

Getting to know Maven 2 - Handout

Managing your builds so they don't manage you

Chris Maki

chrismaki@mac.com

Maven Principles

- *Convention over configuration*
- *Reuse of build logic*
- *Declarative execution*
- *Coherent organization of dependencies*

Maven 2 Glossary

Maven Glossary

- *Artifact* – produced or used by a project, e.g. JAR, WAR, binary distribution
- *Dependency* – the JARs and other artifacts from other projects your project uses
- *Project Object Model* – pom.xml, the metadata for your project

Maven Glossary

- *SuperPOM* – all Maven projects inherit the superPOM
- *Repository* – usually called a repo, is “a structured storage of project artifacts”^[mao]
- *Plugin* – everything in Maven is a plugin

Maven Glossary

- **MOJO** – *play-on-words from POJO, a Maven object that implements a goal - aka a plugin*
- **Goal** – *the action you want a MOJO to perform*

Maven Glossary

- *Lifecycle phases – a standard set of phases your build will pass through. All Maven projects have the same build lifecycle phases*

Resources

Useful Sites

- *Maven home*

<http://maven.apache.org>

- *Sonatype*

<http://www.sonatype.com>

- *DevZuz*

<http://www.devzuz.com>

- *Appfuse reference*

<http://appfuse.org/display/APF/Maven>

Useful Maven Pages

- *pom.xml technical descriptor*

`http://maven.apache.org/ref/current/maven-model/maven.html`

- *Maven plugins*

`http://maven.apache.org/plugins/`

- *Maven documentation index*

`http://maven.apache.org/guides/index.html`

- *Maven articles*

`http://maven.apache.org/articles.html`

Bibliography

- *MDG – Maven Definitive Guide*

<http://www.sonatype.com/book/>

- *BBM – Better Builds with Maven*

<http://www.devzuz.com/web/guest/products/resources#BBWM>

- *MAO – Maven Glossary*

<http://maven.apache.org/misc/glossary.html>

Tips

maven-release-plugin issues

- *You cannot currently use properties for SNAPSHOT version dependencies (MDG-275)*
- *To use a local snapshot repo, you need to add that repo to your settings.xml file, under the profile section.*

Tips & Tricks

- *To increase Maven's memory, use the MAVEN_OPTS environment variable.*


```
export MAVEN_OPTS="-Xmx415m"
```

- *Maven's view of you pom.xml*

```
mvn help:effective-pom
```

- *Generate test results*

```
mvn surefire-report:report
```

```
mvn surefire-report:report-only 
```

Jetty

- *To debug Jetty, when using the Jetty plugin*
 - *you need to debug Maven itself*
 - *use the MAVEN_OPTS variable*

```
export MAVEN_OPTS=" ${MAVEN_OPTS} \  
-Xdebug -Xnoagent \  
-Xrunjdwp:transport=dt_socket, \  
server=y,suspend=y,address=8000, \  
suspend=n"
```

You can now debug Jetty via JPDA