



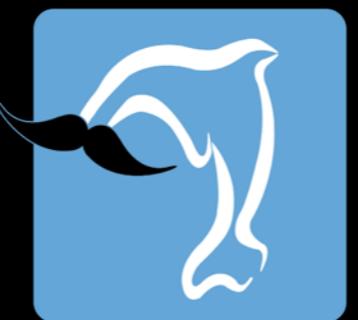
COMO ALTERNATIVA A MAVEN PARA LA CONSTRUCCIÓN DE PROYECTOS.



David Gómez G.
@dgomezg



BIGOTENTIOS MOVEMBER₁₃

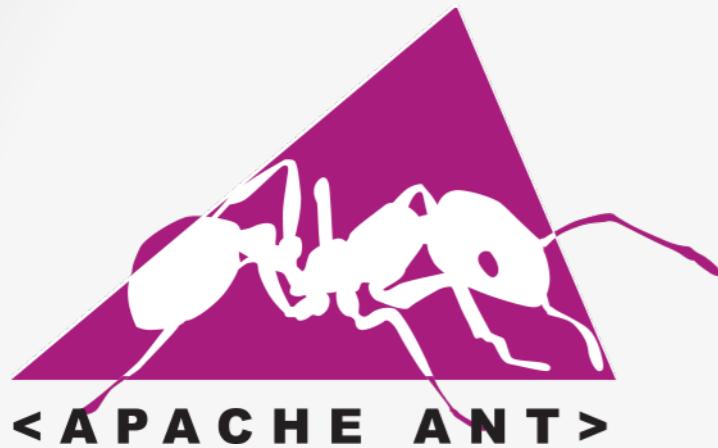


Automatizar la construcción... ¿Por qué?



- Evitar empaquetado manual
- Construir rápidamente y en cualquier entorno
- Builds programadas (CI)

Herramientas



Fichero de descripción

+

Código fuente

maven

+

Gestión de dependencias

 gradle

=

Artefacto



Características



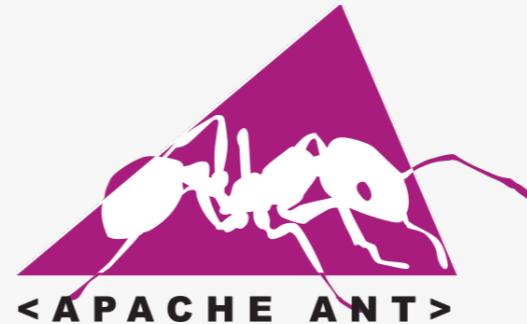
- Basado en Java
- Configuración basada en XML
- Gestor de dependencias adicional (Ivy)
- Muy flexible
 - Todas las tareas se definen específicamente

Pegas

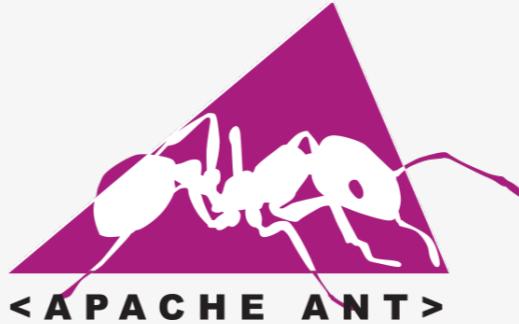


- Muy verboso. XML muy extensos
- Difícil escribir lógica compleja (condiciones)
- Estructura de proyecto libre.
- Difícil de mantener
- Difícil de entender el proceso de construcción.

Ejemplo



Ejemplo



```
<?xml version="1.0" encoding="UTF-8"?>
<project>
    <target name="clean">
        <delete dir="build"/>
    </target>
    <target name="compile">
        <mkdir dir="build/classes"/>
        <javac srcdir="src" destdir="build/classes"/>
    </target>
    <target name="jar">
        <mkdir dir="build/jar"/>
        <jar destfile="build/jar/HelloWorld.jar"
            basedir="build/classes">
            <manifest>
                <attribute name="Main-Class"
                    value="com.autentia.dgomezg.sandbox.ant.HelloWorld"/>
            </manifest>
        </jar>
    </target>

    <target name="run">
        <java jar="build/jar/HelloWorld.jar" fork="true"/>
    </target>
</project>
```

Ejemplo



```

<?xml version="1.0" encoding="UTF-8"?>
<project>
  <target name="clean">
    <delete dir="build"/>
  </target>
  <target name="compile">
    <mkdir dir="build/classes"/>
    <javac srcdir="src" destdir="build/classes" />
  </target>
  <target name="jar">
    <mkdir dir="build/jar"/>
    <jar destfile="build/jar/HelloWorld.jar" >
      <basedir="build/classes" />
      <manifest>
        <attribute name="Main-Class" value="com.autentia.HelloWorld" />
      </manifest>
    </jar>
  </target>
  <target name="run">
    <java jar="build/jar/HelloWorld.jar" />
  </target>
</project>

```

```

dolamroth:ant-sample dgomezg$ ant clean compile jar run
Buildfile: /Users/dgomezg/Documents/workspace/ant-sample/build.xml

clean:

compile:
    [mkdir] Created dir: /Users/dgomezg/Documents/workspace/ant-sample/
build/classes
    [javac] Compiling 1 source file to /Users/dgomezg/Documents/
workspace/ant-sample/build/classes

jar:
    [mkdir] Created dir: /Users/dgomezg/Documents/workspace/ant-sample/
build/jar
    [jar] Building jar: /Users/dgomezg/Documents/workspace/ant-sample/
build/jar/HelloWorld.jar

run:
    [java] Hello World

BUILD SUCCESSFUL
Total time: 1 second
dolamroth:ant-sample dgomezg$

```

Características *maven*

- Convención sobre configuración
 - Estructura de proyecto determinada
 - Proceso de construcción definido
- Gestor de dependencias
- Informes de construcción y de test
- Arquetipos
- Ampliable con Plugins (Mojos)



Pegas

maven

- Estructura demasiado rígida
- Extensión complicada
- Fichero de build verboso
 - Especialmente si configuramos plugins

Ejemplo *maven*



Ejemplo *maven*

```
<?xml version="1.0" encoding="UTF-8"?>
<project>
    <modelVersion>4.0.0</modelVersion>
    <groupId>com.autentia.dgomezg.sandbox</groupId>
    <artifactId>mvn-sample</artifactId>
    <version>1.0.0</version>
</project>
```



Ejemplo maven

```
<?xml version="1.0" encoding="UTF-8"?>
<project>
  <modelVersion>4.0.0</modelVersion>
  <groupId>com.autentia.dgomezg.sand</groupId>
  <artifactId>mvn-sample</artifactId>
  <version>1.0.0</version>
</project>
```

```
dolamroth:mvn-sample dgomezg$ mvn package
[INFO] Scanning for projects...
[INFO]
[INFO] -----
[INFO] Building mvn-sample 1.0.0
[INFO] -----
[INFO] --- maven-resources-plugin:2.5:resources (default-resources) @ mvn-sample ---
[debug] execute contextualize
[WARNING] Using platform encoding (UTF-8 actually) to copy filtered resources, i.e. build is platform dependent!
[INFO] Copying 0 resource
[INFO]
[INFO] --- maven-compiler-plugin:2.3.2:compile (default-compile) @ mvn-sample ---
[INFO] Nothing to compile - all classes are up to date
[INFO]
[INFO] --- maven-resources-plugin:2.5:testResources (default-testResources) @ mvn-sample ---
[debug] execute contextualize
[WARNING] Using platform encoding (UTF-8 actually) to copy filtered resources, i.e. build is platform dependent!
[INFO] Copying 0 resource
[INFO]
[INFO] --- maven-compiler-plugin:2.3.2:testCompile (default-testCompile) @ mvn-sample ---
[INFO] Nothing to compile - all classes are up to date
[INFO]
[INFO] --- maven-surefire-plugin:2.10:test (default-test) @ mvn-sample ---
[INFO] Surefire report directory: /Users/dgomezg/Documents/workspace/mvn-sample/target/surefire-reports

-----
T E S T S
-----

Results :

Tests run: 0, Failures: 0, Errors: 0, Skipped: 0

[INFO]
[INFO] --- maven-jar-plugin:2.3.2:jar (default-jar) @ mvn-sample ---
[INFO] Building jar: /Users/dgomezg/Documents/workspace/mvn-sample/target/mvn-sample-1.0.0.jar
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 0.876s
[INFO] Finished at: Wed Nov 20 13:32:06 CET 2013
[INFO] Final Memory: 7M/245M
[INFO] -----
dolamroth:mvn-sample dgomezg$
```

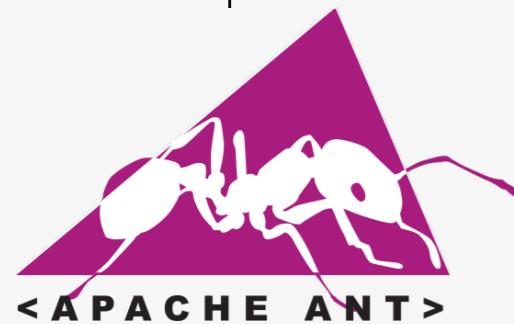
¿Qué se puede mejorar?



- Ficheros de build más concisos
- Convención para estructura
- Flexibilidad para requisitos
- Builds incrementales
- Reutilización de partes del script
- Soporte para implementar lógica

Características gradle

- Flexibilidad
- Definición de nuevas tareas
- Gestor de dependencias (Ivy)
- Convención sobre configuración
- Multi-Módulo
- Plugins
- Gestor de dependencias

**maven**

Características gradle

- Basado en Groovy
 - Lenguaje completo
 - Permite definir lógica más compleja
- Convenciones Flexibles
- Extensibilidad
- Integración con Ant (AntExecutor) y maven



Ejemplo gradle



Ejemplo gradle

```
apply plugin: 'java'

group = 'com.autentia.dgomezg.sandbox'
archivesBaseName = 'gradle-sample'
version = '0.1.0-SNAPSHOT'
```

Ejemplo gradle

```
apply plugin: 'java'

group = 'com.autentia.dgomezg.sandbox'
archivesBaseName = 'gradle-sample'
version = '0.1.0-SNAPSHOT'
```

```
dolamroth:gradle-sample dgomezg$ gradle build
:compileJava UP-TO-DATE
:processResources UP-TO-DATE
:classes UP-TO-DATE
:jar
:assemble
:compileTestJava UP-TO-DATE
:processTestResources UP-TO-DATE
:testClasses UP-TO-DATE
:test
:check
:build

BUILD SUCCESSFUL

Total time: 2.375 secs
dolamroth:gradle-sample dgomezg$
```

Ejemplo gradle

```
apply plugin: 'java'

group = 'com.autentia.dgomezg.sandbox'
archivesBaseName = 'gradle-sample'
version = '0.1.0-SNAPSHOT'

sourceCompatibility = 1.6

jar {
    manifest {
        attributes 'Main-Class': 'com.autentia.dgomezg.sandbox.gradle.HelloWorld'
    }
}
```

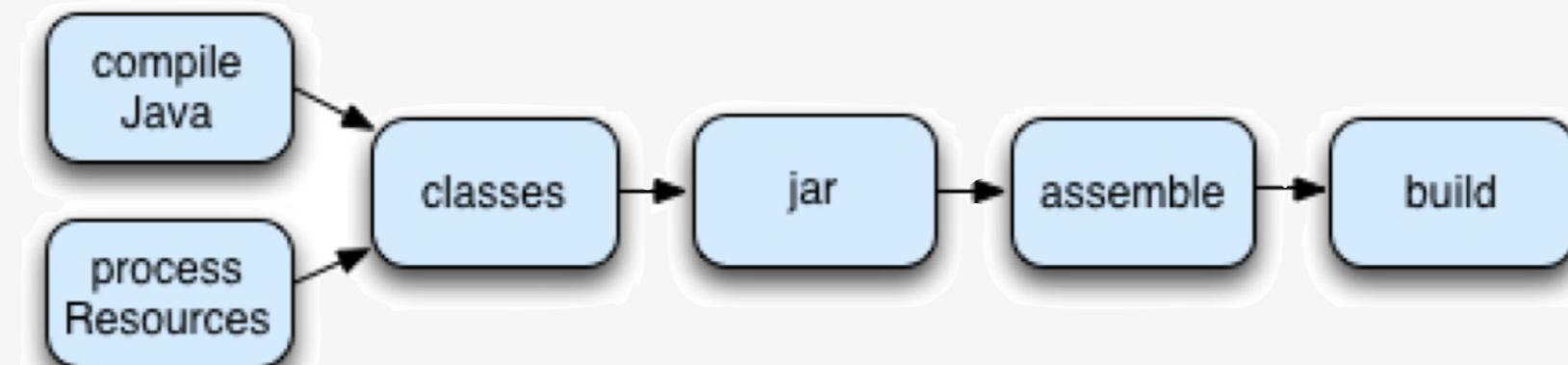
```
.....  
BUILD SUCCESSFUL  
  
Total time: 2.375 secs  
dolamroth:gradle-sample dgomezg$
```

Fundamentos Gradle



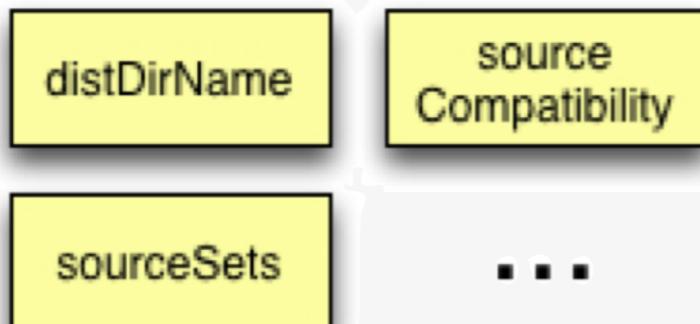
El proceso

Build lifecycle tasks



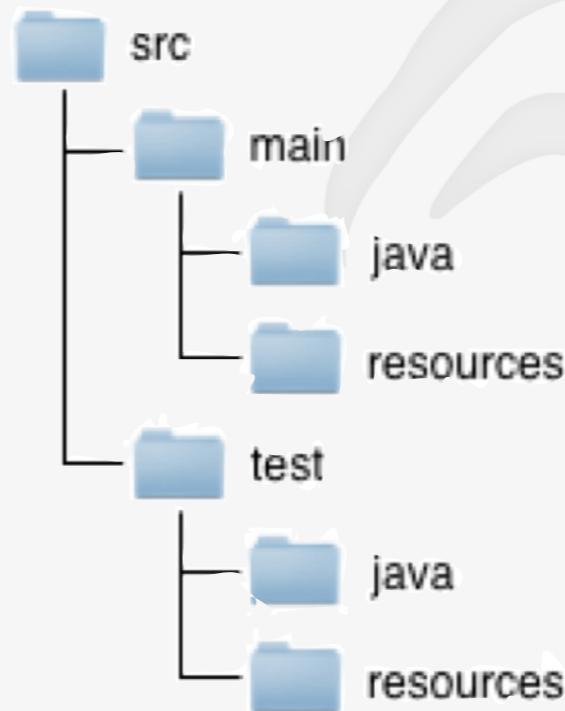
configures

Convention properties



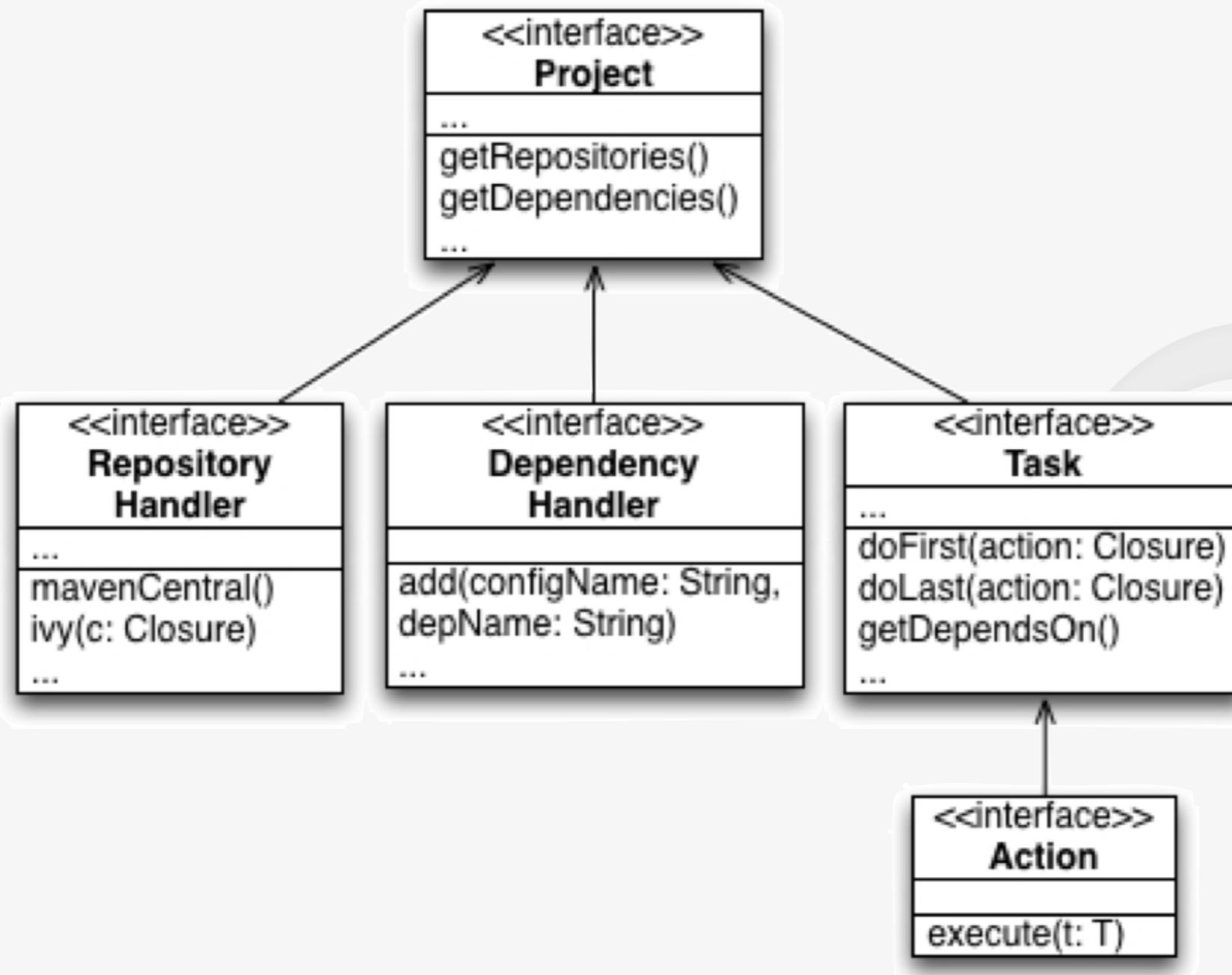
configures

Project layout



Groovy DSL

Todo tiene un equivalente en clases java



Groovy DSL

Mayor flexibilidad a la hora de definir nuevas tareas



Groovy DSL

Mayor flexibilidad a la hora de definir nuevas tareas

```
task projectVersion {  
    doLast {  
        println 'Project version' + project.version  
    }  
}
```



Groovy DSL

Mayor flexibilidad a la hora de definir nuevas tareas

```
task projectVersion {  
    doLast {  
        println 'Project version' + project.version  
    }  
}
```

```
dolamroth:gradle-sample dgomezg$ gradle projectVersion  
:projectVersion  
Project Version: 0.1.0-SNAPSHOT
```

```
BUILD SUCCESSFUL
```

```
Total time: 3.708 secs
```

Groovy DSL

Mayor flexibilidad a la hora de definir nuevas tareas

```
task projectVersion {  
    doLast {  
        println 'Project version' + project.version  
    }  
}
```

```
dolamroth:gradle-sample dgomezg$ gradle projectVersion  
:projectVersion  
Project Version: 0.1.0-SNAPSHOT  
  
BUILD SUCCESSFUL  
  
Total time: 3.708 secs
```

```
dolamroth:gradle-sample dgomezg$ gradle -q projectVersion  
Project Version: 0.1.0-SNAPSHOT  
dolamroth:gradle-sample dgomezg
```

Plugins

Permiten activar funcionalidades



Plugins

Permiten activar funcionalidades

```
apply plugin: 'java'  
apply plugin: 'maven'  
apply plugin: 'eclipse'  
apply plugin: 'war'  
  
group = 'com.autentia.dgomezg.sandbox'  
archivesBaseName = 'gradle-sample'  
version = '0.1.0-SNAPSHOT'  
  
sourceCompatibility = 1.6
```



Plugins

Permiten activar funcionalidades

```
apply plugin: 'java'  
apply plugin: 'maven'  
apply plugin: 'eclipse'  
apply plugin: 'war'      $ gradle tasks  
  
group = 'com.autentia.d...  
archivesBaseName = 'gra...  
version = '0.1.0-SNAPSHOT'  
  
sourceCompatibility = 1  
  
-----  
Build tasks  
-----  
assemble - Assembles the outputs of this project.  
build - Assembles and tests this project.  
buildDependents - Assembles and tests this project and all projects that depend on it.  
buildNeeded - Assembles and tests this project and all projects it depends on.  
clean - Deletes the build directory.  
jar - Assembles a jar archive containing the main classes.  
war - Generates a war archive with all the compiled classes, the web-app content and  
the libraries.  
  
Documentation tasks  
-----  
javadoc - Generates Javadoc API documentation for the main source code.  
  
IDE tasks  
-----  
cleanEclipse - Cleans all Eclipse files.  
eclipse - Generates all Eclipse files.  
  
Other tasks  
-----  
install - Installs the 'archives' artifacts into the local Maven repository.
```

Integración con ant

Propiedad ant disponible en todas las tareas



Integración con ant

Propiedad ant disponible en todas las tareas

```
task projectVersion {  
    doLast {  
        ant.echo(message: 'Project Version: ' + project.version)  
    }  
}
```



Integración con ant

Propiedad ant disponible en todas las tareas

```
task projectVersion {  
    doLast {  
        ant.echo(message: 'Project Version: ' + project.version)  
    }  
}
```

```
dolamroth:gradle-sample dgomezg$ gradle -q projectVersion  
dolamroth:gradle-sample dgomezg$  
dolamroth:gradle-sample dgomezg$ gradle projectVersion  
:projectVersion  
[ant:echo] Project Version: 0.1.0-SNAPSHOT  
  
BUILD SUCCESSFUL  
  
Total time: 2.914 secs  
dolamroth:gradle-sample dgomezg$
```

Integración con maven

Por defecto, toma las convenciones de maven

Se puede integrar con los **repositorios** de maven



Integración con maven

Por defecto, toma las convenciones de maven

Se puede integrar con los **repositorios** de maven

```
repositories {  
    mavenRepo name: 'nexus', url: 'http://terraka:8080/nexus/content/repositories/public'  
    mavenCentral()  
    mavenRepo name: 'spring-milestones', url: 'http://maven.springframework.org/milestone'  
    mavenRepo name: 'java.net', url: 'http://download.java.net/maven/2/'  
}
```

Integración con maven

Por defecto, toma las convenciones de maven

Se puede integrar con los **repositorios** de maven

```
repositories {  
    mavenRepo name: 'nexus', url: 'http://terraka:8080/nexus/content/repositories/public'  
    mavenCentral()  
    mavenRepo name: 'spring-milestones', url: 'http://maven.springframework.org/milestone'  
    mavenRepo name: 'java.net', url: 'http://download.java.net/maven/2/'  
}
```

Dedaración compacta de dependencias

Integración con maven

Por defecto, toma las convenciones de maven

Se puede integrar con los **repositorios** de maven

```
repositories {  
    mavenRepo name: 'nexus', url: 'http://terraka:8080/nexus/content/repositories/public'  
    mavenCentral()  
    mavenRepo name: 'spring-milestones', url: 'http://maven.springframework.org/milestone'  
    mavenRepo name: 'java.net', url: 'http://download.java.net/maven/2/'  
}
```

Declaración compacta de dependencias

```
dependencies {  
    compile "com.google.guava:guava:12.0",  
            "org.apache.commons:commons-lang3:3.1",  
            "org.slf4j:jcl-over-slf4j:1.6.1",  
            "org.springframework.security:spring-security-core:3.0.1.RELEASE",  
            "org.springframework:spring-jdbc:3.0.0.RELEASE",  
            "org.springframework:spring-tx:3.0.0.RELEASE"  
  
    runtime "org.apache.geronimo.specs:geronimo-jpa_3.0_spec:1.1.1",  
            "org.apache.geronimo.specs:geronimo-servlet_2.4_spec:1.1.1"  
  
    testCompile "junit:junit:4.5",  
               "org.springframework:spring-test:3.0.0.RELEASE"  
}
```

Integración con maven

Se pueden excluir dependencias transitivas



Integración con maven

Se pueden excluir dependencias transitivas

```
configurations {  
    sources.exclude group: 'com.google.code.guice'  
    all*.exclude module: 'commons-logging'  
}
```



Integración con maven

Se pueden excluir dependencias transitivas

```
configurations {  
    sources.exclude group: 'com.google.code.guice'  
    all*.exclude module: 'commons-logging'  
}
```

Se pueden subir artefactos al repositorio



Integración con maven

Se pueden excluir dependencias transitivas

```
configurations {  
    sources.exclude group: 'com.google.code.guice'  
    all*.exclude module: 'commons-logging'  
}
```

Se pueden subir artefactos al repositorio

```
apply plugin: 'maven'  
  
uploadArchives {  
  
    repositories.mavenDeployer {  
        def credentials = [username: 'user', password: 'password']  
        uniqueVersion = true  
        snapshotRepository(url: 'http://gitrepo:8081/nexus/content/repositories/snapshots') {  
            authentication(credentials)  
        }  
        repository(url: 'http://gitrepo:8081/nexus/content/repositories/releases') {  
            authentication(credentials)  
        }  
    }  
}
```

Propiedades

Se pueden utilizar propiedades internas



Propiedades

Se pueden utilizar propiedades internas

```
project.ext {  
    springVersion = '3.2.5.RELEASE'  
}  
  
dependencies {  
    compile      "org.springframework.security:spring-security-core:${springVersion}",  
                "org.springframework:spring-jdbc:${springVersion}",  
                "org.springframework:spring-tx:${springVersion}"  
  
    testCompile   "junit:junit:4.5",  
                "org.springframework:spring-test:${springVersion}"  
}
```

Propiedades

Se pueden utilizar propiedades internas

```
project.ext {  
    springVersion = '3.2.5.RELEASE'  
}  
  
dependencies {  
    compile      "org.springframework.security:spring-security-core:${springVersion}",  
                "org.springframework:spring-jdbc:${springVersion}",  
                "org.springframework:spring-tx:${springVersion}"  
  
    testCompile   "junit:junit:4.5",  
                "org.springframework:spring-test:${springVersion}"  
}
```

Se pueden utilizar propiedades externas

Propiedades

Se pueden utilizar propiedades internas

```
project.ext {  
    springVersion = '3.2.5.RELEASE'  
}  
  
dependencies {  
    compile      "org.springframework.security:spring-security-core:${springVersion}",  
                "org.springframework:spring-jdbc:${springVersion}",  
                "org.springframework:spring-tx:${springVersion}"  
  
    testCompile   "junit:junit:4.5",  
                "org.springframework:spring-test:${springVersion}"  
}
```

Se pueden utilizar propiedades externas

```
version=0.1.0-SNAPSHOT  
springVersion=3.2.5.RELEASE  
sharedLocation=https://raw.github.com/autentia/common/
```

Propiedades

Se pueden utilizar propiedades internas

```
project.ext {  
    springVersion = '3.2.5.RELEASE'  
}  
  
dependencies {  
    compile      "org.springframework.security:spring-security-core:${springVersion}",  
                "org.springframework:spring-jdbc:${springVersion}",  
                "org.springframework:spring-tx:${springVersion}"  
  
    testCompile   "junit:junit:4.5",  
                "org.springframework:spring-test:${springVersion}"  
}
```

Se pueden utilizar propiedades externas

```
version=0.1.0-SNAPSHOT  
springVersion=3.2.5.RELEASE  
sharedLocation=https://raw.github.com/autentia/common/
```

gradle.properties

Propiedades

Se pueden utilizar propiedades internas

```
project.ext {  
    springVersion = '3.2.5.RELEASE'  
}  
  
dependencies {  
    compile      "org.springframework.security:spring-security-core:${springVersion}",  
                "org.springframework:spring-jdbc:${springVersion}",  
                "org.springframework:spring-tx:${springVersion}"  
  
    testCompile   "junit:junit:4.5",  
                "org.springframework:spring-test:${springVersion}"  
}
```

Se pueden utilizar propiedades externas

```
version=0.1.0-SNAPSHOT  
springVersion=3.2.5.RELEASE  
sharedLocation=https://raw.github.com/autentia/common/
```

gradle.properties

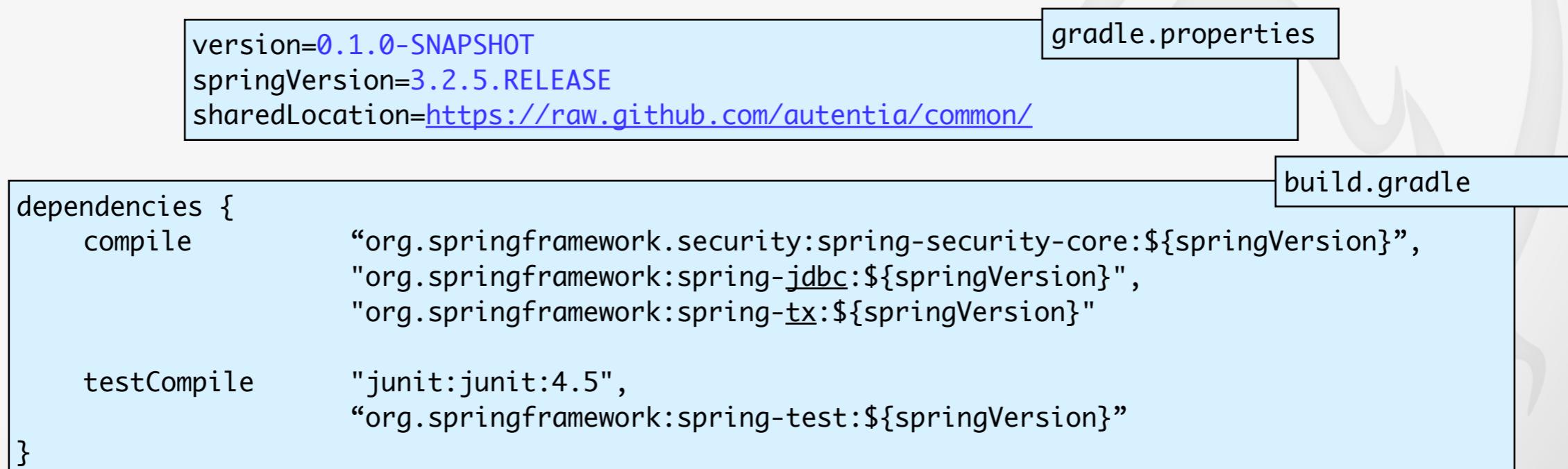
```
dependencies {  
    compile      "org.springframework.security:spring-security-core:${springVersion}",  
                "org.springframework:spring-jdbc:${springVersion}",  
                "org.springframework:spring-tx:${springVersion}"  
  
    testCompile   "junit:junit:4.5",  
                "org.springframework:spring-test:${springVersion}"  
}
```

Propiedades

Se pueden utilizar propiedades internas

```
project.ext {  
    springVersion = '3.2.5.RELEASE'  
}  
  
dependencies {  
    compile      "org.springframework.security:spring-security-core:${springVersion}",  
                "org.springframework:spring-jdbc:${springVersion}",  
                "org.springframework:spring-tx:${springVersion}"  
  
    testCompile   "junit:junit:4.5",  
                "org.springframework:spring-test:${springVersion}"  
}
```

Se pueden utilizar propiedades externas



Gradle Wrapper



Gradle Wrapper



Gradle Wrapper

```
task wrapper(type: Wrapper) {  
    gradleVersion = '1.7'  
}
```



Gradle Wrapper

```
task wrapper(type: Wrapper) {  
    gradleVersion = '1.7'  
}
```



```
$ gradle wrapper  
:wrapper  
  
BUILD SUCCESSFUL  
  
Total time: 5.071 secs  
  
$ ls -l gradle/wrapper  
total 112  
-rw-r--r-- 1 dgomezg staff 49875 Nov 20 16:53 gradle-wrapper.jar  
-rw-r--r-- 1 dgomezg staff     229 Nov 20 16:53 gradle-wrapper.properties  
$ ls gradle*  
gradle.properties      gradlew           gradlew.bat  
$ ./gradlew build  
:compileJava UP-TO-DATE  
:processResources UP-TO-DATE  
:classes UP-TO-DATE  
:war UP-TO-DATE  
:assemble UP-TO-DATE  
:compileTestJava UP-TO-DATE  
:processTestResources UP-TO-DATE  
:testClasses UP-TO-DATE  
:test UP-TO-DATE  
:check UP-TO-DATE  
:build UP-TO-DATE  
  
BUILD SUCCESSFUL  
  
Total time: 3.531 secs
```

Convención flexible

Se pueden adaptar las convenciones de estructura de proyecto



Convención flexible

Se pueden adaptar las convenciones de estructura de proyecto

```
sourceSets {  
    main {  
        java {  
            srcDirs = ['src']  
        }  
    }  
    test {  
        java {  
            srcDirs = ['test']  
        }  
    }  
}  
buildDir = 'out'
```



Convención flexible

Se pueden adaptar las convenciones de estructura de proyecto

```
sourceSets {  
    main {  
        java {  
            srcDirs = ['src']  
        }  
    }  
    test {  
        java {  
            srcDirs = ['test']  
        }  
    }  
}  
buildDir = 'out'
```

```
compileJava {  
    source 'src/protobuf/java'  
    source 'src/main/java'  
}
```

Reutilización

Los archivos se pueden modularizar



Reutilización

Los archivos se pueden modularizar

```
apply plugin: 'java'  
apply plugin: 'maven'  
apply plugin: 'eclipse'  
  
group = 'com.autentia.dgomezg.sandbox'  
archivesBaseName = 'gradle-sample'  
  
sourceCompatibility = 1.6  
  
repositories {  
    //mavenRepo name: 'nexus', url: 'http://terraka:8080/nexus/content/repositories/public'  
    mavenCentral()  
    mavenRepo name: 'spring-milestones', url: 'http://maven.springframework.org/milestone'  
    mavenRepo name: 'java.net', url: 'http://download.java.net/maven/2/'  
}  
  
dependencies {  
    compile "org.springframework.security:spring-security-core:3.0.1.RELEASE",  
            "org.springframework:spring-jdbc:3.0.0.RELEASE",  
            "org.springframework:spring-tx:3.0.0.RELEASE"  
    testCompile "junit:junit:4.5",  
               "org.springframework:spring-test:3.0.0.RELEASE"  
}
```

Reutilización

Los archivos se pueden modularizar

```
apply plugin: 'java'  
apply plugin: 'maven'  
apply plugin: 'eclipse'  
  
group = 'com.autentia.dgomezg.sandbox'  
archivesBaseName = 'gradle-sample'  
  
sourceCompatibility = 1.6  
  
repositories {  
    //mavenRepo name: 'nexus', url: 'http://terraka:8080/nexus/content/repositories/public'  
    mavenCentral()  
    mavenRepo name: 'spring-milestones', url: 'http://maven.springframework.org/milestone'  
    mavenRepo name: 'java.net', url: 'http://download.java.net/maven/2/'  
}  
  
dependencies {  
    compile "org.springframework.security:spring-security-core:3.0.1.RELEASE",  
            "org.springframework:spring-jdbc:3.0.0.RELEASE",  
            "org.springframework:spring-tx:3.0.0.RELEASE"  
    testCompile "junit:junit:4.5",  
               "org.springframework:spring-test:3.0.0.RELEASE"  
}
```

project.gradle

Reutilización

Los archivos se pueden importar



Reutilización

Los archivos se pueden importar

```
apply plugin: 'java'  
apply plugin: 'maven'  
apply plugin: 'eclipse'  
  
group = 'com.autentia.dgomezg.sandbox'  
archivesBaseName = 'gradle-sample'  
  
sourceCompatibility = 1.6  
  
repositories {  
}  
  
dependencies {  
}
```

Reutilización

Los archivos se pueden importar

```
apply plugin: 'java'  
apply plugin: 'maven'  
apply plugin: 'eclipse'  
  
group = 'com.autentia.dgomezg.sandbox'  
archivesBaseName = 'gradle-sample'  
  
sourceCompatibility = 1.6  
  
repositories {  
}  
  
dependencies {  
}
```

project.gradle

Reutilización

Los archivos se pueden importar

```
apply plugin: 'java'  
apply plugin: 'maven'  
apply plugin: 'eclipse'  
  
group = 'com.autentia.dgomezg.sandbox'  
archivesBaseName = 'gradle-sample'  
  
sourceCompatibility = 1.6  
  
repositories {  
    apply from: '../autentia-common/project.gradle'  
  
    dependencies {  
        compile "com.google.guava:guava:r05",  
        "org.apache.commons:commons-lang3:3.1",  
        "org.slf4j:slf4j-api:1.6.1",  
        "ch.qos.logback:logback-classic:0.9.24",  
        "org.springframework:spring-aop:$springVersion",  
        "org.springframework:spring-beans:$springVersion",  
        "org.springframework:spring-context:$springVersion",  
        "org.springframework:spring-core:$springVersion",  
        "org.springframework:spring-web:$springVersion",  
        "org.springframework.data:spring-data-mongodb:1.0.1.RELEASE",  
        "org.mongodb:mongo-java-driver:2.7.1"  
  
        testCompile "org.mockito:mockito-core:1.8.5"  
    }  
}
```

project.gradle

Reutilización

Los archivos se pueden importar

```
apply plugin: 'java'  
apply plugin: 'maven'  
apply plugin: 'eclipse'  
  
group = 'com.autentia.dgomezg.sandbox'  
archivesBaseName = 'gradle-sample'  
  
sourceCompatibility = 1.6
```

project.gradle

```
repositories {  
    apply from: '../autentia-common/project.gradle'  
}  
  
dependencies {  
    compile "com.google.guava:guava:r05",  
    "org.apache.commons:commons-lang3:3.1",  
    "org.slf4j:slf4j-api:1.6.1",  
    "ch.qos.logback:logback-classic:0.9.24",  
    "org.springframework:spring-aop:$springVersion",  
    "org.springframework:spring-beans:$springVersion",  
    "org.springframework:spring-context:$springVersion",  
    "org.springframework:spring-core:$springVersion",  
    "org.springframework:spring-web:$springVersion",  
    "org.springframework.data:spring-data-mongodb:1.0.1.RELEASE",  
    "org.mongodb:mongo-java-driver:2.7.1"  
  
    testCompile "org.mockito:mockito-core:1.8.5"  
}
```

build.gradle

Reutilización

Los archivos se pueden **compartir**



Reutilización

Los archivos se pueden **compartir**

```
version=0.1.0-SNAPSHOT
springVersion=3.2.5.RELEASE
sharedLocation=https://raw.github.com/autentia/common/
#commitId of the version to be used
commonComponentsVersion=c7e2110511112c5055e0aaaa7db953b35129a95b
```



Reutilización

Los archivos se pueden **compartir**

```
version=0.1.0-SNAPSHOT
springVersion=3.2.5.RELEASE
sharedLocation=https://raw.github.com/autentia/common/
#commitId of the version to be used
commonComponentsVersion=c7e2110511112c5055e0aaaa7db953b35129a95b
```

gradle.properties



Reutilización

Los archivos se pueden **compartir**

```
version=0.1.0-SNAPSHOT  
springVersion=3.2.5.RELEASE  
sharedLocation=https://raw.github.com/autentia/common/  
#commitId of the version to be used  
commonComponentsVersion=c7e2110511112c5055e0aaaa7db953b35129a95b
```

gradle.properties

```
apply from: '$sharedLocation/$commonComponentsVersion/autentia.gradle'

dependencies {
    compile
        "com.google.guava:guava:r05",
        "org.apache.commons:commons-lang3:3.1",
        "org.slf4j:slf4j-api:1.6.1",
        "ch.qos.logback:logback-classic:0.9.24",
        "org.springframework:spring-aop:$springVersion",
        "org.springframework:spring-beans:$springVersion",
        "org.springframework:spring-context:$springVersion",
        "org.springframework:spring-core:$springVersion",
        "org.springframework:spring-web:$springVersion",
        "org.springframework.data:spring-data-mongodb:1.0.1.RELEASE",
        "org.mongodb:mongo-java-driver:2.7.1"

    testCompile
}
```

Reutilización

Los archivos se pueden **compartir**

```
version=0.1.0-SNAPSHOT  
springVersion=3.2.5.RELEASE  
sharedLocation=https://raw.github.com/autentia/common/  
#commitId of the version to be used  
commonComponentsVersion=c7e2110511112c5055e0aaaa7db953b35129a95b
```

gradle.properties

```
apply from: '$sharedLocation/$commonComponentsVersion/autentia.gradle'  
  
dependencies {  
    compile "com.google.guava:guava:r05",  
            "org.apache.commons:commons-lang3:3.1",  
            "org.slf4j:slf4j-api:1.6.1",  
            "ch.qos.logback:logback-classic:0.9.24",  
            "org.springframework:spring-aop:$springVersion",  
            "org.springframework:spring-beans:$springVersion",  
            "org.springframework:spring-context:$springVersion",  
            "org.springframework:spring-core:$springVersion",  
            "org.springframework:spring-web:$springVersion",  
            "org.springframework.data:spring-data-mongodb:1.0.1.RELEASE",  
            "org.mongodb:mongo-java-driver:2.7.1"  
  
    testCompile "org.mockito:mockito-core:1.8.5"  
}
```

build.gradle

Extensibilidad

Podemos ampliar la funcionalidad fácilmente

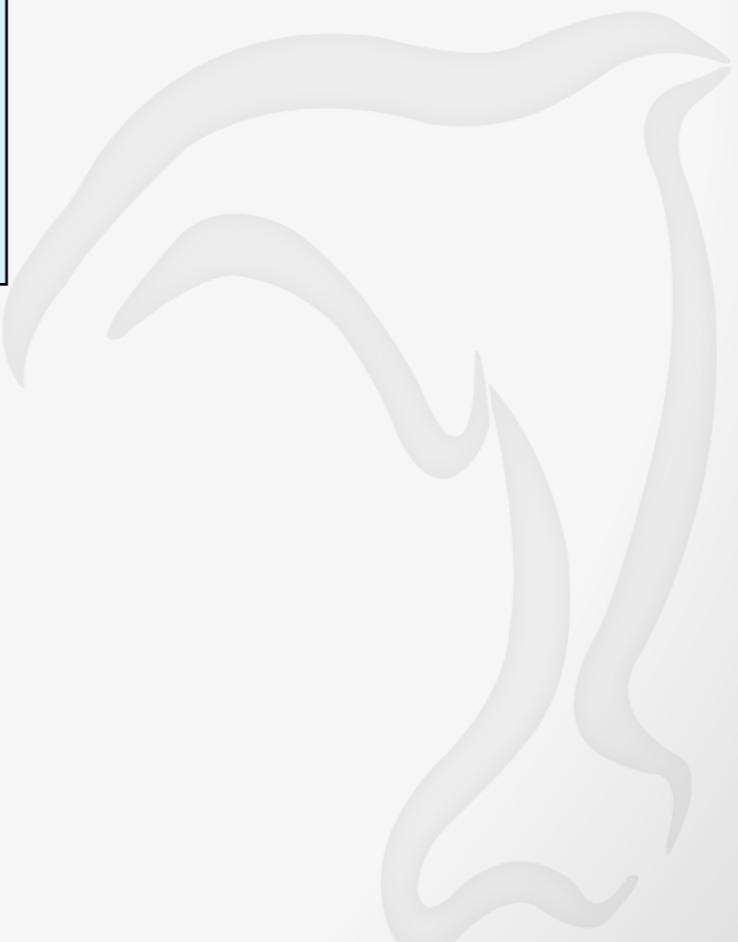


Extensibilidad

Podemos ampliar la funcionalidad fácilmente

```
task deployJetty(dependsOn: assertJettyHome) << {
    new File("$jettyHomecontexts/${webappName}.xml").write (
"""
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE Configure PUBLIC "-//Mort Bay Consulting//DTD Configure//EN"
"http://jetty.mortbay.org/configure.dtd">
<Configure class="org.mortbay.jetty.webapp.WebAppContext">
    <Set name="contextPath"/>/${webappName}</Set>
    <Set name="resourceBase">${projectDir}/src/main/webapp</Set>
</Configure>
""")
}

task undeployJetty(dependsOn: assertJettyHome) << {
    ant.delete file: "$jettyHomecontexts/${webappName}.xml"
}
```



Extensibilidad

Podemos ampliar la funcionalidad fácilmente

jetty.gradle

```
task deployJetty(dependsOn: assertJettyHome) << {
    new File("$jettyHomecontexts/${webappName}.xml").write (
"""
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE Configure PUBLIC "-//Mort Bay Consulting//DTD Configure//EN"
"http://jetty.mortbay.org/configure.dtd">
<Configure class="org.mortbay.jetty.webapp.WebAppContext">
    <Set name="contextPath"/>/${webappName}</Set>
    <Set name="resourceBase">${projectDir}/src/main/webapp</Set>
</Configure>
""")
}

task undeployJetty(dependsOn: assertJettyHome) << {
    ant.delete file: "$jettyHomecontexts/${webappName}.xml"
}
```

Extensibilidad

Podemos ampliar la funcionalidad fácilmente

jetty.gradle

```
task deployJetty(dependsOn: assertJettyHome) << {
    new File("$jettyHomecontexts/${webappName}.xml").write (
"""
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE Configure PUBLIC "-//Mort Bay Consulting//DTD Configure//EN"
"http://jetty.mortbay.org/configure.dtd">
<Configure class="org.mortbay.jetty.webapp.WebAppContext">
    <Set name="contextPath"/>/${webappName}</Set>
    <Set name="resourceBase">${projectDir}/src/main/webapp</Set>
</Configure>
""")
}

task undeployJetty(dependsOn: assertJettyHome) << {
    ant.delete file: "$jettyHomecontexts/${webappName}.xml"
}
```

```
apply from: '$sharedLocation/$commonComponentsVersion/jetty.gradle'

project.ext {
    webappName=gradle-sample
    jettyHome= System.getenv("JETTY_HOME");
}
...
```

Extensibilidad

Podemos ampliar la funcionalidad fácilmente

jetty.gradle

```
task deployJetty(dependsOn: assertJettyHome) << {
    new File("$jettyHomecontexts/${webappName}.xml").write (
"""
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE Configure PUBLIC "-//Mort Bay Consulting//DTD Configure//EN"
"http://jetty.mortbay.org/configure.dtd">
<Configure class="org.mortbay.jetty.webapp.WebAppContext">
    <Set name="contextPath">/${webappName}</Set>
    <Set name="resourceBase">${projectDir}/src/main/webapp</Set>
</Configure>
""")
}

task undeployJetty(dependsOn: assertJettyHome) << {
    ant.delete file: "$jettyHomecontexts/${webappName}.xml"
}
```

build.gradle

```
apply from: '$sharedLocation/$commonComponentsVersion/jetty.gradle'

project.ext {
    webappName=gradle-sample
    jettyHome= System.getenv("JETTY_HOME");
}
...
```

'Arquetipos'

Hay un proyecto de plugin que procura cubrir ese hueco:

<https://github.com/townsfolk/gradle-templates>



'Arquetipos'

Hay un proyecto de plugin que procura cubrir ese hueco:

<https://github.com/townsfolk/gradle-templates>

```
apply from: 'http://www.tellurianring.com/projects/gradle-plugins/gradle-templates/1.3/apply.groovy'
```

```
...
```

'Arquetipos'

Hay un proyecto de plugin que procura cubrir ese hueco:

<https://github.com/townsfolk/gradle-templates>

```
apply from: 'http://www.tellurianring.com/projects/gradle-plugins/gradle-templates/1.3/apply.groovy'
```

```
...
```

```
$ gradle tasks
```

```
Template tasks
```

```
-----  
createGradlePlugin - Creates a new Gradle Plugin project in a new directory named after your project.  
createGroovyClass - Creates a new Groovy class in the current project.  
createGroovyProject - Creates a new Gradle Groovy project in a new directory named after your project.  
createJavaClass - Creates a new Java class in the current project.  
createJavaProject - Creates a new Gradle Java project in a new directory named after your project.  
createScalaClass - Creates a new Scala class in the current project.  
createScalaObject - Creates a new Scala object in the current project.  
createScalaProject - Creates a new Gradle Scala project in a new directory named after your project.  
createWebappProject - Creates a new Gradle Webapp project in a new directory named after your project.  
exportAllTemplates - Exports all the default template files into the current directory.  
exportGroovyTemplates - Exports the default groovy template files into the current directory.  
exportJavaTemplates - Exports the default java template files into the current directory.  
exportPluginTemplates - Exports the default plugin template files into the current directory.  
exportScalaTemplates - Exports the default scala template files into the current directory.  
exportWebappTemplates - Exports the default webapp template files into the current directory.  
initGradlePlugin - Initializes a new Gradle Plugin project in the current directory.  
initGroovyProject - Initializes a new Gradle Groovy project in the current directory.  
initJavaProject - Initializes a new Gradle Java project in the current directory.  
initScalaProject - Initializes a new Gradle Scala project in the current directory.  
initWebappProject - Initializes a new Gradle Webapp project in the current directory.
```

Y mucho más

- Posibilidad de ejecutar gradle como **demonio**
- Toda la potencia de un **lenguaje para los scripts**
- Integración con sistemas de Integración Continua
 - Jenkins
 - ClinkerHQ
 - Consola



Conclusiones

- No más XML Oriented Programming
- Concisión y legibilidad
- Toda la potencia de las **convenciones** de maven
- Integración con **Ant**
- Gestor de **dependencias**
- Flexibilidad de un **lenguaje de programación**
- Modularización y reutilización

