

Building a SDK¹

learnings from building the
sensorberg Android SDK

```
git clone https://github.com/falkorichter/  
presentations;cd building-a-SDK
```

pull requests are welcome!



about sensorberg

- integrate iBeacon interactions in your apps
- complete solution
 - iOS, Android, Windows SDK
 - Showcase apps, Utilities
 - Campaign management Backend + Frontend
- lot's of open source software

WE`RE HIRING -> sensorberg.com/jobs

shameless plug:
tune your
podcatchers to
androidpodcast.de²

² better SSL support coming soon

The screenshot shows a web browser window with three tabs at the top: "Sensorberg Showcase – Ar" (active), "Deckset for Mac: Turn your", and "pandroid". The main content area displays the "pandroid" website. The header features the "pandroid" logo and the tagline "Der ganzheitliche Android-Podcast". A navigation bar includes "HOME" (selected) and "ÜBER". Below, an episode card for "PD049 - I/O 2015" is shown, dated "Juni 3, 2015". The card includes a play button icon, the episode title, a description ("Google I/O 2015 und einige kleine Themen"), and download links for "MP3 Audio [83 MB]" and "Download". A link "Wieder einmal mit Dirk & Falko:" is also present. To the right, there's a sidebar with "RSS feeds" (including "Abonnieren" and "MP3 RSS") and "Kategorien" (Deutsch, News, Sprache). At the bottom, a "tags" section lists "entwicklung", "mikro.fm", "news", "politik", and "rant".

pandroid
Der ganzheitliche Android-Podcast

HOME ÜBER

PD049 - I/O 2015

Juni 3, 2015

PD049 – I/O 2015
Google I/O 2015 und einige kleine Themen

00:00:00 00:00:00

MP3 Audio [83 MB] Download Show URL

Wieder einmal mit Dirk & Falko:

Link:

- <https://github.com/mortenjust/androidtool-mac>
- <https://corner.squareup.com/2015/05/leak-canary.html>
- Bluetooth Advertising Bacon
 - <https://jira.cyanogenmod.org/browse/BACON-3435>
 - auf nach reddit damit
- Multidex
 - <https://github.com/sensorberg-dev/android-sdk/commit/5d9b13b0d297f52c6295609281c93425b76f9d55>
 - <https://github.com/sensorberg-dev/android-sdk/blob/master/android-sdk/build.gradle>
 - <https://developer.android.com/tools/building/multidex.html>
 - <https://android.googlesource.com/platform/frameworks/multidex/+/master/library/src/android/support/multidex/MultiDex.java>
- JRebel

about me

github.com/falkorichter

@volkersfreunde

mobile software developer @ Sensorberg & proud dad

since 1515

about.me/falkorichter

pilot of small things™ -> contact me for Berlin FPV racing

A close-up photograph of a baby lying on its back on a light-colored wooden surface. The baby has dark hair and blue eyes. A white milestone card with a yellow circle containing the number '1' and the word 'month' is tucked under the baby's head. The card also features a small green and orange chick illustration. The baby is wearing a light-colored onesie with a repeating pattern of stylized animals.

#awwww

Building a SDK

open source at developer.sensorberg.com/android/

many learnings and good practices

all samples from actual problems

all code (used to be) live

general thoughts

listen to your customers

new features tailored to individual customers

don't listen to customers, you are the expert

release early, release often (at least internally)

think twice

about everything

sure let's move this class, wait, a customer is using it, all
his code will not compile...

his imports are broken

offer support

github issues

zendesk

gitter.im chat³

telefono

workshops

direct interaction might be the best

³ <https://gitter.im/sensorberg-dev/android-sdk>

logging

ship a powerfull logging engine!

domain logging output

disable all log output by default

logging can be turned on

ideally log4j

→ log to file, send to developers...

→ log to server



VolleyLog.java

```
/**  
 * Logging helper class.  
 * <p/>  
 * to see Volley logs call:<br/>  
 * {@code <android-sdk>/platform-tools/adb shell setprop log.tag.Volley VERBOSE}  
 */  
public class VolleyLog {  
    public static String TAG = "SensorbergVolley";  
  
    public static boolean DEBUG = Log.isLoggable(TAG, Log.VERBOSE);
```

Check `VolleyLog.MarkerLog` for useful domain specific
logging

Logger.log.LogError(e)

```
public class Logger {
```

```
    public static Log log;
```

- our custom Logger interface
- using secret codes to enable logging at runtime

testing

no fancy setup

recently switched to Espresso 2.0 with JUnit 4

add your own testrunner from the very start

test on devices

testing - multidex

:android-sdk:dexDebugTest FAILED

FAILURE: Build failed with an exception.

* What went wrong:

Execution failed for task ':android-sdk:dexDebugTest'.

> com.android.ide.common.internal.LoggedErrorException: Failed to run command:

 /usr/local/opt/android-sdk/build-tools/22.0.1/dx --dex --output /Users/falkorichter/Documents/workspaces/sensorberg/github-android-sdk/android-sdk/build/intermediates/dex/test/debug --input-list=/Users/falkorichter/Documents/workspaces/sensorberg/github-android-sdk/android-sdk/build/intermediates/tmp/dex/test/debug/inputList.txt

Error Code: 2

Output: androidTestCompile 'com.squareup:fest-android:1.0.8'

 2

Output: androidTestCompile 'org.mockito:mockito-core:1.9.5'

Output: androidTestCompile 'com.google.dexmaker:dexmaker:1.0'

Output: androidTestCompile 'com.google.dexmaker:dexmaker-mockito:1.0'

Output: androidTestCompile ('com.android.support.test.espresso:espresso-core:2.0') {

Output: at com.android.dx.merge.InstructionTransformer\$StringVisitor.visit(InstructionTransformer.java:72)

Output: at com.android.dx.merge.CodeReader.visitAll(CodeReader.java:104)

Output: at com.android.dx.CodeReader.visitAll(CodeReader.java:89)

Output: } at com.android.dx.merge.InstructionTransformer.transform(InstructionTransformer.java:49)

Output: at com.android.dx.merge.DexMerger.transformCode(DexMerger.java:842)

Output: at com.android.dx.merge.DexMerger.transformClassData(DexMerger.java:786)

Output: at com.android.dx.merge.DexMerger.mergeDexes(DexMerger.java:171)

Output: at com.android.dx.merge.DexMerger.merge(DexMerger.java:1218)

Output: at com.android.dx.command.dexer.Main.mergeLibraryDexBuffers(Main.java:454)

Output: } at com.android.dx.command.dexer.Main.runMonoDex(Main.java:303)

Output: at com.android.dx.command.dexer.Main.run(Main.java:246)

Output: at com.android.dx.command.Main.main(Main.java:106)

* Try:

Run with --stacktrace option to get the stack trace. Run with --info or --debug option to get more log output.

BUILD FAILED

Total time: 20.609 secs

→ **github-android-sdk** git:(master) x

we are having the multidex problem 😳

multidex test project

```
androidTestCompile ('com.android.support:multidex:1.0.0')
```

```
public class SensorbergTestRunner extends android.support.test.runner.AndroidJUnitRunner {  
  
    @Override  
    public void onCreate(Bundle arguments) {  
        MultiDex.install(getApplicationContext());  
        super.onCreate(arguments);  
    }  
}
```

- not all tests found and executed on Dalvik, but no crashes
- no problems with ART

 Done: 74 of 74 (34.918 s)

↑
↓
↔
↔
↑
↓
↔
↔
Running tests
Test running startedFinish

```
Testing started at 10:24 ...
Installing com.sensorberg.sdk.test
DEVICE SHELL COMMAND: pm install -r "/data/local/tmp/com.sensorberg.sdk.test"
pkg: /data/local/tmp/com.sensorberg.sdk.test
Success
```

 Done: 250 of 250 (25.014 s)

↑
↓
↔
↔
↑
↓
↔
↔
Running tests
Test running startedFinish

```
Testing started at 10:25 ...
Installing com.sensorberg.sdk.test
DEVICE SHELL COMMAND: pm install -r "/data/local/tmp/com.sensorberg.sdk.test"
pkg: /data/local/tmp/com.sensorberg.sdk.test
Success
```

dogfooding

use services like hockeyapp

"release early, release often"

play alpha & beta

you want problems with your SDK/library happen to a
colleague

build an app Sensorberg Showcase

Sensorberg GmbH - 4. März 2015 -  USK ab 0 Jah

be your own customer

Software & Demos

build your own sample apps

Installieren

build lot's of samples



Zur Wunschliste hinz

answer integration question with samples



"developer mode"

build process

make sure you can release at the speed you wish, if necessary every hour.

Bugs are okay, if you can fix them within hours and deploy them that fast, customers will be impressed.

graduation





Android Tools Project Site

Search this site

Projects Overview

Schedule

Create Status

Roadmap

Download

View Data

Recent Changes

Technical docs

New Build System

Known Issues

Tips

Build Overview

Contributing

Feedback

Technical docs > New Build System >

AAR Format

/AndroidManifest.xml (mandatory)

The manifest is required for distribution of an Android Aar Project.

/classes.jar (mandatory)

The file extension is .aar, and the maven artifact type should be aar as well, but the file itself a simple zip file with the following entries:

/res/ (mandatory)

↳ /AndroidManifest.xml (mandatory)

- /classes.jar (mandatory)

- /res/ (mandatory)

↳ /R.txt (mandatory)

- /assets/* (optional)

- /libs/*.jar (optional)

↳ /jni/<abi>/*.so (optional)

- /proguard.txt (optional)

↳ /lint.jar (optional)

/R.txt (mandatory)

These entries are directly at the root of the zip file.

/assets/ (optional)

The R.txt file is the output of aapt with --output-text-symbols.

/libs/.jar (optional)

[Sign in](#) | [Recent Site Activity](#) | [Report Abuse](#) | [Print Page](#) | Powered By [Google Sites](#)

/proguard.txt (optional)

/lint.jar (optional)

fatjar / fataar

fataar = aar with all dependencies

not solved:

- * [http://stackoverflow.com/questions/28605367/
library-with-bundles-dependencies-fat-aar](http://stackoverflow.com/questions/28605367/library-with-bundles-dependencies-fat-aar)
- * <https://github.com/vRallev/jarjar-gradle>



releasing

jcenter ftw

your own public nexus

github pages

branch: mvn-repo ▾

android-sdk / com / sensorberg / sdk / sensorb

This branch is 47 commits ahead, 73 commits behind master

"Release v0.9.3"



Continuous Integration authored on 17 Mar

..

0.3.4

"Release 0.3.4"

0.3.5

"Release 0.3.5"

0.3.6

"Release 0.3.6"

0.4.0

"Release 0.4.0"

0.4.1

"Release 0.4.1"

0.4.2

"Release 0.4.2"

0.6.0

"Release v0.6.0"

0.7.0

"Release v0.7.0"

0.7.1

"Release v0.7.1"

0.7.2

"Release v0.7.2"

0.8.0

"Release v0.8.0"

0.8.1

"Release v0.8.1"

0.8.2

"Release v0.8.2"

github pages

check `uploadGithub.gradle` and `verifyTasks.gradle`

- clone repo, deploy artifact, push to mvn-repo
- be strict: don't remove releases, don't override releases

```
repositories {  
    maven {  
        url "https://raw.github.com/sensorberg-dev/android-sdk/mvn-repo";  
    }  
}
```



Easy to use package repository for GitHub

JitPack builds GitHub projects on demand and publishes ready-to-use packages
Easy to use package repository for GitHub

GitHub repo url

Look up

How to

To get a GitHub project into your build:

Step 1. Add the JitPack repository to your build file

gradle

maven

```
repositories {  
    maven {  
        url "https://jitpack.io"  
    }  
}
```

architecture

- everything should be testable (if you need to)
- mock sever requests
- mock device callbacks
- blocking threading for tests

Dependency injection

do.it.

Platform class

our cheap dependency injection pattern

abstracts all Android specific code

Bluetooth callbacks

Threading

Settings.java

have many internal settings

build an API for remote changing them

document them to individual customers/users

good for testing as well

feature flags

use the settings or other measures.

ship code early, activate later

test on production code

native dependencies

it works

we're shipping with realm.io as a dependency

gradle ftw

secret codes

###code### in the dialer

- enable features at runtime
- no persistence!
- good for switching environments

Secret Codes

Firefly - June 15, 2014 -  USK: All ages

Tools

Install



Add to Wishlist

★★★★★ (621)

reviews

you are alone

you know nothing.

review your code

explain it to colleagues

invite somebody external

talk about in at a conference, meetup...

dependencies

- choose wisely

- be forward compatible

- be aware of updates, at least test with newer dependencies

- update your SDK regularly to use the latest dependencies

dependencies

Package Name



Sorted By Priority



"Name Mangling"^{Android-sdk}

-> change the package of dependencies if necessary

1. clone

networkstate



1.0.0-RC10
May 26, 2015

2. change package



1.0.0
Mar 12, 2015

3. push

resolver



1.0.0
Mar 12, 2015

4. release⁷

The resolver is a microservice that you can run in your own infrastructure or hosted by us, responsible for delivering all necessary data to th...

⁷ <https://bintray.com/sensorberg/maven>

android-sdk-bootstrapper



1.0.0-RC10
May 26, 2015

okvolley



1.1.1
Apr 24, 2015

volley



1.2.1
Apr 24, 2015

Feedback

volley

repackage volley (git subtree)

volley is super powerfull and you should use it

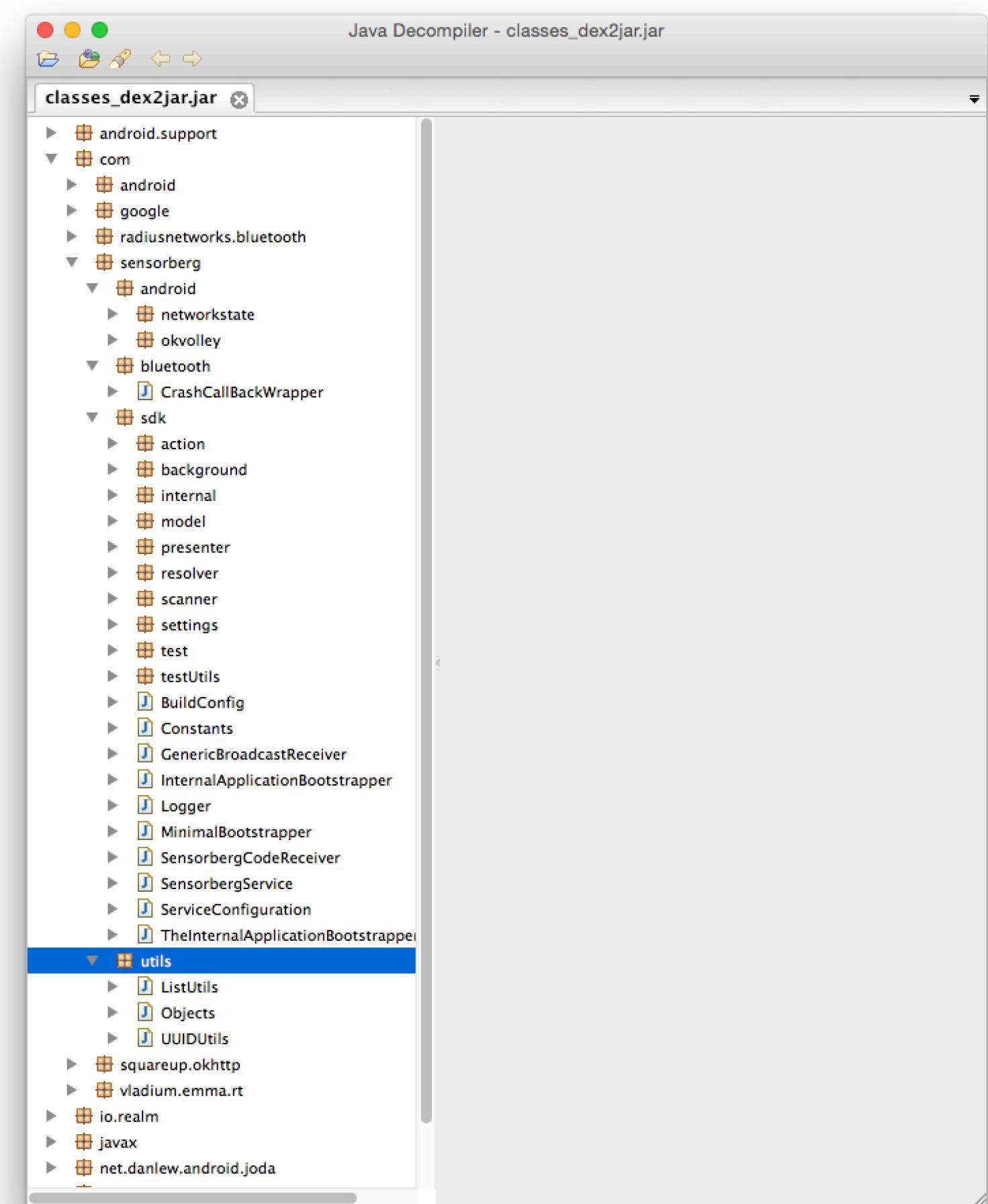
security?

You ship a jar/aar with class files

JD-Gui -> java

all your source are belong to every customer

Decide who you trust, it's really hard to hide from your host app



security?

if you really want to be safe, pull all the strings

SSL pinning:

- customer can override the cert check

obfuscation

Proguard ftw

- we never choose to ship proguarded code
- useful when stripping unused classes
 - ship an ProGuard rules file and let the customer decide

dexguard might be better (no experience)

harder troubleshooting

proguard

check android-sdk/build.gradle
release a proguard file for your
application

```
buildTypes {  
    debug {  
        multiDexEnabled true  
        testCoverageEnabled true;  
        buildConfigField "String", "RESOLVING"  
    }  
    release {  
        minifyEnabled false  
        proguardFile getDefaultProguardFile("proguard-android.txt")  
        proguardFile 'proguard-release.txt'  
    }  
    minified {  
        minifyEnabled true  
        proguardFile getDefaultProguardFile("proguard-android.txt")  
        proguardFile 'proguard-proguarded.txt'  
    }  
}
```

Questions

pull request, issues on github

falko@sensorberg.com about sensorberg

@volkersfreunde for everything else



slides⁵: 

⁵ <https://github.com/falkorichter/presentations>