Maven

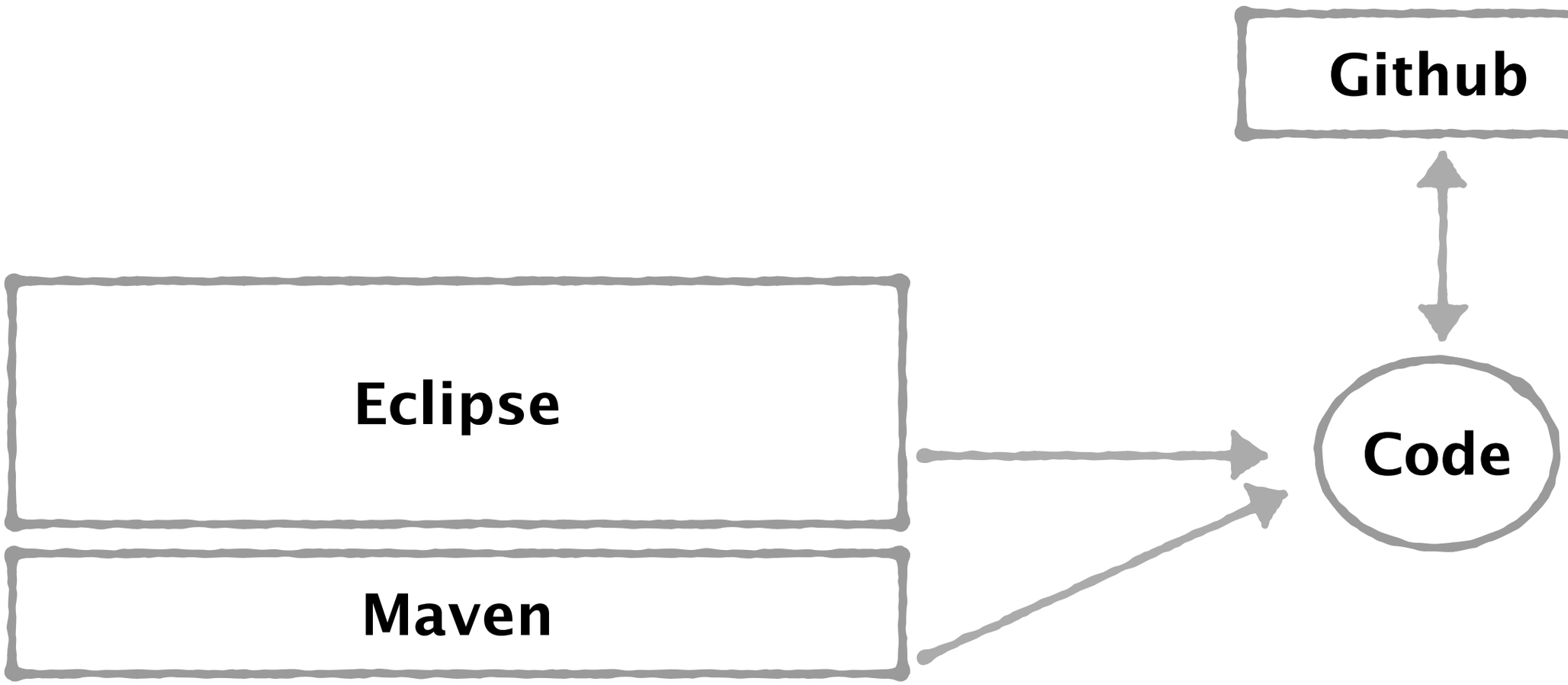
brief introduction

Bledar Aga
http://scg.unibe.ch/wiki/students/bledaraga

b
UNIVERSITÄT
BERN

environment

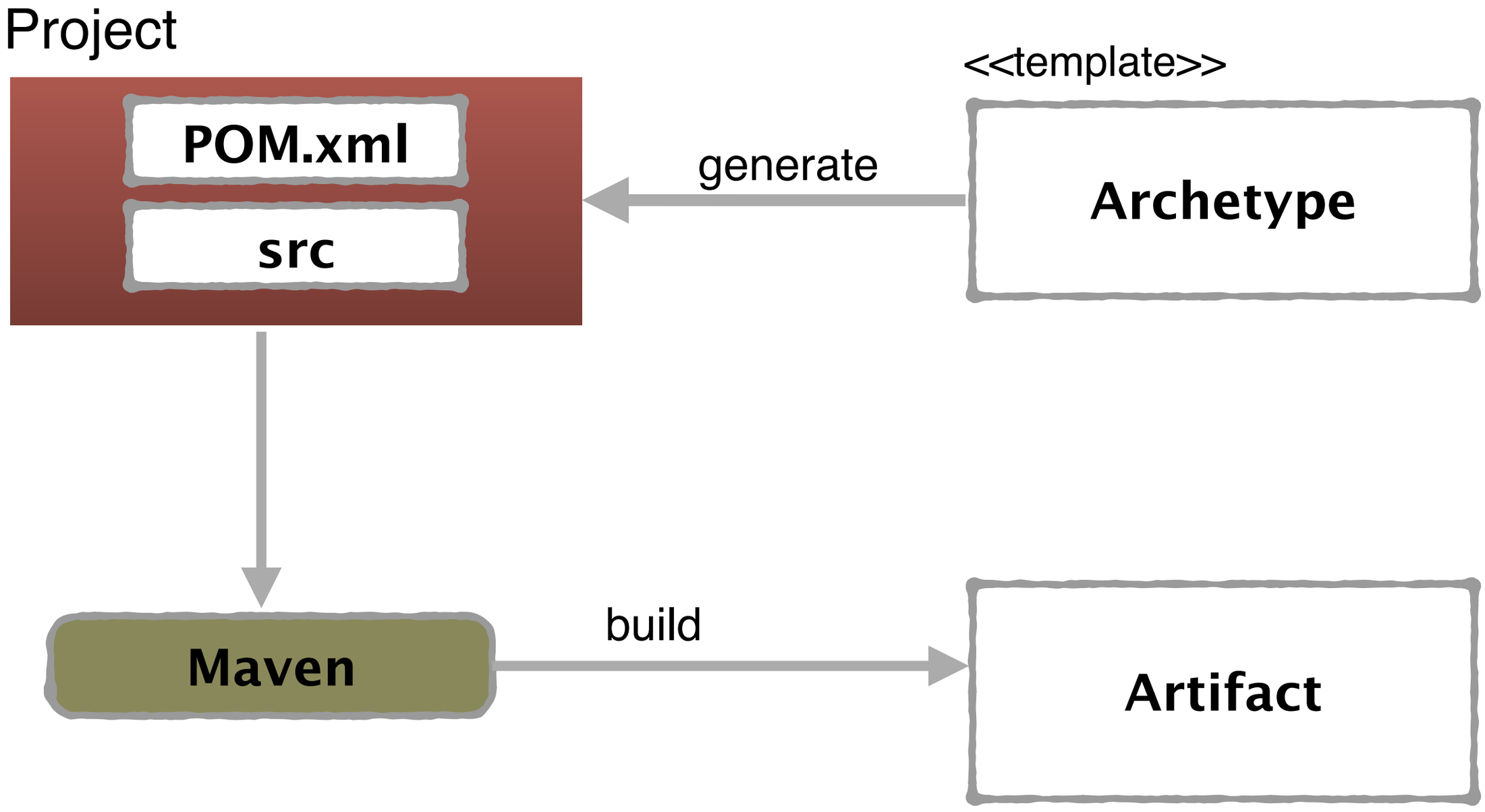
- eclipse: IDE
- github: versioning
- maven: build automation



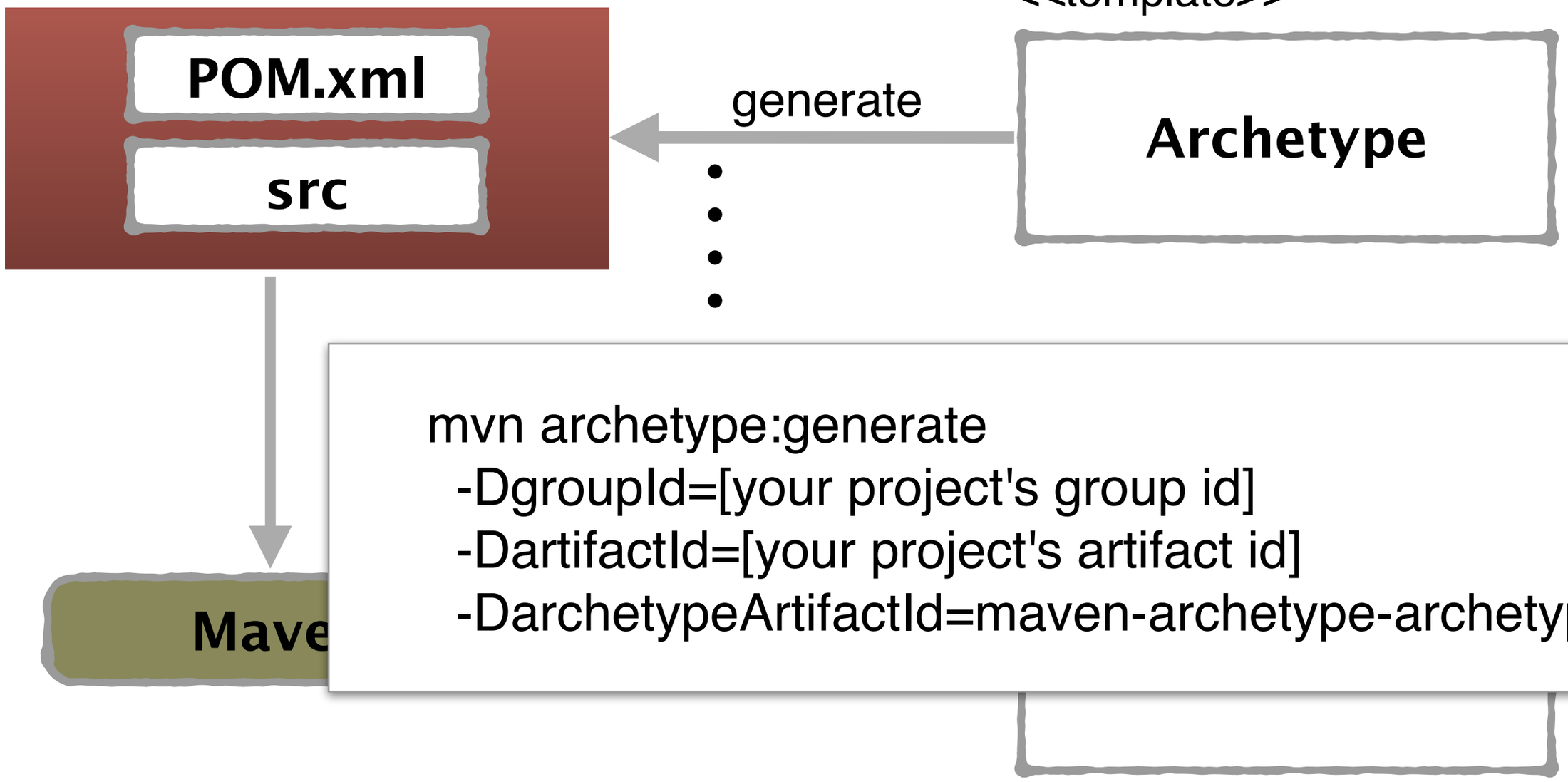
Maven

what is maven?

- Builds
- Documentation
- Reporting
- Dependencies
- Releases
- Distribution


Project
POM.xml
src
generate
<<template>>
Archetype
Maven
build
Artifact

Project

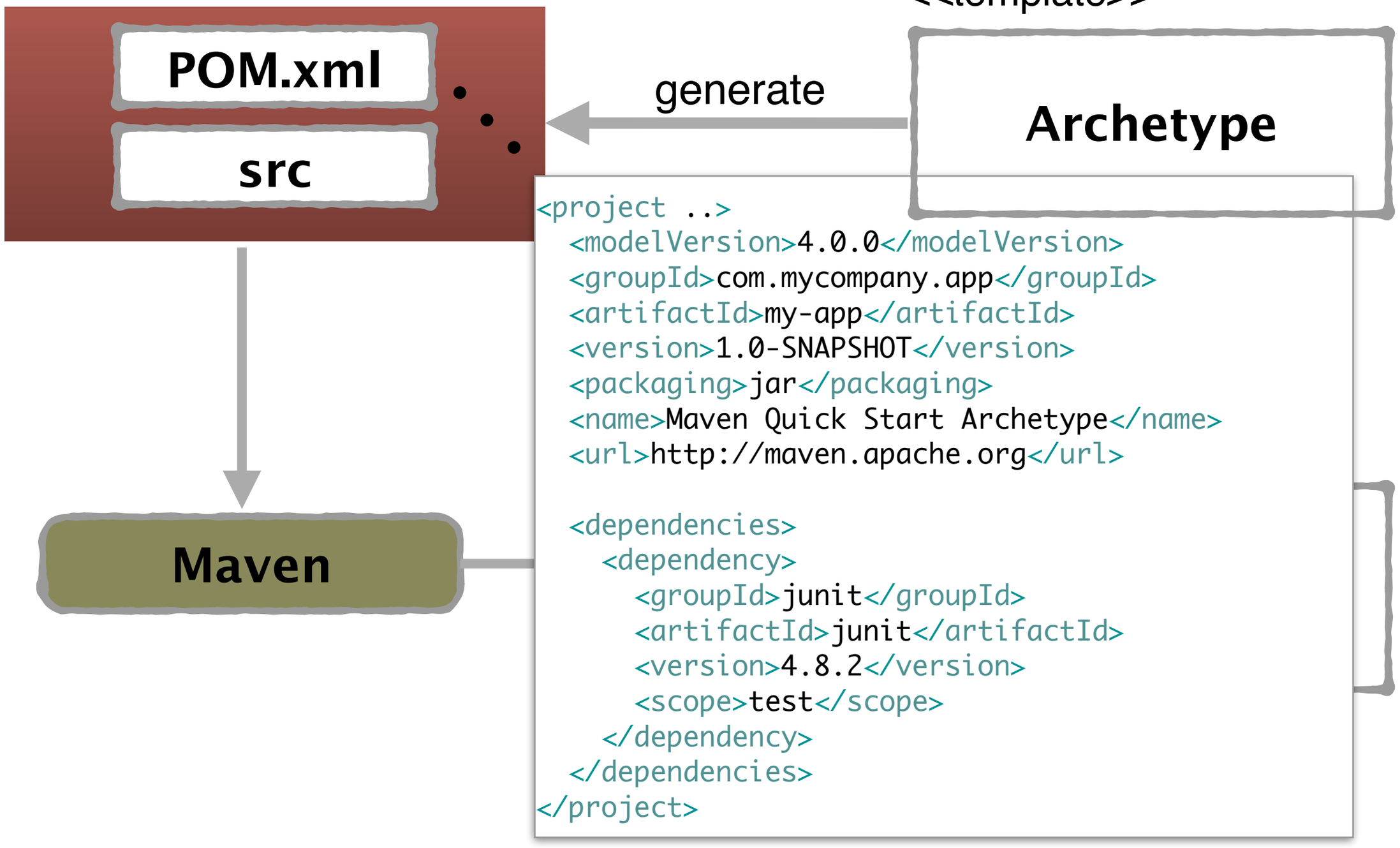

POM.xml
src
generate
<<template>>
Archetype
Mave
mvn archetype:generate
-DgroupId=[your project's group id]
-DartifactId=[your project's artifact id]
-DarchetypeArtifactId=maven-archetype-archetype

Maven

archetype:

- archetype is a Maven project templating toolkit
  - maven project template is a good starting point
- contains the **project prototype** you wish to create
- once you created and deployed, it can be used by all developers within your organization

Project

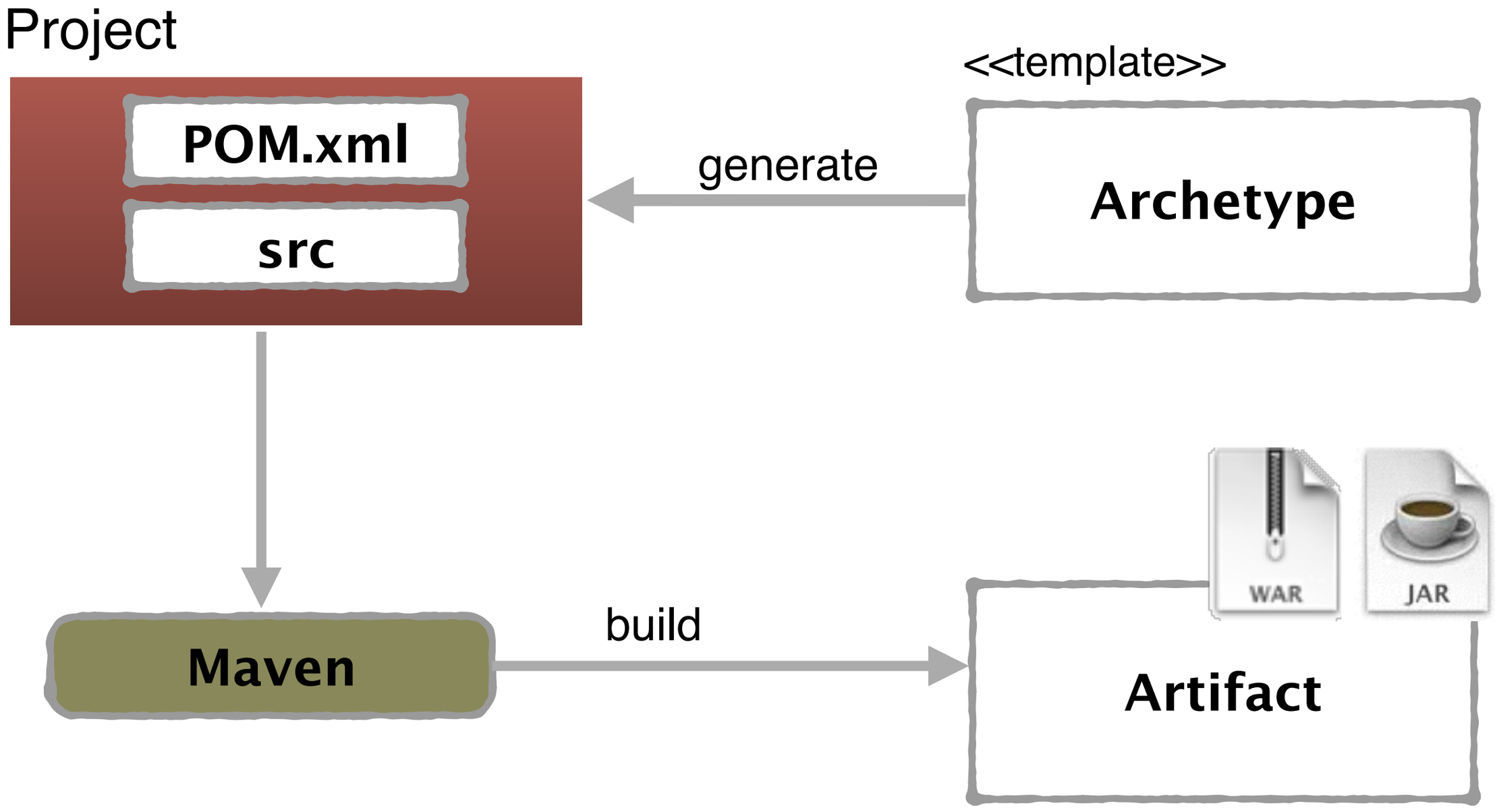

POM.xml
src
<<template>>
generate
Archetype
Maven
<project ..>
<modelVersion>4.0.0</modelVersion>
<groupId>com.mycompany.app</groupId>
<artifactId>my-app</artifactId>
<version>1.0-SNAPSHOT</version>
<packaging>jar</packaging>
<name>Maven Quick Start Archetype</name>
<url>http://maven.apache.org</url>
<dependencies>
<dependency>
<groupId>junit</groupId>
<artifactId>junit</artifactId>
<version>4.8.2</version>
<scope>test</scope>
</dependency>
</dependencies>
</project>

Maven

POM:

- project object model: pom.xml
- provides a uniform build system
- it is the core of the project's configuration in Maven

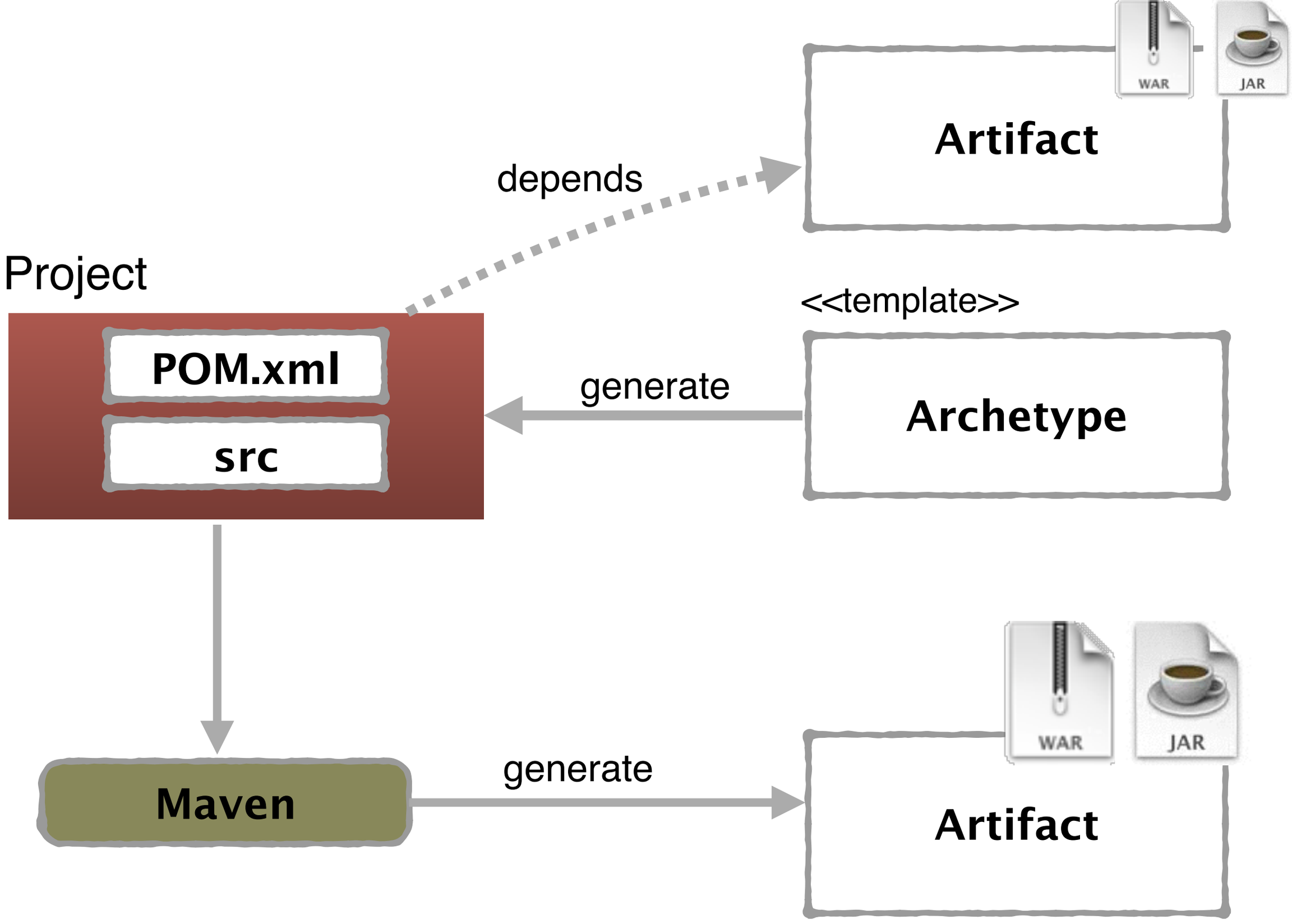
```
<project ..>
  <modelVersion>4.0.0</modelVersion>
  <groupId>com.mycompany.app</groupId>
  <artifactId>my-app</artifactId>
  <version>1.0-SNAPSHOT</version>
  <packaging>jar</packaging>
  <name>Maven Quick Start Archetype</name>
  <url>http://maven.apache.org</url>

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


Project
POM.xml
src
generate
<<template>>
Archetype
Maven
build
WAR
JAR
Artifact

Maven

artifact:

- is the result of the Maven build
- usually a Jar/War file
- is uniquely identified by:
  - a groupID: usually a reversed domain name, like ch.unibe.scg.ese
  - an artifactID: name of the application
  - and a version string


WAR
JAR
Artifact
depends
Project
POM.xml
src
<<template>>
generate
Archetype
Maven
generate
WAR
JAR
Artifact

Maven

dependency:

- most every project depends upon others to build and run correctly
- maven downloads and links the dependencies on compilation or other goal that require them
- dependencies create the cornerstone of the POM

```
[…]
   <dependencies>
      <dependency>
         <groupId>junit</groupId>
         <artifactId>junit</artifactId>
         <version>4.0</version>
         <type>jar</type>
         <scope>test</scope>
         <optional>true</optional>
      </dependency>
      ...
   </dependencies>
[…]
```

Maven

plugin:

- Maven is at its heart a plugin execution framework
  - all the work is done by plugins
- in practical plugins are used to:
  - create jar/war files, compile code, unit testing code, create documentation and so on…
- there are two categories of plugins:
  - build plugins: executed during the build, configured in the <build/> element
  - reporting plugins: executed during the site generation, configured in the <reporting/> element

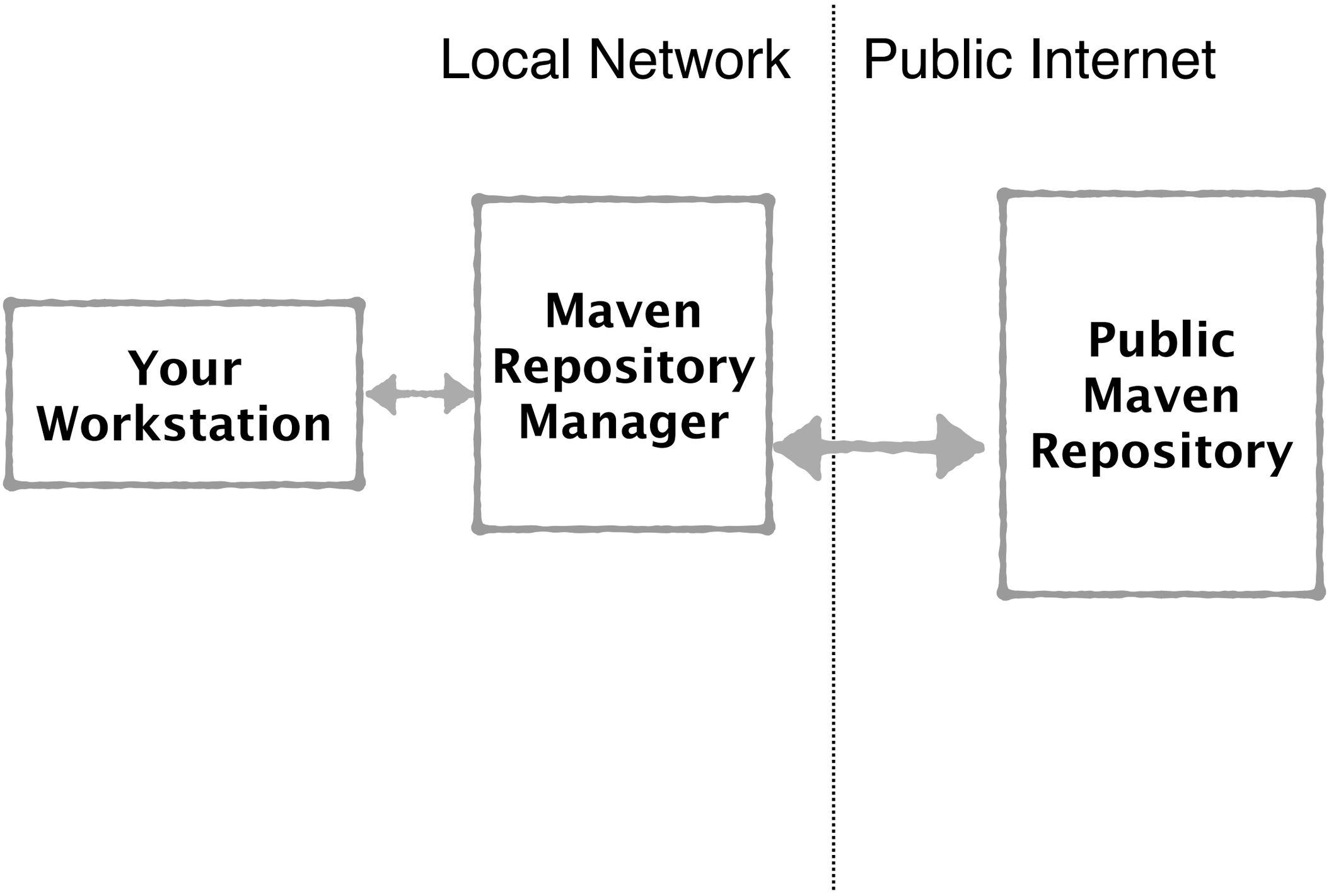
Maven

plugin:

- example: clean plugin attempts to clean the files and directories generated by Maven during its build
- to run the plugin: mvn clean:clean

```
[...]
  <build>
    <plugins>
      <plugin>
        <artifactId>maven-clean-plugin</artifactId>
        <version>2.5</version>
        <executions>
          <execution>
            <id>auto-clean</id>
            <phase>initialize</phase>
            <goals>
              <goal>clean</goal>
            </goals>
          </execution>
        </executions>
      </plugin>
    </plugins>
  </build>
[...]
```

Maven


Local Network
Public Internet
Your Workstation
Maven Repository Manager
Public Maven Repository

Maven

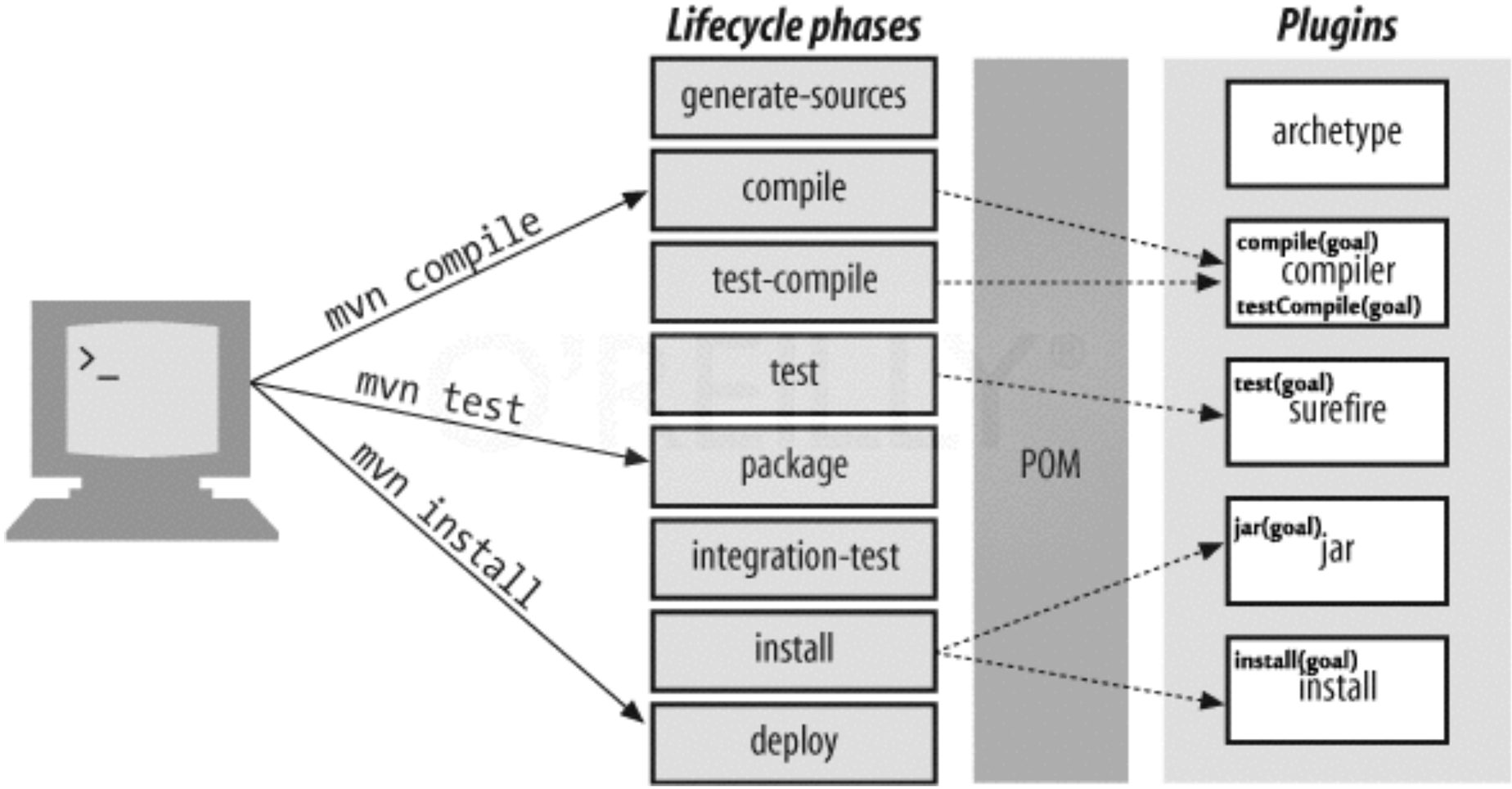
repository:

- arbitrary and accessible location designed to store the artifacts that maven builds
- repository type:
  - **public** -> central repository where all artifacts are stored
  - **local** -> downloaded artifacts used for build (*HOME/.m2/repository*)
  - **remote** -> centralized repository with artifacts used for build (*local network; for big organizations*)
    - Example: http://www.sonatype.org/nexus/

Maven

lifecycle:

- three build-in build life cycles:
  - **default**, clean and site
- the default build lifecycle is made up of following phases:



Maven

lifecycle:

- the phases are executed sequentially to complete the default lifecycle
- mvn deploy: all phases are executed
- mvn install deploy: the phases are executed twice until install and at the end deploy once

Maven

demo:

- how to build simple java app -> from command line
- how to build simple java app -> from eclipse

Maven

references:

- get started: http://maven.apache.org/guides/getting-started/
- archetype: http://maven.apache.org/guides/introduction/introduction-to-archetypes.html
- POM: http://maven.apache.org/guides/introduction/introduction-to-the-pom.html
- dependencies: http://maven.apache.org/pom.html
- plugin: http://maven.apache.org/plugins/maven-dependency-plugin/
- concept of repositories: http://docs.codehaus.org/display/MAVENUSER/Maven+Concepts+Repositories
- lifecyle: http://maven.apache.org/guides/introduction/introduction-to-the-lifecycle.html