



Maven

Softwareprojekte mit Kultur

Patrick Zeising

14. Juni 2011

Konfigurationsmanagement

14. Juni 2011

Motivation

- Projektaufbau unterschiedlich
- Abläufe zum Übersetzen und Deployen unterschiedlich
- Verwendete Tools, Prozesse, Skripte unterschiedlich
- Pflege von Strukturen und Prozessen aufwändig
- Hohe Hürden für den Neueinstieg
- Steile Lernkurven und lange Einarbeitungszeit

Ziele

Vereinheitlichung der Strukturen, Abläufe und eingesetzten Tools.

- Hohe Evolvierbarkeit
- Niedrige Hürden für Projekt-Einsteiger
- Flache Lernkurven
- Kürzere Einarbeitungszeit in Projekte

Effektive, günstige und qualitativ hochwertige Softwareentwicklung.

Was ist Maven?

14. Juni 2011

Maven ist ...

- ein deklaratives Projektmanagement-Framework
- eine Laufzeitumgebung für Plugins
- komplett in Java geschrieben und Open Source

Maven managed Softwareprojekte.

Besonderheiten

- Ganzheitliche Projektphilosophie:
Kompilieren – Testen – Packen – Deployment – Release
- Konvention geht über Konfiguration
- Aufgabenteilung in Plugins
- Project Object Model - POM
- Verwaltung von Abhängigkeiten

Maven-Philosophie

„... visibility, reusability, maintainability and comprehensibility.“
nach: [Better Builds with Maven]

Konvention über Konfiguration

- Namenskonventionen
Transparenz: beispiel-1.0-SNAPSHOT.jar
- Standards für Verzeichnisse

Sourcen	src/main/java
Test-Sourcen	src/test/java
Output	target
Projekt-Dokumentation	src/site
- Pro Maven-Projekt genau ein Build-Ergebnis (Artefakt)
Single Responsibility Principle (SRP)
Separation of Concerns (SoC)

Build Lifecycle

- ...
- 6. process-resources
- 7. compile
- ...
- 12. process-test-resources
- 13. test-compile
- 14. test
- ...
- 16. package
- ...
- 21. install
- 22. deploy

Wiederverwendbarkeit

- Build-Prozesse aufgeteilt in Plugins
SoC: Übersetzen, Testen, JAR erzeugen, Javadoc erzeugen, . . .
- Deklarative Konfiguration
Was statt Wie.
- Project Object Model (POM)
zentrale, vollständige XML-Beschreibung eines Projekts

Project Object Model

POM

```
<project>  
  <modelVersion>4.0.0</modelVersion>  
  <groupId>de.mavenbuch.beispiele</groupId>  
  <artifactId>beispiel</artifactId>  
  <version>1.0-SNAPSHOT</version>  
  <packaging>jar</packaging>  
</project>
```

Koordinaten

groupId : artifactId : version : packaging

de/mavenbuch/beispiele/beispiel-1.0-SNAPSHOT.jar

Projektbeziehungen

Vererbung

```
<project>
  <parent>
    <groupId>de.mavenbuch.beispiele</groupId>
    <artifactId>mavenbuch-parent</artifactId>
    <version>1.0-SNAPSHOT</version>
  </parent>
  [...]
</project>
```

Projektbeziehungen

Aggregation

```
<project>
  [...]
  <modules>
    <module>web-layer</module>
    <module>persistence-layer</module>
    <module>business-layer</module>
  </modules>
  [...]
</project>
```

Verwaltung von Abhängigkeiten

- Abhängigkeiten eindeutig zentral definiert (POM)
Was erwartet das Projekt?
- Auflösung von (transitiven) Abhängigkeiten
Wovon hängen meine Abhängigkeiten ab?
- Zentrale Verwaltung der Abhängigkeiten
lokales und entfernte Repositories

Definition einer Abhängigkeit

```
<project>
  [...]
  <dependencies>
    <dependency>
      <groupId>org.testng</groupId>
      <artifactId>testng</artifactId>
      <version>5.8</version>
      <classifier>jdk15</classifier>
      <scope>test</scope>
    </dependency>
  </dependencies>
  [...]
</project>
```


Demo



Fazit

14. Juni 2011

Maven

- organisiert Build, Assembly, Deployment, Release, Dokumentation, verteiltes Arbeiten,
- fördert Wiederverwendbarkeit von Code- & Build-Prozessen durch Konventionen,
- senkt Wartungsaufwand beim Bauen und Konfigurieren von Projekten,
- fördert Transparenz im gesamten Entwicklungsprozess,
- stellt Software-Tools zur Verfügung.

Auswirkungen im Projekt

- Einheitliche Strukturen
- Bessere Evolvierbarkeit
- Niedrige Hürden für Projekteinsteiger
- Kurze Einarbeitungszeit für Projekte
- Automatisierung von Tests und Deployments

Weiterlesen!

- **Apache Maven Project**, <http://maven.apache.org/>
- Casey, John; Massol, Vincent; Porter, Brett; Sanchez, Carlos; Zyl, Jason van: „**Better builds with Maven**“. Mergere Library Press, 2006, <http://www.maestrodev.com/better-build-maven>
- Michael Albrecht, Patrick Zeising: „**Großprojekte konfiguriert mit Maven 2**“, JavaSPEKTRUM, Ausgabe 02/2009, 20.03.2009
- Martin Spiller: „**Maven 3: Konfigurationsmanagement mit Java**“, mitp Verlag, Juli 2011, <http://www.mavenbuch.de/>
- **Sonatype Books**, <http://www.sonatype.com/Support/Books>

Vielen Dank!

Patrick Zeising

p.zeising@neusta.de

+49 421 20696 222

neusta software development

Ein Unternehmen von team neusta

Contrescarpe 1

28203 Bremen

www.neusta.de | www.team-neusta.de

14. Juni 2011