

Agile Software Development

Produced
by

Eamonn de Leastar (edeleastar@wit.ie)

Department of Computing, Maths & Physics
Waterford Institute of Technology

<http://www.wit.ie>

<http://elearning.wit.ie>

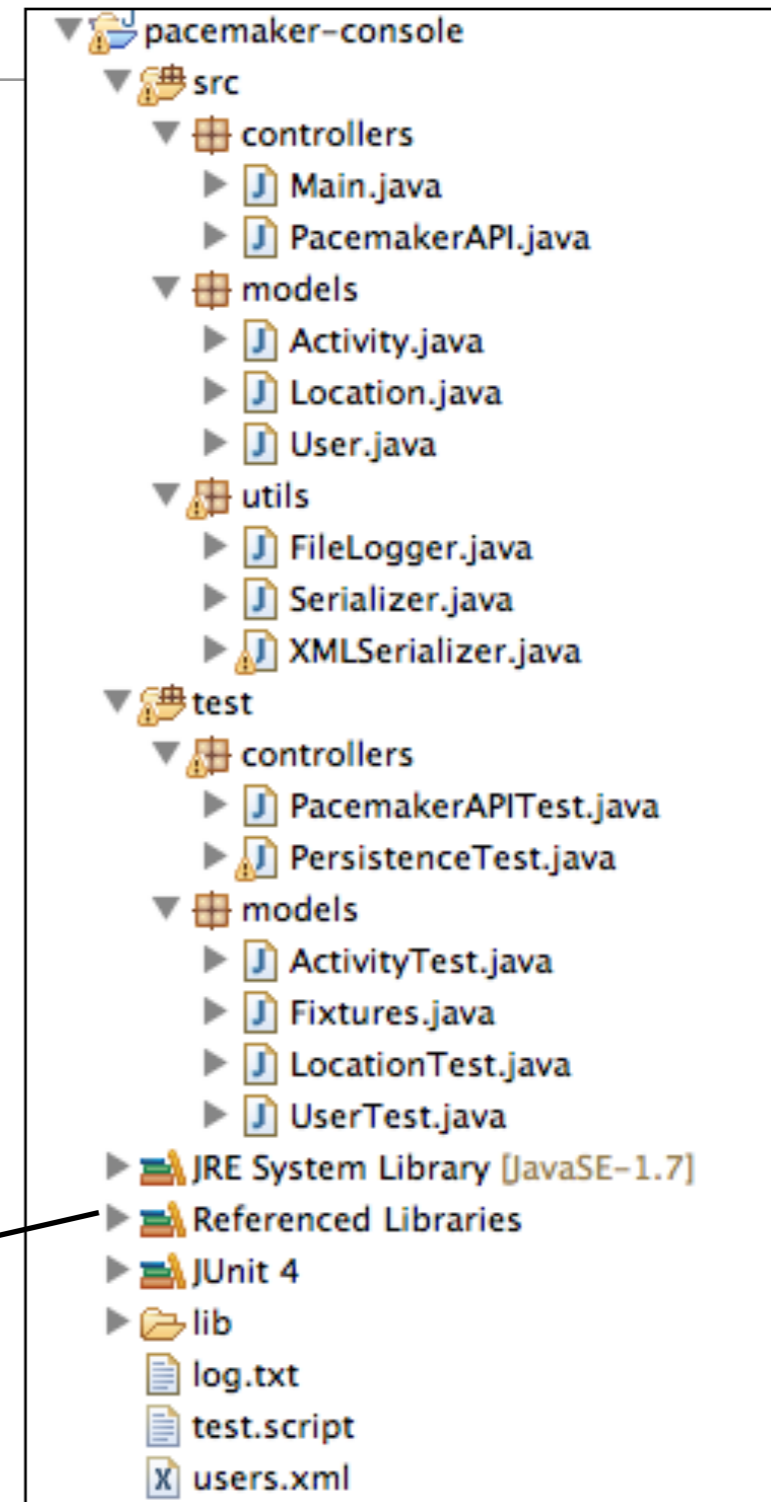


Waterford Institute of Technology
INSTITIÚID TEICNEOLAÍOCHTA PHORT LÁIRCE



pacemaker-console

- Take the pacemaker-console (java) project
- Note the external dependencies:
 - guava-14.0.1.jar
 - xstream-1.4.4.jar
 - junit-4.11.jar
 - asg.cliche-11-413.jar



Project Creation Archetype

- From the command line, execute maven command to create a project skeleton structure:

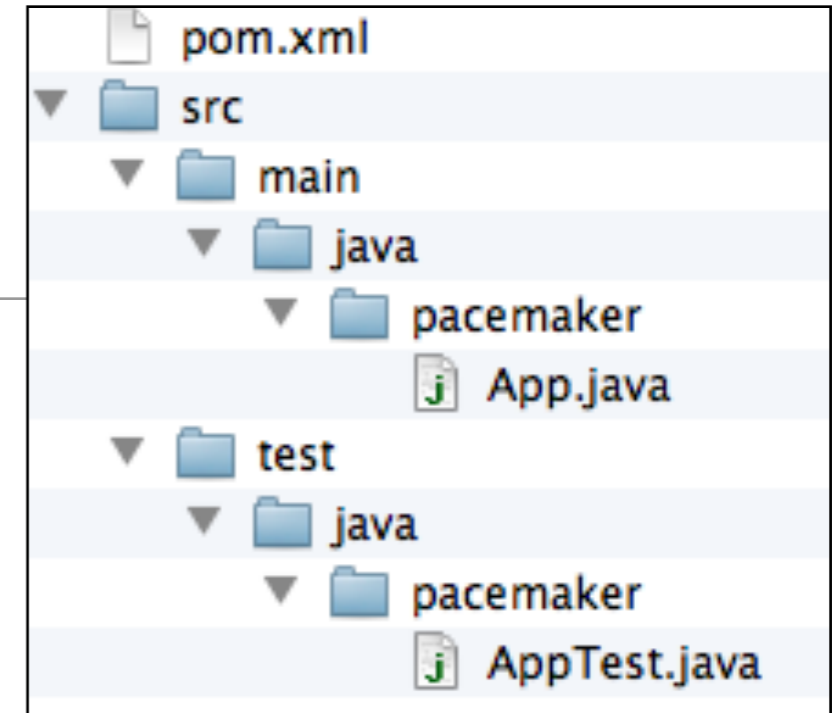
```
mvn archetype:create -DarchetypeGroupId=org.apache.maven.archetypes
-DarchetypeArtifactId=maven-archetype-quickstart
-DgroupId=pacemaker
-DartifactId=pacemaker-console-maven
```

```
Grendel:Session 9 edeleastar$ mvn archetype:create -DarchetypeGroupId=org.apache.maven.archetypes -D
archetypeArtifactId=maven-archetype-quickstart -DgroupId=msccomm -DartifactId=Pim
[INFO] Scanning for projects...
[INFO] Searching repository for plugin with prefix: 'archetype'.
[INFO] -----
[INFO] Building Maven Default Project
[INFO]   task-segment: [archetype:create] (aggregator-style)
[INFO] -----
[INFO] Setting property: classpath.resource.loader.class => 'org.codehaus.plexus.velocity.ContextClass
ClassLoaderResourceLoader'.
[INFO] Setting property: velocimacro.messages.on => 'false'.
[INFO] Setting property: resource.loader => 'classpath'.
[INFO] Setting property: resource.manager.logwhenfound => 'false'.
[INFO] *****
[INFO] Starting Jakarta Velocity v1.4
[INFO] RuntimeInstance initializing.
[INFO] Default Properties File: org/apache/velocity/runtime/defaults/velocity.properties
[INFO] Default ResourceManager initializing. (class org.apache.velocity.runtime.resource.ResourceMan
agerImpl)
[INFO] Resource Loader Instantiated: org.codehaus.plexus.velocity.ContextClassLoaderResourceLoader
[INFO] ClasspathResourceLoader : initialization starting.
[INFO] ClasspathResourceLoader : initialization complete.
[INFO] ResourceCache : initialized. (class org.apache.velocity.runtime.resource.ResourceCacheImpl)
[INFO] Default ResourceManager initialization complete.
[INFO] Loaded System Directive: org.apache.velocity.runtime.directive.Literal
[INFO] Loaded System Directive: org.apache.velocity.runtime.directive.Macro
[INFO] Loaded System Directive: org.apache.velocity.runtime.directive.Parse
[INFO] Loaded System Directive: org.apache.velocity.runtime.directive.Include
[INFO] Loaded System Directive: org.apache.velocity.runtime.directive.Foreach
[INFO] Created: 20 parsers.
[INFO] Velocimacro : initialization starting.
[INFO] Velocimacro : adding VMs from VM library template : VM_global_library.vm
[ERROR] ResourceManager : unable to find resource 'VM_global_library.vm' in any resource loader.
[INFO] Velocimacro : error using VM library template VM_global_library.vm : org.apache.velocity.exc
eption.ResourceNotFoundException: Unable to find resource 'VM_global_library.vm'
[INFO] Velocimacro : VM library template macro registration complete.
[INFO] -----
[INFO] Using following parameters for creating Archetype: maven-archetype-quickstart:RELEASE
[INFO] -----
[INFO] Parameter: groupId, Value: msccomm
[INFO] Parameter: packageName, Value: msccomm
[INFO] Parameter: basedir, Value: /Users/edeleastar/svn/CourseWork/2008/Agile Software Development/s
rc/Session 9
[INFO] Parameter: package, Value: msccomm
[INFO] Parameter: version, Value: 1.0-SNAPSHOT
[INFO] Parameter: artifactId, Value: Pim
[INFO] ***** End of debug info from resources from generated POM *****
****
[INFO] Archetype created in dir: /Users/edeleastar/svn/CourseWork/2008/Agile Software Development/sr
c/Session 9/Pim
[INFO] -----
[INFO] BUILD SUCCESSFUL
[INFO] -----
[INFO] Total time: 2 seconds
[INFO] Finished at: Sat Nov 08 10:46:09 GMT 2008
[INFO] Final Memory: 5M/9M
[INFO] -----
```

```
templates
may NOT replace previous VM
be global in scope if allow
```

Generated Directory Structure

- The generated directory structure contains two dummy java files that can be discarded
- It also generates a POM, which is the basis for the project dependency structure.



```
<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>pacemaker</groupId>
  <artifactId>pacemaker-console-maven</artifactId>
  <version>1.0-SNAPSHOT</version>
  <packaging>jar</packaging>

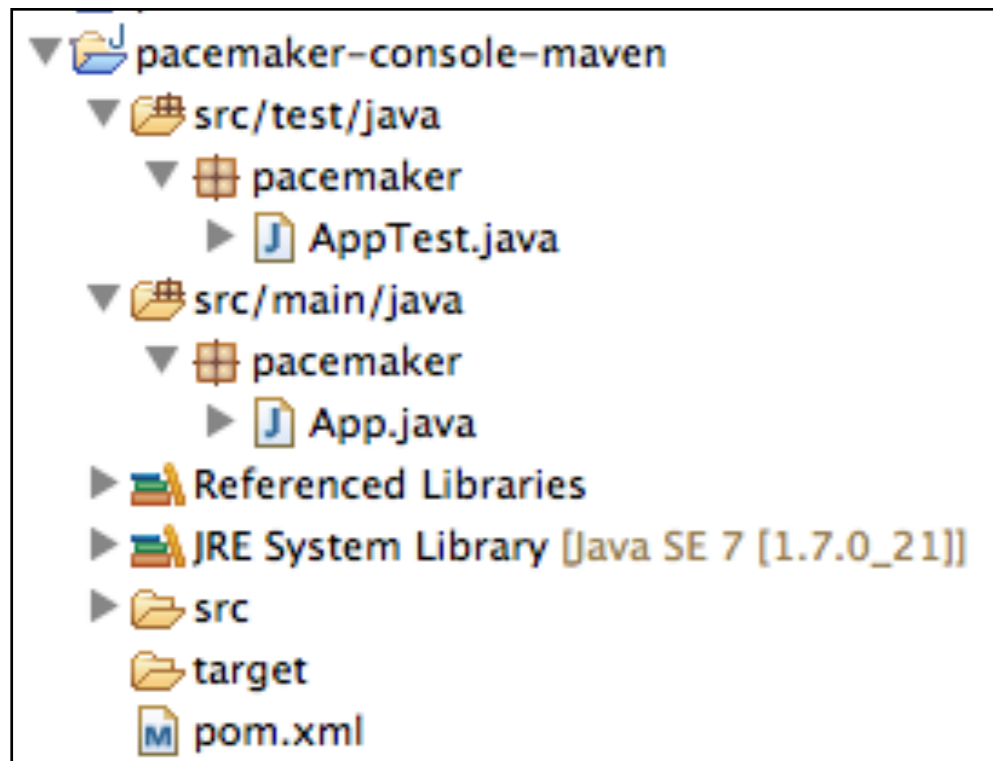
  <name>pacemaker-console-maven</name>
  <url>http://maven.apache.org</url>

  <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>
```

Generate 'Eclipse' project

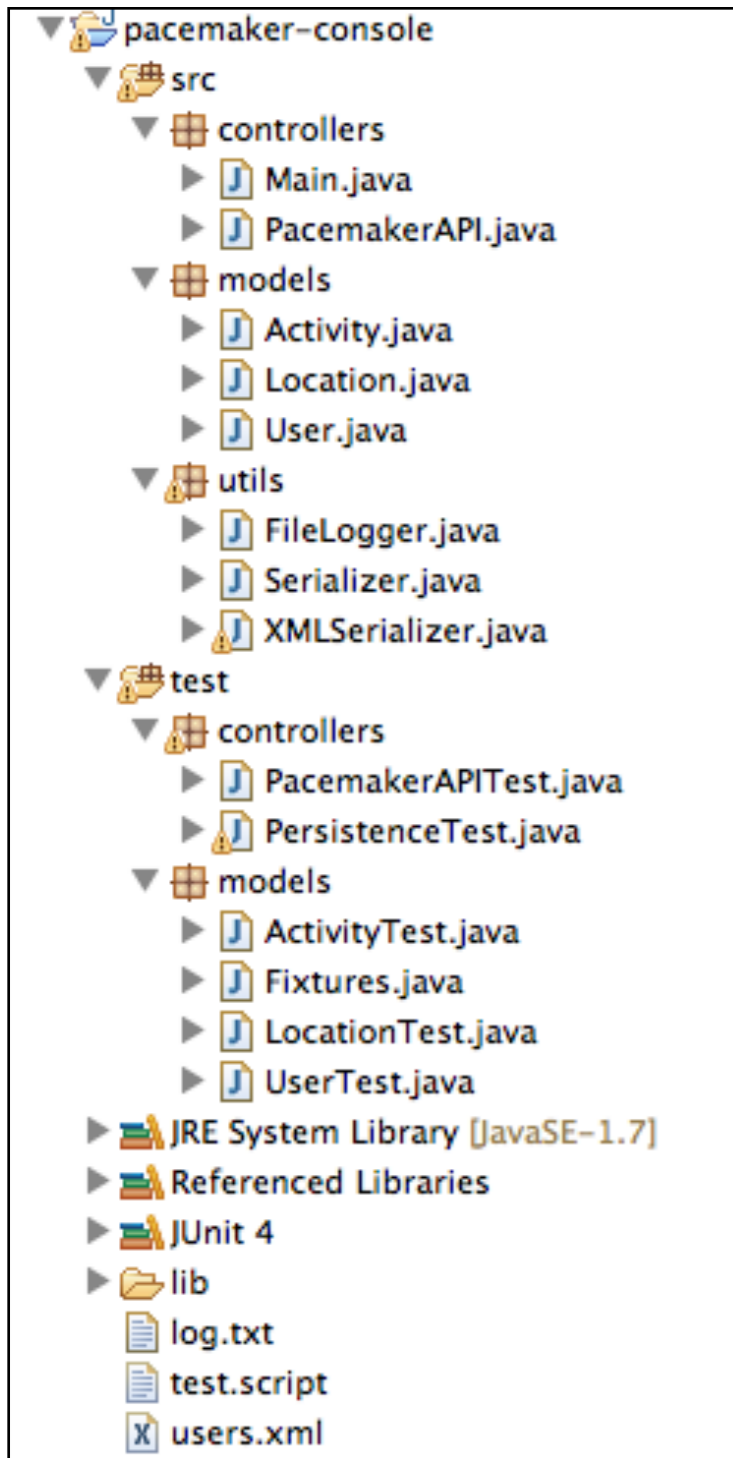
```
mvn eclipse:eclipse
```



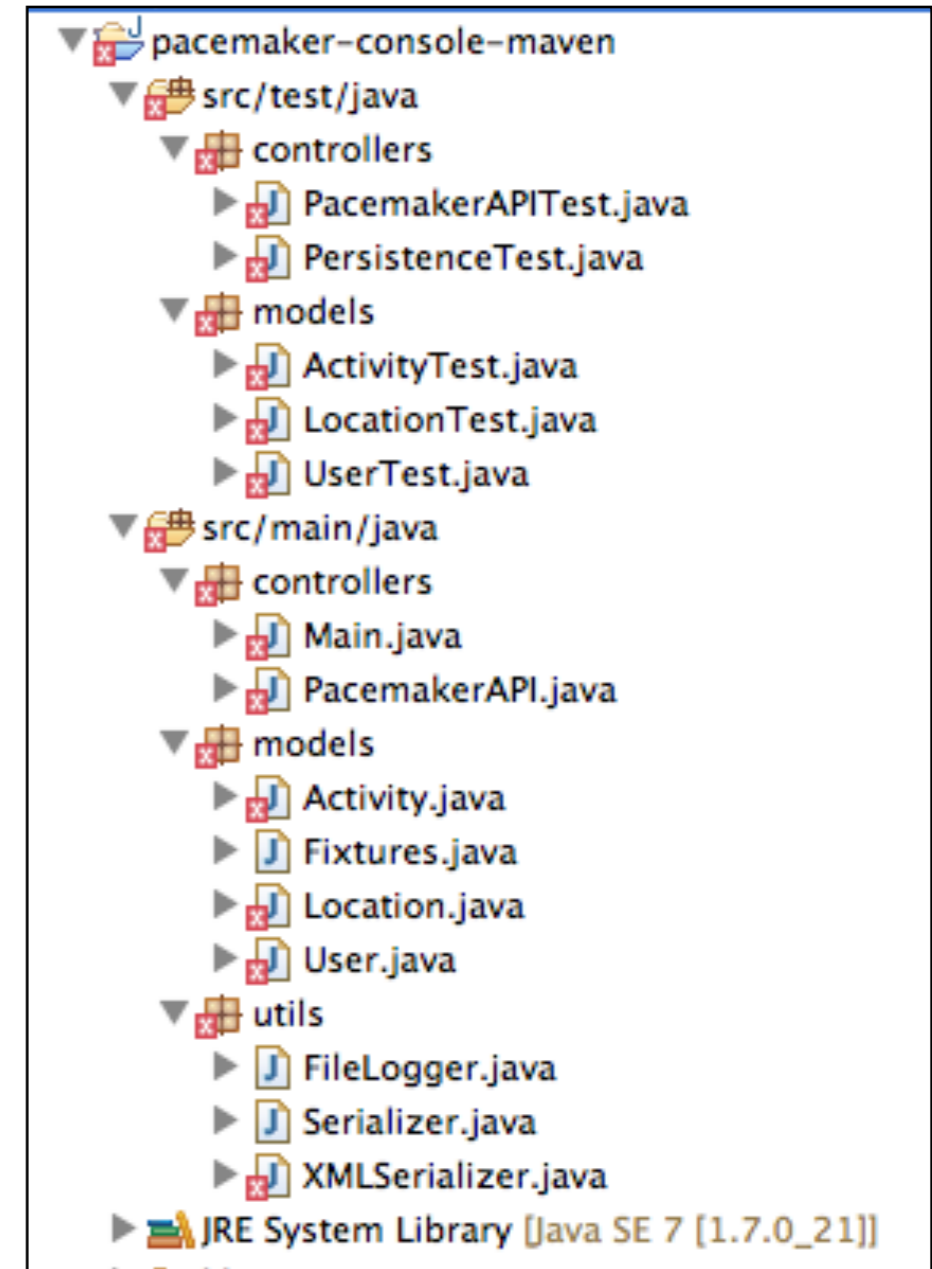
- Generates the '.project' and '.classpath' Eclipse uses to specify project name, structure and dependencies.
- The project can then be 'imported' into eclipse

Manipulate the project in eclipse...

- Copy Paste sources from original project into maven generated project.

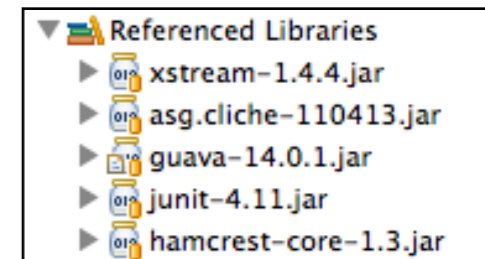


The screenshot shows the project structure for 'pacemaker-console'. It includes a 'src' folder with subfolders 'controllers', 'models', and 'utils', and a 'test' folder with subfolders 'controllers' and 'models'. The 'controllers' folder contains 'Main.java' and 'PacemakerAPI.java'. The 'models' folder contains 'Activity.java', 'Location.java', and 'User.java'. The 'utils' folder contains 'FileLogger.java', 'Serializer.java', and 'XMLSerializer.java'. The 'test' folder contains 'PacemakerAPITest.java' and 'PersistenceTest.java' in the 'controllers' subfolder, and 'ActivityTest.java', 'Fixtures.java', 'LocationTest.java', and 'UserTest.java' in the 'models' subfolder. Below the source code, there are 'Referenced Libraries' including 'JUnit 4' and a 'lib' folder containing 'log.txt', 'test.script', and 'users.xml'.



The screenshot shows the project structure for 'pacemaker-console-maven'. It includes a 'src/test/java' folder with subfolders 'controllers' and 'models', and a 'src/main/java' folder with subfolders 'controllers' and 'models'. The 'controllers' folder contains 'PacemakerAPITest.java' and 'PersistenceTest.java'. The 'models' folder contains 'ActivityTest.java', 'LocationTest.java', and 'UserTest.java'. The 'src/main/java' folder contains 'Main.java' and 'PacemakerAPI.java' in the 'controllers' subfolder, and 'Activity.java', 'Fixtures.java', 'Location.java', and 'User.java' in the 'models' subfolder. The 'utils' folder contains 'FileLogger.java', 'Serializer.java', and 'XMLSerializer.java'. Below the source code, there are 'Referenced Libraries' including 'JUnit 4' and a 'lib' folder containing 'log.txt', 'test.script', and 'users.xml'.

Reference errors - project is missing 'Referenced Libraries'



The screenshot shows the 'Referenced Libraries' for the maven project. It includes 'xstream-1.4.4.jar', 'asg.cliche-110413.jar', 'guava-14.0.1.jar', 'junit-4.11.jar', and 'hamcrest-core-1.3.jar'.

Default POM

```
<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>pacemaker</groupId>
  <artifactId>pacemaker-console-maven</artifactId>
  <version>1.0-SNAPSHOT</version>
  <packaging>jar</packaging>

  <name>pacemaker-console-maven</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>
```

url + version

```
<version>1.0</version>  
<url>http://www.wit.ie</url>
```

```
<project...>  
  <modelVersion>4.0.0</modelVersion>  
  
  <groupId>pacemaker</groupId>  
  <artifactId>pacemaker-console-maven</artifactId>  
  <version>1.0</version>  
  <packaging>jar</packaging>  
  
  <name>pacemaker-console-maven</name>  
  <url>http://www.wit.ie</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>
```


Java 7 Support + 'rev' Junit to 4.11

```
<properties>  
  <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>  
  <maven.compiler.source>1.7</maven.compiler.source>  
  <maven.compiler.target>1.7</maven.compiler.target>  
</properties>
```

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

```
<project ...>

  <groupId>pacemaker</groupId>
  <artifactId>pacemaker-console-maven</artifactId>
  <version>1.0</version>
  <packaging>jar</packaging>

  <name>pacemaker-console-maven</name>
  <url>http://www.wit.ie</url>

  <properties>
    <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
    <maven.compiler.source>1.7</maven.compiler.source>
    <maven.compiler.target>1.7</maven.compiler.target>
  </properties>

  <dependencies>

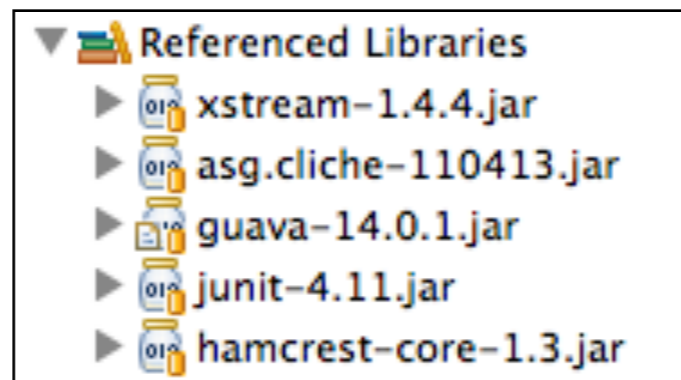
    <dependency>
      <groupId>junit</groupId>
      <artifactId>junit</artifactId>
      <version>4.11</version>
      <scope>test</scope>
    </dependency>

  </dependencies>

</project>
```

Dependencies

- Junit installed
- Need to incorporate
 - guava
 - xstream
- and also
 - asg.cliche
- what about
 - hamcrest-core?



- Well supported, maintained components:
 - xstream, guava
- abandoned, legacy?
 - asg.cliche
- Downstream dependency of junit
 - hamcrest-core

Dependency entries:

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

  <dependency>
    <groupId>com.google.guava</groupId>
    <artifactId>guava</artifactId>
    <version>15.0</version>
  </dependency>

  <dependency>
    <groupId>com.thoughtworks.xstream</groupId>
    <artifactId>xstream</artifactId>
    <version>1.4.4</version>
  </dependency>

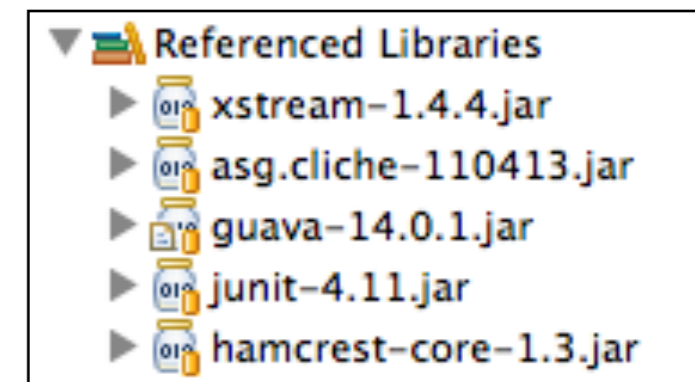
  <dependency>
    <groupId>asg-cliche</groupId>
    <artifactId>asg-cliche</artifactId>
    <version>1.0</version>
  </dependency>
</dependencies>
```

• Junit

• guava

• xstream

• asg.cliche



junit 'depends on' hamcrest', and including junit will automatically incorporate hamcrest into build

```

<project ...>
  <groupId>pacemaker</groupId>
  <artifactId>pacemaker-console-maven</artifactId>
  <version>1.0</version>
  <packaging>jar</packaging>

  <name>pacemaker-console-maven</name>
  <url>http://www.wit.ie</url>

  <properties>
    <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
    <maven.compiler.source>1.7</maven.compiler.source>
    <maven.compiler.target>1.7</maven.compiler.target>
  </properties>

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

    <dependency>
      <groupId>com.google.guava</groupId>
      <artifactId>guava</artifactId>
      <version>15.0</version>
    </dependency>

    <dependency>
      <groupId>com.thoughtworks.xstream</groupId>
      <artifactId>xstream</artifactId>
      <version>1.4.4</version>
    </dependency>

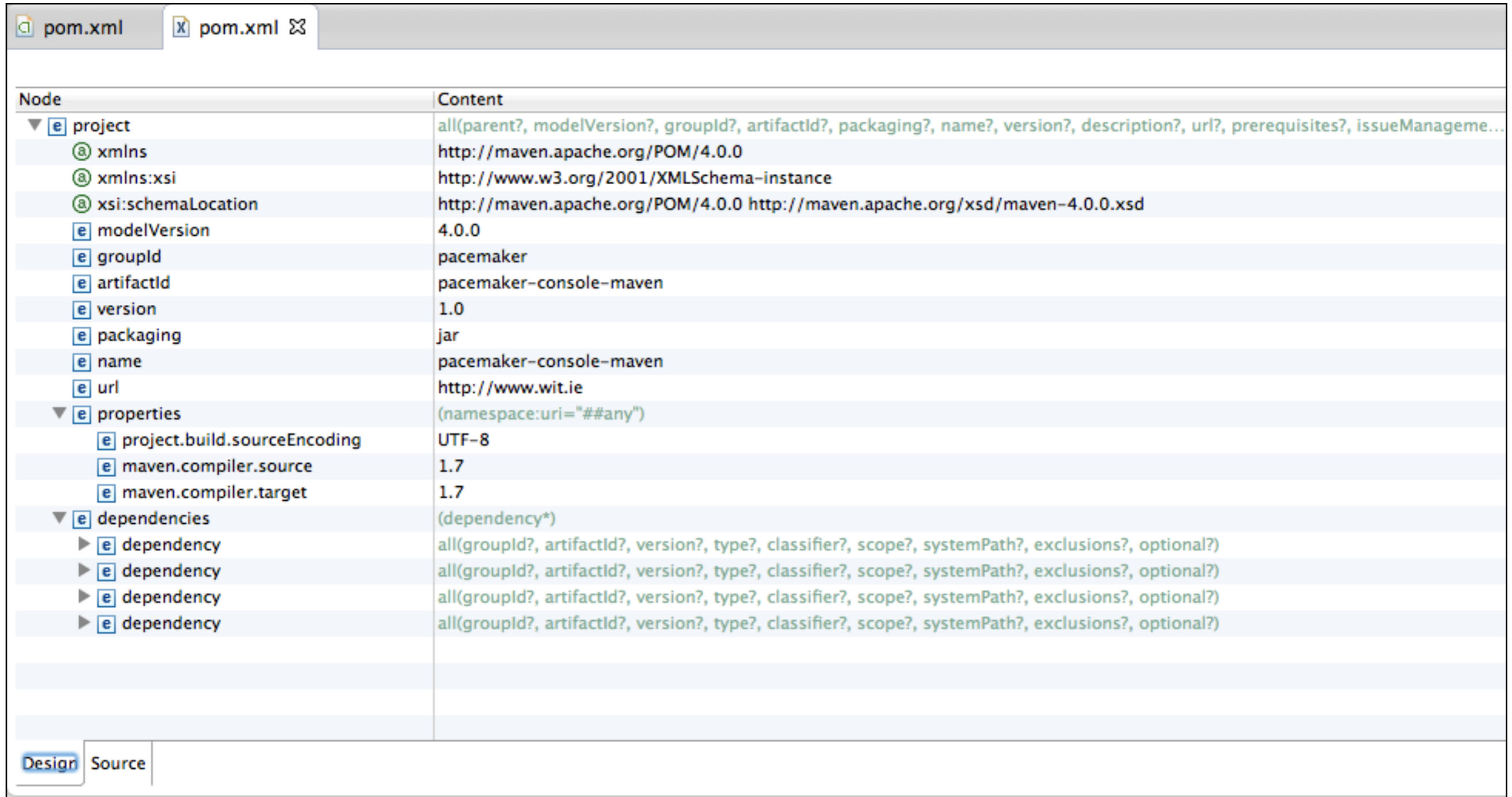
    <dependency>
      <groupId>asg-liche</groupId>
      <artifactId>asg-liche</artifactId>
      <version>1.0</version>
    </dependency>
  </dependencies>
</project>

```

Complete POM

- Project identifiers
- Project version
- Language version
- Dependencies

Design View (Eclipse XML Editor)



The screenshot shows the Eclipse XML Editor in Design View for a pom.xml file. The interface includes a tab bar at the top with two 'pom.xml' tabs. Below the tab bar is a table with two columns: 'Node' and 'Content'. The 'Node' column shows a tree structure of XML elements, and the 'Content' column shows the corresponding XML content. The 'project' node is expanded, showing its attributes and child elements. The 'dependencies' node is also expanded, showing four 'dependency' child elements. At the bottom of the editor, there are two tabs: 'Design' (selected) and 'Source'.

Node	Content
▼ e project	all(parent?, modelVersion?, groupId?, artifactId?, packaging?, name?, version?, description?, url?, prerequisites?, issueManageme...
a xmlns	http://maven.apache.org/POM/4.0.0
a xmlns:xsi	http://www.w3.org/2001/XMLSchema-instance
a xsi:schemaLocation	http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd
e modelVersion	4.0.0
e groupId	pacemaker
e artifactId	pacemaker-console-maven
e version	1.0
e packaging	jar
e name	pacemaker-console-maven
e url	http://www.wit.ie
▼ e properties	(namespace:uri="##any")
e project.build.sourceEncoding	UTF-8
e maven.compiler.source	1.7
e maven.compiler.target	1.7
▼ e dependencies	(dependency*)
▶ e dependency	all(groupId?, artifactId?, version?, type?, classifier?, scope?, systemPath?, exclusions?, optional?)
▶ e dependency	all(groupId?, artifactId?, version?, type?, classifier?, scope?, systemPath?, exclusions?, optional?)
▶ e dependency	all(groupId?, artifactId?, version?, type?, classifier?, scope?, systemPath?, exclusions?, optional?)
▶ e dependency	all(groupId?, artifactId?, version?, type?, classifier?, scope?, systemPath?, exclusions?, optional?)

Eclipse Maven Support (via plugin)

The screenshot displays the Eclipse IDE's Maven Overview for a project named 'pacemaker-console-maven'. The interface is divided into several sections:

- Artifact:** Contains fields for Group Id (pacemaker), Artifact Id (*pacemaker-console-maven), Version (1.0), and Packaging (jar).
- Parent:** A section for defining the parent project, currently empty.
- Properties:** A list of properties including project.build.sourceEncoding (UTF-8), maven.compiler.source (1.7), and maven.compiler.target (1.7). Buttons for 'Create...' and 'Remove' are visible.
- Modules:** A section for defining modules, currently empty with a 'New module element' button.
- Project:** Contains fields for Name (pacemaker-console-maven), URL (http://www.wit.ie), and a large text area for Description.
- Inception:** A field for defining the inception date.
- Organization, SCM, Issue Management, Continuous Integration:** Collapsible sections for defining project metadata.

At the bottom, a navigation bar shows tabs for Overview, Dependencies, Dependency Hierarchy, Effective POM, and pom.xml.

Dependency Hierarchy

- Shows 'implicit' dependencies
 - hamcrest, xmlpull, xpp3_min

The screenshot displays an IDE window titled "Dependency Hierarchy [test]" for the file "pacemaker-console-maven/pom.xml". The window is split into two panels:

- Dependency Hierarchy:** Shows a tree view of dependencies. The root is "junit : 4.11 [test]". Under it are "hamcrest-core : 1.3 [test]" and "guava : 15.0 [compile]". Below "guava" is "xstream : 1.4.4 [compile]". Under "xstream" are "xmlpull : 1.1.3.1 [compile]", "xpp3_min : 1.1.4c [compile]", and "asg-cliche : 1.0 [compile]".
- Resolved Dependencies:** Shows a list of resolved dependencies: "asg-cliche : 1.0 [compile]", "guava : 15.0 [compile]", "hamcrest-core : 1.3 [test]", "junit : 4.11 [test]", "xmlpull : 1.1.3.1 [compile]", "xpp3_min : 1.1.4c [compile]", and "xstream : 1.4.4 [compile]".

Supported Libraries

- junit, guava and xstream are all under active development, and latest versions are deposited in public maven repositories

The screenshot shows the Maven Repository page for the `xstream` artifact. The page includes a search bar, a breadcrumb trail (`home » xstream » xstream`), and a list of available versions. A table lists the versions, their types, and download sizes. A 'Stats' section at the bottom shows a 'Size Growth' chart.

Repository

- Plugins
- Tag Cloud

Artifacts/Jars

Popular Tags

ajax analysis annotations ant apache api archetype aspect asynchronously beans binding bpm build buildsystem bytecode cache cms codecoverage codehaus collections concurrency container database directory distributed doc eclipse ejb esb format framework graph graphics hadoop hibernate html http ide imap io jbi jdbc jdo jini jms jmx jndi jsf jsp language logging mail maven metadata microsoft mock net osgi parser pdf persistence plugin pool portal portlet query regexp rmi rpc rss ruleengine

home » xstream » xstream

xstream

tags: xml

Available versions

Version	Type	Download
1.2.2	release	Binary (0 bytes)
1.2.1	release	Binary (0 bytes)
1.2	release	Binary (335 KB)
1.1.3	release	Binary (256 KB)
1.1.2	release	Binary (242 KB)
1.1.1	release	Binary (219 KB)
1.1	release	Binary (195 KB)
1.0.2	release	Binary (110 KB)
1.0.1	release	Binary (99 KB)
1.0	release	Binary (93 KB)
1.0-rc1	release candidate	Binary (91 KB)
0.6	release	Binary (84 KB)
0.6-rc1	release candidate	Binary (81 KB)
0.5	release	Binary (78 KB)
0.4	release	Binary (66 KB)
0.3	release	Binary (58 KB)
0.2	release	Binary (57 KB)
0.1	release	Binary (52 KB)

Stats

Size Growth

344 KB

Unsupported Libraries

- asg-cliche is not under active development, and is not in any public maven repo.
- To keep our build consistent, we install asg-cliche in our local repo:

```
mvn install:install-file -Dfile=asg-cliche-1.0.jar -DgroupId=asg-cliche -DartifactId=asg-cliche -Dversion=1.0 -Dpackaging=jar
```

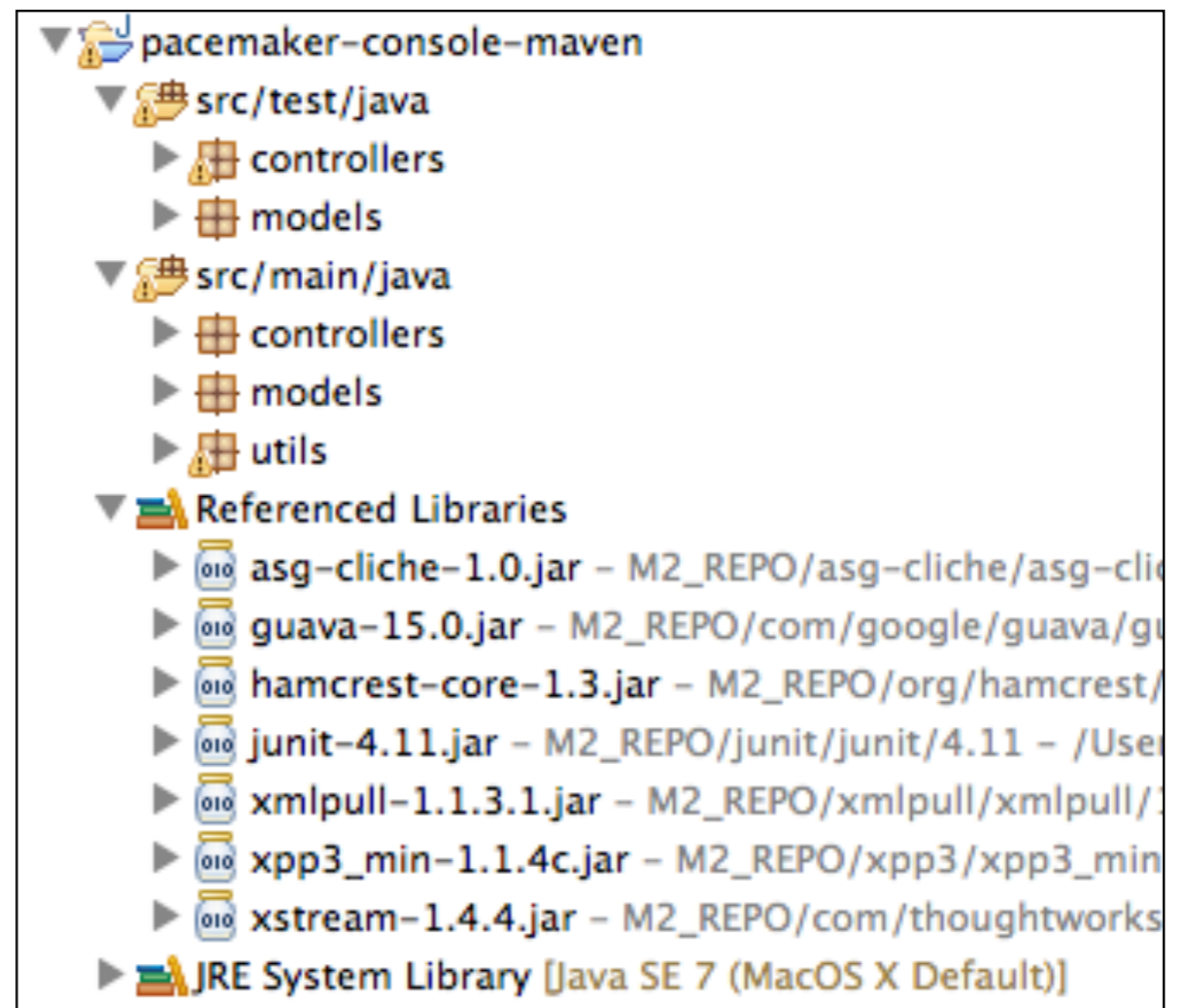
- Our maven build will find it locally, and resolve to that version without searching further.

Regenerate Eclipse Version

- On command line, instruct maven to refresh the eclipse project

```
mvn eclipse:eclipse
```

This will now include dependencies, including inferred upstream dependencies



Maven Lifecycles

- Maven is based around the central concept of a build lifecycle - a clearly defined process for building and distributing a particular artifact
- Only necessary to learn a small set of commands to build any Maven project, and the POM will ensure correct execution
- There are three built-in build lifecycles:
 - default: handles your project deployment
 - clean: handles project re-initialization/clean up
 - site. handles the creation of the project's documentation

Maven Default Lifecycle

- Each of these build lifecycles is defined by a different list of build phases, wherein a build phase represents a stage in the lifecycle.
- For example, the default lifecycle has the following build phases (for a complete list of the build phases, refer to the Lifecycle Reference):
 - validate
 - compile
 - test
 - package
 - integration-test
 - verify
 - install
 - deploy

pacemaker

- Validate
- Compile
- Test
- Package
- Install

```
<project ...>
  <groupId>pacemaker</groupId>
  <artifactId>pacemaker-console-maven</artifactId>
  <version>1.0</version>
  <packaging>jar</packaging>

  <name>pacemaker-console-maven</name>
  <url>http://www.wit.ie</url>

  <properties>
    <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
    <maven.compiler.source>1.7</maven.compiler.source>
    <maven.compiler.target>1.7</maven.compiler.target>
  </properties>

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

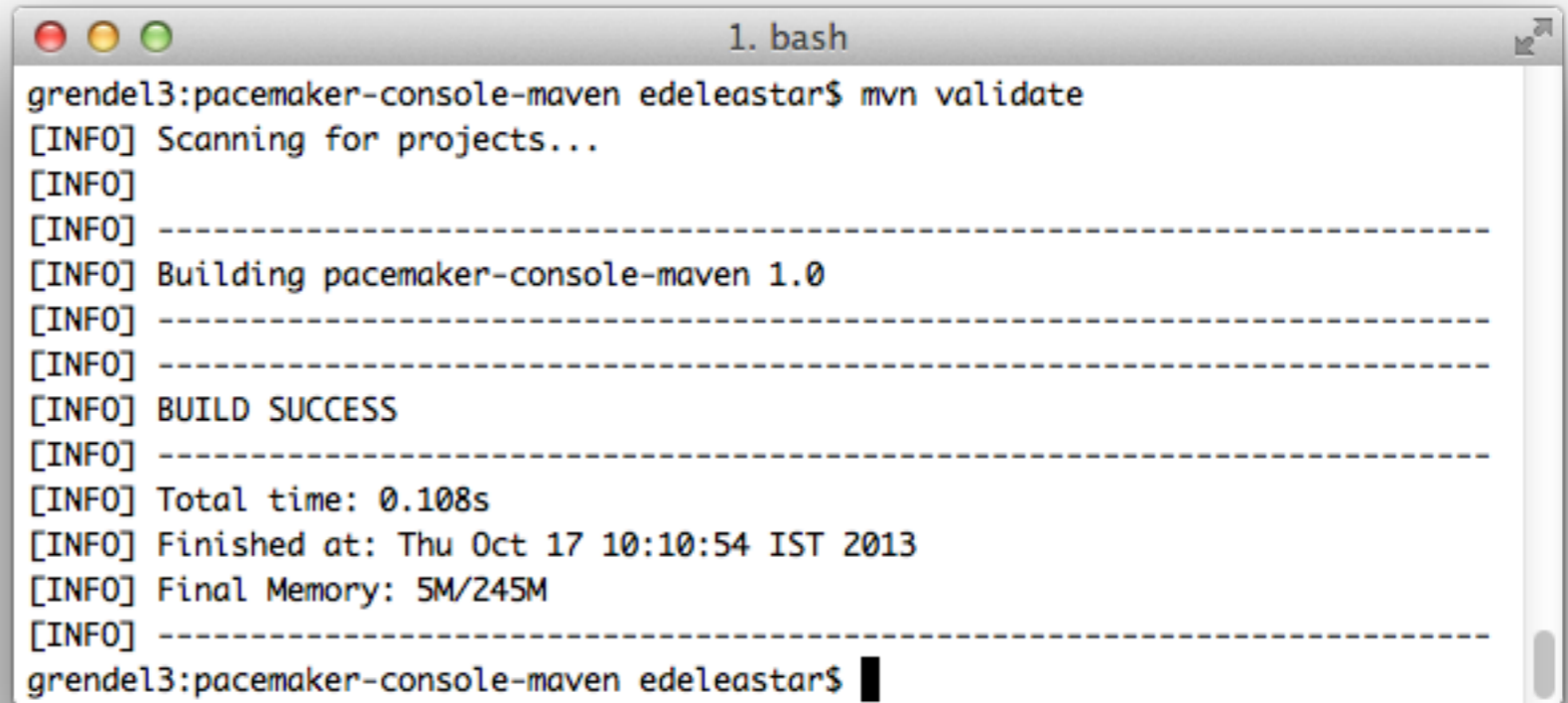
    <dependency>
      <groupId>com.google.guava</groupId>
      <artifactId>guava</artifactId>
      <version>15.0</version>
    </dependency>

    <dependency>
      <groupId>com.thoughtworks.xstream</groupId>
      <artifactId>xstream</artifactId>
      <version>1.4.4</version>
    </dependency>

    <dependency>
      <groupId>asg-cliche</groupId>
      <artifactId>asg-cliche</artifactId>
      <version>1.0</version>
    </dependency>
  </dependencies>
</project>
```

Validate

- Validate the project is correct and all necessary information is available

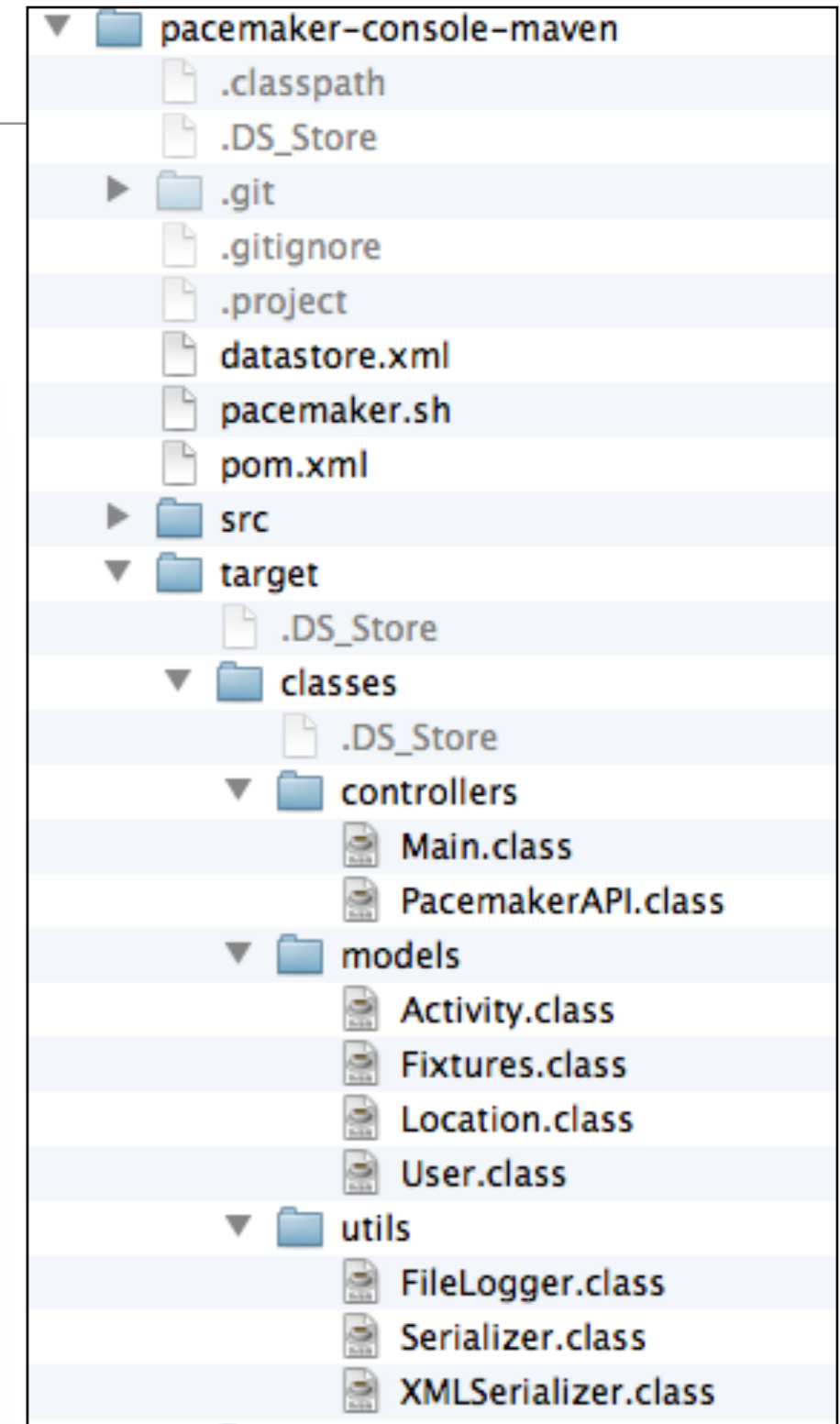
A terminal window titled "1. bash" showing the output of the command "mvn validate". The output includes information about scanning for projects, building the project "pacemaker-console-maven 1.0", and a successful build. The terminal also shows the total time, finish time, and final memory usage.

```
grendel3:pacemaker-console-maven edeleastar$ mvn validate
[INFO] Scanning for projects...
[INFO] -----
[INFO] Building pacemaker-console-maven 1.0
[INFO] -----
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 0.108s
[INFO] Finished at: Thu Oct 17 10:10:54 IST 2013
[INFO] Final Memory: 5M/245M
[INFO] -----
grendel3:pacemaker-console-maven edeleastar$
```

Compile

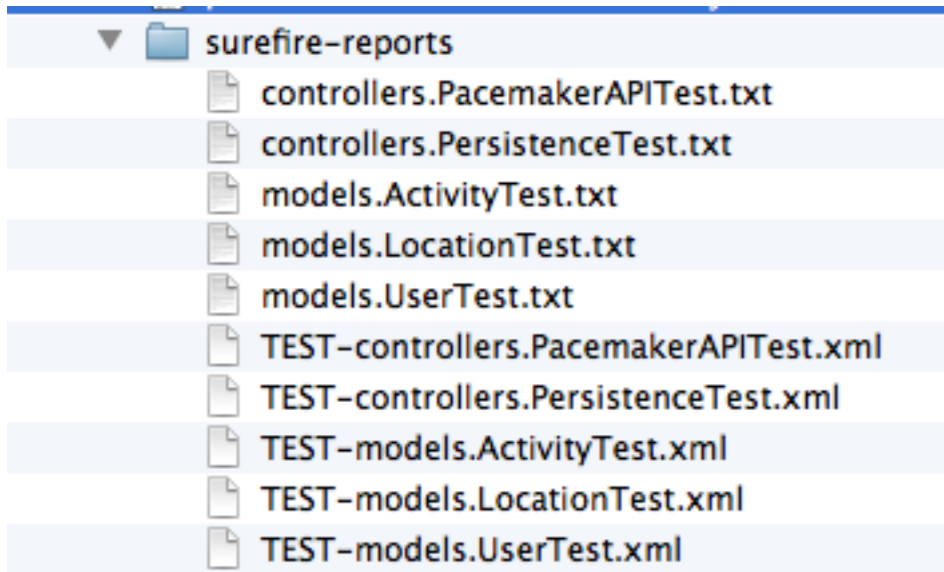
- compile the source code of the project

```
1. bash
grendel3:pacemaker-console-maven edeleastar$ mvn compile
[INFO] Scanning for projects...
[INFO]
[INFO] -----
[INFO] Building pacemaker-console-maven 1.0
[INFO] -----
[INFO]
[INFO] --- maven-resources-plugin:2.6:resources (default-resources) @ pacemaker-console-maven ---
[INFO] Using 'UTF-8' encoding to copy filtered resources.
[INFO] skip non existing resourceDirectory /Users/edeleastar/repos/modules/agile/prj/pacemaker-console-maven/src/main/resources
[INFO]
[INFO] --- maven-compiler-plugin:2.5.1:compile (default-compile) @ pacemaker-console-maven ---
[INFO] Nothing to compile - all classes are up to date
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 1.737s
[INFO] Finished at: Thu Oct 17 10:12:34 IST 2013
[INFO] Final Memory: 8M/245M
[INFO] -----
grendel3:pacemaker-console-maven edeleastar$
```



Test

- test the compiled source code using a suitable unit testing framework. These tests should not require the code be packaged or deployed



```
1. bash
n, secret, bart@simpson.com, {}}, User{30, lisa, simpson, secret, lisa@simpson
.com, {10=Activity{10, run, work, 2.2, []}, 11=Activity{11, walk, shop, 2.5, [
]}}}]
User to search for:
User{30, lisa, simpson, secret, lisa@simpson.com, {10=Activity{10, run, work,
2.2, []}, 11=Activity{11, walk, shop, 2.5, []}}
Collection
[User{32, maggie, simpson, secret, maggie@simpson.com, {}}, User{29, marge, si
mpson, secret, marge@simpson.com, {8=Activity{8, walk, fridge, 0.001, [Locatio
n{9, 23.3, 33.3}, Location{10, 34.4, 45.2}, Location{11, 25.3, 34.3}, Location
{12, 44.4, 23.3}]}, 9=Activity{9, walk, bar, 1.0, []}}}, User{31, bart, simpso
n, secret, bart@simpson.com, {}}, User{30, lisa, simpson, secret, lisa@simpson
.com, {10=Activity{10, run, work, 2.2, []}, 11=Activity{11, walk, shop, 2.5, [
]}}}]
Tests run: 2, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.322 sec
Running models.ActivityTest
Tests run: 2, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.003 sec
Running models.LocationTest
Tests run: 3, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0 sec
Running models.UserTest
Tests run: 4, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.001 sec

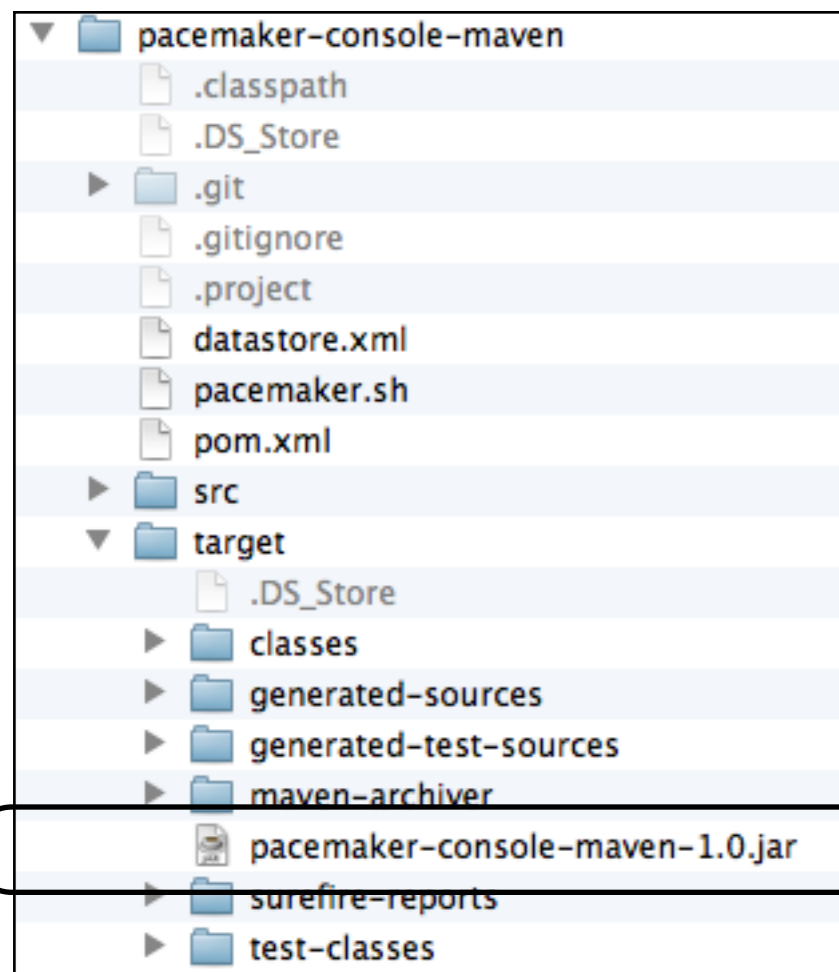
Results :

Tests run: 17, Failures: 0, Errors: 0, Skipped: 0

[INFO] -----
-
[INFO] BUILD SUCCESS
[INFO] -----
-
[INFO] Total time: 2.124s
[INFO] Finished at: Thu Oct 17 10:15:16 IST 2013
[INFO] Final Memory: 7M/245M
[INFO] -----
-
grendel3:pacemaker-console-maven edeleastar$
```

Package

- take the compiled code and package it in its distributable format, such as a JAR.

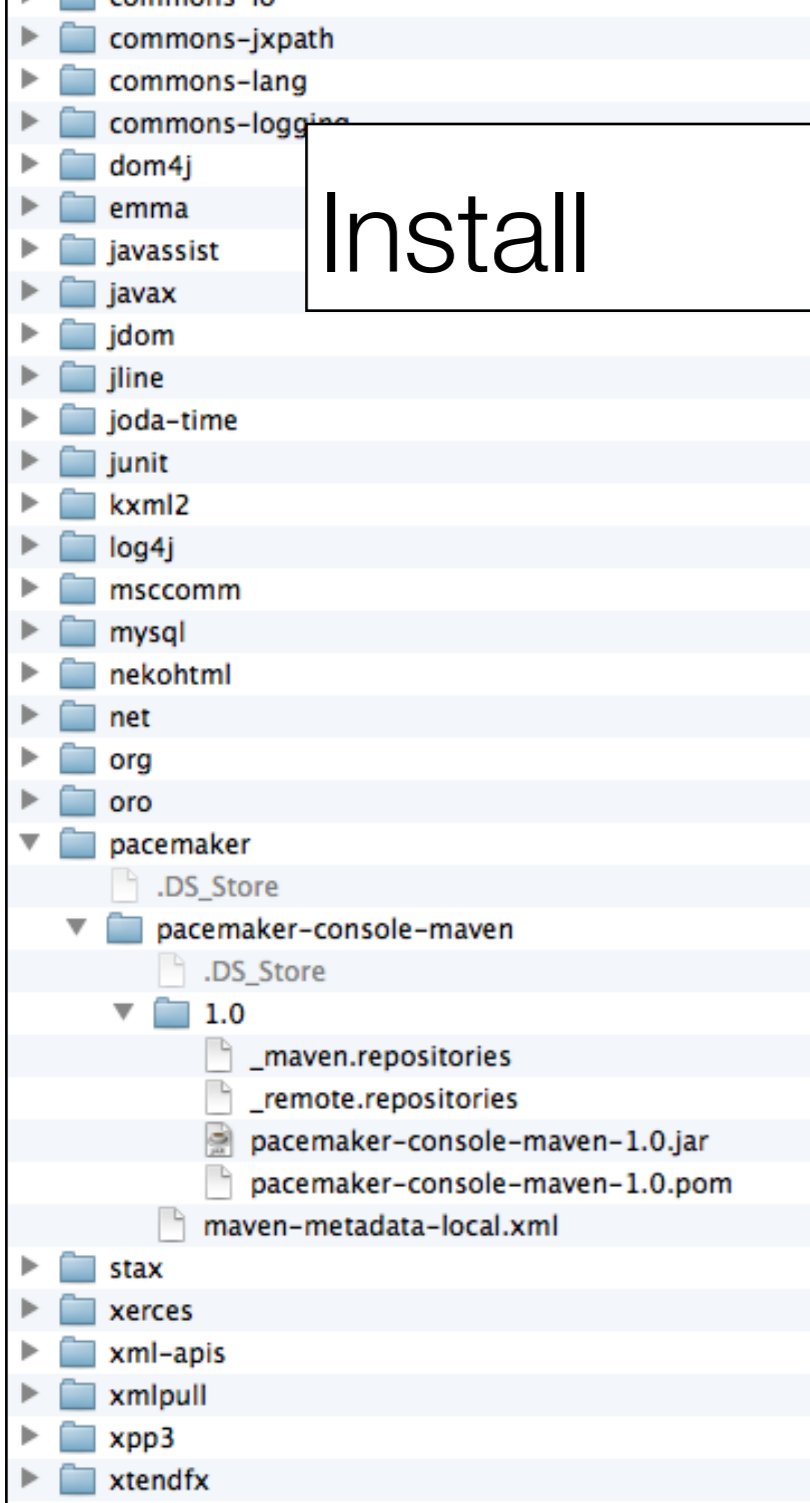


```
1. bash
2.2, []}, 11=Activity{11, walk, shop, 2.5, []}}
Collection
[User{32, maggie, simpson, secret, maggie@simpson.com, {}}, User{29, marge, si
mpson, secret, marge@simpson.com, {8=Activity{8, walk, fridge, 0.001, [Locatio
n{9, 23.3, 33.3}, Location{10, 34.4, 45.2}, Location{11, 25.3, 34.3}, Location
{12, 44.4, 23.3}]}, 9=Activity{9, walk, bar, 1.0, []}}, User{31, bart, simpso
n, secret, bart@simpson.com, {}}, User{30, lisa, simpson, secret, lisa@simpson
.com, {10=Activity{10, run, work, 2.2, []}, 11=Activity{11, walk, shop, 2.5, [
]}}}]
Tests run: 2, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.208 sec
Running models.ActivityTest
Tests run: 2, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0 sec
Running models.LocationTest
Tests run: 3, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.001 sec
Running models.UserTest
Tests run: 4, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0 sec

Results :

Tests run: 17, Failures: 0, Errors: 0, Skipped: 0

[INFO]
[INFO] --- maven-jar-plugin:2.4:jar (default-jar) @ pacemaker-console-maven ---
[INFO] Building jar: /Users/edeleastar/repos/modules/agile/prj/pacemaker-conso
le-maven/target/pacemaker-console-maven-1.0.jar
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 2.020s
[INFO] Finished at: Thu Oct 17 10:17:11 IST 2013
[INFO] Final Memory: 9M/245M
[INFO] -----
grendel3:pacemaker-console-maven edeleastar$
```



- install the package into the local repository, for use as a dependency in other projects locally

```
1. bash
.com, {10=Activity{10, run, work, 2.2, []}, 11=Activity{11, walk, shop, 2.5, [
]]}]
Tests run: 2, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.209 sec
Running models.ActivityTest
Tests run: 2, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0 sec
Running models.LocationTest
Tests run: 3, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0 sec
Running models.UserTest
Tests run: 4, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0 sec

Results :

Tests run: 17, Failures: 0, Errors: 0, Skipped: 0

[INFO]
[INFO] --- maven-jar-plugin:2.4:jar (default-jar) @ pacemaker-console-maven ---
-
[INFO]
[INFO] --- maven-install-plugin:2.4:install (default-install) @ pacemaker-console-maven ---
[INFO] Installing /Users/edelestar/repos/modules/agile/prj/pacemaker-console-maven/target/pacemaker-console-maven-1.0.jar to /Users/edelestar/.m2/repository/pacemaker/pacemaker-console-maven/1.0/pacemaker-console-maven-1.0.jar
[INFO] Installing /Users/edelestar/repos/modules/agile/prj/pacemaker-console-maven/pom.xml to /Users/edelestar/.m2/repository/pacemaker/pacemaker-console-maven/1.0/pacemaker-console-maven-1.0.pom
[INFO] -----
-
[INFO] BUILD SUCCESS
[INFO] -----
-
[INFO] Total time: 1.978s
[INFO] Finished at: Thu Oct 17 10:18:30 IST 2013
[INFO] Final Memory: 9M/245M
[INFO] -----
-
grendel3:pacemaker-console-maven edelestar$
```

Maven & Eclipse

- The project is now built entirely by the maven build system. However, we can continue to use Eclipse for day to day development.
- To do this, we generate an eclipse project from the maven project structure. Do this by issuing the following maven command:

```
mvn eclipse:eclipse
```

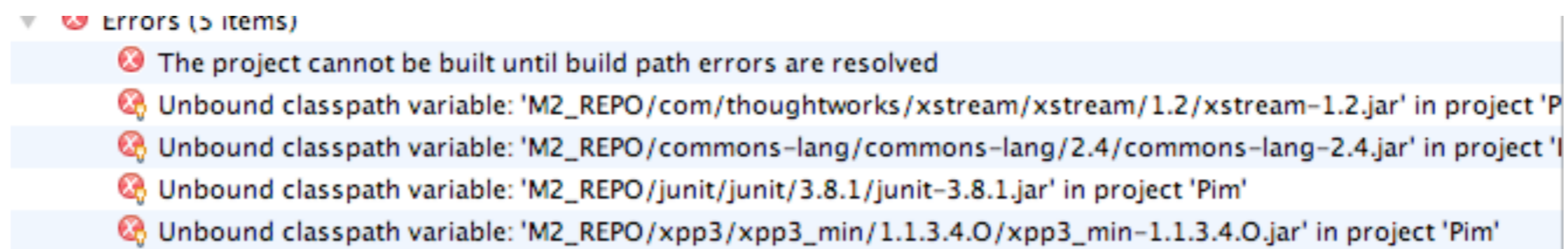
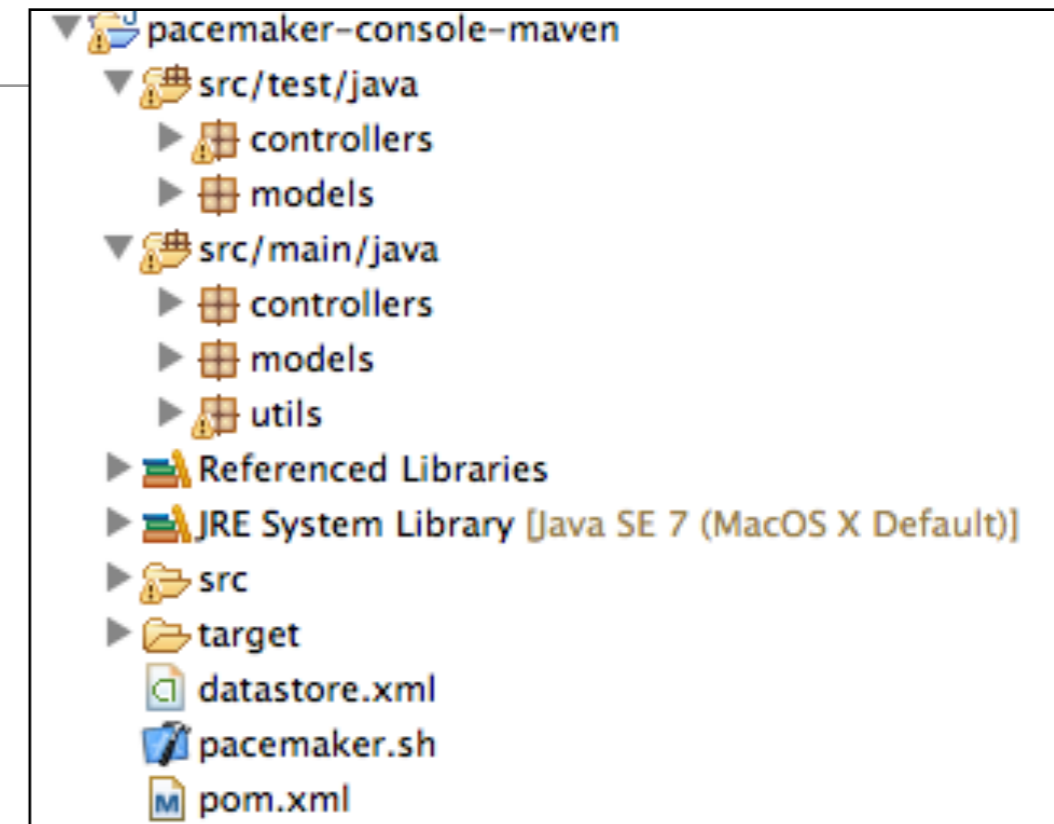
```
1. bash
r/repos/modules/agile/prj/pacemaker-console-maven.
[INFO]
    Sources for some artifacts are not available.
    Please run the same goal with the -DdownloadSources=true parameter in o
rder to check remote repositories for sources.
    List of artifacts without a source archive:
    o asg-cliche:asg-cliche:1.0
    o com.google.guava:guava:15.0
    o com.thoughtworks.xstream:xstream:1.4.4
    o xmlpull:xmlpull:1.1.3.1
    o xpp3:xpp3_min:1.1.4c
    o junit:junit:4.11
    o org.hamcrest:hamcrest-core:1.3

    Javadoc for some artifacts is not available.
    Please run the same goal with the -DdownloadJavadocs=true parameter in
order to check remote repositories for javadoc.
    List of artifacts without a javadoc archive:
    o asg-cliche:asg-cliche:1.0
    o com.google.guava:guava:15.0
    o com.thoughtworks.xstream:xstream:1.4.4
    o xmlpull:xmlpull:1.1.3.1
    o xpp3:xpp3_min:1.1.4c
    o junit:junit:4.11
    o org.hamcrest:hamcrest-core:1.3

[INFO] -----
-
[INFO] BUILD SUCCESS
[INFO] -----
-
[INFO] Total time: 1.593s
[INFO] Finished at: Thu Oct 17 10:20:11 IST 2013
[INFO] Final Memory: 10M/245M
[INFO] -----
-
grendel3:pacemaker-console-maven edeleastar$
```

Eclipse Project Structure

- This will have produced the required .classpath and .project files in the pim directory. You can now import this project into eclipse using the normal import->project menus.
- Although will import successfully, it will not build

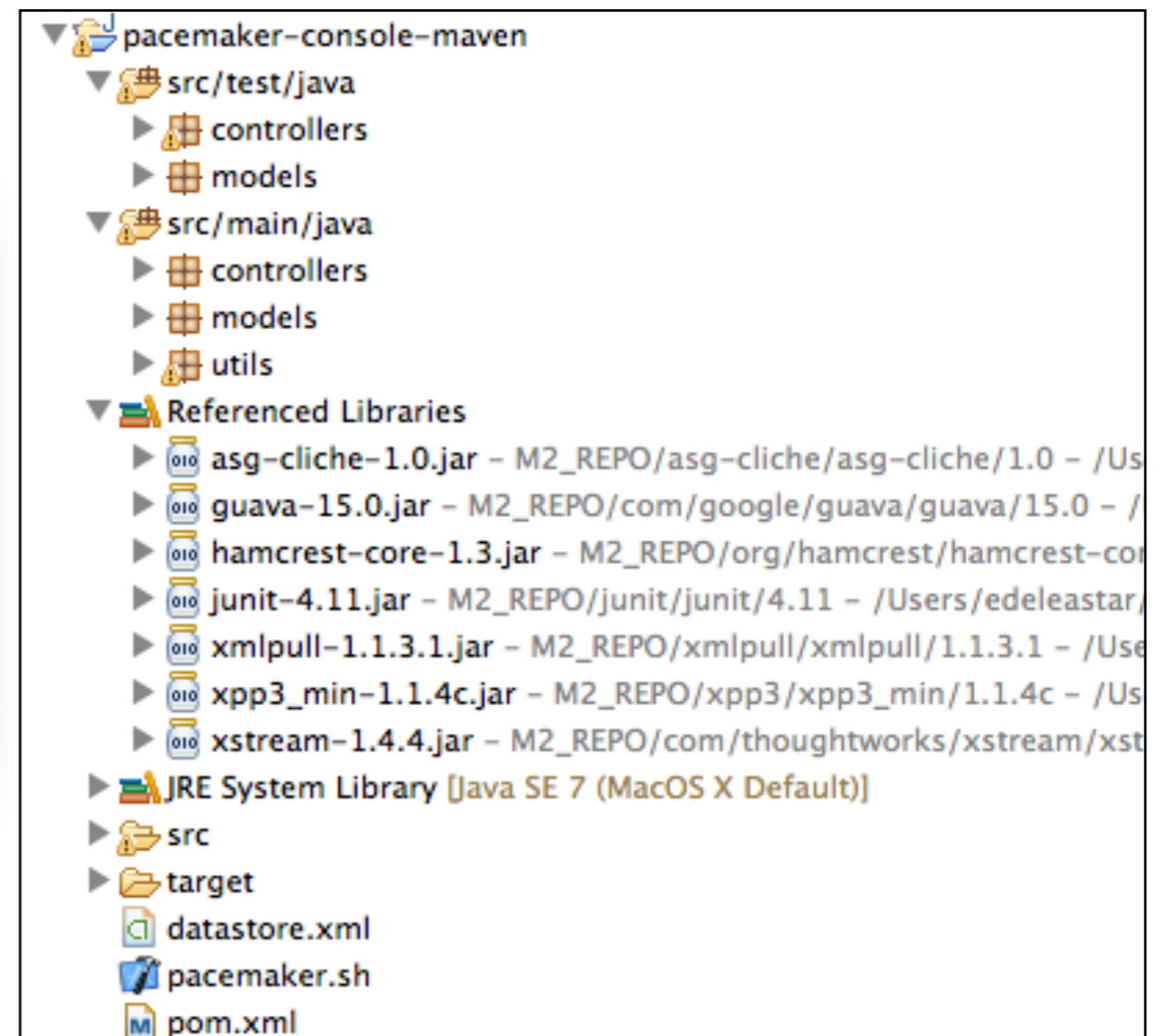
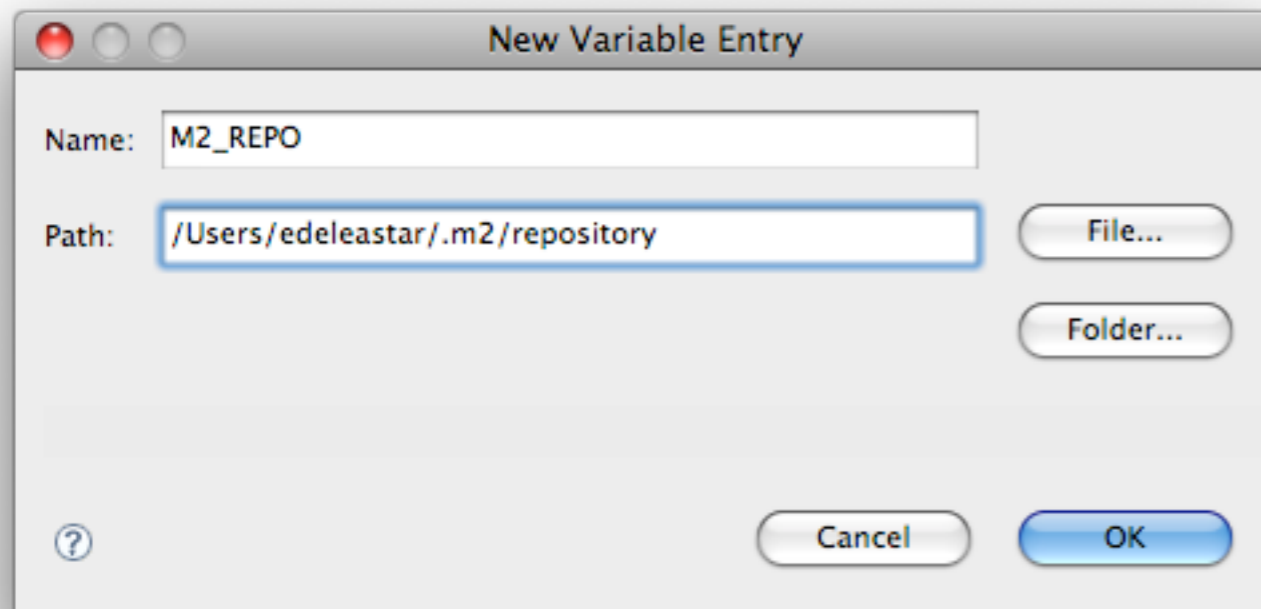


Eclipse .classpath and .project files

```
<classpath>
  <classpathentry kind="src" path="src/test/java" output="target/test-classes" including="**/*.java"/>
  <classpathentry kind="src" path="src/main/java" including="**/*.java"/>
  <classpathentry kind="output" path="target/classes"/>
  <classpathentry kind="var" path="M2_REPO/asg-cliche/asg-cliche/1.0/asg-cliche-1.0.jar"/>
  <classpathentry kind="var" path="M2_REPO/com/google/guava/guava/15.0/guava-15.0.jar"/>
  <classpathentry kind="var" path="M2_REPO/org/hamcrest/hamcrest-core/1.3/hamcrest-core-1.3.jar"/>
  <classpathentry kind="var" path="M2_REPO/junit/junit/4.11/junit-4.11.jar"/>
  <classpathentry kind="var" path="M2_REPO/xmlpull/xmlpull/1.1.3.1/xmlpull-1.1.3.1.jar"/>
  <classpathentry kind="var" path="M2_REPO/xpp3/xpp3_min/1.1.4c/xpp3_min-1.1.4c.jar"/>
  <classpathentry kind="var" path="M2_REPO/com/thoughtworks/xstream/xstream/1.4.4/xstream-1.4.4.jar"/>
  <classpathentry kind="con" path="org.eclipse.jdt.launching.JRE_CONTAINER"/>
</classpath>
```

```
<projectDescription>
  <name>pacemaker-console-maven</name>
  <projects/>
  <buildSpec>
    <buildCommand>
      <name>org.eclipse.jdt.core.javabuilder</name>
    </buildCommand>
  </buildSpec>
  <natures>
    <nature>org.eclipse.jdt.core.javanature</nature>
  </natures>
</projectDescription>
```

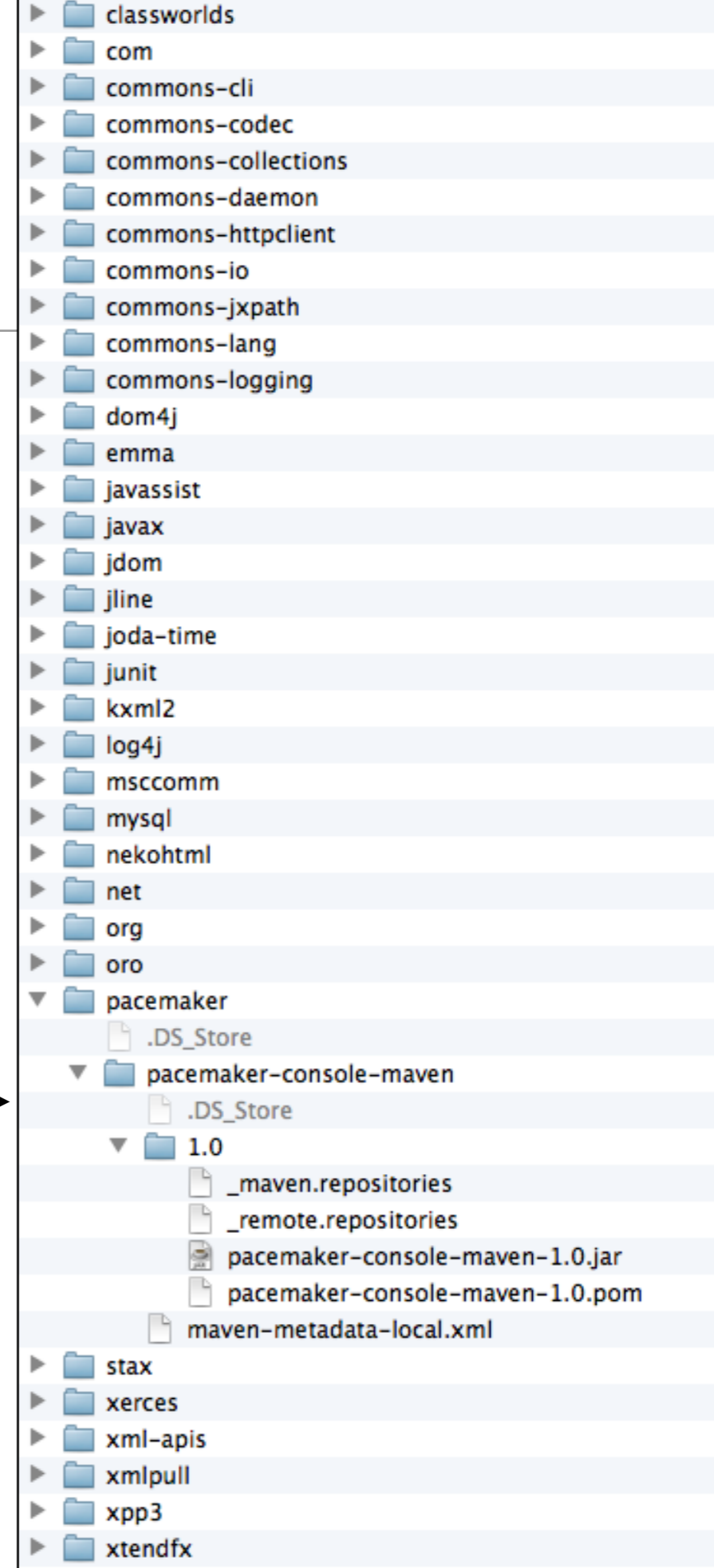
Eclipse Variable Definition



Install

- Copies the generated package to the local Maven repository

```
mvn install
```



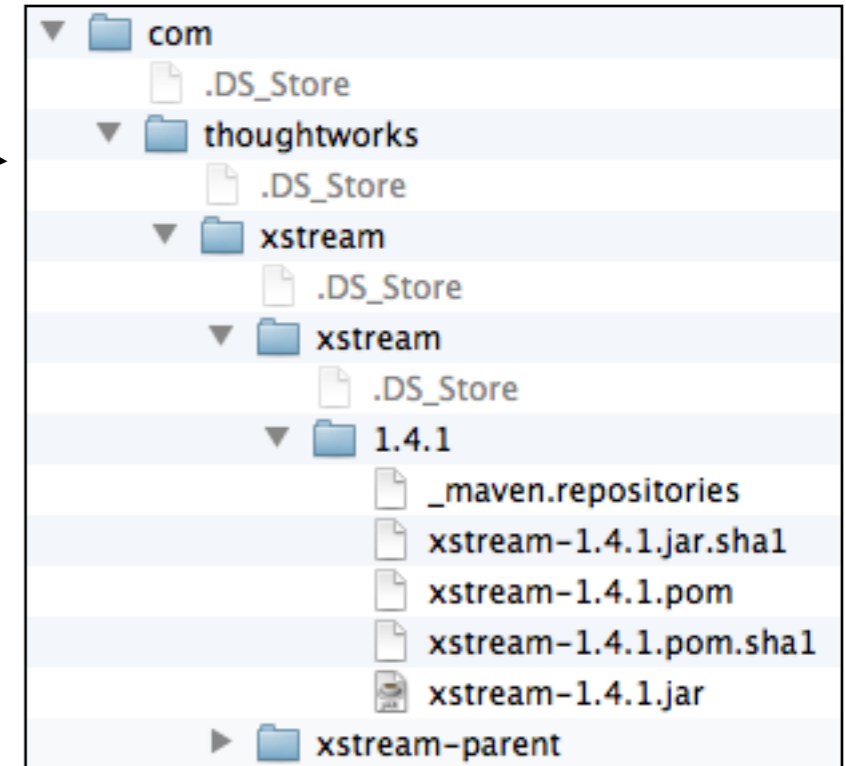
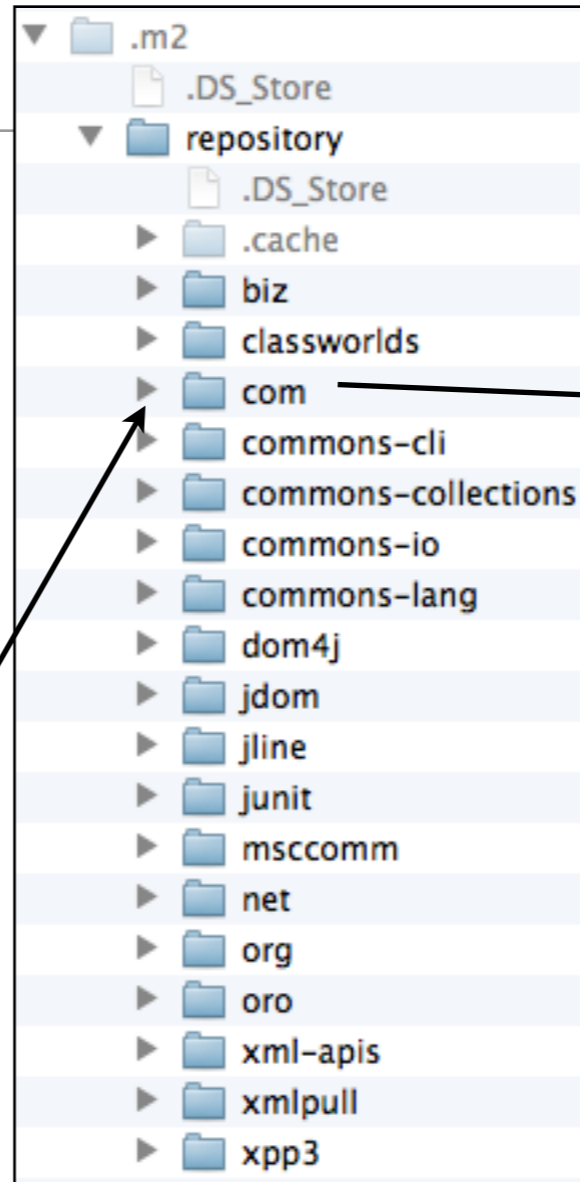
Dependencies also in Repositories

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

```
<dependency>  
  <groupId>com.google.guava</groupId>  
  <artifactId>guava</artifactId>  
  <version>15.0</version>  
</dependency>
```

```
<dependency>  
  <groupId>com.thoughtworks.xstream</groupId>  
  <artifactId>xstream</artifactId>  
  <version>1.4.4</version>  
</dependency>
```

```
<dependency>  
  <groupId>asg-cliche</groupId>  
  <artifactId>asg-cliche</artifactId>  
  <version>1.0</version>  
</dependency>  
</dependencies>
```



Launch Script - Unix

- Maven Repository

- Classpath for pacemaker

- Launch Command

```
M2_REPO=/Users/edelestar/.m2/repository
```

```
export CLASSPATH=%CLASSPATH%:\
```

```
${M2_REPO}/pacemaker/pacemaker-console-maven/1.0/pacemaker-console-maven-1.0.jar:\
```

```
${M2_REPO}/com/thoughtworks/xstream/xstream/1.4.4/xstream-1.4.4.jar:\
```

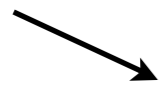
```
${M2_REPO}/com/google/guava/guava/15.0/guava.15.0.jar:\
```

```
${M2_REPO}/asg-cliche/asg-cliche/1.0/asg-cliche-1.0.jar
```

```
java controllers.Main
```

Launch Script - Windows

Maven
Repository



Classpath
for
pacemaker



Launch
Command



```
echo off
set M2_REPO=C:/docume1/eamonn1/.m2/repository

set CLASSPATH=
call cpappend.bat "%M2_REPO%/pacemaker/pacemaker-console-maven/1.0/pacemaker-console-maven-1.0.jar;"
call cpappend.bat "%M2_REPO%/com/thoughtworks/xstream/xstream/1.4.4/xstream-1.4.4.jar;"
call cpappend.bat "%M2_REPO%/com/google/guava/guava/15.0/guava.15.0.jar;"
call cpappend.bat "%M2_REPO%/asg-cliche/asg-cliche/1.0/asg-cliche-1.0.jar;"

java controllers.Main
```

```
set CLASSPATH=%CLASSPATH%;%~f1
```

cppappend.bat

```
grendel3:pacemaker-console-maven edeleastar$ sh ./pacemaker.sh
Welcome to pacemaker-console - ?help for instructions
pm> ?la
abbrev name      params
!rs     !run-script    (filename)
!el     !enable-logging (fileName)
!dl     !disable-logging ()
!gle    !get-last-exception ()
!sdt    !set-display-time (do-display-time)
?l      ?list ()
?l      ?list (startsWith)
?h      ?help ()
?h      ?help (command-name)
?la     ?list-all ()
?ghh    ?generate-HTML-help (file-name, include-prefixed)
cu      create-user    (first name, last name, email, password)
gu      get-user       (email)
gu      get-users      ()
du      delete-user    (email)
aa      add-activity   (user-id, type, location, distance)
al      add-location   (activity-id, latitude, longitude)
pm>
```



maven-android-plugin

(renamed to android-maven-plugin) Easy to use Maven plugin for Android™!

- [Project Home](#)
- [Wiki](#)**
- [Issues](#)
- [Source](#)

Beginner

- [Getting Started](#)
- [Samples](#)
- [Tips and Tricks](#)
- [Debugging](#)
- [Eclipse Integration](#)
- [QuickStartForEclipseProject](#)
- [Glossary](#)

Advanced

- [Google Add Ons \(Maps\)](#)
- [AutomatedScreenshots](#)
- [Shrink, Obfuscate and Optimize with Proguard](#)
- [Signing the APK](#)
- [Zipalign the APK](#)
- [Automated Android Tests](#)
- [CombiningApplicationAndTestProjectTo](#)
- [Android Library Projects \(apklib\)](#)
- [ApkSourcesDependency](#)
- [SynchronisingVersionWithGmaven](#)
- [Native Development](#)
- [Native Libraries as Dependencies](#)
- [Scala](#)

Reference

- [Documentation](#)
- [PluginRenamed](#)
- [Repositories](#)
- [Links, blogs and more](#)
- [Presentations](#)
- [People Using This](#)
- [Changelog](#)

Contributors, Developers..

- [Building the plugin from source](#)
- [More about source and contributing](#)
- [SnapshotRepo](#)
- [DeploymentInstructions](#)
- [OtherProjects](#)

Documentation

Documentation collection for Android Maven Plugin
Featured

Maven: The Complete Reference - Android chapter

Documentation in book form is available with the Android chapter of the free book Maven: The complete R
<http://www.sonatype.com/books/mvnref-book/reference/android-dev.html>

Wiki Documentation

Start here

- [GettingStarted](#) quickly!
- See some [Samples](#) of how to use Android Maven Plugin
- [Tips and tricks](#) for using the plugin even easier

Eclipse

- [EclipseIntegration](#) - help collecti
- [QuickStartForEclipseProject](#)

Reference

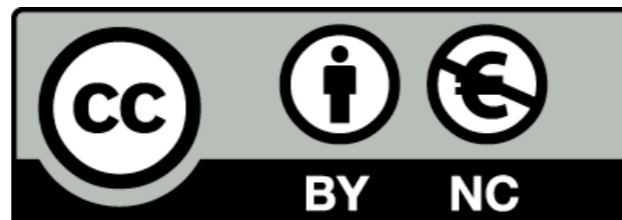
- [Maven Site](#) - Auto-generated doc
 - [Plugin goals reference](#) - All
- See [AddOns](#) for Building for And
- See [ApkLib](#) for how to bring in o
- Signing**
 - See [SigningAPKWithMaven](#)
 - See [ZipalignAPKBuiltByMA](#)

See also

- Take part in the mailing list [Mav](#)

Comment by XXXX, May 2, 2013

I found the Maven android plugin on several plugin collection page, so I thought this will be a resource I should use, but I gave up, because I didn't even found out what is this in a few minutes. You should really write at least one paragraph about this project for the newbies like me. I am not too lazy to put time in learning, but I will not spend time for something which has no real documentation, and I will presume there is none, if I didn't found out what is this on the project homepage.



Except where otherwise noted, this content is licensed under a Creative Commons Attribution-NonCommercial 3.0 License.

For more information, please see <http://creativecommons.org/licenses/by-nc/3.0/>

