Gradle

Java Build Automation with Passion

ivan.macalak@posam.sk

JUGZA
Java User Group Žilina

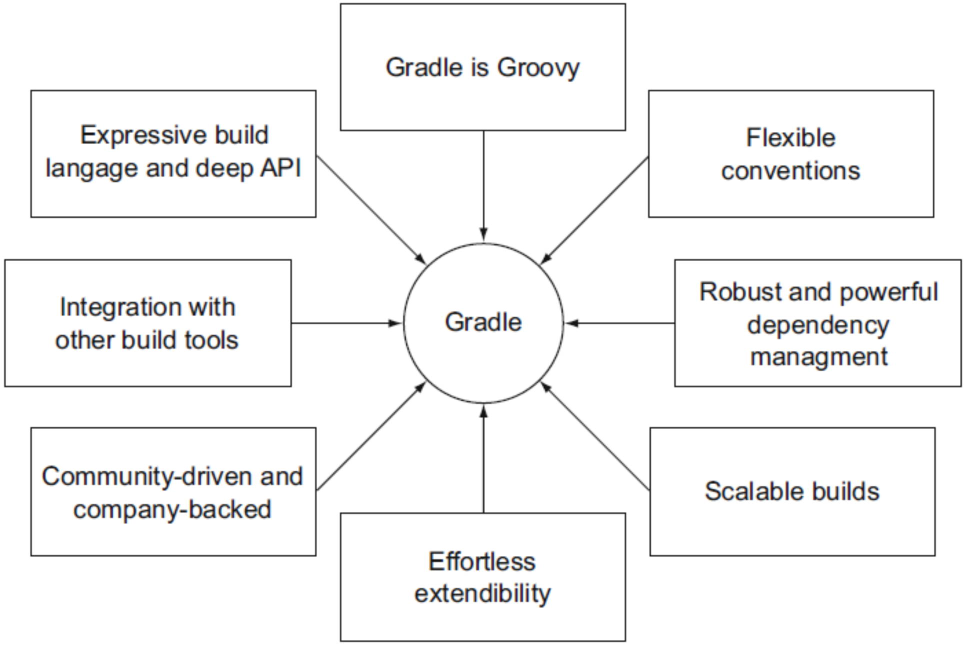
Gradle introduction

- Gradle is flexible general purpose build tool
- It combines Ant with Dependency management and Maven conventions
- It provides Groovy based DSL interface for writing build scripts
- Conventions with great flexibility

<APACHE ANT>	Flexibility Full control Chaining of targets
ivy™	Dependency management
maven	Convention over configuration Multimodule projects Extensibility via plugins
Gant	Groovy DSL on top of Ant

Gradle offers

- Build automation, checking, testing, publishing
- Dependency management based on Apache Ivy
- Support of multi project builds
- Incremental builds
- Pluggable architecture
- IDE support and integration


Gradle is Groovy
Expressive build langage and deep API
Flexible conventions
Integration with other build tools
Gradle
Robust and powerful dependency managment
Community-driven and company-backed
Scalable builds
Effortless extendibility

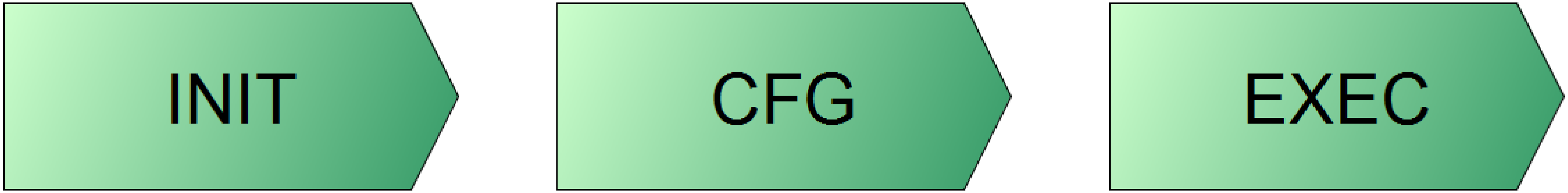
Gradle build scripts

- Build automation, checking, testing, publishing
- Project is made up of one or more Tasks
- Build scripts are code –you can use Groovy
- Task dependencies
- You can use Ant task in gradle script
- Tasks come from plug-ins
- Default Tasks

```
<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>com.mycompany.app</groupId>
   <artifactId>my-app</artifactId>
   <packaging>jar</packaging>
   <version>1.0-SNAPSHOT</version>

   <dependencies>
      <dependency>
         <groupId>junit</groupId>
         <artifactId>junit</artifactId>
         <version>4.11</version>
         <scope>test</scope>
      </dependency>
   </dependencies>
</project>
```

```
apply plugin: 'java'
group = 'com.mycompany.app'
archivesBaseName = 'my-app'
version = '1.0-SNAPSHOT'

repositories {
   mavenCentral()
}

dependencies {
   testCompile 'junit:junit:4.11'
}
```

Gradle build lifecycle



INIT

CFG

EXEC

Gradle Dependency Management

- Declarative dependencies grouped into configurations

```
dependencies {
    compile  'org.hibernate:hibernate-core:3.6.7.Final'
    testCompile  'junit:junit:4.11‘
}
```

- Repositories (Maven, Ivy, Local directory)

```
repositories {
    jcenter()
}
```

- Publishing artifacts

Gradle command line

- Built-in tasks
  - listing projects: `gradle projects`
  - listing tasks: `gradle tasks`
  - listing project dependencies: `gradle dependencies`
- Multiple tasks execution: `gradle compile test`
- Excluding tasks: `gradledist -x test`
- You can use gradleGUI: `gradle –gui`

Gradle wrapper

- Automatically downloads Gradle distribution
- You should check it into version control
- Use `gradlew` instead `gradle` command
- Available on Windows and Linux

Gradle daemon

- Background process
- Speeds up the build
- Ideal when you build frequently
- Can be enabled via CLI, or for environment
  - `gradle -daemon`
  - `GRADLE_OPTS: -Dorg.gradle.daemon=true`
  - `org.gradle.daemon=true` in the `GRADLE_USER_HOME/gradle.properties`

Gradle build environment

- GRADLE_OPTS or JAVA_OPTS
- properties in gradle.properties in following order
  - From gradle.properties located in project build dir
  - From gradle.properties located in gradle user home
  - from system properties, e.g. when -Dsome.property is used in the command line

Gradle more than Java

- Multilanguage (Polygot) builds
  - Java+Groovy+Scala…
- Official build system for Android studio
- Can be used to build C++ projects

Gradle Integration

- Java IDE integration plug-ins
  - Intellij IDEA, Eclipse, NetBeans
- Gradle plugin for IDE project files generation
  - eclipse, idea
- CI Servers integration
  - Jenkins, TeamCity
- http://gradle.org/why/integrates-with-everything/

References

- https://docs.gradle.org/current/dsl/index.html
- https://plugins.gradle.org/
- http://gradle.org/maven_vs_gradle/
- http://gradle.org/case-study-continuous-delivery-netflix/
- http://gradle.org/case-study-gradle-continuous-delivery-linkedin/
- http://gradle.org/open-source-build-system-evaluation-in-the-age-of-continuous-delivery-part-1/