
```
BUILD SUCCESSFUL
```

Alexander Mera (@alexmera) — @CLOJUG [23MAY2015]

AGENDA

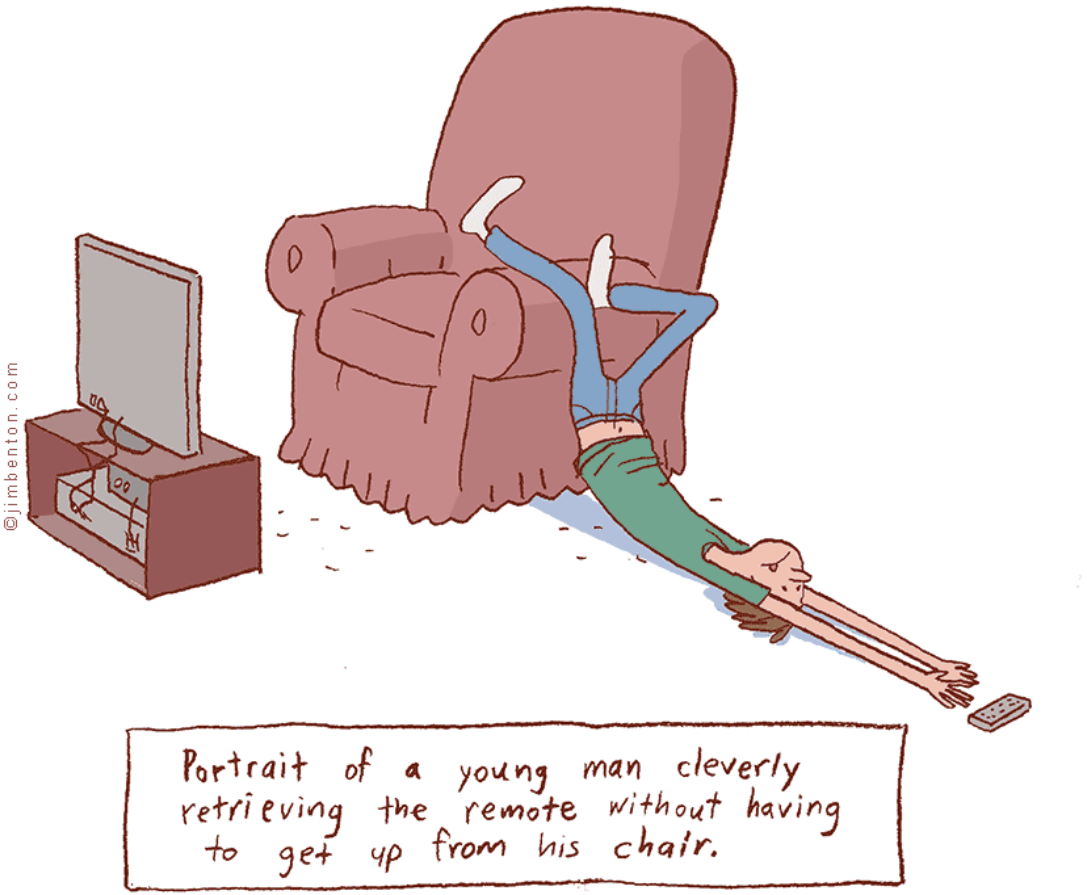
1. ¿Por qué necesito una *build tool*?
2. La evolución de las *build tools* para Java.
    a. Apache Ant + Ivy
    b. Apache Maven
    c. Gradle
3. ¿Qué es la gestión de dependencias?
    a. Apache Maven
    b. Ivy

AGENDA (cont.)

4. Introducción a Gradle: *Project Automation Tool*
    a. Principales características y funcionalidades
    b. Instalación y configuración
    c. Gradle Wrapper
5. Demostración: Aplicación Java sencilla

¿Por qué necesito una *build tool*?

Porque “soy perezoso”…



…y “prefiero evitar la fatiga”…

…de repetir una y otra vez una serie de tareas con el único propósito de lograr siempre el mismo objetivo.

¿Cúal es el objetivo de las *build tools*?

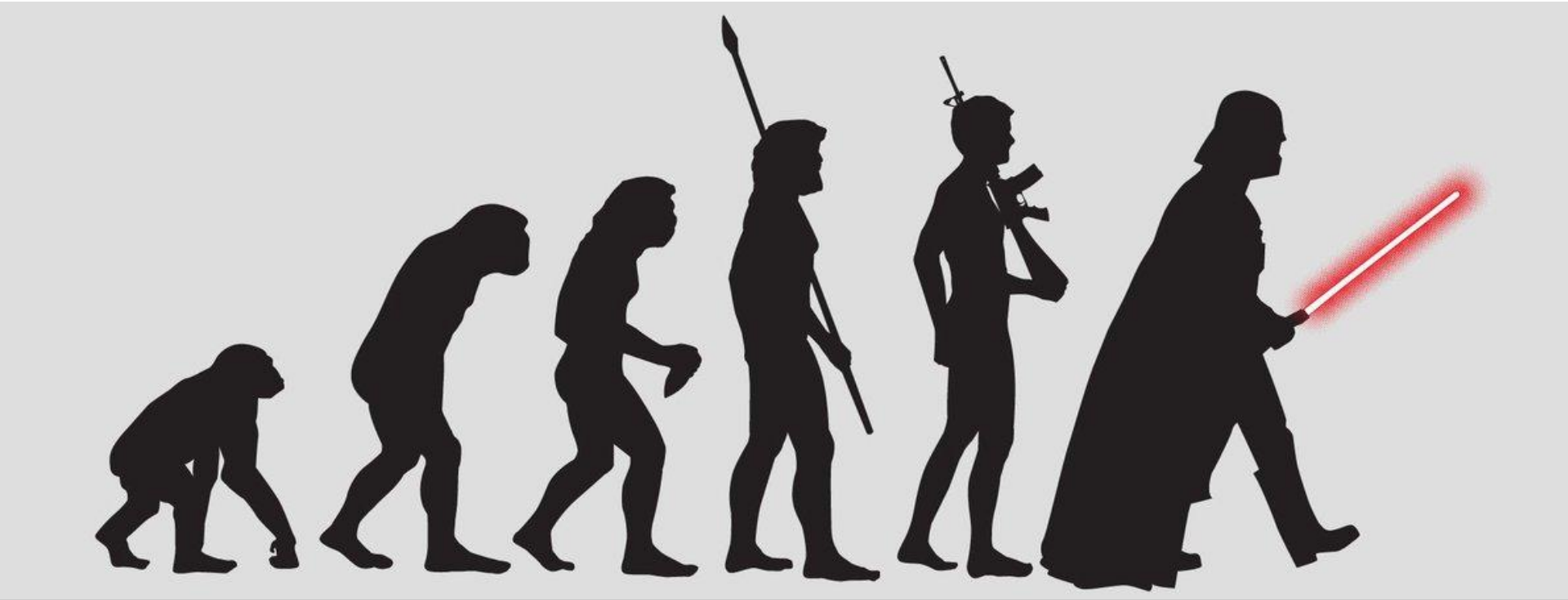
El principal objetivo de las *build tools* es el de **compilar y construir** una **imagen de software** usable a partir de su código fuente.



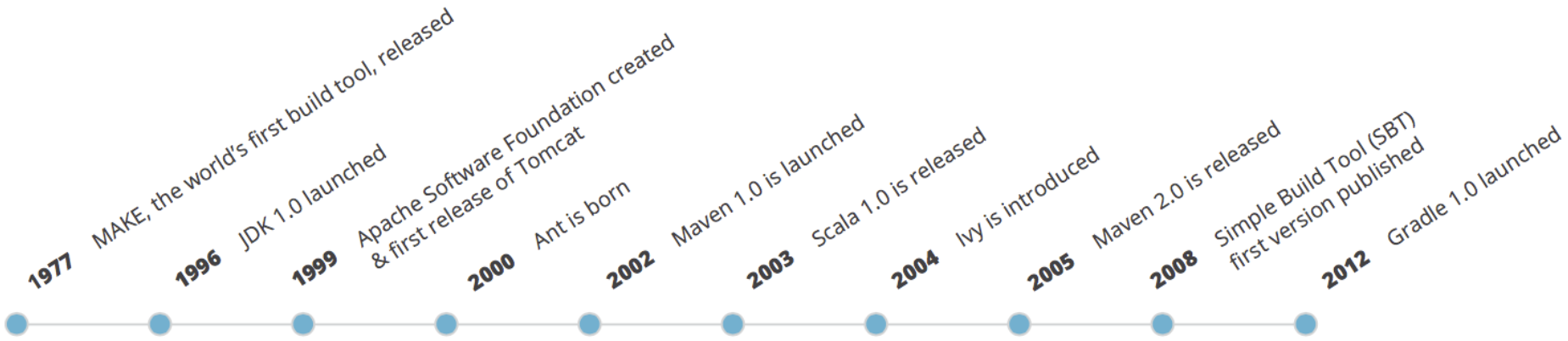
Una “buena” *build tool* debería:

- Gestionar dependencias
- Permitir compilación incremental
- Gestionar los recursos
- Soportar diferentes perfiles
- Facilitar la automatización

La evolución de las *build tools* (JVM)



Línea de tiempo




```
<project>
    <target>
        <task>
```

- Lanzado en el año 2000
- Primer build tool “moderno”
- Control completo
- Basado en XML:
  - build.xml
  - ivy.xml (dependencias)

```
<project xmlns:ivy="antlib:org.apache.ivy.ant" name="java-build-tools"
default="jar">

    <property name="src.dir" value="src"/>
    <property name="build.dir" value="build"/>
    <property name="classes.dir" value="${build.dir}/classes"/>
    <property name="jar.dir" value="${build.dir}/jar"/>
    <property name="lib.dir" value="lib" />
    <path id="lib.path.id">
        <fileset dir="${lib.dir}" />
    </path>

    <target name="resolve">
        <ivy:retrieve />
    </target>

    <target name="clean">
        <delete dir="${build.dir}"/>
    </target>

    <target name="compile" depends="resolve">
        <mkdir dir="${classes.dir}"/>
        <javac srcdir="${src.dir}" destdir="${classes.dir}"
classpathref="lib.path.id"/>
    </target>

    <target name="jar" depends="compile">
        <mkdir dir="${jar.dir}"/>
        <jar destfile="${jar.dir}/${ant.project.name}.jar"
basedir="${classes.dir}"/>
    </target>

</project>
```

Maven™

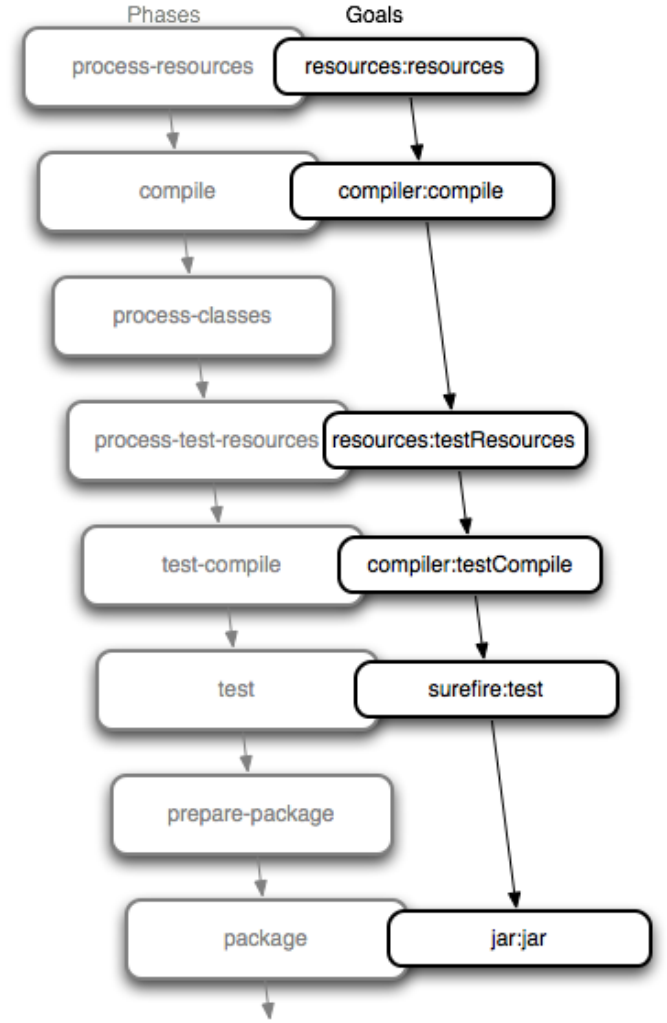
[project]

[lifecycle]

[phase]

[plugin:goal]

- Lanzado en el año 2004
- Convención sobre configuración
- Ciclos de vida
- **Gestión de dependencias**
- Basado en XML: POM.xml



Note: There are more phases than shown above, this is a partial list

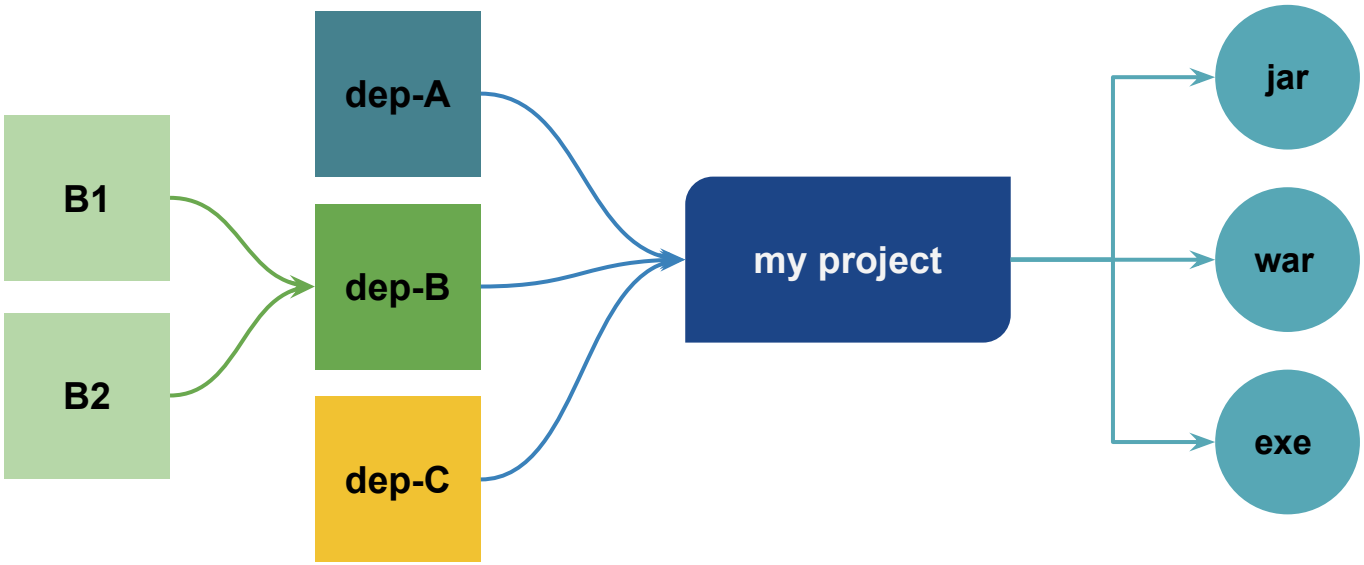
gradle


```
[project]
    [task]
```

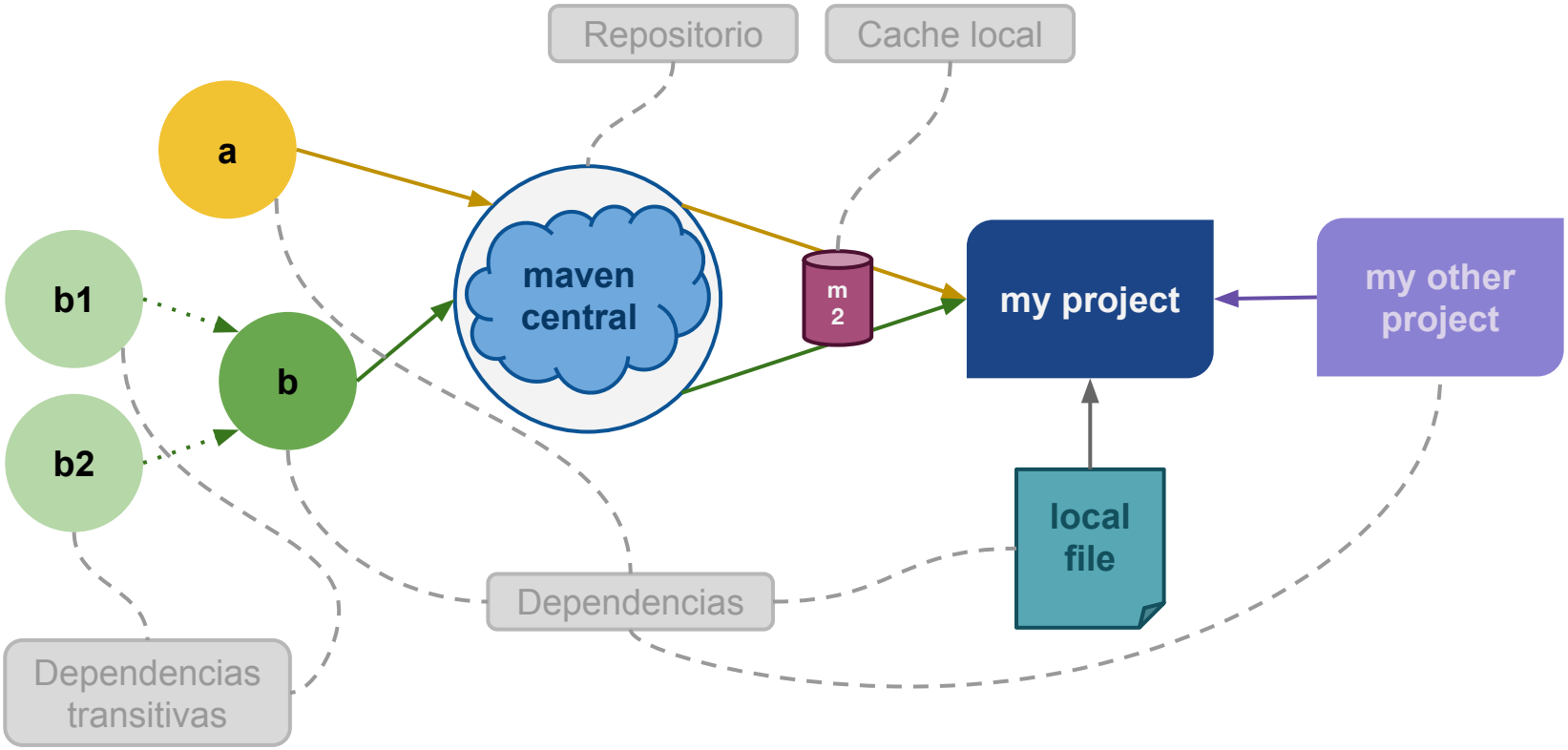
- Lanzado en el año 2009
- Gestión de dependencias
- Convenciones flexibles
- Groovy DSL
- Plugins

Gestión de dependencias

¿Qué es?



Conceptos básicos



```
<ivy-module version="2.0">
    <info organisation="org.apache" module="java-build-tools"/>
    <dependencies>
        <dependency org="junit" name="junit" rev="4.11"/>
        <dependency org="org.hamcrest" name="hamcrest-all" rev="1.3"/>
    </dependencies>
</ivy-module>
```

Maven™

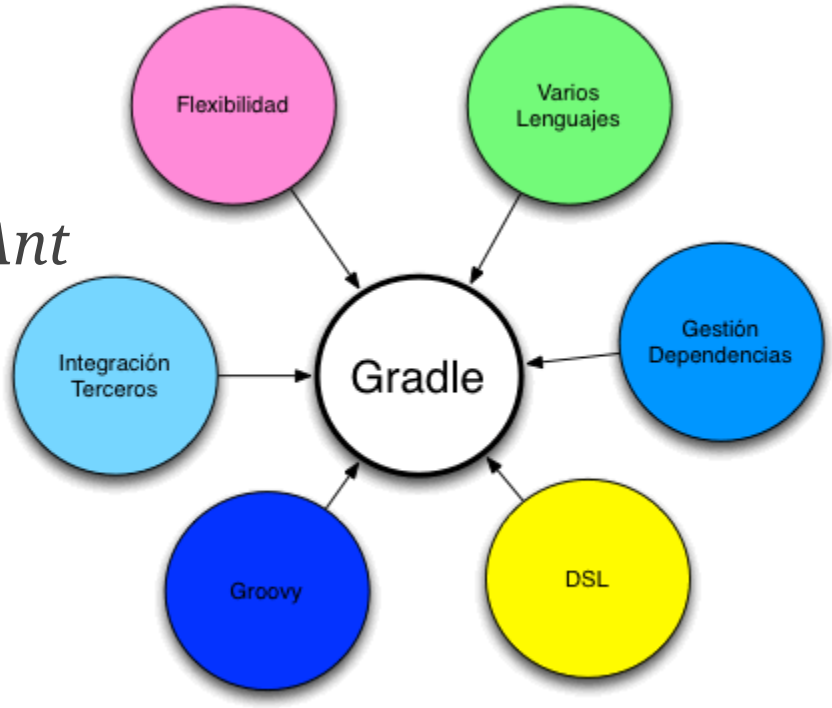
```
<dependencies>
    <dependency>
        <groupId>junit</groupId>
        <artifactId>junit</artifactId>
        <version>4.11</version>
    </dependency>
    <dependency>
        <groupId>org.hamcrest</groupId>
        <artifactId>hamcrest-all</artifactId>
        <version>1.3</version>
    </dependency>
</dependencies>
```

```
dependencies {
    testCompile group: 'junit', name: 'junit', version: '4.11'
    testCompile group: 'org.hamcrest', name: 'hamcrest-all', version: '1.3'
}
```

Introducción a Gradle:
Project Automation Tool

Características y funciones

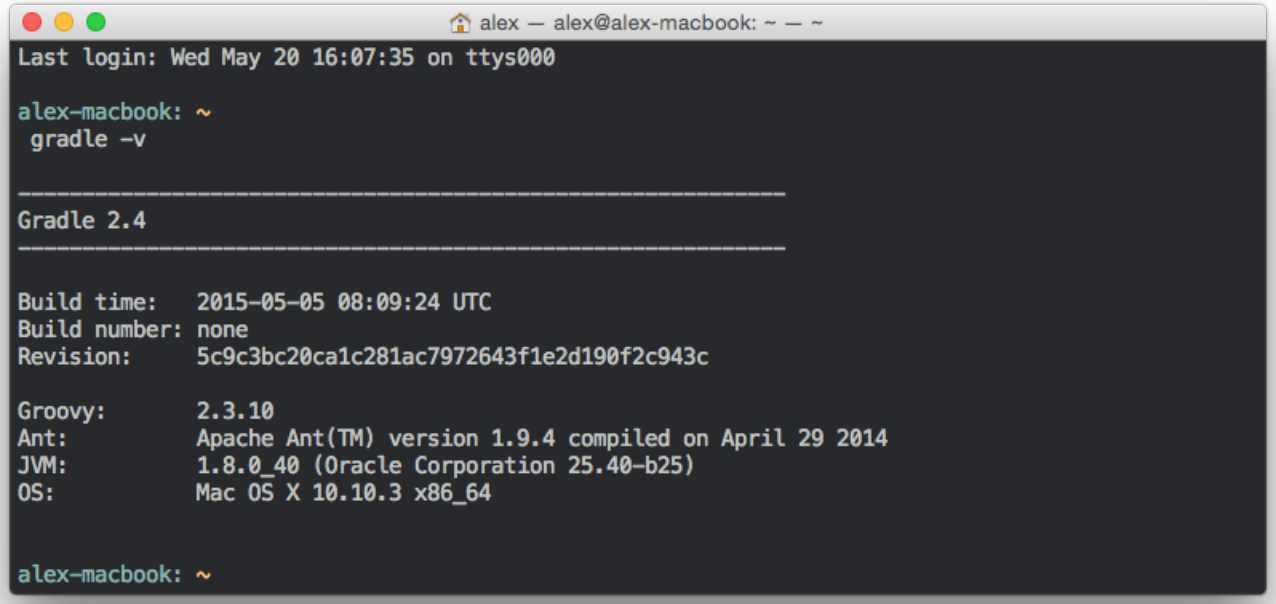
- *Builds* declarativos y por convención
- Multi-proyectos
- *Builds* escritos en Groovy (DSL)
- Integración completa con *Ant Tasks*
- Repositorios Maven, Ivy y archivos locales
- Plugins, plugins, plugins




Downloads - Gradle
Alexander
gradle.org/downloads/
gradle
DOWNLOAD GRADLE
The latest release of Gradle is version 2.4, released on 5th May 2015.
Complete distribution (binaries, sources and offline documentation)
Binary only distribution (no documentation or source code)
Gradle source code (just the Gradle source code; not a usable Gradle installation)

Instalación

- Requiere JDK o JRE versión 6 o superior
- Desempaquetar el archivo **`.zip`** descargado – **`GRADLE_HOME`**
- Adicionar **`GRADLE_HOME/bin`** a la variable de entorno **`PATH`**



¡Hola mundo!

build.gradle

```
task hello {
    doLast {
        println 'Hello world!'
    }
}
```

Output of **gradle -q hello**

```
> gradle -q hello
Hello world!
```

build.gradle

```
task hello << {
    println 'Hello world!'
}
```

¡Es Groovy!

build.gradle

```
task upper << {
    String someString = 'mY_nAmE'
    println "Original: " + someString
    println "Upper case: " + someString.toUpperCase()
}
```

Output of **gradle -q upper**

```
> gradle -q upper
Original: mY_nAmE
Upper case: MY_NAME
```

build.gradle

```
task count << {
    4.times { print "$it " }
}
```

Output of **gradle -q count**

```
> gradle -q count
0 1 2 3
```

Métodos

build.gradle

```
task checksum << {
    fileList('../antLoadfileResources').each {File file ->
        ant.checksum(file: file, property: "cs_$file.name")
        println "$file.name Checksum: ${ant.properties["cs_$file.name"]}"
    }
}

task loadfile << {
    fileList('../antLoadfileResources').each {File file ->
        ant.loadfile(srcFile: file, property: file.name)
        println "I'm fond of $file.name"
    }
}

File[] fileList(String dir) {
    file(dir).listFiles({file -> file.isFile() } as FileFilter).sort()
}
```

Output of **gradle -q loadfile**

```
> gradle -q loadfile
I'm fond of agile.manifesto.txt
I'm fond of gradle.manifesto.txt
```

Ciclo de vida

settings.gradle

```
println 'This is executed during the initialization phase.'
```

build.gradle

```
println 'This is executed during the configuration phase.'

task configured {
    println 'This is also executed during the configuration phase.'
}

task test << {
    println 'This is executed during the execution phase.'
}

task testBoth {
    doFirst {
      println 'This is executed first during the execution phase.'
    }
    doLast {
      println 'This is executed last during the execution phase.'
    }
    println 'This is executed during the configuration phase as well.'
}
```

Output of **gradle test testBoth**

```
> gradle test testBoth
This is executed during the initialization phase.
This is executed during the configuration phase.
This is also executed during the configuration phase.
This is executed during the configuration phase as well.
:test
This is executed during the execution phase.
:testBoth
This is executed first during the execution phase.
This is executed last during the execution phase.

BUILD SUCCESSFUL

Total time: 1 secs
```

- Inicialización
- Configuración
- Ejecución

Gradle Wrapper

- Batch script para Windows
- Shell script para Linux y OS X
- Debe incluirse en el control de versiones – git
- Luego de integrado, NO requiere de una instalación local
- Garantiza el uso de la versión adecuada de Gradle

¡Úsalo en todos tus builds!

build.gradle

```
task wrapper(type: Wrapper) {
    gradleVersion = '2.0'
}
```

Output of **gradle wrapper --gradle-version 2.0**

```
> gradle wrapper --gradle-version 2.0
:wrapper

BUILD SUCCESSFUL

Total time: 1 secs
```

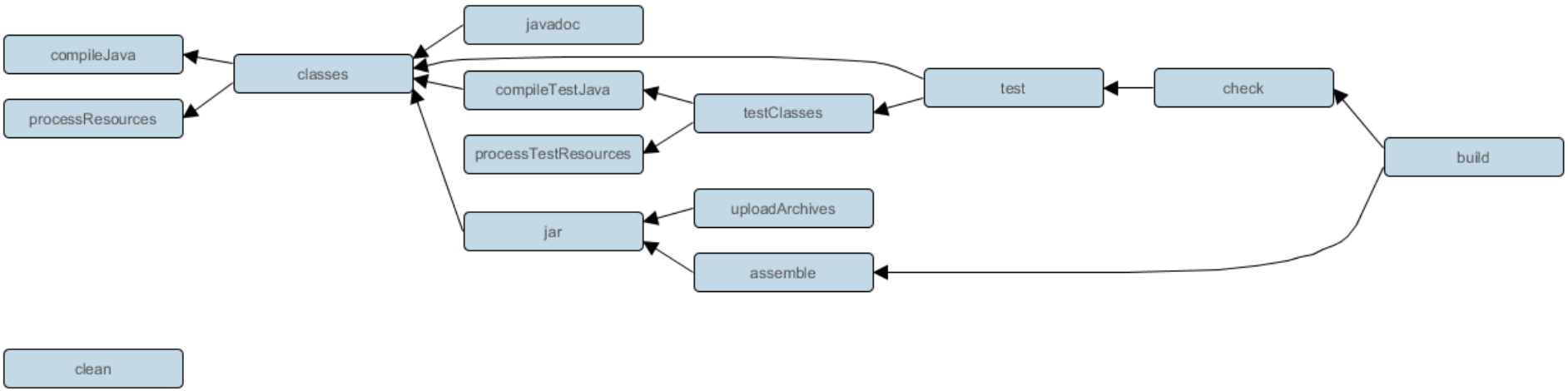
Build layout

```
simple/
  gradlew
  gradlew.bat
  gradle/wrapper/
    gradle-wrapper.jar
    gradle-wrapper.properties
```

Java plugin

build.gradle

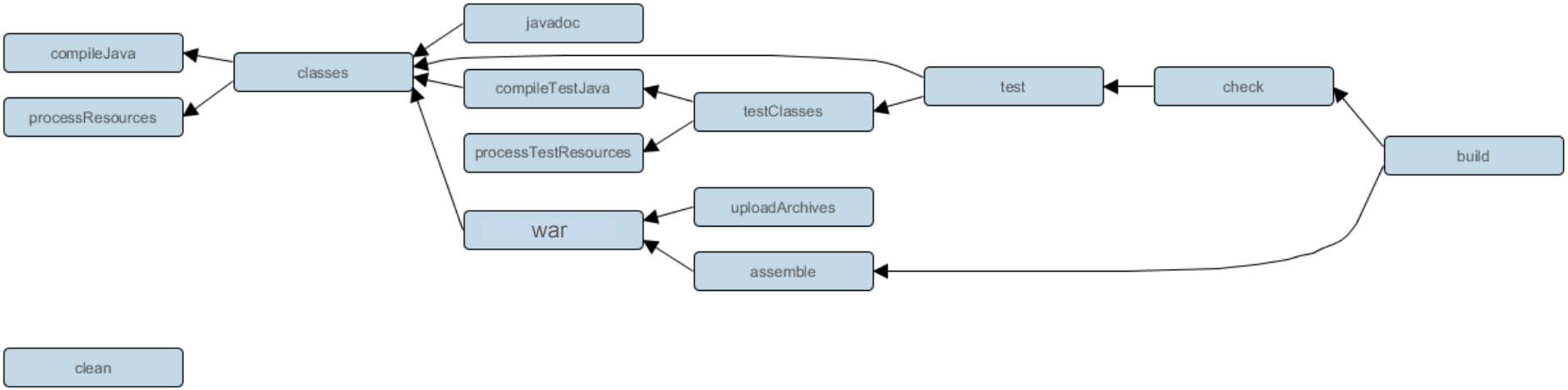
```
apply plugin: 'java'
```



War plugin

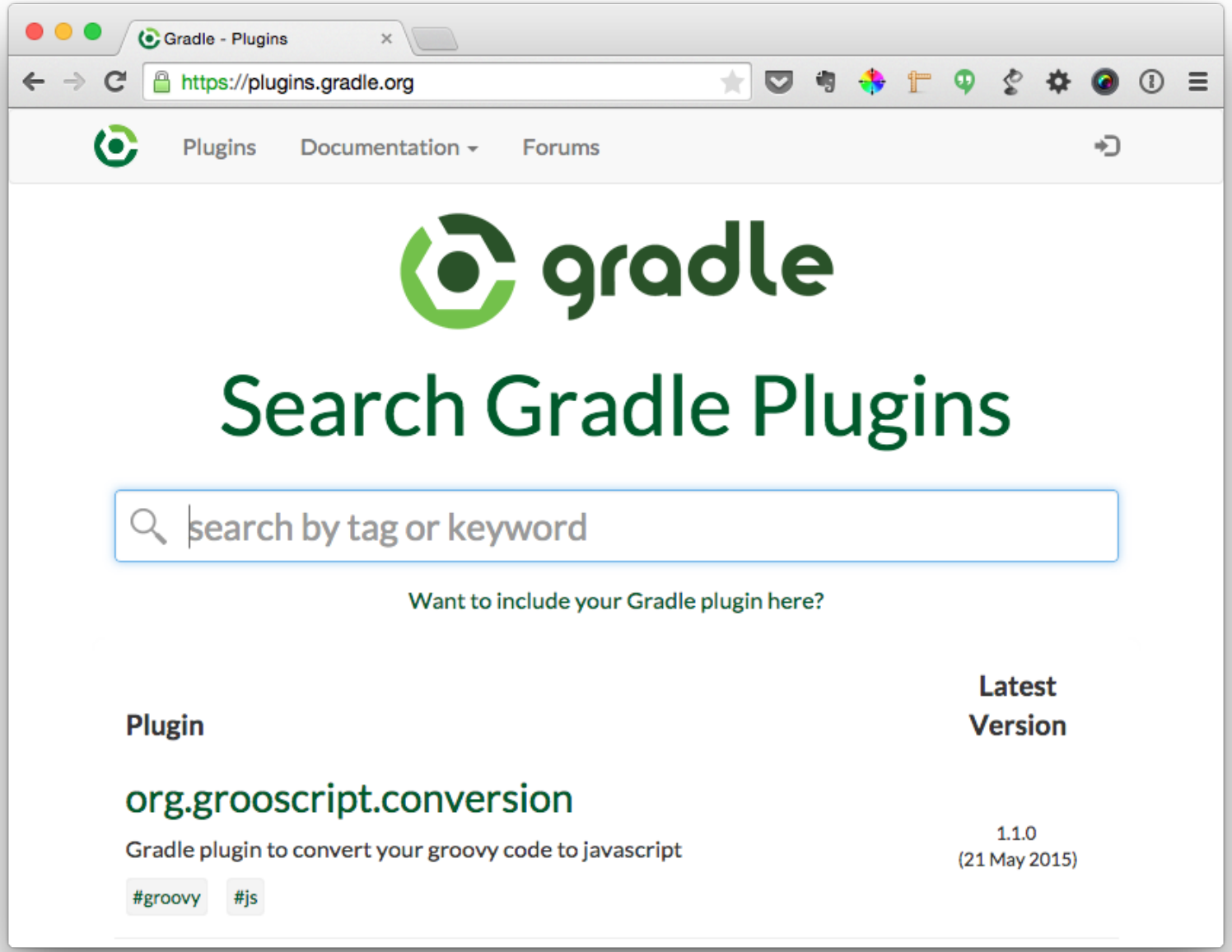
build.gradle

```
apply plugin: 'war'
```



Plugins

Lenguajes	Integración	Proceso
java groovy scala antlr — **En incubación**: assembler c cpp objective-c objective-cpp windows-resources	application ear jetty maven osgi war — **En incubación**: distribution java-library-distribution ivy-publish maven-publish	checkstyle eclipse eclipse-wtp findbugs idea jdepend pmd signing sonar — **En incubación**: jacoco sonar-runner


Gradle - Plugins
https://plugins.gradle.org
Plugins
Documentation
Forums
gradle
Search Gradle Plugins
search by tag or keyword
Want to include your Gradle plugin here?
Plugin
Latest Version
org.grooscript.conversion
Gradle plugin to convert your groovy code to javascript
#groovy
#js
1.1.0
(21 May 2015)

Fuentes

- https://docs.gradle.org/current/userguide/userguide
- http://zeroturnaround.com/rebellabs/java-build-tools-part-1-an-introductory-crash-course-to-getting-started-with-maven-gradle-and-ant-ivy/
- http://zeroturnaround.com/rebellabs/java-build-tools-how-dependency-management-works-with-maven-gradle-and-ant-ivy/
- http://technologyconversations.com/2014/06/18/build-tools/
- http://www.drdobbs.com/jvm/why-build-your-java-projects-with-gradle/240168608