

# Web 4

Maven



# BUILD TOOLS

- Ant

→ can be started from IDE

- Maven

→ but is executed *INDEPENDENT* from IDE

- Gradle

- sbt (Scala)

- Rake (Ruby)

- ...

→ so can also be used on a server, ...



<http://maven.apache.org/>

# POM.XML

## MAVEN BUILD SCRIPT

```
<?xml version="1.0" encoding="UTF-8"?>
<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>uc11.web3.demos</groupId>
  <artifactId>maven-demo</artifactId>
  <version>1.0-SNAPSHOT</version>
  <name>MavenDemo</name>
  <packaging>war</packaging>
```

Is that all?  
Convention over Configuration!

# POM

- = Project Object Model
- groupId
  - uniquely identifies your project across all projects
  - should follow Java's package name rules
- artifactId
  - is the name of the jar without version
  - name you want with lowercase letters and no strange symbols
- version
  - if you distribute it, then you can choose any typical version with numbers and dots (1.0, 1.1, 1.0.1, ...)

# CONVENTION OVER CONFIGURATION

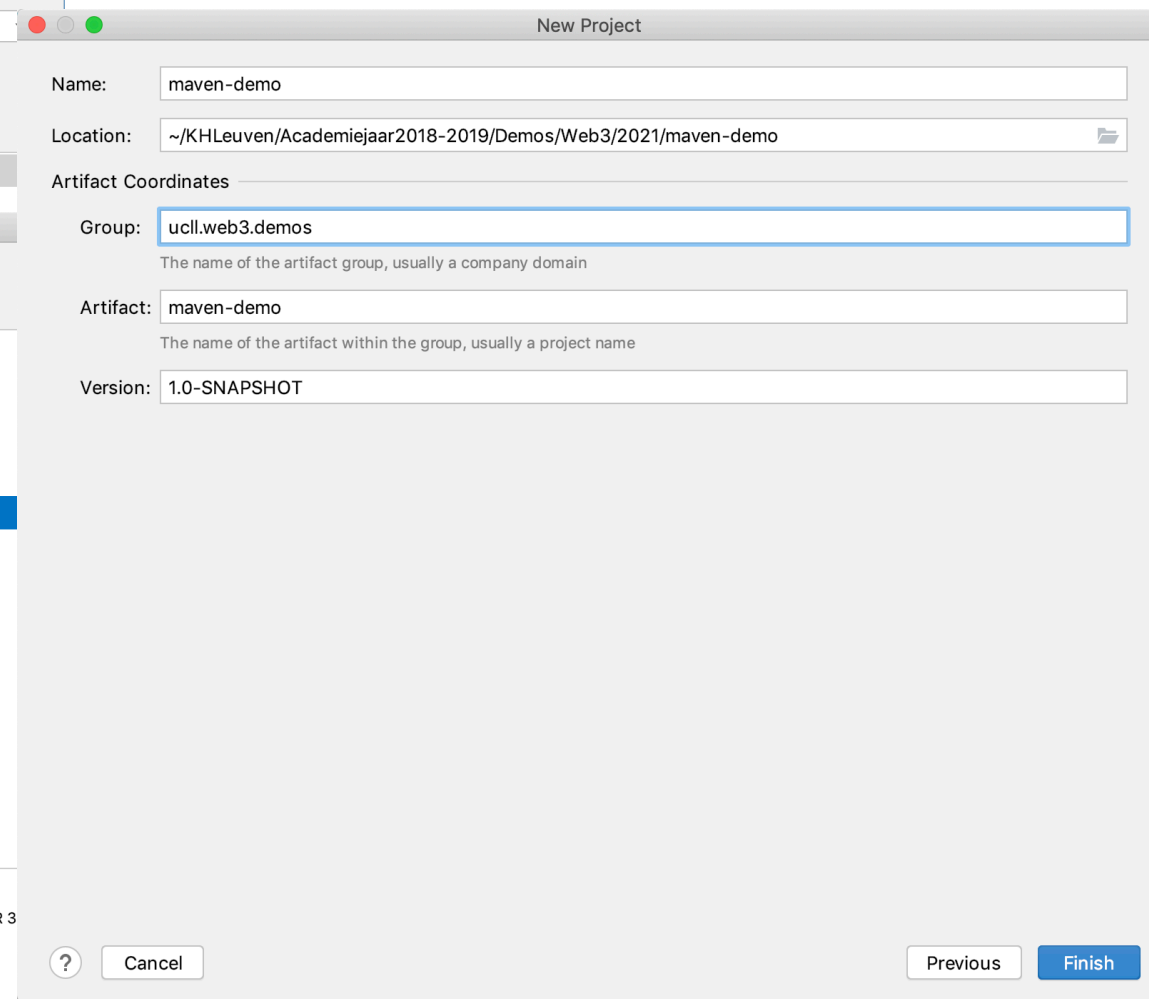
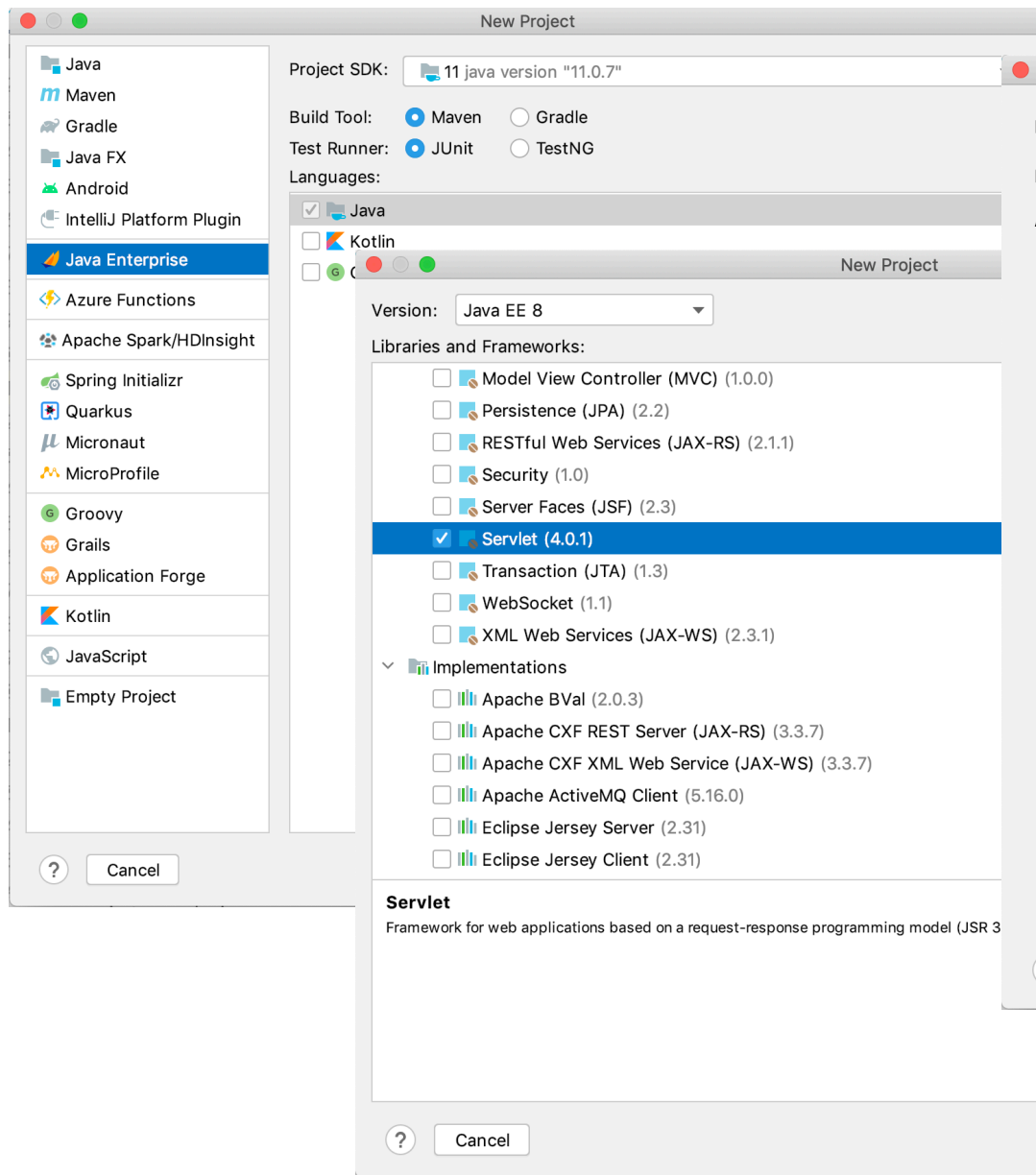
- You do not have to write anything if you
  - don't need anything extra and
  - use the project structure Maven expects

**Start Demo**

CREATE MAVEN PROJECT







IntelliJ IDEA File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help Thu 12:59 Elke Steegmans

MavenDemo - pom.xml (maven-demo)

Week01\_Maven > m pom.xml

Project Week01\_Maven [MavenDemo] ~/KHL

- .idea
- src
  - main
    - java
    - resources
  - webapp
    - WEB-INF
      - web.xml
  - test
    - java
    - resources
- MavenDemo.iml
- m pom.xml
- External Libraries
- Scratches and Consoles

```
<?xml version="1.0" encoding="UTF-8"?>
<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/maven-v4_0_0.xsd">
  <modelVersion>4.0.0</modelVersion>

  <groupId>ucll.web3.demos</groupId>
  <artifactId>maven-demo</artifactId>
  <version>1.0-SNAPSHOT</version>
  <name>MavenDemo</name>
  <packaging>war</packaging>

  <properties>
    <maven.compiler.target>1.8</maven.compiler.target>
    <maven.compiler.source>1.8</maven.compiler.source>
    <junit.version>5.6.2</junit.version>
  </properties>

  <dependencies>
    <dependency>
      <groupId>javax.servlet</groupId>
      <artifactId>javax.servlet-api</artifactId>
      <version>4.0.1</version>
      <scope>provided</scope>
    </dependency>
  </dependencies>
</project>
```

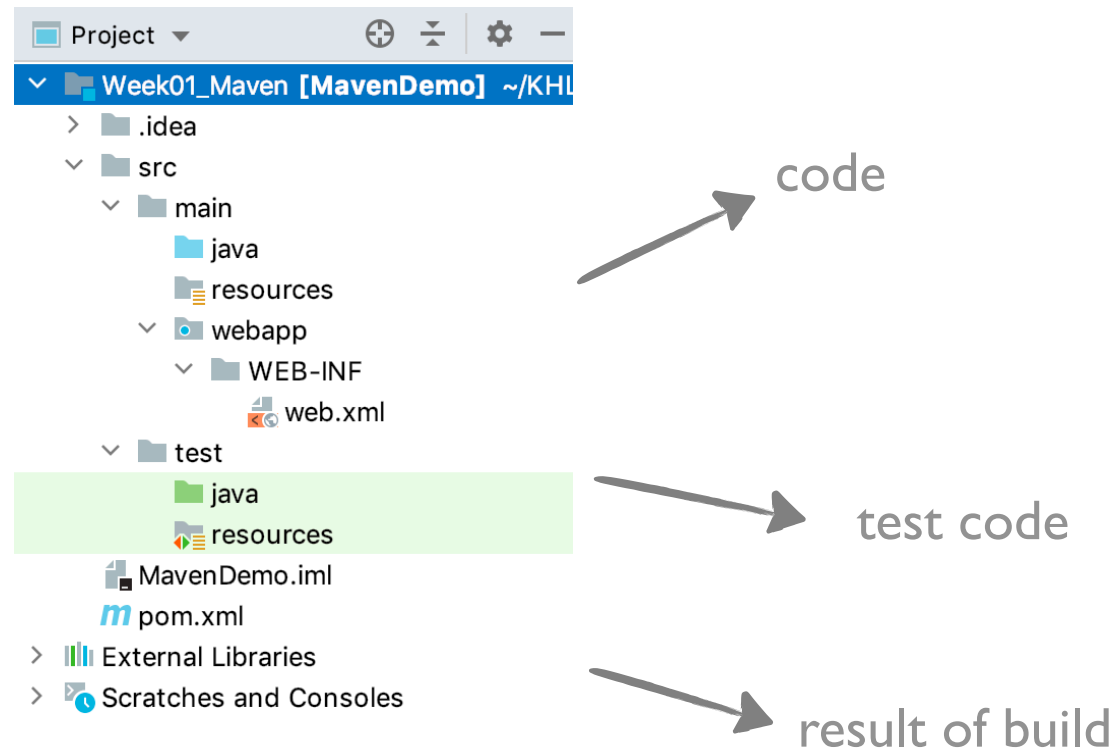
Maven

- MavenDemo
  - Lifecycle
    - clean
    - validate
    - compile
    - test
    - package
    - verify
    - install
    - site
    - deploy
  - Plugins
  - Dependencies

6: Problems Terminal Azure Activity Log TODO Azure Streaming Log Cosmos Serverless Spark Jobs 8: Services Build Java Enterprise Event Log

12:1 LF UTF-8 4 spaces


# MAVEN PROJECT STRUCTURE

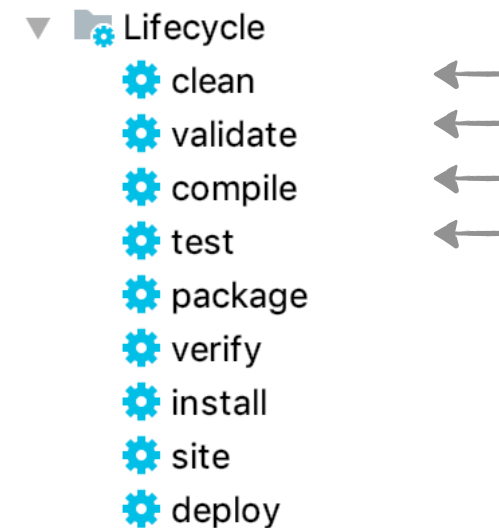


# MAVEN BUILD



- Central concept in Maven: *build lifecycle*
- A build lifecycle is made up of *phases*
- A phase is a step of the build proces

- Example: maven **package**  phase
  - build script is executed
  - phase *package* will be executed
  - but previous phases are executed first
    - all goals of these phases will be executed



# DEPENDENCY MANAGEMENT

- Manage libraries needed for
  - compiling
  - testing
  - running

JUnit, JSTL, PostgreSQL, ...

# DEPENDENCY MANAGEMENT

```
<?xml version="1.0" encoding="UTF-8"?>
<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>uc11.web3.demos</groupId>
  <artifactId>maven-demo</artifactId>
  <version>1.0-SNAPSHOT</version>
  <name>MavenDemo</name>
  <packaging>war</packaging>
```

```
<dependencies>
  <dependency>
    <groupId>javax.servlet</groupId>
    <artifactId>javax.servlet-api</artifactId>
    <version>4.0.1</version>
    <scope>provided</scope>
  </dependency>
```

→ Jar is:

- downloaded and
- stored in local repository

Extra jars needed → dependency in pom.xml

```
<dependency>  
  <groupId>org.seleniumhq.selenium</groupId>  
  <artifactId>selenium-java</artifactId>  
  <version>3.141.59</version>  
</dependency>
```

**Start Demo**

Where did maven  
find the jar?!

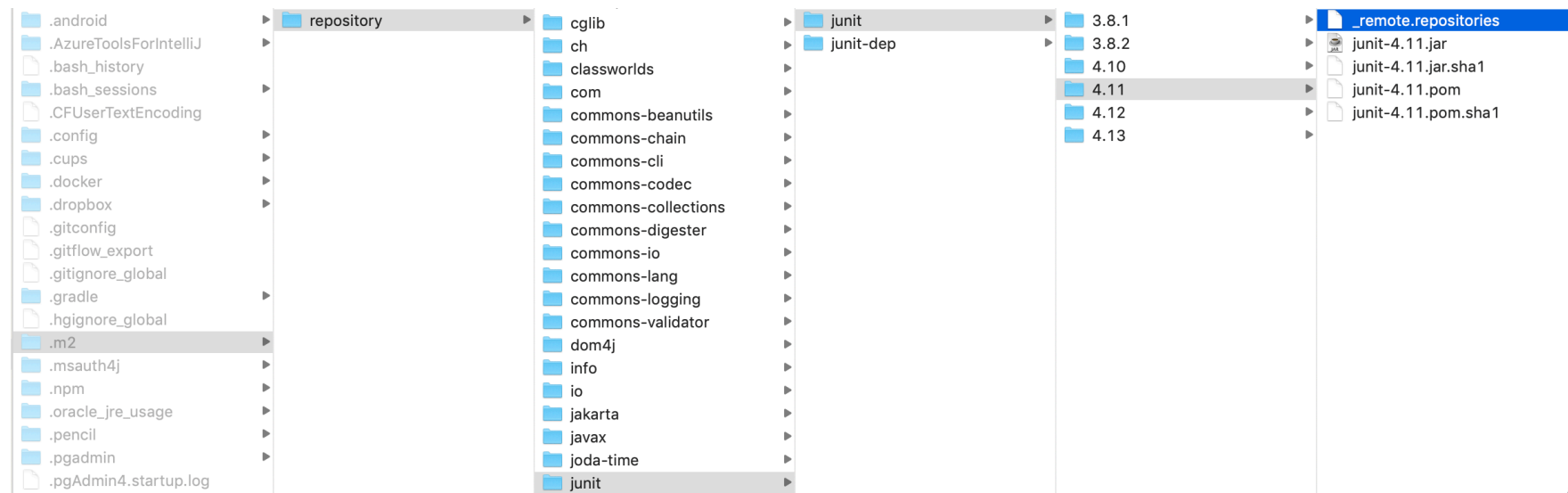
ADD SELENIUM



# REPOSITORIES

- = directory with all the jars needed, ...
- 3 types
  - *local*: on your computer
  - *central*: provided by Maven community
  - *remote*: custom

# LOCAL REPOSITORY



- In Windows: `C:\Documents and Settings\<your-username>\.m2`
- In Unix/Mac OS X: `~/ .m2`

# CENTRAL REPOSITORY

search.maven.org

The Central Repository

The Search Engine for

New: About Central

Advanced Se

Apache Maven Resources | About S

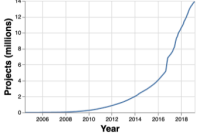
Apache and Apache Maven are trademarks of the Apache Software Foundation. The C complement Apache Maven and should not be confus

MVNREPOSITORY

Search for groups, artifacts, categories

Search

Indexed Artifacts (17.8M)



Popular Categories

Aspect Oriented

Actor Frameworks

Application Metrics

Build Tools

Bytecode Libraries

Command Line Parsers

Cache Implementations

Cloud Computing

Code Analyzers

Collections

Configuration Libraries

Core Utilities

Date and Time Utilities

Dependency Injection

Embedded SQL Databases

HTML Parsers

HTTP Clients

I/O Utilities

JDBC Extensions

JDBC Pools

Home » junit » junit

JUnit

JUnit is a unit testing framework for Java, created by Erich Gamma and Kent Beck.

License

EPL 1.0

Categories

Testing Frameworks

Tags

testing | junit

Used By

101,668 artifacts

Note: This artifact was moved to:

org.junit.jupiter » junit-jupiter-api

Central (30)

Redhat GA (3)

Redhat EA (3)

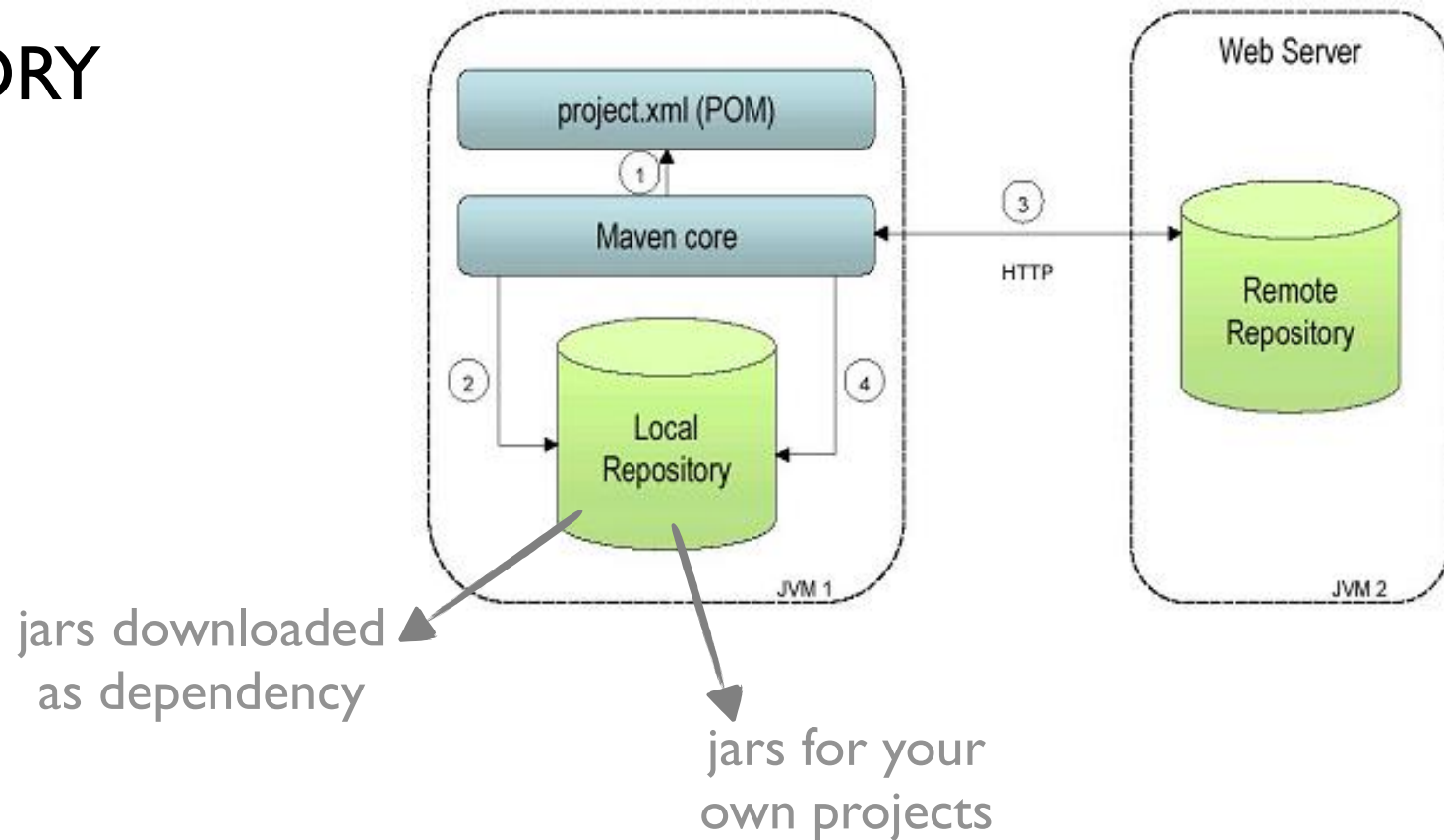
JBoss 3rd-party (1)

ICM (8)

Alfresco (1)

	Version	Repository	Usages	Date
4.13.x	4.13	Central	4,894	Jan, 2020
	4.13-rc-2	Central	55	Dec, 2019
	4.13-rc-1	Central	53	Oct, 2019
	4.13-beta-3	Central	294	May, 2019
	4.13-beta-2	Central	118	Feb, 2019
	4.13-beta-1	Central	67	Nov, 2018
4.12.x	4.12	Central	52,462	Dec, 2014
	4.12-beta-3	Central	30	Nov, 2014
	4.12-beta-2	Central	31	Sep, 2014

# REPOSITORY



- In Windows: `C:\Documents and Settings\<your-username>\.m2`
- In Unix/Mac OS X: `~/ .m2`

**Start Demo**

ADD JAVA+JSTL CODE



???