ACM HotMobile 2013 Poster: Bugu: An Application Level Power Profiler and Analyzer for Mobile Devices

Youhuizi Li **Hui Chen** **Weisong Shi**
huizi,huichen,weisong@wayne.edu
Department of Computer Science, Wayne State University, Detroit, MI, USA

*Mobile devices, such as smart phones and tablets, have become an integral part of our daily life, providing a lot of fancy and powerful applications. To understand and solve the battery drain problem, we design and implement the **Bugu** service which targets the applications running on mobile devices, analyzes event-power relationship, and provides users an overview of the power behavior of applications. We envision that three groups of people will benefit from the Bugu service. For end users, they know applications' power behavior which in turn helps them to decide which applications to install and run. For application developers, they could understand which events cause such amount of power dissipation and focus on optimizing them. For system developers, the insights provided by the Bugu service will enable them to understand the potential problem of the system so that further optimization can be enhanced.*

I. Introduction

Nowadays, mobile devices have become an important part of our life. At the same time, more and more applications are developed. There is no doubt that these applications make our life more convenient, but they are also big energy consumers and significantly influence battery lifetime and user experience. As an end user, we want to know *for the same functionality, which application is more energy-friendly?* Except the battery issue, energy efficient applications are more competitive on the market.From the viewpoint of application developers, they care more about *why my applications consume such amount of power?* System developers focus on the whole system, not just some components or specific applications. Answering the question *how to save and effectively control system power* is the final goal of system developers. The first action is understanding the energy consumption of the system and applications.

In order to answer the questions raised by these three groups of people, we designed and implemented the Bugu service, which is an application level power profiler and analyzer. As Figure 1 illustrates, the Bugu server returns related applications' power information to end users and gives them more suggestions when they choose applications. For application developers, aside from the similar applications' power information gathered from the server, the Bugu client also shows event information of their applications, so that these power hotspots can be optimized. From the viewpoint of system developers, detailed system

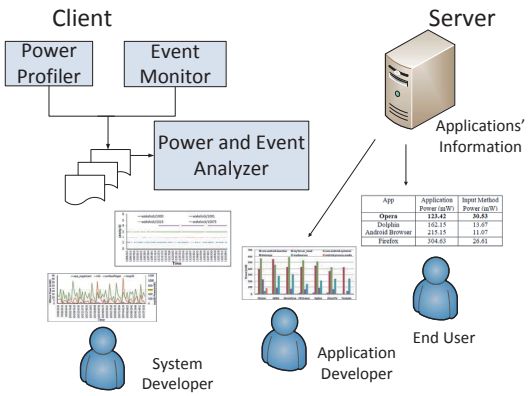

Figure 1: The overview of Bugu.

power information provided by the Bugu client is helpful for them to adopt power saving mechanisms, such as background processes' power variation. In detail, the Bugu client demonstrates power and event information in figures and tables for easy understanding.

II. System Design

The Bugu server has two functions: collecting application power information from the Bugu client and supplying this information to users. Users can contribute their data to the Bugu server, which will help future customers. Through crowdsourcing, we build a large database of application power consumption information on different types of mobile devices. After ranking these applications, users can get a better understanding before installing them. We categorize the applications into several categories: video, game, so-

cial network client, and so on. Application developers can compare the power consumption of their application with others to evaluate the products.

The Bugu client has three main functions: estimating application-level power consumption, monitoring system and application events and displaying the information to the user in a meaningful way. The power profiler uses a group of energy models to estimate energy consumption based on how much of each hardware resource was utilized by each application. More information about power models can be found in our former paper [2]. The power profiler considers the power consumption of the following components: CPU, Wi-Fi, 3G, GPS, sensors, bluetooth, screen, radio, and so on. Besides, Bugu also monitor the events of a system and applications. Those events include: wakelock, Wi-Fi state change, bluetooth state change, audio and video state change and so forth. For example, we could know when an application acquired or released a wakelock, which represents the privilege to use several hardware devices.

III. Evaluation

We use a video application as an example to show how Bugu works. In our experiment, the new application developed is $YouTube$, and we test the basic functionality of it. From Figure 2, we can see that system processes which support our applications consume much more power than the application itself, such as $systemui$ which draws the user interface and $mediaserver$ which provides sound and other support for media. On the perspective of target application, $YouTube$ is in a good situation, its power is lower than others. Hence, for end user, it is a good choice from power point of view. Figure 3 presents the event information of $YouTube$. The x-axis is time; y-axis is event ID which is defined by event type (wakelock, sensor, screen, etc.) and the process that generates these events. Except system processes, $YouTube$ also occupies the wakelock for a long time. That's the part developers can analyze more to optimize the application, such as improve wakelock utilization.

Bugu provides power information of all processes running in the system, which helps system developers to know the whole picture. Similar with the video application experiment, most target applications' power are low, while system processes still consume a lot of power. Moreover, they periodically occupy the wakelock make the system can hardly in "sleep" mode. Hence, system developers should focus more on optimizing these background processes.

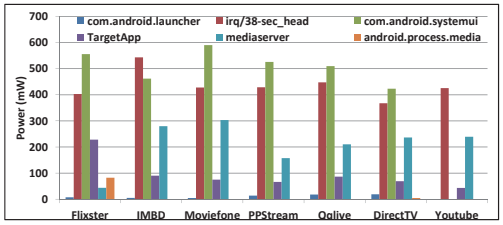

Figure 2: The power comparison of seven video applications.

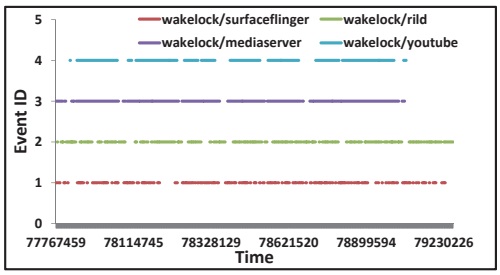

Figure 3: YouTube event information on Nexus S.

IV. Implication and Conclusion

We found that the system processes, such as $rild$, $wpa\_supplicant$, $systemui$, consume much energy even in background. Besides, wakelock causes a particular "no sleep" bug and the hardware interrupts generated by sensors also consume a lot of energy. If operating systems provide more energy efficient APIs/services for developers, the energy of applications will automatically decrease. From the comparison data recorded in the Bugu server, we know the applications have different power behavior even with same functionality, and users prefer to choose energy-friendly software. So, it is possible and necessary to optimize applications' energy [1].

In this paper, we proposed Bugu, an application level power profiler and analyzer for mobile devices. Bugu is available at `http://codegreen.cs.wayne.edu/bugu/`. The Bugu server provides power information of different applications, while the client side analyzes event-power relationship for specific application. We implemented Bugu on Android platform and did several experiments to evaluate it. The results of the Bugu service are useful for many energy/power related researches on mobile devices.

References

[1] Susanne Albers. Energy-efficient algorithms. *Commun. ACM*, 53(5):86–96, May 2010.

[2] Hui Chen, Youhuizi Li, and Weisong Shi. Fine-grained power management using process-level profiling. In *Sustainable Computing: Informatics and Systems*, SUSCOM, Jan 2012.