



Hello Gradle

TestNG, Eclipse, IntelliJ IDEA

Óbudai Egyetem, Java Enterprise Edition

Műszaki Informatika szak

Labor 2

Bedők Dávid
2017.09.17.
v0.2

Java project struktúra

Avagy hogyan szervezzük forrásainkat?

- ▷ **javac**
- ▷ **IDE** (Eclipse, IntelliJ IDEA, ...)
- ▷ **build tool**

Miket érdemes figyelembe venni?

- ▷ Lesznek **egység tesztjeink**?
 - külön source folder a teszteknek (**test** és **main**)
- ▷ Lesznek **erőforrásaink**?
 - külön source folder az erőforrásoknak (**resources**)
- ▷ Lesznek **nem java forrásaink** is?
 - külön source folder a forrásoknak (**java, scala, groovy**, stb.)

Java Build eszközök:

- ▷ batch files / bash scripts
- ▷ Apache ANT (+ Apache IVY)
- ▷ Apache Maven
- ▷ Gradle

Java filozófia

- ▷ átláthatóság, egyértelműség
- ▷ **classpath** (cp) használata
- ▷ JAR, WAR, EAR, SAR*, APK** csomagolások alkalmazása

SAR*: JBoss specifikus
APK**: android specifikus

Java project struktúra

javac

Csak a "fantázia" szab határt a konfigurációnak (nincsenek szabályok)!
javac programnak megadjuk az összes olyan könyvtárat, ahol forrásállományok találhatóak.

Könyvtár struktúra

```
bin/  
src1/ → source folder  
src2/ → source folder
```

Az Application osztály használja az ImperialToMetricCalculator osztály egy példányát. Mindkettő azonos csomagba fordul, ezért az alkalmazás nem importálja a kalkulátort, futás időben egy helyen lesznek.

 \hellotest

```
1 > javac -d ./bin ./src1/hu/qwaevisz/demo/Application.java  
    ./src2/hu/qwaevisz/demo/ImperialToMetricCalculator.java  
2 > java -cp ./bin hu.qwaevisz.demo.Application
```

Java project struktúra

Eclipse IDE

Eclipse "Java project" alapértelmezett konfigurációja:

Könyvtár struktúra

```
bin/  
src/ → source folder
```

Eclipse "Java project" egység tesztekkel konfigurációja:

Könyvtár struktúra

```
bin/  
src/  
  main/ → source folder  
  test/ → source folder
```

A legtöbb IDE-ben ezek teljeskörűen konfigurálható elemek (Eclipse: Project properties | Java Build Path | Source tab)!

Gradle java plugin alapértelmezett konfigurációja:

Könyvtár struktúra

```
bin/  
src/  
  main/  
    java/ → source folder  
    resources/ → source folder  
  test/  
    java/ → source folder  
    resources/ → source folder
```

Ezekről Gradle-ben is teljes körűen el lehet térni, de nem érdemes. Ha élünk ezzel, akkor minimális konfigurációval el tudjuk kezdeni a munkát!

Megjegyzés: A resources könyvtárak bár tipikusan nem tartalmazzak lefordítandó forrást, ugyanúgy rajta vannak a classpath-on, hogy runtime a tartalmuk feldolgozható legyen.

JAR - Java ARchive

ZIP formátum, mely (Java) byte code-okat (*.class), konfigurációs állományokat (pl. *.properties, *.xml, stb.) illetve egy speciális metaadatokat (kulcs-érték párokat) tartalmazó **MANIFEST.MF** állományt tartalmaz.

Könyvtár struktúra

```
META-INF/  
  MANIFEST.MF  
hu/  
  qvaevisz/  
    demo/  
      HelloWorld.class  
      Lorem.class  
log4j.xml
```

Struktúrája előre definiált, opcionálisan forrásállományokat (pl. *.java, *.groovy, stb.) is magával hordoz (azonos helyen a byte code-dal).

```
1 Manifest-Version: 1.0  
2 Created-By: 1.7.0_67 (Oracle Corporation)
```

MANIFEST.MF

Executable JAR file

A Main-Class kulcsnak szerepelnie kell a MANIFEST.MF állományban, és a belépési pontként szolgáló osztály full qualified neve lesz az értéke.

```
1 Manifest-Version: 1.0
2 Created-By: 1.7.0_67 (Oracle Corporation)
3 Main-Class: hu.qwaevisz.demo.Application
```

MANIFEST.MF

```
1 > cd bin
2 > jar cvfe calculator.jar hu.qwaevisz.demo.Application
   hu/qwaevisz/demo/Application.class
   hu/qwaevisz/demo/ImperialToMetricCalculator.class
3 > cd ..
4 > java -jar bin/calculator.jar
```

create new archive
verbose
specify archive **f**ile name (2)
specify **e**ntry point (main class) (3)



Eclipse IDE for Java EE Developers

Letöltés: <https://www.eclipse.org/downloads/>

Verzió: 4.7.0

Telepítés: unzip vagy installer

Integrált kiegészítők (plugins):

- ▷ Gradle
- ▷ Maven
- ▷ Git
- ▷ EclEmma Java Code Coverage
- ▷ ...

Magyar billentyűzet kiosztás esetén a "{" (Ctrl + B) használata: Preferences | General | Keys | Skip all breakpoints (Ctrl + Alt + B) → Unbind

Eclipse IDE alapvető kezelése: <http://users.nik.uni-obuda.hu/bedok.david/jse.html>

Továbbiak telepítése (Help / Eclipse Marketplace):

- ▷ **TestNG** (filter: testng)
 - <http://beust.com/eclipse>


Eclipse beállításai

Code Style Formatter

Window | Preferences (type: formatter)

▷ Java | Code Style | Formatter

- New... / Import...: **uni-obuda-java-formatter**
 - Initialize: Eclipse [build-in]
 - Indentation | Indent | Statement within 'switch' body
 - Line Wrapping | General | Maximum line width: 160
 - Line Wrapping | Enum declaration
 - * Policy: Wrap all elements, every element on a new line
 - * Constants policy: Wrap all elements, every element on a new line + Force split
 - Comments | Line width | Maximum: 120

 \eclipse\uni-obuda-java-formatter.xml

Window | Preferences (type: save actions)

▷ Java | Editor | Save Actions

- Perform the selected actions on save
 - **Format source code** (all lines)
 - Organize imports
 - Additional actions - Configure
 - * Code Organizing: Remove trailing whitespaces
 - * Code Style: Use blocks in if/while/for/do statements
 - * Member Accesses: Use 'this' qualifier for field accesses:
Always
 - * Member Accesses: Use 'this' qualifier for method accesses:
Always
 - * Unnecessary Code: Remove unused imports



Letöltés: <https://www.jetbrains.com/idea/>

- ▷ **Kereskedelmi termék**
- ▷ Community verzió pl. JavaEE-t nem támogat, de ezen IDE támogatás nélkül is tökéletesen alkalmas professzionális munkára (akár JavaEE projektek kezelésére is)
- ▷ Néhol gyorsabb mint Eclipse
- ▷ Más billentyűkiosztás, átszokni nem könnyű, de van lehetőség más IDE kiosztásának használatára
- ▷ Beépített Maven/Gradle/Git plugin

Hello World

src | **main** | java | hu | qvaevisz | hello | Application.java

```
1 package hu.qvaevisz.hello;
2
3 public class Application {
4
5     public static void main(final String[] args) {
6         System.out.println("Hello World");
7     }
8
9     public int add(final int a, final int b) {
10        return a + b;
11    }
12
13 }
```

Application.java



[gradle|maven]\helloworld

TestNG

3rd party library

- ▷ <http://testng.org/>
- ▷ GitHub: <https://github.com/cbeust/testng>
- ▷ Verzió: **6.11**
- ▷ Artifactory URL:
 - 'org.testng:testng:6.11'
 - group/groupId: **org.testng**
 - name/artifactId: **testng**
 - version: **6.11**

Egység teszt TestNG-vel

src | test | java | hu | qwaevisz | hello | ApplicationTest.java

```
1 package hu.qwaevisz.hello;
2
3 import org.testng.Assert;
4 import org.testng.annotations.Test;
5
6 public class ApplicationTest {
7
8     @Test
9     public void addNumbers() {
10         Application app = new Application();
11         Assert.assertEquals(app.add(2, 3), 5);
12     }
13
14 }
```

ApplicationTest.java



- ▷ <https://gradle.org/>
- ▷ Letöltés: <https://gradle.org/releases/>
- ▷ Verzió: 4.1
- ▷ Java, C++, Python és megannyi további programozási nyelv támogatása
- ▷ monorepo és multi-repo támogatás
- ▷ multi-language, multi-platform, multi-project és multi-channel software fejlesztés (SaaS: Software as a Service)
- ▷ Telepítés: unzip

Környezeti változók:

- ▷ **GRADLE_HOME** → c:\apps\gradle-2.6
- ▷ **Path** módosítása → %Path%;%GRADLE_HOME%\bin

- ▷ Apache **Ant** alapú build rendszer
 - <http://ant.apache.org/>
- ▷ Apache **Ivy** stílusában készült dependency management
 - <http://ant.apache.org/ivy/>
- ▷ intelligens alapértelmezett környezetek Apache **Maven** alapján
 - <https://maven.apache.org/>
- ▷ a **Git** sebességével és hash algoritmusával
- ▷ metaprogramozás Apache **Groovy** / JetBrains **Kotlin** alapokon
 - <http://groovy-lang.org/>
 - <https://kotlinlang.org/>
- ▷ Directed Acyclic Graph (DAG)


```
1 > gradle --version
2
3 -----
4 Gradle 4.1
5 -----
6
7 Build time:      2017-08-07 14:38:48 UTC
8 Revision:       941559e020f6c357ebb08d5c67acdb858a3defc2
9
10 Groovy:         2.4.11
11 Ant:           Apache Ant(TM) version 1.9.6 compiled on June 29
12              2015
13 JVM:           1.8.0_101 (Oracle Corporation 25.101-b13)
14 OS:            Mac OS X 10.12.6 x86_64
```

- ▶ Gradle **eclipse plugin**-ja legyártja az összes szükséges konfigurációs állományt ahhoz, hogy eclipse project lehessen (semmi extra nem kell egy eclipse projekthez, néhány IDE specifikus XML állomány).
- ▶ Eclipse **Gradle plugin**-ja felismeri a gradle konfigurációs állományait, és annak megfelelően képes kezelni a projektet ezt követően.
- ▶ A kettő együttes alkalmazása régebben gyakoribb volt, ma már a Gradle hivatalos Eclipse pluginje kényelmes, kompromisszummentes és professzionális megoldást ad.

Hello Gradle!

Hello World alkalmazás kezelése Gradle segítségével



[gradle|maven]\helloworld

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

build.gradle

```
1 > gradle clean build
2
3 > Configure project :
4 Task name: info
5 Project name: helloworld
6
7
8 BUILD SUCCESSFUL in 1s
9 5 actionable tasks: 5 executed
```

▷ output: build\libs\helloworld.jar (non-executable)

▷ clean, build → 'java' plugin task-jai

- build függő taskjai: compileJava, jar, assemble, test, check, ...

Gradle task készítése

```
1 apply plugin: 'java'
2
3 sourceCompatibility = 1.7
4 version = '1.0'
5
6 task info() {
7     println "Task name: " + name
8     println "Project name: " + project.name
9     println "Project version: " + version
10 }
```

build.gradle

```
1 > gradle info
2
3 > Configure project :
4 Task name: info
5 Project name: helloworld
6 Project version: 1.0
7
8
9 BUILD SUCCESSFUL in 0s
```

Executable JAR

Gradle jar task módosítása

```
1 [..]
2 jar {
3     manifest {
4         attributes 'Implementation-Title': 'Gradle Demo
5                 Application',
6                 'Implementation-Version': version,
7                 'Main-Class': 'hu.qwaevisz.gradle.Application'
8     }
9 }
10 task run( type: Exec ) {
11     workingDir 'build/libs'
12     commandLine 'java', '-jar', "${project.name}-${version}.jar"
13 }
```

build.gradle

```
1 > gradle clean build run
2
3 [..]
4
5 > Task :run
6 Hello World
7
8 BUILD SUCCESSFUL in 2s
9 6 actionable tasks: 6 executed
```

Gradle TestNG integráció

```
1 [..]
2 def testngVersion = '6.9.10'
3
4 repositories { mavenCentral() }
5
6 dependencies {
7     testCompile group: 'org.testng', name: 'testng' , version:
8         testngVersion
9 }
10 test {
11     useTestNG()
12 }
```

Alapértelmezett test keretrendszer: **junit.**

build.gradle

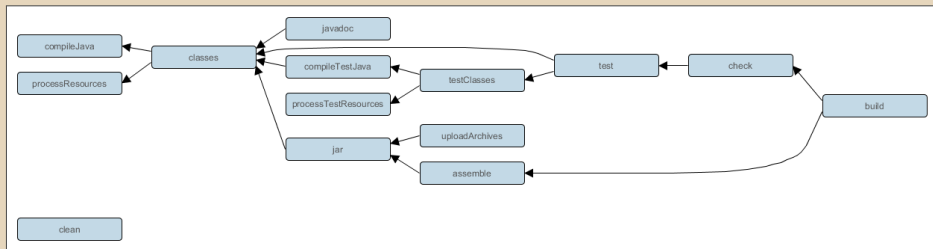
```
1 > gradle test
2 [..]
3 > Task :test
4
5 Gradle suite > Gradle test > hu.qwaevisz.hello.ApplicationTest.addNumbers FAILED
6     java.lang.AssertionError at ApplicationTest.java:11
7
8 1 test completed, 1 failed
9 [..]
10 BUILD FAILED in 1s
11 3 actionable tasks: 3 executed
```

Gradle Java plugin

Tasks

https://docs.gradle.org/current/userguide/java_plugin.html

Task: compileJava, compileTestJava, jar, javadoc, test, clean, ...

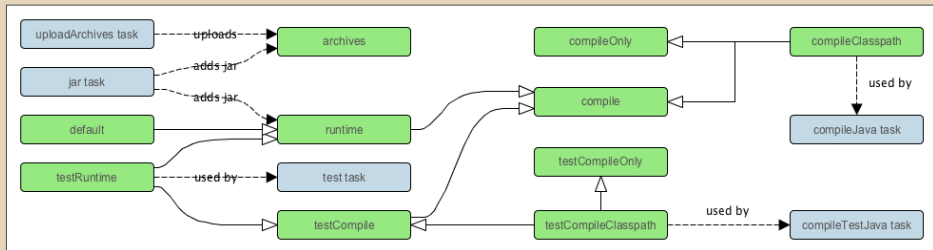


Gradle Java plugin

Dependency Configurations

- compile** Fordítási és futási időben szükséges függőség.
- compileOnly** Fordítási időben szükséges függőség.
- runtime** Futási időben szükséges függőség.
- testCompile** Kiegészítő függőség test task fordításakor és futásakor.
- testRuntime** Kiegészítő futási idejű függőség test task futásakor.
- archives** A projekt által létrejött artifact (pl. jar)

... ..



Gradle Eclipse integráció

```
1 [...]
2 apply plugin: 'eclipse'
3 [...]
```

build.gradle

```
1 > gradle eclipse
```

További taskok: eclipseClasspath, eclipseJdt, eclipseProject

Könyvtár struktúra

```
settings/
  org.eclipse.jdt.core.prefs
.classpath
.project
```

File | Import... | Gradle | Existing Gradle Project

- ▷ Project root directory: \helloworld
- ▷ Import options: Gradle wrapper

Mit honnan?

Az Eclipse Gradle plugin (buildship) elsősorban a Gradle projekt struktúra kezelése, a projektek importálása miatt fontos. Eclipse-ből futtatni Gradle task-okat már ízlés dolga (van aki minden ilyesmit Eclipse-ből szeret futtatni, van aki ezt külön terminal/command window-ban teszi meg).

Gradle IntelliJ integráció

```
1 [...]
2 apply plugin: 'idea'
3 [...]
```

build.gradle

```
1 > gradle idea
```

További taskok: cleanIdea, ideaModule, ideaProject, ideaWorkspace

Könyvtár struktúra

```
helloworld.iml
helloworld.ipr
helloworld.iws
```