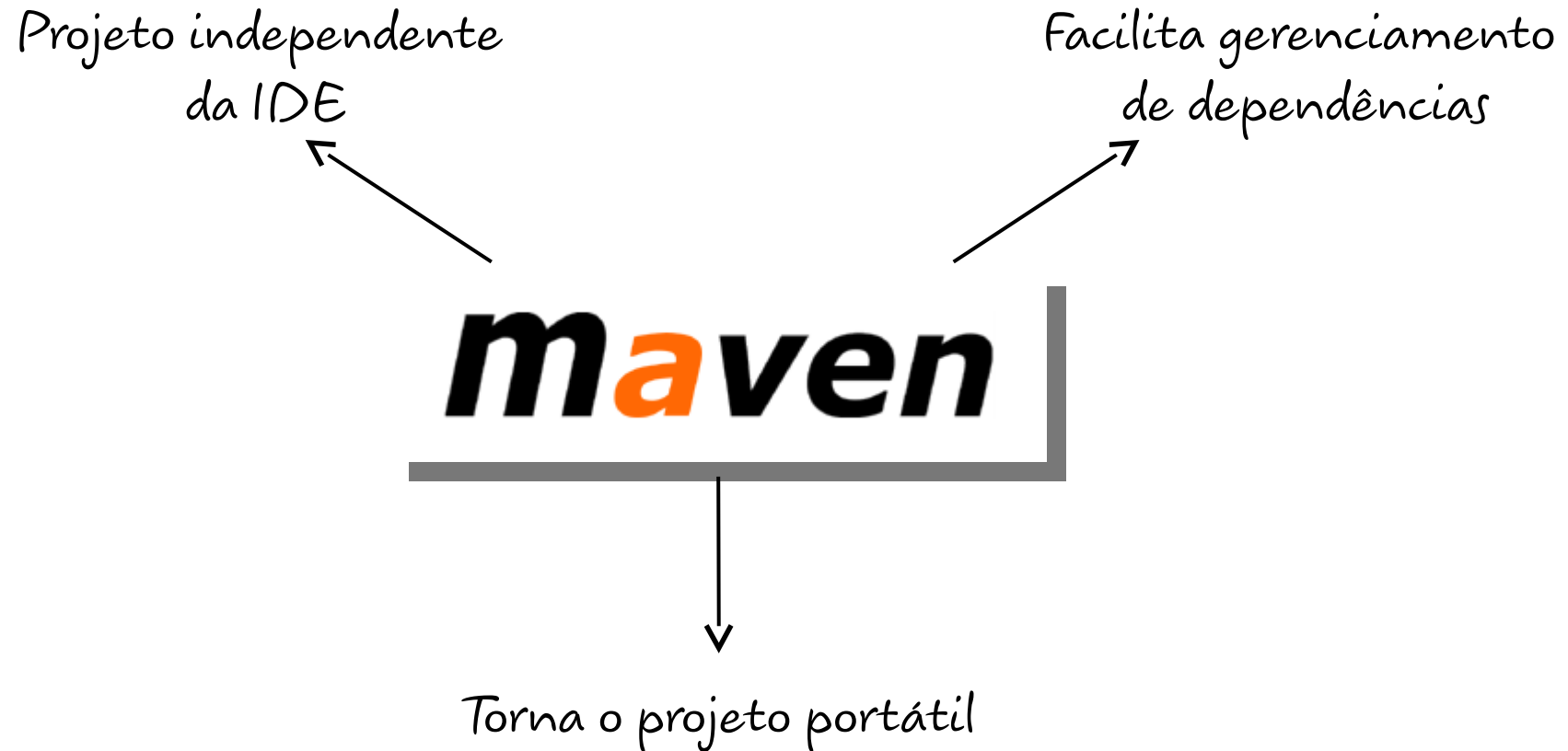


Maven



Instalando o Maven

Execute o comando para instalar a versão mais atual do maven

```
sudo apt-get install maven
```

Execute o comando `mvn --version` para verificar a versão instalada

```
Apache Maven 3.0.5
Maven home: /usr/share/maven
Java version: 1.7.0_80, vendor: Oracle Corporation
Java home: /usr/lib/jvm/java-7-oracle/jre
Default locale: pt_BR, platform encoding: UTF-8
OS name: "linux", version: "3.16.0-44-generic", arch: "amd64", family: "unix"
```

Criação de um projeto

Atributos Obrigatórios

Group Id

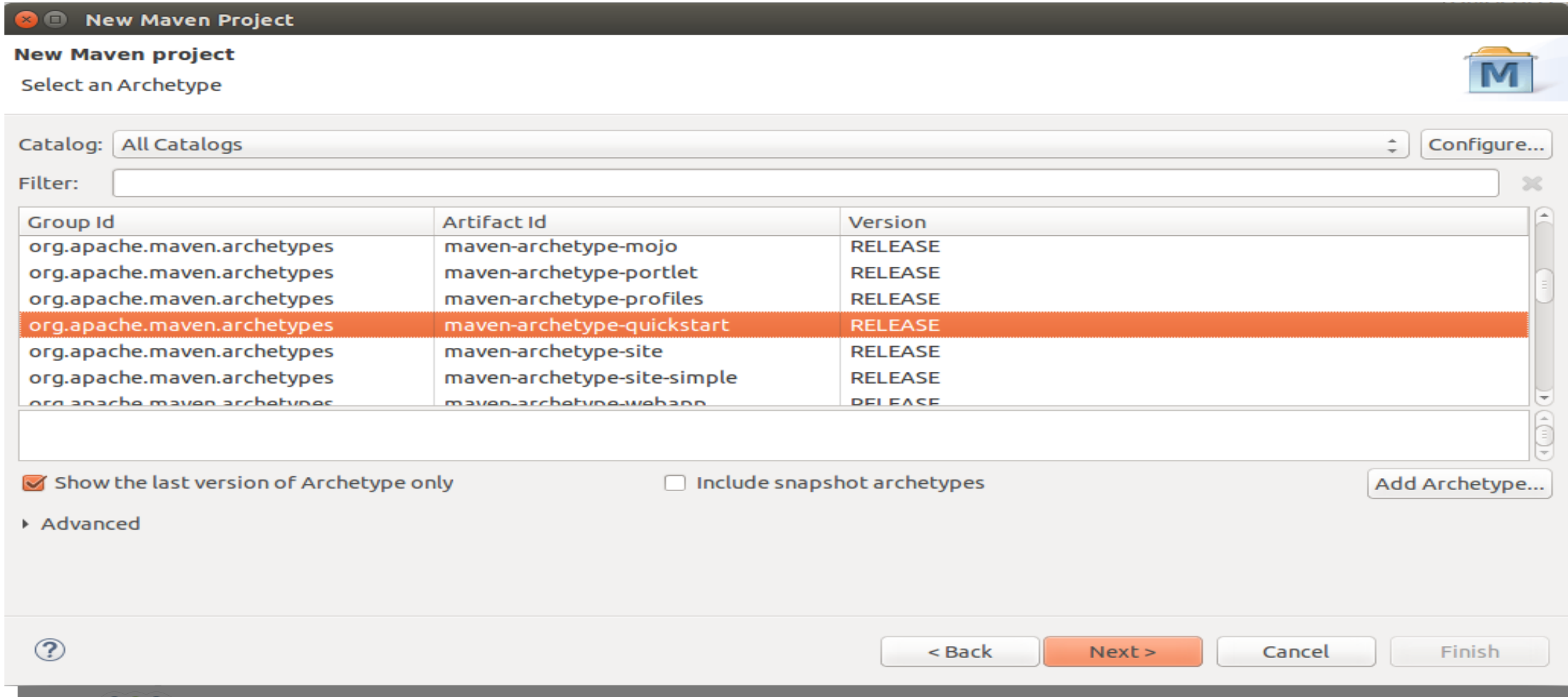
ID do grupo do projeto.
Identificador único de
uma organização

```
mvn archetype:create  
-DgroupId=[group id do seu projeto]  
-DartifactId=[artifact id do seu projeto]  
-DarchetypeArtifactId=maven-archetype-webapp
```

Artifact ID

ID do projeto. Nome
do projeto.

Criando o mesmo projeto utilizando o eclipse



Criando o mesmo projeto utilizando o eclipse

New Maven Project

New Maven project
Specify Archetype parameters

Group Id:

Artifact Id:

Version:

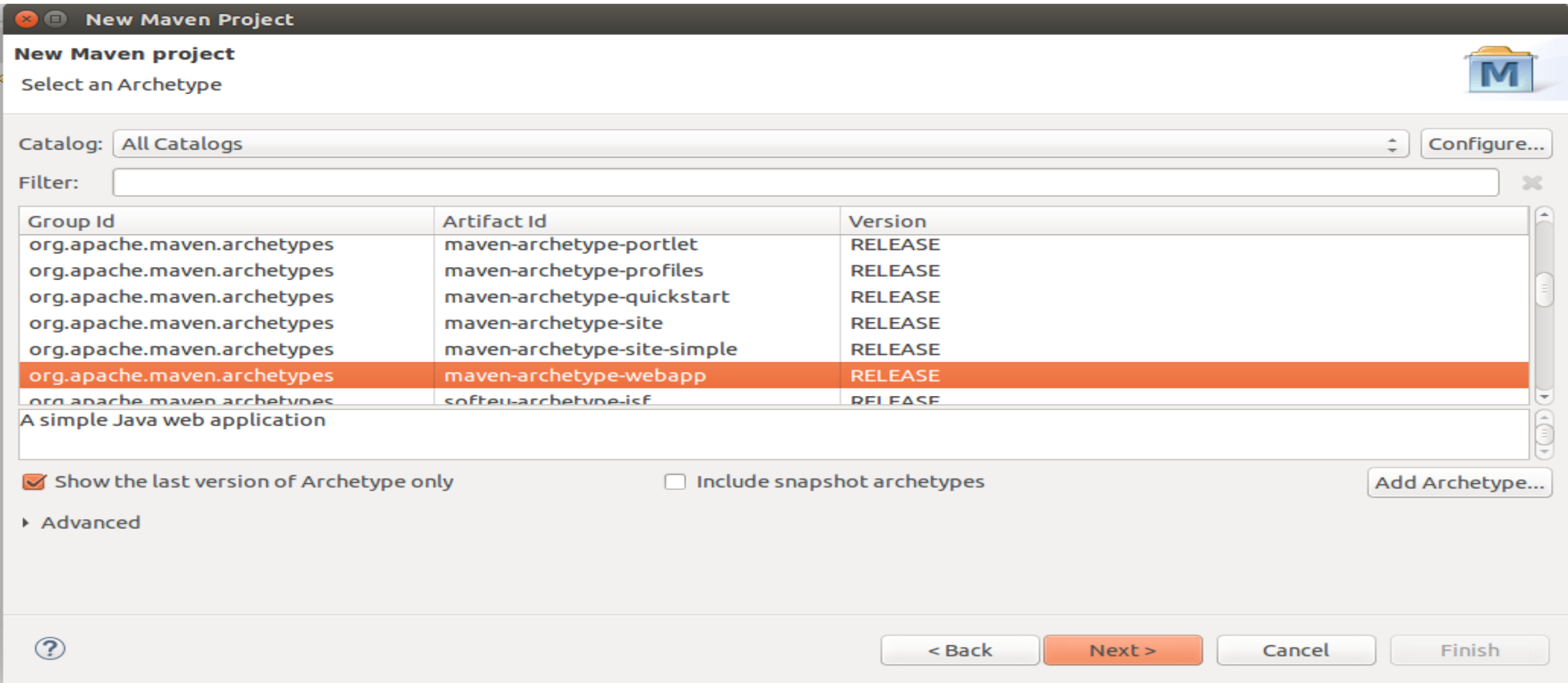
Package:

Properties available from archetype:

Name	Value
------	-------

► Advanced

Criando o mesmo projeto utilizando o eclipse



New Maven Project

Select an Archetype

Catalog: All Catalogs Configure...

Filter:

Group Id	Artifact Id	Version
org.apache.maven.archetypes	maven-archetype-portlet	RELEASE
org.apache.maven.archetypes	maven-archetype-profiles	RELEASE
org.apache.maven.archetypes	maven-archetype-quickstart	RELEASE
org.apache.maven.archetypes	maven-archetype-site	RELEASE
org.apache.maven.archetypes	maven-archetype-site-simple	RELEASE
org.apache.maven.archetypes	maven-archetype-webapp	RELEASE
org.apache.maven.archetypes	software-archetype-isf	RELEASE

A simple Java web application

☒ Show the last version of Archetype only ☐ Include snapshot archetypes Add Archetype...

▶ Advanced

? < Back Next > Cancel Finish

Criando o mesmo projeto utilizando o eclipse

New Maven Project

New Maven project
Specify Archetype parameters

Group Id:

Artifact Id:

Version:

Package:

Properties available from archetype:

Name	Value
------	-------

► Advanced

Tipos de projeto empacotamento

Empacotamento para projetos libs (padrão do maven)

```
<packaging>jar</packaging>
```

Empacotamento para projetos modulares

```
<packaging>pom</packaging>
```

Empacotamento para projetos EJB

```
<packaging>ejb</packaging>
```

Empacotamento para projetos web

```
<packaging>war</packaging>
```


Criando Projeto Maven Simples Não é feita uma seleção de arquétipo

New Maven Project

New Maven project

❌ Enter a group id for the artifact.

Artifact

Group Id:

Artifact Id:

Version: 0.0.1-SNAPSHOT

Packaging: jar

Name:

Description:

Parent Project

Group Id:

Artifact Id:

Version: Browse... Clear

▶ Advanced

? < Back Next > Cancel Finish

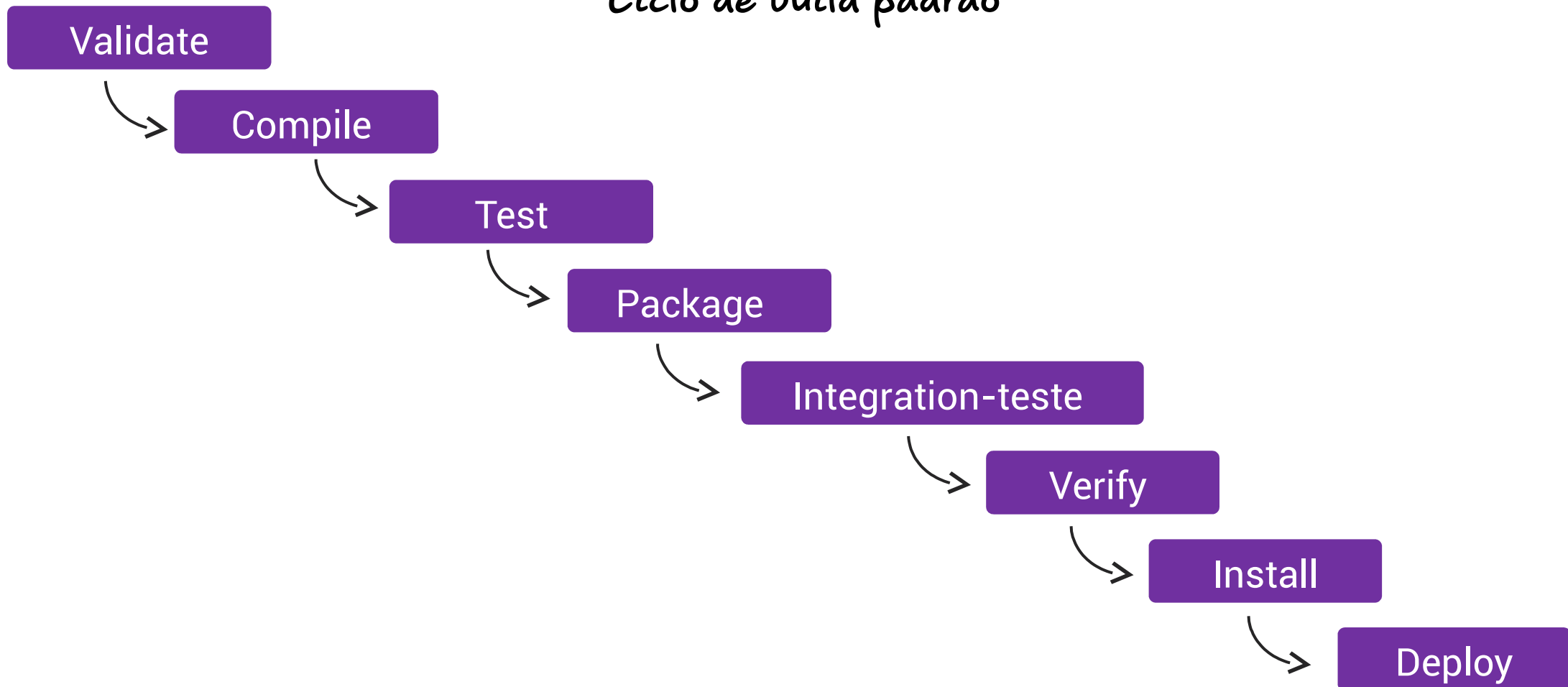
Alvos Maven

No diretório do projeto, os seguintes alvos estão

Alvos	Função
<code>mvn generate-sources</code>	Gera qualquer código fonte necessário para a aplicação, geralmente feito usando algum plugin apropriado.
<code>mvn compile</code>	Compila o projeto e o código fonte.
<code>mvn test-compile</code>	Compila os testes unitários do projeto.
<code>mvn test</code>	Executa os testes unitários no diretório src/test
<code>mvn package</code>	Empacota o código compilado no seu formato distribuível (JAR, WAR, etc).
<code>mvn integration-test</code>	Processa e implanta o pacote em um ambiente onde os teste de integração podem ser rodados.
<code>mvn install</code>	Instala o pacote no repositório local para ser usado como dependência em outros projetos na máquina local.
<code>mvn deploy</code>	Feito em um ambiente de integração ou de release, copia o pacote final para o repositório remoto para ser compartilhado com outros desenvolvedores projetos.

Ciclo de Vida Maven

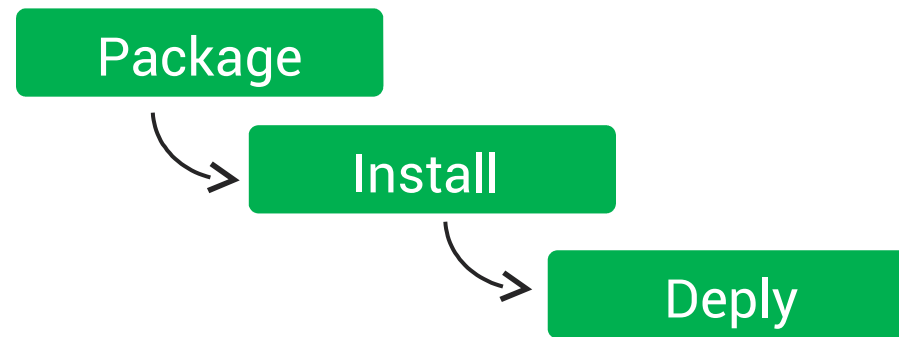
Ciclo de build padrão



*Executar **mvn test**, por exemplo, fará com que todas as fases anteriores também sejam executadas.*

Ciclo de Vida Maven

Empacotamento POM



Preparando o projeto para o Eclipse

Comando

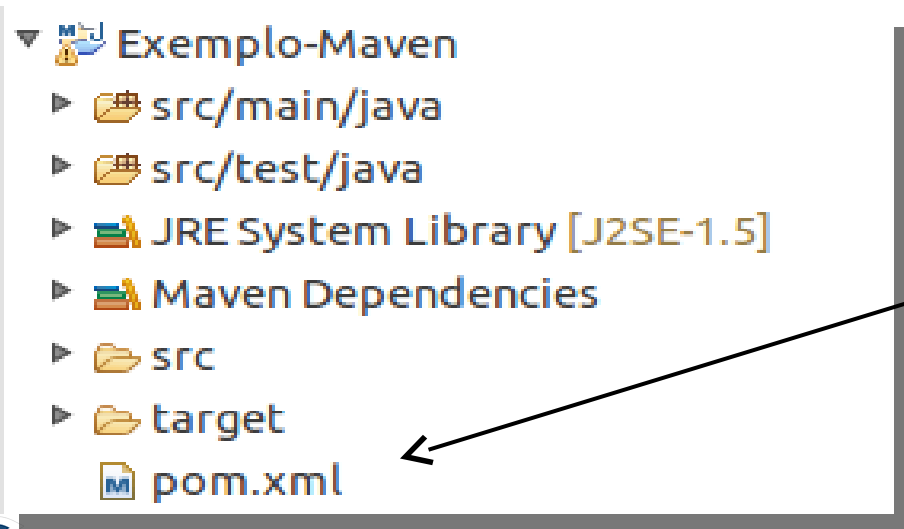
```
mvn eclipse:eclipse
```

Caso seja um projeto web

```
mvn eclipse:eclipse -Dwtpversion=2.0
```

No Eclipse:

Arquivo > Import... > Maven > Existing Maven Projects



Ao criar um projeto maven, um arquivo chamado pom.xml será gerado.

Pom.xml

Declaração de dependências, estrutura e características de um projeto.

Representação xml de um projeto maven

pom.xml

Todos os arquivos POM estendem de um super POM

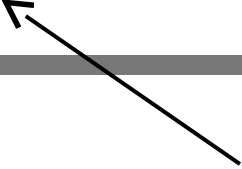
Maven utiliza o POM efetivo (`mvn help:effective-pom`)

Pom.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>threeway.projeto.maven</groupId>
  <artifactId>Exemplo-Maven</artifactId>
  <version>0.0.1-SNAPSHOT</version>
  <name>Exemplo Maven</name>
  <description>Descrição do projeto</description>

  <dependencies>
    <!-- Aqui ficam as dependências do projeto -->
  </dependencies>

</project>
```



Ainda podemos ter mais propriedades além das dependências, como configurações de build, informações do projeto (nome, url, desenvolvedores, etc.) e configurações do ambiente (repositórios, perfis, etc.).

Estrutura

```
<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>

  <!-- Relações do POM -->
  <groupId>threeway.maven</groupId>
  <artifactId>ExemploMaven</artifactId>
  <version>0.0.1-SNAPSHOT</version>
  <packaging>jar</packaging>

  <!-- Informações do Projeto -->
  <name>ExemploMaven</name>
  <url>http://maven.apache.org</url>

  <properties>
    <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
  </properties>

  <dependencies>
    <dependency>
      <groupId>junit</groupId>
      <artifactId>junit</artifactId>
      <version>3.8.1</version>
      <scope>test</scope>
    </dependency>

    <!-- Outras dependências -->

  </dependencies>

  <modules> </modules>
```


Estrutura

```
<!-- Informações do Projeto -->
<description> </description>
<inceptionYear> </inceptionYear>
<licenses> </licenses>
<developers> </developers>
<contributors> </contributors>
<organization> </organization>

<!-- Configurações de Build -->
<build> </build>
<reporting> </reporting>

<!-- Informações de Ambiente -->
<issueManagement> </issueManagement>
<ciManagement> </ciManagement>
<mailingLists> </mailingLists>
<scm> </scm>

<!-- Ambiente do Maven -->
<prerequisites> </prerequisites>
<repositories> </repositories>
<pluginRepositories> </pluginRepositories>
<distributionManagement> </distributionManagement>
<profiles> </profiles>

</project>
```

Transitividade de Dependências

Nível
1 2 3
A → B → C

C é dependência transitiva de A

Problema: mesmo Artifact-Id e Group-Id

- **Primeiro mais Próximo**

Dependências de um nível mais alto tem prioridade sobre as de níveis mais baixos.
Dependência direta tem prioridade sobre uma transitiva.

- **Primeiro Encontrado**

Em um mesmo nível, a primeira dependência encontrada é usada
(mesmo group-id/artifact-id e diferentes versões).

Exclusão de Dependências

```
<exclusions>
  <exclusion>
    <artifactId> </artifactId>
    <groupId> </groupId>
  </exclusion>
</exclusions>
```

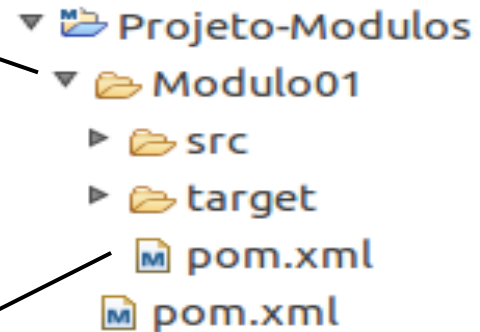
- ✓ Conflitos entre versões de um mesmo artefato.
- ✓ Conflito entre artefatos do seu projeto e artefatos da plataforma de implantação.
- ✓ Excluir dependências transitivas não desejadas.

POM.xml Projeto multimodulo

```
<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>threeway.projeto.maven</groupId>
  <artifactId>Projeto-Modulos</artifactId>
  <version>0.0.1-SNAPSHOT</version>
  <packaging>pom</packaging>

  <modules>
    <module>Modulo01</module>
  </modules>
</project>
```

Empacotamento deve ser do tipo pom



Projeto pai passa para seus módulos: Group id, Version, Dependencias, Plugins

Dependency Management

```
<modules>
  <module>Modulo01</module>
</modules>

<dependencyManagement>
  <dependencies>
    <dependency>
      <groupId>junit</groupId>
      <artifactId>junit</artifactId>
      <version>3.8.1</version>
    </dependency>
  </dependencies>
</dependencyManagement>
```

Dependência no Modulo01

```
<dependencies>
  <dependency>
    <groupId>junit</groupId>
    <artifactId>junit</artifactId>
  </dependency>
</dependencies>
```

A versão é especificada
no *pom* pai

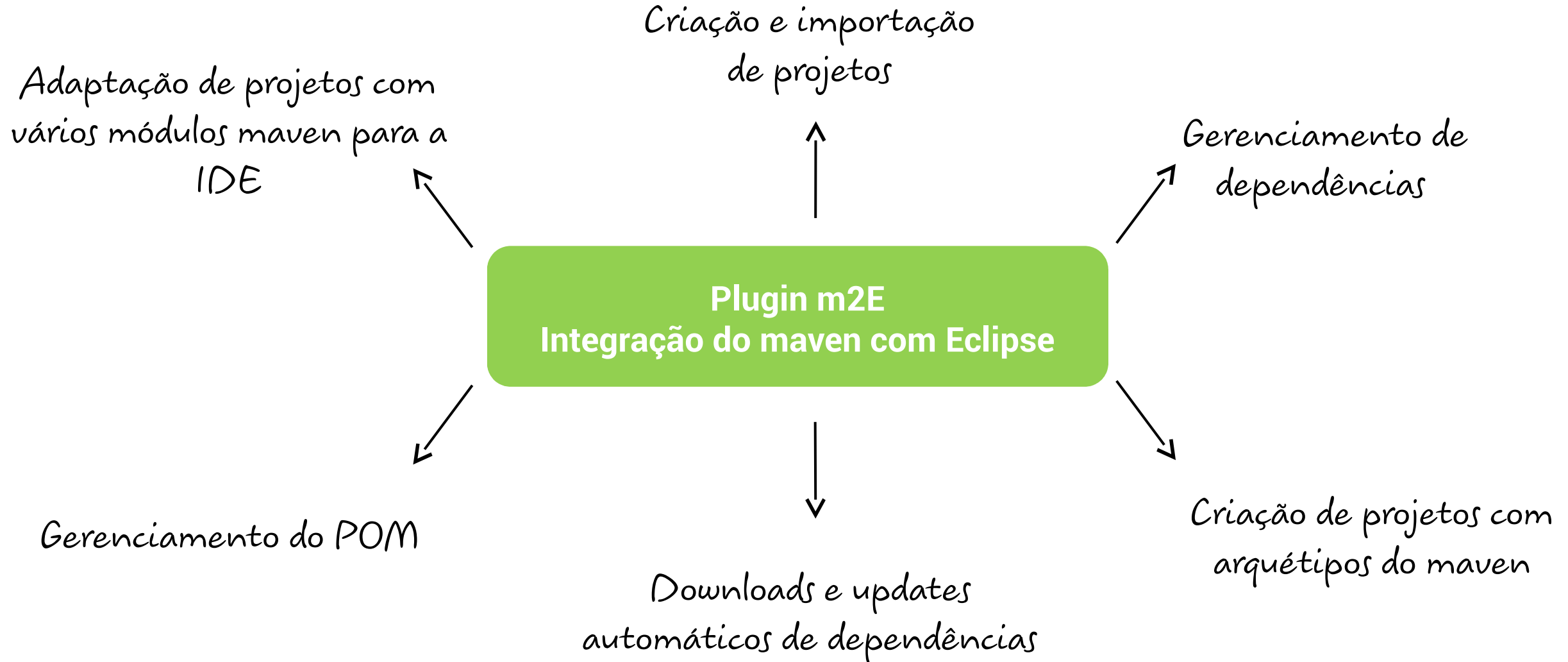
Permite melhor controle de versão de dependências em projetos multimódulos

Apache Maven Compiler Plugin

```
<build>
  <plugins>
    <plugin>
      <groupId>org.apache.maven.plugins</groupId>
      <artifactId>maven-compiler-plugin</artifactId>
      <version>3.3</version>
      <configuration>
        <source>1.8</source>
        <target>1.8</target>
      </configuration>
    </plugin>
  </plugins>
</build>
```

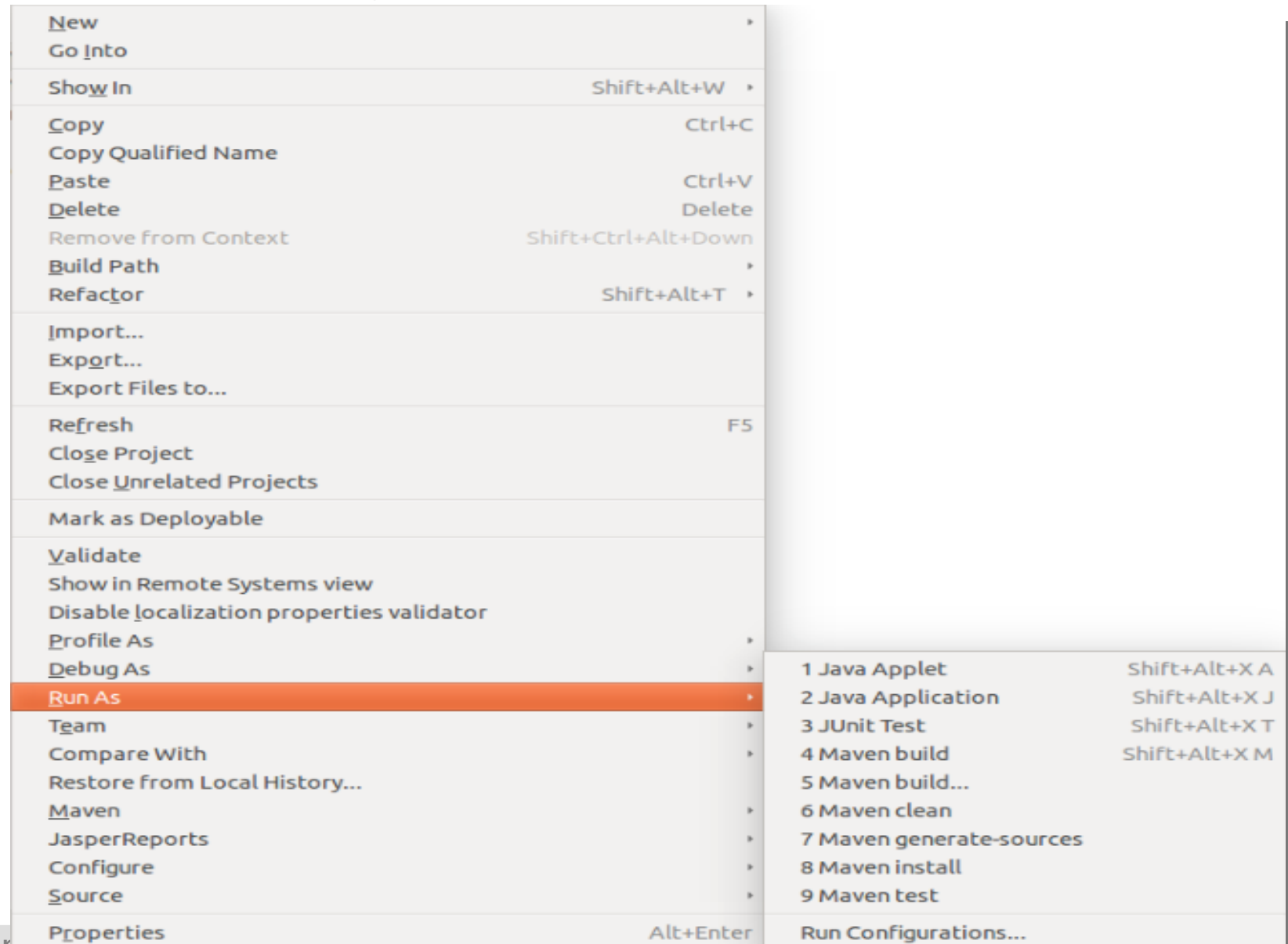
Indica qual versão da JDK o projeto será compilado.

Plugin m2E Integração do maven com Eclipse



Algumas versões do eclipse já vem com esse plugin!

Menu de ações utilizando o plugin m2E



Menu de ações utilizando o plugin m2E

