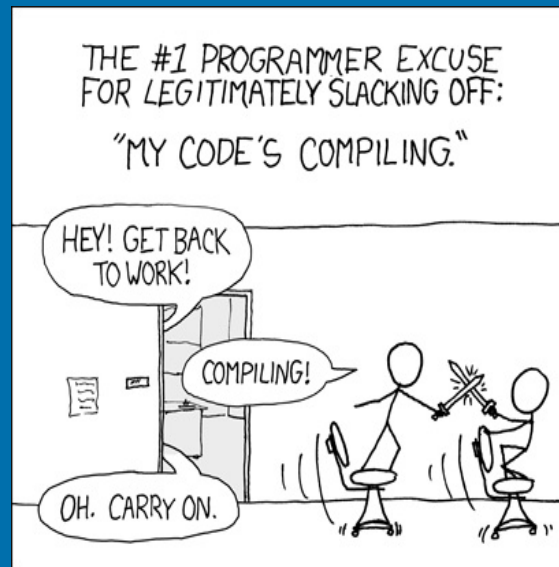


Workshop

Let Gradle build your Java apps

github.com/weel/gradle-java-workshop



Morten Weel Johnsen

About me

- Developer at Iterate
- Android, Java, Frontend/backend developement
- Migrated from Maven/Ant/Make to gradle

Expected outcome?

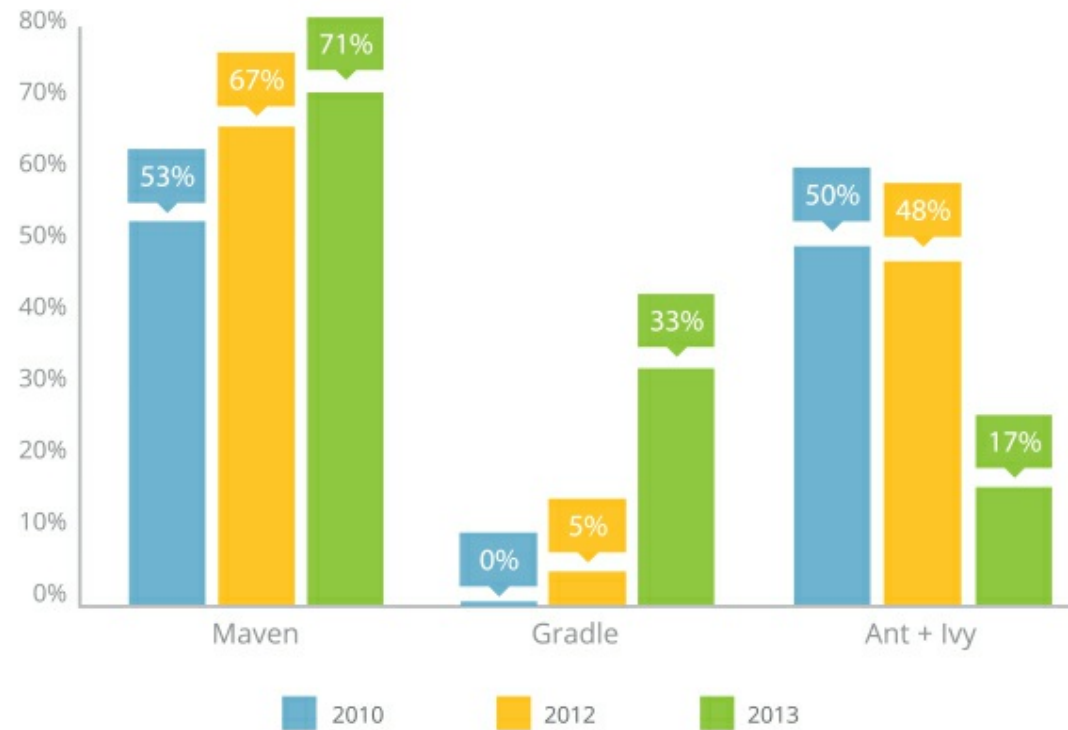
Be able to ...

- Build java applications with Gradle
- Migrate from Maven to Gradle

Agenda

- Motivation & Background
- Maven vs Gradle
- Gradle Intro
- Build Java Webapps with Gradle
- Migrating from Maven to Gradle
- Bonus

Build Tools Popularity - Late 2010 to Mid 2013



Why choose Gradle over Maven

- DSL instead of XML
- Documentation is awesome
- Both uses maven repository
- No boilerplate
- Convention over configuration
- Gradle Wrapper
- Gradle Daemon for faster builds

Getting started:

```
git clone git@github.com:weel/gradle-java-  
workshop.git
```

Task 1 - Getting started

1. Download from <http://www.gradle.org/> (version 2.1)
2. Install gradle & add to path
3. Run 'gradle init' in an empty folder
4. Run 'gradle tasks' to get a list of all tasks available

Projects and tasks

Everything in Gradle sits on top of two basic concepts: projects and tasks

Gradle Tasks

Documentation

```
task copyPoems(type: Copy) {  
  from 'text-files'  
  into 'build/poems'  
}
```

Task 2 - Gradle tasks

- 1.
- 2.
- 3.
- 4.

Gradle Plugins

Documentation

apply plugin: 'java'

#Other common plugins: android, jetty, maven +++

Maven vs Gradle dependencies

```
<dependency>  
  <groupid>com.google.guava</groupid>  
  <artifactid>guava</artifactid>  
  <version>16.0.1</version>  
</dependency>
```

compile 'com.google.guava:guava:16.0.1'

Or

compile group:'com.google.guava' name: 'guava' version: '16.0.1'

Gradle structure

```
application/  
├── build.gradle    # Contains build script  
├── gradlew        # Downloads gradle  
└── wrapper  
    ├── gradle-wrapper.jar  
    └── gradle-wrapper.properties
```

Minimal Gradle #1

```
# empty build.gradle
```

Minimal Gradle #2

```
# build.gradle
apply plugin:'java'

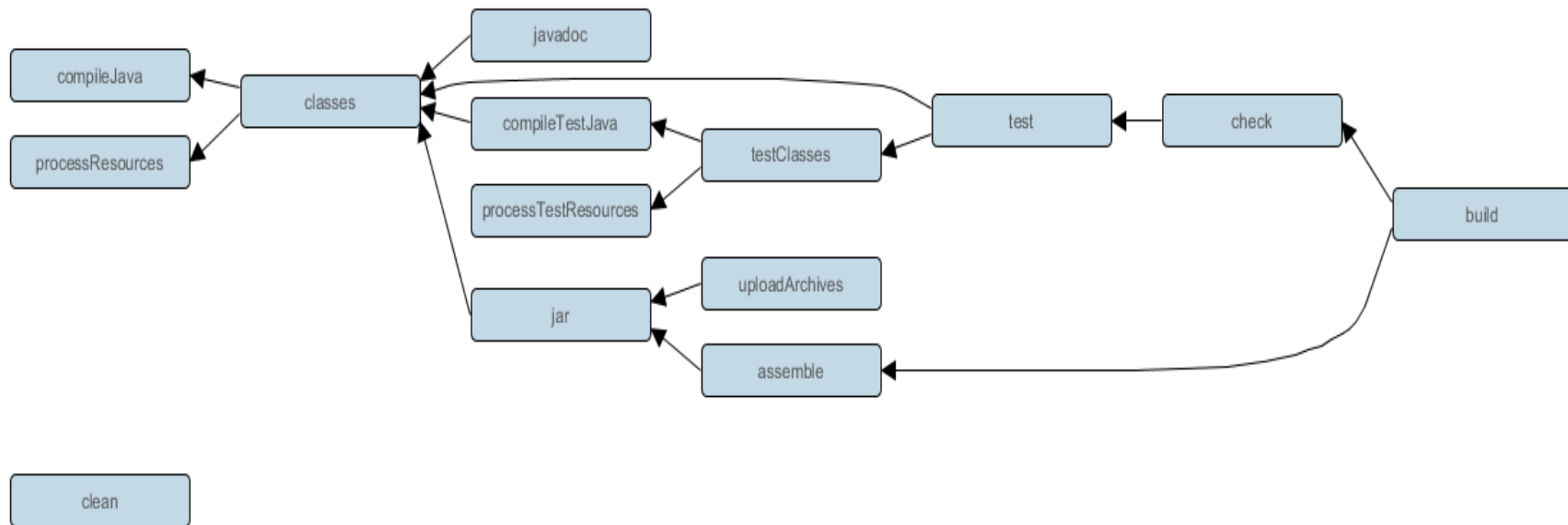
group = 'no.gradlejavazone.samples'
version = '1.0.0'
description = "Samples"

sourceCompatibility = 1.8
targetCompatibility = 1.8

repositories {
    maven { url "http://repo.maven.apache.org/maven2" }
}

dependencies {
    compile group: 'org.hibernate', name: 'hibernate', version:'3.1.0'
    compile 'org.hibernate:hibernate-core:3.6.7'
}
```


The Java plugin



Archetypes

```
gradle init --type java-library
```

Table 23.4. Java plugin - default project layout

Directory	Meaning
src/main/java	Production Java source
src/main/resources	Production resources
src/test/java	Test Java source
src/test/resources	Test resources
src/ <i>sourceSet</i> /java	Java source for the given source set
src/ <i>sourceSet</i> /resources	Resources for the given source set

Task 2 - Running a simple junit testcase

1. Create an empty build.gradle file
2. Add dependency for latest junit version
3. Create a Junit test in src/test/java/
4. Run tests

Task 3 - Running a simple Java application - Tutorial

1. Apply plugin 'application'
2. Create a java file in src/main/java with a main method
3. Add guava as dependency
4. Run

Defining Main-class attribute

```
jar {  
    manifest {  
        attributes 'Main-Class': 'net.petrikainulainen.gradle.HelloWorld'  
    }  
}
```

Task 4 - Migrating from Maven to Gradle manually

1. Based on simple REST example Restify

Task 4 - Migrating from Maven to Gradle using migrating tool

1.

Gradle ShadowJar plugin

```
shadowJar {  
  mergeServiceFiles {  
    exclude 'META-INF/*.SA'  
  }  
}
```

Gradle Wrapper

Minimal wrapper to download and use specified Gradle version.

```
./gradlew tasks # Downloads gradle and runs command to show tasks
```

Get started from terminal

```
$ ./gradlew tasks # Lists all tasks available
```

```
$ ./gradlew jettyRun # Builds and run on jetty
```

```
$ ./gradlew jettyRunWar # Builds and run war on jetty
```

```
$ ./gradlew build
```

Bonus

<http://www.gradle.org/learn> Gradle Goodness

Gradle daemon

Add to gradle.properties in folder ~/.gradle

org.gradle.daemon=true

or start with

gradle --daemon

Encoding UTF-8

#For specific task compileJava

```
compileJava.options.encoding = 'UTF-8'
```

or for all compile tasks

```
    apply plugin: 'java'
```

```
tasks.withType(Compile) {  
    options.encoding = 'UTF-8'  
}
```

Publishing artifacts

```
apply plugin: 'java'  
apply plugin: 'maven'
```

```
group = 'com.gradleware.samples'
```

```
uploadArchives {  
    repositories.mavenDeployer {  
        repository(url: "file:///Users/mccm06/Documents/Temp/Scratch/mytemprepo/")  
    }  
}
```

Profiling

```
$ gradle --profile build
```

Profiled with tasks: build

Run on: 2010/11/18 - 22:18:45

Summary

Total Build Time	14.421
Startup	5.749
Settings and BuildSrc	0.856
Loading Projects	0.196
Configuring Projects	4.480
Total Task Execution	2.988

Configuration

: 4.309

Task Execution

:	2.988	(total)
:test	2.642	
:jar	0.213	
:compileJava	0.099	
:compileTestJava	0.008	
:compileTestGroovy	0.007	
:compileGroovy	0.005	
:processTestResources	0.004	
:classes	0.002	Did No Work
:testClasses	0.002	Did No Work
:assemble	0.002	Did No Work
:processResources	0.002	
:build	0.001	Did No Work
:check	0.001	Did No Work

Autocompletion in bash

<https://gist.github.com/nolanlawson/8694399>

IDE support

Master your tool

“I’ve known people who have not mastered their tools who are good programmers, but not a tool master who remained a mediocre programmer.” – Kent Beck

Questions?