



# INFO 498 D: Android Development

---

JOEL ROSS

WINTER 2016

# Plan for the Day

---

1. Introductions and syllabus
2. Android History
3. break!
4. Getting Started: "Hello World"

# Introductions

---

# Who is Joel?

**UCI** Department of Informatics  
Donald Bren School of Information & Computer Sciences



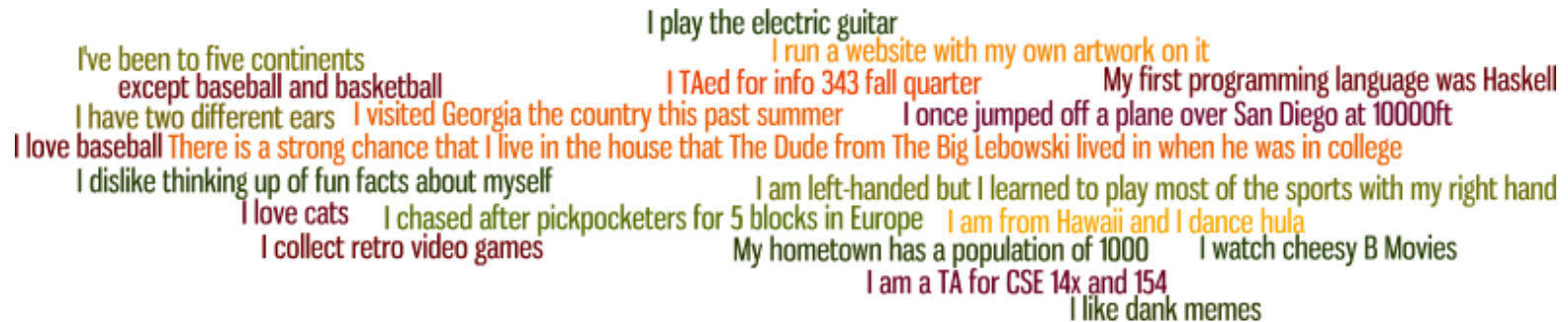
**UBICOMP**



# Who are you?

## Ask the person across from you

- Who are you?
- Where are you from?
- What are you most excited about for this class?



A word cloud of student self-descriptions, with text arranged in a roughly circular shape. The text includes:

- I play the electric guitar
- I run a website with my own artwork on it
- My first programming language was Haskell
- I TAed for info 343 fall quarter
- I once jumped off a plane over San Diego at 10000ft
- I visited Georgia the country this past summer
- There is a strong chance that I live in the house that The Dude from The Big Lebowski lived in when he was in college
- I am left-handed but I learned to play most of the sports with my right hand
- I am from Hawaii and I dance hula
- I watch cheesy B Movies
- My hometown has a population of 1000
- I am a TA for CSE 14x and 154
- I like dank memes
- I collect retro video games
- I chased after pickpocketers for 5 blocks in Europe
- I dislike thinking up of fun facts about myself
- I love cats
- I love baseball
- I have two different ears
- I've been to five continents except baseball and basketball

# Course Materials

## **On Canvas**

<https://canvas.uw.edu/courses/1023396>

## **On GitHub**

<https://github.com/info498d-w16/>

Let your TA know your username!



# What is Android?

---

# Some Android History

**2003**      Founded by Android Inc.

**2005**      Acquired by Google

**2007**      Launch of Open Handset Alliance





# Some Android History

## 2008 First Android Device (HTC Dream / G1)

528Mhz ARM

256mb RAM

3.2" screen at 320x480px



## 2010 First Nexus Device (Nexus One)

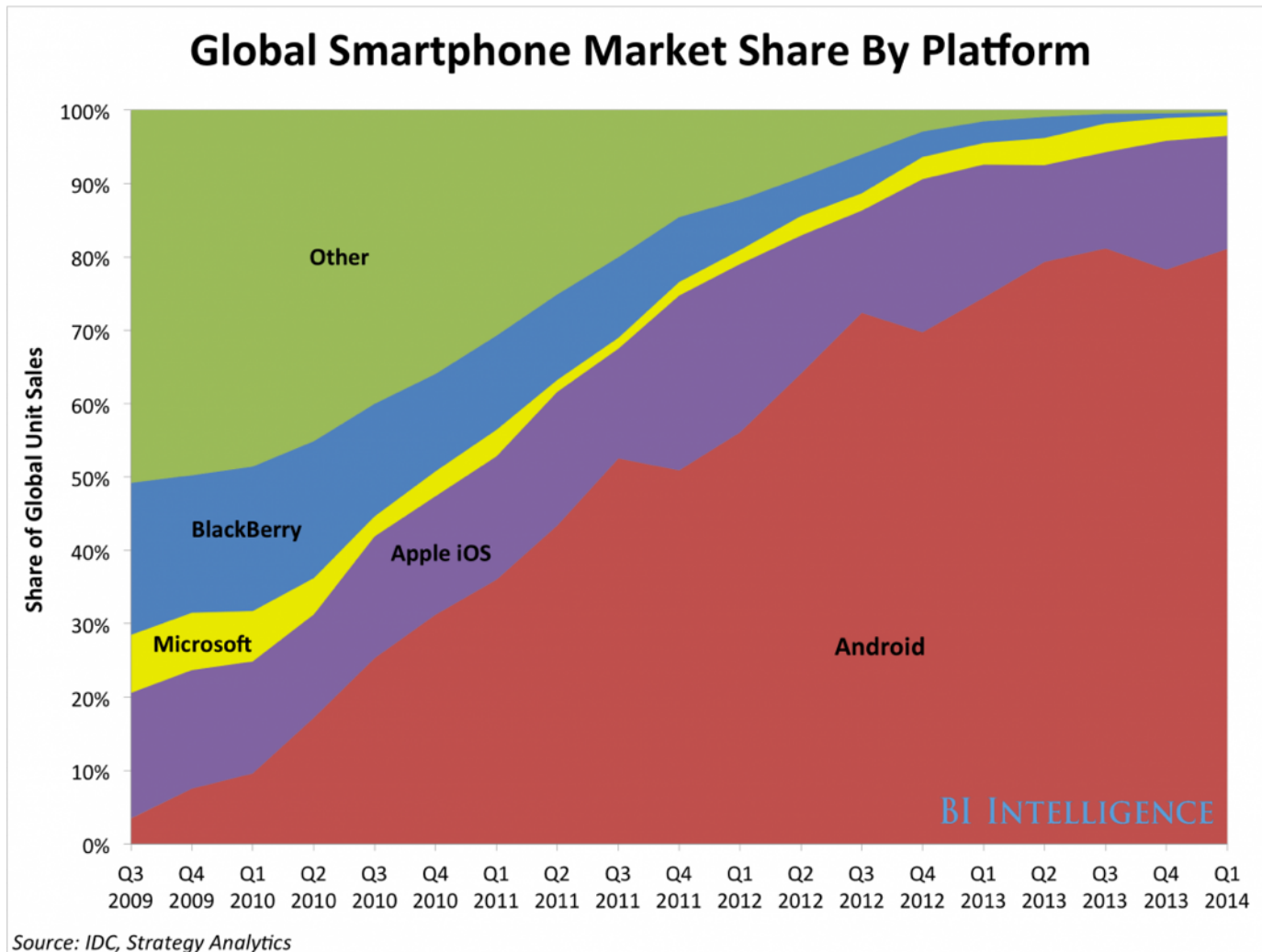
1Ghz Scorpion

512mb RAM

3.7" screen at 480x800px



# Popularity

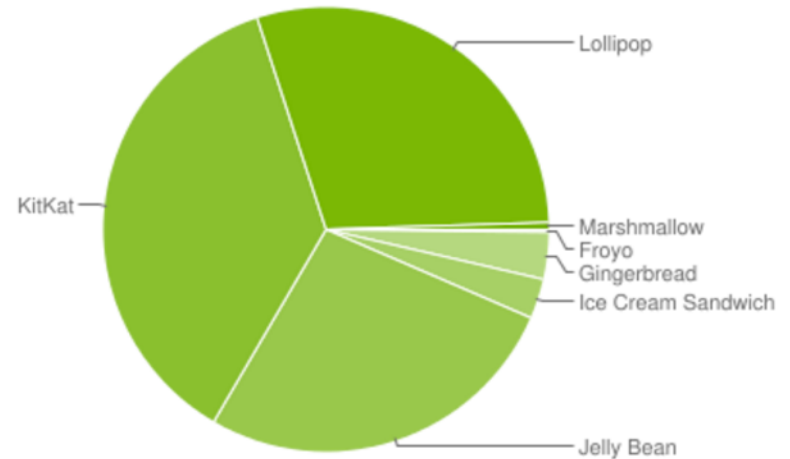


# Android Versions

Date	Version	Nickname	API Level
Sep 2008	1.0	Android	1
Apr 2009	1.5	Cupcake	3
Sep 2009	1.6	Donut	4
Oct 2009	2.0	Eclair	5
May 2010	2.2	Froyo	8
Dec 2010	2.3	Gingerbread	9
Feb 2011	3.0	Honeycomb	11
Oct 2011	4.0	Ice Cream Sandwich	14
July 2012	4.1	Jelly Bean	16
Oct 2013	4.4	KitKat	19
Nov 2014	5.0	Lollipop	21
Oct 2015	6.0	Marshmallow	23

# Android Versions

Version	Codename	API	Distribution
2.2	Froyo	8	0.2%
2.3.3 - 2.3.7	Gingerbread	10	3.4%
4.0.3 - 4.0.4	Ice Cream Sandwich	15	2.9%
4.1.x	Jelly Bean	16	10.0%
4.2.x		17	13.0%
4.3		18	3.9%
4.4	KitKat	19	36.6%
5.0	Lollipop	21	16.3%
5.1		22	13.2%
6.0	Marshmallow	23	0.5%



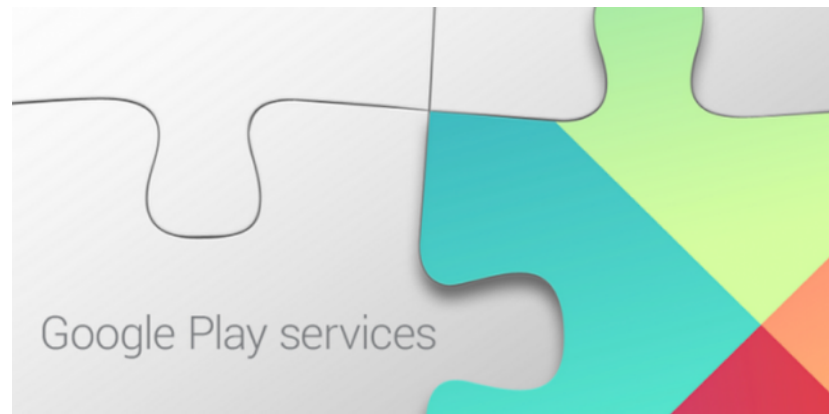
<http://developer.android.com/about/dashboards/>

# Upgrading Versions

Updates through purchasing new devices



Reliant on carriers... but looking to change that



# Legal Battles

ORACLE<sup>®</sup> vs. Google

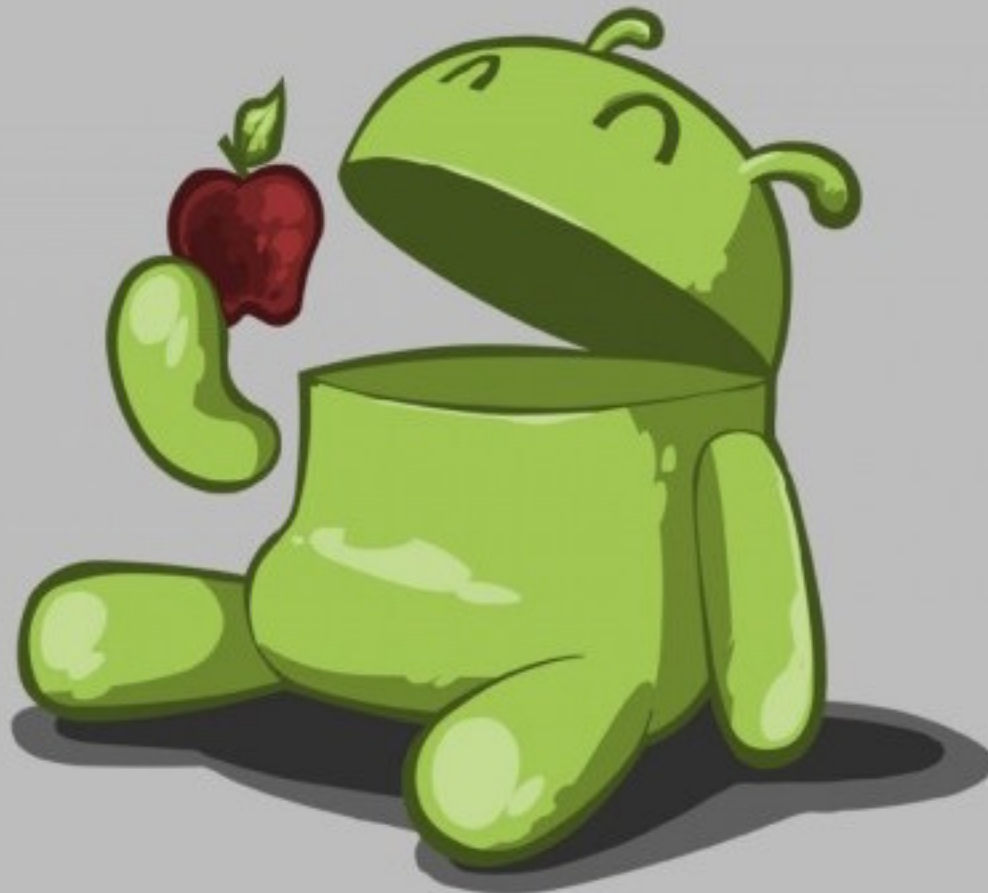
**Claim:** Java API is copyrighted, so Google violated that by using it in Android (see [EFF](#))

**VB**

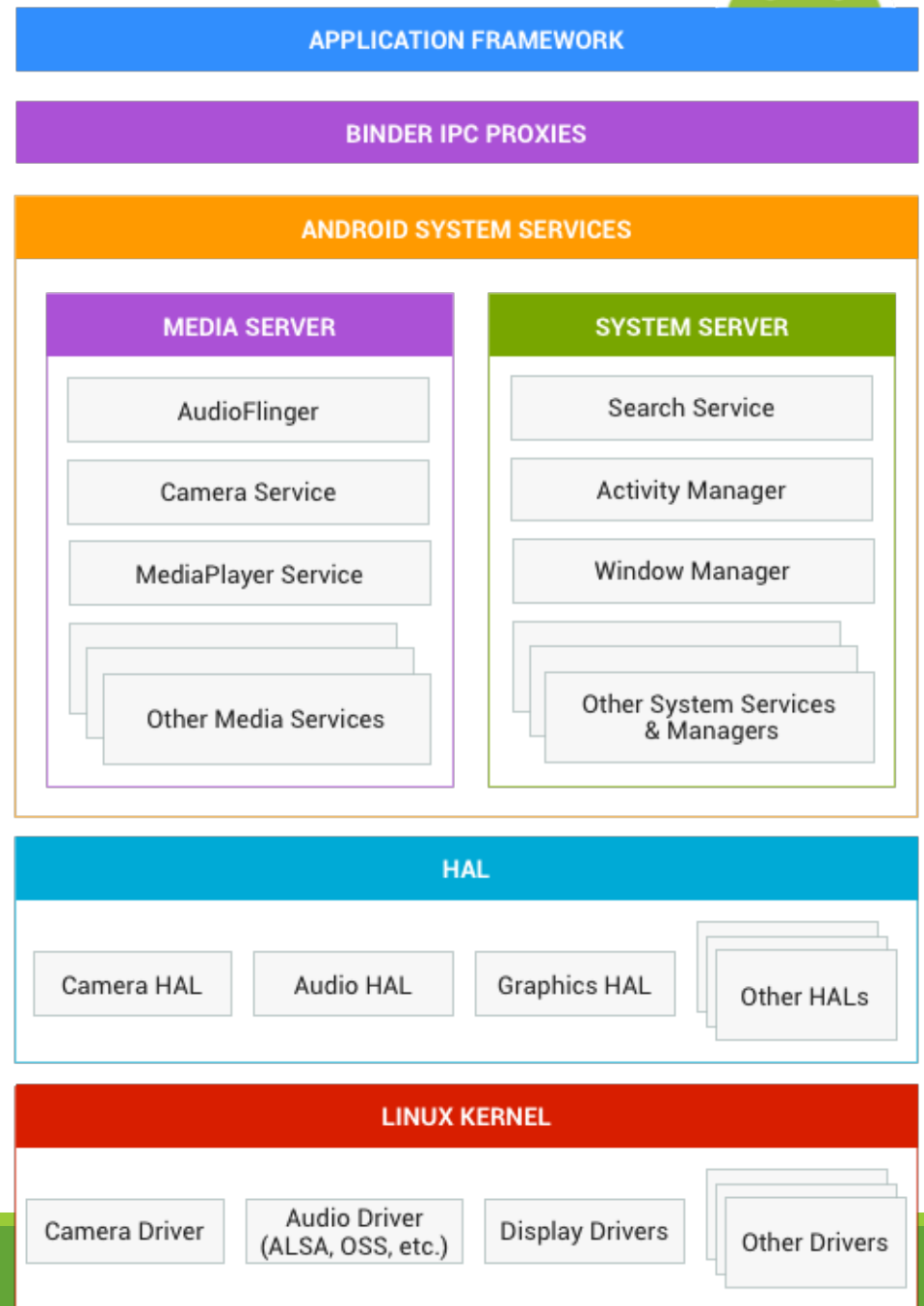


Google confirms next Android version will use Oracle's open-source OpenJDK for Java APIs

EMIL PROTALINSKI DECEMBER 29, 2015 1:31 PM



# Some Architecture





# Building Apps: Dalvik

## Dalvik Pre-Lollipop (5.0)



- Register-based architecture
- bytecode stored in DEX or ODEX files
- **Just-In-Time** compilation to native code

# Building Apps: ART

## **Android RunTime (ART)** Post-Lollipop

- Compile into native code on installation ("Ahead Of Time")
- Accepts DEX bytecode (for backwards compatibility)

<https://source.android.com/devices/tech/dalvik/>

# Packaging Apps: **APKs**

- Basically **.zip** files
- "side-load" or cryptographically sign for upload to Play Store



# Building Process

1. Generate Java source files (e.g. from XML)
2. Compile Java source into JVM bytecode
3. "dex" JVM bytecode into Dalvik bytecode
4. Pack with graphics and assets into **.apk**
5. Cryptographically sign **.apk**
6. ...
7. Profit!

# Development Hardware

- Windows, Mac, or Linux (because Java!)
  - emulator sucks on Windows
  - use the Intel HAXM for acceleration!
- Physical Android Device
  - turn on developer options!

# Development Software

- Java SDK
- Gradle or Apache ANT
- Android Studio IDE
- Android SDK (command-line tools)
  - **android** manage SDK/AVDs
  - **emulator** run the emulator
  - **adb** "android debugging bridge" connect to devices



# Android Studio

<https://github.com/info498d-w16/lectures>

