

Instalando 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

```
mvn archetype:generate  
-DgroupId={project-packaging}  
-DartifactId={project-name}  
-DarchetypeArtifactId={archetype}
```



Atributos Obrigatórios

Group Id

ID do grupo do projeto. Identificador único de uma organização.

Artifact ID

ID do projeto. Nome do projeto.

Version

Versão do projeto (gerada automaticamente na criação).


Criando projeto Desktop

maven-archetype-quickstart
cria um projeto maven de
exemplo.

```
mvn archetype:generate \  
-DgroupId=threeway.projeto \  
-DartifactId=ExemploMaven \  
-DarchetypeArtifactId=maven-archetype-quickstart \  
-DinteractiveMode=false
```



```
-----  
Using following parameters for creating project from Old (1.x) Archetype: maven-archetype-quickstart:1.0  
-----  
Parameter: groupId, Value: threeway.projeto  
Parameter: packageName, Value: threeway.projeto  
Parameter: package, Value: threeway.projeto  
Parameter: artifactId, Value: ExemploMaven  
Parameter: basedir, Value: /home/fabrica-jorge/TesteMaven  
Parameter: version, Value: 1.0-SNAPSHOT  
project created from Old (1.x) Archetype in dir: /home/fabrica-jorge/TesteMaven/ExemploMaven  
-----  
BUILD SUCCESS  
-----  
Total time: 19.448s  
Finished at: Tue Nov 24 10:29:45 BRST 2015  
Final Memory: 15M/149M
```

Criando o mesmo projeto utilizando o eclipse

 **New Maven Project**

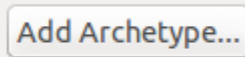
New Maven project

Select an Archetype


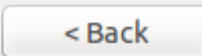
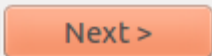
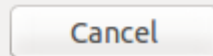
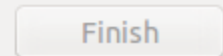
Catalog: All Catalogs  

Filter:

Group Id	Artifact Id	Version
org.apache.maven.archetypes	maven-archetype-mojo	RELEASE
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


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

Advanced

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
------	-------

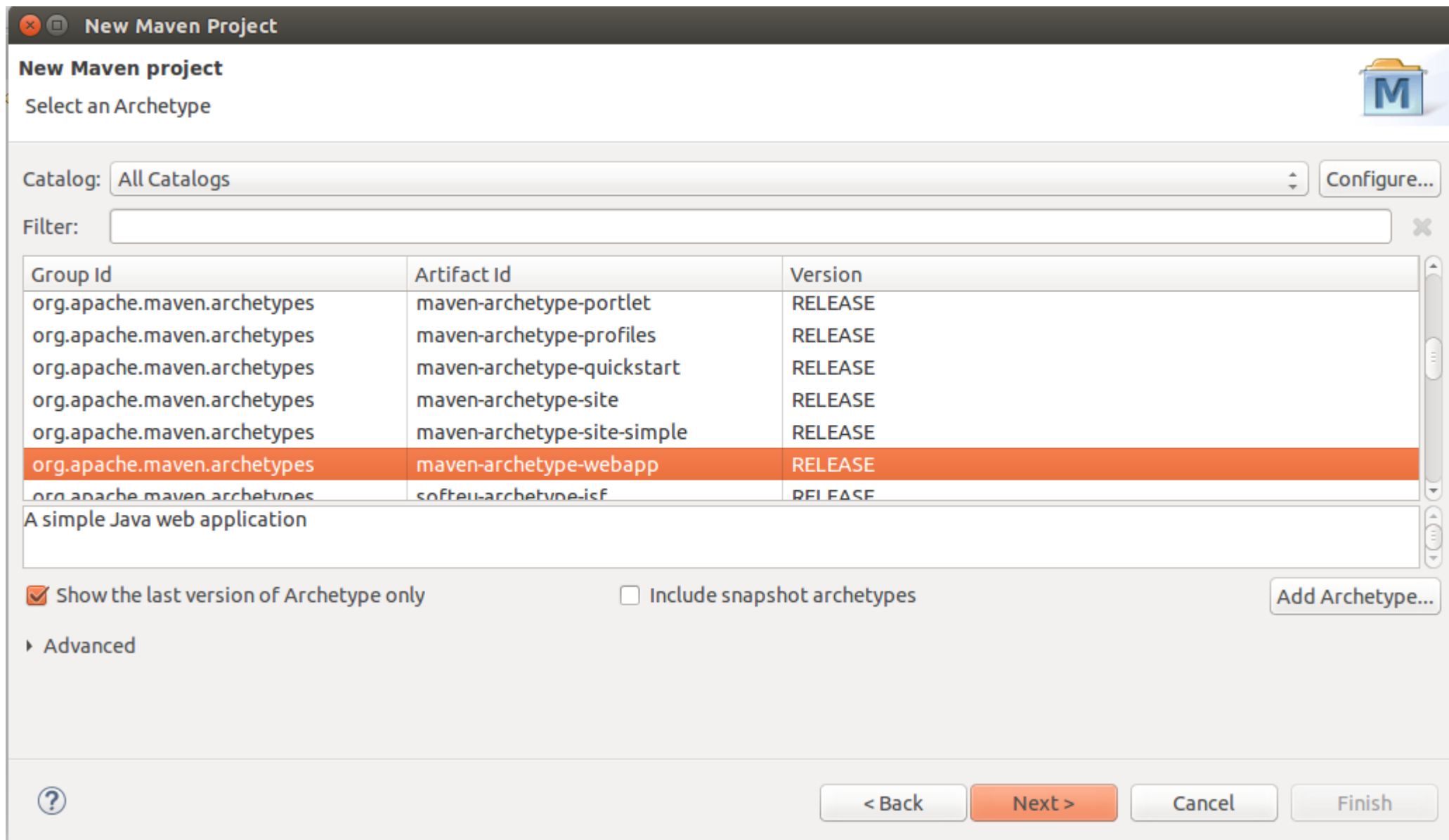
▸ Advanced

Criando projeto WEB


```
mvn archetype:generate \  
-DgroupId=threeway.projeto.mavenweb \  
-DartifactId=ExemploMavenWeb \  
-DarchetypeArtifactId=maven-archetype-webapp \  
-DinteractiveMode=false
```


```
-----  
Using following parameters for creating project from Old (1.x) Archetype: maven-archetype-webapp:1.0  
-----  
Parameter: groupId, Value: threeway.projeto.mavenweb  
Parameter: packageName, Value: threeway.projeto.mavenweb  
Parameter: package, Value: threeway.projeto.mavenweb  
Parameter: artifactId, Value: ExemploMavenWeb  
Parameter: basedir, Value: /home/fabrica-jorge/TesteMaven  
Parameter: version, Value: 1.0-SNAPSHOT  
project created from Old (1.x) Archetype in dir: /home/fabrica-jorge/TesteMaven/ExemploMavenWeb  
-----  
BUILD SUCCESS  
-----  
Total time: 9.661s  
Finished at: Tue Nov 24 10:43:30 BRST 2015  
Final Memory: 16M/217M  
-----
```

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



Tipos de projeto empacotamento

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

Empacotamento para projetos libs (padrão do maven)

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

Empacotamento para projetos modulares

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

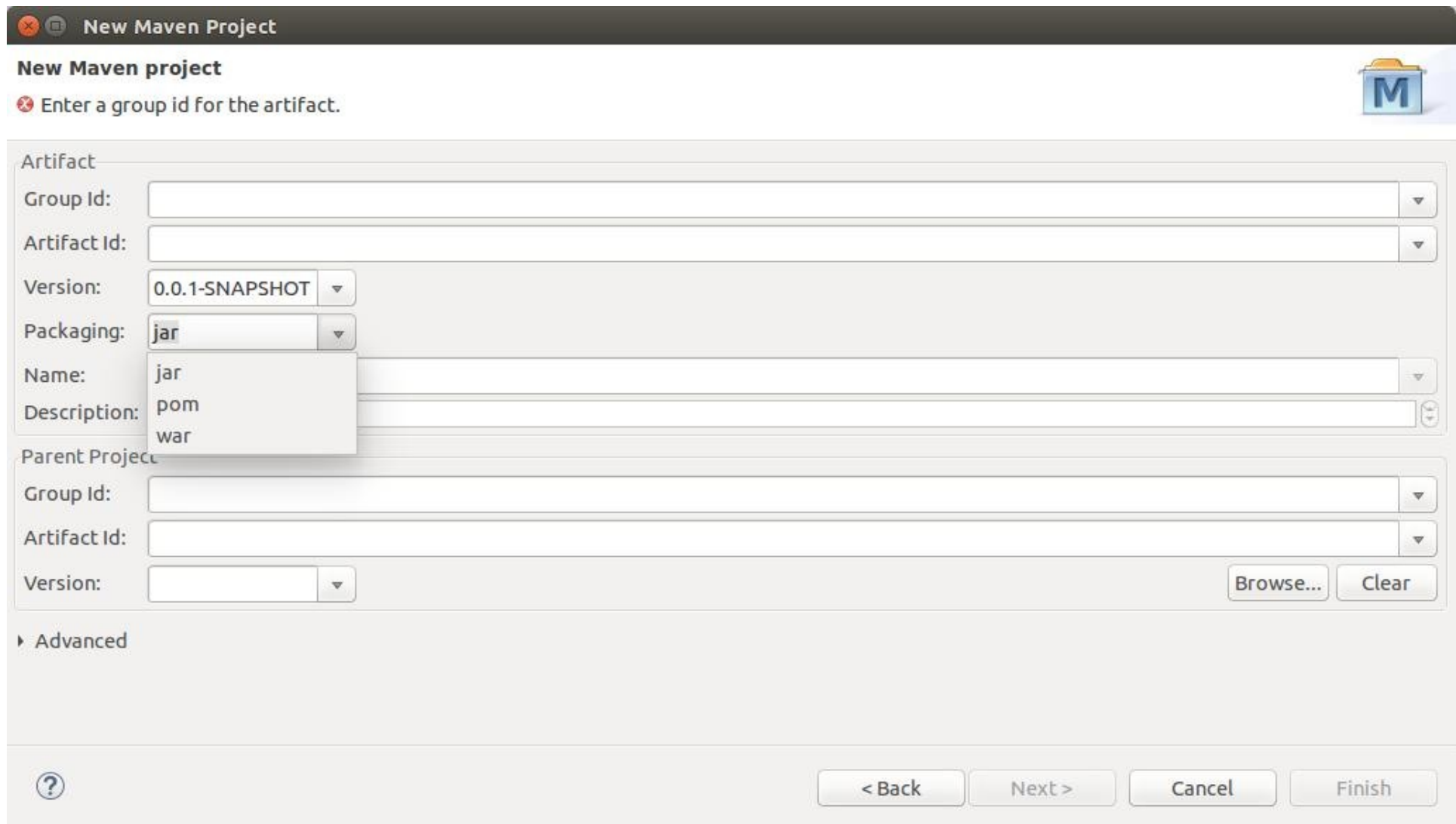
Empacotamento para projetos EJB

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

Empacotamento para projetos web

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: jar

Description: pom

war

Parent Project

Group Id:

Artifact Id:

Version: Browse... Clear

▶ Advanced

? < Back Next > Cancel Finish

Alvos Maven

No diretório do projeto, os seguintes alvos estão disponíveis:

mvn generate-sources

Gera qualquer código fonte necessário para a aplicação, geralmente feito usando algum plugin apropriado.

mvn compile

Compila o projeto e o código fonte.

mvn test-compile

Compila os testes unitários do projeto.

mvn test

Executa os testes unitários no diretório src/test.

mvn package

Empacota o código compilado no seu formato distribuível (JAR, WAR, etc).

mvn integration-test

Processa e implanta o pacote em um ambiente onde os teste de integração podem ser rodados.

mvn install

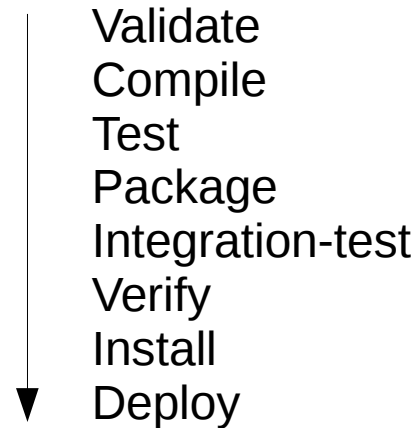
Instala o pacote no repositório local para ser usado como dependência em outros projetos na máquina local.

mvn deploy

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

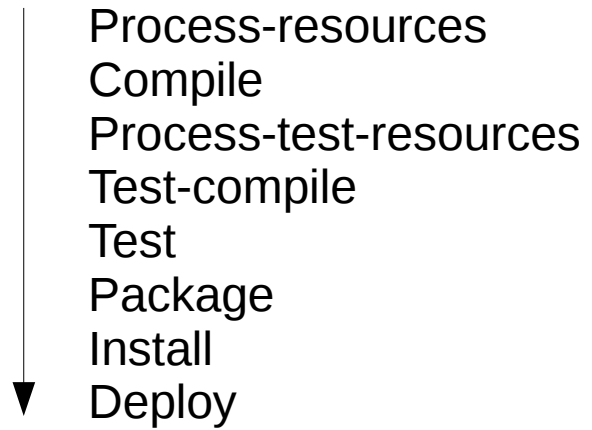
Ciclo de build padrão



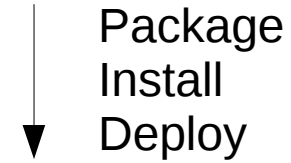
Executar ***mvn test***, por exemplo, fará com que todos as fases anteriores também sejam executadas.

Ciclo de Vida Maven

Empacotamento JAR

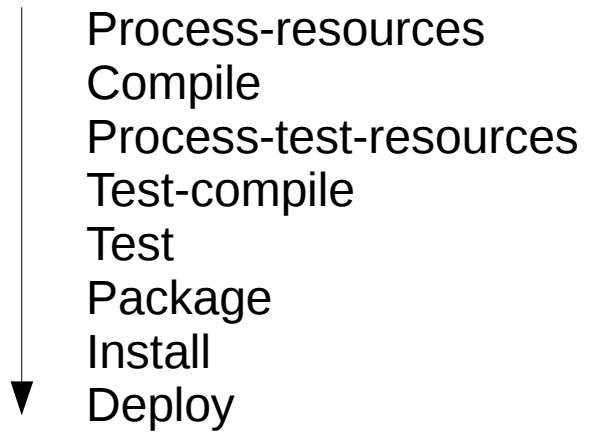


Empacotamento POM

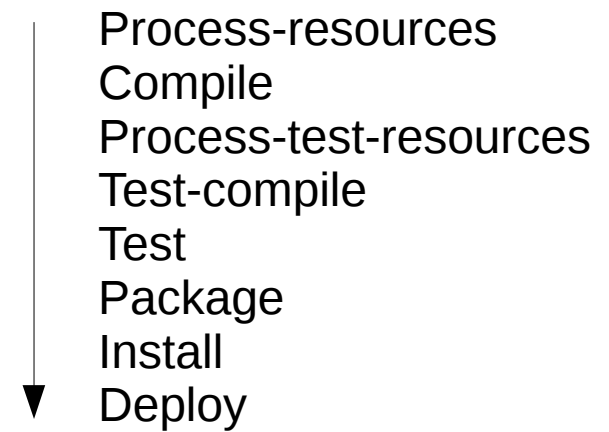


Ciclo de Vida Maven

Empacotamento EJB



Empacotamento WAR



Preparando o projeto para o Eclipse

Comando

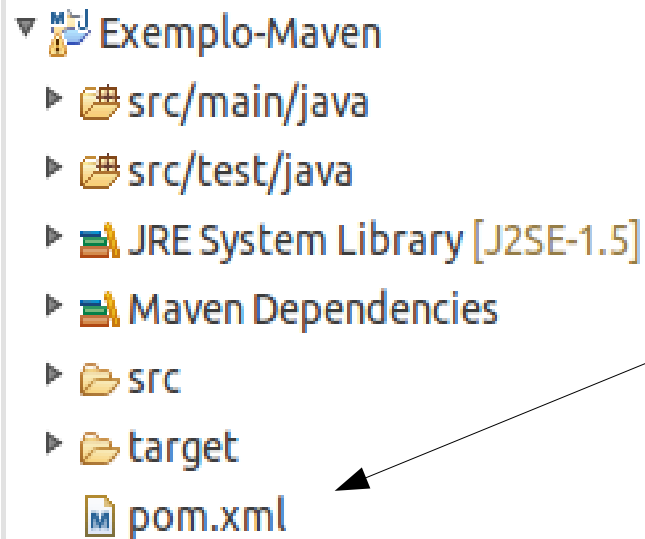
```
mvn eclipse:eclipse
```

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.

Declaração de dependências, estrutura e características de um projeto.

Representação xml de um projeto maven

pom.xml

```
graph TD; pomxml[pom.xml] --> desc1[Declaração de dependências, estrutura e características de um projeto.]; pomxml --> desc2[Representação xml de um projeto maven]; pomxml --> desc3[Maven utiliza o POM efetivo (mvn help:effective-pom)]; pomxml --> desc4[Todos os arquivos POM estendem de um super POM];
```

Todos os arquivos POM estendem de um super POM

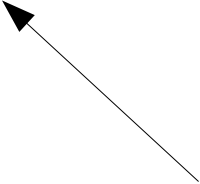
Maven utiliza o POM efetivo
(***mvn help:effective-pom***)

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.).

Mais informações sobre o POM
<https://maven.apache.org/pom.html>

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>

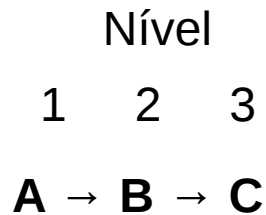
  <!-- Configurações de Build -->
  <build> </build>
  <reporting> </reporting>

  <!-- 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



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.


Dependência Opcional

Alternativa para exclusão de dependências

Projeto A depende de B

```
<dependencies>  
  <dependency>  
    <groupId>threeway.opcional</groupId>  
    <artifactId>ProjetoB</artifactId>  
    <version>0.0.1-SNAPSHOT</version>  
    <optional>>false</optional>  
  </dependency>  
</dependencies>
```

Tag *<optional>*



Projeto B depende de C

```
<dependencies>  
  <dependency>  
    <groupId>threeway.opcional</groupId>  
    <artifactId>ProjetoC</artifactId>  
    <version>0.0.1-SNAPSHOT</version>  
    <optional>>true</optional>  
  </dependency>  
</dependencies>
```

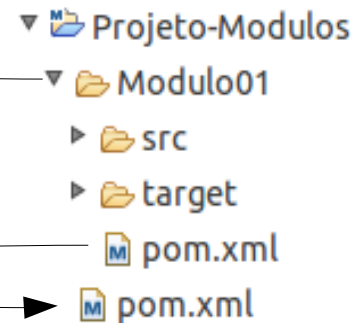
Impede que as dependências de C se propaguem transitivamente para projetos dependentes de B (projeto A)

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

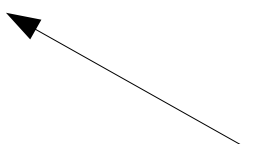
```
<modules>
  <module>Modulo01</module>
</modules>
```

```
<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>
```

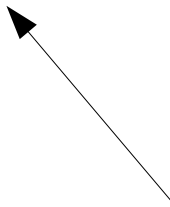
A versão é especificada
no ***pom*** pai



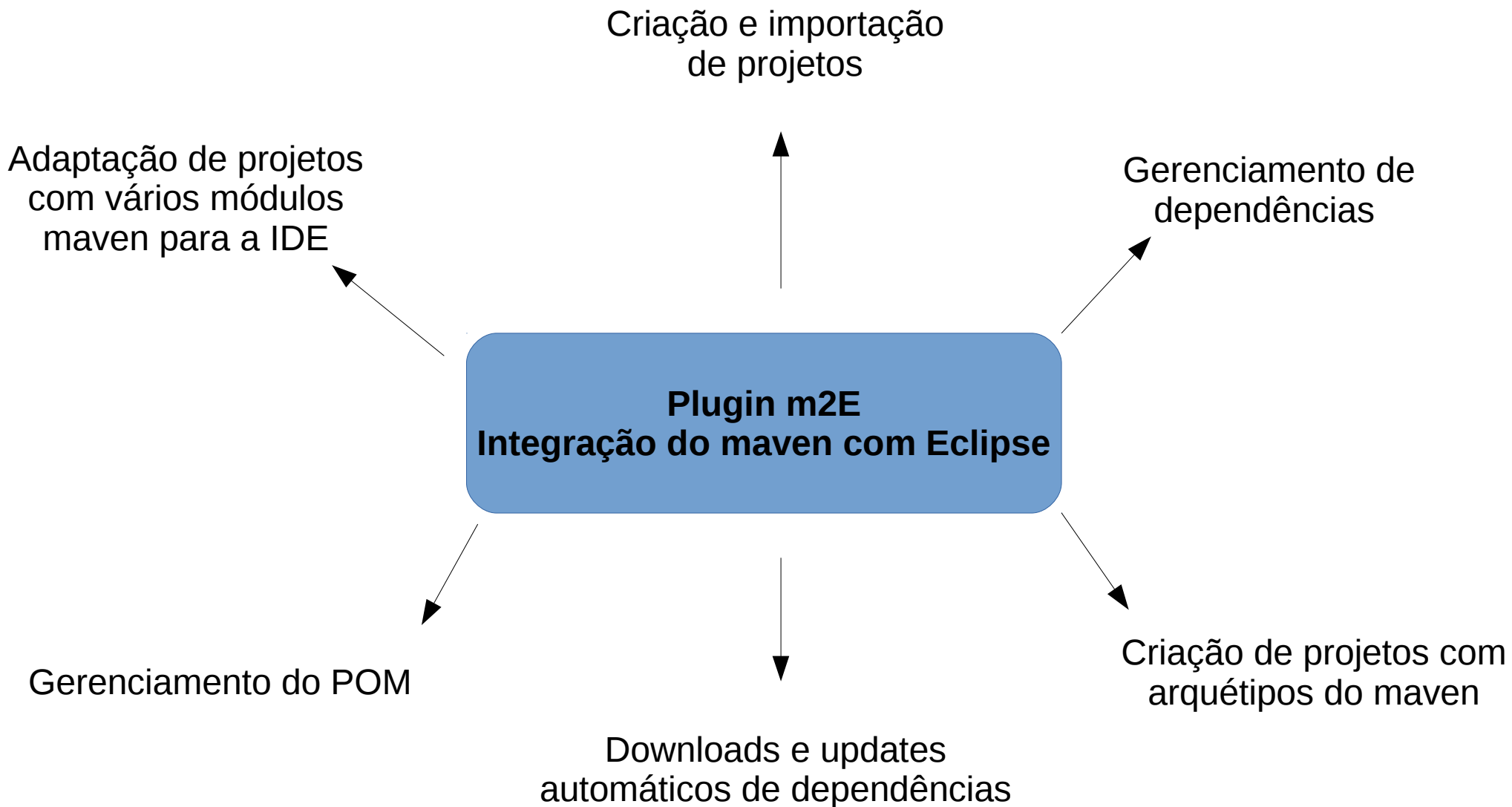
Permite melhor controle de versão de dependências
em projetos multimodulos

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.



Algumas versões do eclipse já vem com esse plugin!

Menu de ações utilizando o plugin m2E

<u>N</u> ew		
Go <u>I</u> nto		
Sho <u>w</u> In	Shift+Alt+W	
<u>C</u> opy	Ctrl+C	
Copy Qualified Name		
<u>P</u> aste	Ctrl+V	
<u>D</u> elete	Delete	
Remove from Context	Shift+Ctrl+Alt+Down	
<u>B</u> uild Path		
Refa <u>c</u> tor	Shift+Alt+T	
<u>I</u> mport...		
Exp <u>o</u> rt...		
Export Files to...		
Re <u>fr</u> esh	F5	
Clo <u>s</u> e Project		
Clo <u>s</u> e <u>U</u> nrelated Projects		
Mark as Deployable		
<u>V</u> alidate		
Show in Remote Systems view		
Disa <u>b</u> le <u>l</u> ocalization properties validator		
<u>P</u> rofile As		
<u>D</u> ebug As		
<u>R</u> un As		
<u>T</u> eam		
Compare With		
Restore from Local History...		
<u>M</u> aven		
JasperReports		
Configure		
<u>S</u> ource		
Properties	Alt+Enter	
		1 Java Applet Shift+Alt+X A
		2 Java Application Shift+Alt+X J
		3 JUnit Test Shift+Alt+X T
		4 Maven build Shift+Alt+X M
		5 Maven build...
		6 Maven clean
		7 Maven generate-sources
		8 Maven install
		9 Maven test
		Run Configurations...

Menu de ações utilizando o plugin m2E

