The Java Build Tool Roundup

Tim Hegemann

2018-12-12

Outline

Introduction

Hello World

More Sophisticated Examples

Live Demo?

Introduction

Why Java Build Automation Software?

- Dependency resolution
- Resource/classpath management
- Automatic source generation
- Compiler invocation
- Unit/integration tests execution
- Packaging/verification
- Deployment/publishing
- Multi-project handling
- ...

Apache Maven

- Release: 2004
- Written in: Java
- Configured in: XML
- Licence: Apache 2.0

Apache Maven

- Release: 2004
- Written in: Java
- Configured in: XML
- Licence: Apache 2.0

Pros	Cons
• widely used (75%)	• XML
• many plugins / archetypes	• build config is very verbose
	• obscure error reporting
	• unintuitive

Gradle

- Release: 2007
- Written in: Java, Groovy, Kotlin
- Configured in: Groovy or Kotlin flavoured DSL
- Licence: Apache 2.0

Gradle

- Release: 2007
- Written in: Java, Groovy, Kotlin
- Configured in: Groovy or Kotlin flavoured DSL
- Licence: Apache 2.0

Pros	Cons
• default for Android Apps	• conflicting plugins
• highly configurable	• breaking changes
• many plugins	• bad build script error checking
• very fast	

sbt

- Release: 2008
- Written in: Scala
- Configured in: sbt (Scala DSL) or plain Scala
- Licence: Apache 2.0 (parts under BSD-style licences)

sbt

- Release: 2008
- Written in: Scala
- Configured in: sbt (Scala DSL) or plain Scala
- Licence: Apache 2.0 (parts under BSD-style licences)

Pros	Cons
• excellent Scala support	• slow
• concise build definitions	• complex and inconsistent
• mighty and flexible	
• typechecked build scripts	

Mill

- Release: 2018
- Written in: Scala
- Configured in: Scala
- Licence: MIT Licence

Mill

- Release: 2018
- Written in: Scala
- Configured in: Scala
- Licence: MIT Licence

Pros	Cons
• simple and fast	• missing features/doc
• machine readable json output	• uncommon directory structure
• build-ins for package/publish/run	• missing IDE/CI/... support
• typechecked build scripts	

Hello World

Task: Compile and Run a Hello World Program

Take the following hello world example

```
public class HelloWorld {
    public static void main(String[] args) {
        System.out.println("Hello World!");
    }
}
```

Perform two simple tasks

1. Compile the code
2. Run the code

The shortest build file wins ;)

Project Structure

Maven

```
.
├── pom.xml
├── src
│   ├── main
│   │   └── java
│   │       └── HelloWorld.java
│   └── test
│       └── java
│           └── <SomeTest>.java
└── target
    ├── classes
    │   └── HelloWorld.class
    ├── hello-world-1.3.3.7.jar
   [...]
```

Gradle

```
.
├── build
│   ├── classes
│   │   └── java
│   │       └── main
│   │           └── HelloWorld.class
│   ├── libs
│   │   └── hello-world.jar
│  [...]
├── build.gradle
├── settings.gradle
└── src
    └── main
        └── java
            └── HelloWorld.java
```

Project Structure

sbt

```
.
├── build.sbt
├── src
│   ├── main
│   │   └── java
│   │       └── HelloWorld.java
│   └── test
│       └── java
│           └── <SomeTest>.java
├── target
│   ├── scala-2.12
│   │   ├── classes
│   │   │   └── HelloWorld.class
│   │   └── hello_2.12-0.1.0-SNAPSHOT.jar
│  [...]
└── project
    ├── build.properties
    ├── plugins.sbt
   [...]
```

Mill

```
.
├── build.sc
├── hello
│   ├── src
│   │   └── HelloWorld.java
│   └── test
│       └── <SomeTest>.java
└── out
    ├── hello
    │   ├── [...]
    │   ├── compile
    │   │   ├── dest
    │   │   │   ├── classes
    │   │   │   │   └── HelloWorld.class
    │   │   │   └── zinc
    │   │   ├── log
    │   │   └── meta.json
   [...]
```

Maven Build Definition

```xml
<project xmlns="http://maven.apache.org/POM/4.0.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
  ↪ http://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>

  <groupId>de.uniwue.okinf</groupId>
  <artifactId>hello-world</artifactId>
  <version>1.3.3.7</version>

  <properties>
    <maven.compiler.source>11</maven.compiler.source>
    <maven.compiler.target>11</maven.compiler.target>
    <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
  </properties>

```

...

Maven Build Definition

...

```
  <build><plugins>
      <plugin>
        <groupId>org.codehaus.mojo</groupId>
        <artifactId>exec-maven-plugin</artifactId>
        <version>1.6.0</version>
        <executions><execution>
            <phase>package</phase>
            <goals>
              <goal>java</goal>
            </goals>
        </execution></executions>
        <configuration>
          <mainClass>HelloWorld</mainClass>
        </configuration>
      </plugin>
  </plugins></build>
</project>
```

Maven Output

```
# tim @ x260 in ~/uni/okinf/java-build-tool-roundup/playground/helloworld/maven [0:54:10]
$ mvn package
[INFO] Scanning for projects...
[INFO]
[INFO] -------------------< de.uniwue.okinf:hello-world >--------------------
[INFO] Building hello-world 1.3.3.7
[INFO] --------------------------------[ jar ]---------------------------------
[INFO]
[INFO] --- maven-resources-plugin:2.6:resources (default-resources) @ hello-world ---
[INFO] Using 'UTF-8' encoding to copy filtered resources.
[INFO] skip non existing resourceDirectory /home/tim/uni/okinf/java-build-tool-roundup/playground/helloworld/maven/src/main/resources
[INFO]
[INFO] --- maven-compiler-plugin:3.1:compile (default-compile) @ hello-world ---
[INFO] Changes detected - recompiling the module!
[INFO] Compiling 1 source file to /home/tim/uni/okinf/java-build-tool-roundup/playground/helloworld/maven/target/classes
[INFO]
[INFO] --- maven-resources-plugin:2.6:testResources (default-testResources) @ hello-world ---
[INFO] Using 'UTF-8' encoding to copy filtered resources.
[INFO] skip non existing resourceDirectory /home/tim/uni/okinf/java-build-tool-roundup/playground/helloworld/maven/src/test/resources
[INFO]
[INFO] --- maven-compiler-plugin:3.1:testCompile (default-testCompile) @ hello-world ---
[INFO] No sources to compile
[INFO]
[INFO] --- maven-surefire-plugin:2.12.4:test (default-test) @ hello-world ---
[INFO] No tests to run.
[INFO]
[INFO] --- maven-jar-plugin:2.4:jar (default-jar) @ hello-world ---
[INFO] Building jar: /home/tim/uni/okinf/java-build-tool-roundup/playground/helloworld/maven/target/hello-world-1.3.3.7.jar
[INFO]
[INFO] --- exec-maven-plugin:1.6.0:java (default) @ hello-world ---
Hello World!
[INFO] ------------------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------------------
[INFO] Total time:  1.966 s
[INFO] Finished at: 2018-12-12T00:54:33+01:00
[INFO] ------------------------------------------------------------------------
```

Gradle Build Definition

```
1  plugins {
2      id 'java'
3      id 'application'
4  }
5
6  mainClassName = 'HelloWorld'
```

Gradle Build Definition & Output

```
1  plugins {
2      id 'java'
3      id 'application'
4  }
5
6  mainClassName = 'HelloWorld'
```

```
# tim @ x260 in ~/uni/okinf/java-build-tool-roundup/playground/helloworld/gradle [1:26:03]
$ gradle run
> Task :compileJava
> Task :processResources NO-SOURCE
> Task :classes

> Task :run
Hello World!

BUILD SUCCESSFUL in 1s
2 actionable tasks: 2 executed
```

Mill Build Definition

```
1 import mill._, scalalib._
2
3 object hello extends JavaModule
```

Mill Build Definition & Output

```
1  import mill._, scalalib._
2
3  object hello extends JavaModule
```

```
# tim @ x260 in ~/uni/okinf/java-build-tool-roundup/playground/helloworld/mill [1:31:11]
$ mill hello.run
[19/29] hello.compile
[info] Compiling 1 Java source to /home/tim/uni/okinf/java-build-tool-roundup/playground/helloworld/mill/out/hello/compile/dest/classes ...
[info] Done compiling.
[29/29] hello.run
Hello World!
```

sbt Build Definition

sbt Build Definition & Output

```
# tim @ x260 in ~/uni/okinf/java-build-tool-roundup/playground/helloworld/sbt [1:28:13]
$ sbt run
[warn] No sbt.version set in project/build.properties, base directory: /home/tim/uni/okinf/java-build-tool-roundup/playground/helloworld/sbt
[info] Loading settings for project global-plugins from idea.sbt ...
[info] Loading global plugins from /home/tim/.sbt/1.0/plugins
[info] Set current project to sbt (in build file:/home/tim/uni/okinf/java-build-tool-roundup/playground/helloworld/sbt/)
[info] Updating ...
[info] Done updating.
[info] Compiling 1 Java source to /home/tim/uni/okinf/java-build-tool-roundup/playground/helloworld/sbt/target/scala-2.12/classes ...
[info] Done compiling.
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by com.google.protobuf.UnsafeUtil (file:/home/tim/.sbt/boot/scala-2.12.7/org.scala-sbt/sbt/1.2.6/protobuf-java
-3.3.1.jar) to field java.nio.Buffer.address
WARNING: Please consider reporting this to the maintainers of com.google.protobuf.UnsafeUtil
WARNING: Use --illegal-access=warn to enable warnings of further illegal reflective access operations
WARNING: All illegal access operations will be denied in a future release
[info] Packaging /home/tim/uni/okinf/java-build-tool-roundup/playground/helloworld/sbt/target/scala-2.12/sbt_2.12-0.1.0-SNAPSHOT.jar ...
[info] Done packaging.
[info] Running HelloWorld
Hello World!
[success] Total time: 2 s, completed 12 Dec 2018, 01.28.29
```

More Sophisticated Examples

Simple sbt Project

build.sbt

```
enablePlugins(JavaAppPackaging)

name := "XWiki2html"
organization := "de.uniwue.jpp"
version := "0.9"
libraryDependencies ++= xwiki :+ log4j

lazy val xwiki = Seq(
    "org.xwiki.rendering" % "xwiki-rendering-syntax-xwiki21",
    "org.xwiki.rendering" % "xwiki-rendering-syntax-xhtml",
    "org.xwiki.commons" % "xwiki-commons-component-default")
.map(_ % "10.1")

lazy val log4j = "ch.qos.logback" % "logback-classic" % "1.3.0-alpha4"
```

Simple sbt Project

plugins.sbt

```
addSbtPlugin("com.typesafe.sbt" % "sbt-native-packager" % "1.3.15")
```

Simple mill Project

build.sc

```
import mill._, scalalib._

object main extends JavaModule {

  object test extends Tests {

    def ivyDeps = Agg(
      ivy"com.novocode:junit-interface:0.11",
      ivy"junit:junit:4.12",
      ivy"org.assertj:assertj-core:3.11.1")

    def testFrameworks = Seq("com.novocode.junit.JUnitFramework")
  }
}
```

Multi Build sbt Project

```
organization := "de.uniwue"
scalaVersion := "2.12.6"
version := "3.6.4"
name := "agent-prototype"
enablePlugins(JavaAppPackaging)

libraryDependencies ++= Seq(
  "com.typesafe" % "config" % "1.3.3",
  "com.typesafe.scala-logging" %% "scala-logging" % "3.9.0",
  "com.github.pathikrit" %% "better-files" % "3.4.0",
  "org.scala-lang.modules" %% "scala-xml" % "1.1.0",
  "ch.qos.logback" % "logback-classic" % "1.2.3") ++ akka ++ circe

lazy val akka = Seq(
  "com.typesafe.akka" %% "akka-remote",
  "com.typesafe.akka" %% "akka-actor"
).map(_ % "2.5.14")
```

Multi Build sbt Project

```
lazy val circe = Seq(
  "io.circe" %% "circe-core",
  "io.circe" %% "circe-generic-extras",
  "io.circe" %% "circe-parser"
).map(_ % "0.10.1")

lazy val serverDummy = project.in(file("dummy"))
        .settings(
          scalaVersion := "2.12.6",
          name := "agent-dummy",
          libraryDependencies ++= akka ++ circe,
          unmanagedBase := baseDirectory.value / ".." / "lib")
        .enablePlugins(JavaAppPackaging)
```

Live Demo?