

ESA Buildmanagement

von Florian Goldbach

Es folgt eine Beschreibung für das build file von Ant und Gradle.

Ant build.xml file

Das ant build.xml file funktioniert wie folgt.

Wenn Sie es mit „ant“ ausführen, werden erstmal nur die Ordner für das Projekt angelegt – also es passiert nicht viel.

Sie müssen „ant“ gar nicht ausführen. Stattdessen können Sie „ant get“ als ersten Befehl ausführen.

Sie erhalten daraufhin ein Beispielprojekt. Alle Dateien dafür werden von meinem Github Repository „AntFeatureExploration“ geladen.

Ich habe die Befehle „ant“ und „ant get“ mit Absicht getrennt, damit man nicht durch versehentlich Eingabe eines einfachen „ant“-Befehls die gesamten Projektdateien mit den ursprünglichen aktualisiert. „ant get“ kann man versehentlich nicht so schnell eingeben.

Die Tasks CLEAN, GET, COMPILE, JAR, TEST, RUN, DOC sind benutzbar – allerdings in Kleinschreibung!

Gradle build.gradle file

Hier habe ich mich exemplarisch gegen den vorherigen Ansatz entschieden. Also „gradle compile“ lädt zuerst die Daten aus dem Repository bevor es sie kompiliert.

Es handelt sich natürlich nur um ein Beispielprojekt, es hat hier also nicht direkt einen Sinn. Allerdings: Stelle man sich vor, dass man in einem mit Github geklonten Repository arbeitet, welches man dann nach jeder Änderung „committet“, wäre es sinnvoll.

Die Tasks CLEAN, GET, COMPILE, JAR, TEST, RUN, DOC lassen sich hier wie folgt ausführen:

CLEAN	- gradle clean
GET	- gradle donwloadFiles
COMPILE	- gradle compileJava ..und um die Test files zu kompilieren: gradle compileTestJava
JAR	- gradle jar
TEST	- gradle test
RUN	- gradle run
DOC	- gradle javadoc