

Title

Detecting of Build Smells in Build Configurations

Contact/Questions

Univ.- Ass. DI Christian Macho (christian.macho@aau.at), E.2.64

Context/Introduction

Build systems are widely used in today's software projects to automate integration and build processes. Similar to source code, build configuration need to be maintained to avoid outdated configurations, and build breakage as a consequence. Recent work indicates that neglected build maintenance is one of the most frequently occurring reasons why builds break. The context of this topic is situated in finding methods and developing tools that support the developers and release engineers in improving the quality of their build configuration.

Problem Description and Target

Code Smells are parts in Source Code files that are considered to need improvements in terms of code quality. Their detection is a common technique to identify areas in Source Code that need special attention in terms of the need to improve the code. Source Code Smells have been extensively studied in recent research and there are already tools that can detect Source Code smells. However, finding smells in Build Configurations has not been studied. The goal of this topic is to identify common smells in Build Configurations, to automate the identification process, and to study the impact of such smells on the build procedure.

Tasks

- Identification of smells in Build Configurations
- Automation of the smell identification with a prototyping tool
- Evaluating the impact of build smells on the build procedure

Type

Praktikum/B.Sc. Thesis/ M.Sc. Thesis

Prerequisites

- Knowledge in Programming (preferably in Java)
- Knowledge in Software Engineering
- Basic Knowledge in Build Systems (i.e. Apache Maven)

Titel

Detecting of Build Smells in Build Configurations

Kontakt/Fragen

Univ.- Ass. DI Christian Macho (christian.macho@aau.at), E.2.64

Kontext/Einführung

Build Systeme werden in heutigen Software Engineering Projekten immer häufiger eingesetzt, um die Integration und den Build Prozess zu automatisieren. Genauso wie Source Code muss auch die Konfiguration der Build Systeme gewartet werden, um zu verhindern, dass die Konfiguration veraltet und im schlimmsten Fall der Build Prozess fehlschlägt. Kürzliche Arbeiten zeigen, dass vernachlässigte Wartung der Build Konfiguration eine der häufigsten Ursachen für fehlgeschlagene Builds sind. Der Kontext dieses Themas liegt darin, Methoden zu finden und Tools zu entwickeln die die Entwickler und Release Engineers unterstützen, die Qualität der Build Konfiguration zu steigern.

Problem Beschreibung und Ziel

Code Smells sind Stellen im Source Code die eine Überarbeitung benötigen, um die Code Qualität zu verbessern. Die Erkennung von Code Smells ist eine bekannte Technik, um Stellen im Code zu finden, die eine besondere Aufmerksamkeit benötigen. Während Source Code Smells in früheren Studien ausführlich untersucht wurden, gibt es wenig Arbeiten, die Smells in Konfigurationen für Build Systeme adressieren. Das Ziel dieses Themas ist, häufige Smells in Build Konfigurationen zu identifizieren, den Identifizierungsprozess zu automatisieren, und die Auswirkungen von Build Smells auf den Build Prozess zu untersuchen.

Aufgaben

- Smells in Build Konfigurationen identifizieren
- Automation des Identifizierungsprozesses mittels eines Prototypen
- Auswertung der Auswirkung der Smells auf den Build Prozess

Typ

Praktikum/B.Sc. Thesis/ M.Sc. Thesis

Voraussetzungen

- Programmierkenntnisse (vorzugsweise in Java)
- Software Engineering Kenntnisse
- Grundkenntnisse mit Build Systemen (Apache Maven)