A Thesis

entitled

Optimizing Android Memory Management by Predicting User Behavior

by

Srinivas Muthu

Submitted to the Graduate Faculty as partial fulfillment of the requirements for the

Masters of Science Degree in Engineering

Dr. Jackson Carvalho, Committee Chair

Dr. Mansoor Alam, Committee Member

Dr. Henry Ledgard, Committee Member

Dr. Patricia R. Komuniecki, Dean
College of Graduate Studies

The University of Toledo
December 2015

An Abstract of

Optimizing Android Memory Management
by Predicting User Behavior

by

Srinivas Muthu

Submitted to the Graduate Faculty as partial fulfillment of the requirements for the
Masters of Science Degree in Engineering

The University of Toledo
December 2015

With the advent of increase in the amount of RAM available in Android smart-phones, there is a case to be made that this additional memory availability can be put to better use. By default, every Android application runs in its own Linux process. Android starts the process when any of the application's components need to be executed, then shuts down the process when it's no longer needed or when the system must recover memory for other applications. When a smart-phone that runs on the Android operating system is active (turned on), the RAM contains all the active processes and services (processes that run in the background). In addition to these processes and services, the RAM also contains cached background processes. These processes are kept in memory so that in the event the user clicks any of these applications, they can be loaded onto the screen quickly as they're already in memory and don't have to be fetched from the disk. If Android needs to reclaim memory for other processes, it eliminates cached background processes through an LRU scheme. Ideally we would like every application the user clicked to be present in memory but due to constraints in RAM availability, Android employs the LRU scheme to decide which processes to retain. I postulate that looking into user behavior could help us better determine which applications to cache in memory, as cached background processes. I parse the user's Calendar for contextual clues and gather information

that could help me predict which application a user is about to use. I measure the cache efficiency (Cache hits and misses, CHR) for the default CRA Android employs (LRU), a pure prediction based CRA and finally a hybrid approach that combines the LRU approach with the prediction approach. I also demonstrate why the hybrid CRA is the most efficient one.

To my parents who've always put my needs ahead of theirs.

Acknowledgments

First and foremost, I'd like to thank Dr. Jackson Carvalho for mentoring me throughout my time here at UT. Without his guidance, I would'nt be half the student I am today. I'd like to thank Dr. Mansoor Alam for believing in me and offering me tuition scholarship to pursue graduate studies, here at UT. I'd like to thank Dr. Lawrence Thomas for being an exemplary professor and his words of wisdom have always guided me in tough times.

I am grateful to Dr. Henry Ledgard for taking the time to be on my committee and mentoring me during my freshman year. I'd like to thank Dr. Donald White and his students for helping me with data collection, design and representation. It was nothing short of a privelege to work with Dr. White and his students. I'd like to thank Dean Nagi G. Naganathan for employing me and guiding me over the course of my time in graduate school.

I'd like to thank all my friends and family for supporting me throughout my time here at UT but I'd like to especially mention Sandy Stewart for everything she's done for me. This page is'nt enough to expound on the details but it suffices to say I would'nt be here without her and she is like a second mother to me.

Contents

List of Tables

List of Figures

List of Abbreviations

RAM	Random Access Memory
LRU	Least Recently Used
CHR	Cache Hit Ratio
CRA	Cache Replacement Algorithm
OS	Operating System
AOSP	Android Open Source Project
APK	Android Package
IPC	Inter Process Communication
IC	Integrated Circuits
MB	Mega-Byte
GB	Giga-Byte

List of Symbols

$\ddagger$ .... the degree to which the flayrod has gone out of skew on tredel
$\triangle$ .... the ratio of the M2 monetary aggregate to the Monetary Base

α .... angle of rotation around internal rotation axis
β .... the number of people named "Bob"

Q .... Tobin's q; the ratio of the market value of installed capital to the replacement cost of capital
Y .... Gross Domestic Product (adjusted for inflation)

Preface

This thesis is original, unpublished, independent work by the author, Srinivas Muthu under the tutelage of Dr. Jackson Carvalho.

Chapter 1

Introduction

1.1 Android Operating System

1.1.1 Overview of the Android OS

Android is a mobile OS based on the Linux kernel and designed primarily for touchscreen devices such as smart-phones and tablets [1]. In addition to touchscreen devices, Android TV, Android Auto and Android Wear are emerging technologies with specialized user interfaces. Globally, it is the most popular mobile OS [2]. Android has an active community of developers and enthusiasts who use the AOSP source code to develop and distribute their own modified versions of the operating system [4]. Android homescreens are typically made up of app icons and widgets. App icons launch the associated app, whereas widgets display live, auto-updating content such as the weather forecast, the user's email inbox, or a news ticker directly on the homescreen [5].

Internally, Android OS is built on top of a Linux kernel. On top of the Linux kernel, there are the middleware, libraries and APIs written in C and application software running on an application framework. Development of the Linux kernel continues independently of other Android's source code bases.

Figure 1-1: [3] Android System Architecture

1.1.2 Applications, Activities and Services

1.1.2.1 Applications

Android apps are written in the Java programming language. The Android SDK tools compiles the code, along with any data and resource files into an APK, which is an archive file with an .apk suffix. One APK file contains all the contents of an Android app and is the file that Android-powered devices use to install the app [6]. The Android operating system is a multi-user Linux system in which each app is a different user. Each process has its own virtual machine (VM), so an app's code runs in isolation from other apps. By default, every app runs in its own Linux process.

1.1.2.2 Activities

An Activity is an application component that provides a screen with which users can interact in order to do something, such as dial the phone, take a photo, send an email, or view a map [7]. An application usually consists of multiple activities that are loosely bound to each other. Typically, one activity in an application is specified as the main activity, which is presented to the user when launching the application for the first time. Each activity can then start another activity in order to perform different actions.

1.1.3 Services

A Service is an application component that can perform long-running operations in the background and does not provide a user interface [8]. Another application component can start a service and it will continue to run in the background even if the user switches to another application. Additionally, a component can bind to a service to interact with it and even perform IPC. For example, a service might handle network transactions, play music, perform file I/O, or interact with a content provider, all from the background.

1.2 Growth of available RAM over the years

1.2.1 Advances in Technology

RAM is a form of computer data storage. A RAM device allows data items to be accessed (read or written) in almost the same amount of time irrespective of the physical location of data inside the memory. The overall goal of using a RAM device is to obtain the highest possible average access performance while minimizing the total cost of the entire memory system. Today, random-access memory takes the form of IC(s).

1.2.1.1 Moore's Law

Moore's law is the observation that the number of transistors in a dense IC doubles approximately every two years.

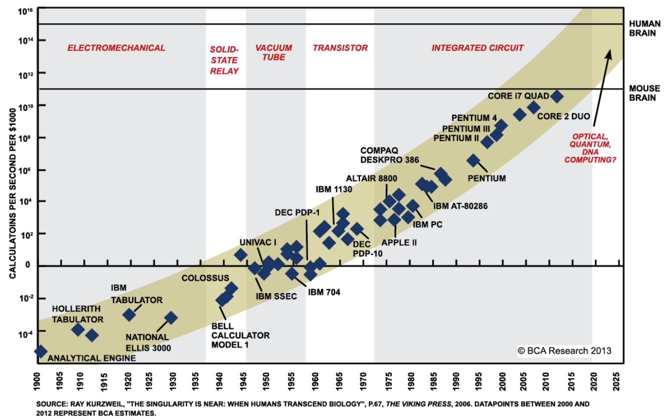

Figure 1-2: [9] Moore's Law

Advancements in digital electronics are strongly linked to Moore's law, especially in the context of memory capacity. T-Mobile G1, the very first Android smart-phone to be released back in 2008 [10] has a 256 MB RAM [11]. In comparison, the Nexus 6P that was released in September 2015 has a 3 GB RAM and some devices like the LG T585 have a 16 GB RAM [12][13], which amounts to 64 times the memory capacity of the T-Mobile G1. The rapid growth in the amount of RAM available to Android devices has in part led to newer possibilities and in our case, a better CRA for determining which processes (in the context of Android applications) remain in memory.

1.2.2 Demise of Task Killers

One of the main benefits of the Android OS is the fact that unlike certain other OS(s), it can run apps in the background. This enables us to have multiple applications open at the same time which results in true multitasking. Thus, RAM availability is highly desirable [14]. A popular misconception is that forcibly removing applications from memory in order to ’free up RAM’ will result in increased performance. In fact, several task-killer applications promise to do precisely that. With the advancement in the amount of RAM available, the debate should be about how to better use all this extra space, not killing applications to ’free up’ more space. In fact, the Android OS can go one step further and pro-actively cache applications that the user might use in the near future. We’ll analyze this prospect in more detail in the upcoming chapters.

1.3 Current CRA used by Android

1.3.1 LRU

A

1.3.2 Alternatives

A

1.3.3 Exceptions to the Rule

A

1.4 Goals and Objectives

1.4.1 A User-Centric CRA

A

1.4.2 Improvement in CHR and overall efficiency

A

1.4.3 Effects on the End User

A

1.5 Organization of Thesis

Describe briefly what each subsequent section aims to address.

Chapter 2

Related Work

Chapter 3

Phase A Design

Chapter 4

Phase B Design

Chapter 5

Data Analysis and Results

Chapter 6

Conclusion and Future Work

References

[**1**] "The Android Source Code" *http://source.android.com/source/index.html*, October 2015. Online; Last Accessed: October 22, 2015.

[**2**] "The Most Popular Smartphone Operating Systems Globally," *http://www.lifehacker.com.au/20 most-popular-smartphone-operating-systems-around-the-world/*, September 2014. Online; Last Accessed: October 22, 2015.

[**3**] "Android System Architecture" *https://mahalelabs.files.wordpress.com/2013/03/androidstack-1.jpg*, March 2013. Online; Last Accessed: October 22, 2015.

[**4**] "Best custom ROMs for the Samsung Galaxy S2" *http://www.cnet.com/uk/how-to/best-custom-roms-for-the-samsung-galaxy-s2/*, April 2012. Online; Last Accessed: October 22, 2015.

[**5**] "Widgets — Android Developers" *http://developer.android.com/design/patterns/widgets.html*, October 2015. Online; Last Accessed: October 22, 2015.

[**6**] "Application Fundamentals — Android Developers" *http://developer.android.com/guide/compon* October 2015. Online; Last Accessed: October 22, 2015.

[**7**] "Activities — Android Developers" *http://developer.android.com/guide/components/activities.h* October 2015. Online; Last Accessed: October 22, 2015.

[**8**] "Services — Android Developers" *http://developer.android.com/guide/components/services.htm* October 2015. Online; Last Accessed: October 22, 2015.

[**9**] "Wikimedia Upload" *http://www.extremetech.com/wp-content/uploads/2015/04/MooresLaw2.p* October 2015. Online; Last Accessed: October 22, 2015.

[**10**] "A Brief History of Android Phones" *http://www.cnet.com/news/a-brief-history-of-android-phones/*, August 2011. Online; Last Accessed: October 22, 2015.

[**11**] "GSM Arena" *http://www.gsmarena.com/t_mobile_g1-2533.php*, October 2008. Online; Last Accessed: October 22, 2015.

[**12**] "Nexus 6P" *http://www.androidcentral.com/nexus-6p*, October 2015. Online; Last Accessed: October 22, 2015.

[**13**] "Phone Egg — LG T585" *http://us.phonegg.com/phone/4783-LG-T585*, November 2013. Online; Last Accessed: October 22, 2015.

[**14**] "Android PSA: Stop Using Task Killer Apps" *http://phandroid.com/2011/06/16/android-psa-stop-using-task-killer-apps-now/*, June 2011. Online; Last Accessed: October 22, 2015.

Appendix A

Insert the Heading to Appendix A

[Insert text to Appendix A (if appendix is needed)]

Appendix B

Insert the Heading to Appendix B

[Insert text to Appendix B (if appendix is needed)]