

How to

Gradle with Android

Patrick Favre-Bulle

What's happening?

INTRODUCTION

GRADLES FEATURES

TECHNICAL DETAILS

HANDS ON

A Brief Introduction

(Yawn!)

GET YOUR FACTS STRAIGHT:

Gradle is Project Automation Tool

Gradle uses Groovy as DSL

Gradle \neq Android Plugin for Gradle

TIMELINE:

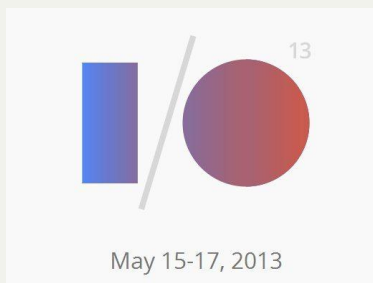
A bunch of Features get released

2013

2014

NOW

May: Google introduces
Gradle + Android Studio

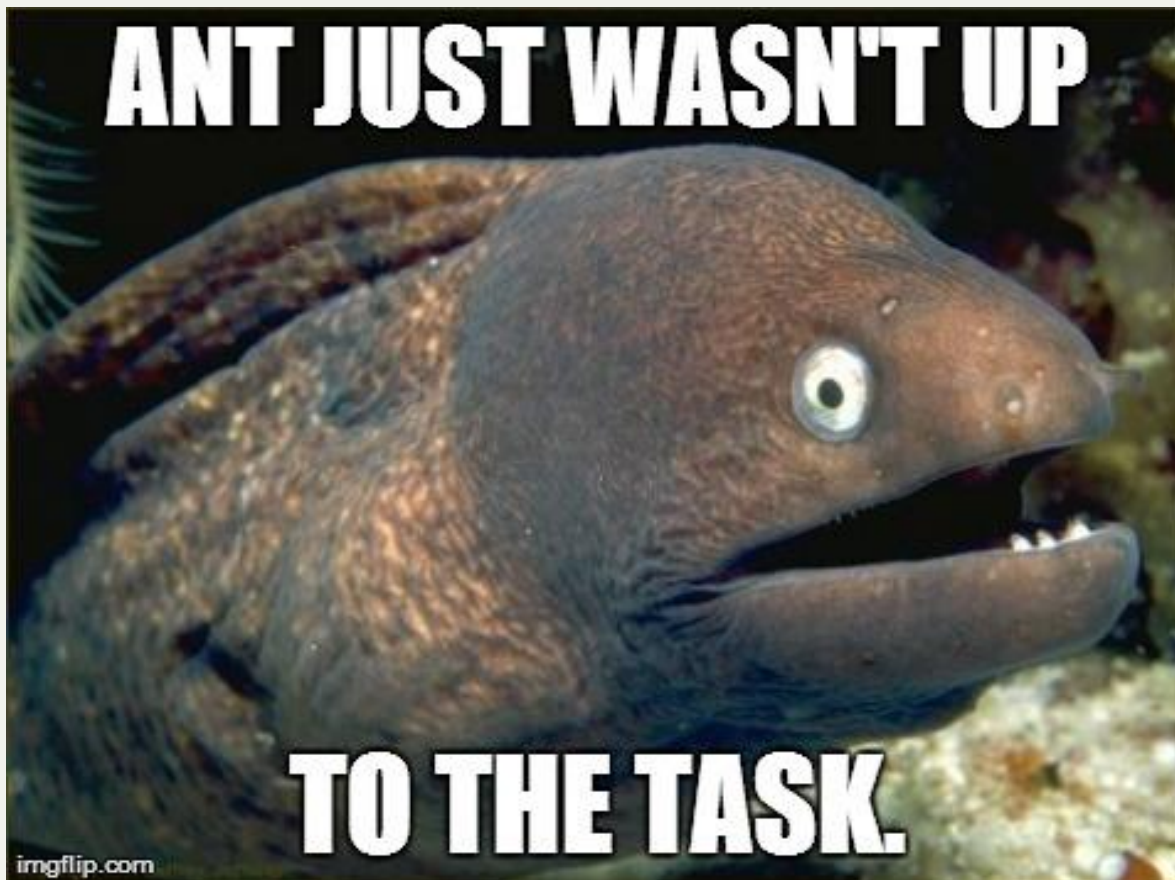


Gradle: v1.11
Android Plugin: 0.9.0
Android Studio: 0.5.1



**Why a new Build System,
Google, oh good why?**

ANT IS NICE BUT



TO THE TASK.

Why is

Gradle Awesome?

Simple, Declarative, Domain-specific Language

Why u need dis?

Convention over Configuration but Flexible

Android Build DSL

Gradle does not force its way (unlike others *cough* Maven)

A Single Build System with good integration

Why u need dis?

IDE, CI-Server, Commandline same tooling

Multi Project Build

Ant support

Dependency Management

Why u need dis?

No Lib-Chaos

Easy handling

Use Maven Repos

A LOT of Android specific Features with the Plugin:

Build Types & Flavors

Signing, NDK, ProGuard, Lint

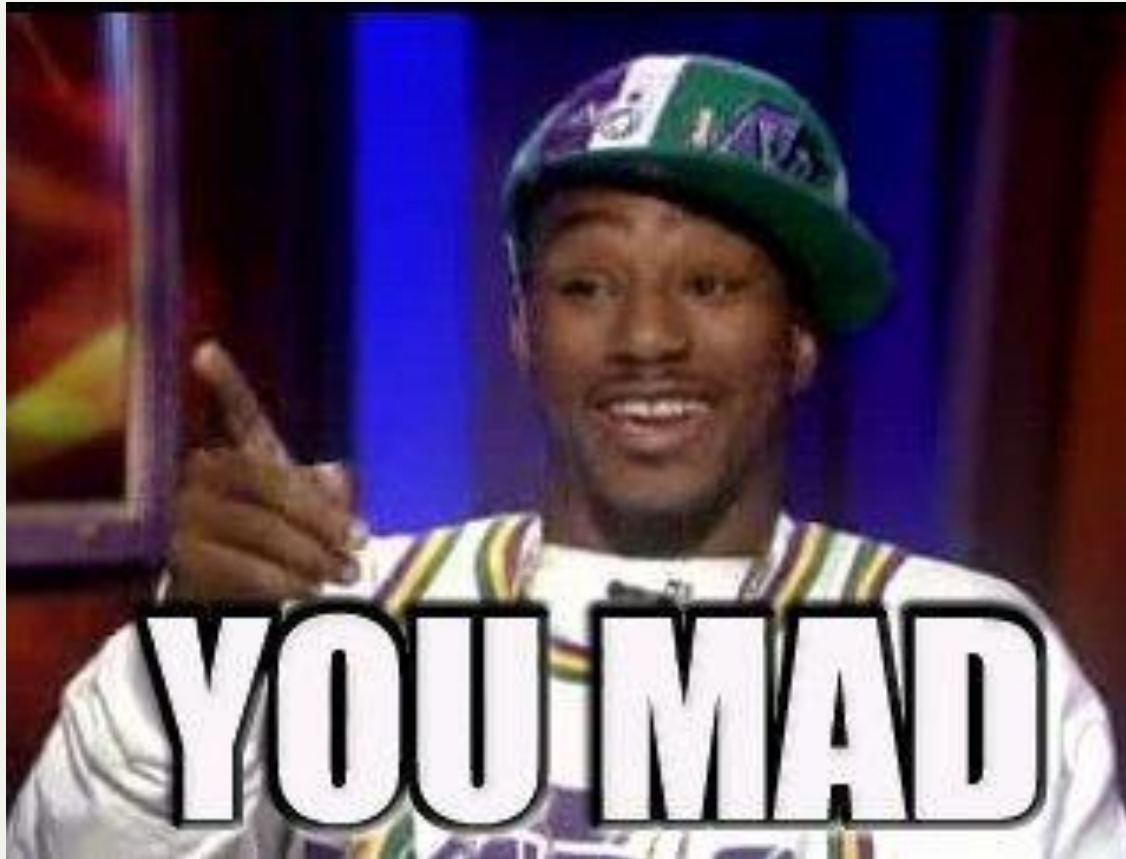
More and more every release (3-6 Weeks)

And don't forget:

DECENT

TESTING SUPPORT

So why not Maven?



How can I start?

Directory Structure

- app_name
 - src
 - main
 - java
 - res
 - AndroidManifest.xml
 - androidTest (new in 0.9)
 - build.gradle
 - settings.gradle
 - build.gradle

Hello World (1)

build.gradle

```
buildscript {  
    repositories {  
        mavenCentral()  
    }  
    dependencies {  
        classpath 'com.android.tools.build:gradle:0.9.+'  
    }  
}  
  
allprojects {  
    repositories {  
        mavenCentral()  
    }  
}
```

GRADLE

settings.gradle

```
include ':app_name'
```

GRADLE

Hello World (2)

app_name/build.gradle

GRADLE

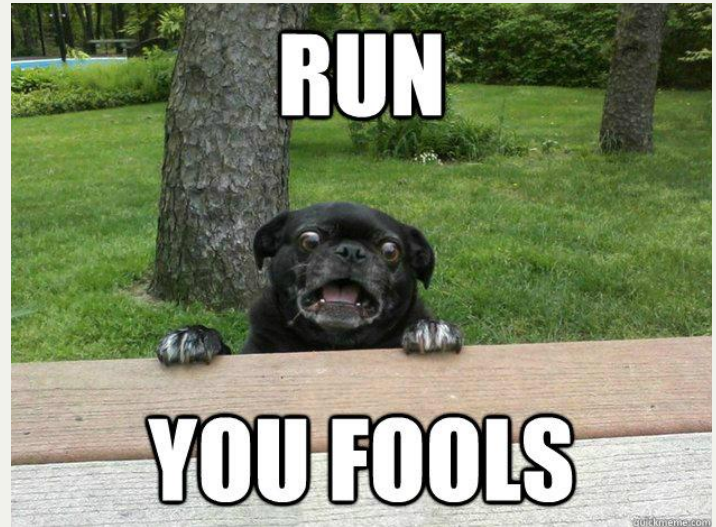
```
apply plugin: 'android'

android {
    compileSdkVersion 19
    buildToolsVersion "19.0.3"

    defaultConfig {
        minSdkVersion 8
        targetSdkVersion 19
    }
}

dependencies {
    compile 'com.android.support:support-v4:19.0.1'
    compile 'com.android.support:appcompat-v7:19.0.1'
}
```

Run Hello World



```
gradle clean
```

```
gradle build // (check & assemble)
```

```
gradle installDebug
```

Dependencies

GRADLE

```
...
dependencies {
    compile 'com.android.support:support-v4:19.0.1'
    compile 'com.google.zxing:core:2.0'
    compile 'com.actionbarsherlock:actionbarsherlock:4.4.0@aar'
    compile files('libs/volley.jar')
    compile project(':SomeAndroidLibModule')
}
...
```

Use any Maven Repo. Yay!

The new Format: AAR



Modules

- app_name
 - src/main/java
 - build.gradle
- lib_project1
 - src/main/java
 - build.gradle
- lib_project2
 - src/main/java
 - build.gradle

`settings.gradle`

```
include ':app_name', ':lib_project1', ':lib_project1'
```

Default Config

```
android {  
    ...  
    defaultConfig {  
        versionCode 12  
        versionName "App-Name"  
        minSdkVersion 10  
        targetSdkVersion 19  
    }  
}
```

GRADLE

Set Manifest Properties for all Builds

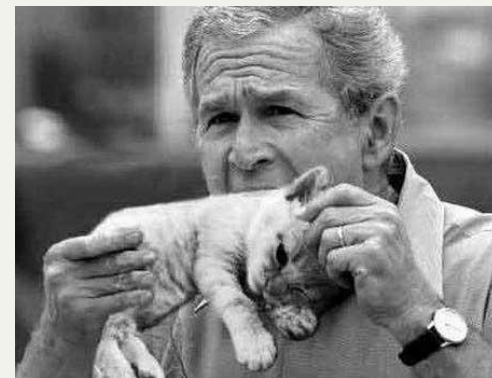
Build Types

GRADLE

```
android {  
    ...  
    buildTypes {  
        debug {  
            runProguard false  
        }  
        release {  
            runProguard true  
            proguardFiles getDefaultProguardFile('proguard-android.txt'), 'pG.txt'  
        }  
    }  
}
```

More: Package-Suffix, Siginig config

Build Flavors



Think: Free Version vs. Paid Version

	Debug	Release
Free-Flavor	Debug-Free	Release-Free
Paid-Flavor	Debug-Paid	Release-Paid

Build Type + Product Flavor = Build Variant

Build Flavors (2)

GRADLE

```
android {  
    ...  
  
    productFlavors {  
        flavor1 {  
            packageName "com.example.flavor1"  
            versionCode 20  
        }  
  
        flavor2 {  
            packageName "com.example.flavor2"  
            minSdkVersion 14  
        }  
    }  
}
```

source dir, config, assemble cmds

Tasks & Ant

```
task readGitLog(dependsOn: assembleDebug) {  
    def git_version = new ByteArrayOutputStream()  
    exec {  
        commandLine 'git', 'log', '-1', '--format=%h'  
        standardOutput = git_version  
    }  
    git_version = git_version.toString().trim()  
    ...  
}
```

GRADLE

```
task hello {  
    ant.echo('hello from Ant')  
    ant.properties['buildDir'] = buildDir  
    ant.taskdef(...)  
}
```

GRADLE

Testing

`src/androidTest`

`assembleTest` installs and runs Testapp

`src/build/androidTest-results` Report as XML

Gradle Wrapper

Convenience Feature:

Auto-DLs Gradle Distribution to
`$USER/.gradle`

Migrating



~~the twilight saga~~

eclipse



esme libral7

Eclipse with Gradle

Install Gradle & Android Plugin Change Directories:

GRADLE

```
android {  
    sourceSets {  
        main {  
            manifest.srcFile 'AndroidManifest.xml'  
            java.srcDirs = ['src']  
            resources.srcDirs = ['src']  
            res.srcDirs = ['res']  
            assets.srcDirs = ['assets']  
        }  
  
        androidTest.setRoot('tests')  
    }  
}
```

Issues

I wouldn't call it fast

Constant Development

Added complexity

Common Mistakes

Missing SDK Resources

Outdated Versions

Don't share IDE stuff

Just show

me the **stuff!**

Ready for Prime Time?

Sure.



End.