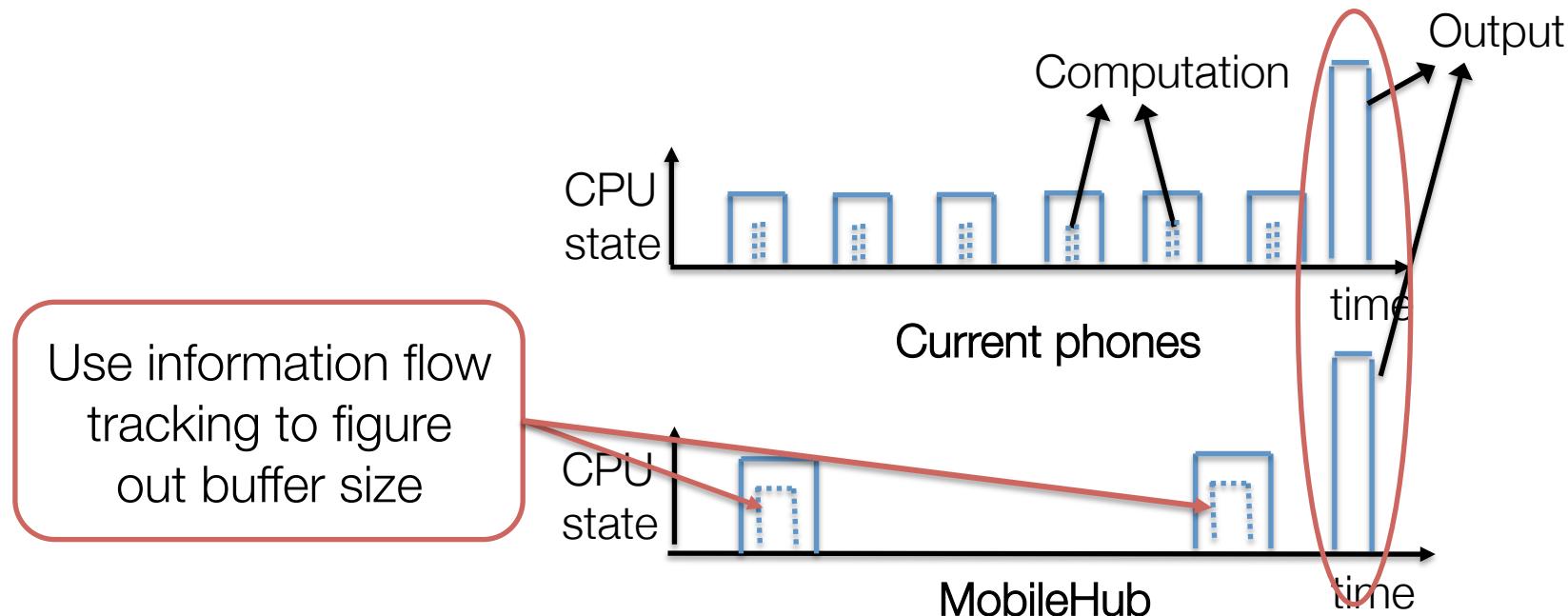
TI's Tiva



How to use the sensor hub?

MobileHub: rewriting existing apps

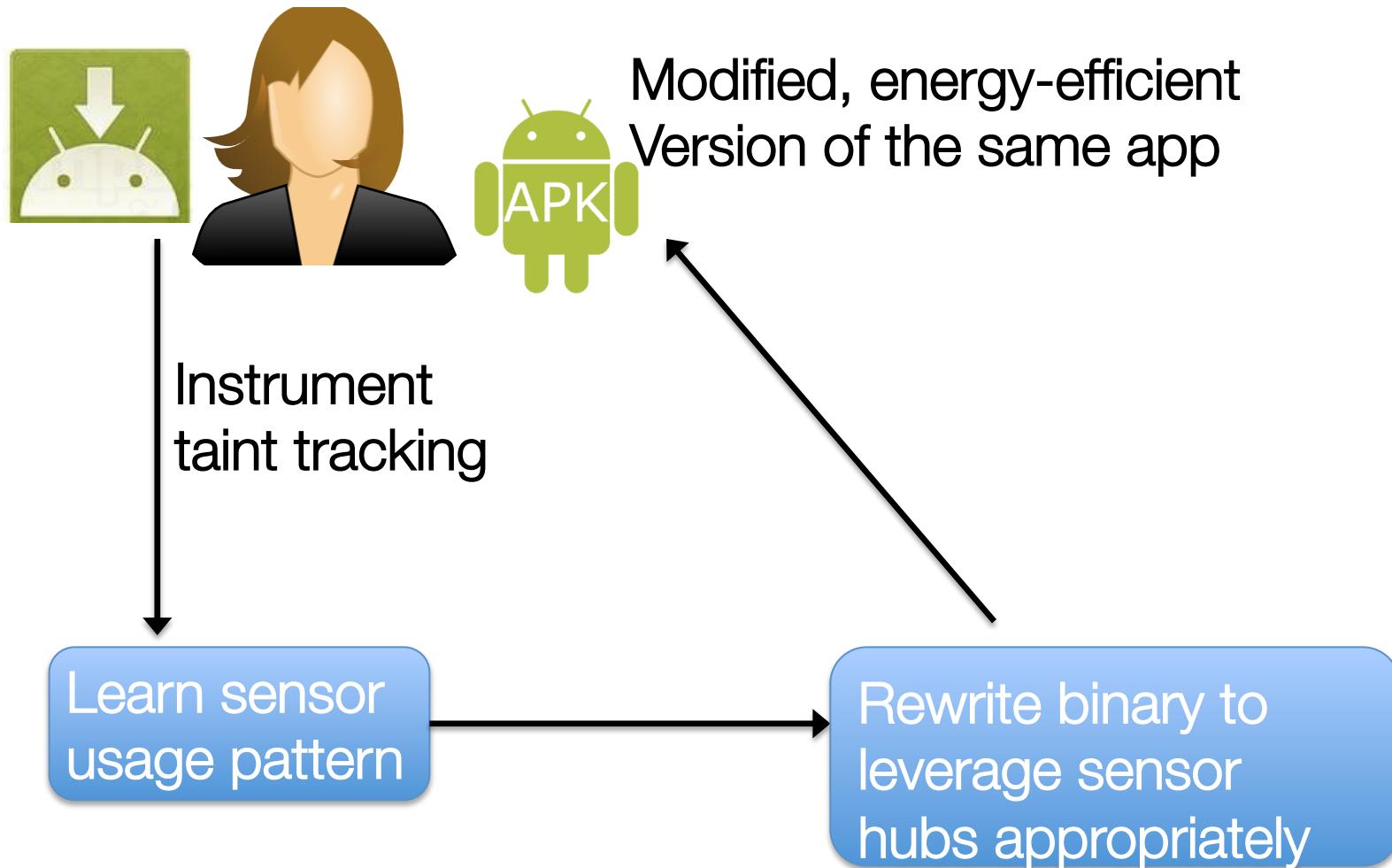
- Sensor Hub: simply buffering
- App: use sensor hub without re-coding



MobileHub: Three Components

- Information flow tracking
 - Study how apps use sensor data
- Protocol, API, and policies design
 - Provide an easy-to-use toolkit for developers
- Re-architecting the OS and rewriting the apps
 - Enabling sensor hub in the mobile system

Overview



Data flow tracking

- Adapt TaintDroid system to track explicit information flow
 - Taint source: all different types of sensors
 - Taint sink: UI update, send to network and save to disk

Control flow tracking: instrumentation

```
1: void onSensorChanged(SensorEvent value) {  
2:     avg = (value.x+value.y)/2;  
3:     if (avg > THRESHOLD) {  
4:         stepCounter++;  
5:         updateUI(stepCounter);  
6:     }  
7: }
```

Control flow tracking: instrumentation



```
1: void onSensorChanged(SensorEvent value) {  
2:     avg = (value.x+value.y)/2;  
3:     if (avg > THRESHOLD) {  
4:         stepCounter++;  
5:         updateUI(stepCounter);  
6:     }  
7: }
```

Control flow tracking: instrumentation

```
1: void onSensorChanged(SensorEvent value) {  
2:     avg = (value.x+value.y)/2;  
3:     if (avg > THRESHOLD) {  
4:         stepCounter++;  
5:         updateUI(stepCounter);  
6:     }  
7: }
```

Control flow tracking: instrumentation

```
1: void onSensorChanged(SensorEvent value) {  
2:     avg = (value.x+value.y)/2;  
3:     if (avg > THRESHOLD) {  
4:         stepCounter++;  
5:         taint(stepCounter);  
6:     }  
7: }
```



Control flow tracking: instrumentation

```
1: void onSensorChanged(SensorEvent value) {  
2:     avg = (value.x+value.y)/2;  
3:     if (avg > THRESHOLD) {  
4:         stepCounter++;  
5:         taint(stepCounter);  
6:     }  
7: }
```

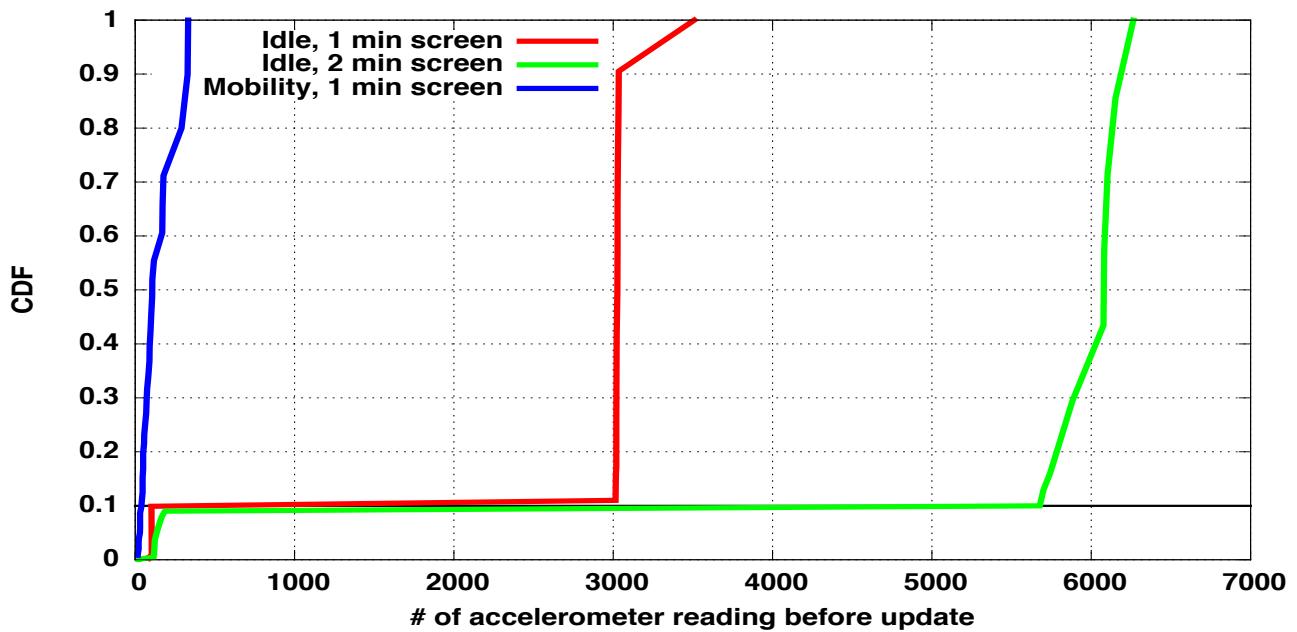


A red starburst shape with the word "Captured!" written in white inside it, indicating that the variable "stepCounter" has been tainted or captured.

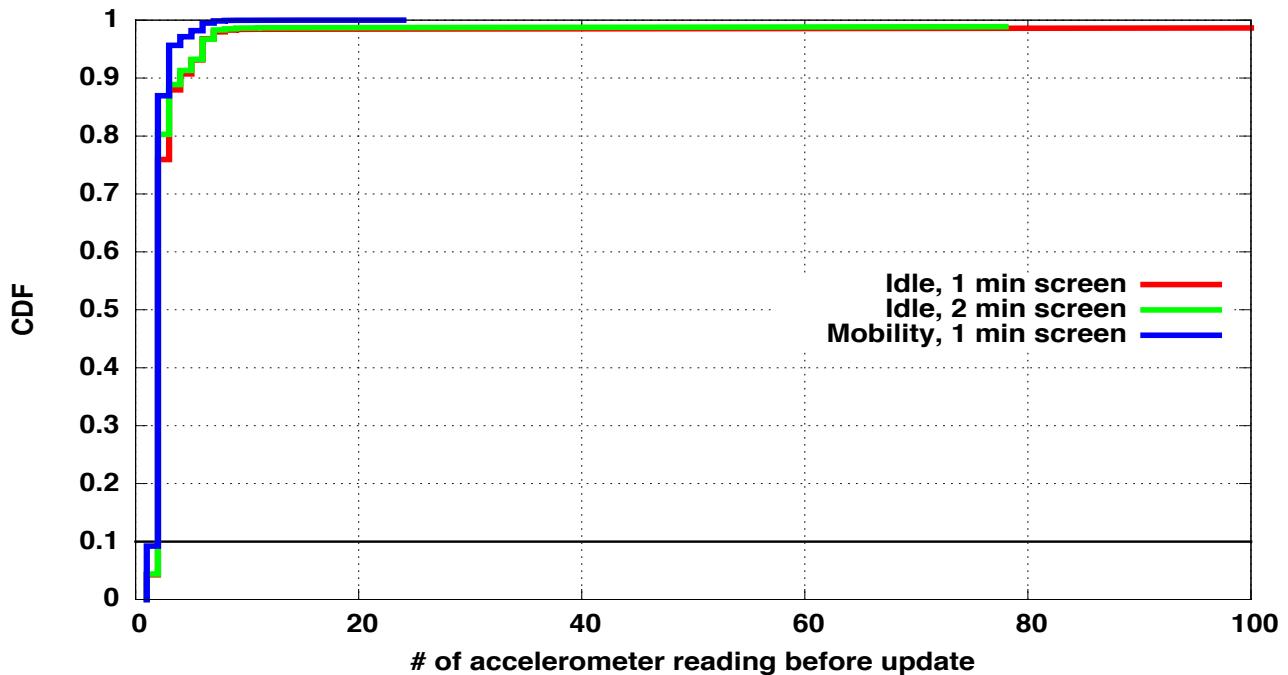
Tainted fields

Name	Total fields	Total tainted fields	Data flow taint	Control flow taint
nWalk	506	28	10.7%	100.0%
Walking	497	12	66.7%	58.3%
Pedometer	304	27	11.1%	100.0%
Pedometer Pro	689	27	44.4%	66.7%
Universal	440	8	87.5%	12.5%
Step Counter	685	5	20.0%	100.0%
Simple Steps	125	18	38.9%	61.1%

Pedometer



Pedometer Pro



API

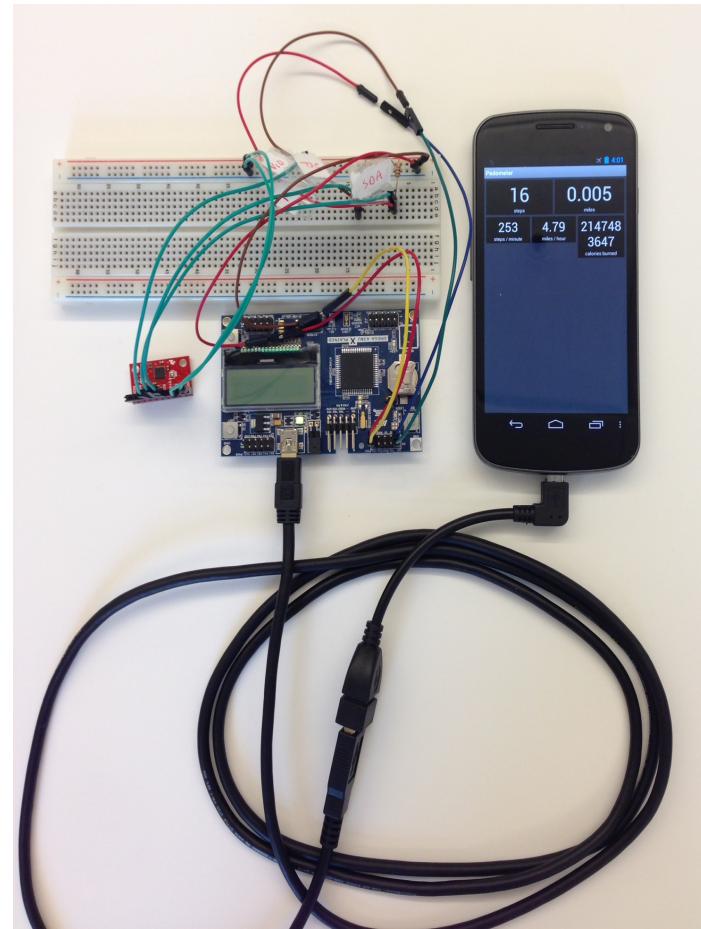
```
boolean registerListener(  
    SensorEventListener listener,  
    Sensor sensor,  
    int rate,  
    int bufferSize,  
    Conditions cond);  
  
void unregisterListener(  
    SensorEventListener listener,  
    Sensor sensor,  
    Conditions cond);
```

Policy

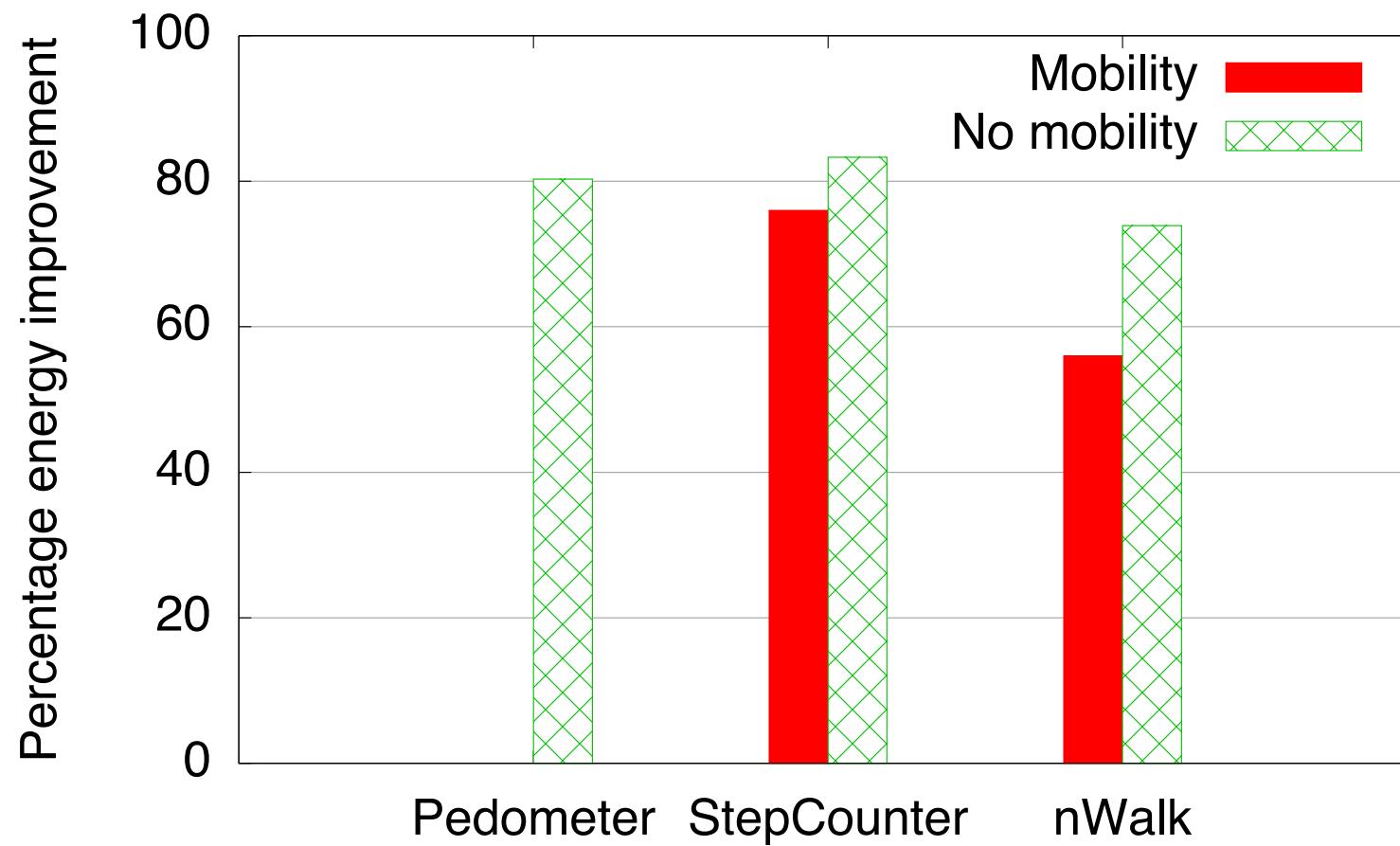
- Foreground
 - No buffering
- Background
 - Buffer size depends on different condition
 - Still / Walking / Running / ...

Prototype

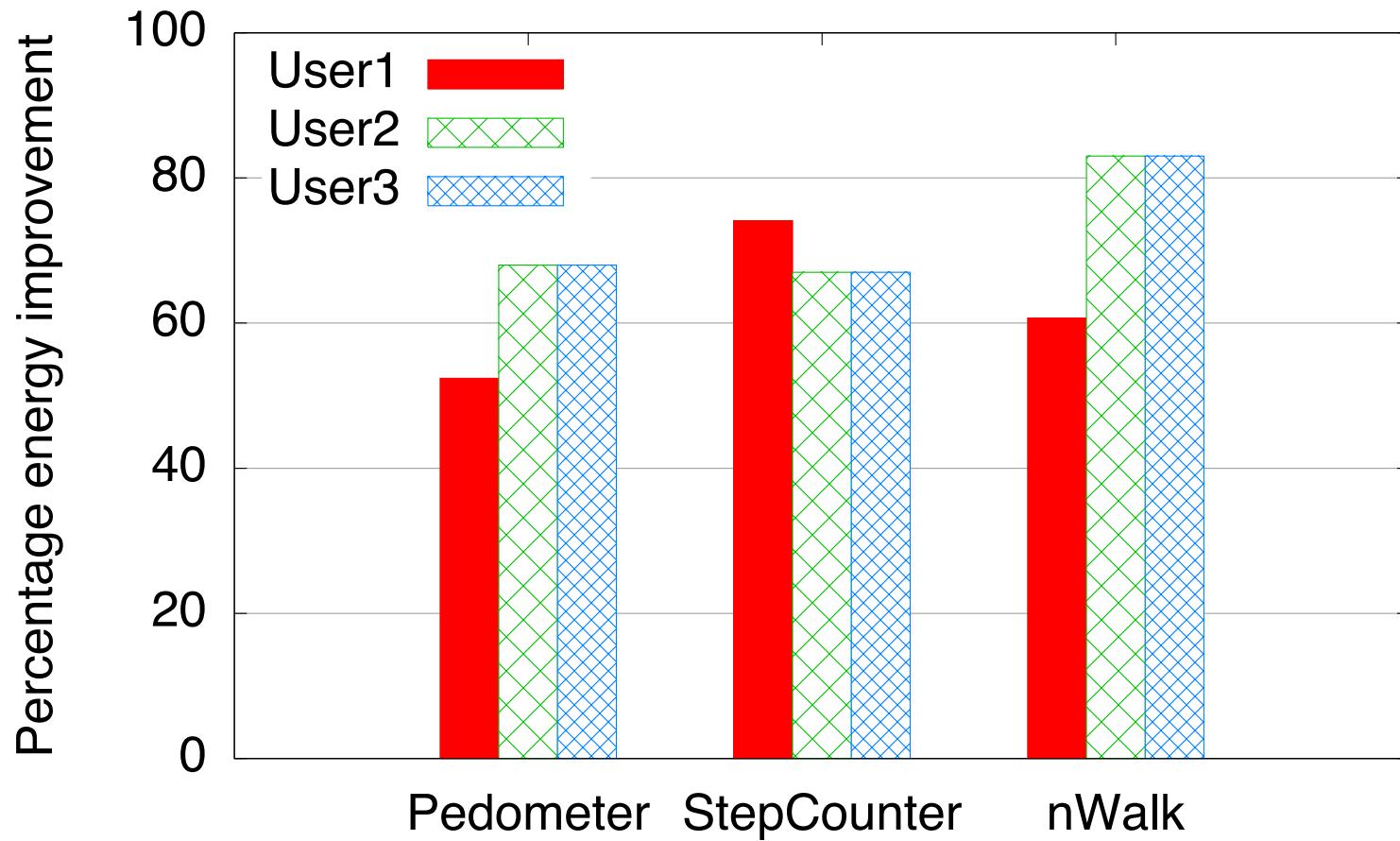
- Galaxy Nexus
 - Android 4.2.2
- Atmel AVR Xmega-A3BU
- Connection: USB On-The-Go



Energy consumption comparison



Trace Comparison



Conclusion

- Design and implementation of MobileHub
- An information flow tool that tracks data and control flows
- An evaluation that shows using MobileHub can achieve up to 80% power gain