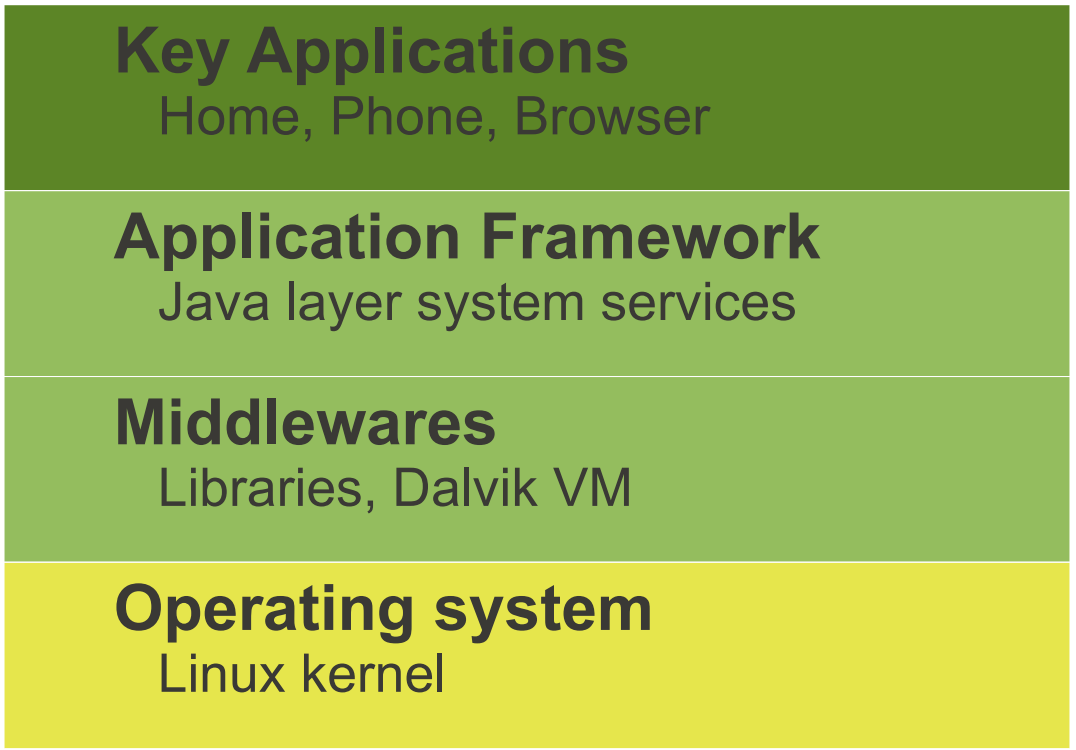
I. Android Internals

Android platform

Android is a Linux-based operating system and a software stack designed for mobile devices



Android platform

Developed by **Google***

- 2005 – purchased Android Inc
- 2007 – established Open Handset Alliance (OHA)
- 2012 – Oracle v. Google on Java API

Compatibility Definition Document

to certify devices

SDK + tools

to create applications

Android platform

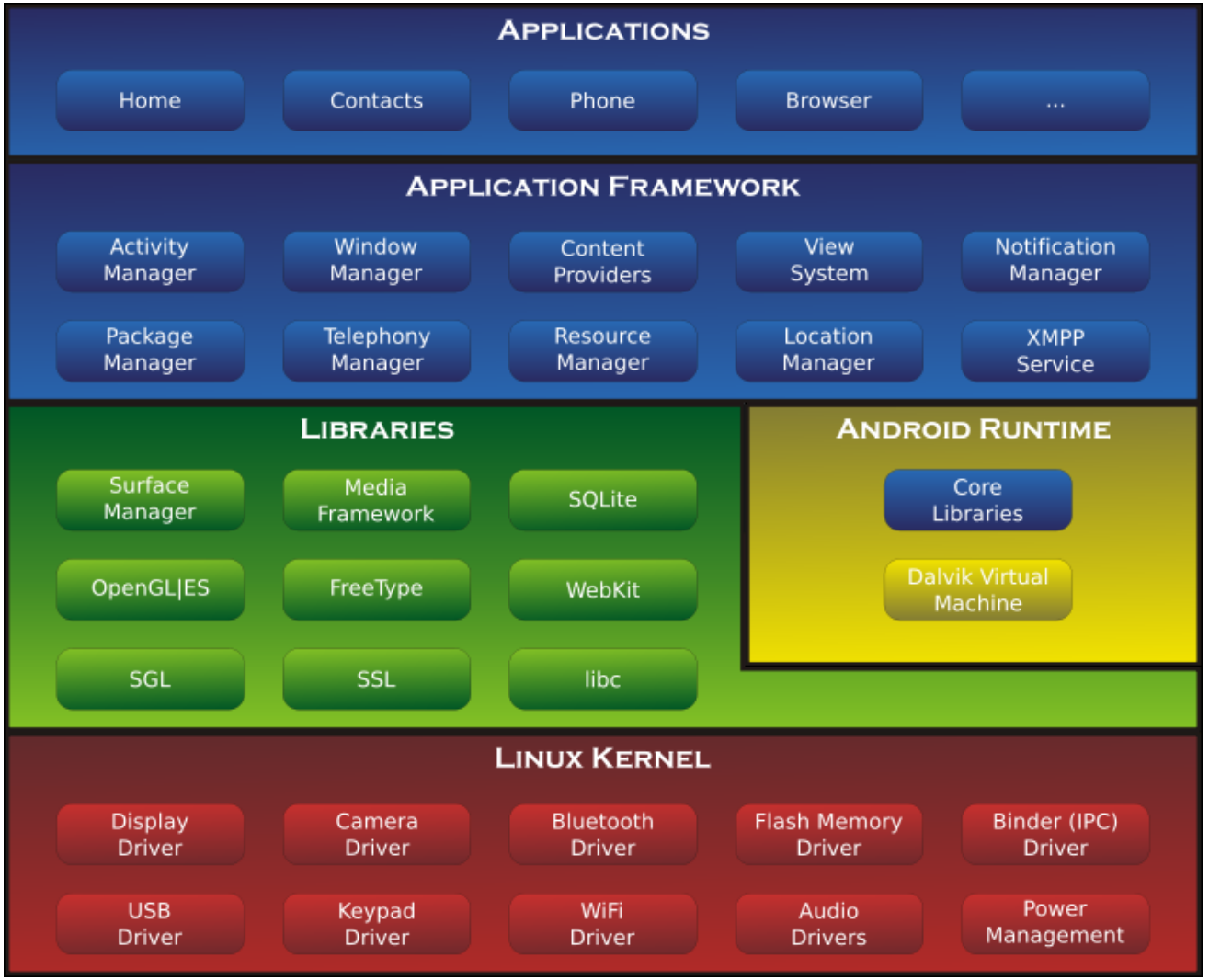
Main hardware platform for Android is **ARM**

- support for x86 from the Android x86 project

Advanced RISC Machine

- 7.9 billion shipped in 2011
  - 95% smartphones
  - 90% hard disk drives
- Reduced instruction set computing (RISC)
- Lower costs, Less heat, Less power usage
- Reduced complexity and simpler design
- Good interfacing to many devices

Android Anatomy



Android Linux Kernel

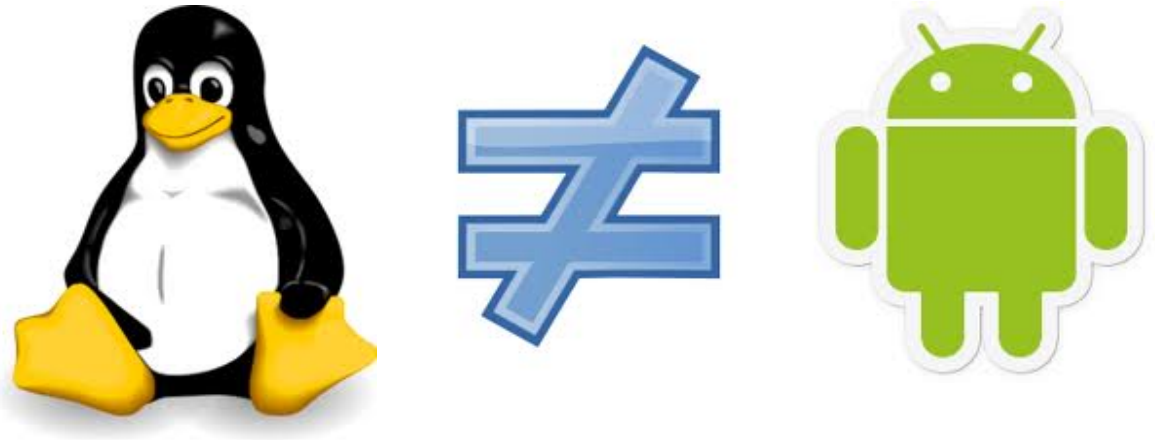
Android kernel based on **Linux** 2.6 and 3.x (Android 4.0)

- Why Linux?
  - Great memory and process management
  - Permissions-based security model
  - Proven driver model
  - Support for shared libraries
  - Open source

Android Linux Kernel

Android “**is not Linux**”

- Users never see Linux sub system
- Not the full set of standard Linux utilities
- August 2011, Linus Torvalds - "eventually Android and Linux would come back to a common kernel, but it will probably not be for four to five years"



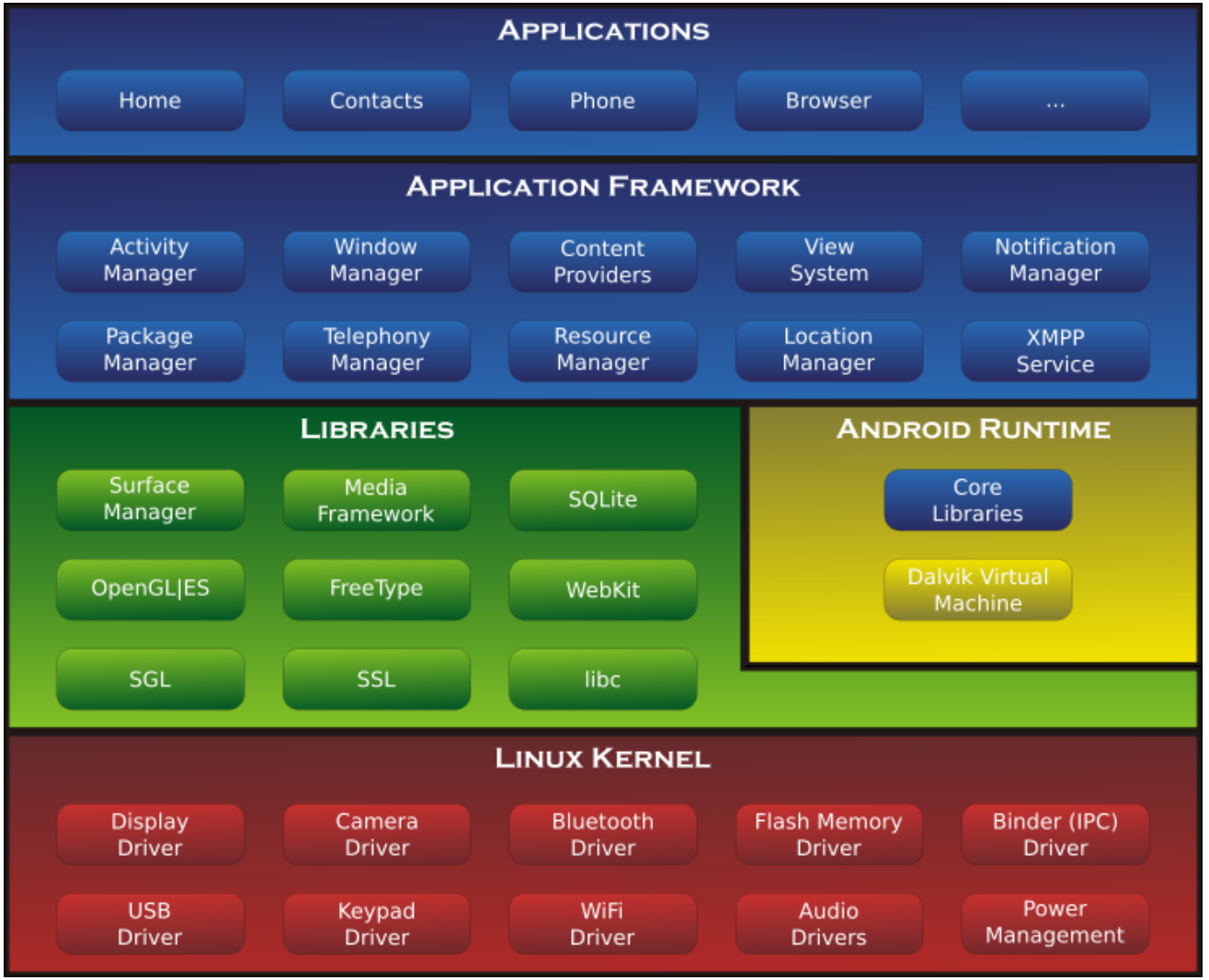
Android Linux Kernel

Android **patch** of kernel enhancements

- Power Management
- Ashmem
- Binder IPC
- Low Memory Killer

December 2011 - Android Mainlining Project to put some Android drivers, patches and features back into the Linux kernel

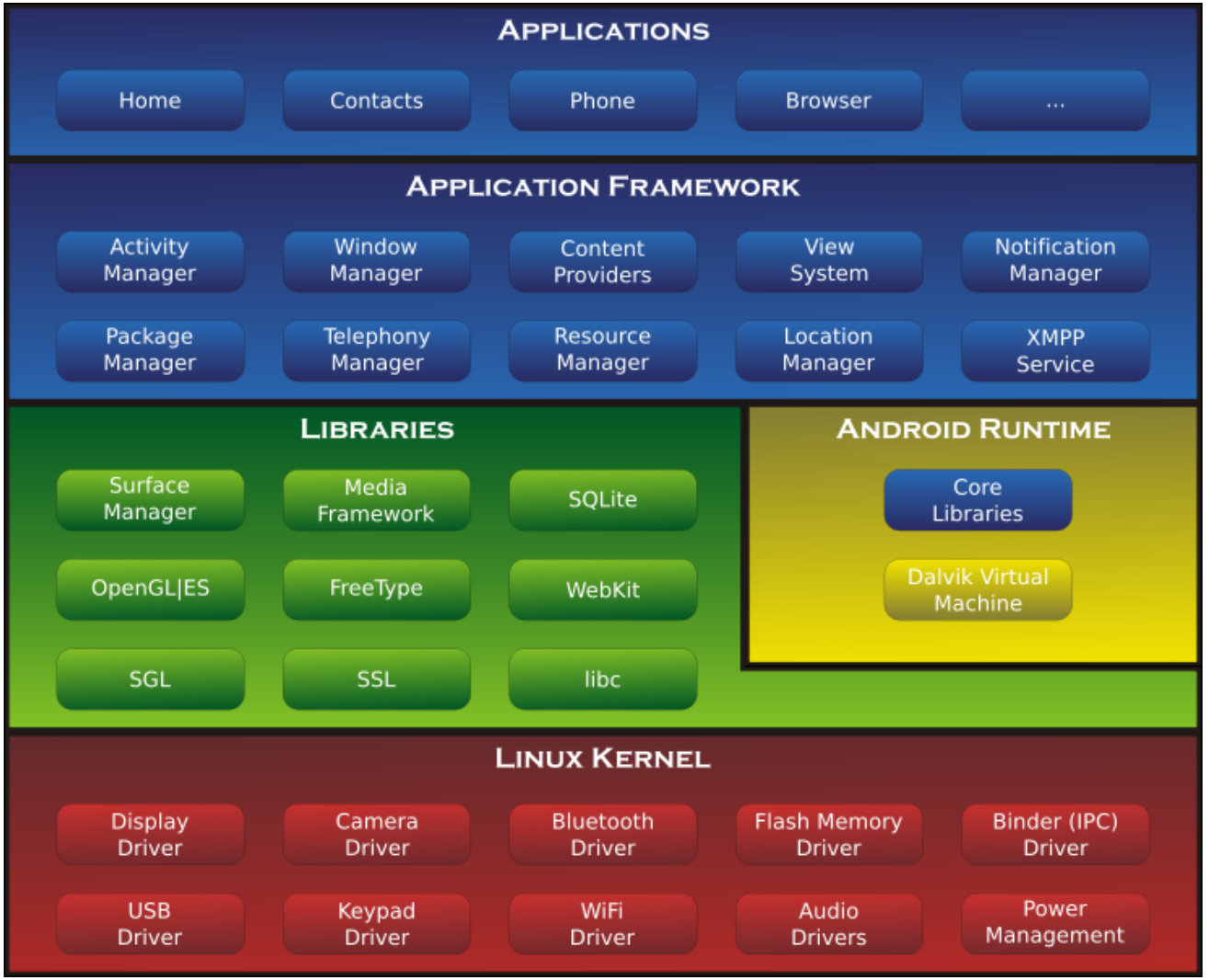
Android Native Libraries



Android Native Libraries

- **Bionic Libc**
  - Fast and small, license-friendly, optimized for mobile
- **Function Libraries**
  - WebKit, Media Framework, SQLite
- **Native Servers**
  - Surface Manager, Audio Manager
- **Hardware Abstraction Libraries**
  - WiFi, Bluetooth, GPS, Camera

Android Runtime



Android Runtime

On Android you develop in **Java**

... but Android does not run Java Bytecode !

Android Runtime

Custom Virtual Machine. Why?

Constraints	**Oracle JVM problems**
Designed for 64M RAM No swap Phone tech v. Desktop CPU specific	Memory Hog Slow startup IP licensing restrictions

Android Runtime

Dalvik Virtual Machine

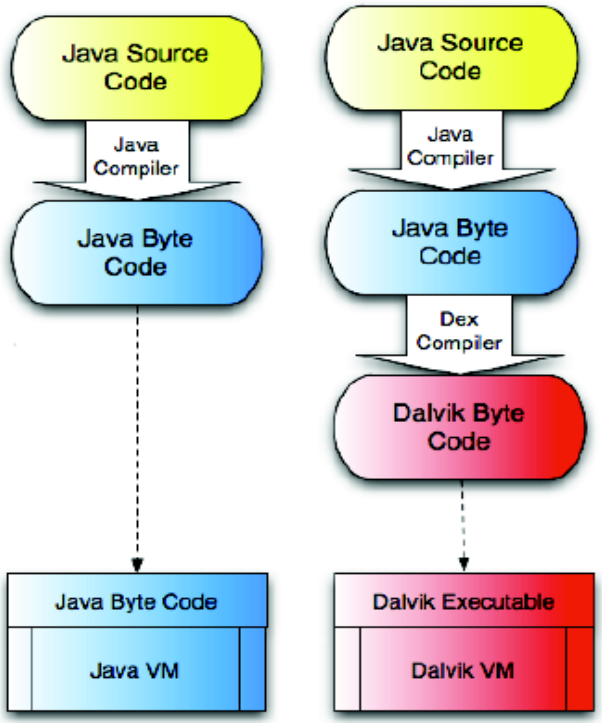
- Custom VM optimized for mobile devices
- Register-based JVM
- More efficient and compact
- Use memory efficiently
- Dalvik Executable Code (.dex)
  - 30% fewer instructions
  - 35% fewer code units
- Trace JIT compiler (since 2.2)

Android Runtime

Android Java = Java language + Dalvik + Apache Harmony

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

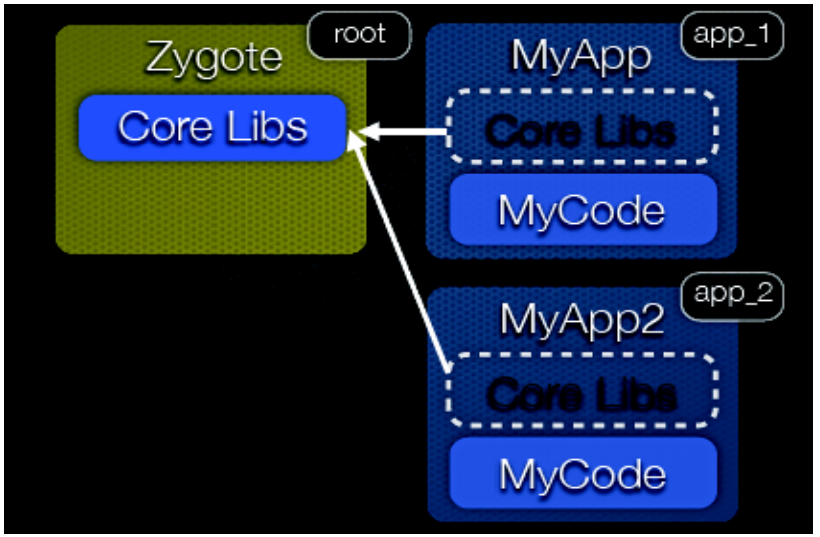
Sun-Java = Java language + JVM + JDK



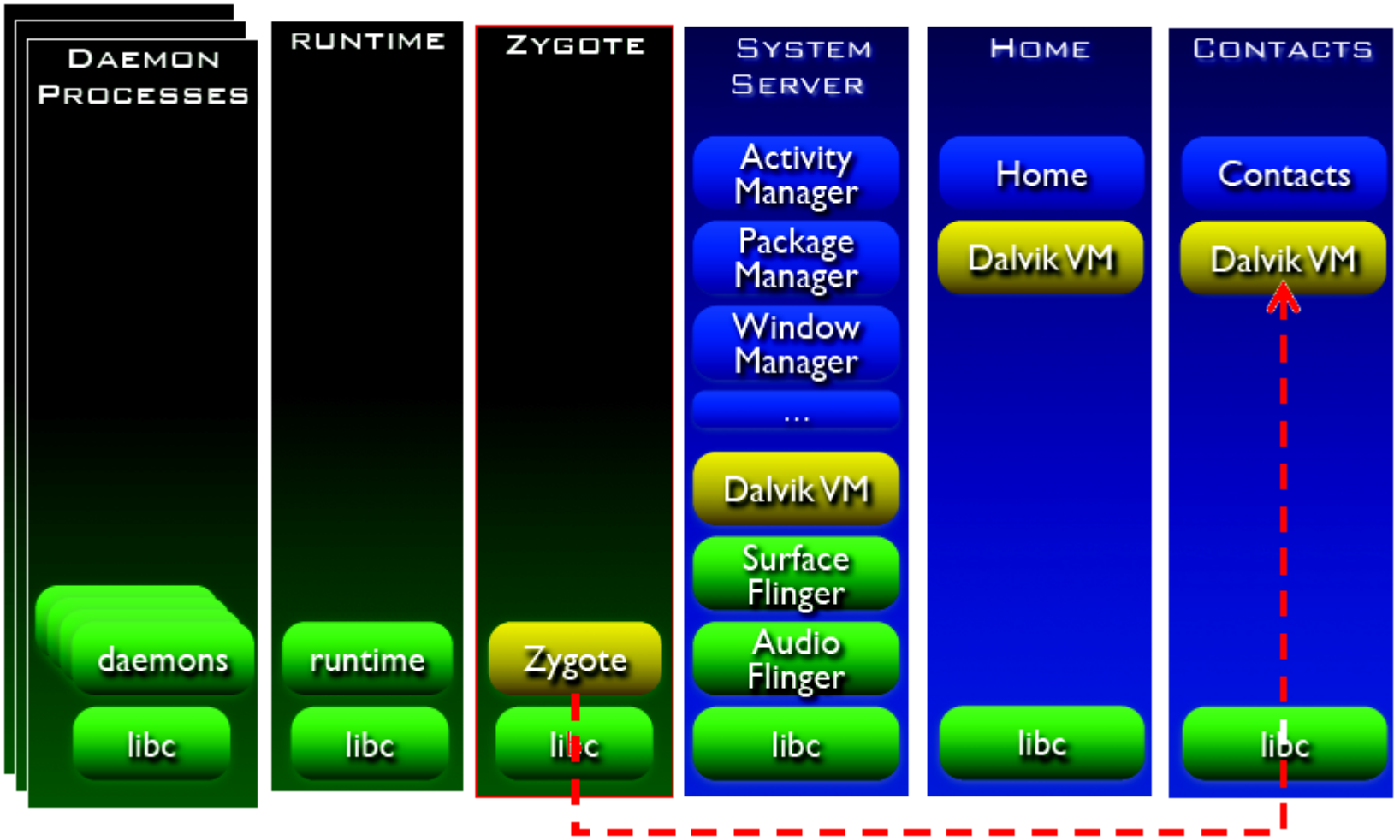
Android Runtime

Zygote spawns VM processes

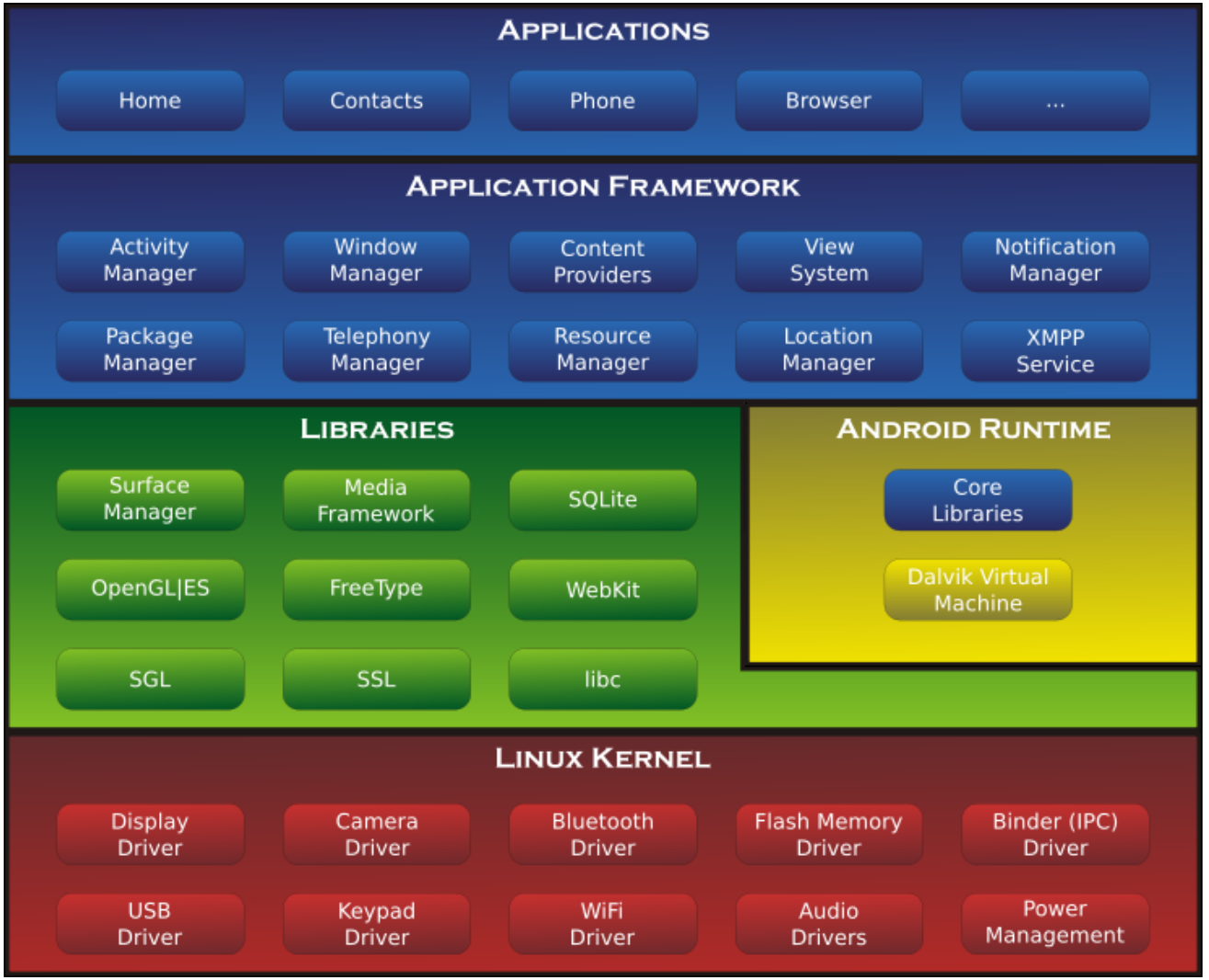
- Already has core libraries loaded
- When an app is launched, zygote is forked
- Fork core libraries are shared with zygote
- Solves HotSpot start-up performance and memory problems



Android Runtime



Android Runtime



Android Runtime

- **Core libraries**
  - Core APIs for Java language (IO, Structures, ...)
- **Core Platform Services, Application Framework**
  - Activity Manager, Package Manager, Window Manager, Resource Manager, Content Providers, View System
- **Hardware Services**
  - Telephony, Location, Bluetooth, WiFi, USB, Sensor

II. Android Developer Toolchain

Android offer

The first **open* and comprehensive** platform for mobile devices

- an operating system, user-interface and applications
- a mobile phone without the proprietary obstacles that have hindered mobile innovation
- an open-source development platform

Android offer

- Apps are equivalent parts of the phone using the same API
- Reuse and replacement of application components
- GSM, EDGE, 3G, 4G, LTE data transfer
- Comprehensive location-based APIs
- Wi-Fi hardware access and peer-to-peer connection

Android offer

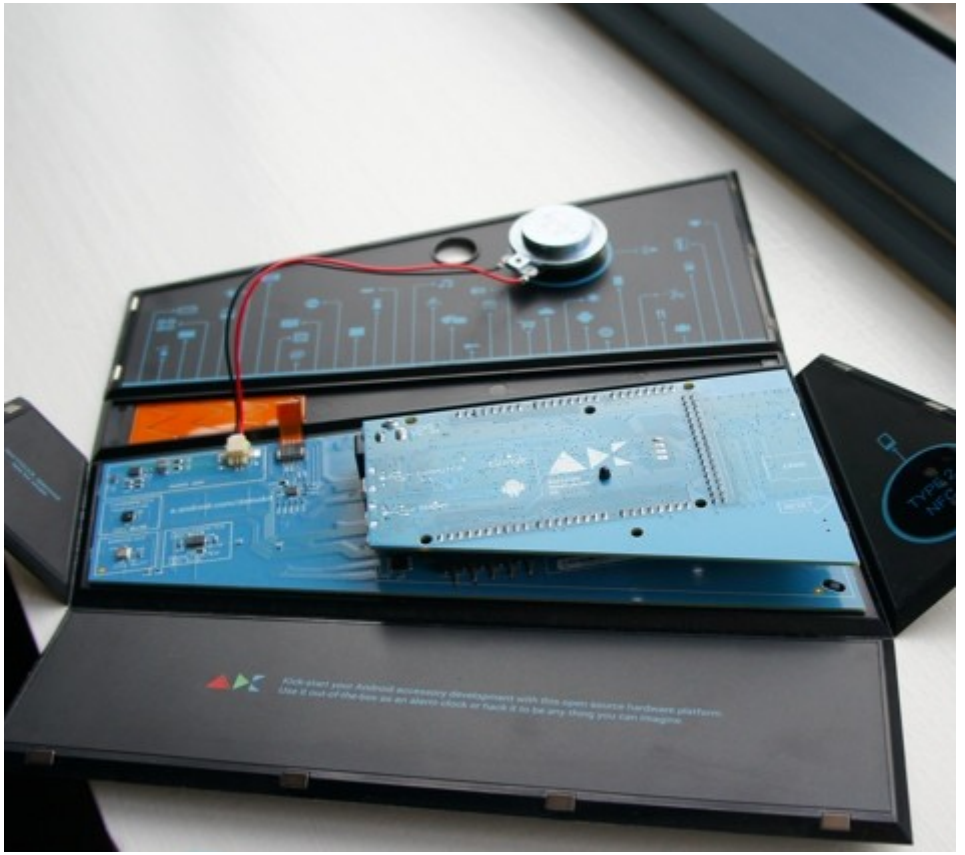
- Full multimedia hardware control, media libraries
- APIs for sensor hardware
- Libraries for Bluetooth and NFC
- Shared data stores and APIs for contacts, social networking
- Background Services, applications, and processes
- Home-screen Widgets and Live Wallpaper

Android Developer Challenge

- A developer challenge Google hold every year for Android mobile platform
- Demonstrates the capabilities of Android platform
- Developers submit their apps to one of 10 specially-designated ADC categories
- Top 20 applications in each category will proceed to the second round
- Community vote constituting 40% of the final judging score

Accessory Development Kit

- RI for building accessories for Android
- Source code and hardware specifications
- ADK 2012 is based on the Arduino open source electronics prototyping platform



Install Android

What do we need for Android?

- Android SDK
- ADT (Android Developer Tools)
- AVD (Android Virtual Device)

Android SDK

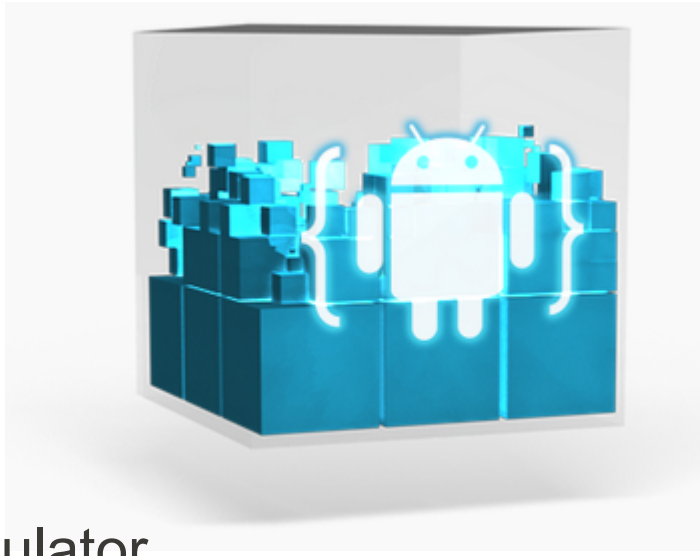
- Android APIs

  Developer access to the Android stack
- Development tools

  Compile and debug your applications
- Android Virtual Device Manager
- Emulator
- Documentation
- Sample code

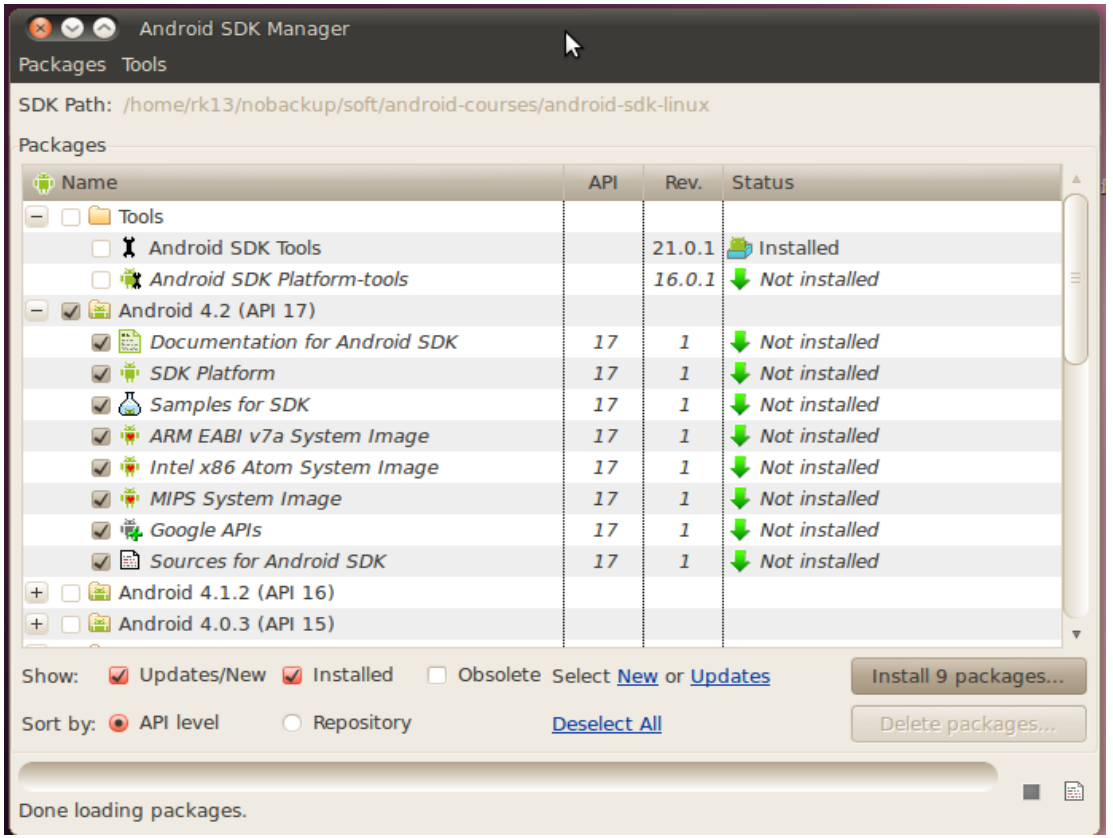
Android SDK

- http://developer.android.com/sdk (Linux, Mac OS X, Windows)
- **Installation Option1: SDK Tools Only**
  - Initially contains only the basic SDK tools
  - Need to install the Platform-tools and Android platform(s)
- **Installation Option2: ADT Bundle**
  - Eclipse + ADT plugin
  - Android SDK Tools
  - Android Platform-tools
  - The latest Android platform
  - The latest Android system image for the emulator



Android SDK

SDK Manager.exe (Windows) v. **tools/android** (Mac, Linux)



Android SDK

SDK **packages**

<sdk>/tools/	Tools for debugging and testing, utilities to develop an app (sdk manager, ddms, emulator)
<sdk>/platform-tools/	Platform-dependent tools for developing, support latest features of the Android platform and are updated with new platform (adb, fastboot, aapt)
<sdk>/docs/ **<sdk>/sources/**	Latest documentation for the Android APIs
<sdk>/platforms/	a) SDK Platform for each version of Android, android.jar with a fully compliant Android library b) System images for Android emulator
<sdk>/add-ons/ **<sdk>/extras/**	Special APIs and static libraries you can include in your app sources in order to use powerful APIs that aren't available in the standard platform

Android Development Tools

- Eclipse plugin
- Professional-grade development environment
- Free, open-source, and runs on most major OS
- Provides the emulator




Android Development Tools

- Tooling
  - Wizard for creating new project
  - WYSWYG UI Designer
  - Code / XML editor
- Integrated with
  - AVD Manager
  - DDMS

ADT Installation Options

- **ADT Bundle**

  provides everything you need to start - Eclipse IDE with built-in ADT

  Eclipse Juno Release

- **Existing IDE**

  Download ADT plugin

  - Help > Install New Software.
  - Add Repository

    https://dl-ssl.google.com/android/eclipse/

  Configure ADT plugin

Android Emulator

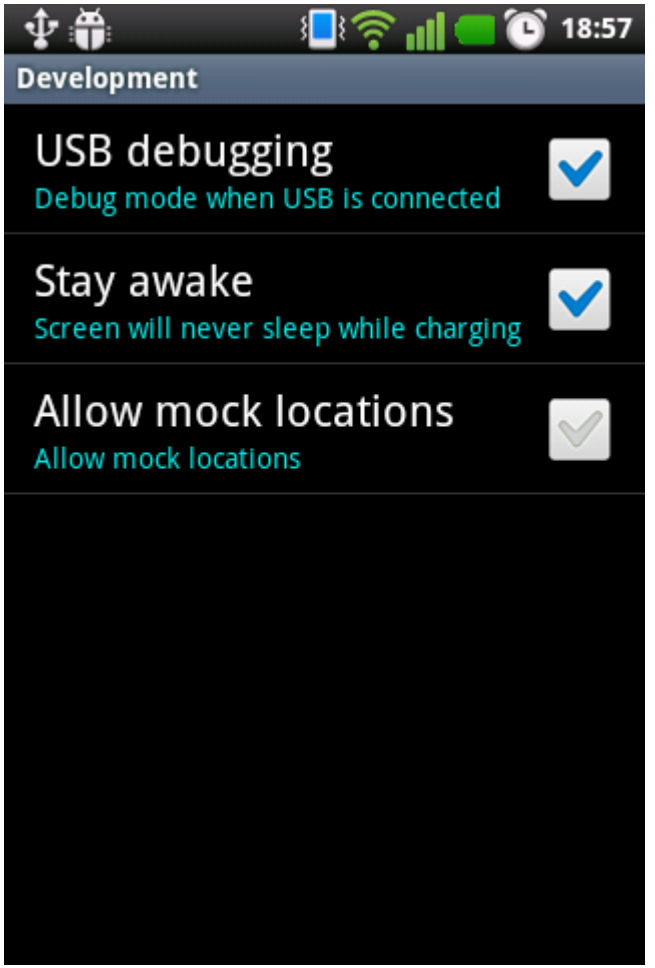
- QEMU-based
- Same image as on device
- Same toolchain to work with device or emulator
- Mimics all of the hardware and software features of a typical mobile device
- Startup may be slow, try “Graphics Acceleration” and “Virtual Machine Acceleration”, or use a device instead

Android Virtual Device

- Android Virtual Device (AVD) is an emulator configuration to model an actual device by defining hardware and software options
  - Hardware profile - hardware features (camera, physical, memory)
  - Mapping to a system image - version of the Android platform will run on the virtual device
  - Dedicated storage area - the device's user data, emulated SD card
  - Other options - emulator skin, emulated SD card

Develop on Hardware Devices

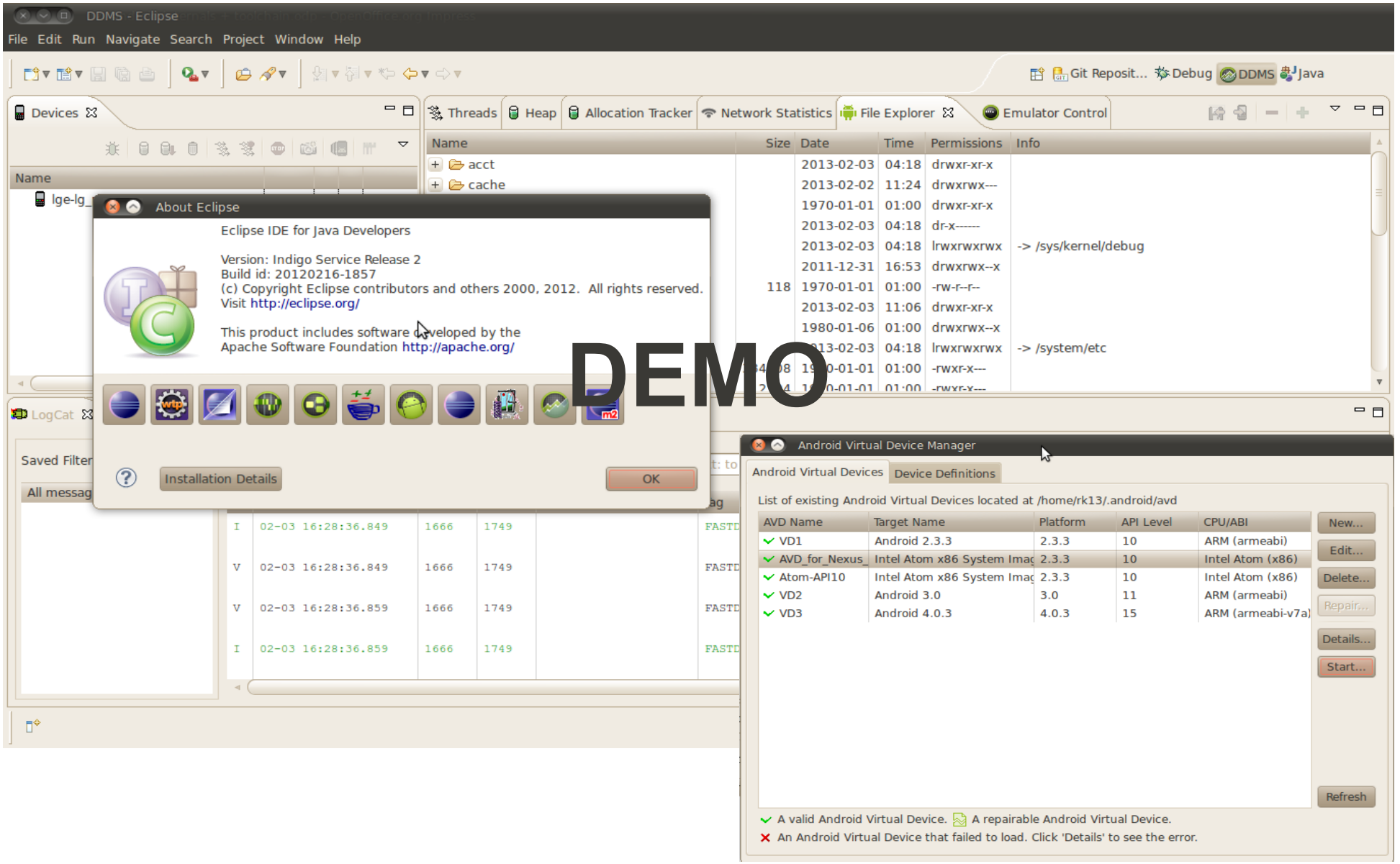
- Use any Android hardware device
- Deploy your app to connected devices directy from the IDE
- Live, on-device debugging, testing, and profiling



- Enable debugging over USB
- Developer options in the system Settings

Dalvik Debug Monitor Server (DDMS)

- Works with both the emulator and a connected device
- Capabilities
  - Heap usage for a process
  - Track memory allocation of objects
  - Work with an emulator or device's file system
  - Examining thread information
  - Network Traffic tool (Android 4.0)
  - LogCat


DDMS - Eclipse
File Edit Run Navigate Search Project Window Help
Devices
Threads
Heap
Allocation Tracker
Network Statistics
File Explorer
Emulator Control
Name
Size
Date
Time
Permissions
Info
acct
cache
2013-02-03 04:18 drwxr-xr-x
2013-02-02 11:24 drwxrwx---
1970-01-01 01:00 drwxr-xr-x
2013-02-03 04:18 dr-x------
2013-02-03 04:18 lrwxrwxrwx -> /sys/kernel/debug
2011-12-31 16:53 drwxrwx--x
1970-01-01 01:00 -rw-r--r--
2013-02-03 11:06 drwxr-xr-x
1980-01-06 01:00 drwxrwx--x
2013-02-03 04:18 lrwxrwxrwx -> /system/etc
1970-01-01 01:00 -rwxr-x---
About Eclipse
Eclipse IDE for Java Developers
Version: Indigo Service Release 2
Build id: 20120216-1857
(c) Copyright Eclipse contributors and others 2000, 2012. All rights reserved.
Visit http://eclipse.org/
This product includes software developed by the
Apache Software Foundation http://apache.org/
Installation Details
OK
DEMO
LogCat
Saved Filter
All messag
02-03 16:28:36.849 1666 1749
02-03 16:28:36.849 1666 1749
02-03 16:28:36.859 1666 1749
02-03 16:28:36.859 1666 1749
Android Virtual Device Manager
Android Virtual Devices
Device Definitions
List of existing Android Virtual Devices located at /home/rk13/.android/avd
AVD Name
Target Name
Platform
API Level
CPU/ABI
VD1 Android 2.3.3 2.3.3 10 ARM (armeabi)
AVD_for_Nexus_ Intel Atom x86 System Imag 2.3.3 10 Intel Atom (x86)
Atom-API10 Intel Atom x86 System Imag 2.3.3 10 Intel Atom (x86)
VD2 Android 3.0 3.0 11 ARM (armeabi)
VD3 Android 4.0.3 4.0.3 15 ARM (armeabi-v7a)
New...
Edit...
Delete...
Repair...
Details...
Start...
Refresh
A valid Android Virtual Device.
A repairable Android Virtual Device.
An Android Virtual Device that failed to load. Click 'Details' to see the error.

Usefull SDK utilities

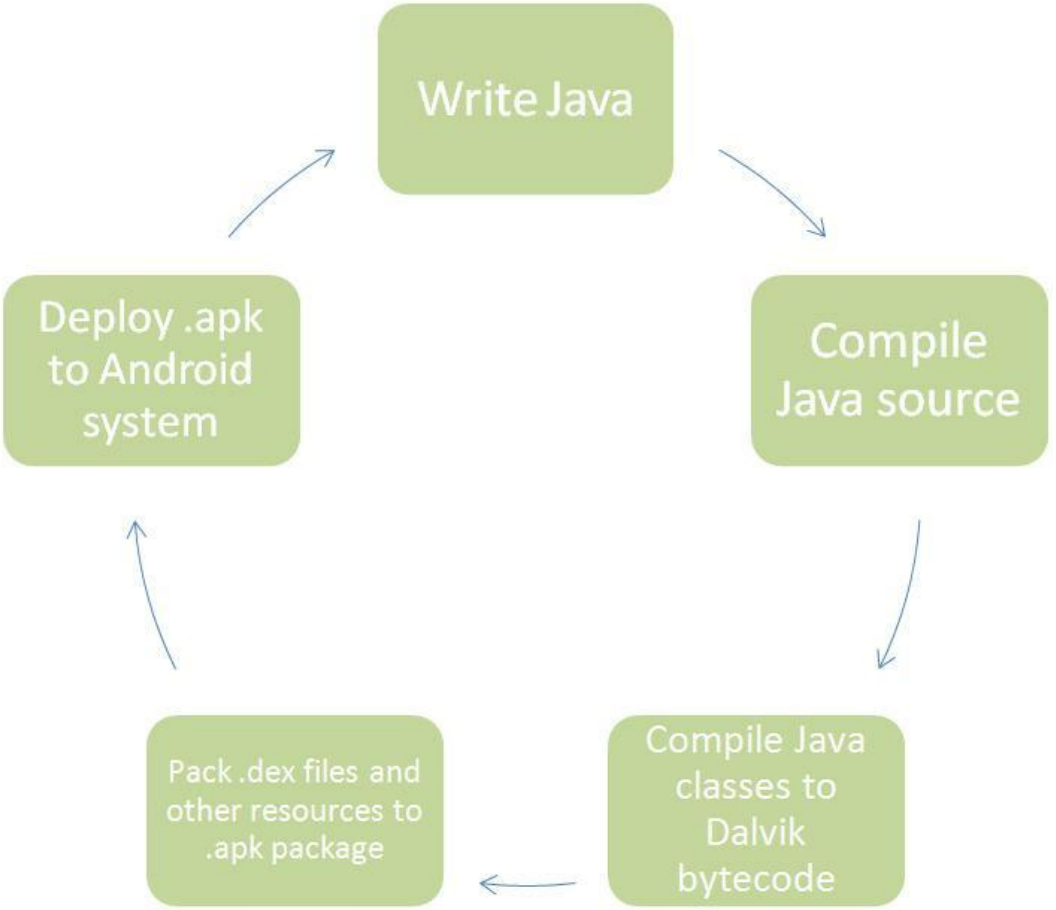
Android Debug Bridge (adb)	Command line tool to communicate with an emulator instance or device (client, server, daemon) Important commands: - devices – show device list - install – install application file - pull / push – copy files to/from device - logcat - print log data - shell – start remote shell on device
Android Device Monitor (monitor)	Tool for Android application debugging and analysis without ADT
Hierarchy Viewer (hierarchyviewer)	Debug and optimize your user interface. Visual representation of the layout's View hierarchy
ProGuard	Obfuscates your code

Native development kit (NDK)

If the speed and efficiency of C/C++ is required

- Tools to build and compile your native code for the device architecture
- A way to package your library into the APK file to distribute your application easily
- A set of native system headers supported for Android platform (libc, libz, liblog, JNI headers)
- Documentation, samples

Application creation cycle



Hello, world ... finally