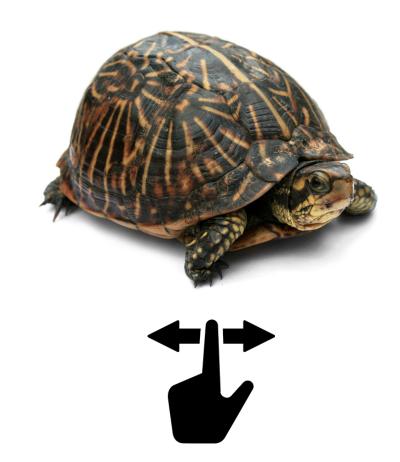
Building Android Apps

Disclaimers

Getting started



app/build.gradle

```
apply plugin: 'com.android.application'
android {
  compileSdkVersion 23
  buildToolsVersion '23.0.2'
  defaultConfig {
  buildTypes {
    release {
dependencies {
```

Relevant Build Files

```
app
build.gradle
build.gradle
gradle
    wrapper
       gradle-wrapper.jar
       gradle-wrapper.properties
gradle.properties
gradlew
gradlew.bat
local.properties
settings.gradle
```

Project structure

```
$ ./gradlew -q projects

Root project 'turtler'
\--- Project ':app'
```

Available Tasks

\$./gradlew tasks

```
connectedAndroidTestDebug
           buildDependents uninstallDebugAndroidTest
   installDebugAndroidTest androidDependencies
                  testDebug uninstallRelease
                   wrapper assembleAndroidTest assembleRelease compileReleaseSources deviceAndroidTest
              installDebug properties lintDebug components
            signingReport init uninstallAll tasks lint
      projects clean check buildNeeded lintRelease help uninstallDebug assemble build testRele
                                                      testRelease
         jarReleaseClasses compileDebugSources
                           connectedCheck deviceCheck
                compileDebugAndroidTestSources test
         dependencies jarDebugClasses dependencyInsight
                        compileReleaseUnitTestSources
assembleDebugAndroidTest connectedAndroidTest
                        mockableAndroidJar compileDebugUnitTestSources (Word)ItOut
```

Gradle wrapper++

\$ gw clean aD

```
connectedAndroidTestDebug
           buildDependents uninstallDebugAndroidTest
   installDebugAndroidTest androidDependencies
                  testDebug uninstallRelease
                   wrapper assembleAndroidTest assembleRelease compileReleaseSources deviceAndroidTest
              installDebug properties lintDebug components
            signingReport init uninstallAll tasks lint
      projects clean check buildNeeded lintRelease help uninstallDebug assemble build testRele
                                                      testRelease
         jarReleaseClasses compileDebugSources
                           connectedCheck deviceCheck
                compileDebugAndroidTestSources test
         dependencies jarDebugClasses dependencyInsight
                        compileReleaseUnitTestSources
assembleDebugAndroidTest connectedAndroidTest
                        mockableAndroidJar compileDebugUnitTestSources (Word ItOut
```

Using a Gradle plugin

```
/build.gradle
buildscript {
   repositories {
       jcenter()
   dependencies
      classpath 'com.android.tools.build:gradle:1.3.1'
      classpath 'com.novoda:gradle-android-command-plugin:1.5.0'
app/build.gradle
apply plugin: 'com.novoda.android-command'
$ gw runDebug
```

Dependencies

```
dependencies {
   compile 'com.google.guava:guava:18.0'
   testCompile 'junit:junit:4.12'
   testCompile 'org.mockito:mockito-core:1.9.5'
}
```

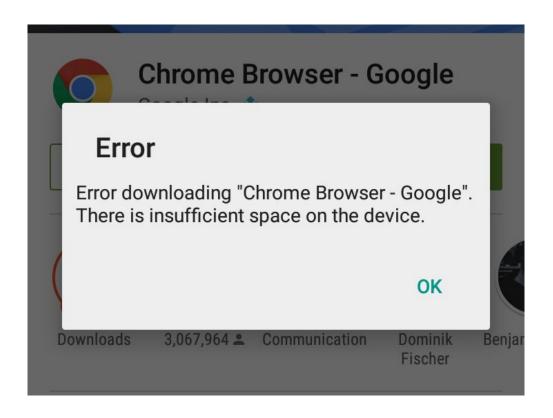
Dependencies

```
apply plugin: 'com.neenbedankt.android-apt'
dependencies {
             'com.google.dagger:dagger-compiler:2.0'
   apt
             'com.google.dagger:dagger:2.0'
   compile
             'com.google.guava:quava:18.0'
   compile
   testCompile 'junit:junit:4.12'
   testCompile 'org.mockito:mockito-core:1.9.5'
```

Dependencies

```
apply plugin: 'com.neenbedankt.android-apt'
dependencies {
      'com.google.dagger:dagger-compiler:2.0'
   apt
   compile 'com.google.dagger:dagger:2.0'
   compile 'com.google.guava:guava:18.0'
   paidCompile project(':lvl')
   testCompile 'junit:junit:4.12'
   testCompile 'org.mockito:mockito-core:1.9.5'
```

Proguard



Proguard

```
buildTypes {
 release {
  minifyEnabled true
   shrinkResources true
   proguardFiles getDefaultProguardFile('proguard-android.txt'),
                 'proquard.cfg'
proguard.cfg
#Dagger2
-dontwarn dagger.shaded.auto.common.**
-dontwarn dagger.internal.codegen.**
-keepattributes Signature
```

Archive Release

```
ext.VERSION_NAME=1.0.0
...

task archiveRelease(type: Copy) {
   from 'build/outputs/apk', 'build/outputs/'
   into "releases/${VERSION_CODE}"
   include('android-release.apk', 'mapping/**')
   rename ('android-release.apk',
        "${APP_NAME}_${BUILD_DATE}_${VERSION_NAME}_${VERSION_CODE}.apk")
}
```

First Success

Scaling up

- re-use
- testing
- quality assurance
- continuous integration
- push-button release

Re-use - Core project

```
core/
gradle init --type java-library
app/build.gradle
dependencies {
    compile project(':core')
settings.gradle
include ':core'
```

Re-use - Annotation processing

core/build.gradle

```
apply plugin: 'net.ltgt.apt'

dependencies {
    apt        'com.google.auto.value:auto-value:1.1'
    compile        'com.squareup.okhttp:okhttp:2.5.0'
    ...
}
```

Re-use - Script client

```
scripts/
gradle init --type groovy-library
scripts/build.gradle
dependencies {
    compile project(':core')
settings.gradle
include ':scripts'
```

Re-use - Script client

scripts/src/main/groovy/UserExplorer.groovy

```
def operations = createUserOperations()
(1...100).each {
   try {
      def profile = operations.readProfile(it)
      println profile
   } catch (e) {
      println e.message
```

Re-use - Reducing duplication

settings.gradle

```
rootproject.children.each { project ->
          project.buildFileName = "${project.name}.gradle"
}
```

IntegrationTests in Core

```
sourceSets
   integrationTest {
       java { srcDir 'src/integrationtest/java' }
       resources { srcDir 'src/integrationtest/resources' }
       compileClasspath += sourceSets.main.runtimeClasspath
task integrationTest(type: Test) {
   testClassesDir =
                 sourceSets.integrationTest.output.classesDir
   classpath += sourceSets.integrationTest.runtimeClasspath
```

Keeping Up To Date

```
apply plugin: 'com.github.ben-manes.versions'
```

\$ gw dependencyUpdates

```
The following dependencies have later milestone versions:
- com.fasterxml.jackson.core:jackson-annotations [2.1.2 -> 2.6.0-rc1]
- com.fasterxml.jackson.core:jackson-core [2.1.1 -> 2.6.0-rc1]
- com.fasterxml.jackson.core:jackson-databind [2.1.2 -> 2.6.0-rc1]
- com.github.kevinsawicki:http-request [4.0 -> 9.9]
- joda-time:joda-time [2.3 -> 2.8]
- org.mockito:mockito-core [1.9.0 -> 2.0.8-beta]
```

Generated report file build/dependencyUpdates/report.txt

Code Quality Tools

```
apply plugin: 'checkstyle'
task checkstyleMain(type: Checkstyle) {
  description 'Checks whether code complies with coding rules.'
   ignoreFailures !project.failFastOnError
  doLast {
       if (project.hasProperty('checkstyle.html')) {
         ant.xslt(in: reports.xml.destination,
               style: file("$rulesDir/checkstyle/noframes-sorted.
 xsl"),
               out: new File (reports.xml.destination.parent,
                         name - 'checkstyle' + '.html'))
```

Code Quality Tools

```
apply plugin: 'checkstyle'
task checkstyleMain(type: Checkstyle) {
  description 'Checks whether code complies with coding rules.'
  ignoreFailures !project. failFastOnError
  doLast {
      if (project.hasProperty('checkstyle.html')) {
        ant.xslt(in: reports.xml.destination,
  $ gw checkstyleMain -PfailFastOnError -Pcheckstyle.html
```

Code Quality Tools

```
apply plugin: 'checkstyle'
task checkstyleMain(type: Checkstyle) {
  description 'Checks whether code complies with coding rules.'
   ignoreFailures !project.failFastOnError
  doLast {
       if (project.hasProperty('checkstyle.html')) {
         ant.xslt(in: reports.xml.destination,
               style: file("$rulesDir/checkstyle/noframes-sorted.
 xsl"),
               out: new File (reports.xml.destination.parent,
                         name - 'checkstyle' + '.html'))
```

Continuous Integration



System.getenv("...")

Jenkins

BUILD_NUMBER

Updating the Android SDK

apply plugin: 'android-sdk-manager'

```
$ gw assembleDebug
Android SDK not found. Downloading...
> Configuring > 1/5 projects > :android
SDK extracted at '/Users/leck/.android-sdk'. Writing to local.
properties.
Build tools 21.1.2 missing. Downloading...
> Configuring > 2/5 projects
Platform tools missing. Downloading...
Compilation API android-21 missing. Downloading...
. . .
```

Releasing from CI

- update Version
- commit changes
- upload to Stores



Release from CI - Versioning

```
def versionFile = file('version.properties')

Version version = Version.read(versionFile)

task incrementVersion << {
    Version.increment(versionFile)
}</pre>
```

Release from CI - Versioning

```
def versionFile = file('version.properties')
Version version = Version.read(versionFile)
task incrementVersion buildSrc
  Version.increment(v L src
                          L— main
                               — groovy
                                  - Version.groovy
```

Release from CI - git

```
buildscript {
    dependencies {
       classpath 'org.ajoberstar:qrqit:1.1.0'
ext
   git = org.ajoberstar.grgit.Grgit.open file('.')
git.with {
    add 'version.properties'
    commit (message: "Bumping version to $VERSION CODE")
   push()
```

Publishing - Github

```
extends org.gradle.api.DefaultTask
def release = repository.createRelease(releaseTag)
       .name(releaseName)
       .body(releaseBody)
       .commitish(commitish)
       .draft(draft)
       .create()
assets.each { asset ->
    release.uploadAsset(asset, 'application/octet-stream')
```

buildSrc/src/main/groovy/ReleaseToGithub.groovy

Publishing - Github

app/build.gradle

```
applicationVariants.all { variant ->
    project.task("releaseGithub$variant.name", type: ReleaseToGithub) {
        releaseTag = "${new Date().format('YY.MM.dd')}"
        releaseName = buildReleaseName(variant)
        releaseBody = buildReleaseBody(variant)
        assets = files(apkFile, mappingFile)
        commitish = project.latestCommitHash
        ...
}
```

Publishing - Google Play

app/build.gradle

publishRelease

```
apply plugin: 'com.github.triplet.play'
```

adds Tasks:

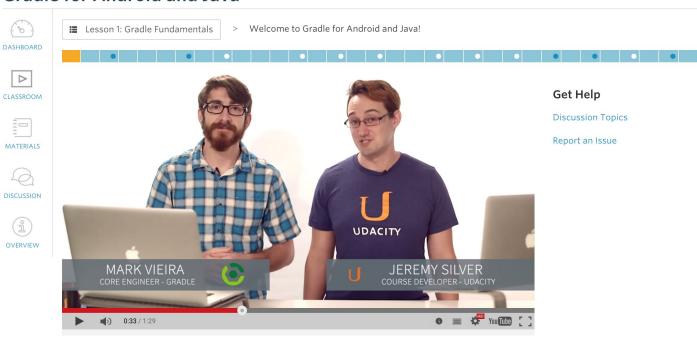
```
bootstrapReleasePlayResources
publishApkRelease
publishListingRelease
```

https://github.com/Triple-T/gradle-play-publisher

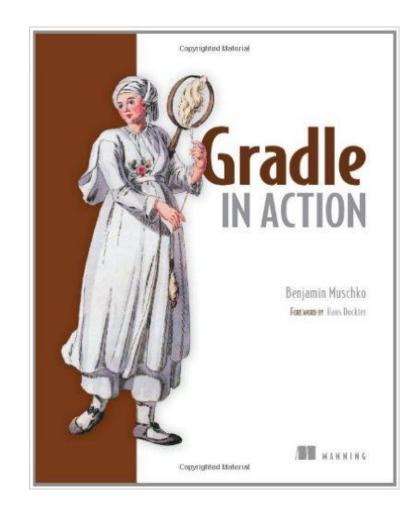
Take away

Get started

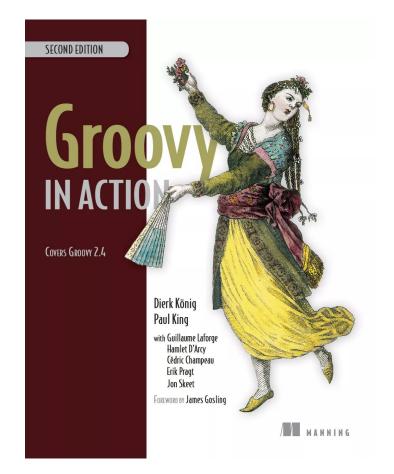
Gradle for Android and Java



Get into it



Go beyond



TDTD - Tester-Driven Talk Development

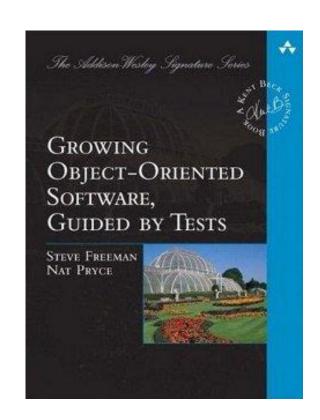
Thanks to the folks from

Android Stammtisch Berlin

for serving as Testers for a lot of the presented material.

Upcoming book:

Growing Subject-Oriented Talks, Guided by Testers





Volker Leck Florian Mierzejewski

Danke!

- devisnik / florianmski
- 🛂 +VolkerLeck / +FlorianMierzejewski
- 🕠 devisnik / florianmski

Tips & Tricks



Shared methods

build.gradle

```
File rules(String relativeFilePath) {
        file("shared/rules/$relativeFilePath")
}
```

app/build.gradle

```
findbugs {
    excludeFilter rules('findbugs/exclude.xml')
    ...
}
```

Shared dependencies

dependencies.gradle

```
ext {
    guava_lib = 'com.google.guava:guava:18.0'
    ...
    testing_libs = [junit_lib, assertJ_lib]
}
```

build.gradle

```
apply from: 'dependencies.gradle'
dependencies {
    compile guava_lib
    ...
    testCompile testing_libs
}
```

Shared buildConfig - before

app/build.gradle

```
android {
    buildTypes {
        debug
             buildConfigField 'String', 'TAG', '"DEBUG TAG"'
        release
             buildConfigField 'String', 'TAG', '"RELEASE TAG"'
```

Shared buildConfig - after

app/build.gradle

```
android {
    buildTypes {
        debug {
             tag 'DEBUG TAG'
        release {
             tag 'RELEASE TAG'
```

Shared buildConfig - how

buildconfig.gradle

```
ext.tag = { tag ->
    buildConfigField 'String', 'TAG', '"'+ tag +'"'
}
```

build.gradle

```
android {
    buildTypes.all {
        apply from: 'buildconfig.gradle', to: delegate
    }
    ...
}
```

Auto Login

```
task autoLogin(type: com.novoda.gradle.command.Input) {
   script {
      text 'my fb email@gmail.com'
      down()
      text 'awesomeSuperStrongPwd92'
      enter()
```

Sync images

```
task syncPhotos(type: com.novoda.gradle.command.Files) {
   deviceId {
     def moto = android.command.devices().find {
        it.brand() == 'motorola'
      moto.id
    script {
      def backupDir = mkdir 'motoPhoto'
      list('/sdcard/DCIM/Camera/').findAll { image ->
                 !new File(backupDir.path, image.name).exists()
      }.each { image ->
                 pull image.path + image.name, backupDir.path
```

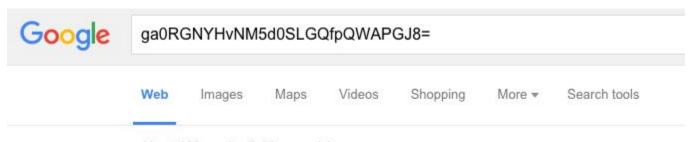
Library Shrinking

```
dependencies {
    classpath 'net.sf.proguard:proguard-gradle:4.11'
task shrinkGuava(type: proguard.gradle.ProGuardTask) {
    injars configurations.injar.files
    libraryjars configurations.libjar.files
    outjars file("build/libs/guava ${guavaVersion} base.jar")
    configuration 'quava.proguard'
guava.proguard:
-dontoptimize
-dontobfuscate
-keep public class com.google.common.base.** { public *; }
```

War story - Facebook

Facebook docs on generating key hashes:

\$ ga0RGNYHvNM5d0SLGQfpQWAPGJ8=



About 169 results (0.22 seconds)

War story - Facebook

```
def message =
'keytool error: java.io.IOException: Keystore was tampered with, or
password was incorrect\\n'
def digest = MessageDigest.getInstance('SHA-1')
def digestBytes = digest.digest(message.bytes)
println digestBytes.encodeBase64()
```

\$ ga0RGNYHvNM5d0SLGQfpQWAPGJ8=

War story - Facebook

```
$ qw app:signingReport
def signingMeta = com.android...tasks.SigningReportTask.metaClass
MetaMethod metaFinger = signingMeta
                          .getStaticMetaMethod('getFingerprint',
                              java.security.cert.Certificate, String)
signingMeta.static.getFingerprint = { Certificate cert, String alg ->
   def base64 = MessageDigest.getInstance(alg)
                      .digest(cert.encoded).encodeBase64()
  metaFinger.invoke(delegate, cert, hashAlgorithm) +
                      "\n$hashAlgorithm (base64): $base64"
```

Groovy on Android

```
buildscript {
  repositories {
     jcenter()
 dependencies {
      classpath 'com.android.tools.build:gradle:1.1.0'
      classpath 'org.codehaus.groovy:gradle-groovy-android-plugin:0.3.6'
apply plugin: 'groovyx.grooid.groovy-android'
dependencies {
    compile 'org.codehaus.groovy:groovy:2.4.3:grooid'
```