

Java EE

Maven

Petite histoire du build

- **Make** ~ - 200 BC
 - Tool historique sous unix pour construire des fichiers et des livrables
- **Ant** ~ 1998
 - Ecrit en Java, permet d'exécuter des tâches, difficile à partager
 - Gestion des dépendances avec Ivy
- **Maven** ~ 2002
 - « Convention over configuration »
 - Extensible grâce à des plugins
 - A résolu la plus part des problèmes que l'on rencontre dans un projet Java (EE)
- **Graddle** ~ 2008
 - Permet de définir des tâches en Groovy
 - Plus flexible



Gestion des dépendances

- Gérer des dépendances: un problème complexe
 - Quelles sont les libs requisent par mon projet ?
 - Ou les trouver ?
 - Dans quelle versions ?
 - Les libs sont-elles compatibles entre elles ?

Définir une dépendance

Dans Maven une dépendance est définie par plusieurs attributs:

- Group ID: un nom de groupe
- Artifact ID: le nom du projet
- Version: La version de l'artefact suivant la convention Majeur.Mineur.Patch(-SNAPSHOT)
- Scope: optionnel, indique à quel moment la dépendance est requise

```
<dependencies>
  <dependency>
    <groupId> junit </groupId>
    <artifactId> junit </artifactId>
    <version> 4.8.2 </version>
    <scope> test </scope>
  </dependency>
</dependencies>
```

Visibilité d'une dépendance

- L'attribut scope permet de définir à quel moment une dépendance est nécessaire
 - **test**: nécessaire uniquement pour compiler et exécuter les tests
 - **provided** : nécessaire uniquement à la compilation, elle sera fournie par l'environnement à l'exécution
 - **compile** : nécessaire à la compilation et à l'exécution
 - **Runtime** : nécessaire uniquement à l'exécution mais pas à la compilation

Conventions

Maven impose une convention sur la structure des projets

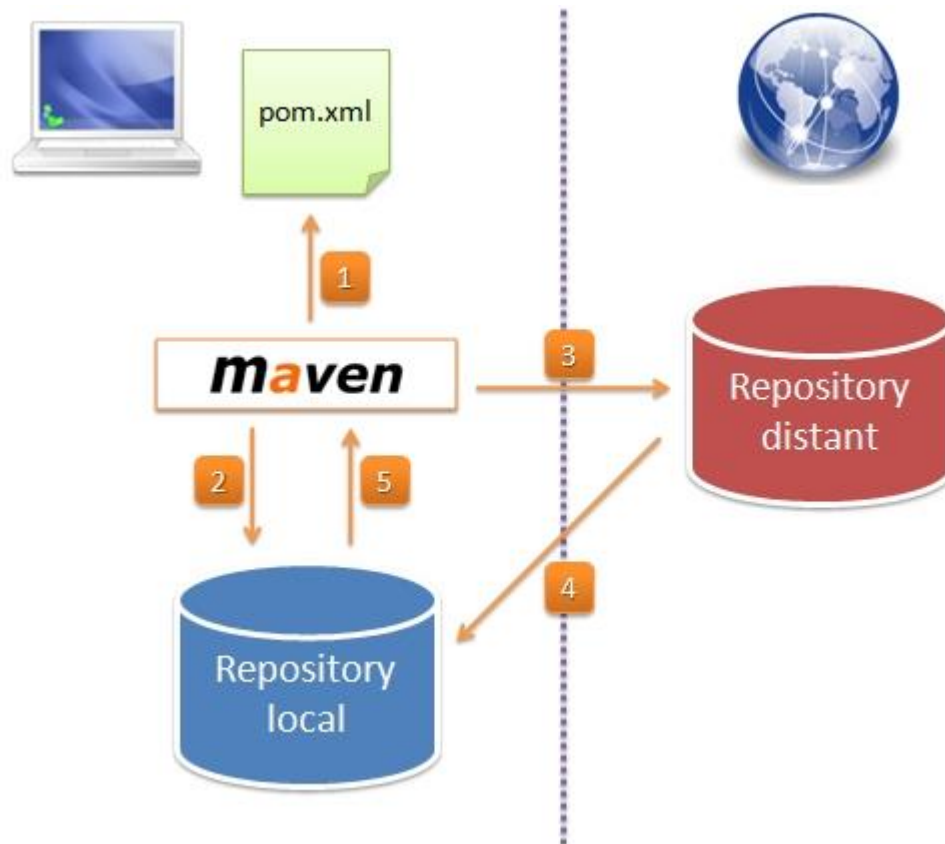
src	tout le code source du projet
src/main	les sources du livrable final
src/test	les sources de tests
src/main/java	le code JAVA
src/main/webapp	les sources web (ex. JSP)
src/main/resources	les ressources non compilées (ex. Configuration)
src/test/java	le code JAVA des tests
src/test/resources	les ressources non compilées des tests

Les repository Maven

- Une repository est constitué d'un ensemble de library identifié par un groupe, un nom et un numéro de version
- Un repository permet de retrouver facilement des library existante
- La plus part des dépendances sont disponibles sur le serveur Maven central
- De nombreuses entreprises possèdent leur propre repository Maven (Nexus)
- C'est LA fonctionnalité de Maven qui a révolutionné le monde JAVA

Repository local vs distant

- Le repository local permet de posséder un cache local des dépendances déjà téléchargé
- Il est commun à tous les projets
- Par défaut Maven utilise en priorité le repository local



Gestion des versions

Maven permet de gérer simplement du versioning :

- Définir différentes version d'un livrable
- Différencier les release et les snapshots

```
<?xml version="1.0" encoding="UTF-8"?>
<project>
  <modelVersion>4.0.0</modelVersion>
  <groupId>org.lds.training</groupId>
  <artifactId>maven-training</artifactId>
  <version>1.0-SNAPSHOT</version>
</project>
```

Autres fonctionnalités de Maven

- Construire des artifact prêt à être livrés sous différents formats: jar, war, ear...
- Gérer du filtering
 - Différencier la configuration de mon application par environnement
- Définir comment exécuter des tests
- Paramétrer de l'analyse statique de code
- Et bien d'autres grâce à l'existence de plugins

Héritage

Il est possible d'utiliser de l'héritage entre fichiers POM.xml

- Factorisation des configurations
- Très utile pour le partage des conventions dans le monde professionnel

```
<?xml version="1.0" encoding="UTF-8"?>
<project>
  <parent>
    <artifactId>maven-training-parent</artifactId>
    <groupId>org.lds.training</groupId>
    <version>1.0</version>
  </parent>
  <modelVersion>4.0.0</modelVersion>
  <artifactId>maven-training</artifactId>
  <packaging>jar</packaging>
</project>
```

Détails sur Maven

- Maven stock les dépendances en local dans le répertoire .m2 (en général situé dans le home)
- Dans ce répertoire on trouve notamment un fichier settings.xml qui permet de configurer des profils par défaut, des proxy, des repository internes...