

Introduction to AOSP

Building your own Android

Wilhelm Fitzpatrick

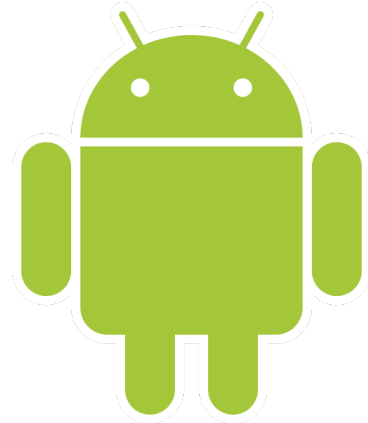
- 19 years of Java
- 4 years of Android
- Work for Cyanogen, Inc.

What's it all about?

- What is Android
- What is AOSP?
- How do you get it?
- How do you build it?
- How do you get started exploring it
- Pragmatic focus
 - a working environment, not architecture
 - emulator, not devices

What is Android

- Most widely used mobile OS
- Most popular Linux distribution ever?
- The Java OS we always dreamed about?
- A trademark of Google, Inc.



Not a trademark



What is AOSP

- Android Open Source Project
- <https://source.android.com/>
- The code for the “OS” part of Android
- *Not* the complete end-user experience

Is AOSP Really Open?

- Open: a complete mobile operating system
- Open: Apache & GPL licensed
- Open: anyone can submit fixes & changes

- Closed: ...but Google decided what goes in
- Closed: new versions developed in private
- Closed: Google apps not included

Only runs on Nexus devices out of the box

Is it Linux?

- Uses Linux kernel and device drivers, but...
- Not LSB
- Kernel extensions
 - Binder
 - Ashmem
 - Wakelocks
- bionic instead of glibc
- missing most of the standard CLI
- unique userspace based heavily on Java

Android's Not Java

- Not derived from the Sun/Oracle stack
- Class library derived from Apache Harmony
- Different bytecode: smali
- VM/Runtime: Dalvik (JIT) or ART (AOT)
- GUI not derived from Swing/AWT
- Language support has fallen behind
 - JDK 6 + selected JDK 7 features.

Dalvik vs ART

- Dalvik (Android 4.4 (KitKat) and older)
 - VM optimized for mobile devices
 - Register-based, not stack-based (JVM)
 - Originally just an interpreter, but later JITed
 - Has its own bytecode and class file format
- ART (Android 5.0 (Lollipop) and newer)
 - Install time compiler, from Dalvik bytecode
 - Trades space for performance & efficiency
 - Overcomes some limits built into Dalvik
 - But... still has an interpreter for when it make sense

Why Do I Care?

If I just want to write Android applications, why care about AOSP?

- Being able to study how the APIs are implemented is the best documentation
- The same language you use to build apps is used to build much of the OS
- A unique strength of Android compared to other mobile (or any?) OS.

What Do I Need?

- A hefty machine
 - at least 16GB of RAM
 - 100GB of disk just to check out
 - at least 150GB of disk to build
- 64-bit Linux
 - Ubuntu 14.04 is official, but others work too
- *or...* recent Mac OS X
 - A little fussier, but still well supported
- Sorry, no Windows
 - ...but you can always use a VM
- Java 6 (Android 4.4-) or Java 7 (Android 5.0)

What are the tools?

- git: for version control
- repo: for managing collections of repositories
- make: for building things
- python: helping to build things
- gcc/Xcode: compiling native stuff
- JDK: compiling Java stuff
- All very well documented:
 - <https://source.android.com/source/building.html>

Ubuntu 14.04 from scratch

```
$ sudo apt-get update
$ sudo apt-get install openjdk-7-jdk
$ sudo apt-get install bison g++-multilib
  git gperf libxml2-utils make
  zlib1g-dev:i386 zip
```

Optional: create...

```
/etc/udev/rules.d/51-android.rules
```

...if you want to talk to physical Android devices

See: <https://source.android.com/source/initializing.html>

Getting AOSP

<https://source.android.com/source/downloading.html>

curl gets repo:

```
$ curl https://storage.googleapis.com/git-repo-downloads/repo > ~/bin/repo  
$ chmod a+x ~/bin/repo
```

pick a version:

<https://source.android.com/source/build-numbers.html>

repo gets the source:

```
$ mkdir WORKING_DIRECTORY  
$ cd WORKING_DIRECTORY  
$ repo init  
  -u https://android.googlesource.com/platform/manifest  
  -b android-5.1.1_r3  
$ repo sync
```



Wait at least an hour...

Building AOSP

<https://source.android.com/source/building-running.html>

set up the environment:

```
$ cd WORKING_DIRECTORY
$ source build/envsetup.sh
```

pick the target:

```
$ lunch aosp_x86-eng
```

run the build:

```
$ make -j16 ← 1-2 x number of hardware threads
```

...wait several hours...

run the emulator:

```
$ emulator -skin WVGA854 ← See development/tools/emulator/skins/ for options
```

Additional Hints

- **ccache**

- Can speed rebuilds by caching C/C++ compiler results:
- `$ export USE_CCACHE=1`
- `$ export CCACHE_DIR=/<path_of_your_choice>/ccache`
- `$ prebuilts/misc/linux-x86/ccache/ccache -M 50G`
- ...but also can be source of stale artifacts that break builds:
- `$ prebuilts/misc/linux-x86/ccache/ccache -C`

When the rubber chicken fails,
try clearing ccache...

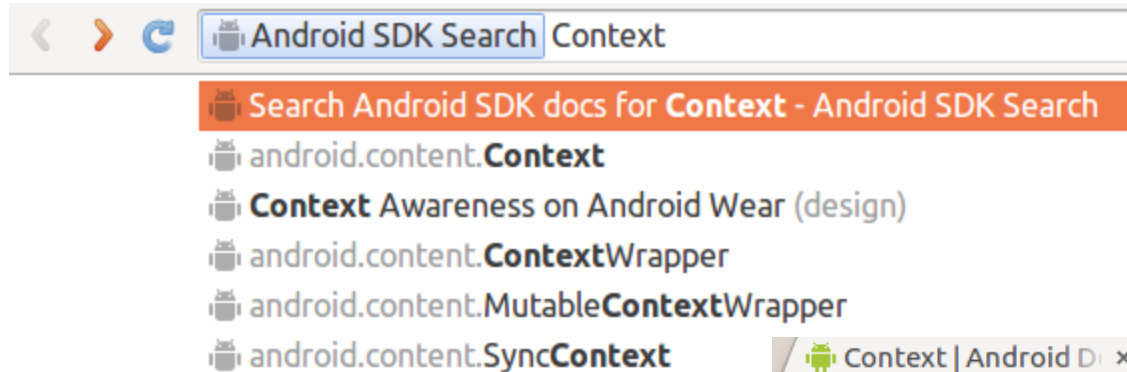
- **Ubuntu 15.04**

- `$ unset JAVA_TOOL_OPTIONS`
 - prevents `jayatanaag.jar` from being appended to classpaths
- Bogus “You have tried to change the API from what has been previously approved.”
 - <https://plus.google.com/+hashcode0f/posts/URHo3hBmfHY>
 - <https://android-review.googlesource.com/#/c/114555/>

Tools for Exploring

- **jgrep and cgrep**
 - tools included with AOSP: search all .java file or all .c/.cpp file for a string
- **Silver Searcher**
 - fast string/file searching through any directory tree
 - <http://geoff.greer.fm/ag/>
- **Android SDK Search**
 - Chrome extension for Android dev docs searching
 - <https://chrome.google.com/webstore/detail/android-sdk-search/hgcbffeicehlpmgmnhnkjbjoldkfhoi>

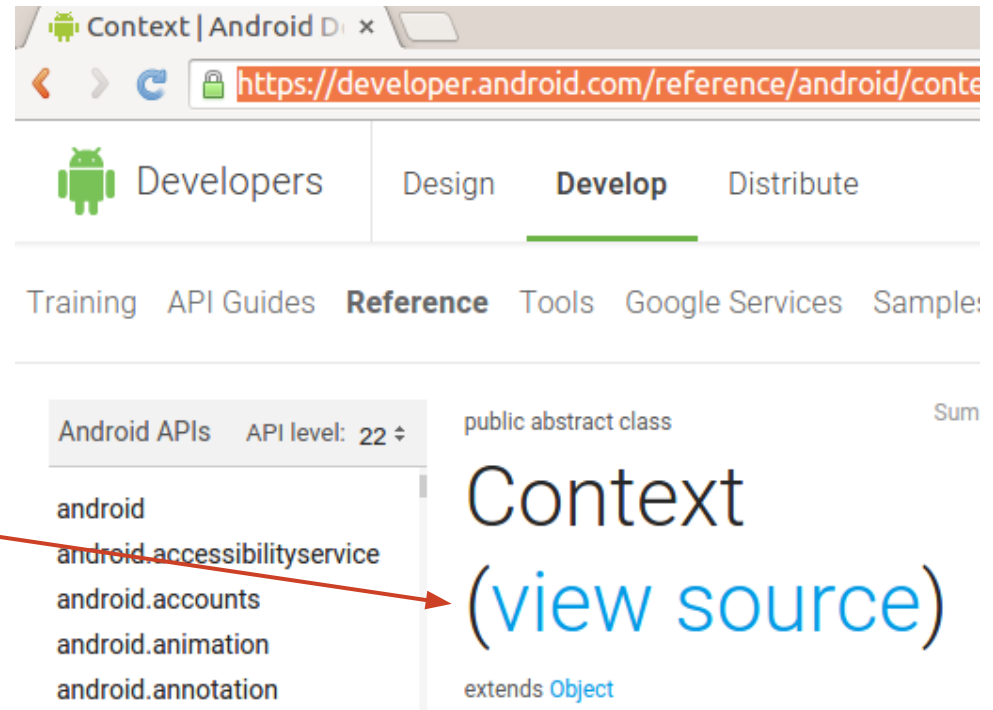
Android SDK Search



Chrome Extension

Fast search Android Developer docs from Chrome address bar

Adds AOSP source links to developer docs displayed in browser



AOSP in Android Studio (1)

```
edit $ANDROID_STUDIO/bin/idea.properties:
```

```
idea.max.intellisense.filesize=4000
```

```
edit /etc/sysctl.conf:
```

```
fs.inotify.max_user_watches = 524288
```

```
$ sudo sysctl -p
```

Ubuntu/Linux



Build and run idegen:

```
$ cd development/tools/idegen/
```

```
$ mm
```

from build/envsetup.sh



```
$ croot
```

```
$ ./development/tools/idegen/idegen.sh
```

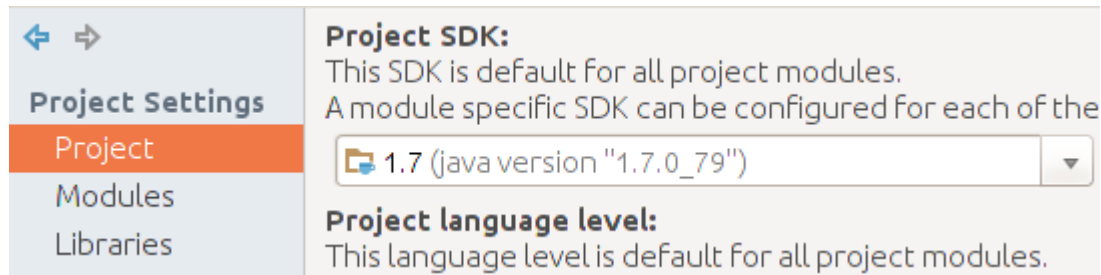
...or...

```
$ java -cp $OUT_DIR_COMMON_BASE/aosp/host/linux-x86/framework/idegen.jar Main
```

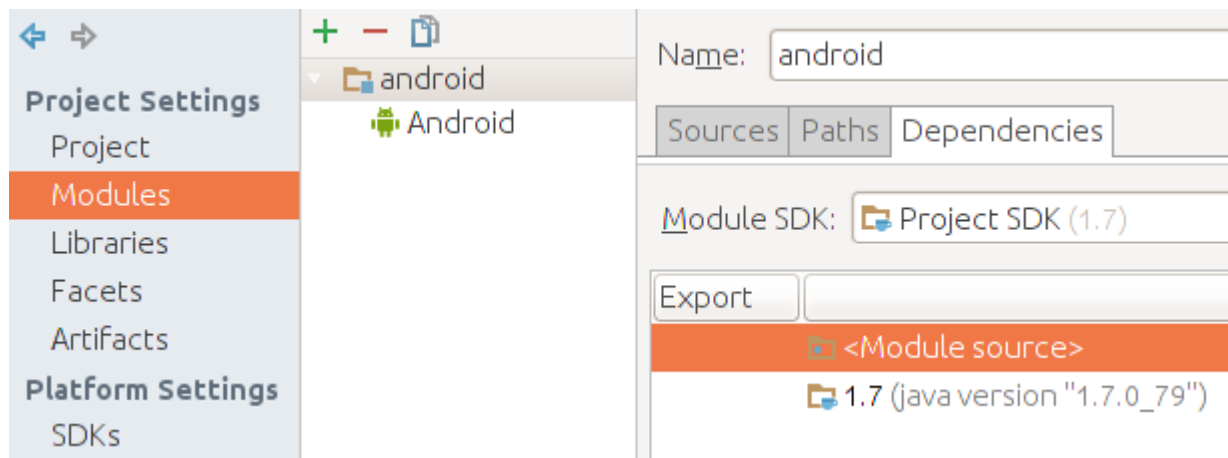
Open the generated `android.ipr` file in your AOSP root with Android Studio

AOSP in Android Studio (2)

- Open File -> Project Structure
- Change Project SDK to “Just Java”



- Change Module SDK, and remove generated .jar dependencies



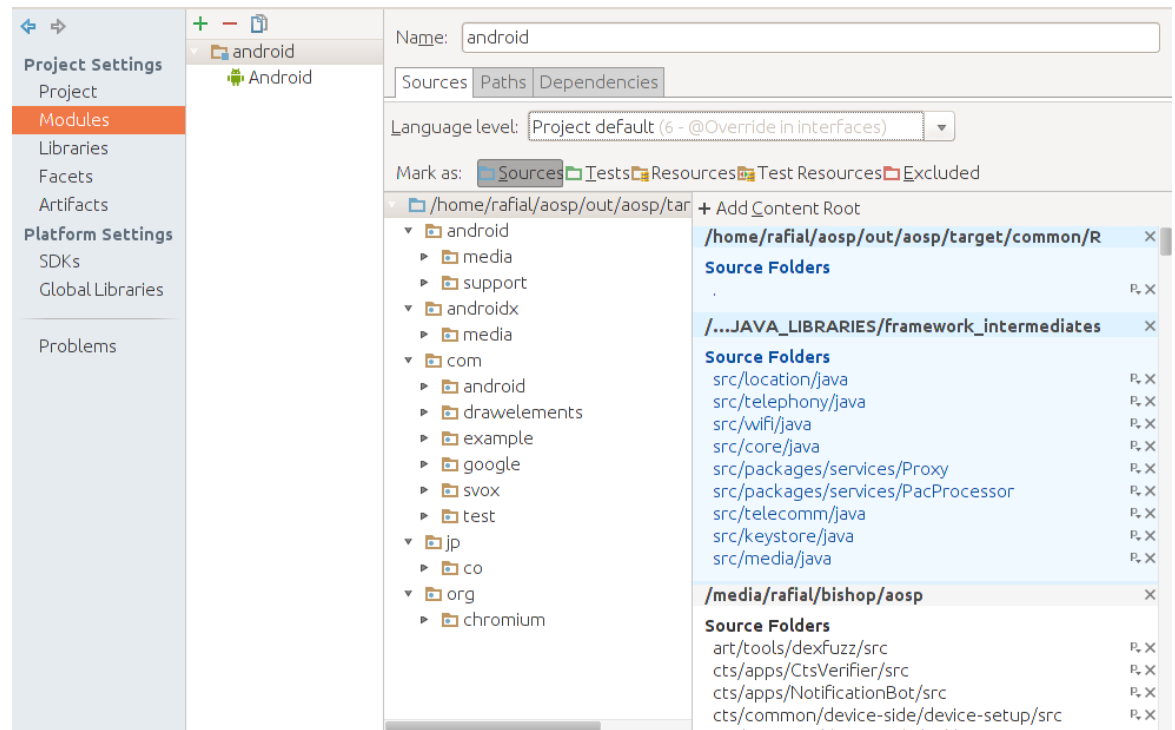
AOSP in Android Studio (3)

Add additional Content Roots in Source Tab:

- `out/aosp/target/common/R` - *then mark as Source*
- `out/aosp/target/common/obj/JAVA_LIBRARIES/framework_intermediates`

R provides generated resources classes to satisfy `R.id.foo` type references.

framework provides classes generated from AIDL files to satisfy service interface calls.

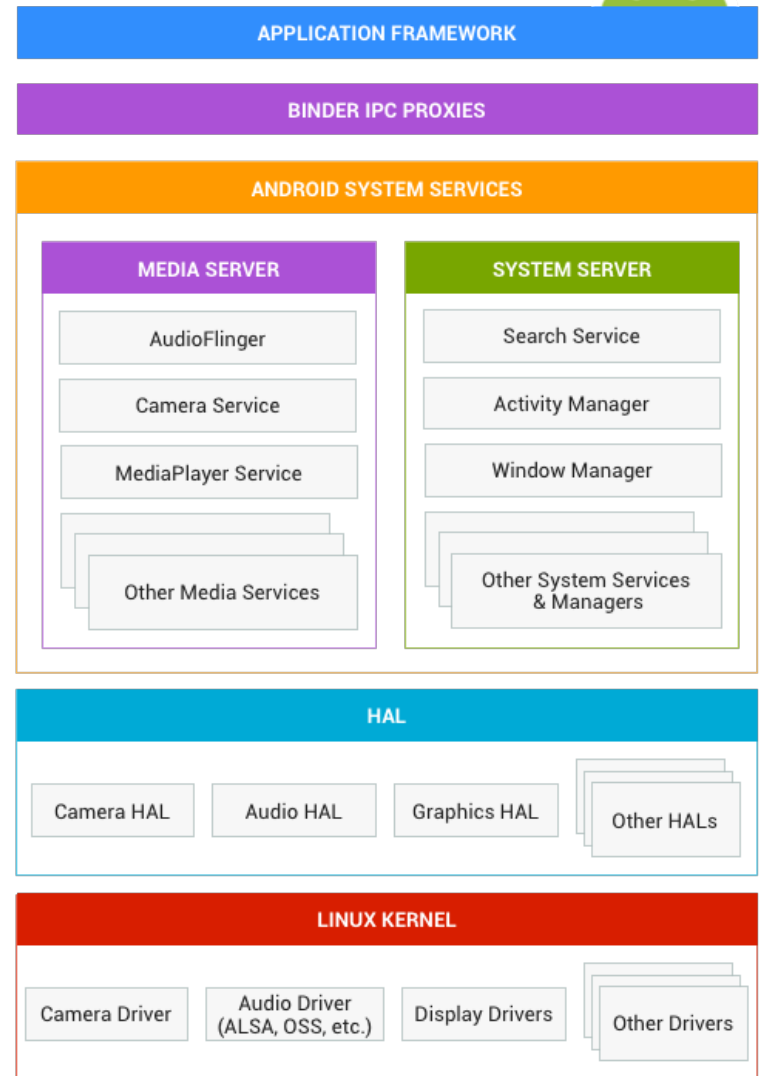


Architecture Overview



The core structure of Android is a collection of cooperating services that communicate through the Binder IPC mechanism to satisfy application API calls, and communicate with the underlying hardware. Some are native code, but many are written all or mostly in Java.

One way to start learning your way around is to start with the APIs your applications use, and follow them down through the AOSP source to see how the pieces are put together.



Glossary

- apk: packaging format for Android applications and some system components. Zip archive with code (classes.dex) and resources
- dex: code packaging format used by Dalvik/ART. Cross between a .class file and a .jar file
- adb: command line tool for interacting with Android devices or emulators. Can be found in AOSP or in the Android SDK.

What is CyanogenMod?



- <http://cyanogenmod.org/>
- Derived from AOSP
- Adds support for many popular devices
- Supports devices EOL'd by manufacturers
- Extends AOSP with interesting features
 - Privacy Guard, Profiles, Quick Tiles, etc.
- Largest community working on fully open source Android
- Similar projects: AOKP, OmniRom, etc.

What is Cyanogen, Inc.?

CYANOGEN

- <http://cyngn.com>
- Extend Android around the principles of Customization, Performance, Security
- Cyanogen OS
 - based on CyanogenMod
 - ships on devices as the original OS
 - passes Google certification, includes Google apps
- A day job for core CyanogenMod devs
- We're hiring!
 - <http://hire.jobvite.com/CompanyJobs/Careers.aspx?c=qRJ9Vfwo>

Learn More

- AOSP Docs:
 - <https://source.android.com/>
- Android Developer Docs:
 - <http://developer.android.com/>
- Embedded Android
 - By Karim J. Yaghmour
 - <http://shop.oreilly.com/product/0636920021094.do>

