



BreizhJug

3 novembre 2016

Build d'une WebApp « moderne » avec



Laurent HUET

 @lhuet35



Application « Fullstack »

- Backend Java
 - Spring Boot
 - Whatever ...
- Frontend
 - Node / Npm / Bower
 - Gulp / Grunt ... et leurs myriades de plugins
 - Les challengers de Gulp / Grunt ...

Challenges

- Mixer les outils du monde Java et du monde Web (JS)
- Habitudes des développeurs
- Reproductibilité
 - Versions de node / npm / bower / gulp / ...

Gradle : 2 stratégies possibles

- Full Java → Gradle only
 - WebJars
 - Plugins JS / CSS / ... disponibles
- Gradle en « master »
 - Utilisation des outils de build JS
 - Piloter le build JS avec Gradle

1^{ère} Stratégie : Full Java - Gradle Only

Gestion des dépendances avec WebJars

- <http://www.webjars.org/>
 - Dépendances Front packagées dans des Jar

```
...  
dependencies {  
    webjars 'org.webjars:jquery:2.1.3'  
    webjars 'org.webjars:bootstrap:3.3.4'  
    webjars 'org.webjars:highlightjs:8.4-4'  
    webjars 'org.webjars:font-awesome:4.3.0-1'  
    webjars 'org.webjars:jquery.tocify.js:1.9.0'  
    webjars 'org.webjars:Magnific-Popup:1.0.0'  
}  
...
```

- Extraction possible des ressources
 - <http://aruizca.com/extract-webjars-static-resources-with-gradle/>

Plugin JS - <http://eriwen.github.io/gradle-js-plugin/>

- Gradle tasks
 - combineJs
 - minifyJs (Google Closure Compiler)
 - gzipJs
 - jshint
 - props2js
 - requireJs

```
plugins {  
    id "com.eriwen.gradle.js" version "2.14.1"  
}
```

Plugin JS

```
// Declare your sources
javascript.source {
  dev {
    js {
      srcDir jsSrcDir
      include "*.js"
      exclude "*.min.js"
    }
  }
  prod {
    js {
      srcDir jsSrcDir
      include "*.min.js"
    }
  }
}
```

```
// Configure the built-in task
combineJs {
  encoding = "UTF-8"
  source = javascript.source.dev.js.files
  dest = file("${buildDir}/all.js")
}
```

```
minifyJs {
  source = combineJs
  dest = file("${buildDir}/all-min.js")
  sourceMap = file("${buildDir}/all.sourcemap.json")
  closure {
    warningLevel = 'QUIET'
  }
}
```

```
jshint {
  source = javascript.source.dev.js.files
  dest = file("${buildDir}/jshint.out")
  reporter = 'checkstyle'
  jshint.options = [expr: "true", unused: "true"]
}
```


Plugin CSS - <http://eriwen.github.io/gradle-css-plugin/>

- Gradle tasks
 - combineCSS
 - minifyCSS (YUI compressor)
 - gzipCSS
 - less
 - csslint

```
plugins {  
    id "com.eriwen.gradle.css" version "2.14.0"  
}
```

Plugin CSS

```
// Declare your sources
css.source {
  dev {
    css {
      srcDir "app/styles"
      include "*.css"
      exclude "*.min.css"
    }
  }
}
```

```
combineCss {
  source = css.source.dev.css.files
  dest = "${buildDir}/all.css"
}

minifyCss {
  source = combineCss
  dest = "${buildDir}/all-min.css"
  yuicompressor { // optional
    lineBreakPos = -1
  }
}

gzipCss {
  source = minifyCss
  dest = "${buildDir}/all.2.0.4.css"
}
```

2^{ème} Stratégie : Gradle en « master »

Outils de build JS

- Node / Npm / Bower
 - Dépendances dans package.json / bower.json
- Gulp / Grunt / ...
 - Build avec outils du monde JS

Plugin node - <https://github.com/srs/gradle-node-plugin>

- Download avec le build
- Versions de node/npm fixées
- Gradle tasks
 - npmInstall
 - npm_<npmCommand>
 - NodeTask / NpmTask

```
plugins {  
    id "com.moowork.node" version "0.13"  
}  
  
node {  
    version = '4.6.0'  
    npmVersion = '2.15.9'  
    download = true  
    // Set the work directory for unpacking node  
    workDir = file("${project.buildDir}/nodejs")  
    // Set the work directory where node_modules should be located  
    nodeModulesDir = file("${project.projectDir}")  
}
```

Plugin node – Bower et Gulp

```
task bowerInstall(type: NodeTask, dependsOn: 'npm_install') {  
    script = file('node_modules/bower/bin/bower')  
    def bowerComponentsDir = "app/bower_components"  
    args = ["--config.directory=${bowerComponentsDir}",  
           "--config.storage.cache=${gradle.getGradleUserHomeDir()}/caches/bower/cache",  
           "--config.storage.packages=${gradle.getGradleUserHomeDir()}/caches/bower/packages",  
           "--config.storage.registry=${gradle.getGradleUserHomeDir()}/caches/bower/registry",  
           "install"]  
    workingDir = file("${project.projectDir}")  
    inputs.files file('bower.json')  
    outputs.files file("${bowerComponentsDir}")  
}
```

```
task gulpDist(type: NodeTask, dependsOn: 'bowerInstall') {  
    script = file('node_modules/gulp/bin/gulp.js')  
    args = ["build"]  
    workingDir = file("${project.projectDir}")  
    inputs.files fileTree('app')  
    outputs.files fileTree('dist')  
}
```

Pour aller plus loin

- Ressources

- <https://github.com/srs/gradle-node-plugin>
- <https://shinesolutions.com/2014/03/19/javascript-webapps-with-gradle/>
- <https://objectpartners.com/2015/04/29/using-gradle-and-bower-to-manage-jscss-dependencies/>

- Exemple d'application : cfp.io

- <https://github.com/lhuet/front-legacy/tree/buildGradle>
- <https://github.com/lhuet/callForPapers/tree/buildGradle>
- <https://github.com/lhuet/auth/tree/buildGradle>