



Auditoria e Qualidade de Sistemas

Prof. Edgard Davidson C. Cardoso



Qualidade de Sistemas

BUILD AUTOMATIZADO

Build Atomatizado

APACHE ANT





Ferramentas de Build

- A ideia de automatizar tarefas rotineiras através de scripts não é nova em ambientes Unix.
 - por exemplo, nos ambientes Unix temos o **make** ou **gnumake**.
- Para o .NET existe o **Nant** e o **MSBuild**
- Para o Ruby temos o **Ruby Rake**
- Para desenvolvimento Java temos o Apache **Ant** e **Maven**.



O Que é Ant?

- Ant é uma ferramenta utilizada para automatizar a construção de software.
- Ele é similar ao make mas é escrito na linguagem Java e foi desenvolvida inicialmente para ser utilizada em projetos desta linguagem.



Build Atomatizado

MAKE



Como funciona o Make

- O comando make faz uma leitura de um arquivo chamado Makefile

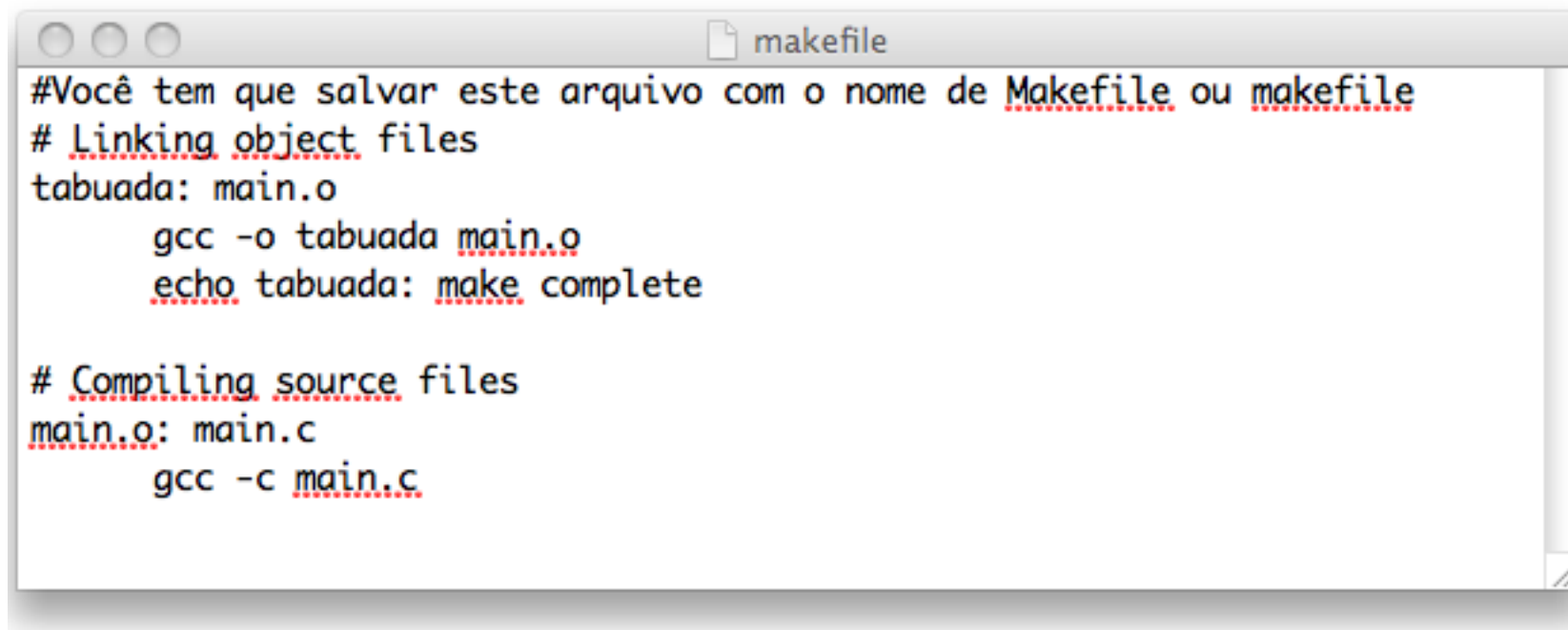


Como funciona o Make

```
/*irei fazer um simples programa em C para demonstrar como funciona*/  
#include <stdio.h>
```

```
int main (int argc, const char * argv[]) {  
    int num1;  
    printf("Este programa mostra a tabuada.\n");  
    printf("Você quer saber a tabuada de que numero:\n");  
    printf("#: ");  
    scanf("%d",&num1);  
    int i=0;  
    for(i=0;i<11;i++){  
        printf("RESULTADO: %d\n",i*num1);  
    }//for  
    getchar();  
    return 0;  
}  
|
```


Como funciona o Make



```
#Você tem que salvar este arquivo com o nome de Makefile ou makefile
# Linking object files
tabuada: main.o
    gcc -o tabuada main.o
    echo tabuada: make complete

# Compiling source files
main.o: main.c
    gcc -c main.c
```



PRONTO!!! O que você precisa fazer agora é só dar o make na linha de comando.

\$ make



Como funciona o Make

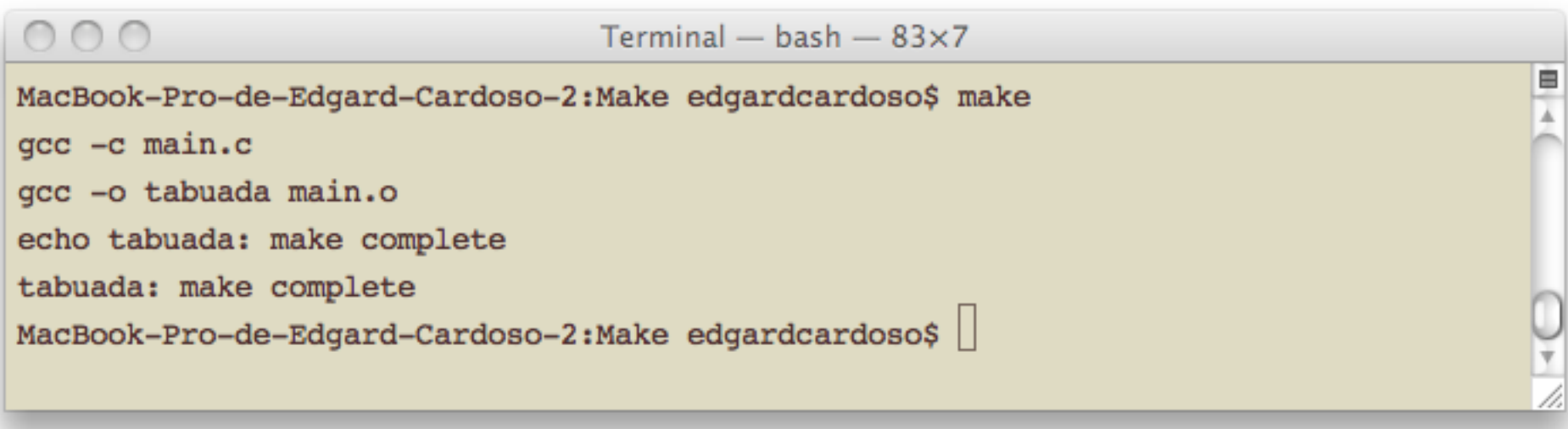
- Arquivos no disco (Finder)

Nome	Data de Modific.	Tamanho	Tipo
 main.c	Hoje, 17:56	4 KB	C Source File
 makefile	Hoje, 17:56	4 KB	Documento

- Arquivos no disco (linha de comando)





```
Terminal — bash — 71x5
MacBook-Pro-de-Edgard-Cardoso-2:Make edgardcardoso$ ls
main.c          makefile
MacBook-Pro-de-Edgard-Cardoso-2:Make edgardcardoso$
```

Como funciona o Make



A terminal window titled "Terminal — bash — 83x7" showing the execution of the 'make' command. The prompt is "MacBook-Pro-de-Edgard-Cardoso-2:Make edgardcardoso\$". The output shows the compilation of 'main.c' into 'main.o' using 'gcc -c', followed by the linking of 'main.o' into 'tabuada' using 'gcc -o tabuada main.o'. The command 'echo tabuada: make complete' is also executed, resulting in the output 'tabuada: make complete'. The prompt returns to "MacBook-Pro-de-Edgard-Cardoso-2:Make edgardcardoso\$".

```
MacBook-Pro-de-Edgard-Cardoso-2:Make edgardcardoso$ make
gcc -c main.c
gcc -o tabuada main.o
echo tabuada: make complete
tabuada: make complete
MacBook-Pro-de-Edgard-Cardoso-2:Make edgardcardoso$
```

Nome	Data de Modific.	Tamanho	Tipo
 main.c	Hoje, 18:30	4 KB	C Source File
 main.o	Hoje, 18:35	4 KB	Documento
 makefile	Hoje, 18:29	4 KB	Documento
 tabuada	Hoje, 18:35	12 KB	Arquivo Executável Unix



Build Atomatizado

APACHE ***ma**ven*



Problemas Comuns

- No seu projeto
 - Existem vários JARs
 - Problemas com dependências e versões
 - Estrutura do projeto
 - Building, publishing e deploying



O que é ***maven***

- Apache Maven é um software gerenciador de dependências.
- Baseado do conceito de um **Project Object Model (POM)** o Maven pode gerenciar o build de um projeto reportando e documentando a partir de um local central de informações



O que é **maven**

- Maven, uma palavra hebraica que significa acumulador de conhecimentos, foi originalmente começou como uma tentativa de simplificar os processos de construção do projeto Jakarta Turbine.
- Havia vários projetos, cada um com sua própria Ant construir arquivos que estavam todos um pouco diferente e JARs foram verificados em CVS.
- Era necessário uma forma padrão para construir os projetos, uma definição clara do que o projeto consistia em, uma maneira fácil de publicar informações sobre o projeto e uma maneira de compartilhar JARs em vários projetos.
- O resultado é uma ferramenta que pode agora ser usado para a construção e gerenciamento de qualquer projeto baseado em Java.



Objetivo do **maven**

- Principal objetivo do Maven é permitir que um desenvolvedor de compreender o estado completo de um esforço de desenvolvimento no mais curto espaço de tempo. Para atingir este objetivo, há várias áreas de preocupação que Maven tenta lidar com:
 - Tornar o processo de construção fácil
 - Fornecer um sistema de construção uniforme
 - Fornecer informações sobre o projeto de qualidade
 - Fornecer orientações para desenvolvimento de melhores práticas
 - Permitir a migração transparente para novas funcionalidades



maven

CONVENTION

over

CONFIGURATION

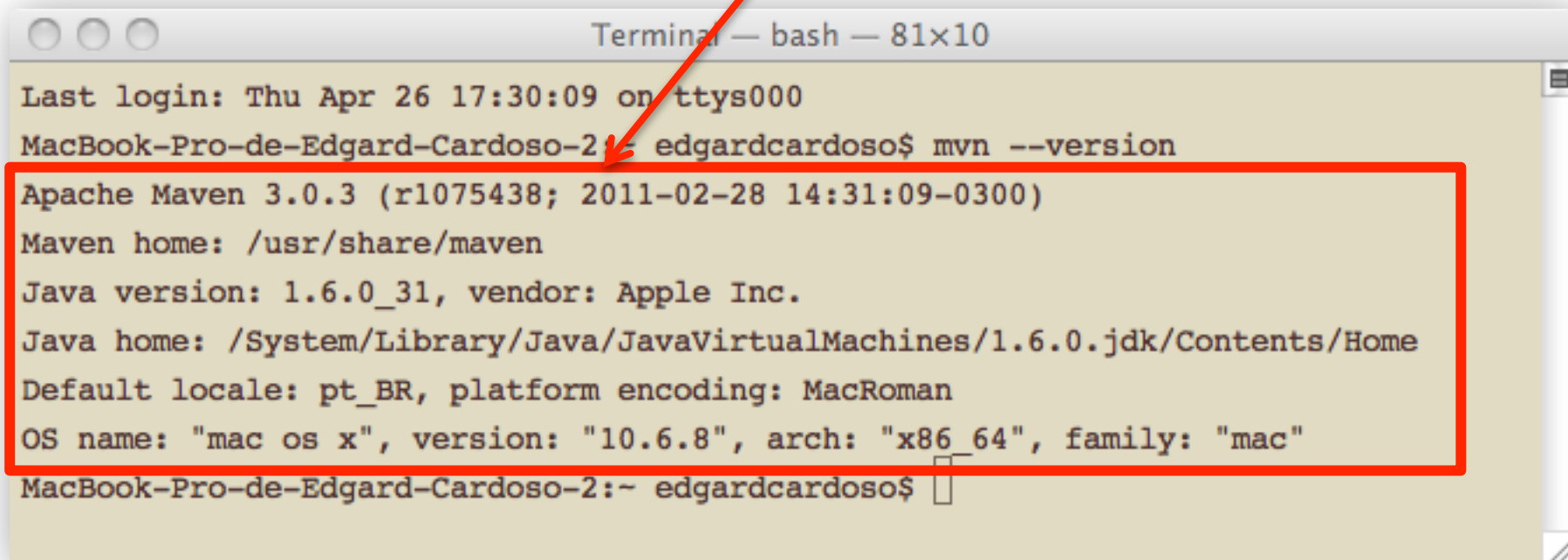


Verifique se o Maven já está na sua Máquina

- No terminal (MAC/Linux) ou no prompt (Windows) digite o comando:

`mvn --version`

Já instalado

A screenshot of a macOS Terminal window titled "Terminal — bash — 81x10". The window shows the output of the command "mvn --version". The output text is: "Last login: Thu Apr 26 17:30:09 on ttys000", "MacBook-Pro-de-Edgard-Cardoso-2:~ edgardcardoso\$ mvn --version", "Apache Maven 3.0.3 (r1075438; 2011-02-28 14:31:09-0300)", "Maven home: /usr/share/maven", "Java version: 1.6.0_31, vendor: Apple Inc.", "Java home: /System/Library/Java/JavaVirtualMachines/1.6.0.jdk/Contents/Home", "Default locale: pt_BR, platform encoding: MacRoman", "OS name: \"mac os x\", version: \"10.6.8\", arch: \"x86_64\", family: \"mac\"", and "MacBook-Pro-de-Edgard-Cardoso-2:~ edgardcardoso\$". A red box highlights the output from "Apache Maven 3.0.3" to the end of the OS information line. A red arrow points from the text "Já instalado" to the red box.

```
Terminal — bash — 81x10
Last login: Thu Apr 26 17:30:09 on ttys000
MacBook-Pro-de-Edgard-Cardoso-2:~ edgardcardoso$ mvn --version
Apache Maven 3.0.3 (r1075438; 2011-02-28 14:31:09-0300)
Maven home: /usr/share/maven
Java version: 1.6.0_31, vendor: Apple Inc.
Java home: /System/Library/Java/JavaVirtualMachines/1.6.0.jdk/Contents/Home
Default locale: pt_BR, platform encoding: MacRoman
OS name: "mac os x", version: "10.6.8", arch: "x86_64", family: "mac"
MacBook-Pro-de-Edgard-Cardoso-2:~ edgardcardoso$
```

Instalando

The screenshot shows the Apache Maven Project download page. A red arrow points from the text 'Fazer Download' to the 'bin.tar.gz' link in the download table.

Download Maven 3.0.4

Maven is distributed in several formats for your convenience. Use a source archive if you intend to build Maven yourself. Otherwise, simply pick a ready-made binary distribution and follow the installation instructions given at the end of this document.

You will be prompted for a mirror - if the file is not found on yours, please be patient, as it may take 24 hours to reach all mirrors.

In order to guard against corrupted downloads/installations, it is highly recommended to [verify the signature](#) of the release bundles against the public [KEYS](#) used by the Apache Maven developers.

Maven is distributed under the [Apache License, version 2.0](#).

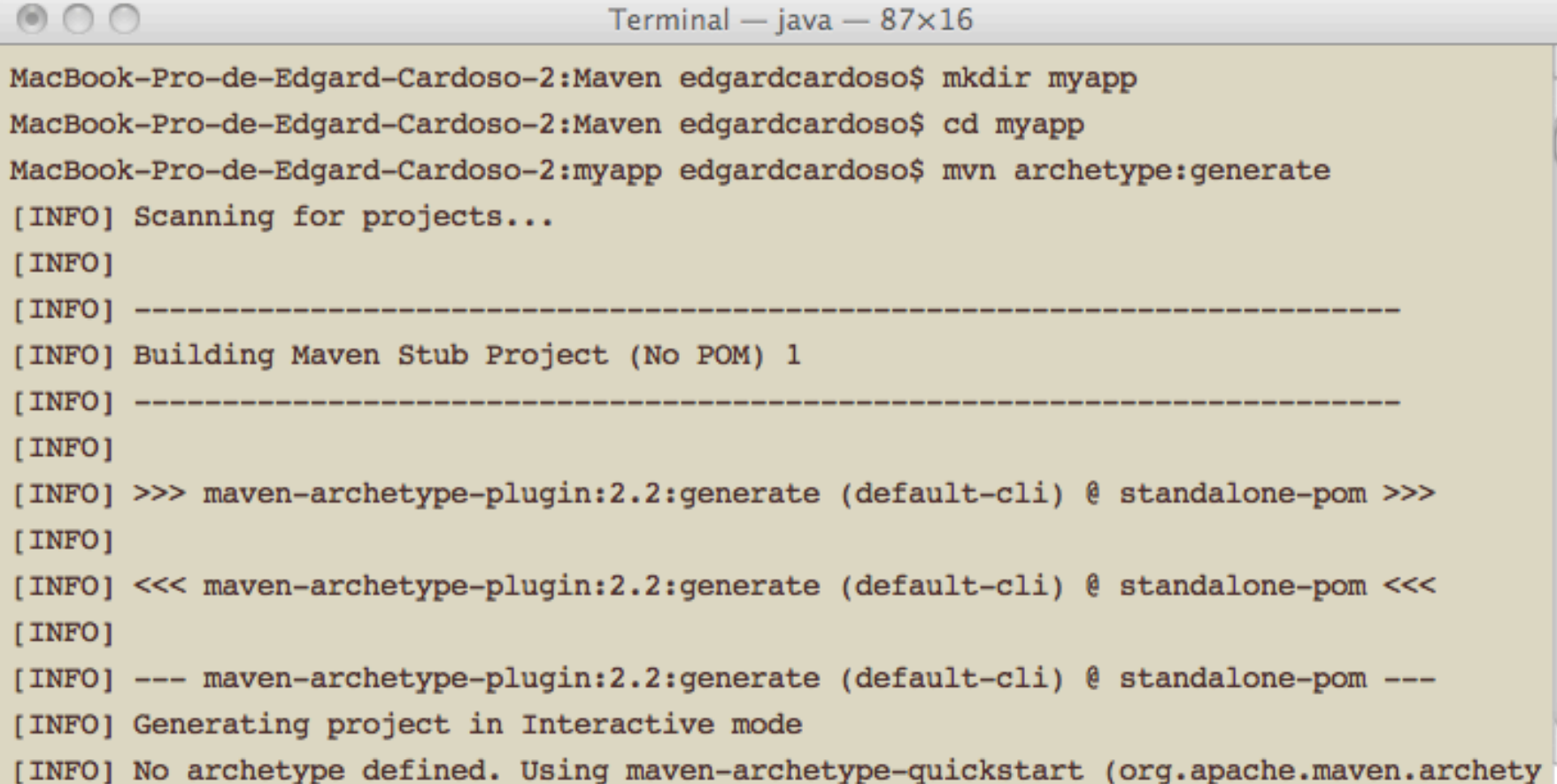
We **strongly** encourage our users to configure a Maven repository mirror closer to their location, please read [How to Use Mirrors for Repositories](#).

Be sure to check the [compatibility notes](#) before using this version to avoid surprises. While Maven 3 aims to be backward-compatible with Maven 2.x to the extent possible, there are still a few significant changes.

	Mirrors	Checksum	Signature
Maven 3.0.4 (Binary tar.gz)	apache-maven-3.0.4-bin.tar.gz	apache-maven-3.0.4-bin.tar.gz.md5	apache-maven-3.0.4-bin.tar.gz.asc
Maven 3.0.4 (Binary zip)	apache-maven-3.0.4-bin.zip	apache-maven-3.0.4-bin.zip.md5	apache-maven-3.0.4-bin.zip.asc
Maven 3.0.4 (Source tar.gz)	apache-maven-3.0.4-src.tar.gz	apache-maven-3.0.4-src.tar.gz.md5	apache-maven-3.0.4-src.tar.gz.asc
Maven 3.0.4 (Source zip)	apache-maven-3.0.4-src.zip	apache-maven-3.0.4-src.zip.md5	apache-maven-3.0.4-src.zip.asc
Release Notes	3.0.4		

Criando um projeto via Maven

- Digite o comando:
`mvn archetype:generate`

A screenshot of a macOS Terminal window titled "Terminal — java — 87x16". The window shows the execution of the Maven command `mvn archetype:generate` in a directory named `myapp`. The output includes status messages like "[INFO] Scanning for projects..." and "[INFO] Building Maven Stub Project (No POM) 1". It also shows the Maven archetype plugin version `2.2` and the mode `Interactive mode`. The final line indicates that no archetype was defined, so it will use the `maven-archetype-quickstart`.

```
MacBook-Pro-de-Edgard-Cardoso-2:Maven edgardcardoso$ mkdir myapp
MacBook-Pro-de-Edgard-Cardoso-2:Maven edgardcardoso$ cd myapp
MacBook-Pro-de-Edgard-Cardoso-2:myapp edgardcardoso$ mvn archetype:generate
[INFO] Scanning for projects...
[INFO]
[INFO] -----
[INFO] Building Maven Stub Project (No POM) 1
[INFO] -----
[INFO]
[INFO] >>> maven-archetype-plugin:2.2:generate (default-cli) @ standalone-pom >>>
[INFO]
[INFO] <<< maven-archetype-plugin:2.2:generate (default-cli) @ standalone-pom <<<
[INFO]
[INFO] --- maven-archetype-plugin:2.2:generate (default-cli) @ standalone-pom ---
[INFO] Generating project in Interactive mode
[INFO] No archetype defined. Using maven-archetype-quickstart (org.apache.maven.archety
```



Criando um projeto via Maven (cont...)

```
Terminal — java — 87x16

ject that combines Scala and Wicket,
    depending on the Wicket-Scala project. Includes an example Specs
    test.)

562: remote -> org.wikbook:wikbook.archetype (-)
563: remote -> org.xaloon.archetype:xaloon-archetype-wicket-jpa-glassfish (-)
564: remote -> org.xaloon.archetype:xaloon-archetype-wicket-jpa-spring (-)
565: remote -> org.xwiki.commons:xwiki-commons-component-archetype (Make it easy to cre
ate a maven project for creating XWiki Components.)
566: remote -> org.xwiki.rendering:xwiki-rendering-archetype-macro (Make it easy to cre
ate a maven project for creating XWiki Rendering Macros.)
567: remote -> org.zkoss:zk-archetype-component (The ZK Component archetype)
568: remote -> org.zkoss:zk-archetype-webapp (The ZK wepapp archetype)
569: remote -> ru.circumflex:circumflex-archetype (-)
570: remote -> se.vgregion.javg.maven.archetypes:javg-minimal-archetype (-)
Choose a number or apply filter (format: [groupId:]artifactId, case sensitive contains)
: 187: 
```

- Choose a number or apply filter (format: [groupId:]artifactId, case sensitive contains)
- 187: remote -> org.apache.maven.archetypes:maven-archetype-quickstart (An archetype which contains a sample Maven project.)



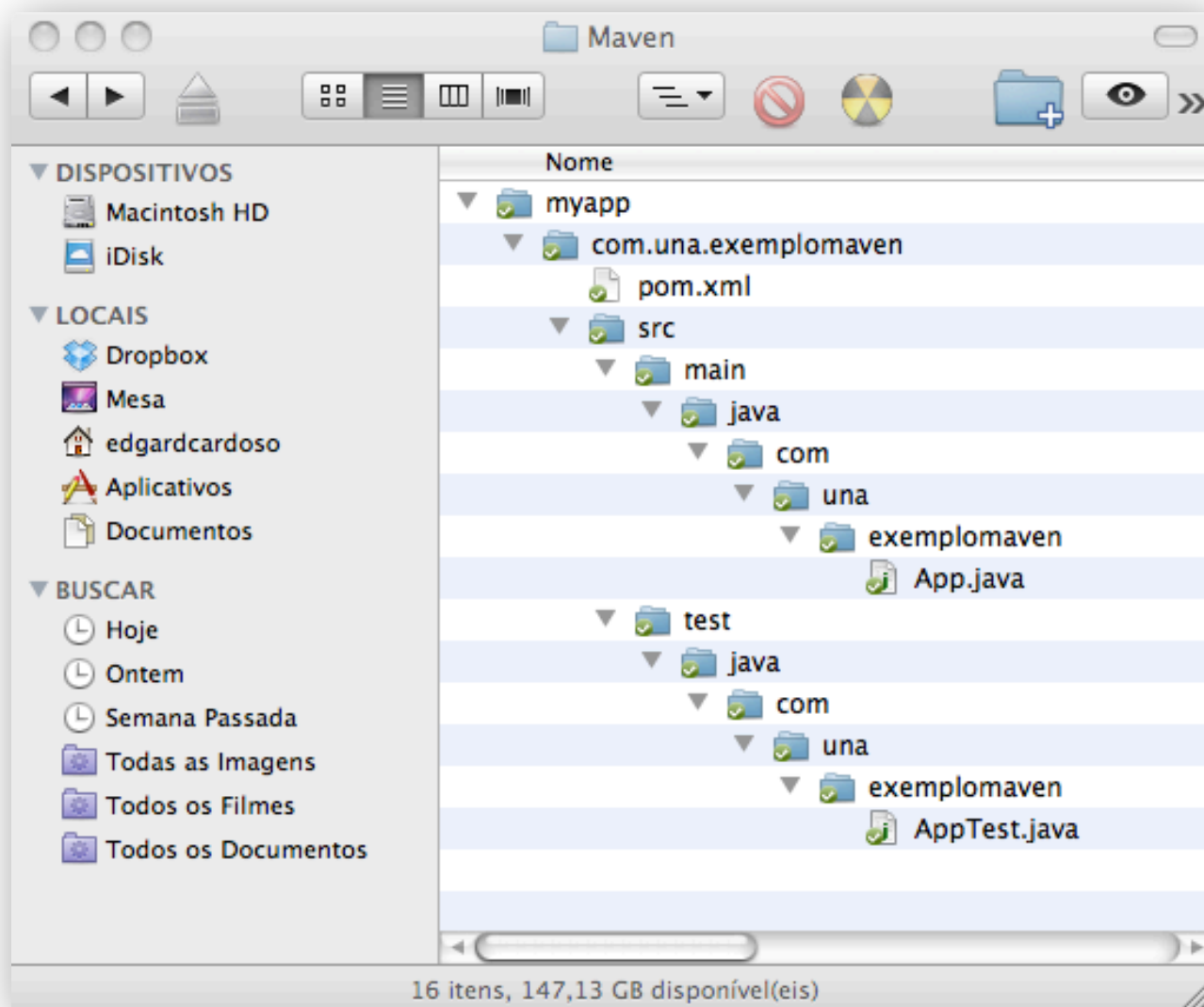
Criando um projeto via Maven (cont...)

```
Terminal — java — 87x16
566: remote -> org.xwiki.rendering:xwiki-rendering-archetype-macro (Make it easy to cre
ate a maven project for creating XWiki Rendering Macros.)
567: remote -> org.zkoss:zk-archetype-component (The ZK Component archetype)
568: remote -> org.zkoss:zk-archetype-webapp (The ZK wepapp archetype)
569: remote -> ru.circumflex:circumflex-archetype (-)
570: remote -> se.vgregion.javg.maven.archetypes:javg-minimal-archetype (-)
Choose a number or apply filter (format: [groupId:]artifactId, case sensitive contains)
: 187:
Choose org.apache.maven.archetypes:maven-archetype-quickstart version:
1: 1.0-alpha-1
2: 1.0-alpha-2
3: 1.0-alpha-3
4: 1.0-alpha-4
5: 1.0
6: 1.1
Choose a number: 6: 
```

```
Terminal — java — 87x11
Choose a number: 6: 6
Define value for property 'groupId': : com.una.exemplomaven
Define value for property 'artifactId': : com.una.exemplomaven
Define value for property 'version': 1.0-SNAPSHOT: : 1.0.0
Define value for property 'package': com.una.exemplomaven: :
Confirm properties configuration:
groupId: com.una.exemplomaven
artifactId: com.una.exemplomaven
version: 1.0.0
package: com.una.exemplomaven
Y: : y 
```

```
Terminal — bash — 87x8
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 1:36.182s
[INFO] Finished at: Thu Apr 26 17:53:48 BRT 2012
[INFO] Final Memory: 7M/81M
[INFO] -----
MacBook-Pro-de-Edgard-Cardoso-2:myapp edgardcardoso$ 
```


Projeto Maven Criado



Arquivo pom.xml

Archetype
Identificador
único do seu projeto

```
<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.una.exemplomaven</groupId>
  <artifactId>com.una.exemplomaven</artifactId>
  <version>1.0.0</version>
  <packaging>jar</packaging>

  <name>com.una.exemplomaven</name>
  <url>http://maven.apache.org</url>

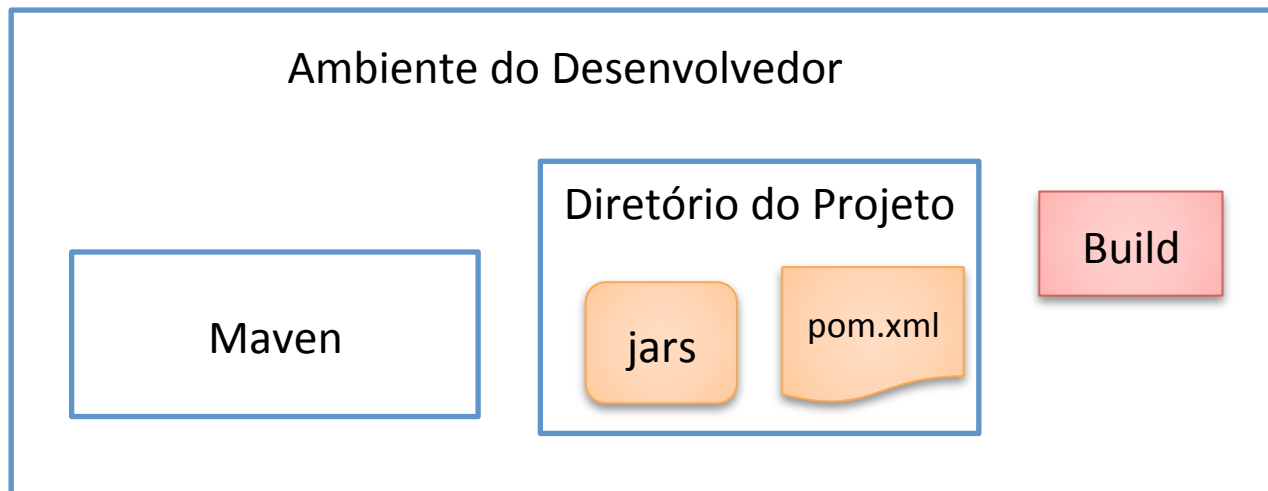
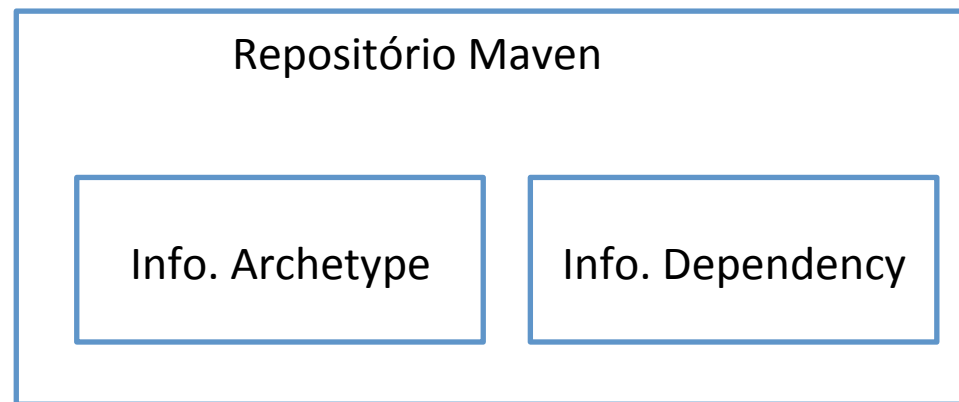
  <properties>
    <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
  </properties>

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

Dependências do
seu projeto



Maven



Criando o Projeto Maven no NetBeans

1. Escolha o projeto

Novo projeto

Escolha o projeto

Categorias:

- Java
- Java Web
- Java EE
- Maven
- Módulos do NetBeans
- Exemplos

Projetos:

- Aplicativo Java
- Aplicação Web
- Módulo EJB
- Aplicativo corporativo
- Pacote OSGi
- Módulo do NetBeans
- Suíte de módulos NetBeans
- Aplicativo NetBeans
- Projeto POM
- Projeto do arquétipo
- Projeto com POM existente

Descrição:
Aplicativo Java SE simples com Maven.

Ajuda < Voltar Próximo >

2. Nome e localização

Novo projeto

Nome e localização

Nome do projeto:

Localização do Projeto: Procurar...

Diretório do projeto:

Id de artefato:

Id de grupo:

Versão:

Pacote: (Opcional)

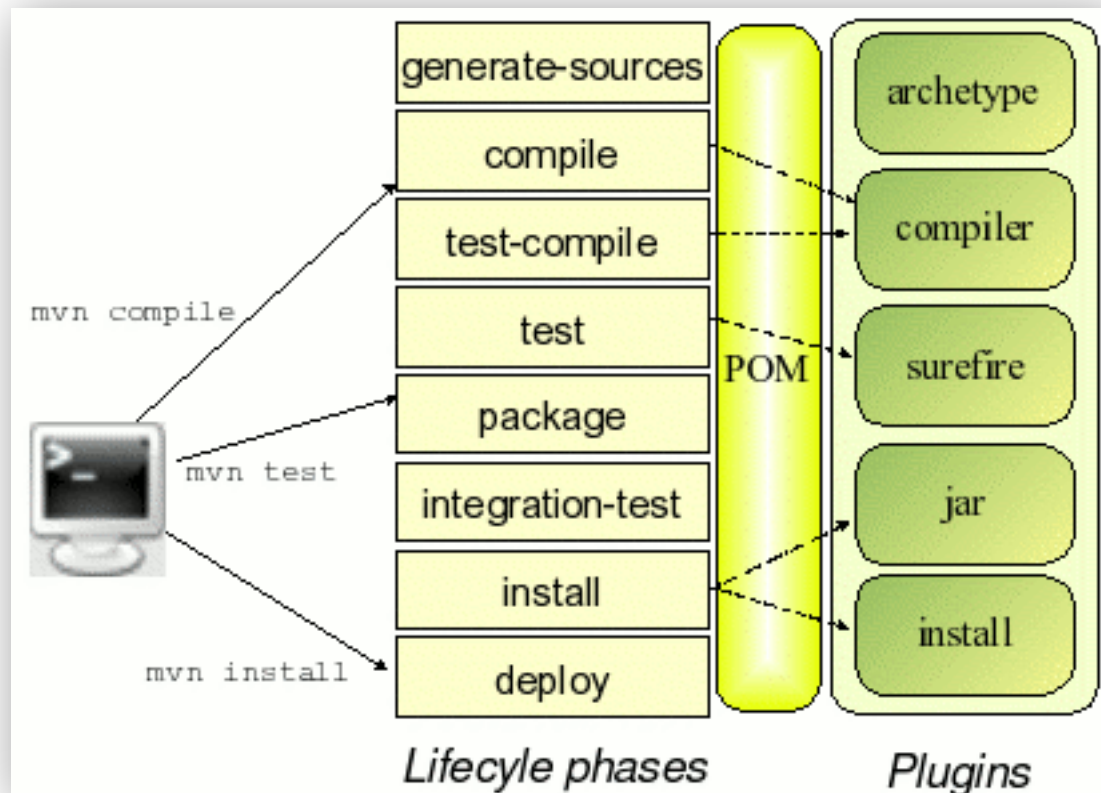
Ajuda < Voltar Próximo > Finalizar Cancelar

3. Estrutura do projeto

exemplomaven

- Pacotes de código-fonte
 - com.una.exemplomaven.exemplomaven
 - App.java
- Pacotes de testes
 - com.una.exemplomaven.exemplomaven
- Dependências
- Testar dependências
 - junit-3.8.1.jar [test]
- Arquivos do projeto
 - pom.xml

Ciclo de vida do projeto



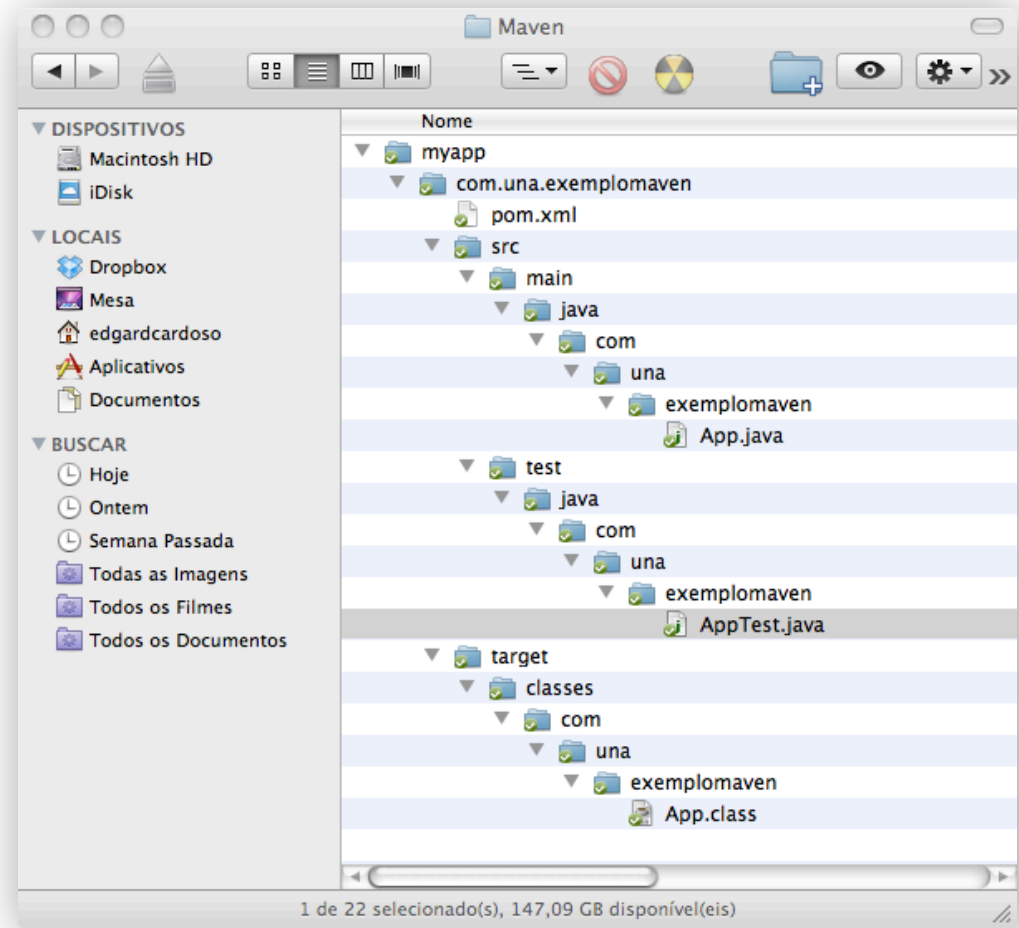
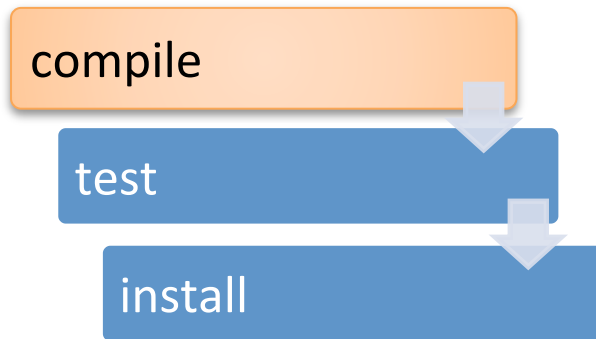


Ciclo de vida do Projeto

- **generate-source:** Gera código fonte, aplicando necessidade adicional, que normalmente é feito pelo plug-ins adequado .
- **compile:** Compila o código fonte do projeto
- **test-compile:** Compila o projeto teste de unidade
- **test:** Executar testes de unidade (normalmente com JUnit) em src / teste
- **package:** moldou o código compilado em seu formato de distribuição (JAR, WAR, etc.)
- **integration-test:** Executar e implanta o pacote se necessário em um ambiente onde os testes de integração pode ser realizada.
- **install:** Instala o produto no armazém local, para uso como uma dependência de outros projetos em sua máquina local.
- **deploy:** Realizado em um ambiente de integração ou de produção, copiar o produto final em um depósito remoto para ser compartilhado com outros desenvolvedores ou projetos.

Compilando o Projeto Maven

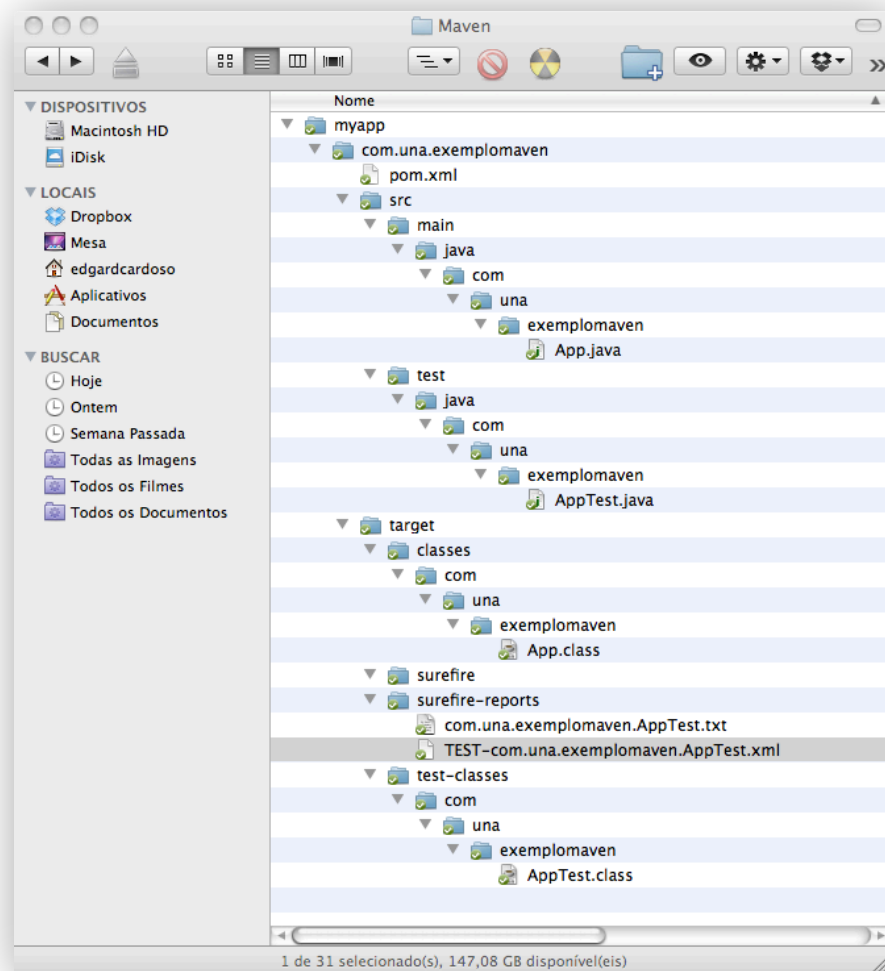
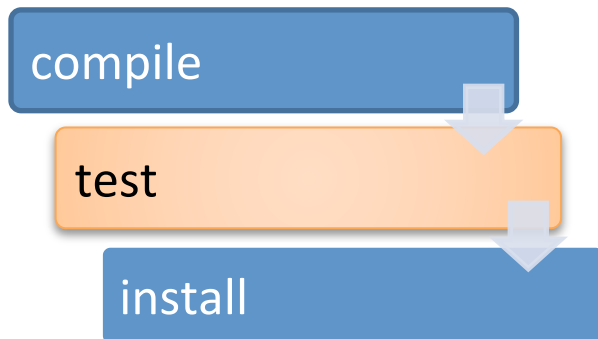
- Para compilar o projeto execute o comando:
`mvn compile`



Testando o Projeto Maven

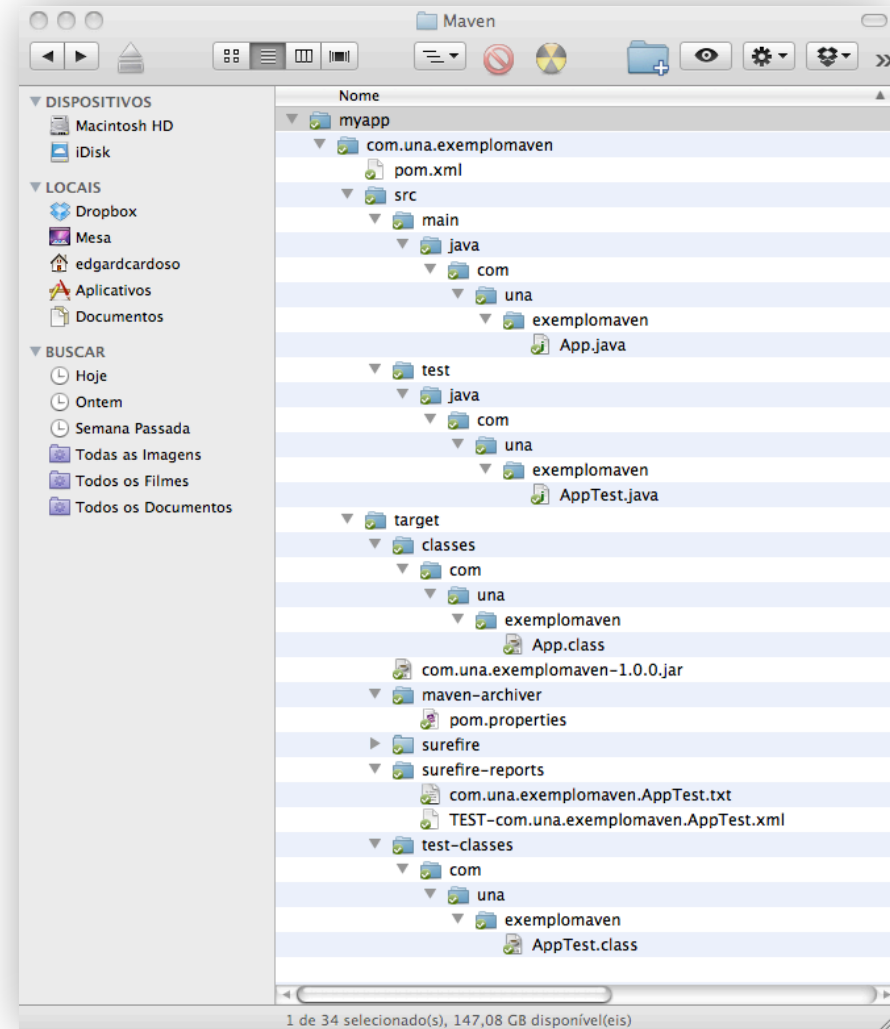
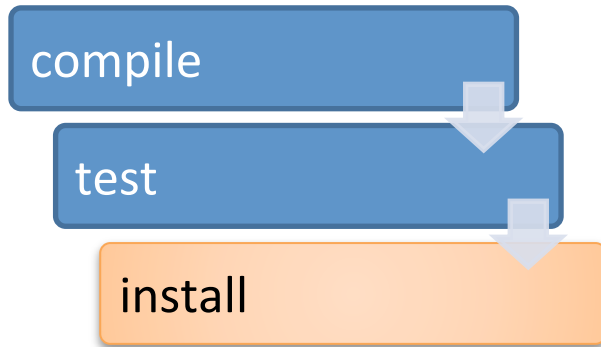
- Para testar o projeto execute o comando:

`mvn test`



Instalando o Projeto Maven

- Para testar o projeto execute o comando:
`mvn install`



Adicionando Dependências

Código Original

```
package com.una.exemplomaven;

/**
 * Hello world!
 *
 */
public class App
{
    public static void main( String[] args )
    {
        System.out.println( "Hello World!" );
    }
}
```

Código Modificado

```
package com.una.exemplomaven;

import org.slf4j.*;

/**
 * Hello world!
 *
 */
public class App
{
    public static void main( String[] args )
    {
        System.out.println( "Hello World!" );
        Logger logger = LoggerFactory.getLogger(App.class);
        logger.info("Hello World!");
    }
}
```

Dependências

Falta adicionar as dependências
De slf4j no arquivo pom.xml

Pesquisando Dependências

Dependência Maven

The screenshot shows the Maven Repository search results for 'slf4j'. The search bar contains 'slf4j' and the results show 'SLF4J API Module' as the top result, circled in red. Below it, 'SLF4J LOG4J-12 Binding' is also visible. The left sidebar shows 'Repository' with 'Plugins' and 'Tag Cloud' options, and 'Artifacts/Jars' with a graph showing growth from 2004 to 2012. Popular tags include 'ajax analysis', 'annotations ant', and 'apache api'.

The screenshot shows the Maven Repository artifact page for 'org.slf4j:slf4j-api:1.6.4'. The page title is 'SLF4J API Module'. The artifact is described as 'The slf4j API'. A large red arrow points from the 'Dependência Maven' header to the 'Maven' tab at the bottom. The 'Maven' tab is selected, showing the dependency XML snippet:

```
<dependency>
  <groupId>org.slf4j</groupId>
  <artifactId>slf4j-api</artifactId>
  <version>1.6.4</version>
</dependency>
```

pom.xml com as novas dependências

```
<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.una.exemplomaven</groupId>
  <artifactId>com.una.exemplomaven</artifactId>
  <version>1.0.0</version>
  <packaging>jar</packaging>

  <name>com.una.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>

    <dependency>
      <groupId>org.slf4j</groupId>
      <artifactId>slf4j-api</artifactId>
      <version>1.6.4</version>
    </dependency>
  </dependencies>
</project>
```

Dependência do slf4j
adicionado

Adicionando Dependência no NetBeans

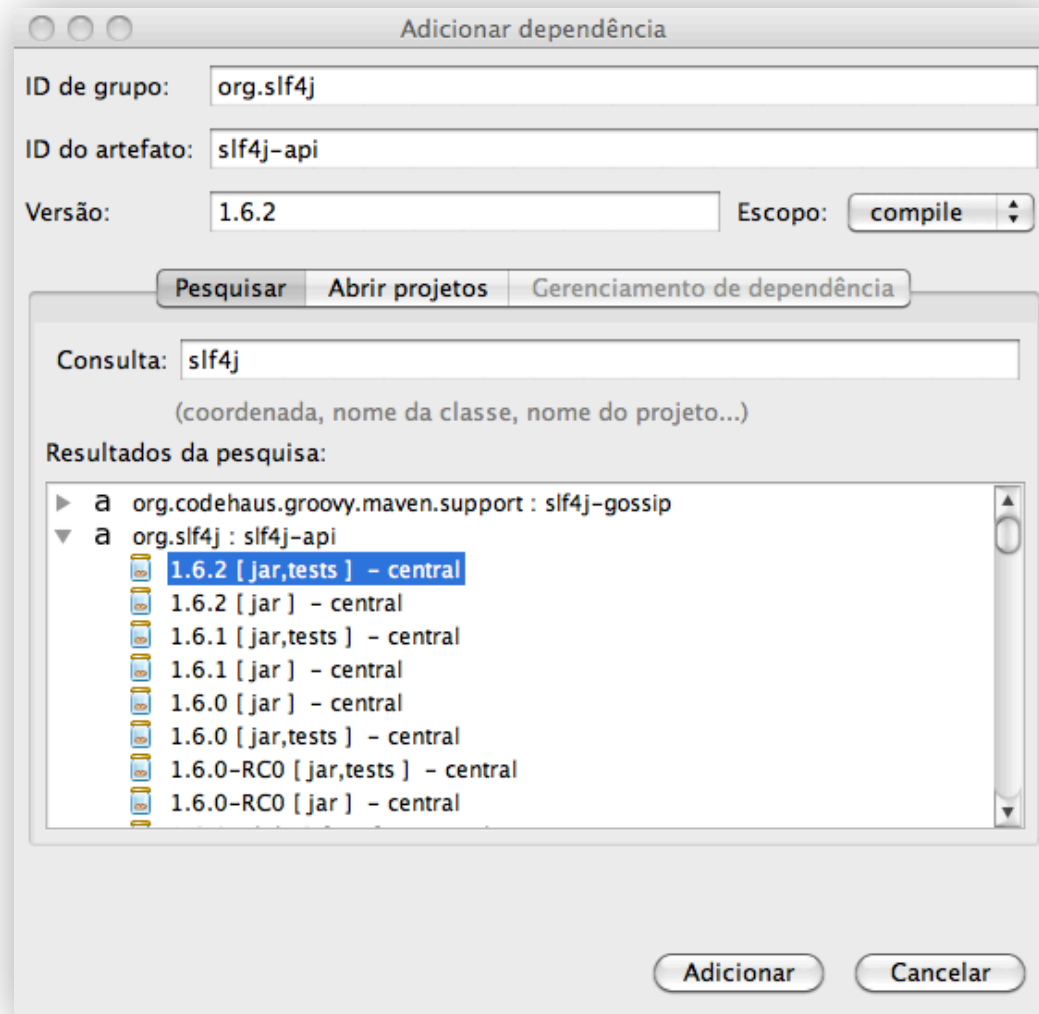


Gráfico de Dependências – NetBeans projeto Web com Hibernate

