

## MASTERARBEIT

### WHITE-BOX KOMPOSITION TEMPLATE-BASIERTER CODE GENERATOREN

Am Lehrstuhl für Software Engineering der RWTH Aachen ist ab sofort folgende Masterarbeit zu vergeben.

#### Ansprechpartner

Dr. Andreas Wortmann  
Tel.: 0241 80 21346  
[wortmann@se-rwth.de](mailto:wortmann@se-rwth.de)



#### Aufgabenstellung

Prof. Dr.  
Bernhard Rumpe  
Tel.: 0241 80 21301  
[rumpe@se-rwth.de](mailto:rumpe@se-rwth.de)

Modellierungssprachen werden durch ihre Syntax (Struktur und Darstellung der Modelle) und Semantik (Bedeutung) definiert. Code Generierung ist eine populäre Möglichkeit die Semantik von Modellierungssprachen in Software zu realisieren. Während die Komposition von unabhängig voneinander entwickelten Sprachsyntaxen (beispielsweise in Form von Grammatiken oder Metamodellen) gut verstanden ist, ist die Komposition von Generatoren ungelöst.

```
<#assign behaviorImplFilename = op.getValue("behaviorImplFilename")>
<#assign behaviorFactoryFilename = op.getValue("behaviorFactoryFilename")>
<#if !op.getValue("behaviorIsManual")>
  <#assign t_behavior = op.getValue("behaviorTemplate")>
  ${op.call(t_behavior, behaviorPackage + "." + behaviorImplFilename, behaviorAST)}
</#if>
  ${op.call(t_behaviorFactory, behaviorPackage + "." + behaviorFactoryFilename, ast)}

private void initialize() {
  ${ast.printName()}Result${ast.printGenericDefinition()} result
  = (${ast.printName()}Result${ast.printGenericDefinition()})
  this.delegate.getInitialValues();
}
<#if hasFields>
  <#foreach port in op.getValue("outputPorts")>
    this.${port.getName()}.setNextValue(result.get${port.getName()}?cap_first());
  </#foreach>
</#if>
```

Am Lehrstuhl für Software Engineering wurde bereits ein erfolgreicher Ansatz für die Black-Box Komposition bestimmter Arten von Code Generatoren entwickelt. Basierend auf diesen Kenntnissen soll in dieser Arbeit ein verallgemeinerbarer Ansatz für die White-Box Komposition beliebiger Template-basierter Code Generatoren entwickelt werden.

#### NOTWENDIGE VORKENNTNISSE

- Programmierung in Java
- Software Engineering

#### WÜNSCHENSWERTE VORKENNTNISSE

- Software Language Engineering mit MontiCore
- FreeMarker Template Engine