

# Open Source Android Development Tools

Manfred Moser

[simpligility.com](http://simpligility.com)

July, 2011



# Table of Contents

## Open Source Android Development Tools - SDK, ADT and beyond

- 1 Android Itself
- 2 Development Tools
- 3 Development Libraries
- 4 Conclusions

# About Manfred

- Android application developer
- Core committer Maven Android Plugin
- Project lead ksoap2-android
- Committer RoboGuice
- Committer Hudson
- Book author, presenter and consultant
- Leader of Vancouver Island JUG, BC, Canada
- Twitter @simpligility , G+ Manfred Moser

# What components make up Android codebase?

## Android Proper

as found on your device

## Android Open Source Project AOSP

subset of above

## Android Software Development Kit SDK

for Java based development applications

## Android Native Development Kit NDK

for C/C++ based development

## Android Open Accessory Development Kit ADK

for USB based hardware hacking

## Android Development Toolkit ADT

Eclipse plugin for Android development

# Android Proper - As Found on Your Device

- Linux
- Apache Harmony
- Lots of other open source components
- Custom Android related components like Dalvik VM
- binary device driver and other blobs
- patched components, custom drivers and different applications from manufacturer and provider

# Android Open Source Project AOSP

- Linux, Apache Harmony, OpenGL ES and lots more
- numerous specific components e.g. Dalvik
- various forks from upstream project
- base for custom roms and such
- various different open source licenses
- source released in drops, sometimes late or not yet

# Android Tools

- development tools like ADT, DDMS and related tools that form SDK
- cooperating with Eclipse projects, external contributors ...
- fully open source, all commits right to public git repo
- available at <http://tools.android.com/>

# Eclipse and ADT and friends

- default supported development environment
- full tool suite including debugging, profiling and so on
- graphical layout editor
- very powerful also with help of further Eclipse plugins (e.g. Mylyn, egit...)
- well architected so that most components work outside Eclipse too



# Other IDE's

- Motorola MOTODEV Studio** <http://developer.motorola.com/docstools/motodevstudio/>  
partly open source, committing upstream to ADT and Eclipse
- Sequoiah** <http://eclipse.org/sequoyah/>
- Jetbrains IntelliJ IDEA CE** <http://www.jetbrains.org/>  
fully open source, includes Android support
- Oracle Netbeans** <http://kenai.com/projects/nbandroid/>  
fully open source, community maintained plugin for Android
- Emacs** <http://gitorious.org/emacs-android-minor-mode>  
fully open source, limited

# Maven Android Plugin and Friends

**Maven Android Plugin** <http://code.google.com/p/maven-android-plugin/>

build apk, deploy to devices, run tests and lots more

**Maven Android SDK Deployer** <https://github.com/mosabua/maven-android-sdk-deployer>

deploy artifacts from SDK to Maven repository

**Android4Maven** <http://sourceforge.net/projects/android4maven/>

bundle android.jar from AOSP to submit to Maven central

**M2E Android** <https://github.com/rgladwell/m2e-android>

Maven build to play nice with ADT

**AndroidSDKFido** <https://github.com/joakime/android-sdkfido>

build source and javadoc artifacts

**Android RIndirect** <https://github.com/akquinet/android-rindirect>

help with component reuse



# Others

Gradle Android Plugin <https://code.google.com/p/gradle-android-plugin/>  
for the Groovy based build system Gradle

SBT Android Plugin <https://github.com/jberkel/android-plugin>  
for the Scala based build system SBT, Scala based Android  
applications

Rake/Ruboto/Maven for JRuby Android applications

# Maven Android Plugin - Example

- deploy to multiple devices and run tests
- reuse of other Maven plugins
- use of libraries and Android components easy
- full release cycle sign, zipalign, automatic versioning, ...
- Proguard support
- Native components and libraries
- Scala support
- more

# Other Development Tools

**Droid at Screen** <http://blog.ribomation.com/2010/01/droidscreens/>  
device screen recorder/projector

**DroidDraw** <http://www.droiddraw.org/>  
UI build and design tool

**dex2jar** <http://code.google.com/p/dex2jar/>  
converter from dex to jar format

**smali/baksmali** <http://code.google.com/p/smali/>  
dex assembler/disassembler

**Android2PO** <https://github.com/miracle2k/android2po>  
Converter for Android strings to gettext

# Java Libraries suitable for Android

Jackson <http://jackson.codehaus.org/>

JSON library

google-gson <http://code.google.com/p/google-gson/>

JSON library

SimpleXML <http://simple.sourceforge.net/home.php>

XML serialization framework

ksoap2-android <http://code.google.com/p/ksoap2-android/>

SOAP library

WSDL2Android <https://github.com/kigero/WSDL2Android>

code generator for ksoap2-android

ormlite <http://ormlite.com/>

light-weight object relational mapping tool

Twitter4J <http://twitter4j.org/>

twitter integration library



# Frameworks for General Purpose Usage

Roboguice <http://roboguice.org>

Google Guice IoC based framework

AndroidAnnotations <http://code.google.com/p/androidannotations/>

annotation based code generation framework

DroidFu <http://github.com/kaeppler/droid-fu>

general purpose collection of helper classes

CommonWare Android Components CWAC <https://github.com/commonsguy>

collection of helper classes and widget

DroidKit <https://github.com/droidkit/droidkit>

collection of Android API extensions

Libs for Android <http://code.google.com/p/libs-for-android/>

collection of libraries

AndroidLibs <http://www.androidlibs.com/>

social and contact related libraries

AndroidAsync <https://bitbucket.org/hal/android-async/>

alternate implementation for asynchronous tasks



# Libraries for Specific Use Cases

**ZXing** <http://code.google.com/p/zxing/>

barcode scanning library and application

**Jon's Java Imaging Library** <http://code.google.com/p/jjil/>

image processing library

**OpenCV-Android** <http://billmccord.github.com/OpenCV-Android/>

real time computer vision library

**Facebook Android SDK** <https://github.com/facebook/facebook-android-sdk>

your guess ;-)

**MapsForge** <http://code.google.com/p/mapsforge/>

OpenStreetMap toolbox

**OSMDroid** <http://code.google.com/p/osmdroid/>

OpenStreetMap toolbox



# Example RoboGuice

```
@Inject myObj;  
@InjectResource(R.string.name)  
@InjectView(R.id.editName)  
@Singleton stateHolder;  
and so on instead of  
MyObject myObj = new MyObject();  
getResources().getString(R.string.name)  
findViewById(R.id.editName)  
and lots more
```



# UI Libraries and Widgets

**GreenDroid** <https://github.com/cyrilmottier/GreenDroid>  
application framework and UI widget collection

**svg-android** <http://code.google.com/p/svg-android/>  
SVG rendering library

**View Flow for Android** <https://github.com/pakerfeldt/android-viewflow>  
horizontally scrolling views

**Android Wheel** <http://code.google.com/p/android-wheel/>  
wheel input control widget

**ActionBarSherlock** <http://actionbarsherlock.com/>  
ActionBar support for tablets and phones

**Android ActionBar** <https://github.com/johannilsson/android-actionbar>  
ActionBar support for tablets and phones

# More UI Libraries and Widgets

Pull to Refresh for Android <https://github.com/johannilsson/android-pulltorefresh>  
list refresh widget

Android ColorPickerPreference <https://github.com/attenzione/android-ColorPickerPreference>  
color picker

Android AutoFitTextView <https://github.com/grantland/android-autofittextview>  
dynamic font resizing in text view

Android TextView Multiline Ellipse <http://code.google.com/p/android-textview-multiline-ellipse/>  
ellipse for multiline text view

Android MapViewBalloons <https://github.com/jgilfelt/android-mapviewballoons>  
UI widget for maps

# Game Development Libraries

**libgdx** <http://libgdx.badlogicgames.com/>

cross-platform 2D and 3D game development framework for Java/C/C++.

**AndEngine** <http://www.andengine.org/>

Java based 2D OpenGL Game Engine for Android

**forget3D** <http://code.google.com/p/forget3d/>

OpenGL ES framework for Android, Win32, WinCE

**min3d** <http://code.google.com/p/min3d/>

lightweight 3d library/framework for Android using Java with OpenGL ES

**Angle** <http://code.google.com/p/angle/>

game library for 2D games using OpenGL ES

# Android Testing Tools

Robotium <http://robotium.org>

*Selenium* for Android

Robolectric <http://robolectric.org>

Android tests run on JVM

Calculon <https://github.com/kaepler/calculon>

Android testing DSL

Android JUnit Report <https://github.com/jsankey/android-junit-report>

tool to load test report from device/emulator

Memory Sucker <https://github.com/nollbit/memory-sucker>

test tool to simulate low memory scenarios

Android Mock <http://code.google.com/p/android-mock/>

object mocking library



# Others of interest

**OpenIntents** <http://code.google.com/p/openintents/>

collection of reusable components and applications

**i-jetty** <http://code.google.com/p/i-jetty/>

servlet container running on the device

**Android Screenshot library** <http://code.google.com/p/android-screenshot-library/>

programmatically take screenshots, n no root required

**Android Alarm Database** <http://code.google.com/p/android-alarm-database/>

alarm application and toolkit

**Application Crash Report for Android ACRA** <http://code.google.com/p/acra/>

crash report library

**Android Error Reporter** <https://github.com/tomquist/Android-Error-Reporter>

error report library



# Other Languages

Java is the main language for development and API but also possible are

- C/C++ (via NDK first class)
- JRuby
- Scala
- Coljure
- JavaScript (e.g. via PhoneGap)
- Processing <http://wiki.processing.org/w/Android>
- C#

# Example - Scala libraries

- Baitha <https://github.com/sattvik/baitha>
- Positronic Net [https://github.com/rst/positronic\\_net](https://github.com/rst/positronic_net)
- Borachio mocking library <http://borachio.com/>



# Is Android Java?

- Yes - default application programming language
- Yes - API is Java based
- No - not using a standard compliant Java Virtual Machine Runtime
- No - only using parts of the standard class libraries and

# Is Android Open Source?

- Yes, in time - AOSP open sourced in drops
- Yes -ADT fully open source
- Yes and no - cooperation with upstream projects patchy but exists
- No - binary blobs for drivers and other parts

# Android part of the Java Community?

- Yes - parts of Android itself
- Yes - tooling around Android
- Yes - lots of libraries and tooling from rest of Java universe
- Yes - lots of people from Java community, also part of Android community
- Yes - lots of JVM related aspects as well e.g. Scala, JRuby, Processing, Groovy...

# Android part of Open Source Community?

- Yes - part of Apache Community
- Yes - part of Eclipse Community
- Yes - part of Ruby, Scala, Groovy/Gradle...
- Yes - lots of open source libraries specifically to Android
- Yes - lots of projects on Github, Google Code, ...
- Yes - move towards Maker community with ADK

# Overall conclusion

- Despite lots of flaws and kinks that make things interesting
- Android is part of the Java and Open Source communities
- Android touches a lot of other communities and brings them together
- Android is a great chance to collaborate and learn

# What can you do?

- Buy an unlocked/unlockable device
- Use a custom ROM
- Ask for open source drops of AOSP
- Encourage patches to upstream projects and
- Ask for open sourcing of any closed parts, tools...
- Contribute and cooperate yourself!