

Maven

Metalcon Workshop

Build lifecycle phases

1. validate
2. compile compile source files
3. test execute test cases (undeployed)
4. package create packed files, e.g. JAR
5. integration-test
6. verify
7. install
- ▼ 8. deploy copy into remote repositories



Create Java project

Practice Session

```
$ mvn archetype:generate  
-DgroupId=de.uniko.mc-ws  
-DartifactId=mvn-shell  
-DarchetypeArtifactId=maven-archetype-quickstart  
-DinteractiveMode=false
```



Project directory

- src/
 - main/java/ source files
 - test/java/ test cases/suites
- pom.xml project information



pom.xml

- general project information
- build plug-ins
- dependencies
- distribution information



Project information

POM

<project ...>

<modelVersion>4.0.0</...>

<groupId>de.uniko.mc-ws

<artifactId>mvn-shell

<version>1.0-SNAPSHOT

Maven Coordinates

...

root node

group identifier

project name

artifact version

developers, repositories, ...



Dependencies

POM

<project ...>

...

<dependencies>

<dependency>

<groupId>junit

<artifactId>junit

<version>4.5

<scope>test

</dependency>

list of projects depending on
single dependency

Maven Coordinates

transitivity limit



Where do I get Coordinates?

- from project distributor
- searchable Maven Central Repository



Work with Eclipse

```
$ mvn eclipse:eclipse
```

(+ import project)

or

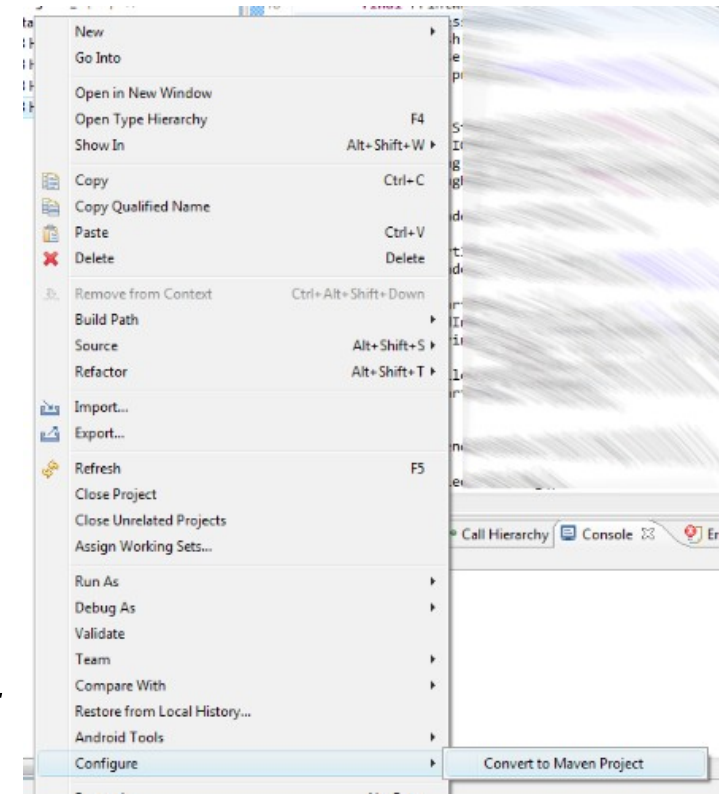
new general project

(+ *Mavenize* via
m2e)

Mavenize projects with m2e

Practice Session

- git: clone mvn-workshop.git
- Eclipse
 - set workspace
 - new general project
 - right-click on project
 - > *Configure*
 - > *Convert to Maven Project*



Missing dependency

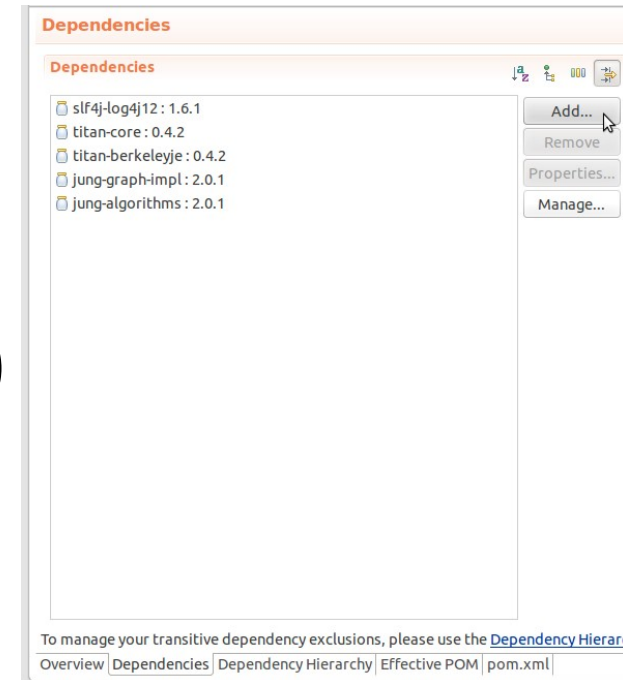
project can not build: missing interface
HttpServlet and others

```
<dependency>  
  <groupId>org.apache.tomcat.maven</groupId>  
  <artifactId>tomcat7-maven-plugin</artifactId>  
  <version>2.1</version>  
  <scope>provided</scope>  
</dependency>
```

Add a dependency

Practice Session

- open pom (text editor)
- add dependency manually
- or
- open pom (m2e POM-Editor)
- go to tab *Dependencies*
- click *Add...*



Maven build plug-ins

```
<project ...>
```

```
...
```

```
<build>
```

```
  <plugins>
```

```
    <plugin>
```

```
      <groupId>org.codehaus.mojo
```

```
      <artifactId>tomcat-maven-plugin
```

```
      <version>1.1
```

```
      <configuration>
```

```
        ...
```

```
      </configuration>
```

```
    </plugin>
```

list of build plugins (goals)
single build plugin

Maven Coordinates

optional plugin configuration

'Missing' build plugin

```
<plugin>
  <groupId>org.codehaus.mojo</groupId>
  <artifactId>tomcat-maven-plugin</artifactId>
  <configuration>
    <url>http://127.0.0.1:8080/manager/text</url>
    <server>tomcat</server>
    <path>/mvn-workshop</path>
  </configuration>
</plugin>
```



Add a build plugin

Practice Session

- open pom (text editor)
- add *plugin* element to node *plugins*

Settings

- project unspecific information
 - general settings, servers
- global settings `$M2_HOME/conf/settings.xml`
- user settings `~/.m2/settings.xml`

Server

- for download/deployment

```
<servers>
```

```
  <server>
```

```
    <id>mvn
```

server identifier for POM

```
    <username>tomcat
```

username

```
    <password>tomcat
```

password (plain text/encrypted)

```
  </server>
```

Add a server

Practice Session

- open settings file (depends on needs)
- add *server* element to node *servers*

```
<server>  
  <id>mvn</id>  
  <username>...</username>  
  <password>...</password>  
</server>
```

Deploy and visit page

Practice Session

```
$ mvn tomcat:deploy
```

and visit:

localhost:8080/mvn-workshop/hello?id=123

Nope.

- tests fail
- in class *ResponseString* change `==` to `!=` in line 22
- retry
- be happy



deploy to remote repository

- settings: add deployment server
- pom: add repository [, extension]



Add server

Practice Session

```
<server>  
  <id>metalcon-repo</id>  
  <username>metalcon</username>  
  <password>{...}</password>  
</server>
```

Encrypt password

- generate master password

```
$ mvn --encrypt-master-password <master-pw>
```

- create ~/.m2/settings-security.xml

```
<settingsSecurity>  
  <master>{...}</master>  
</settingsSecurity>
```

```
$ mvn --encrypt-password <pw>
```

Add repository to POM (deploy)

Practice Session

- add *repository* element to node *distributionManagement*

```
<repository>
  <id>metalcon-repo</id>
  <uniqueVersion>false</uniqueVersion>
  <name>Metalcon Repository</name>
  <url>scp://metalcon2.physik.uni-mainz.de/metalcon-mvn/</url>
  <layout>default</layout>
</repository>
```


Add extension for SCP

Practice Session

- add *extension* element to node *extensions*
(parent node: build)

```
<extension>  
  <groupId>org.apache.maven.wagon</groupId>  
  <artifactId>wagon-ssh</artifactId>  
</extension>
```



Deploy

```
$ mvn deploy
```

Add repository to POM (download)

- add *repository* element to node *repositories*

```
<repository>
  <id>metalcon-repo</id>
  <url>scp://metalcon2.physik.uni-mainz.de/metalcon-mvn/</url>
  <snapshots><enabled>true</enabled></snapshots>
  <releases><enabled>true</enabled></releases>
</repository>
```

Sources

- <https://maven.apache.org/pom.html>
(detailed POM Guide)
- <http://search.maven.org/>
(search Maven Central Repository)
- <http://mvnrepository.com/>
(like above, more detail but with advertising)
- <https://maven.apache.org/settings.html>
(settings reference)



Sources

- <https://maven.apache.org/guides/mini/guide-encryption.html>
(encrypted passwords)
-