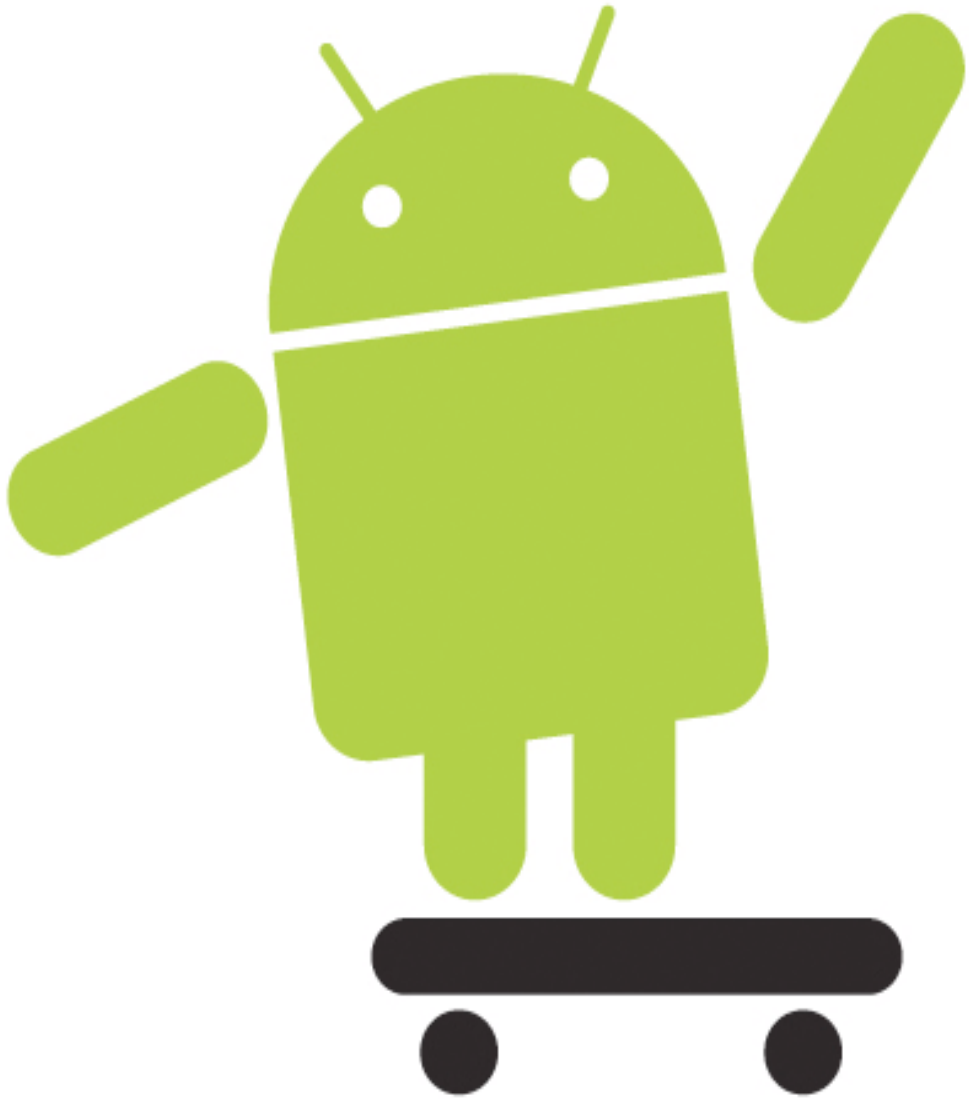


marakana

Android: A 9,000-foot Overview

marakana

About Marko Gargenta

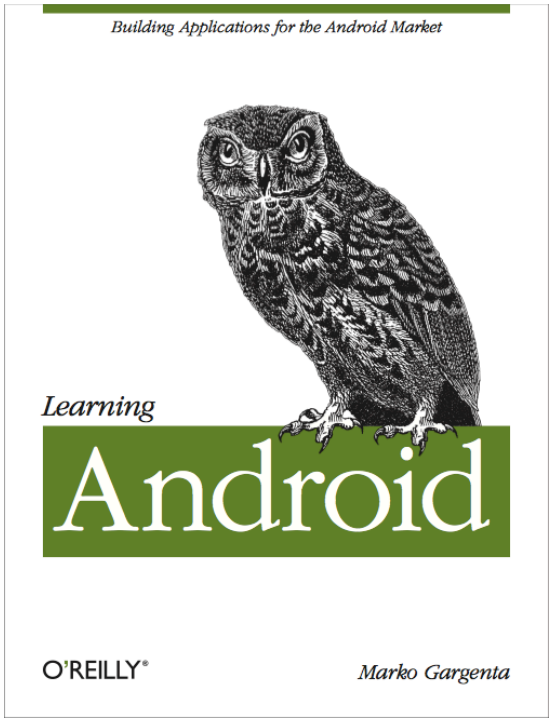
Developed Android Bootcamp for Marakana.

Trained over 1,000 developers on Android.

Clients include Qualcomm, Sony-Ericsson, Motorola, Texas Instruments, Cisco, Sharp, DoD.

Author of upcoming *Learning Android* by O'Reilly.

Spoke at OSCON, ACM, IEEE, SDC. Organizes SFAndroid.org

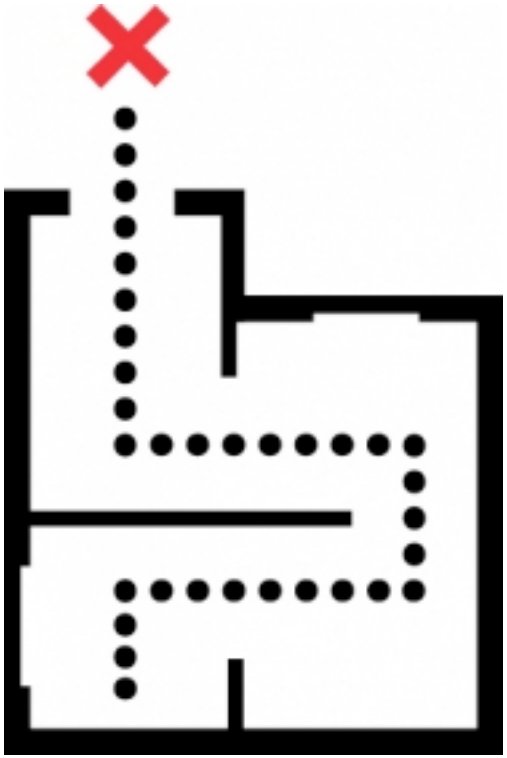


	2010
Trips	19
Days	71
Distance	109,976 mi
Cities	17
Countries	8

marakana

Agenda

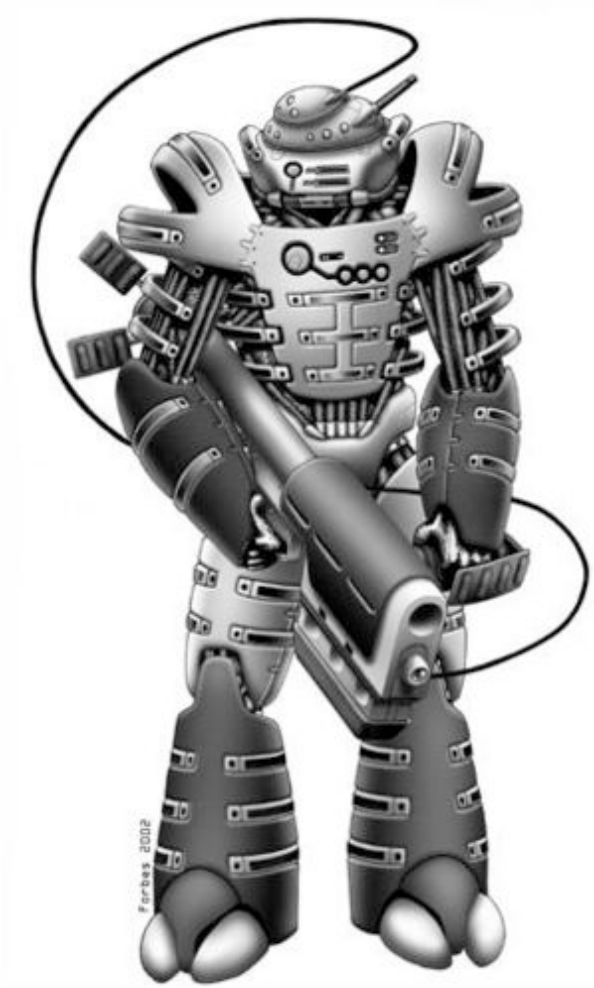
- Market Space
- The Stack
- Android SDK
- Hello World!
- Main Building Blocks
- Android User Interface
- Operating System Features
- Debugging
- Summary



marakana

History

2005	Google buys Android, Inc. Work on Dalvik starts
2007	Open Handset Alliance announced Early Software Development Kit
2008	HTC G1 Announced SDK 1.0 Released
2009	G2 + 20 other phones released Cupcake, Donut, Éclair
2010	Zillion devices FroYo, Gingerbread, JIT



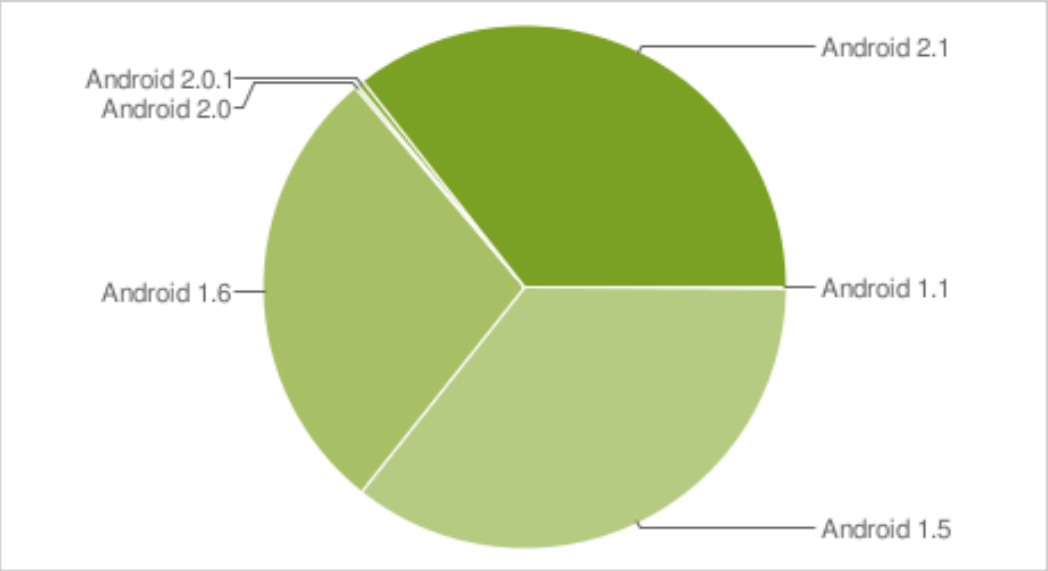
marakana

Versions

Version	API Level	Nickname
Android 1.0	1	
Android 1.1	2	
Android 1.5	3	
Android 1.6	4	
Android 2.0	5	
Android 2.01	6	
Android 2.1	7	
Android 2.2	8	

marakana

Version Distribution



Android Platform	Percent of Devices
Android 1.1	0.1%
Android 1.5	34.1%
Android 1.6	28.0%
Android 2.0	0.2%
Android 2.0.1	0.4%
Android 2.1	37.2%

Data collected during two weeks ending on May 17, 2010

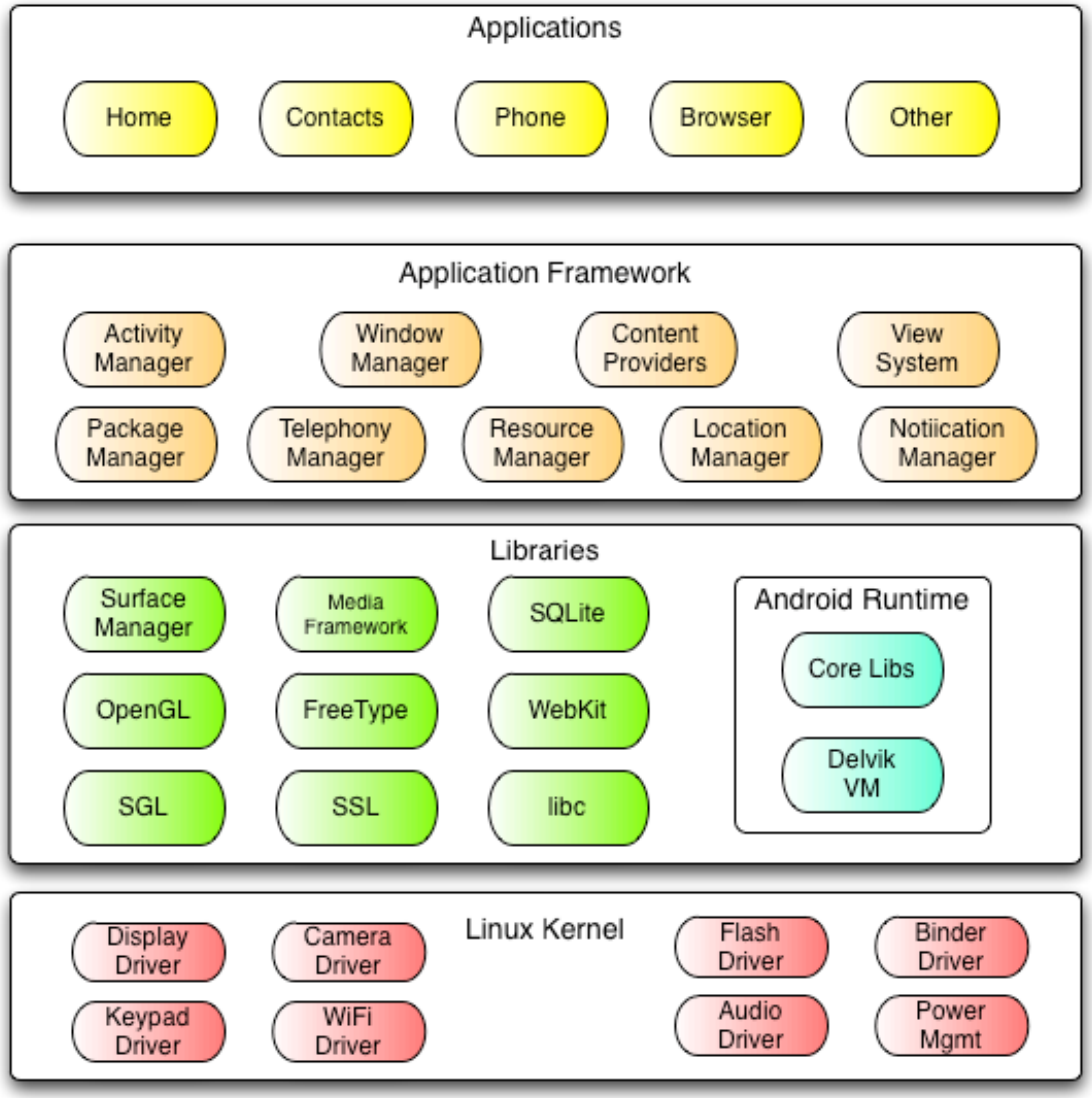
Source: Android.com

marakana

ANDROID STACK

marakana

The Stack



Linux Kernel

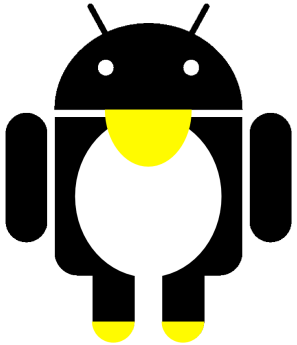
Android runs on Linux.

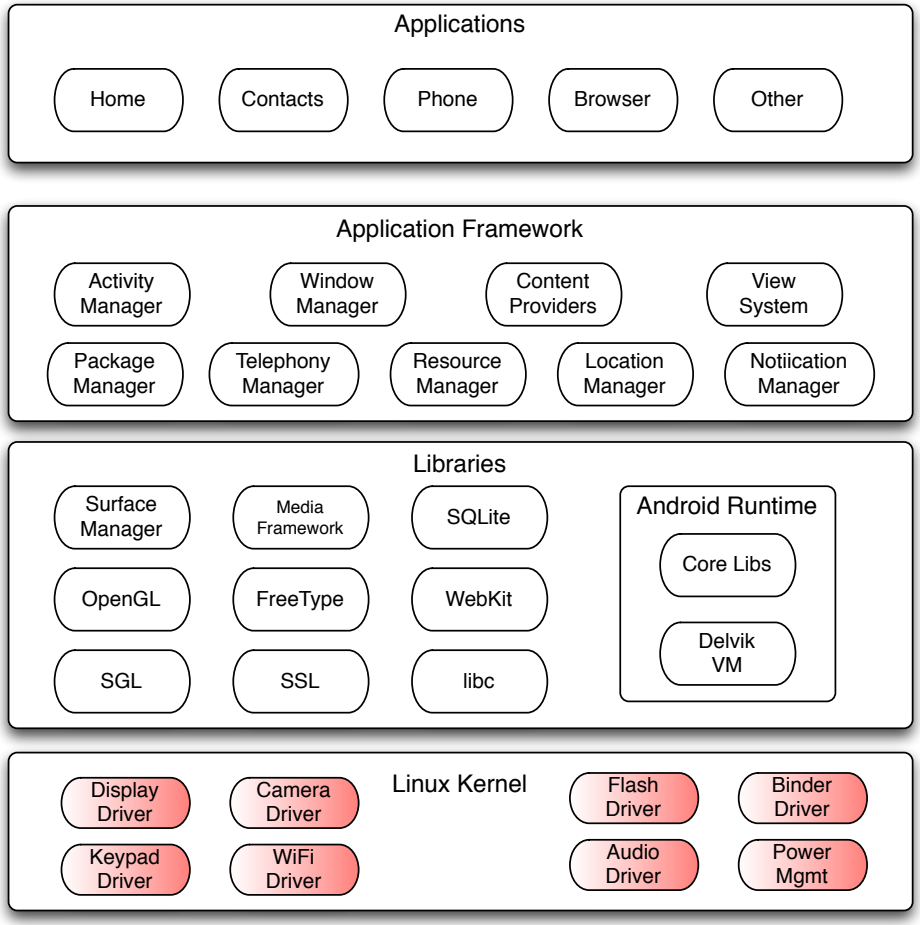
Linux provides as well as:

- Hardware abstraction layer
- Memory management
- Process management
- Networking

Users never see Linux sub system

The adb shell command opens Linux shell





marakana

Native Libraries

Bionic, a super fast and small license-friendly libc library optimized for embedded use

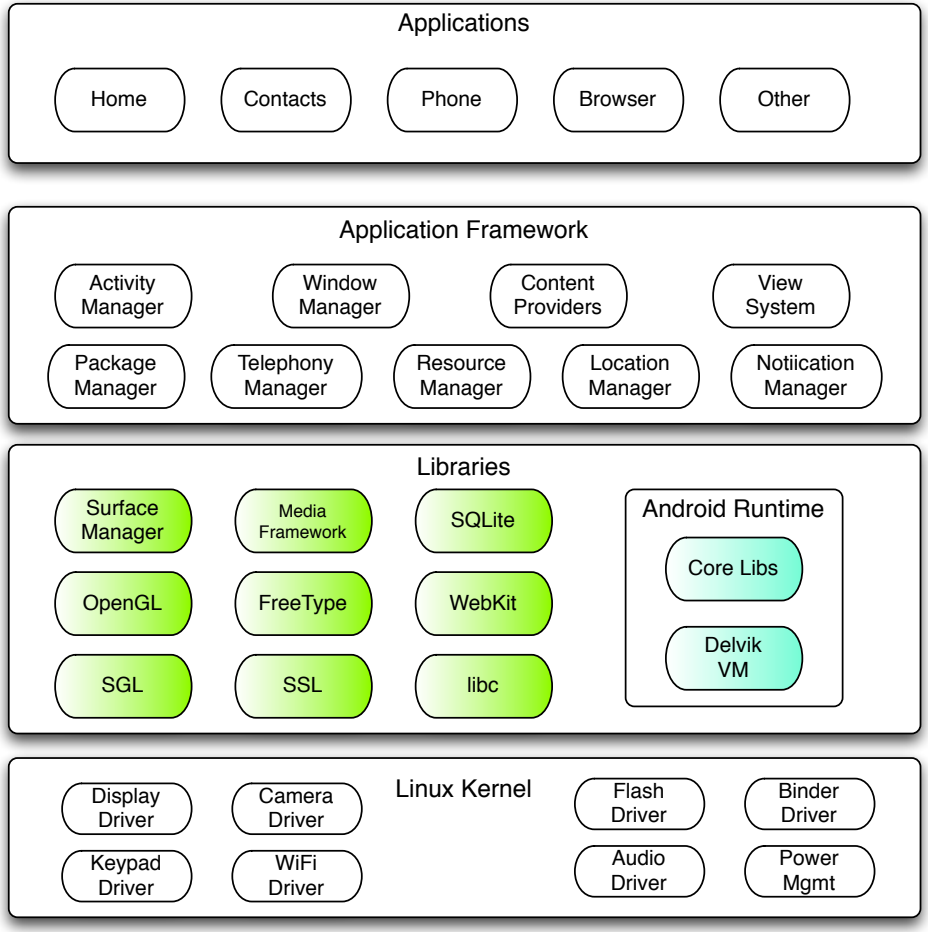
Surface Manager for composing window manager with off-screen buffering

2D and 3D graphics hardware support or software simulation

Media codecs offer support for major audio/video codecs

SQLite database

WebKit library for fast HTML rendering



Dalvik

Dalvik VM is Google's implementation of Java VM

Optimized for mobile devices

Key Dalvik differences:

- Register-based versus stack-based VM
- Dalvik runs .dex files
- More efficient and compact implementation
- Different set of Java libraries than SDK

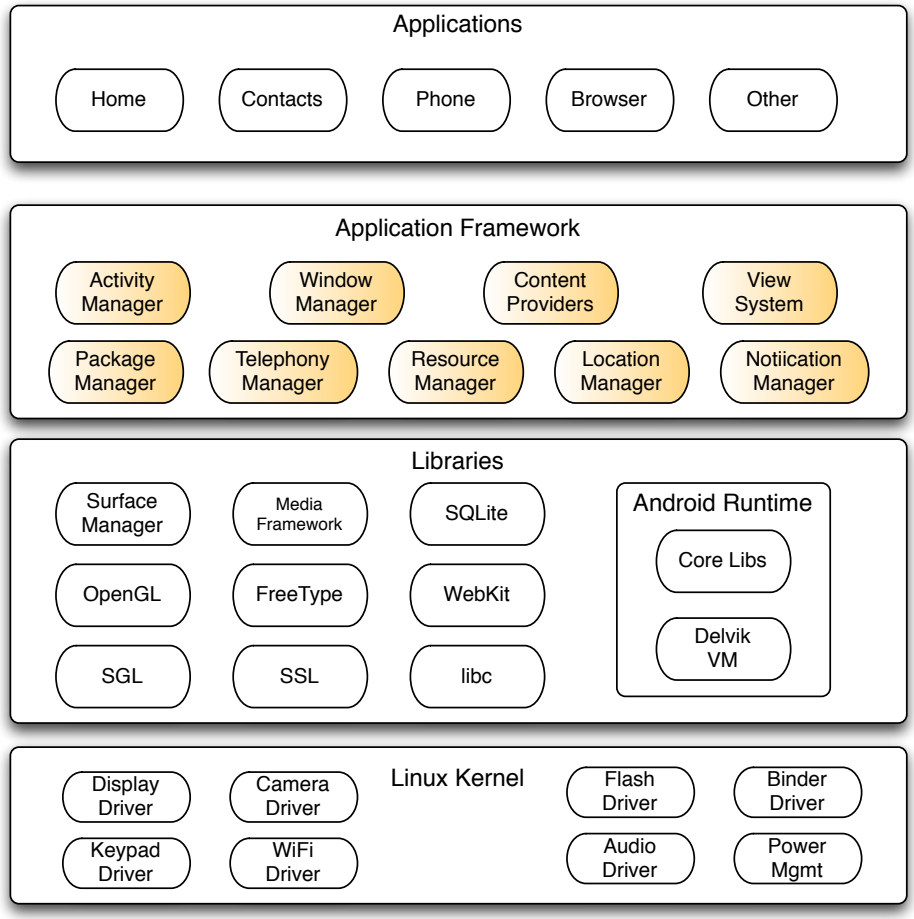
marakana

Application Framework

The rich set of system services wrapped in an intuitive Java API.

This ecosystem that developers can easily tap into is what makes writing apps for Android easy.

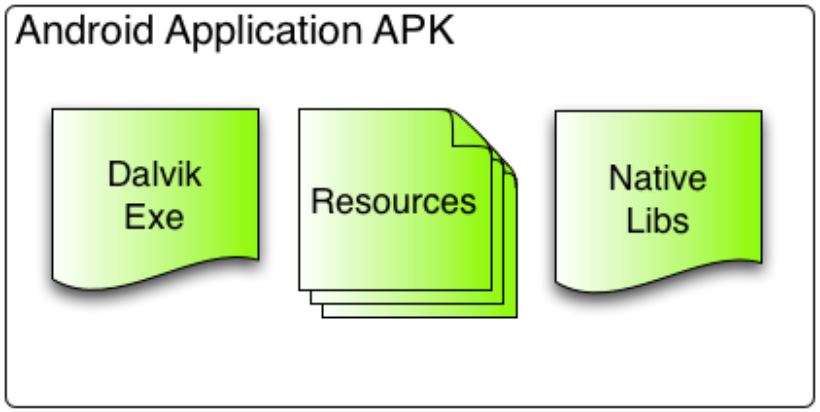
Location, web, telephony, WiFi, Bluetooth, notifications, media, camera, just to name a few.



marakana

Applications

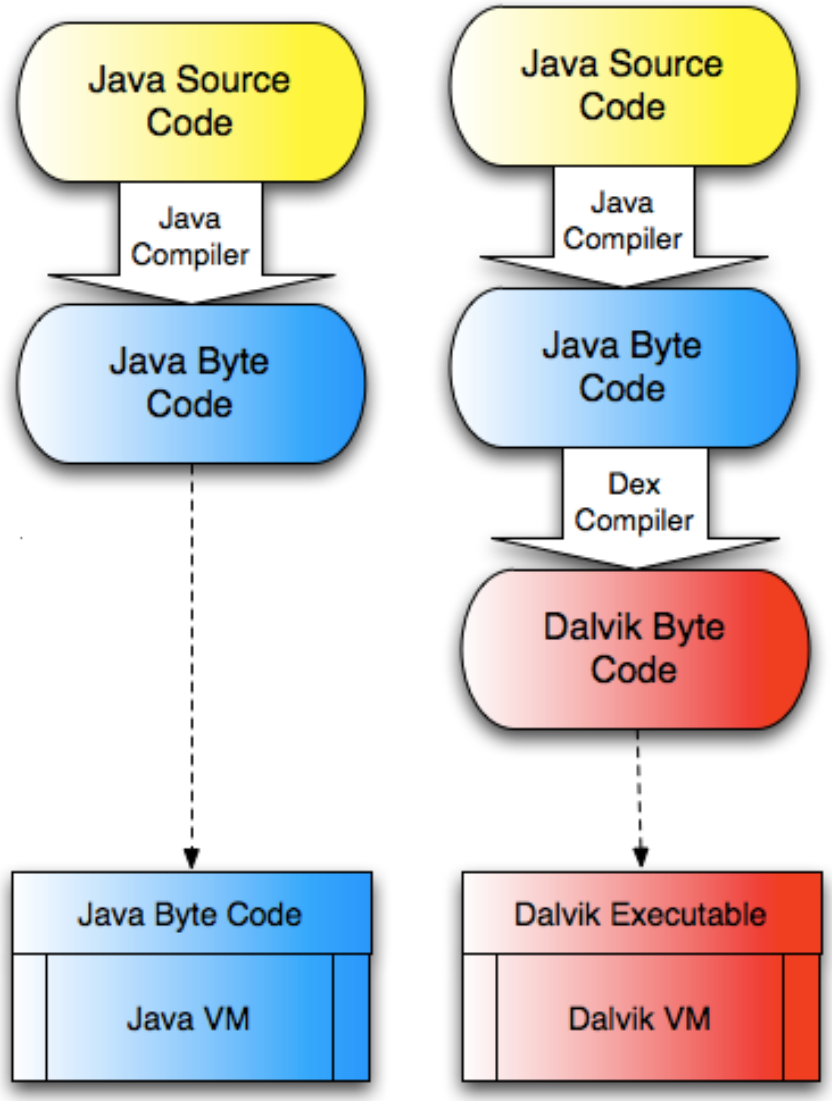
Dalvik Executable + Resources = APK
Must be signed (but debug key is okay for development)
Many markets with different policies



marakana

Android and Java

Android Java =
Java SE –
AWT/Swing +
Android API



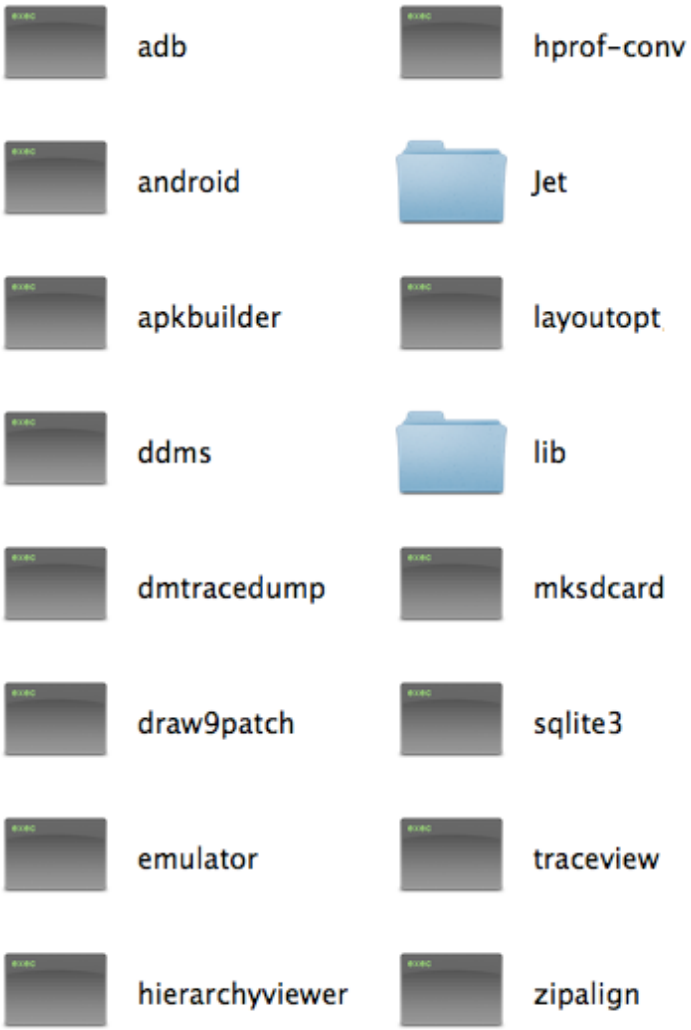
marakana

Android SDK - What's in the box

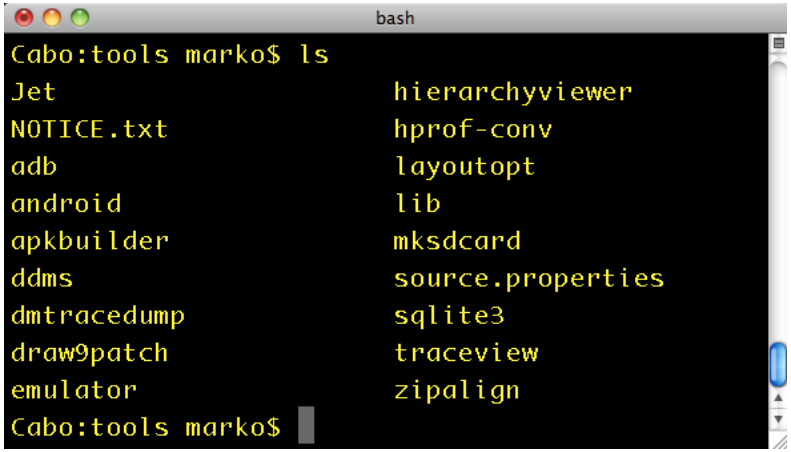
SDK

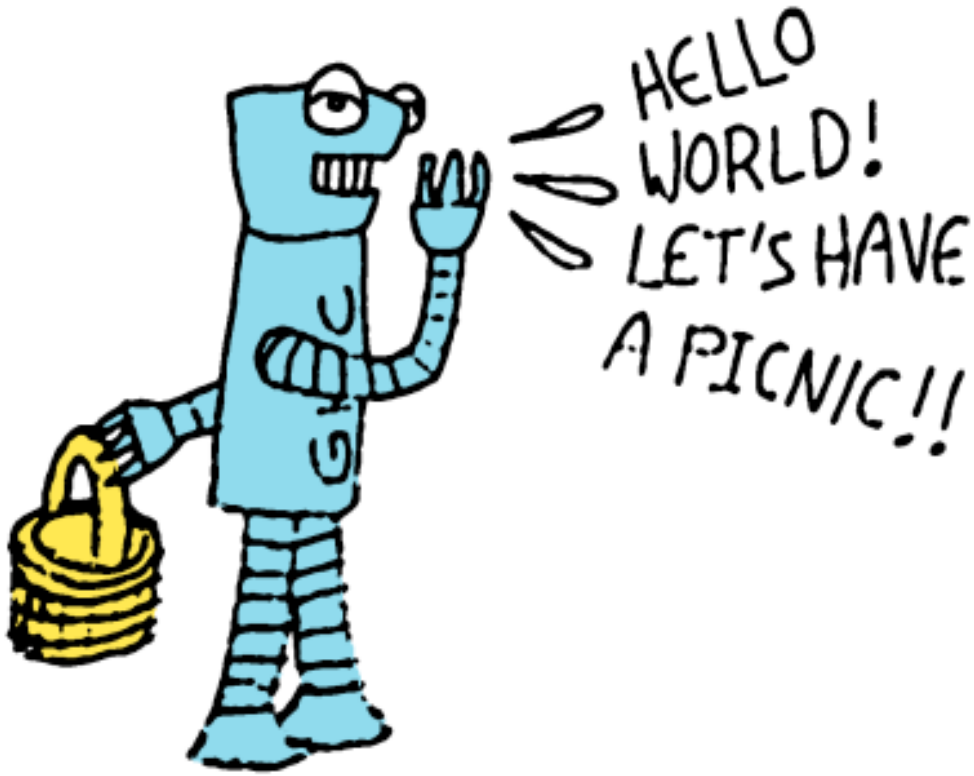
Tools
Docs
Platforms
    Data
    Skins
    Images
    Samples
Add-ons
    Google Maps

The Tools



Tools are important part of the SDK. They are available via Eclipse plugin as well as command line shell.





HELLO WORLD!

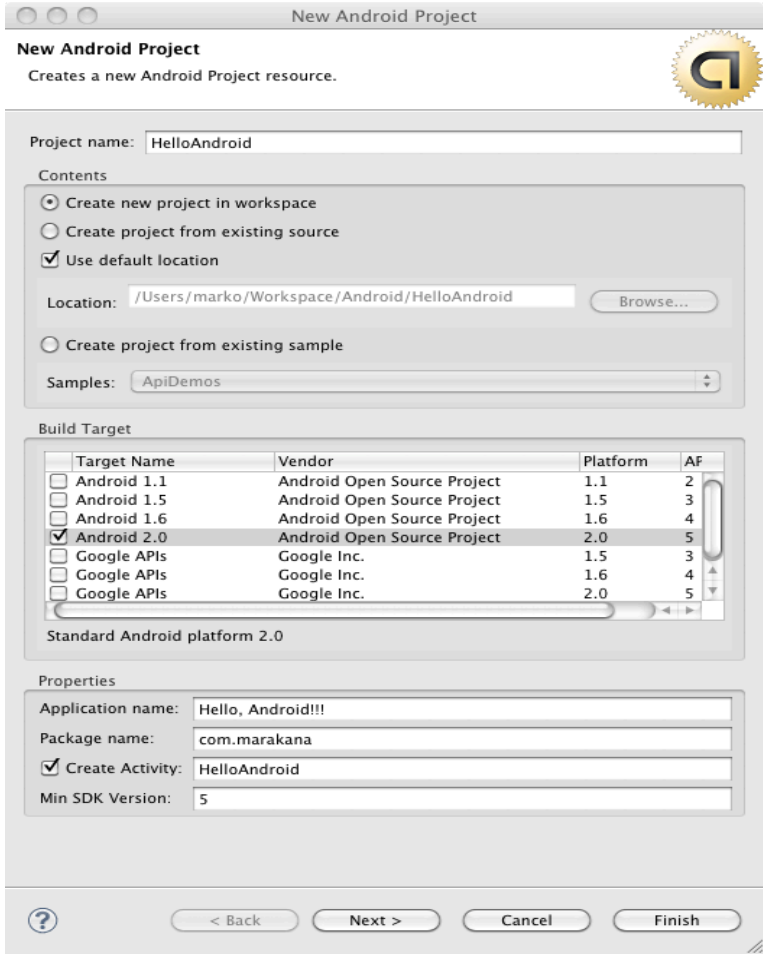
marakana

Create New Project

Use the Eclipse tool to create a new Android project.

Here are some key constructs:

Project	Eclipse construct
Target	minimum to run
App name	whatever
Package	Java package
Activity	Java class



The Manifest File

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.marakana"
    android:versionCode="1"
    android:versionName="1.0">
  <application android:icon="@drawable/icon"
        android:label="@string/app_name">
    <activity android:name=".HelloAndroid"
          android:label="@string/app_name">
      <intent-filter>
        <action android:name="android.intent.action.MAIN" />
        <category android:name="android.intent.category.LAUNCHER" />
      </intent-filter>
    </activity>
  </application>
  <uses-sdk android:minSdkVersion="5" />
</manifest>
```

The Layout Resource

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    >
<TextView
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:text="@string/hello"
    />
</LinearLayout>
```

marakana

The Java File

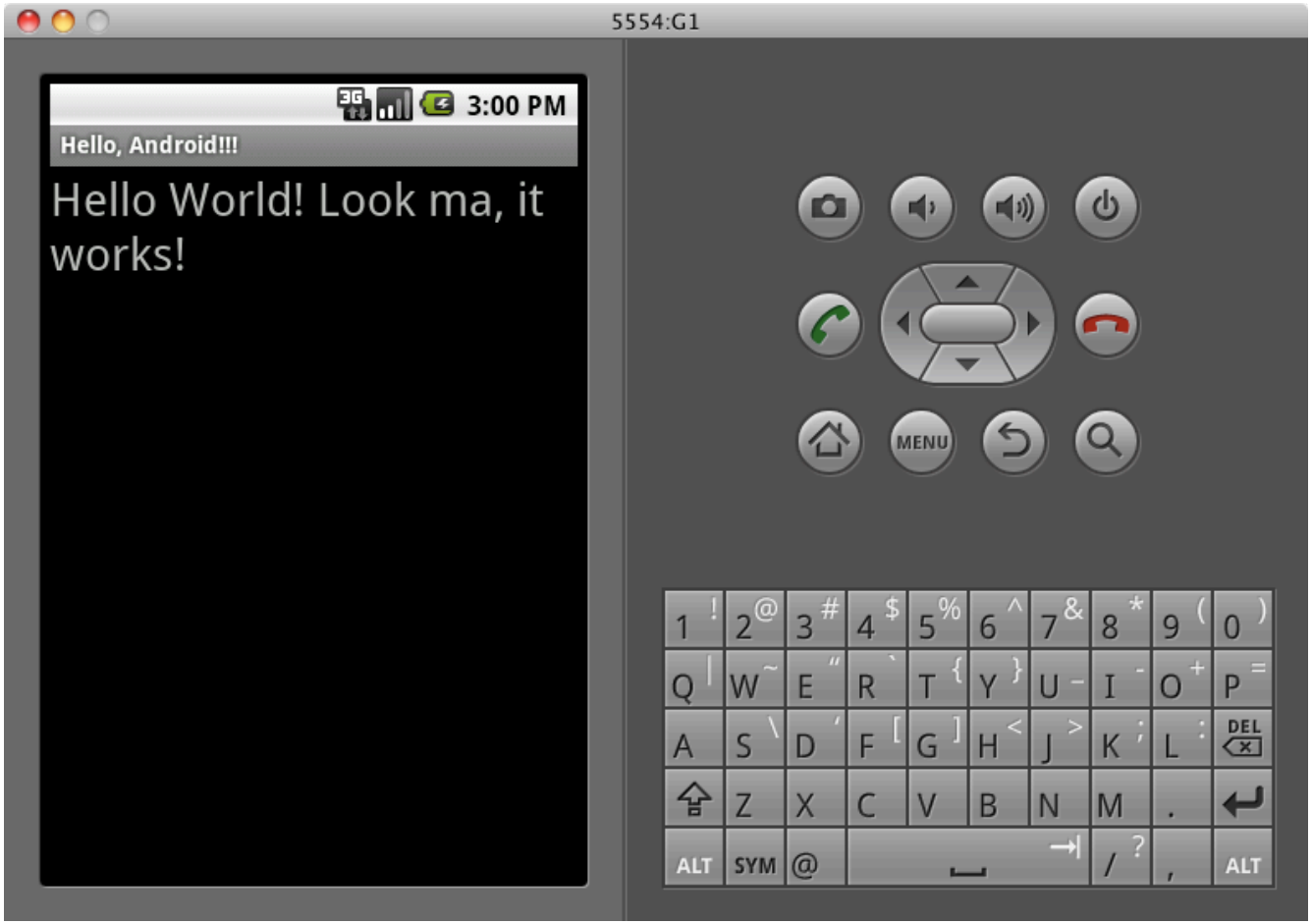
```
package com.marakana;

import android.app.Activity;
import android.os.Bundle;

public class HelloAndroid extends Activity {
    /** Called when the activity is first created. */
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);
    }
}
```

marakana

Running on Emulator



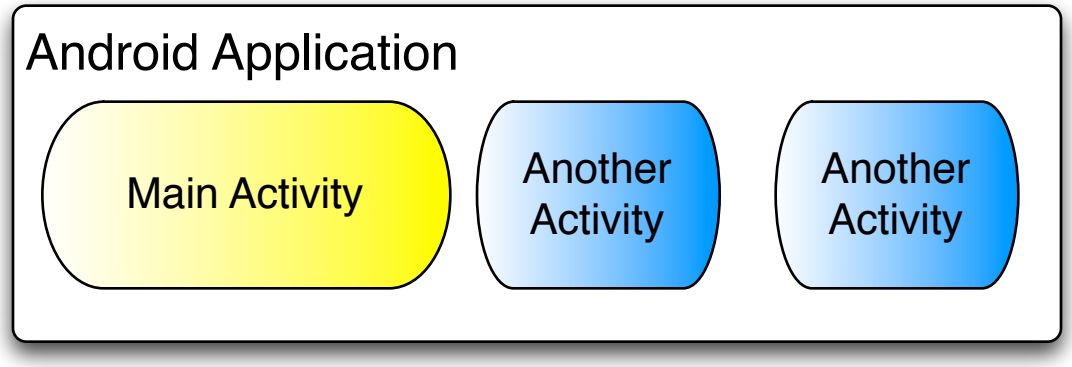
marakana

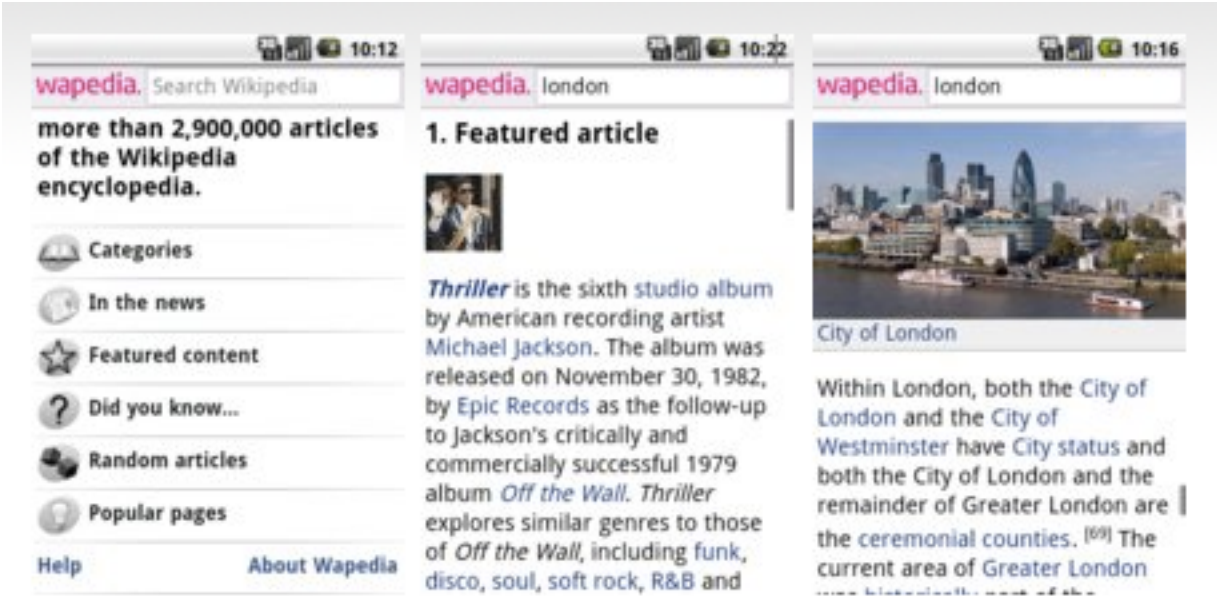
MAIN BUILDING BLOCKS

marakana

Activities

Activity is to an application what a web page is to a website. Sort of.

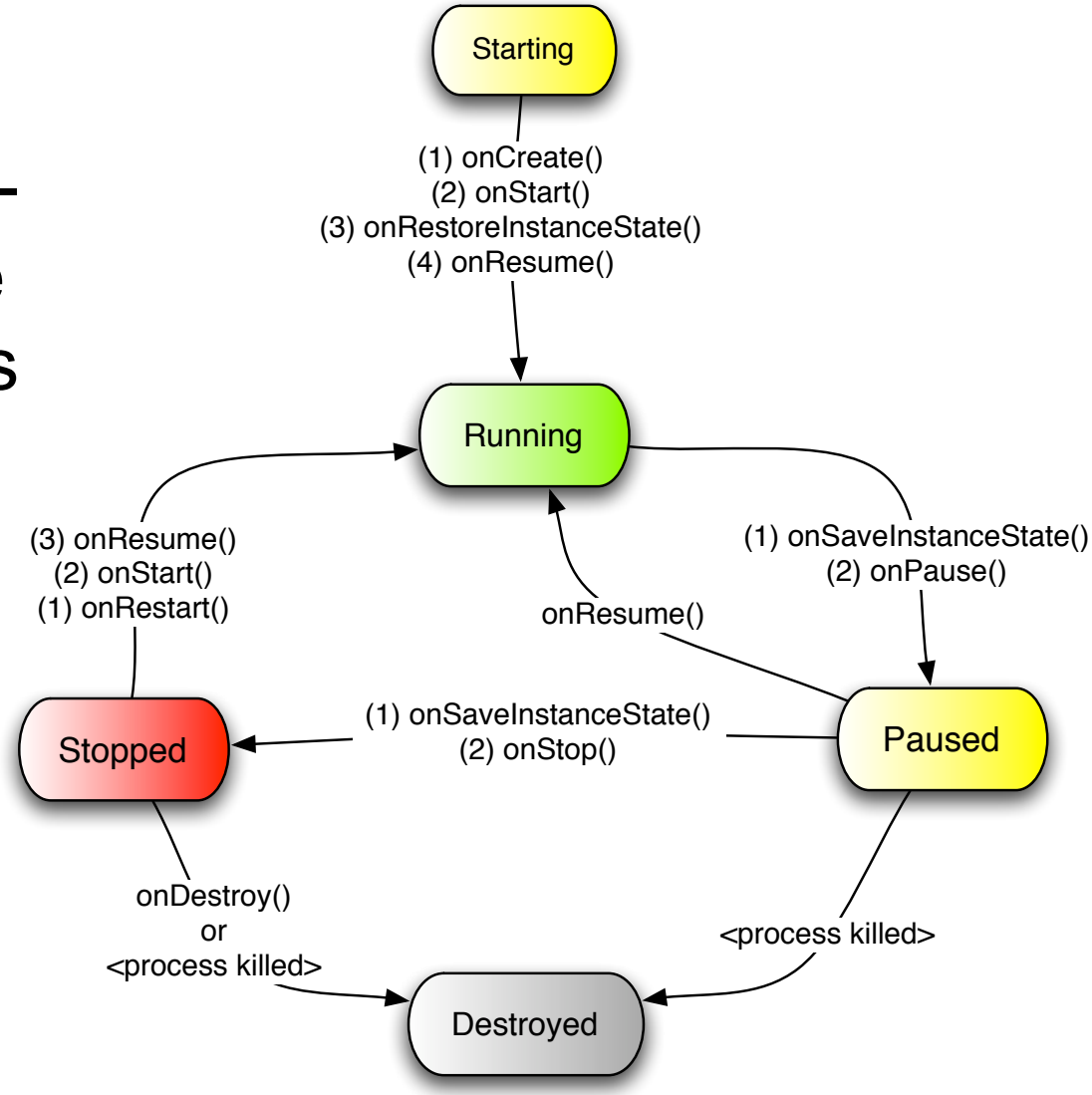




marakana

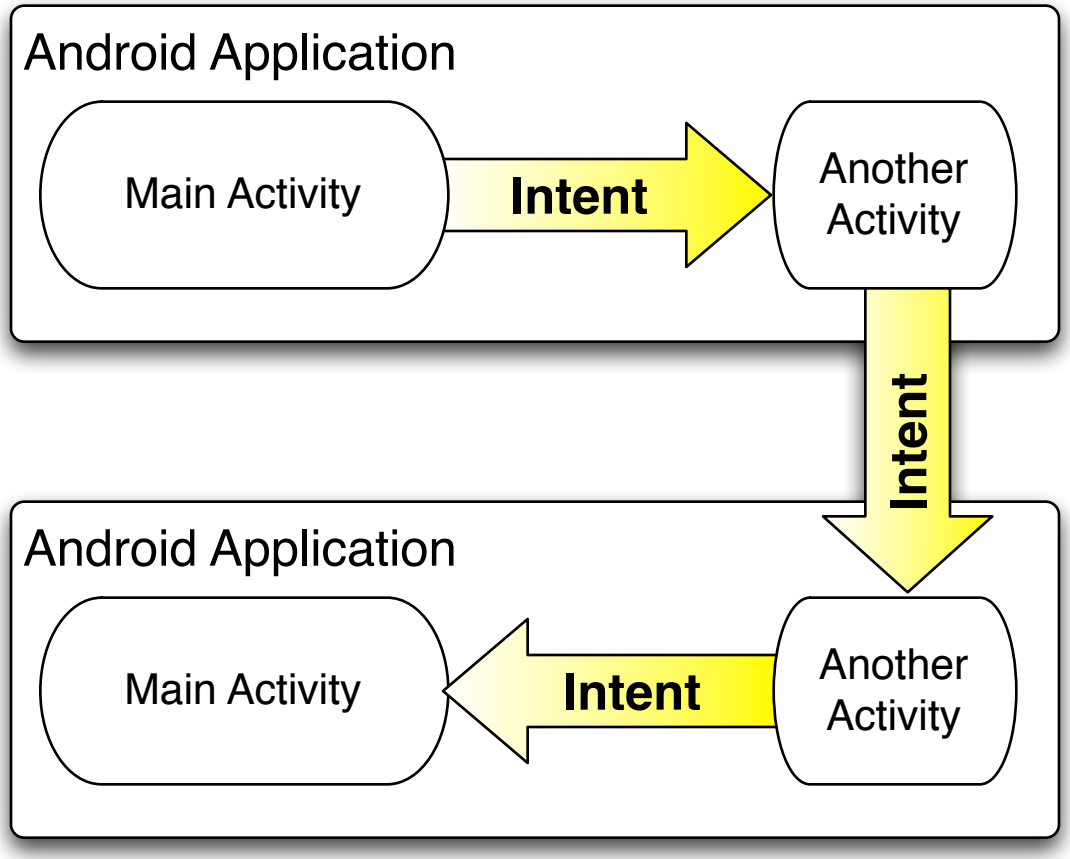
Activity Lifecycle

Activities have a well-defined lifecycle. The Android OS manages your activity by changing its state. You fill in the blanks.



Intents

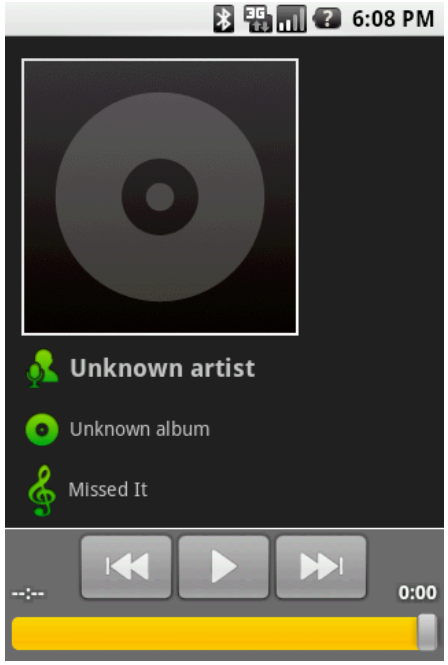
Intents are to Android apps what hyperlinks are to websites. They can be implicit and explicit. Sort of like absolute and relative links.

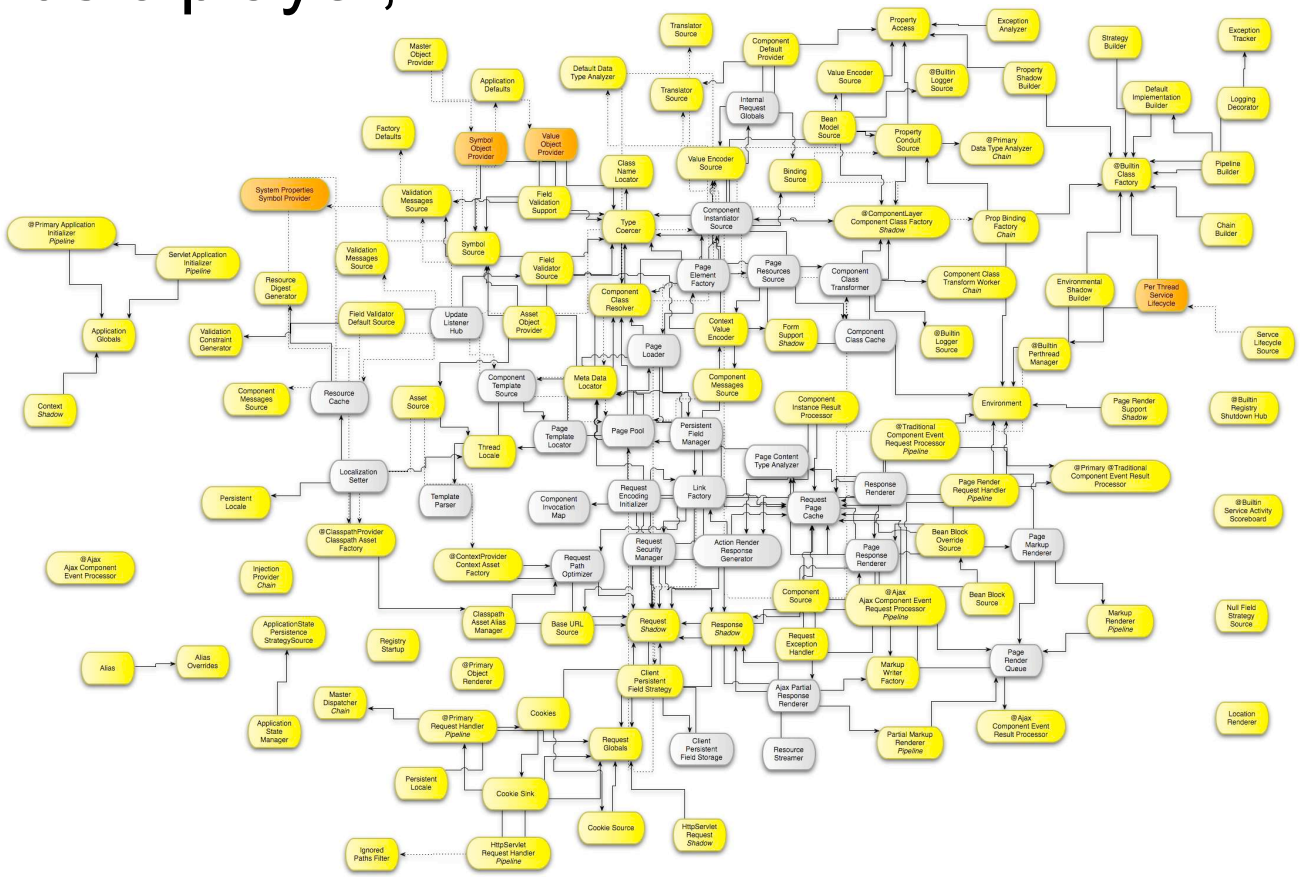


marakana

Services

A service is something that can be started and stopped. It doesn't have UI. It is typically managed by an activity. Music player, for example

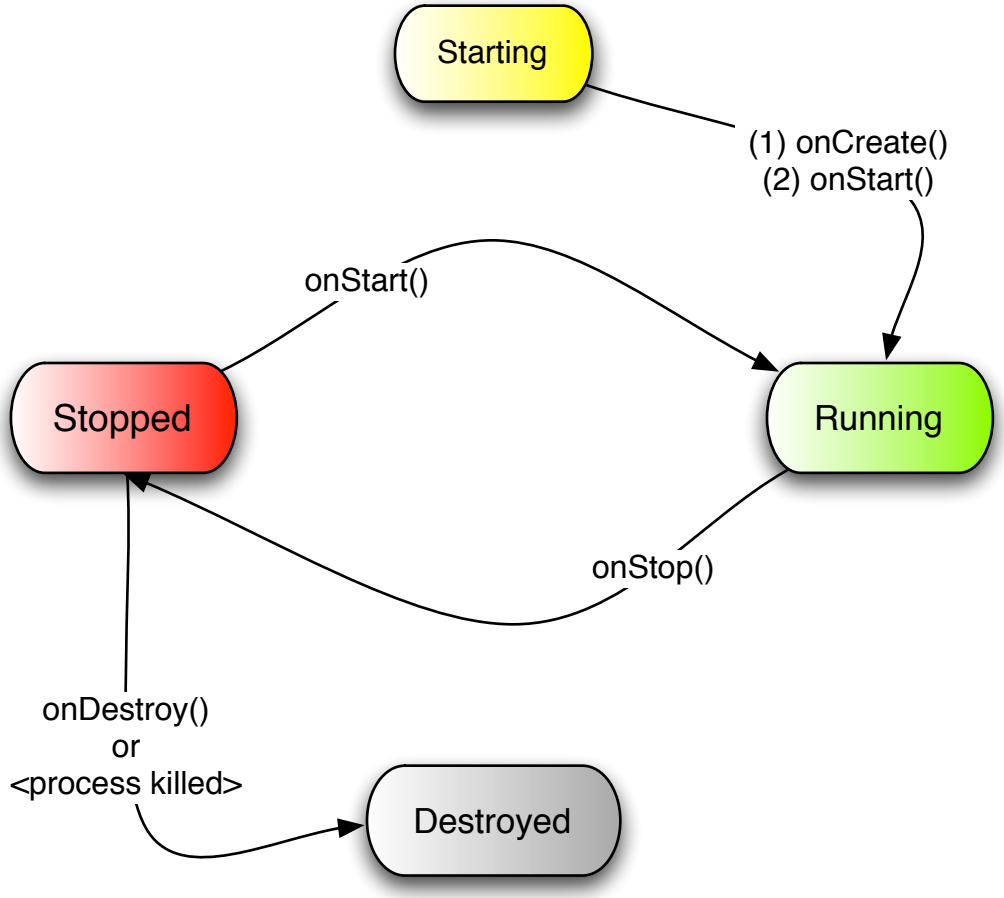




marakana

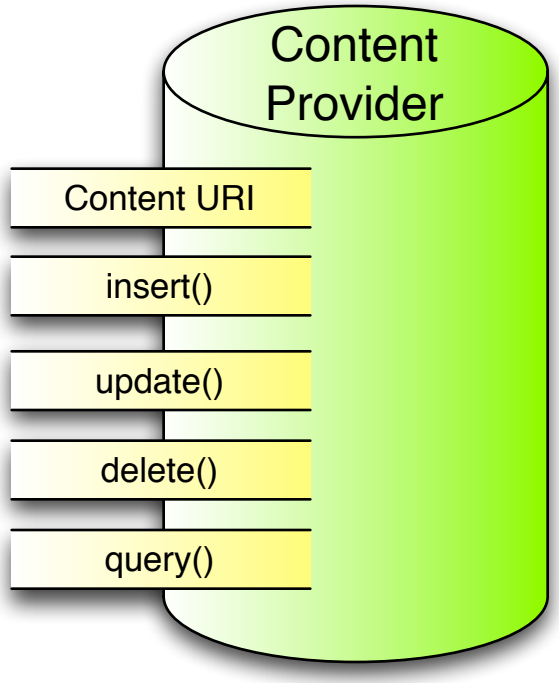
Service Lifecycle

Service also has a lifecycle, but it's much simpler than activity's. An activity typically starts and stops a service to do some work for it in the background. Such as play music, check for new tweets, etc.

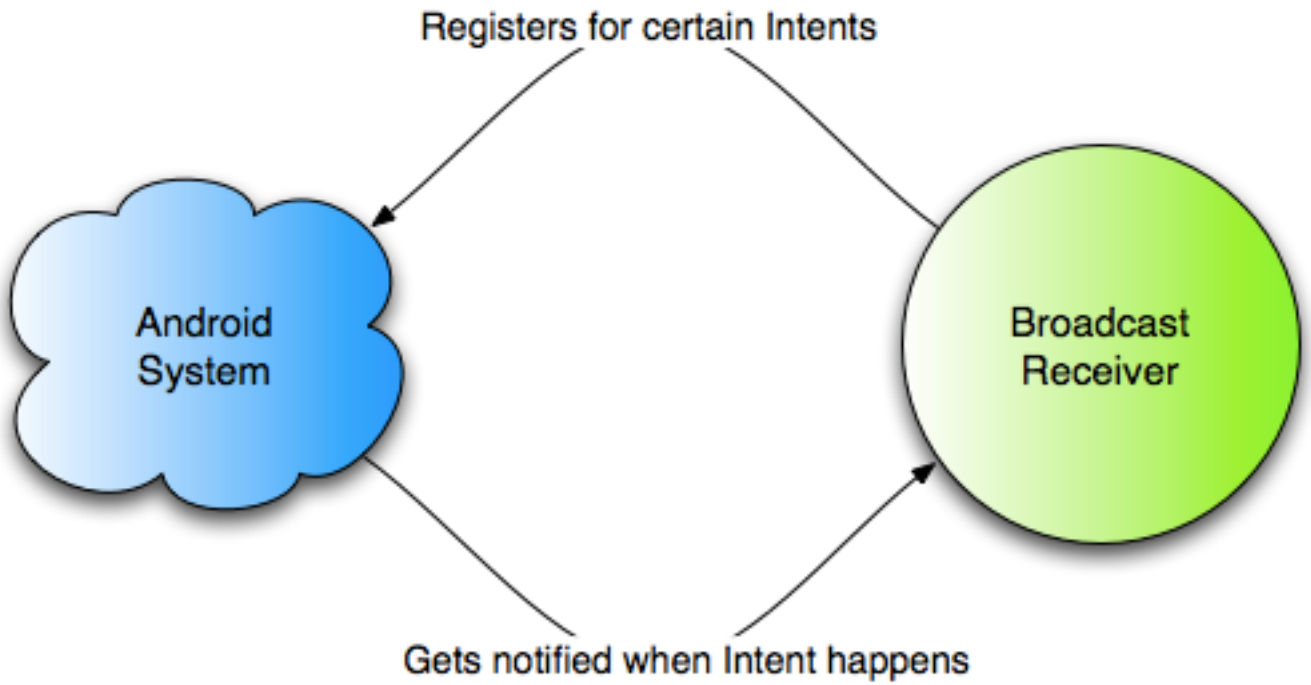


Content Providers

Content Providers share content with applications across application boundaries.
Examples of built-in Content Providers are: Contacts, MediaStore, Settings and more.



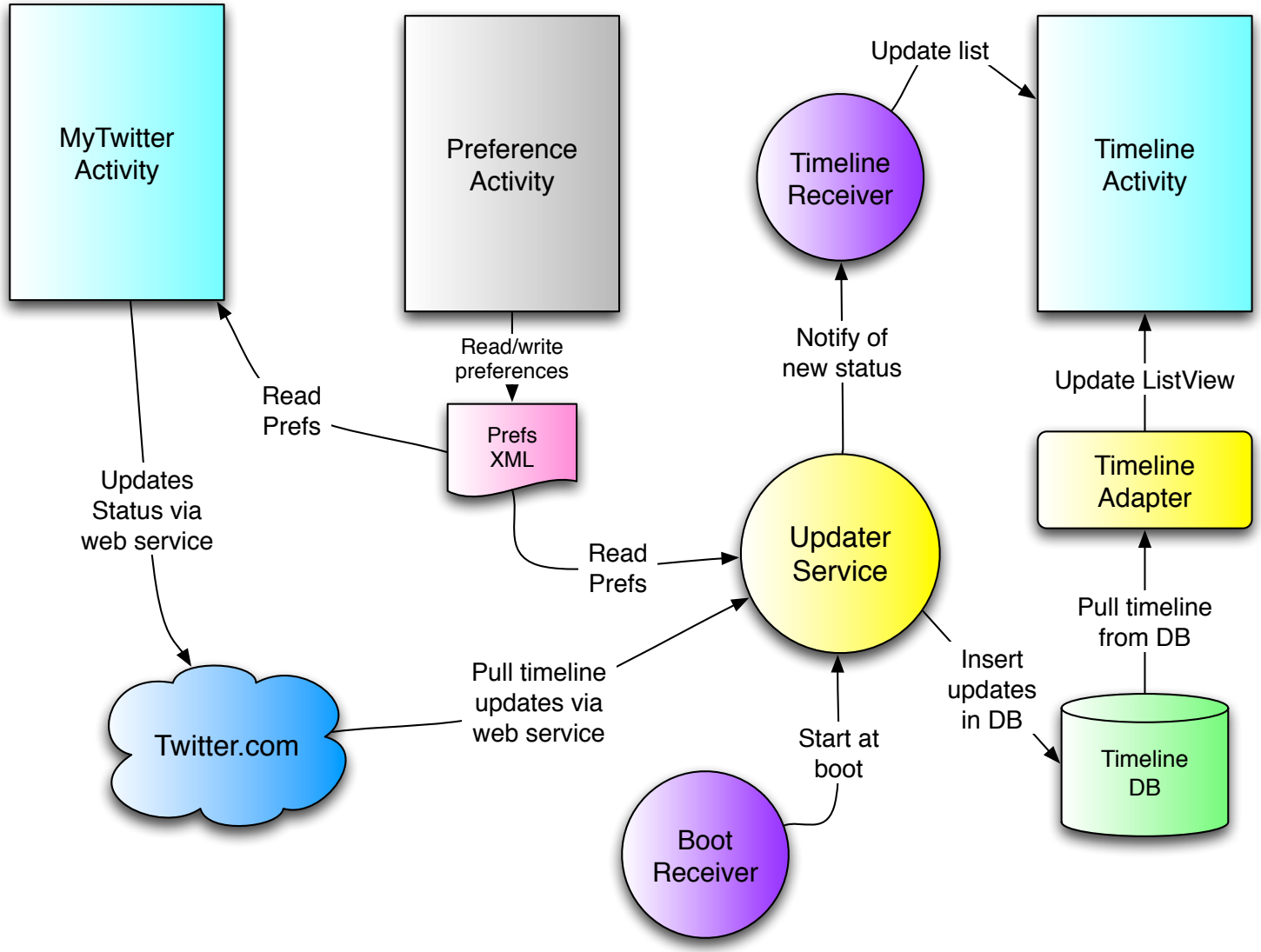
Broadcast Receivers



An Intent-based publish-subscribe mechanism. Great for listening system events such as SMS messages.

marakana

MyTwitter – A Real World App



marakana

ANDROID USER INTERFACE

marakana

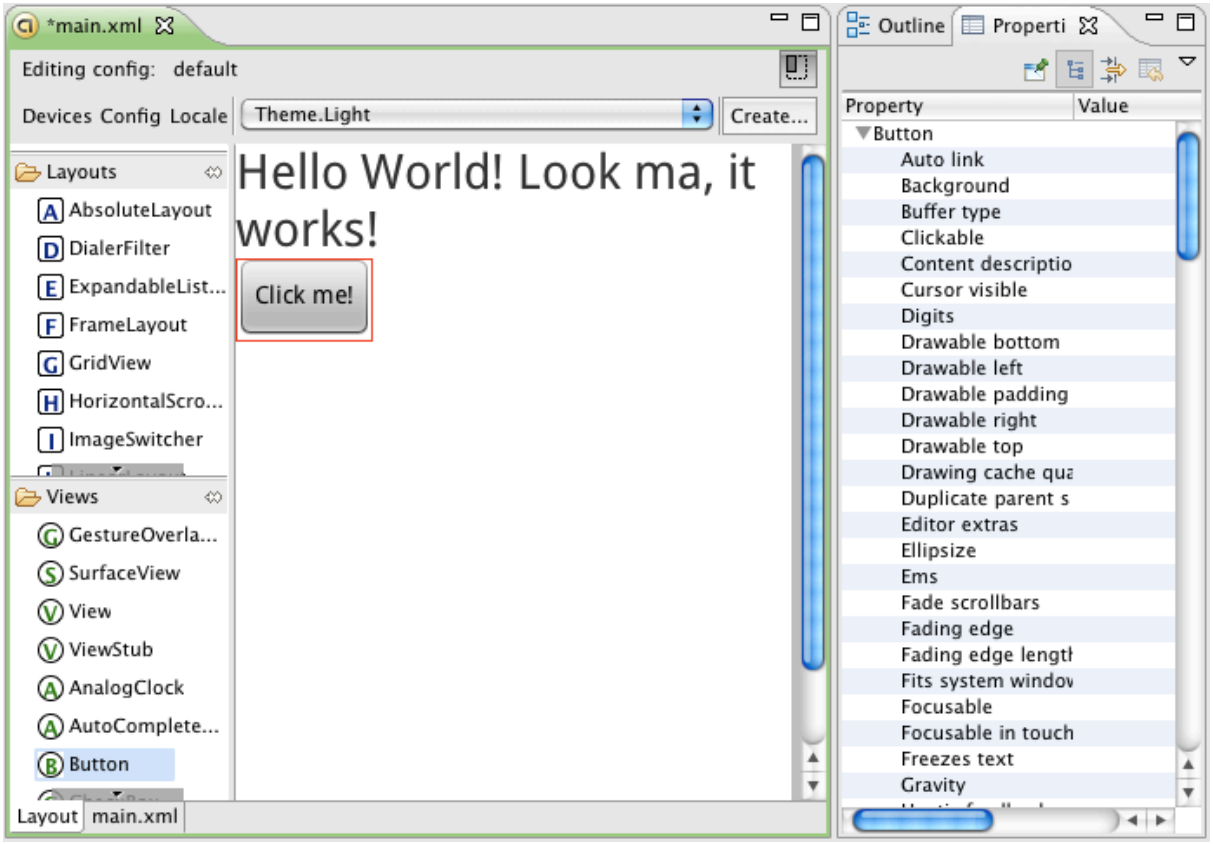
Two UI Approaches

Procedural	Declarative
You write Java code Similar to Swing or AWT	You write XML code Similar to HTML of a web page

You can mix and match both styles. Declarative is preferred: easier and more tools

marakana

XML-Based User Interface



Use WYSIWYG tools to build powerful XML-based UI.
Easily customize it from Java. Separate concerns.

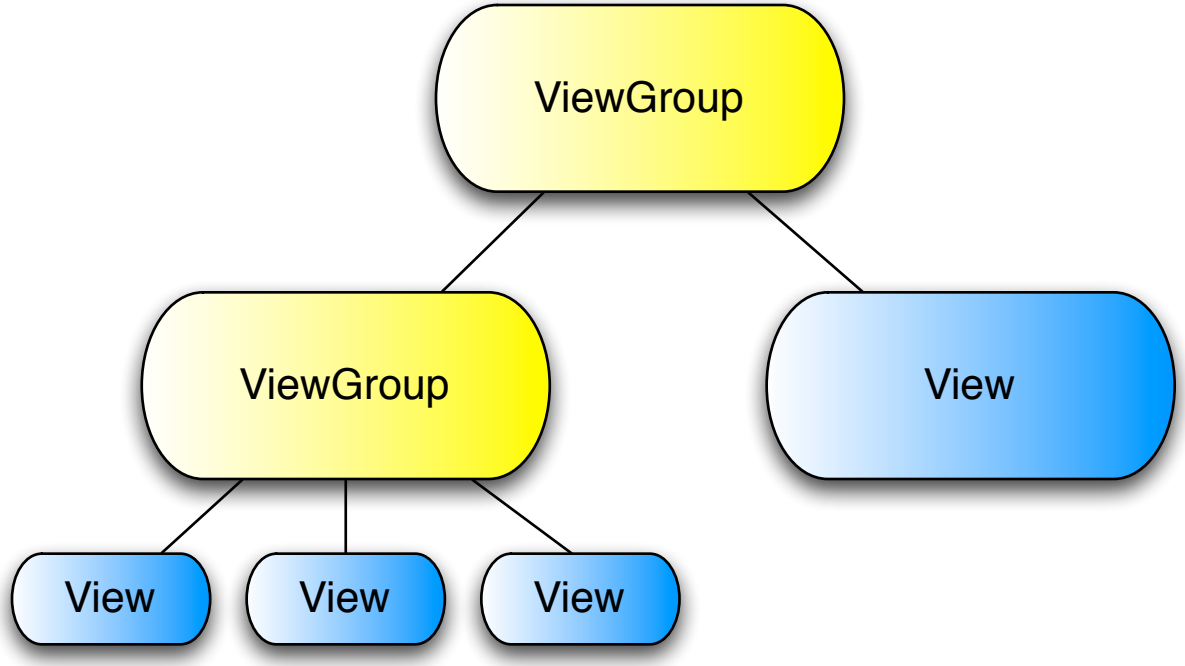
marakana

Dips and Sps

px (pixel)	Dots on the screen
in (inches)	Size as measured by a ruler
mm (millimeters)	Size as measured by a ruler
pt (points)	1/72 of an inch
dp (density-independent pixel)	Abstract unit. On screen with 160dpi, 1dp=1px
dip	synonym for dp and often used by Google
sp	Similar to dp but also scaled by users font size preference

marakana

Views and Layouts



ViewGroups contain other Views but are also Views themselves.

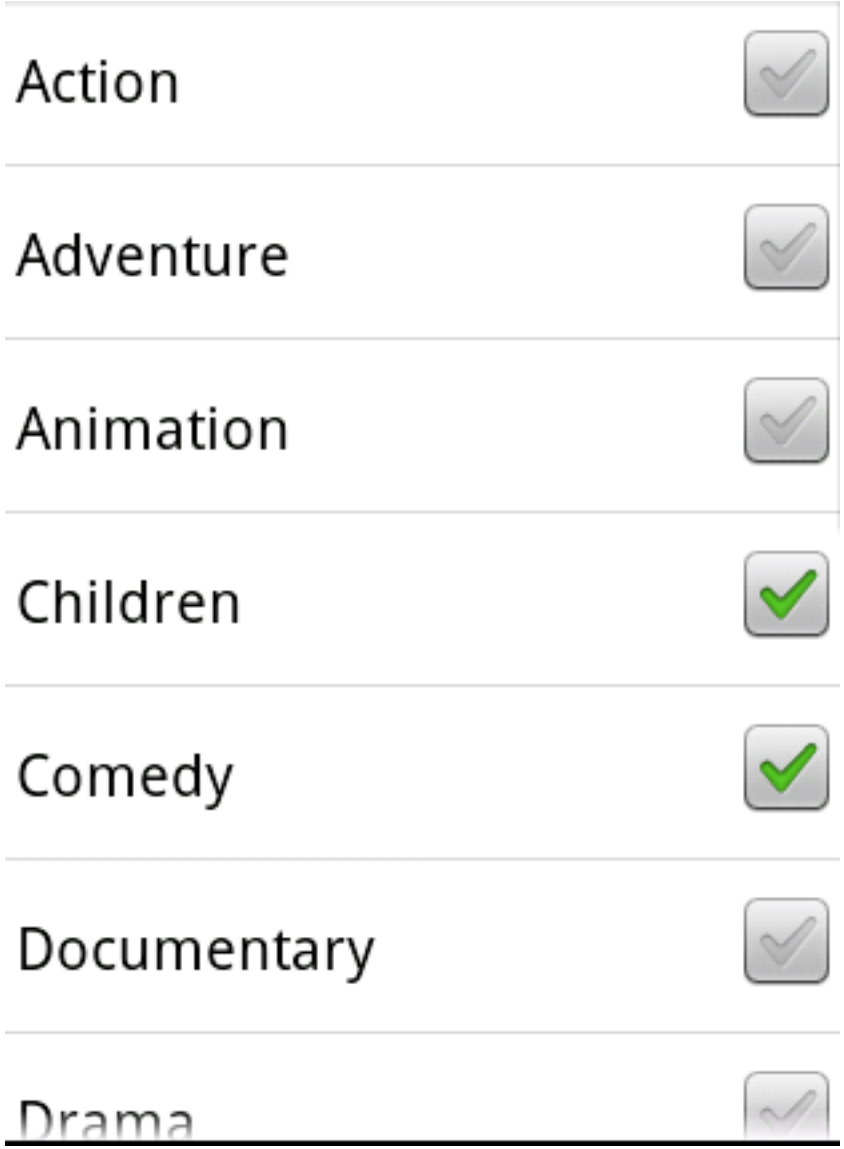
Common UI Components

Android UI includes many common modern UI widgets, such as Buttons, Tabs, Progress Bars, Date and Time Pickers, etc.

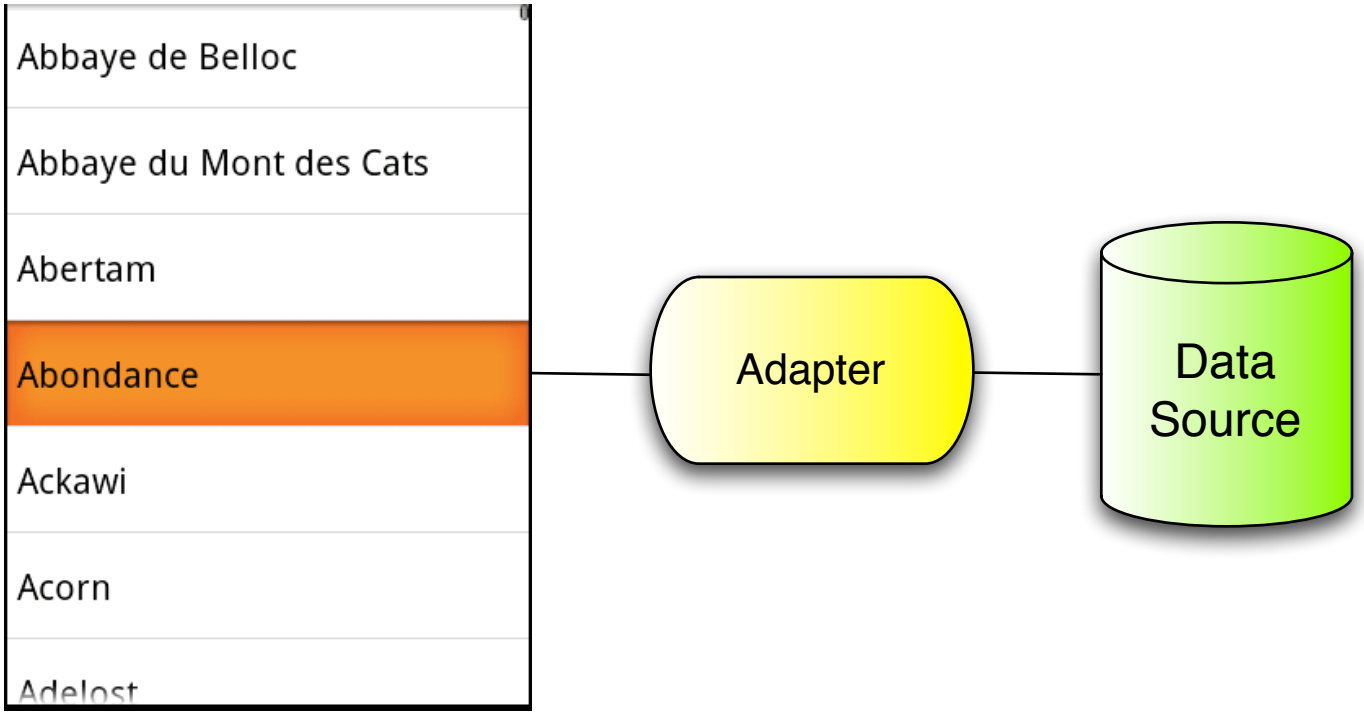
marakana

Selection Components

Some UI widgets may be linked to zillions of pieces of data. Examples are ListView and Spinners (pull-downs).



Adapters



To make sure they run smoothly, Android uses Adapters to connect them to their data sources. A typical data source is an Array or a Database.

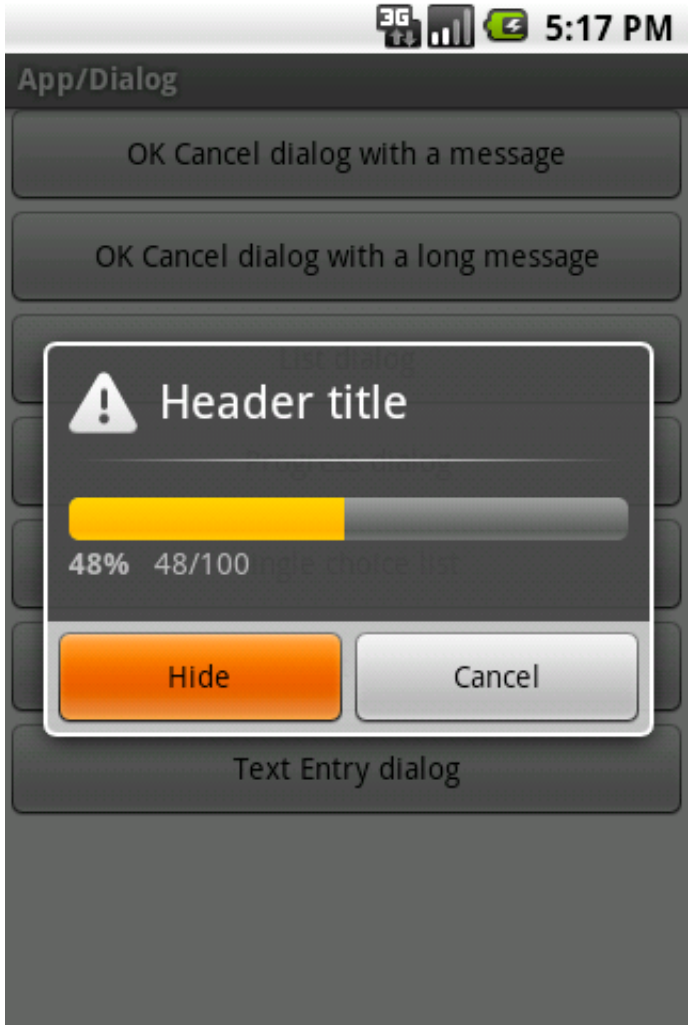
marakana

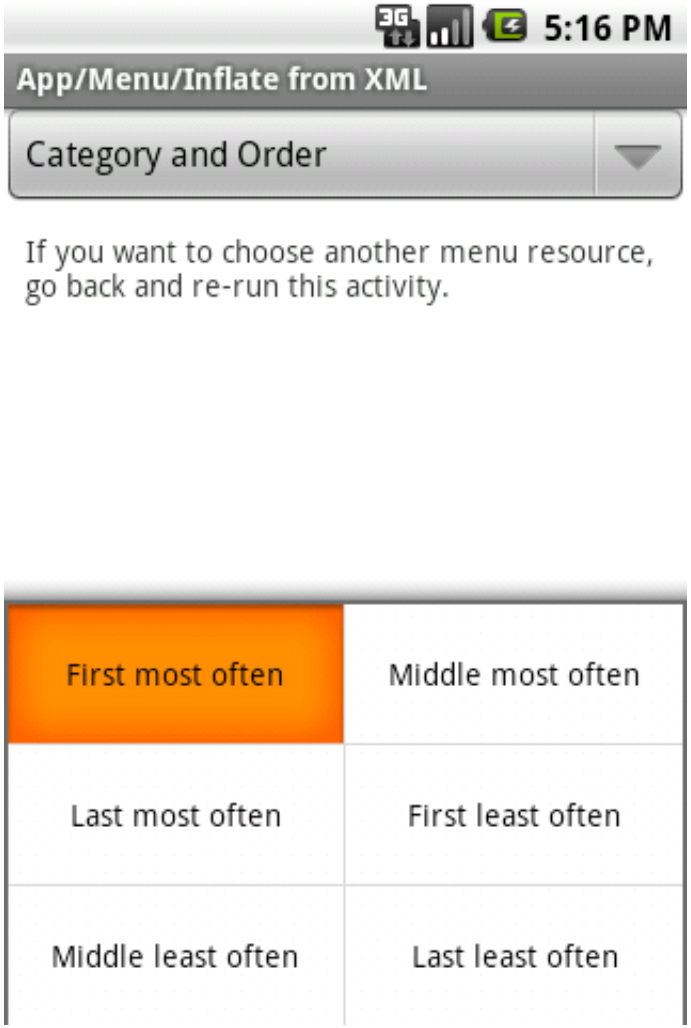
Complex Components

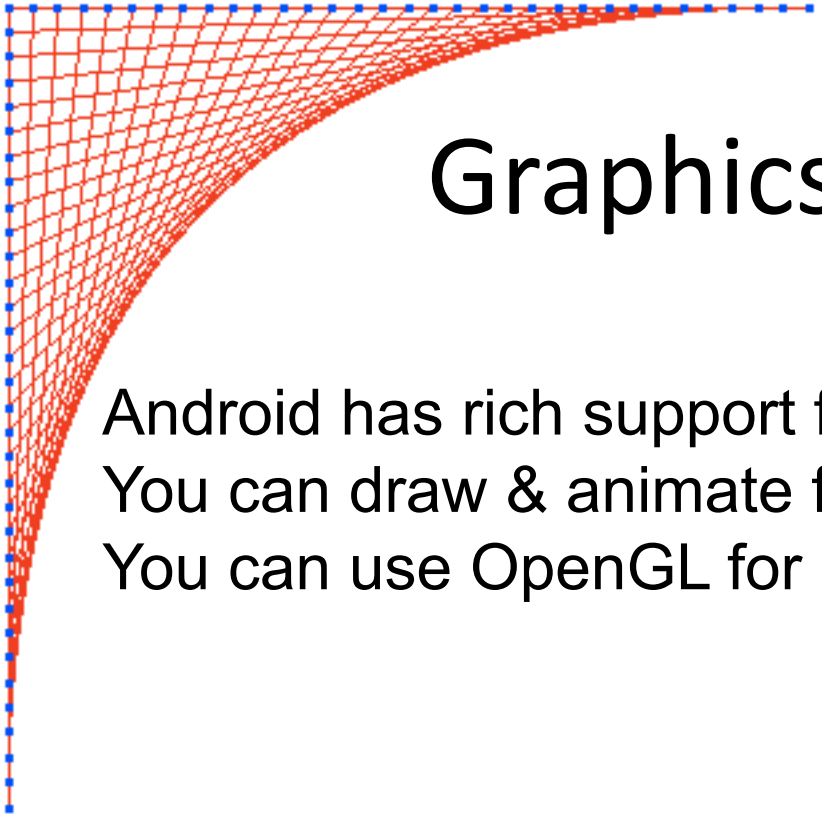
Certain high-level components are simply available just like Views. Adding a Map or a Video to your application is almost like adding a Button or a piece of text.

marakana

Menus and Dialogs







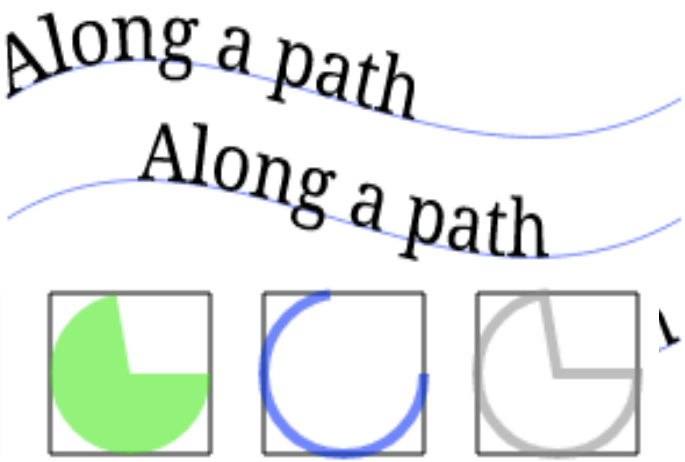
marakana

Graphics & Animation

Android has rich support for 2D graphics.
You can draw & animate from XML.
You can use OpenGL for 3D graphics.

Default

Custom



Multimedia

AudioPlayer lets you simply specify the audio resource and play it.

VideoView is a View that you can drop anywhere in your activity, point to a video file and play it.

XML:

```
<VideoView
    android:id="@+id/video"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:layout_gravity="center" />
```

Java:

```
player = (VideoView) findViewById(R.id.video);
player.setVideoPath("/sdcard/samplevideo.3gp");
player.start();
```

Google Maps

Google Maps is an add-on in Android.
It is not part of open-source project.

However, adding Maps is relatively easy using **MapView**.

XML:

```
<com.google.android.maps.MapView
android:id="@+id/map"
android:clickable="true"
android:layout_width="fill_parent"
android:layout_height="fill_parent"
android:apiKey="0EfLSgdSCWIN…A"
/>
```

marakana

OPERATING SYSTEM FEATURES

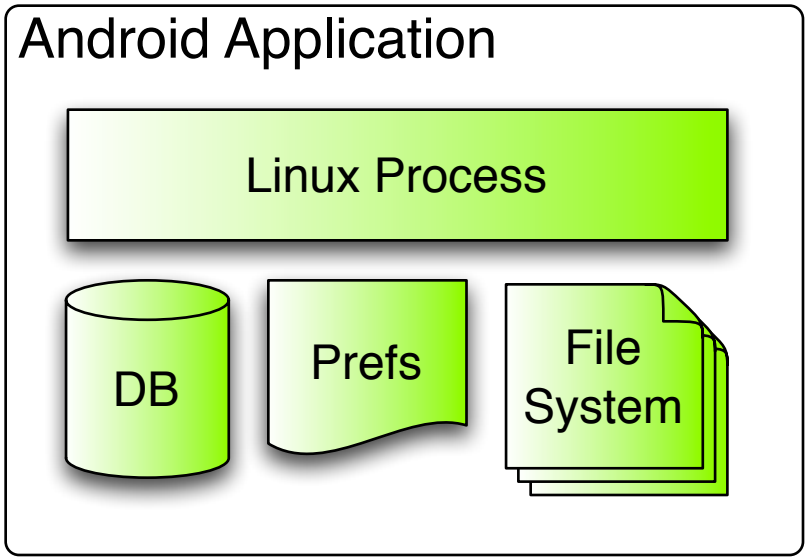
marakana

Security

Each Android application runs inside its own Linux process.

Additionally, each application has its own sandbox file system with its own set of preferences and its own database.

Other applications cannot access any of its data, unless it is explicitly shared.



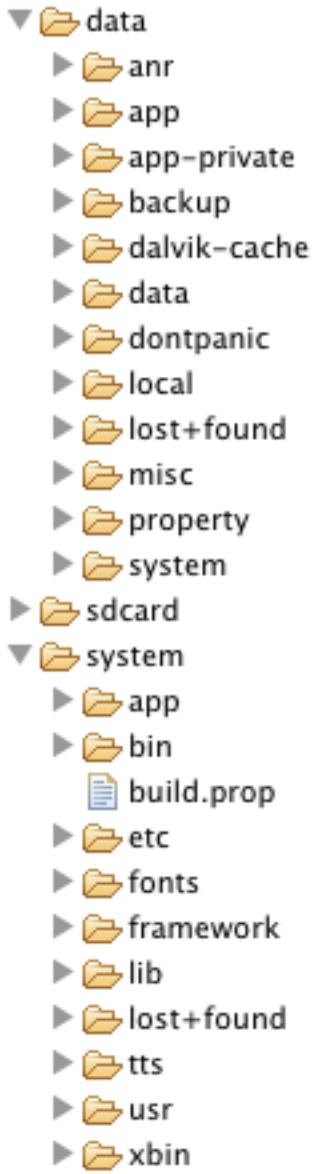
marakana

File System

The file system has three main mount points. One for system, one for the apps, and one for whatever.

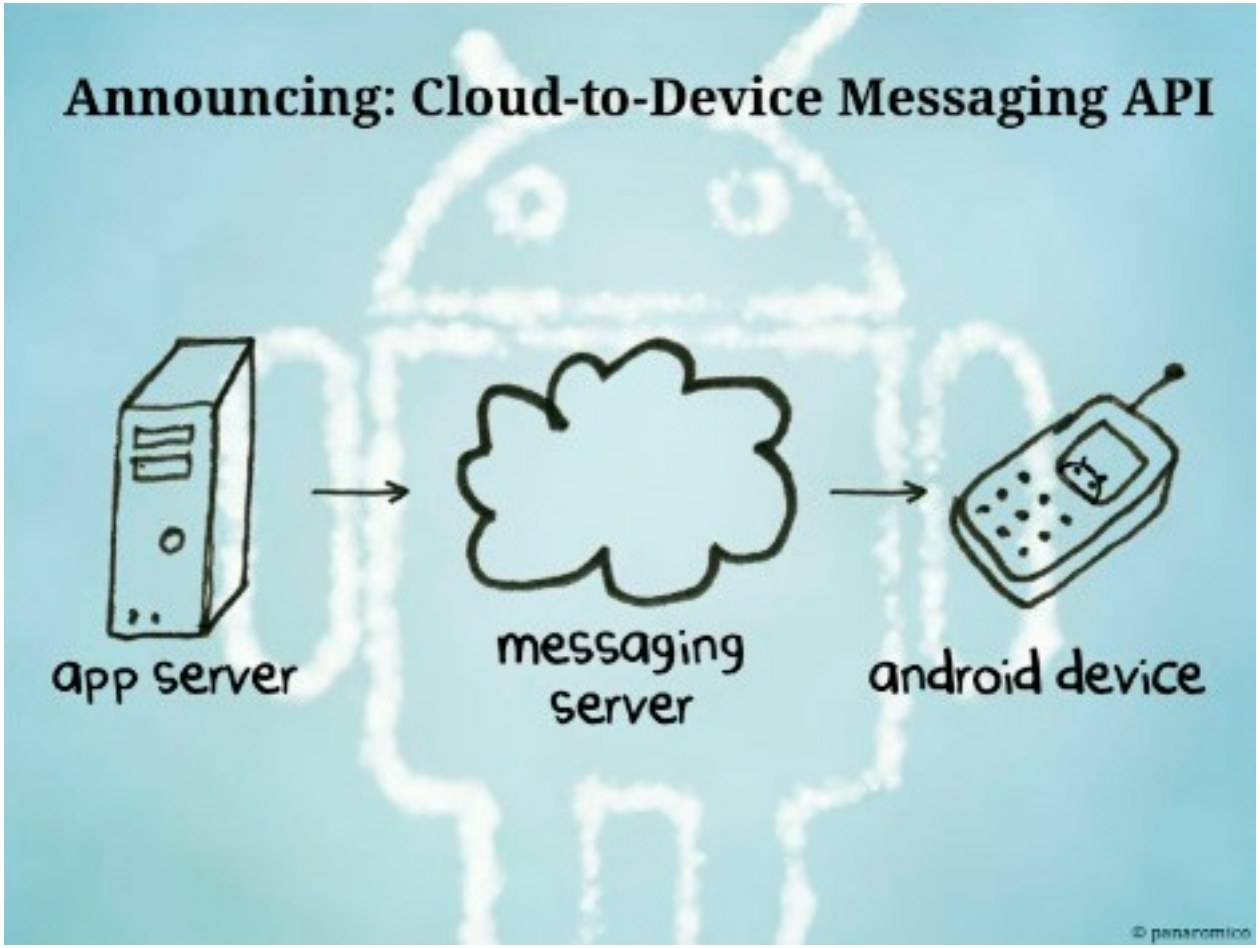
Each app has its own sandbox easily accessible to it. No one else can access its data. The sandbox is in /data/data/com.marakana/

SDCard is expected to always be there. It's a good place for large files, such as movies and music. Everyone can access it.



marakana

Cloud to Device Push



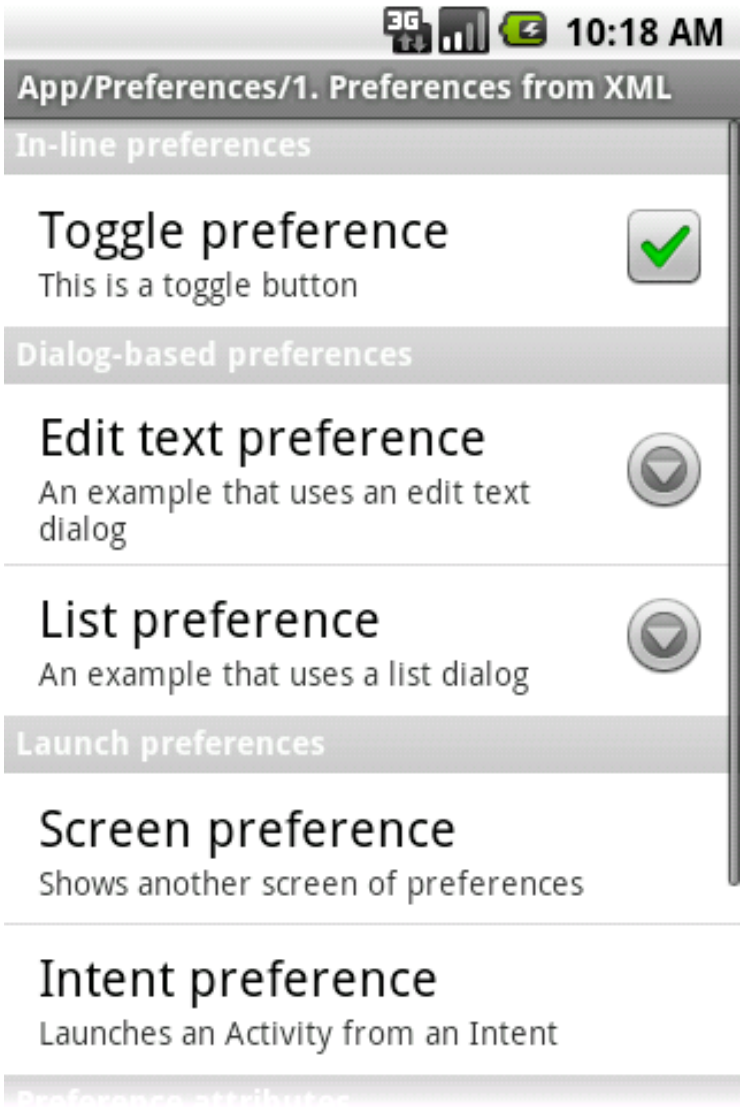
Big deal for many pull-based apps. Will make devices use less battery.

marakana

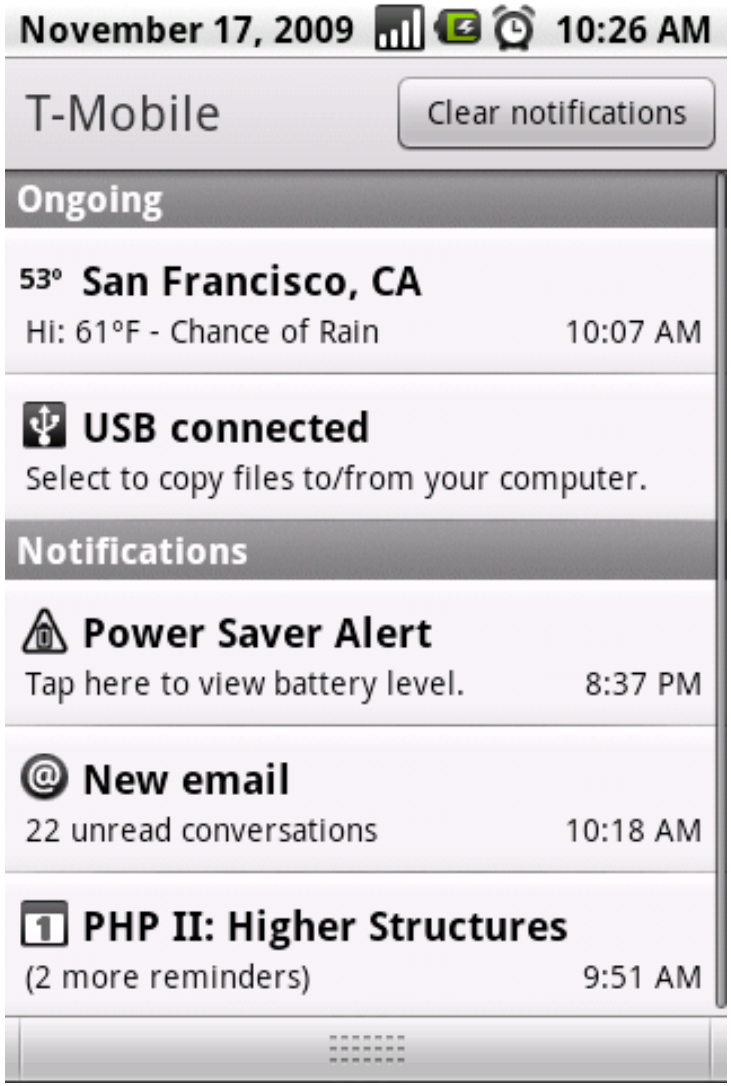
Preferences

Your app can support complex preferences quite easily.

You define your preferences in an XML file and the corresponding UI and data storage is done for free.



Notifications



Notifications are useful for applications to notify user of things going on in the background.

Notifications are implemented via Notification Manager.

marakana

SQLite Database

Android ships with SQLite3

SQLite is

Zero configuration
Serverless
Single database file
Cross-Platform
Compact
Public Domain

Database engine.

May you do good and not evil
May you find forgiveness for yourself and forgive others
May you share freely, never taking more than you give.

DEBUGGING ANDROID APPS

marakana

LogCat

The universal, most versatile way to track what is going on in your app.

Can be viewed via command line or Eclipse.

Logs can be generated both from SDK Java code, or low-level C code via Bionic libc extension.

Threads | Heap | File Explorer | LogCat

Log | GolfScore | ServicesDemo | MyService | HelloWorld | GeoTwitter (31)

Time	pid	tag	Message
01-13 07:00:33	223	MediaScannerS	done scanning volume external
01-13 07:00:33	223	MediaScannerS	start scanning volume external
01-13 07:00:33	223	SafeMediaScan	SafeMediaScanner constructor
01-13 07:00:33	223	SafeMediaScan	Scan directories external
01-13 07:00:34	223	SafeMediaScan	processDirectory: /sdcard, WBMP.OISAFE,PNG.OISAFE,GIF.OISAFE,BM
01-13 07:00:34	223	SafeMediaScan	processDirectory: /sdcard/LOST.DIR, WBMP.OISAFE,PNG.OISAFE,GIF.
01-13 07:00:34	223	SafeMediaScan	processDirectory: /sdcard/download, WBMP.OISAFE,PNG.OISAFE,GIF.
01-13 07:00:34	223	SafeMediaScan	processDirectory: /sdcard/dcim, WBMP.OISAFE,PNG.OISAFE,GIF.OISA
01-13 07:00:34	223	SafeMediaScan	processDirectory: /sdcard/dcim/100ANDRO, WBMP.OISAFE,PNG.OISAFE
01-13 07:00:34	223	SafeMediaScan	prescan time: 40ms
01-13 07:00:34	223	SafeMediaScan	scan time: 18ms
01-13 07:00:34	223	SafeMediaScan	postscan time: 0ms
01-13 07:00:34	223	SafeMediaScan	total time: 58ms
01-13 07:00:34	223	MediaScannerS	done scanning volume external
01-13 07:00:34	192	MediaScanner	prescan time: 197ms
01-13 07:00:34	192	MediaScanner	scan time: 348ms
01-13 07:00:34	192	MediaScanner	postscan time: 0ms
01-13 07:00:34	192	MediaScanner	total time: 545ms
01-13 07:00:34	192	MediaScannerS	done scanning volume external
01-13 07:01:16	50	InputManagerS	Starting input on non-focused client com.android.internal.view.
01-13 07:01:16	50	InputManagerS	Client not active, ignoring focus gain of: com.android.internal
01-13 07:01:16	50	ActivityManag	Displayed activity com.marakana/.GeoTwitter: 46853 ms (total 75
01-13 07:01:16	50	ARMAssembler	generated scanline 00000077:03545404 00000A04 00000000 [29 ip
01-13 07:01:16	50	ARMAssembler	generated scanline 00000177:03515104 00001A01 00000000 [73 ip
01-13 07:02:23	215	TweetsAdapter	bindView 6578988894
01-13 07:02:23	215	TweetsAdapter	bindView 6578889126
01-13 07:02:23	215	TweetsAdapter	bindView 6578808550
01-13 07:02:23	215	TweetsAdapter	bindView 6578760298
01-13 07:02:23	215	TweetsAdapter	bindView 6578637256
01-13 07:02:23	215	TweetsAdapter	bindView 6578275745
01-13 07:02:23	215	TweetsAdapter	bindView 6578263740
01-13 07:02:23	215	TweetsAdapter	bindView 6578213789
01-13 07:02:23	215	TweetsAdapter	bindView 6578988894
01-13 07:02:23	215	TweetsAdapter	bindView 6578889126
01-13 07:02:23	215	TweetsAdapter	bindView 6578808550
01-13 07:02:23	215	TweetsAdapter	bindView 6578760298
01-13 07:02:23	215	TweetsAdapter	bindView 6578637256
01-13 07:02:23	215	TweetsAdapter	bindView 6578275745
01-13 07:02:23	215	TweetsAdapter	bindView 6578263740
01-13 07:02:23	215	TweetsAdapter	bindView 6578213789

Filter:

marakana

Debugger

Debug

Thread [<3> main] (Suspended (breakpoint at line 102 in UpdaterService$Updater))
- UpdaterService$Updater.run() line: 102
- Handler.handleCallback(Message) line: 587
- Handler.dispatchMessage(Message) line: 92
- Looper.loop() line: 123
- ActivityThread.main(String[]) line: 4310
- Method.invokeNative(Object, Object[], Class, Class[], Class, int, boolean) line: not available [native method]
- Method.invoke(Object, Object...) line: 521
- ZygoteInit$MethodAndArgsCaller.run() line: 860
- ZygoteInit.main(String[]) line: 618
- NativeStart.main(String[]) line: not available [native method]

Thread [<13> Binder Thread #2] (Running)
Thread [<11> Binder Thread #1] (Running)

Variables | Breakpoints

Name	Value
this	UpdaterService$Updater (id=830065735048)
count	-1
this$0	UpdaterService (id=830065715928)
db	SQLiteDatabase (id=830065717728)
dbHelper	DbHelper (id=830065716160)
handler	Handler (id=830065734120)
mActivityManager	ActivityManagerProxy (id=830065578456)

android.database.sqlite.SQLiteDatabase@43c5b1e0

UpdaterService.java

```
96    public void run() {
97      initTwitter();
98      // Update the database
99      try {
100       count = dbHelper.updateDatabase(db, twitter.getFriendsTimeline());
101     } catch (TwitterException e) {
102       Log.d(TAG, "UpdaterService Updater died: " + e);
103     }
104
105     // Post a notification & Send Broadcast
106     if (count > 0) {
107       PendingIntent pendingIntent = PendingIntent.getActivity(
108           UpdaterService.this, 1, new Intent(UpdaterService.this,
```

Outline
- DELAY : long
- prefs : SharedPreferences
- dbHelper : DbHelper
- db : SQLiteDatabase
- twitter : Twitter
- handler : Handler
- updater : Updater
- onBind(Intent) : IBinder
- onCreate() : void
  - new OnSharedPreferenceChangeListener()
- onStart(Intent, int) : void
- onDestroy() : void
- initTwitter() : void
- Updater
  - count : int
  - run() : void

Console | Tasks

Android

```
[2010-01-13 07:06:17 - GeoTwitter]Automatic Target Mode: using existi
[2010-01-13 07:06:17 - GeoTwitter]Application already deployed. No nee
[2010-01-13 07:06:17 - GeoTwitter]Starting activity com.marakana.GeoTw
[2010-01-13 07:06:19 - GeoTwitter]ActivityManager: Starting: Intent {
[2010-01-13 07:06:20 - GeoTwitter]Attempting to connect debugger to '
```

LogCat

Log | GolfScore | ServicesDemo | MyService | HelloWorld | GeoTwitter (74)

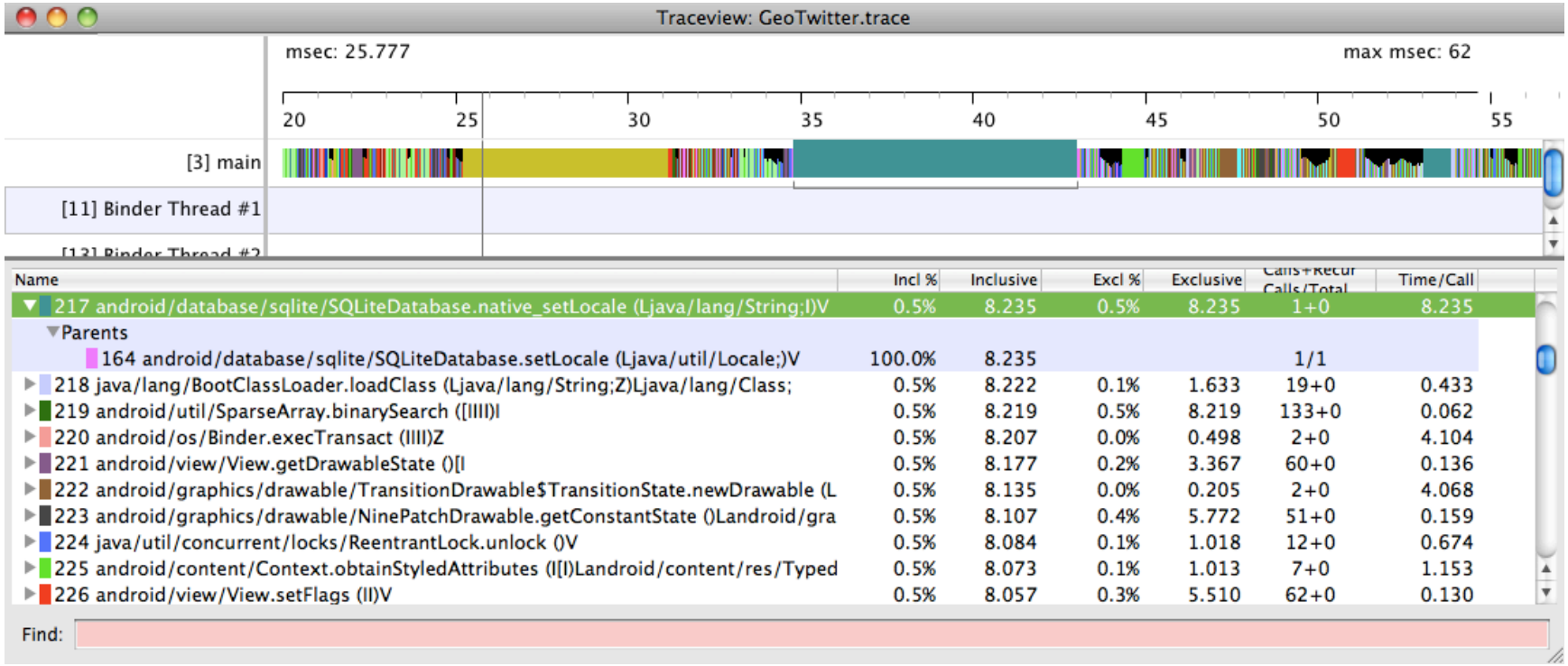
Time	pid	tag	Message
01-13 07:00:33	223	SafeMediaScan	SafeMediaScanner constructor
01-13 07:00:33	223	SafeMediaScan	Scan directories external
01-13 07:00:34	223	SafeMediaScan	processDirectory: /sdcard, WBMP.OISAFE,PNG.OISAFE,GIF.OISAFE,BM
01-13 07:00:34	223	SafeMediaScan	processDirectory: /sdcard/LOST.DIR, WBMP.OISAFE,PNG.OISAFE,GIF.
01-13 07:00:34	223	SafeMediaScan	processDirectory: /sdcard/download, WBMP.OISAFE,PNG.OISAFE,GIF.

Filter:

Your standard debugger is included in SDK, with all the usual bells & whistles.

marakana

TraceView



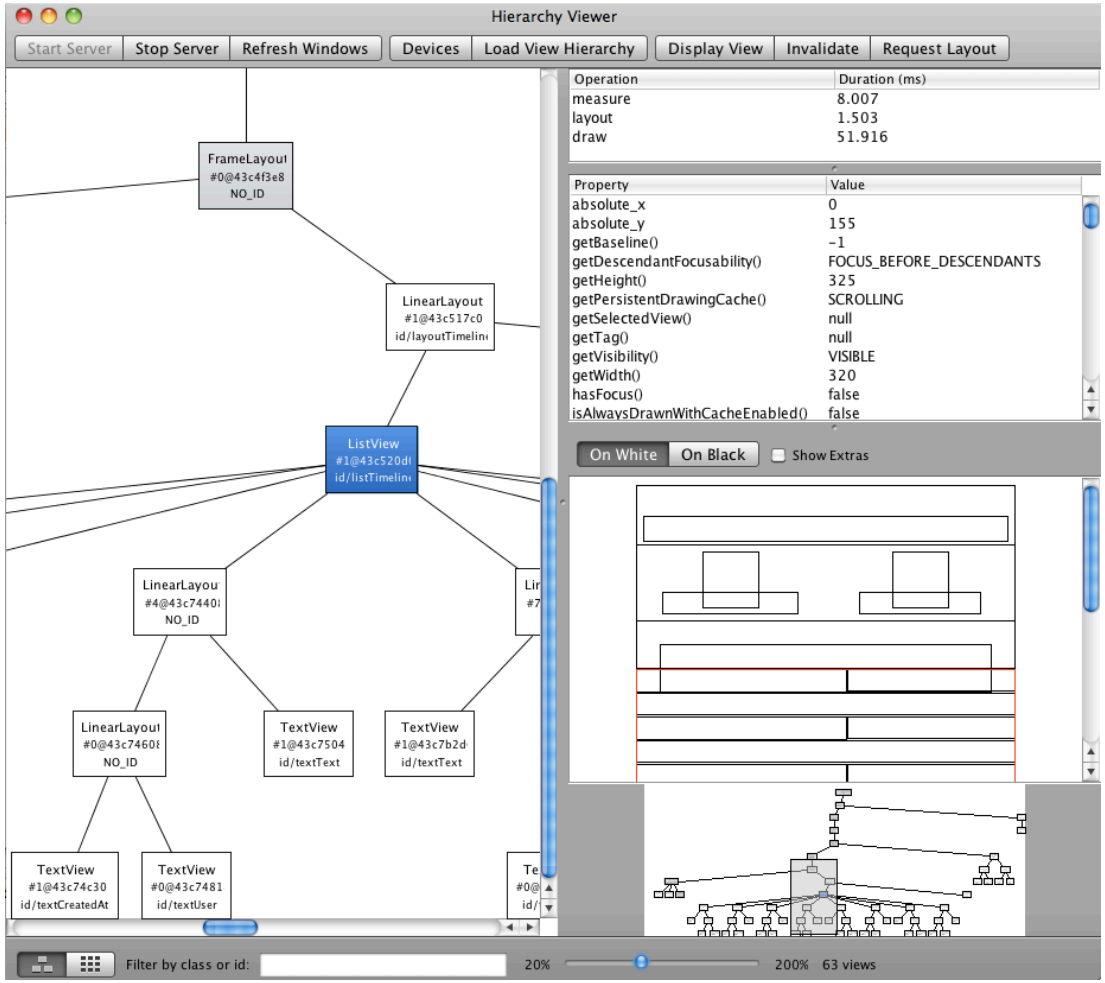
TraceView helps you profile you application and find bottlenecks. It shows execution of various calls through the entire stack. You can zoom into specific calls.

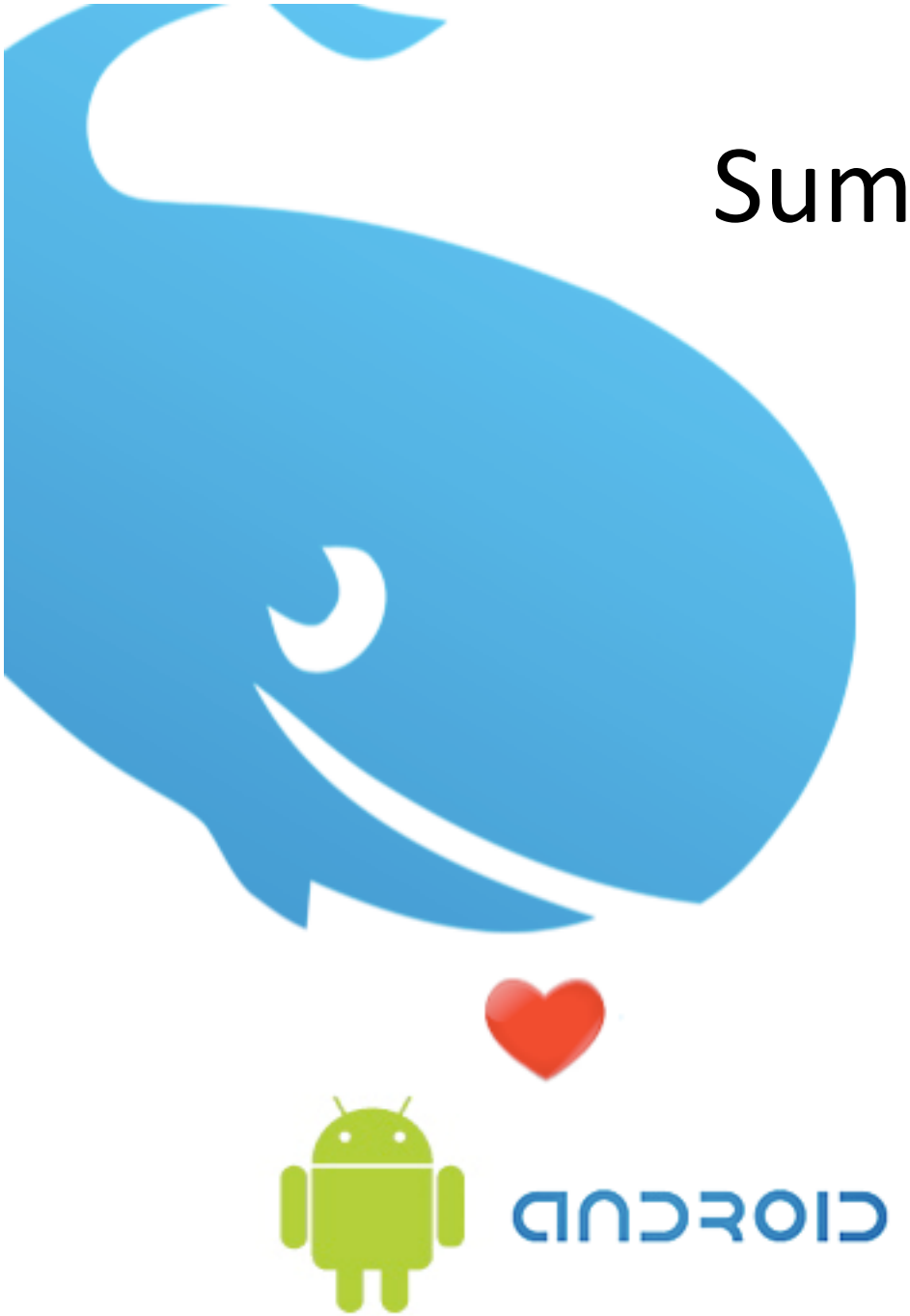
marakana

Hierarchy Viewer

Hierarchy Viewer helps you analyze your User Interface.

Base UI tends to be the most “expensive” part of your application, this tool is very useful.





Summary

Android is open and complete system for mobile development. It is based on Java and augmented with XML.

Android is being adopted very quickly both by users, carriers, and manufacturers.

It takes about 3-5 days of intensive training to learn Android application development for someone who has basic Java (or similar) experience.

Marko Gargenta, Marakana.com
marko@marakana.com
+1-415-647-7000