Singapore Management University
Institutional Knowledge at Singapore Management University

Research Collection School Of Information Systems

School of Information Systems

Demo: Profiling power utilization behaviours of smartwatch applications

Joseph Joo Keng CHAN
Singapore Management University, joseph.chan.2012@msis.smu.edu.sg

Lingxiao JIANG
Singapore Management University, lxjiang@smu.edu.sg

Rajesh Krishna BALAN
Singapore Management University, rajesh@smu.edu.sg

Youngki LEE
Singapore Management University, YOUNGKILEE@smu.edu.sg

Archan MISRA
Singapore Management University, archanm@smu.edu.sg

Follow this and additional works at: https://ink.library.smu.edu.sg/sis_research

Part of the Software Engineering Commons

Citation
CHAN, Joseph Joo Keng; JIANG, Lingxiao; BALAN, Rajesh Krishna; LEE, Youngki; and MISRA, Archan. Demo: Profiling power utilization behaviours of smartwatch applications. (2016). *MobiSys '16 Companion: Proceedings of the 14th Annual International Conference on Mobile Systems, Applications, and Services Companion, Singapore, June 26-30*. 96-96. Research Collection School Of Information Systems.
Available at: https://ink.library.smu.edu.sg/sis_research/3278

Demo: Profiling Power Utilization Behaviours of Smartwatch Applications

Joseph Chan Joo Keng, Lingxiao Jiang, Rajesh Krishna Balan, Youngki Lee and
Archan Misra
School of Information Systems
Singapore Management University
{joseph.chan.2012}@phdis.smu.edu.sg, {lxjiang, rajesh, youngkilee,
archanm}@smu.edu.sg

1. INTRODUCTION

Smartwatches complement the main mobile phone and are able to profile user-activity as well as provide links, updates and notifications with work or personal utilities (e.g. Email, Social Media, Messaging etc.). Although very promising, smartwatches are still limited by low battery life. This is due to the small size of the battery as well as the need to continuously perform sensing. Battery drain issues by apps as well as the system are a common complaint by users. Improved tools for power analysis and profiling of smartwatch apps can help both developers (by providing a platform for pre-release analysis) and users (by advising them on app characteristics prior to installation).

2. PROFILING POWER UTILIZATION OF ANDROID WEAR

We implemented a power testing analysis platform tool that profiles the power consumption of the Android Wear smartwatch, and relates it to any method-call from within the binary. The setup consists of the smartwatch linked up to the Monsoon power meter [3], and running a self instrumented Android SDK tool that traces and outputs all methods (developer or system) of a particular app in a time series. The method traces logged are then related to the power meter readings, and synchronized with the CPU timestamp in an approach similar to A.Banerjee el al. [1]. The challenges we had to overcome include accurate synchronization as well as decoupling of multiple call traces.

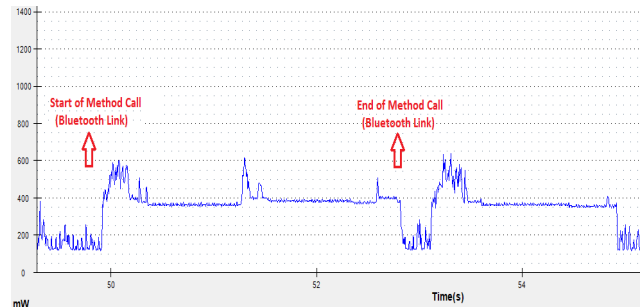We run automated testing of the mobile app run-

Figure 1: Power Utilization (mW) over Time (s) for Entry and Exit points of a Bluetooth-related method call for an Android Wear app

ning on the paired mobile phone using automated app testers [2]. All method program traces as well as power meter readings during traversal were logged. An analysis program then allows observation and analysis of the time series power variations during the duration of invocation of any specified method call that we choose. By studying these power variations, we can profile the apps during instances of high power utilization of the on-board sensors. Beside analysis of power utilization behaviours, this also allows us to potentially uncover bugs and inefficiencies in app code.

2.1 Demonstration Details

We demonstrate our platform and how the apps are tested together with the logging of runtime method calls and power readings. Next, we show how our tool analyzes the resulting logs. We provide the required method names and signatures of interest (smartwatch hardware or sensor-related), and the tool outputs and plots the resulting power profile during the execution of these methods.

3. REFERENCES

[1] A. Banerjee, L. K. Chong, S. Chattopadhyay, and A. Roychoudhury. Detecting energy bugs and hotspots in mobile apps. In *Proceedings of the 22nd ACM SIGSOFT International Symposium on Foundations of Software Engineering*, pages 588–598. ACM, 2014.

[2] J. C. J. Keng, L. Jiang, T. K. Wee, and R. K. Balan. Graph-aided directed testing of android applications for checking runtime privacy behaviours. In *11th IEEE/ACM International Workshop on Automation of Software Test (AST 2016)*, 2016.

[3] M. Solutions. Inc.,'monsoon power monitor,'.