Programming Handheld Systems

Adam Porter

The Android Platform

The Android Platform

A software stack for mobile devices:

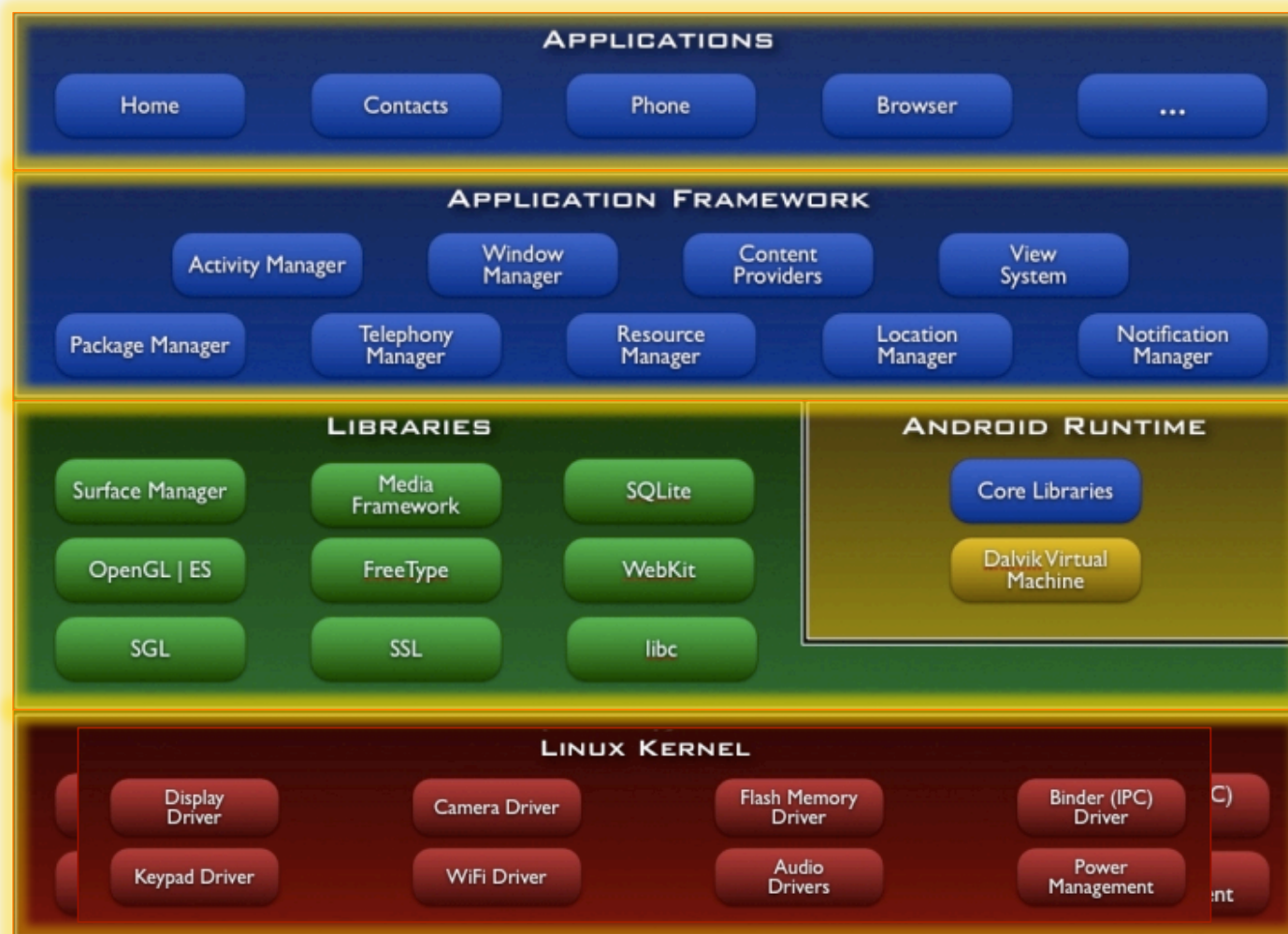- OS kernel, system libraries, application frameworks & key apps

Android SDK for creating apps

- Libraries & development tools

- Lots of documentation. Start browsing today!

See: http://developer.android.com/training

THE ANDROID ARCHITECTURE

Linux Kernel – standard services

Security

Memory & process management

File & network I/O

Device drivers

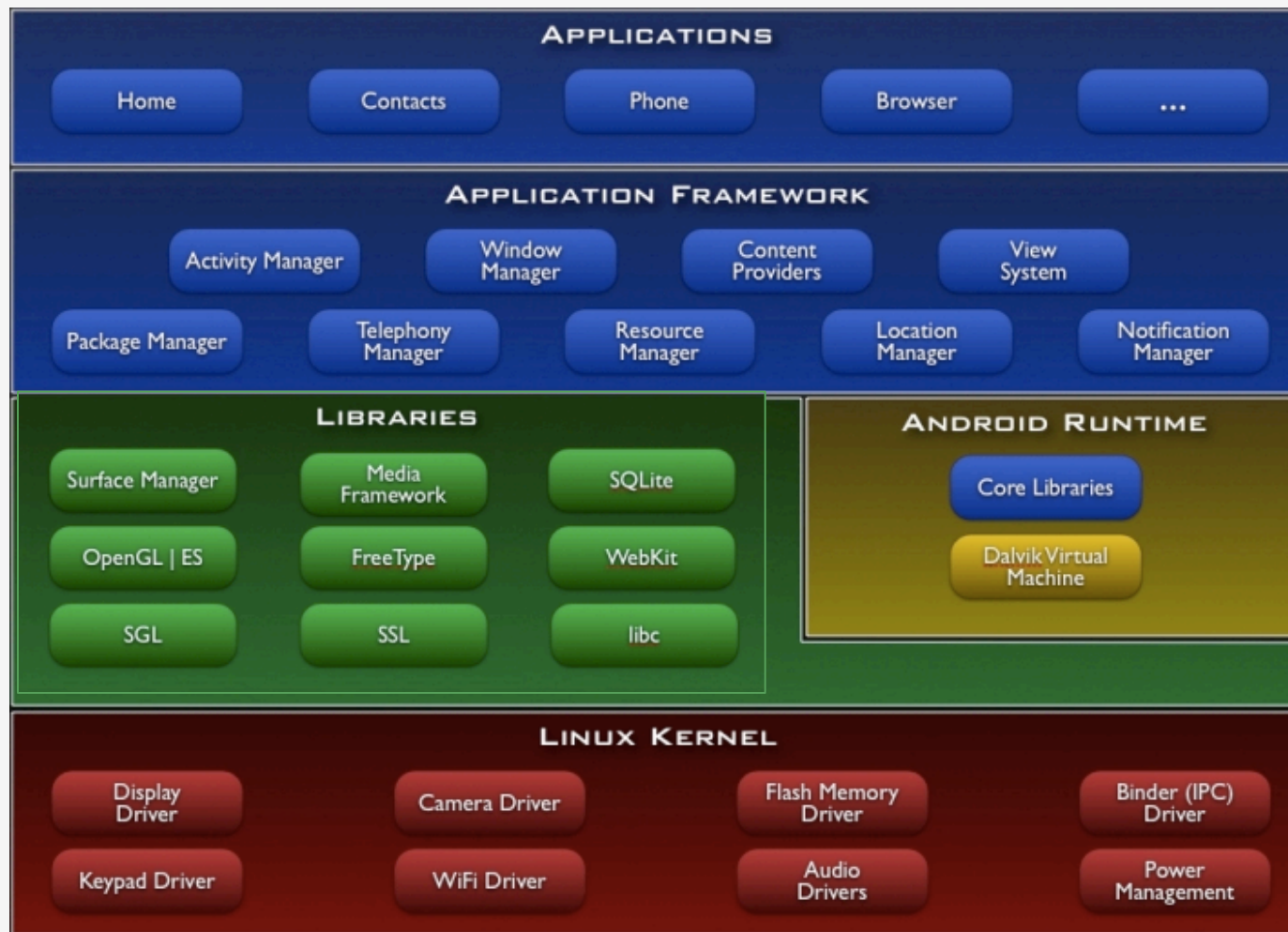# Linux Kernel – Android-specific

Power management

Android shared memory

Low memory killer

Interprocess communication

And much more

THE ANDROID ARCHITECTURE

APPLICATIONS

| Home | Contacts | Phone | Browser | ... |

APPLICATION FRAMEWORK

| Activity Manager | Window Manager | Content Providers | View System |
| Package Manager | Telephony Manager | Resource Manager | Location Manager | Notification Manager |

LIBRARIES

Surface Manager	Media Framework	SQLite
OpenGL	ES	FreeType	WebKit
SGL	SSL	libc

ANDROID RUNTIME

Core Libraries

Dalvik Virtual Machine

LINUX KERNEL

| Display Driver | Camera Driver | Flash Memory Driver | Binder (IPC) Driver |
| Keypad Driver | WiFi Driver | Audio Drivers | Power Management |

Libraries

System C library

  Bionic libc

Surface Mgr.

  Display
  management

Media
Framework

  Audio/ Video
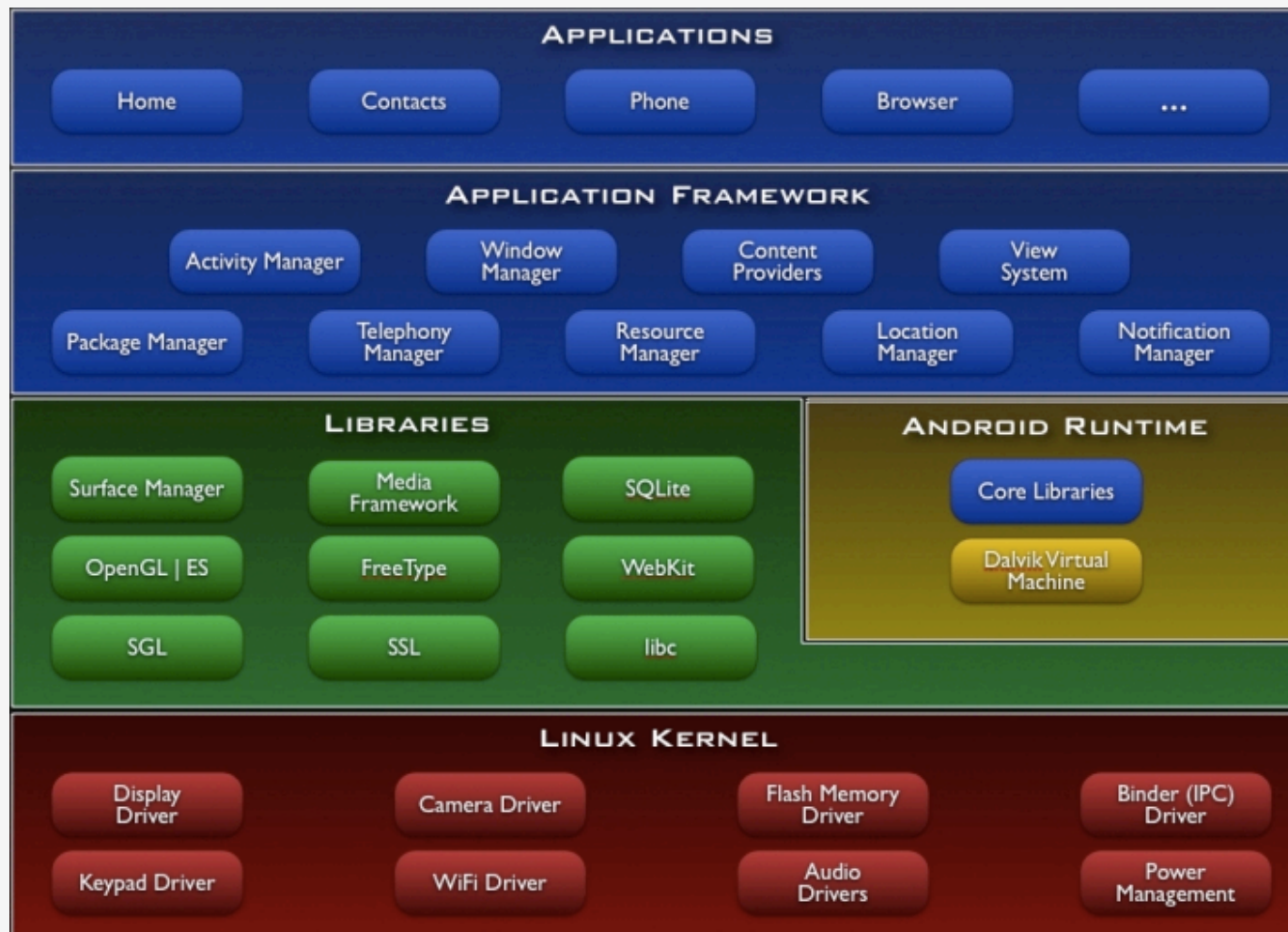
Webkit

  browser engine

OpenGL

  Graphics engines

SQLite

  Relational
  database engine

The Android Architecture

Android Runtime

Two main components
  Core Java libraries
  Dalvik Virtual Machine

Core Java Libraries

basic java classes -- java.*, javax.*

app lifecycle -- android.*

Internet/Web services -- org. *

Unit testing -- junit.*

Dalvik Virtual Machine

Apps are executed by the dalvik virtual machine

Typical workflow

App written in java

compiled to Java bytecode files

dx converts java bytecode files to a single dex bytecode file (classes.dex)

Dalvik executes dex bytecode file

Dalvik Virtual Machine

Designed for resource-constrained environments
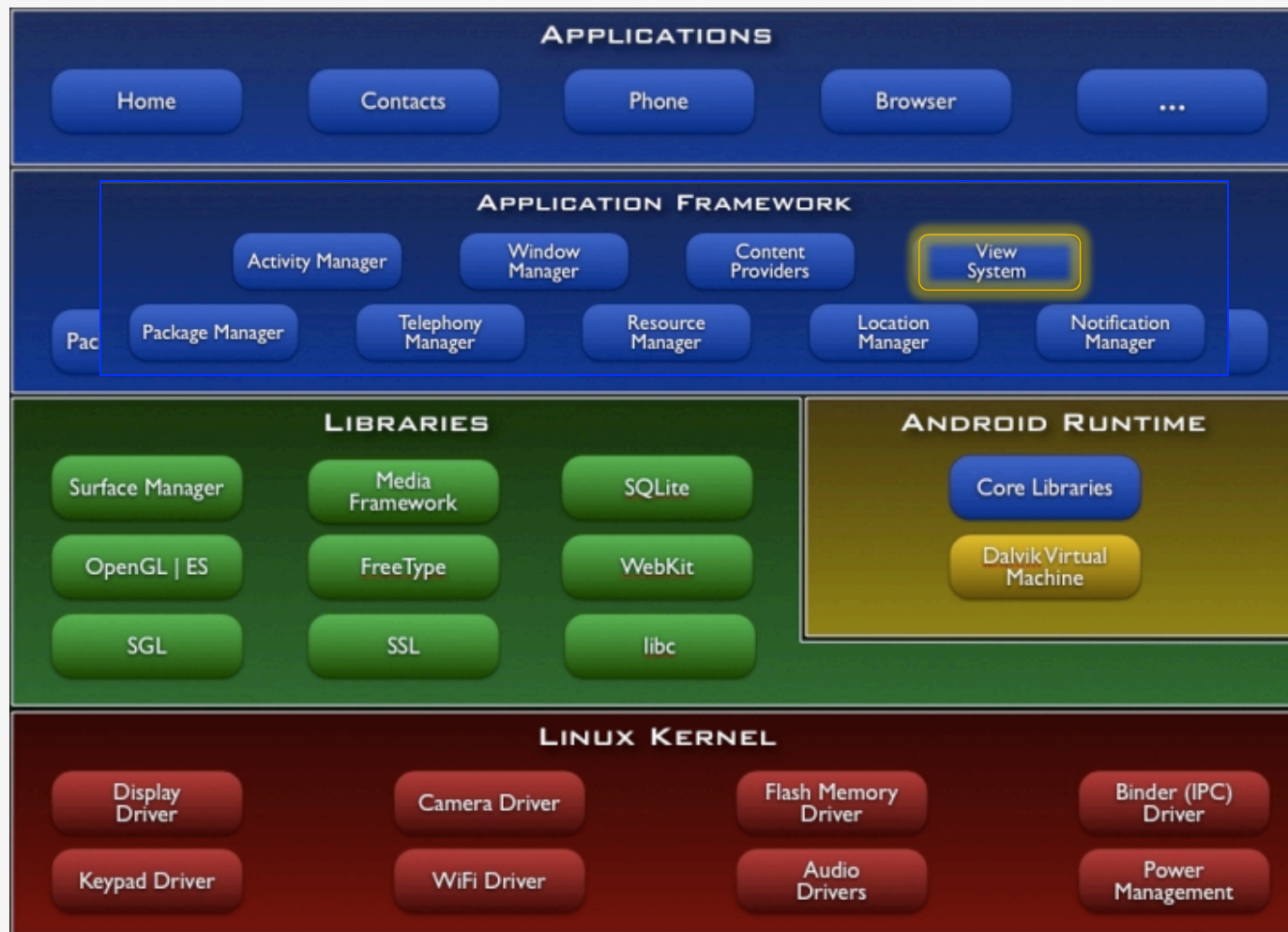
- Slower CPU
- Less RAM
- Limited battery life

Dalvik Virtual Machine

See:

Dalvik VM Internals by Dan Bornstein

http://www.youtube.com/watch?v=ptjedOZEXPM

THE ANDROID ARCHITECTURE

Package Manager

Keeps track of app packages on device

APPS   WIDGETS

Netflix

News & Weather

Nova Launcher

OverDrive Media

People

Phone

Play Books

Play Movies & TV

Play Music

Play Store

PocketCloud

QuoteViewer

ROM Manager

Root Checker

Screencast

Settings

ssh/sftp Server

SuperSU

Talk

Terminal Emulator

Voice Dialer

Voice Search

YouTube
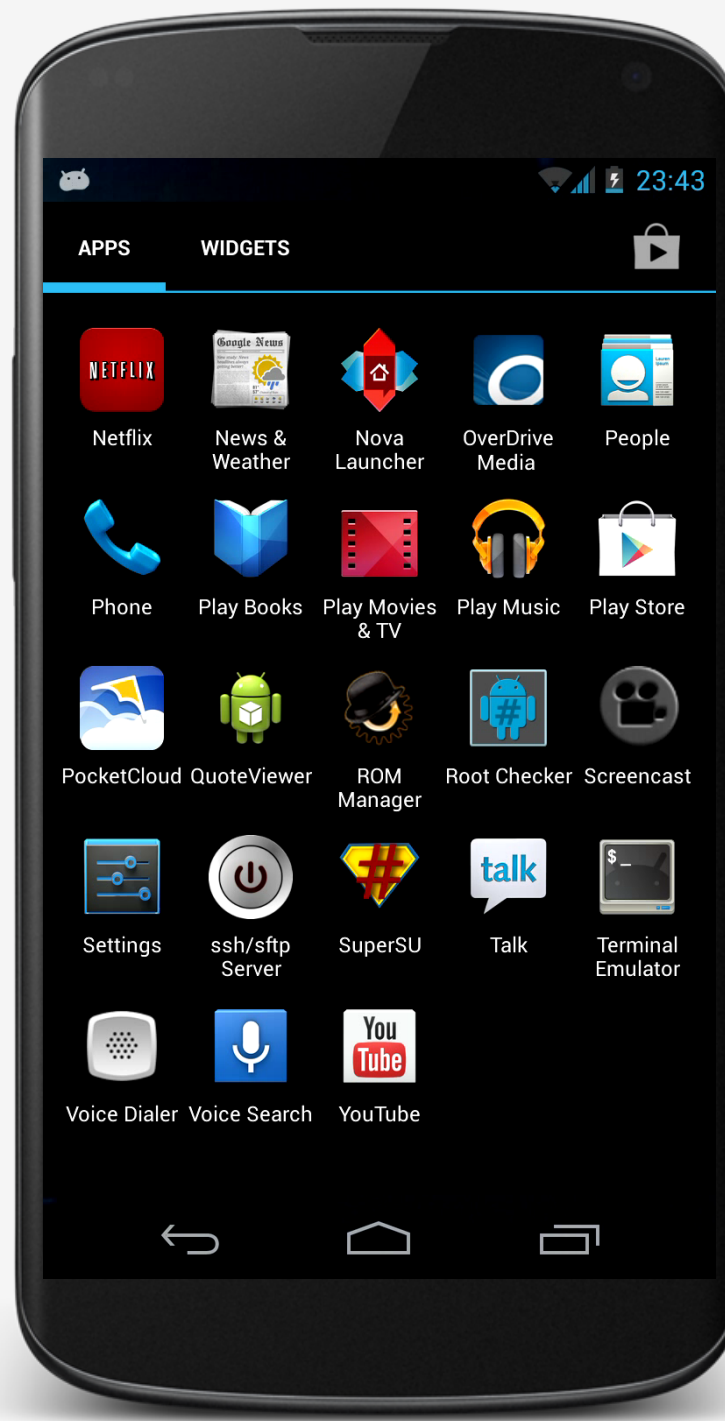
Window Manager

Manages the windows comprising
an App

NOTIFICATION BAR

SUBWINDOW
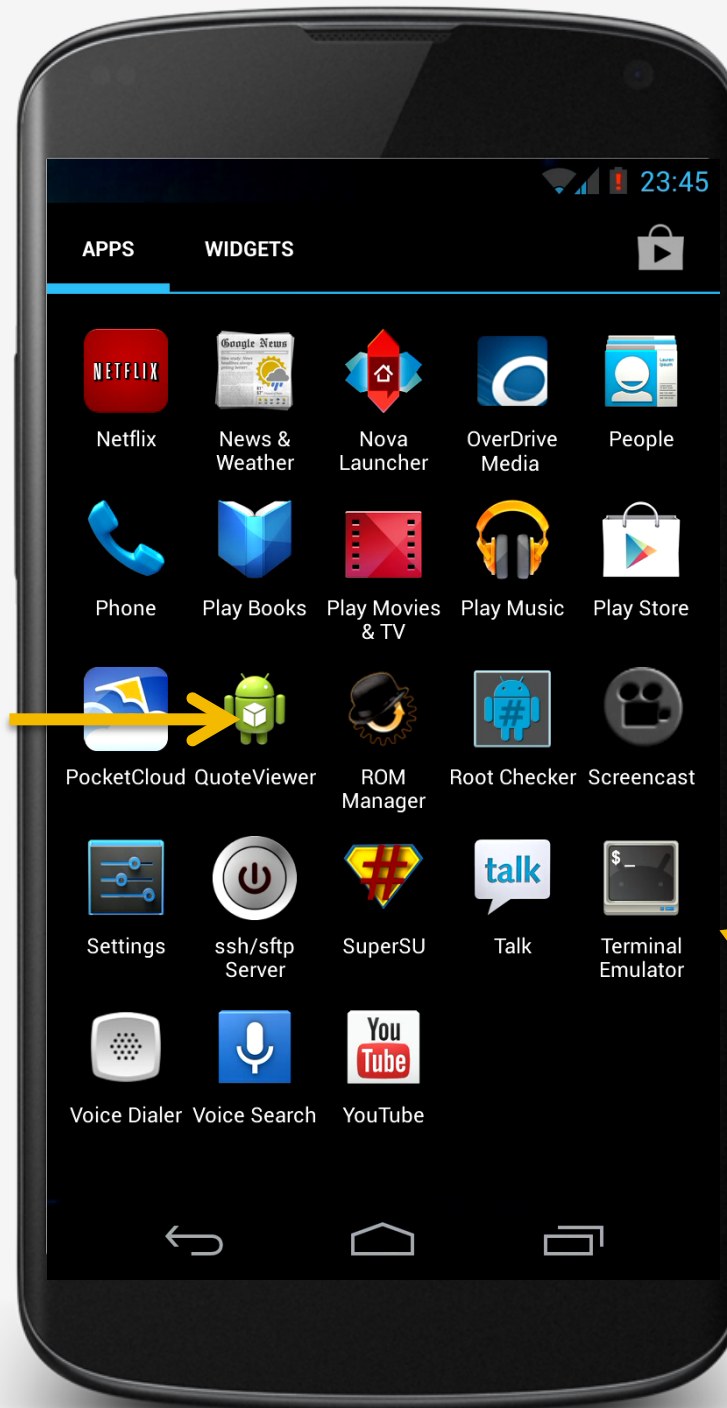
MAIN WINDOW

View System

Provides common user interface elements

    E.g., icons, text entry boxes, buttons and more

Tabs

APPS   WIDGETS

23:47

Netflix

News & Weather

Nova Launcher

OverDrive Media

People

Phone

Play Books

Play Movies & TV

Play Music

Play Store

PocketCloud

QuoteViewer

ROM Manager

Root Checker

Screencast

Settings

ssh/sftp Server

SuperSU

Talk

Terminal Emulator

Voice Dialer

Voice Search

YouTube

TEXTVIEW →

BUTTONS

APPS   WIDGETS   23:49

Netflix   News & Weather   Nova Launcher   OverDrive Media   People

Phone   Play Books   Play Movies & TV   Play Music   Play Store

PocketCloud   QuoteViewer   ROM Manager   Root Checker   Screencast

Settings   ssh/sftp Server   SuperSU   Talk   Terminal Emulator

Voice Dialer   Voice Search   YouTube
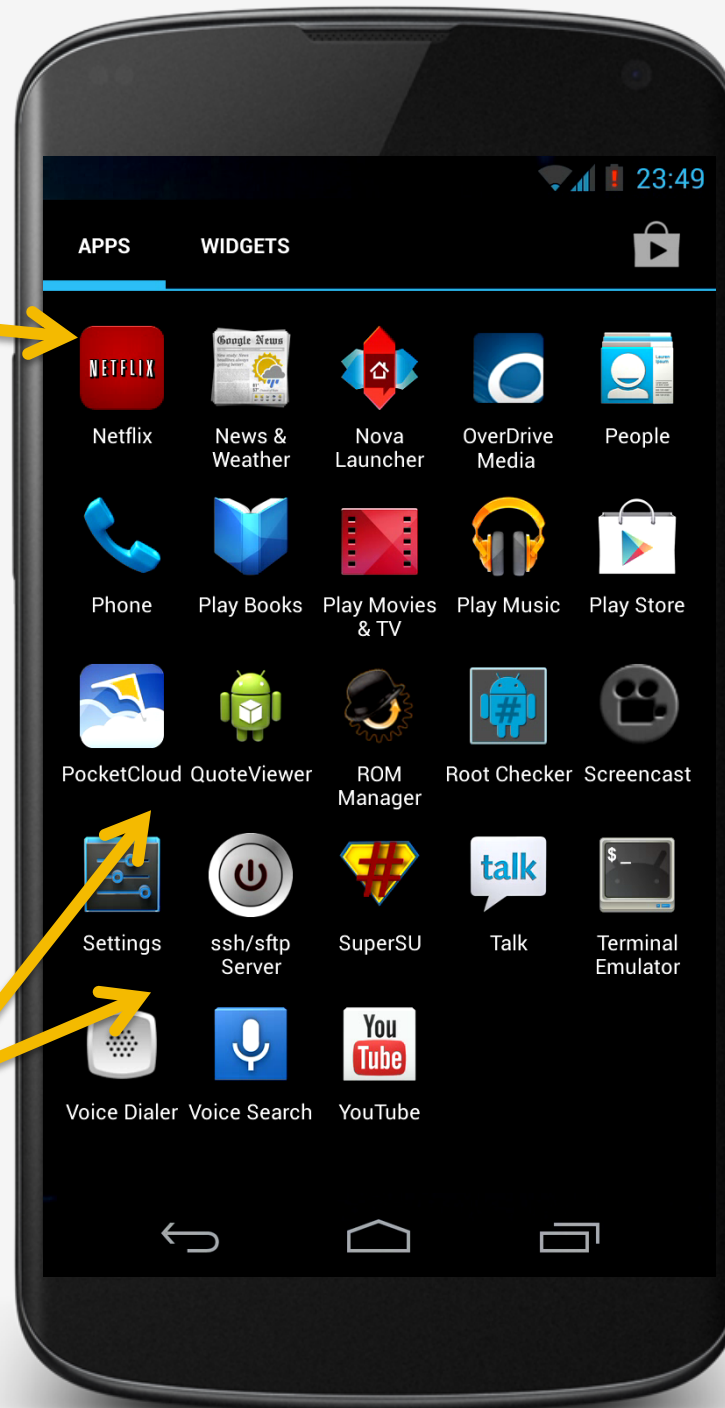
Resource Manager

Manages non-compiled resources

e.g., strings, graphics, & layout files

23:50

Netflix

News & Weather

Nova Launcher

OverDrive Media

People

Phone

Play Books

Play Movies & TV

Play Music

Play Store

PocketCloud

QuoteViewer

ROM Manager

Root Checker

Screencast

Settings

ssh/sftp Server

SuperSU

Talk

Terminal Emulator

Voice Dialer

Voice Search

YouTube
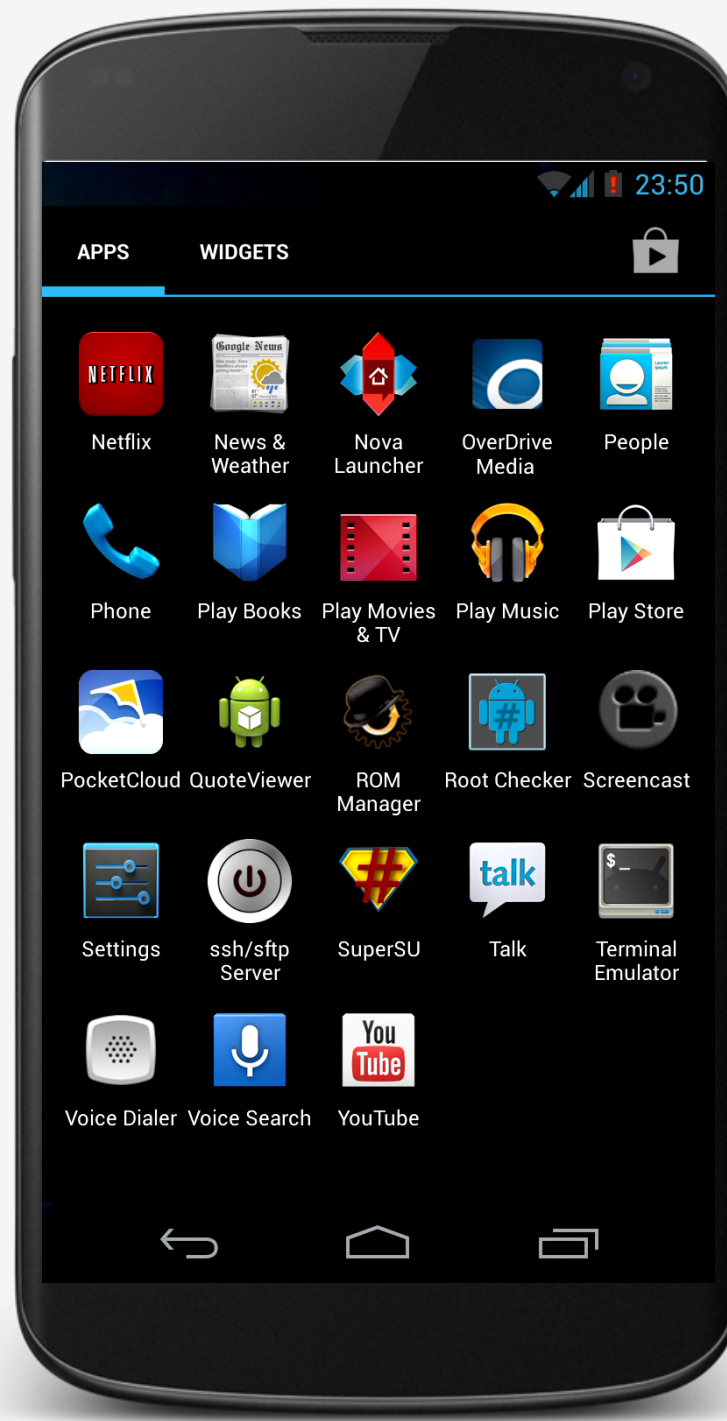
Activity Manager

Manages app lifecycle and navigation stack
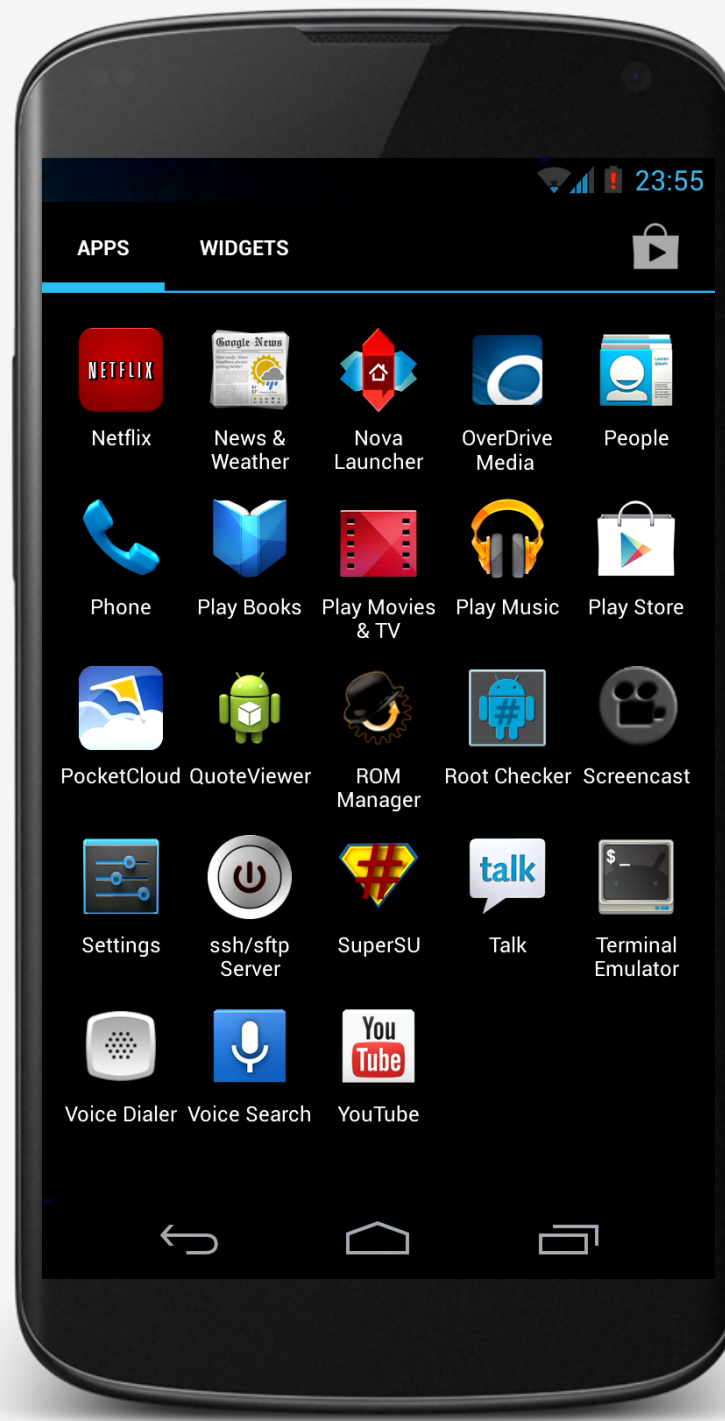
ContentProvider

Inter-application data sharing
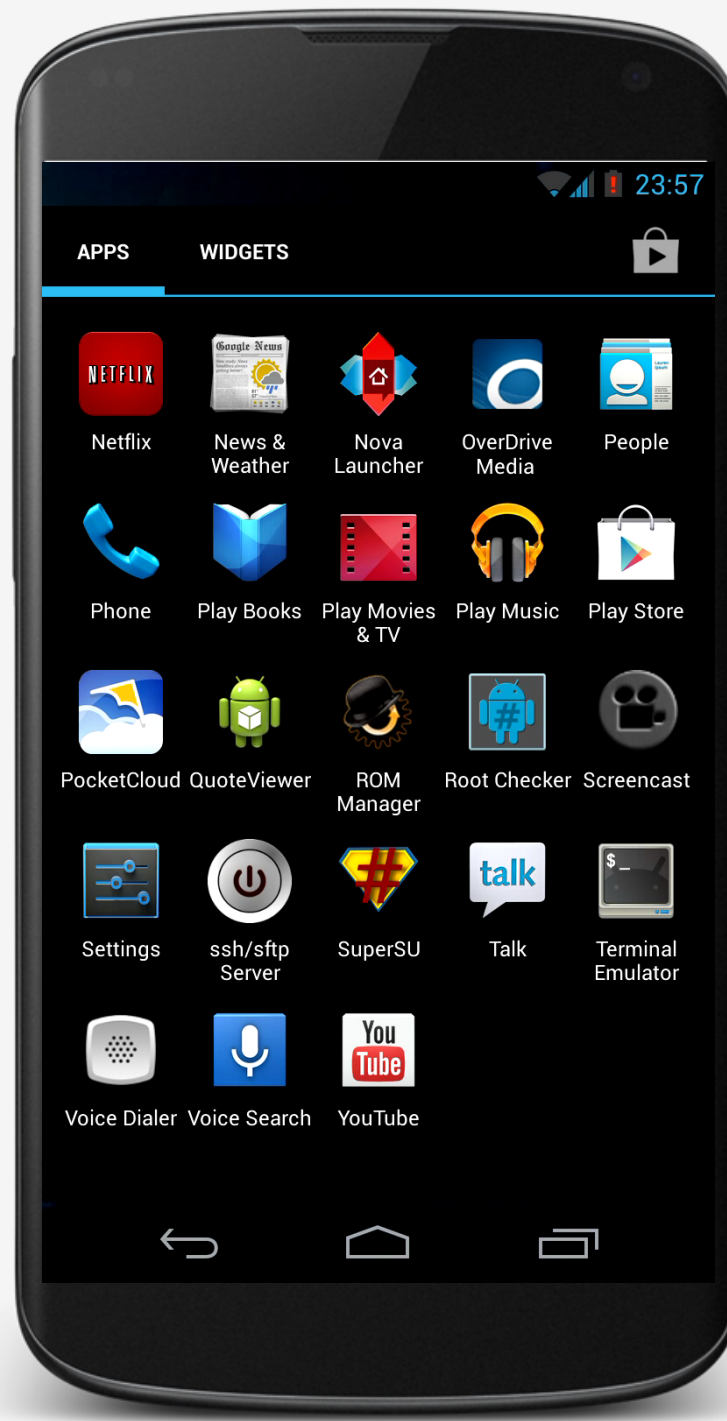
LocationManager

Provides location & movement information

00:00

Netflix	News & Weather	Nova Launcher	OverDrive Media	People
Phone	Play Books	Play Movies & TV	Play Music	Play Store
PocketCloud	QuoteViewer	ROM Manager	Root Checker	Screencast
Settings	ssh/sftp Server	SuperSU	Talk	Terminal Emulator
Voice Dialer	Voice Search	YouTube		
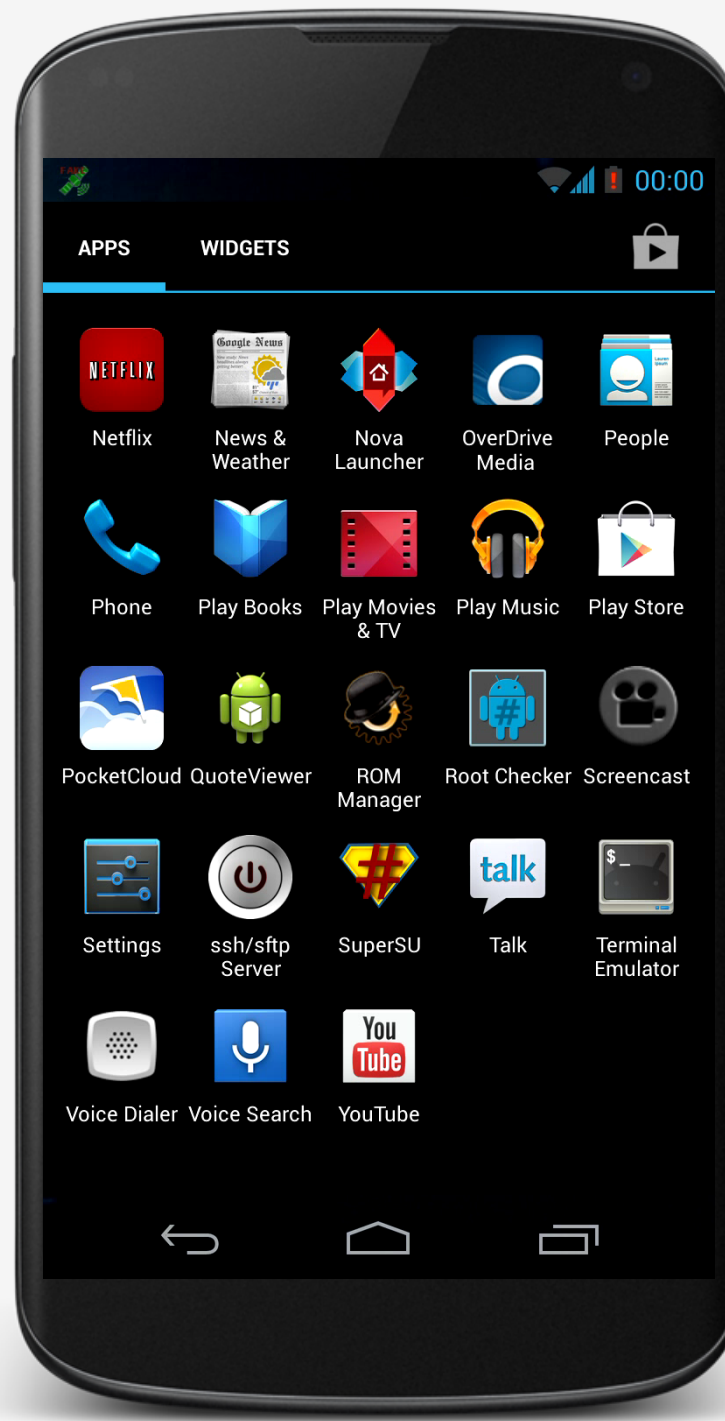
NotificationManager

Place notification icons in the status
bar when important events occur

09:53

Netflix

News & Weather

Nova Launcher

OverDrive Media

People

Phone

Play Books

Play Movies & TV

Play Music

Play Store

PocketCloud

QuoteViewer

ROM Manager

Root Checker

Screencast

Settings

ssh/sftp Server

SuperSU

Talk

Terminal Emulator

Voice Dialer

Voice Search

YouTube
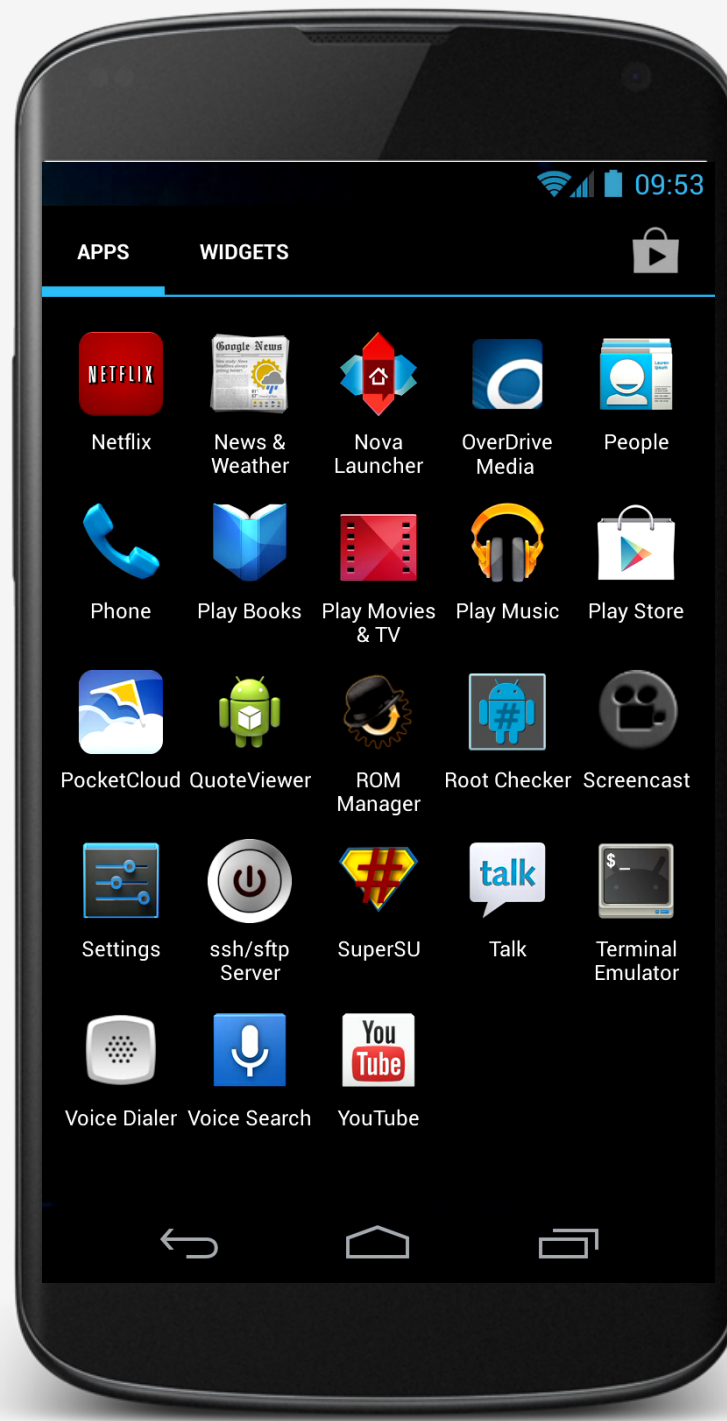
NOTIFICATION
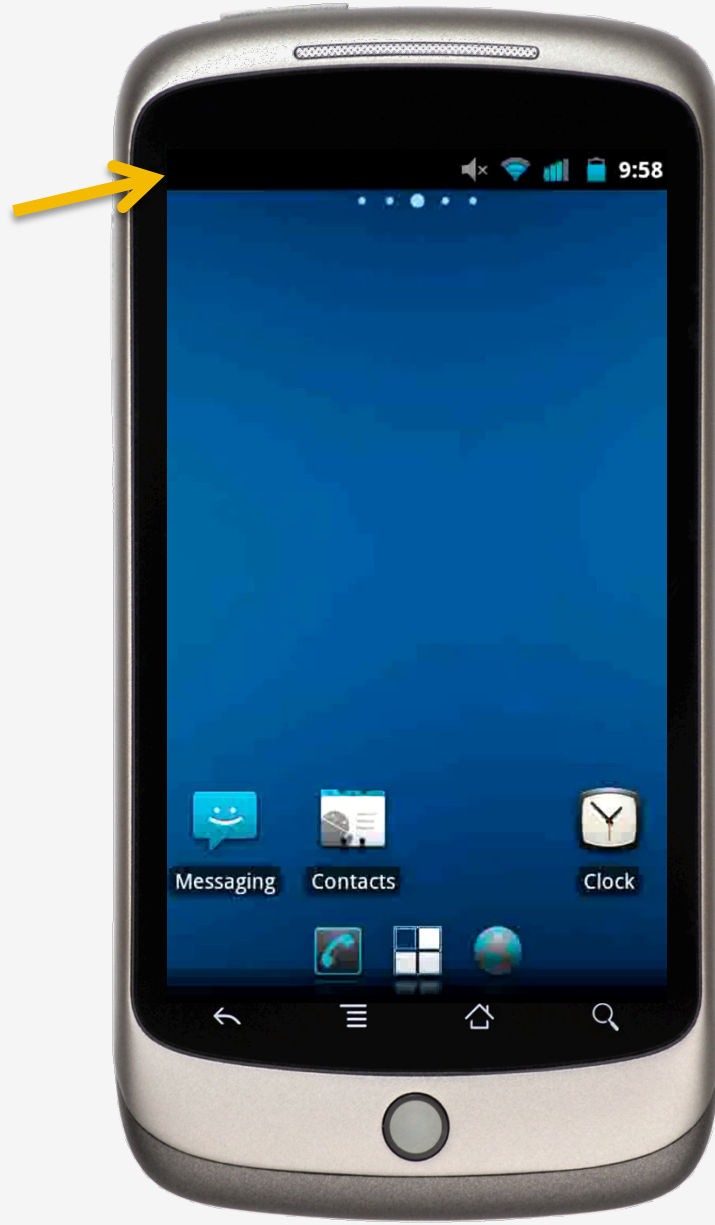
THE ANDROID ARCHITECTURE
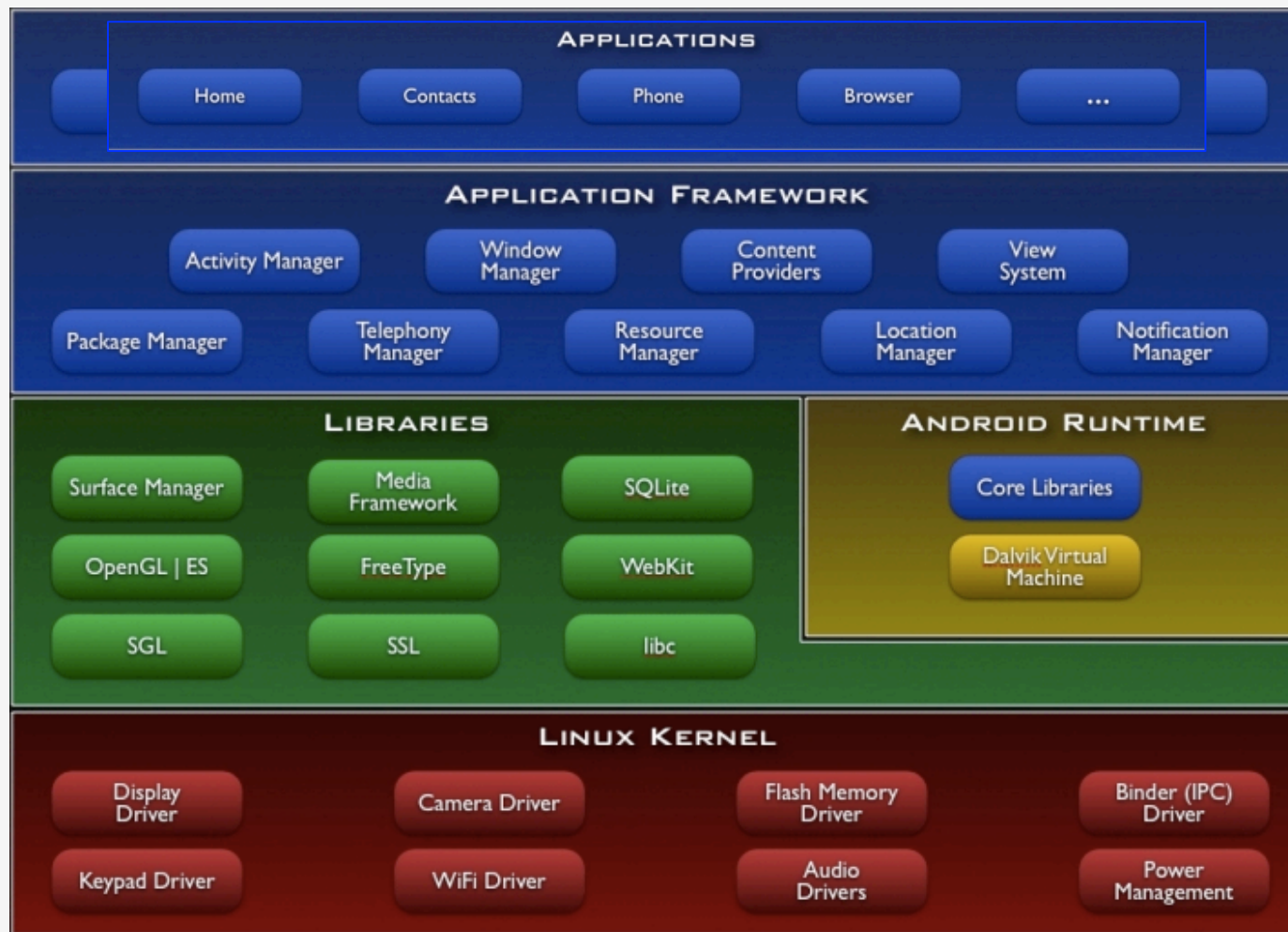
Applications

Standard apps include:

Home — main screen

Contacts — contacts database

Phone — dial phone numbers

Browser — view web pages

Email reader —compose & read email messages

Applications

Nothing special about these apps

You can substitute your own or 3rd party app for any of them

Next Time

Android Development Environment

Programming Handheld Systems

Adam Porter

The Android Development Environment

The Android Platform

A multi-layered software stack for building and running mobile applications

The Android Development Environment

Your workbench for writing Android applications

Today's Topics

Installing the Android Developer Tools (ADT) Bundle

Using the Eclipse IDE

Using the Android emulator

Debugging Android applications

Other tools

Prerequisites

Supported Operating Systems:

Windows XP, Vista, or Windows 7

Mac OS X 10.5.8 or later (x86 only)

Various Linux distributions

See: http://developer.android.com/sdk

Prerequisites

Make sure you have the Java Development Kit (JDK6) installed

See:

http:// www.oracle.com/technetwork/java/javase/downloads

Getting Started

Download & install the Android
Developer Tools (ADT) Bundle

See: http://developer.android.com/sdk

ADT Bundle

Latest Android platform

Eclipse + ADT plugin

Latest system image for emulator

Additional development tools
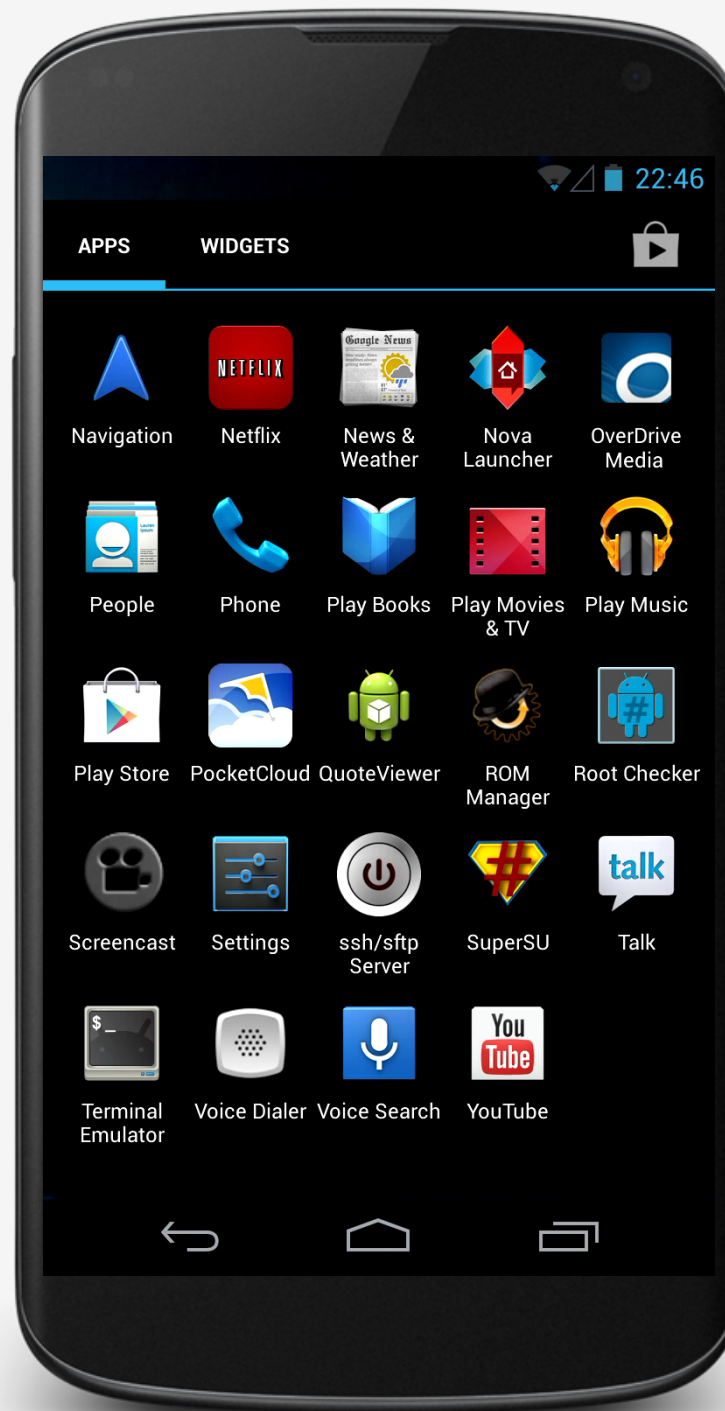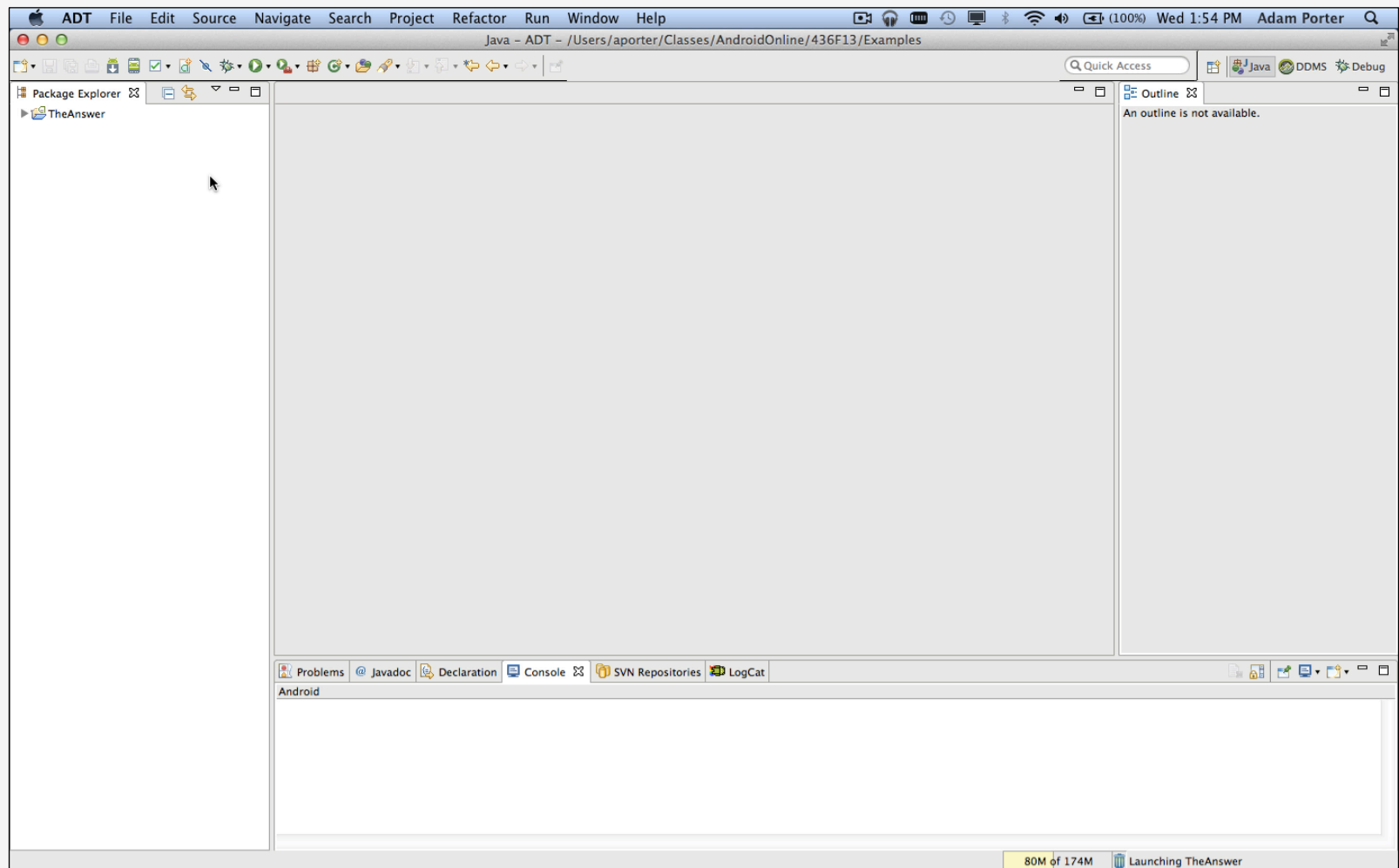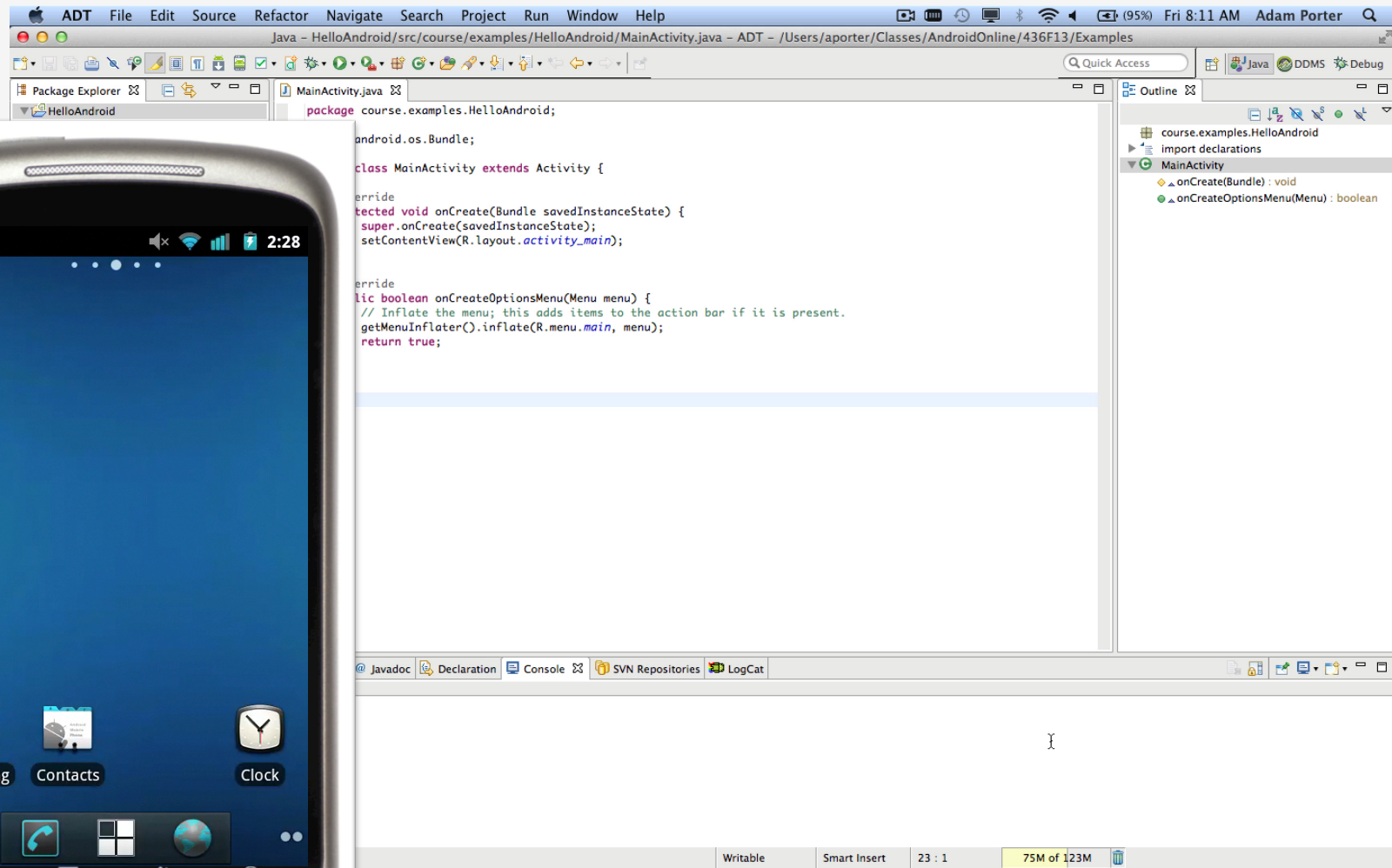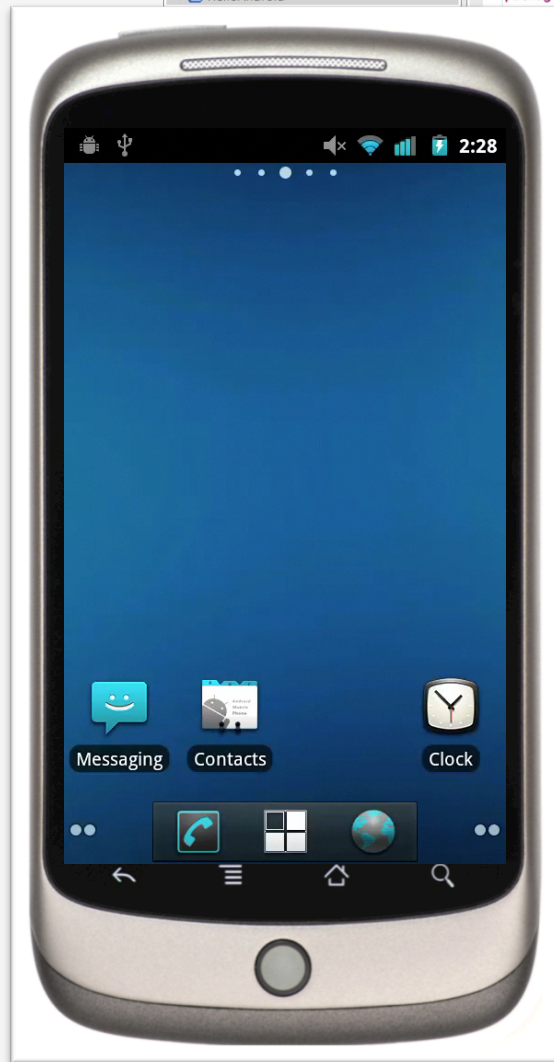
The Android Emulator

Runs virtual devices

Java – HelloAndroid/src/course/examples/HelloAndroid/MainActivity.java – ADT – /Users/aporter/Classes/AndroidOnline/436F13/Examples

Quick Access   Java   DDMS   Debug

Package Explorer ⊠
▶ HelloAndroid
▶ TheAnswer

activity_main.xml   MainActivity.java ⊠

```
package course.examples.HelloAndroid;

import android.os.Bundle;

public class MainActivity extends Activity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }

    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        // Inflate the menu; this adds items to the action bar if it is present.
        getMenuInflater().inflate(R.menu.main, menu);
        return true;
    }

}
```

Outline ⊠
course.examples.HelloAndroid
▶ import declarations
▼ MainActivity
    ○ onCreate(Bundle) : void
    ○ onCreateOptionsMenu(Menu) : boolean

Problems   @ Javadoc   Declaration   Console ⊠   SVN Repositories   LogCat

Android

55M of 115M

The Android Emulator

Pros

- Doesn't require an actual phone
- Hardware is reconfigurable
- Changes are non-destructive

The Android Emulator

Cons

Can be very slow

Some features unavailable

- e.g., no support for bluetooth or USB connections

Performance / user experience can be misleading

Advanced Features

Can emulate many different device/ user characteristics, such as:

- Network speed/latencies
- Battery power
- Location coordinates
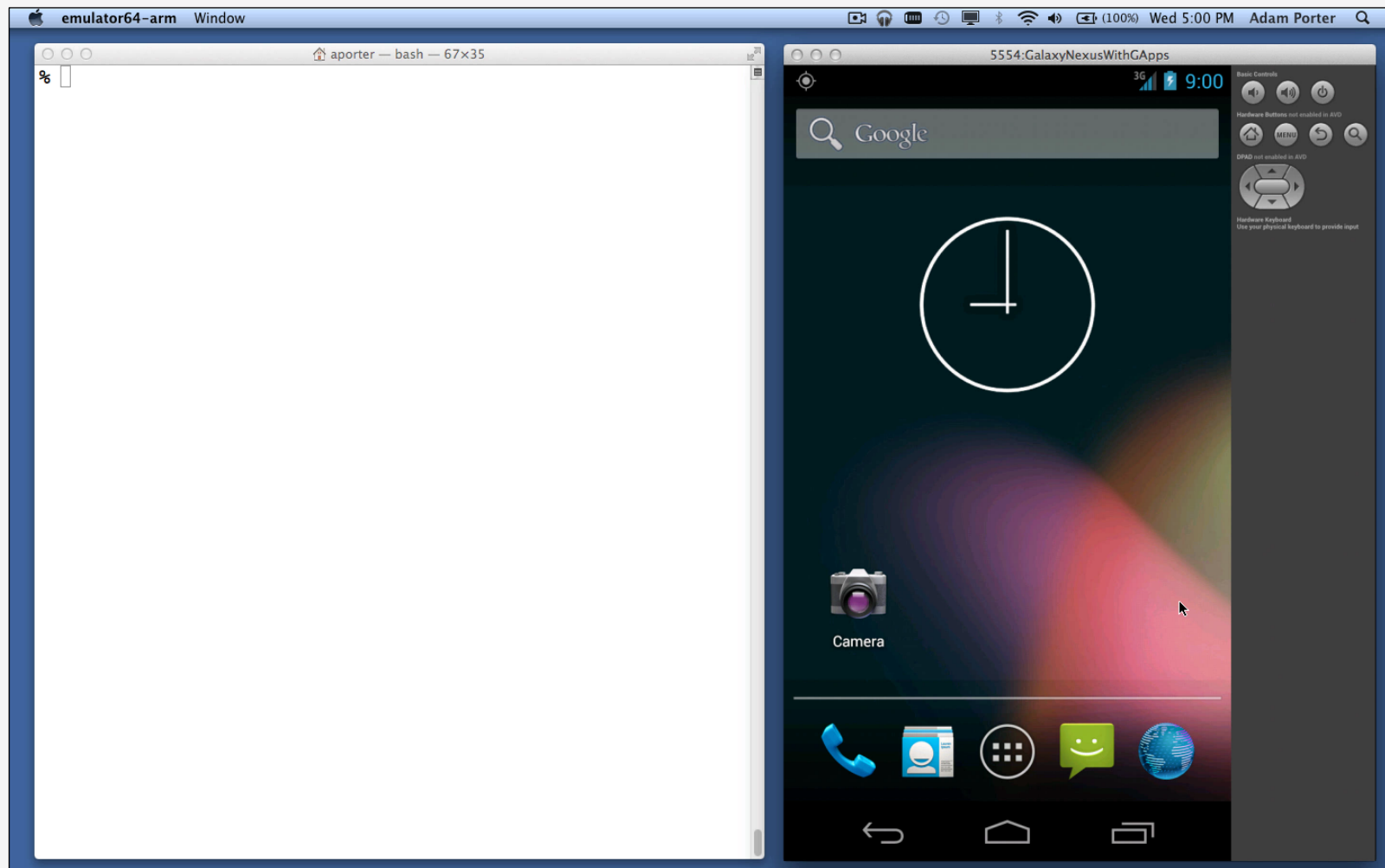
Advanced Features

Emulate incoming phone calls &
SMS messages
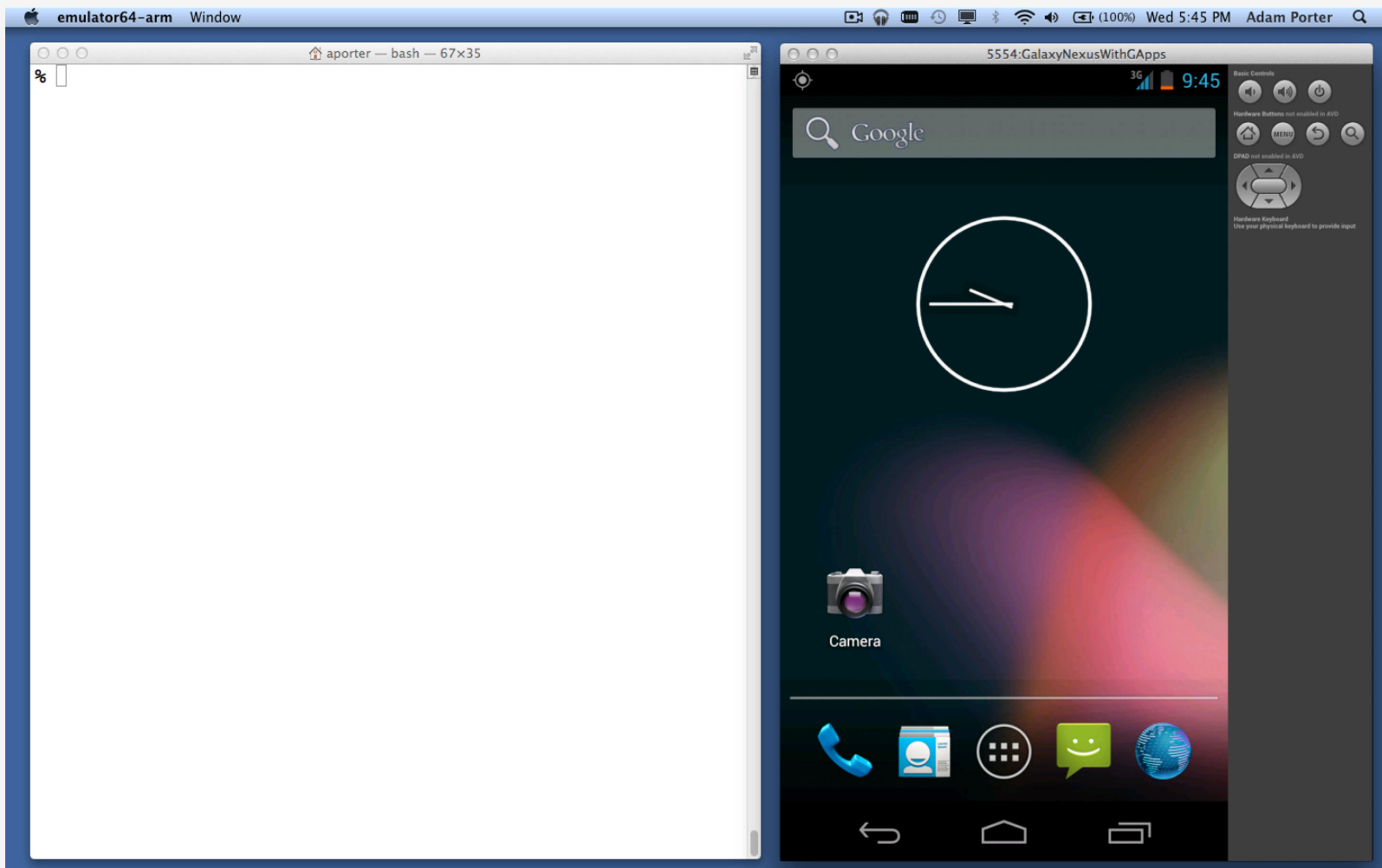
The Android Emulator

Can interconnect multiple emulators
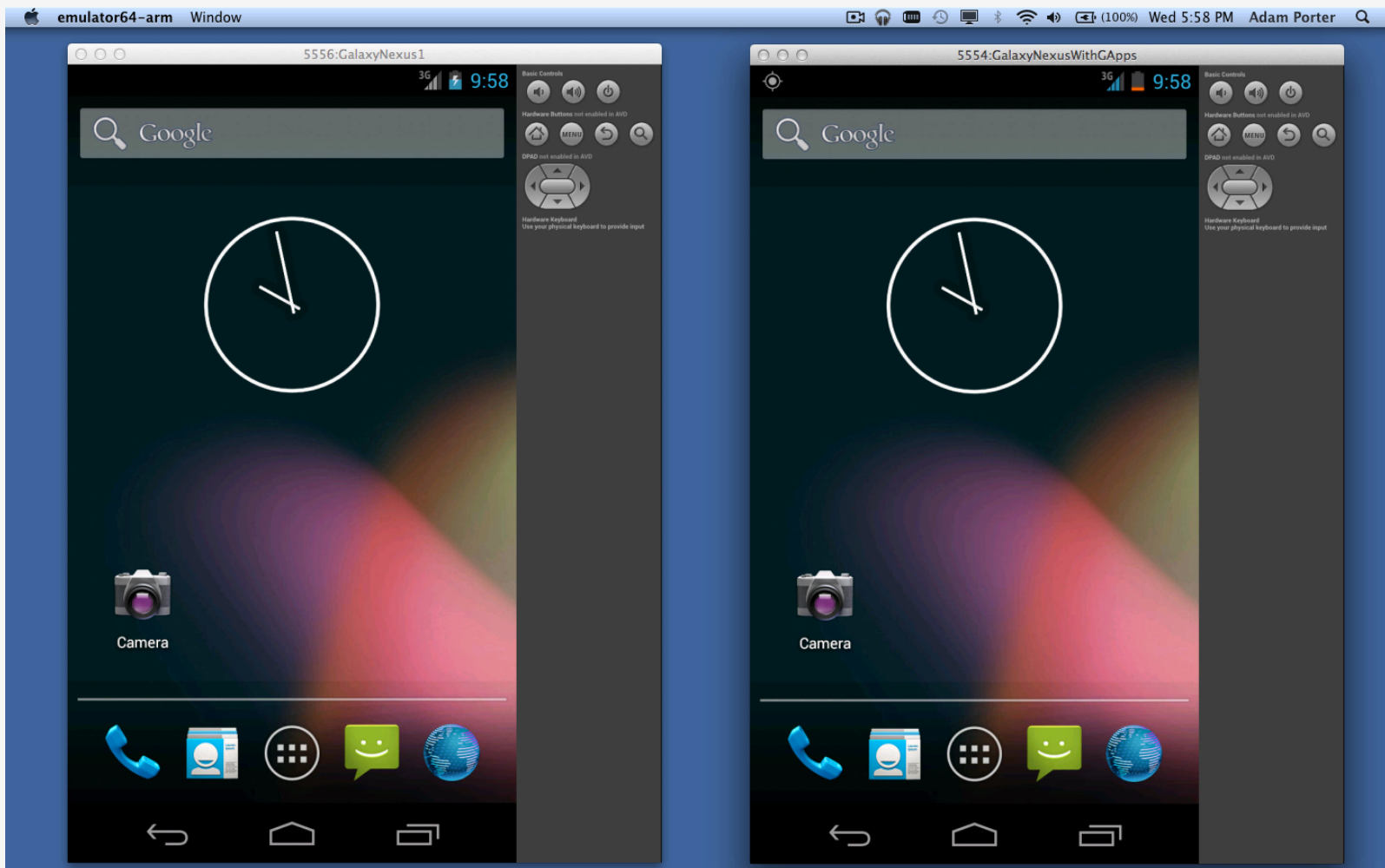
Advanced Features

Many more options

See:

http://developer.android.com/guide/
developing/devices/emulator.html

Debugger

Tool for examining the internal state of a running application
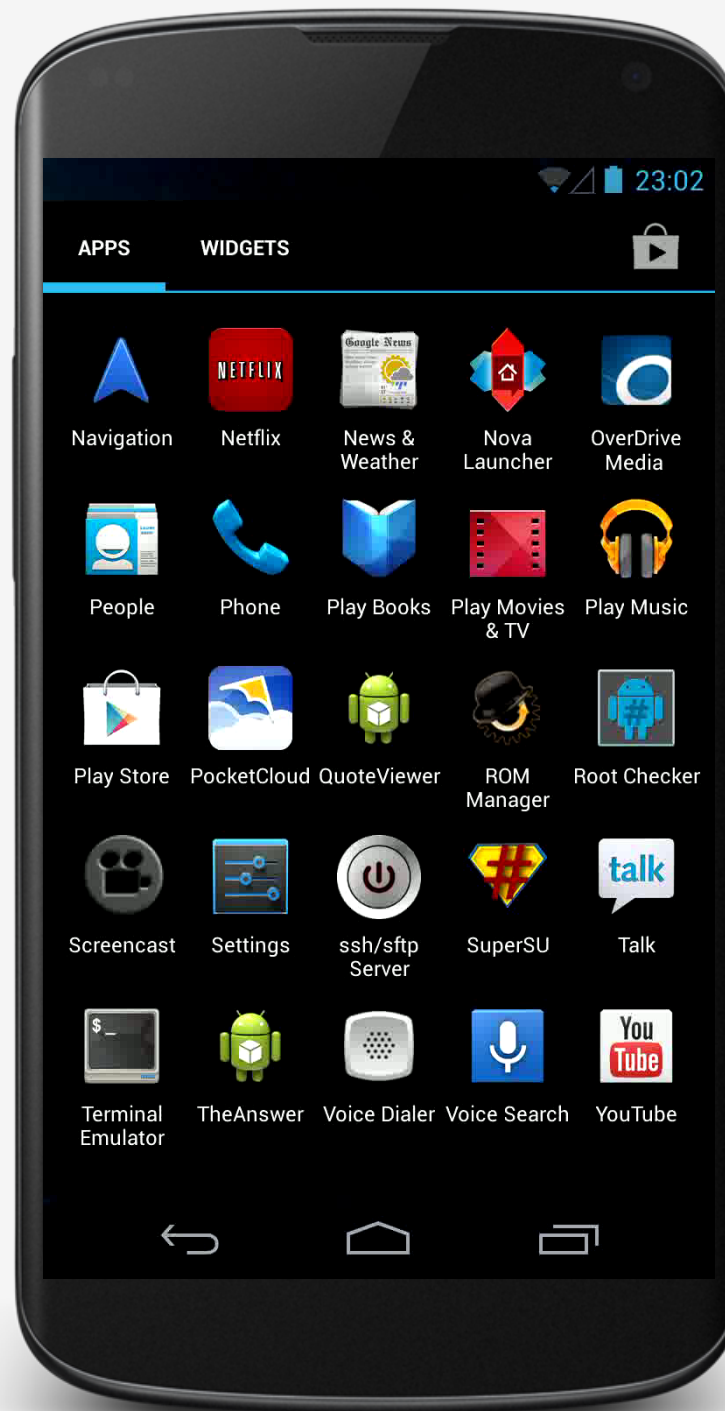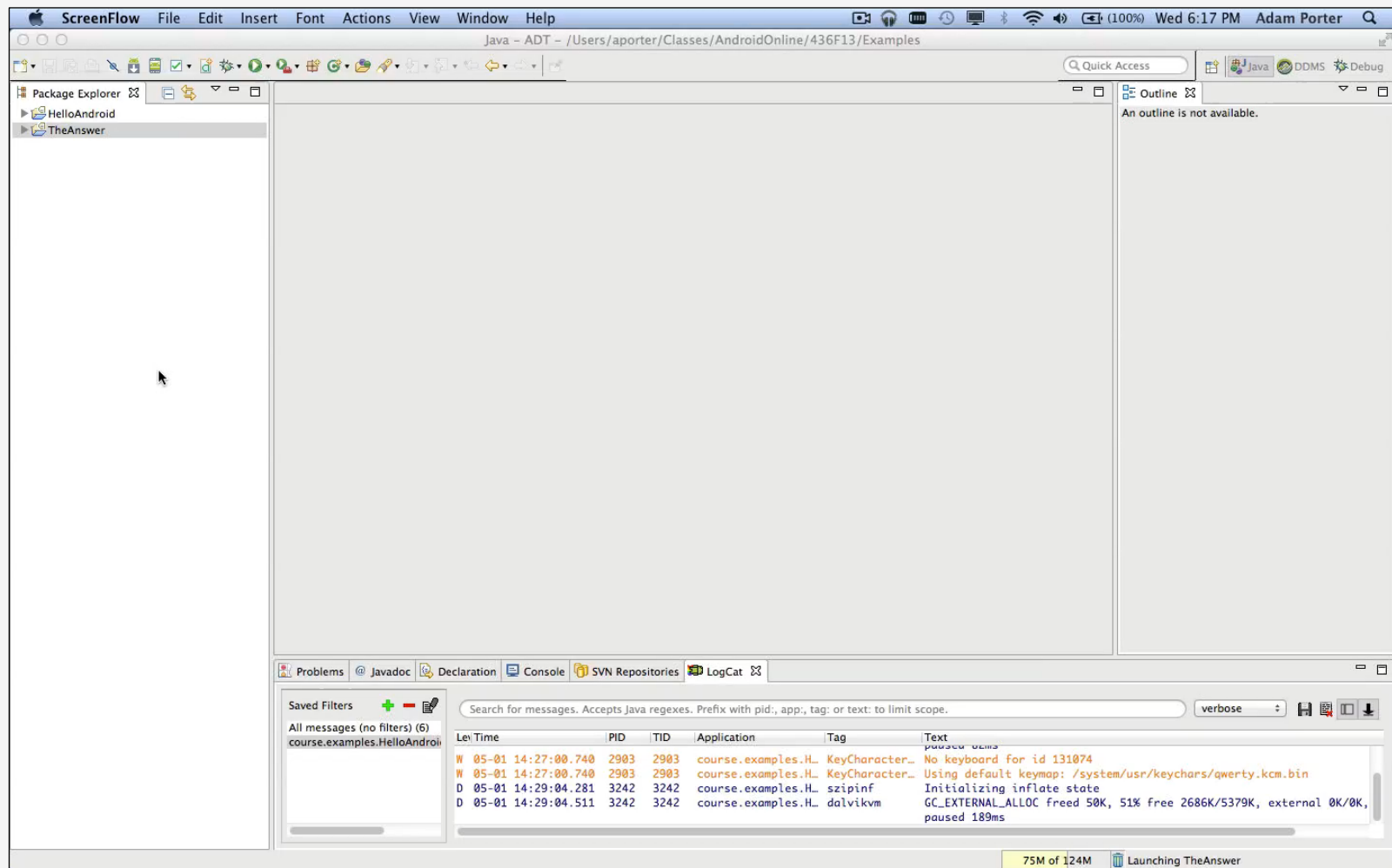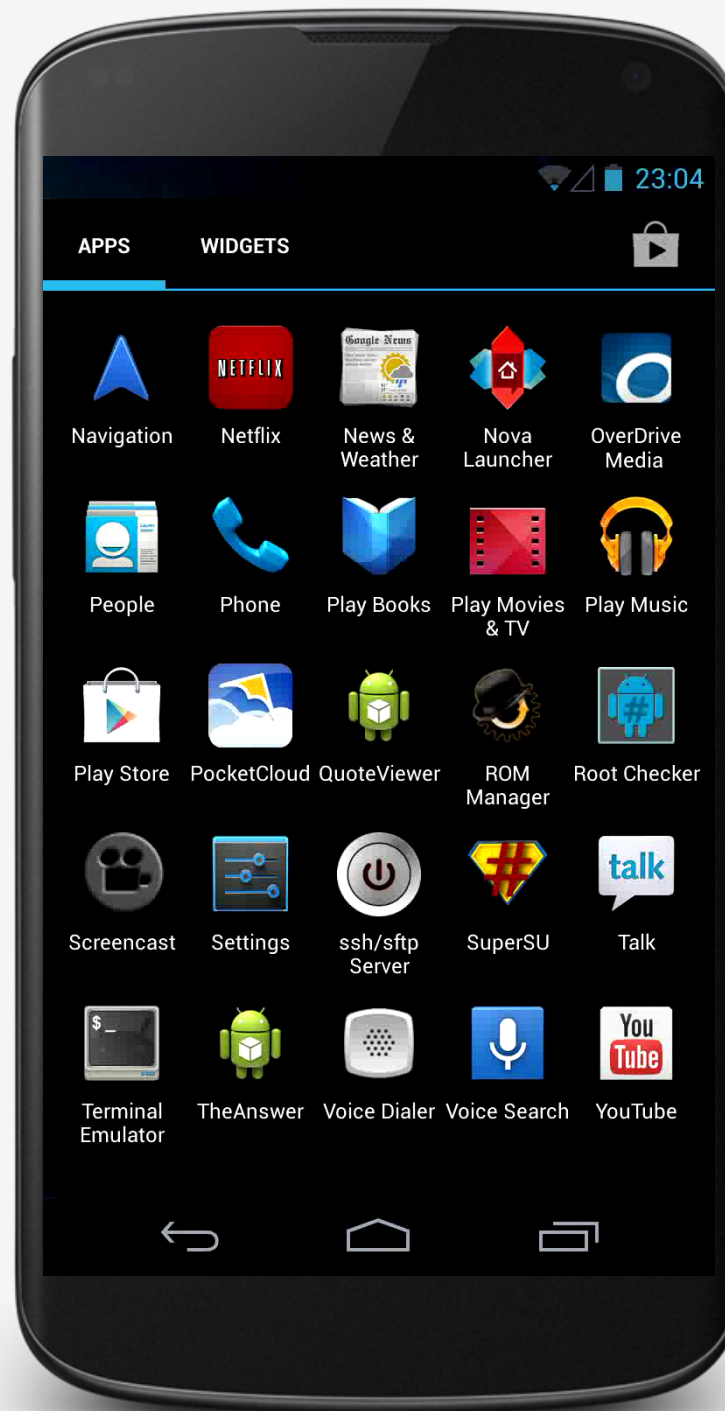
Dalvik Debug Monitor Service (DDMS)

General tools for monitoring application behaviors

Dalvik Debug Monitor Service (DDMS)

DDMS includes

File Explorer

Logcat

Traceview

Hierarchyview

Java – ADT – /Users/aporter/Classes/AndroidOnline/436F13/Examples

Quick Access   Java DDMS Debug

Package Explorer

- ▶ HelloAndroid
- ▶ TheAnswer

Outline

An outline is not available.

Problems @ Javadoc Declaration Console SVN Repositories LogCat

Android

55M of 113M
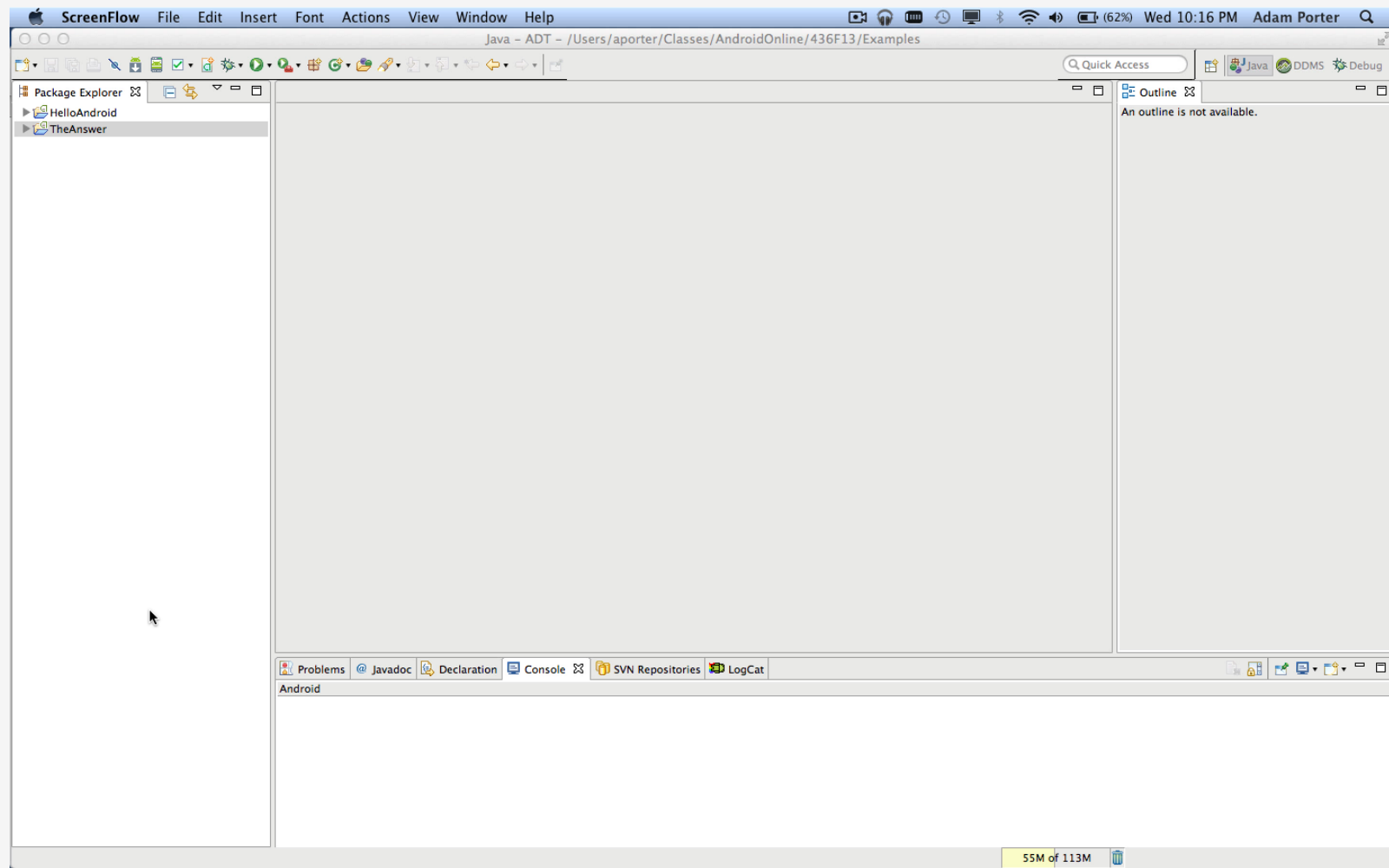
Traceview

Graphically display method traces taken from running application

Java – TheAnswer/src/course/examples/TheAnswer/TheAnswer.java – ADT – /Users/aporter/Classes/AndroidOnline/436F13/Examples

```
Q Quick Access          🖳 Java  DDMS  Debug
```

Package Explorer ⬚

- ▶ HelloAndroid
- ▼ TheAnswer
  - ▼ src
    - ▼ course.examples.TheAnswer
      - ▶ TheAnswer.java
  - ▶ gen [Generated Java Files]
  - ▶ Android 4.2.2
  - assets
  - ▶ bin
  - ▶ res
  - AndroidManifest.xml
  - proguard.cfg
  - project.properties

TheAnswer.java ⬚

```java
package course.examples.TheAnswer;

import android.app.Activity;

public class TheAnswer extends Activity {

    private static final int [] answers = {42,-10,0,10, 100,1000};
    private static final int answer = 42;

    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);

        int val = findAnswer();
        String output = (val == answer) ? String.valueOf(answer) : "We may never know....";

        TextView tv = new TextView(this);
        tv.setText("The answer to Life, the Universe and Everything is:\n" + output);

        setContentView(tv);
    }

    private int findAnswer() {
        for (int val : answers) {
            if (val == answer) return val;
        }
        return -1;
    }
}
```

Outline ⬚

- ⊞ course.examples.TheAnswer
- ▶ import declarations
- ▼ ⊙ TheAnswer
  - S F answers : int[]
  - S F answer : int
  - ● onCreate(Bundle) : void
  - ■ findAnswer() : int

Problems @ Javadoc Declaration ▣ Console ⬚ SVN Repositories LogCat

Android

```
[2013-05-01 22:32:35 - TheAnswer] ------------------------------
[2013-05-01 22:32:35 - TheAnswer] Android Launch!
[2013-05-01 22:32:35 - TheAnswer] adb is running normally.
[2013-05-01 22:32:35 - TheAnswer] Performing course.examples.TheAnswer.TheAnswer activity launch
[2013-05-01 22:32:35 - TheAnswer] Automatic Target Mode: Unable to detect device compatibility. Please select a target device.
[2013-05-01 22:32:39 - TheAnswer] Application already deployed. No need to reinstall.
[2013-05-01 22:32:39 - TheAnswer] Starting activity course.examples.TheAnswer.TheAnswer on device HT03RP901884
[2013-05-01 22:32:40 - TheAnswer] ActivityManager: Starting: Intent { act=android.intent.action.MAIN cat=[android.intent.category.LAUNCHER] cmp=course.examples.TheAn
[2013-05-01 22:32:41 - TheAnswer] Attempting to connect debugger to 'course.examples.TheAnswer' on port 8614
```

Writable   Smart Insert   1 : 1   72M of 129M   Launching TheAnswer

UI Hierarchy Viewer

Shows the runtime organization of the user interface

ADT   File   Edit   Source   Refactor   Navigate   Search   Project   Run   Window   Help   (25%)   Wed 11:15 PM   Adam Porter

Java – TheAnswer/src/course/examples/TheAnswer/TheAnswer.java – ADT – /Users/aporter/Classes/AndroidOnline/436F13/Examples

Quick Access   Java   DDMS   Debug

Package Explorer ⊠
- ▶ HelloAndroid
- ▶ TheAnswer

TheAnswer.java ⊠

Outline ⊠
- course.examples.TheAnswer
- ▶ import declarations
- ▼ TheAnswer
  - answers : int[]
  - answer : int
  - onCreate(Bundle) : void
  - findAnswer() : int

```java
package course.examples.TheAnswer;

import android.app.Activity;

public class TheAnswer extends Activity {

    private static final int [] answers = {42,-10,0,10, 100,1000};
    private static final int answer = 42;

    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);

        int val = findAnswer();
        String output = (val == answer) ? String.valueOf(answer) : "We may never know....";

        TextView tv = new TextView(this);
        tv.setText("The answer to Life, the Universe and Everything is:\n" + output);

        setContentView(tv);
    }

    private int findAnswer() {
        for (int val : answers) {
            if (val == answer) return val;
        }
        return -1;
    }
}
```

Problems   @ Javadoc   Declaration   Console ⊠   SVN Repositories   LogCat

Android

65M of 122M

Next Time

Application Fundamentals