

Master  
Synchronizing Software Design Models and Program Code with Concrete and  
Symbolic Mappings

Synchronisation von Software Design Modellen und Programmcode mit  
konkreten und symbolischen Mappings

### Motivation — Deutsch

Unser Framework Codeling<sup>1</sup> synchronisiert Modelle und Code, indem es die Notwendigkeit von Modellen als separates Artefakt eliminiert. Das Tool übersetzt Modelle in Quellcode-Muster und zurück. Dies ermöglicht es, bei Bedarf zuverlässig Modelle wie Zustandsautomaten aus dem Quellcode zu extrahieren und Änderungen in den extrahierten Modellen an den Code weiterzugeben. Der generierte Code hat wohldefinierte Schnittstellen für die Interaktion mit seinem kontextuellen Code, d.h. dem Rest des Systems. Das Framework beschreibt für Modell/Code-Mappings Übersetzungen, Code-Libraries und Stubs zur Ausführung der Modelle zur Laufzeit. Die Entwicklung dieser Artefakte ist in diesem Framework zeitaufwändig und fehleranfällig. Durch die Beschreibung von expliziten Mappings kann die Erzeugung der Artefakte automatisiert und damit wesentlich verbessert werden. In dieser Masterarbeit untersuchen Sie, wie Mappings zwischen Software Design-Modellen und Code-Strukturen für das Framework Codeling spezifiziert werden kann, sodass die genannten Artefakte automatisiert erzeugt werden können.

In dieser Masterarbeit werden Sie den folgenden Forschungsfragen nachgehen:

- Wie können Model/Code-Mappings formuliert werden, sodass Übersetzungen, Libraries und Runtime-Stubs generiert werden können?
- Wie können in diesem Zusammenhang symbolische Mappings umgesetzt werden?

Für die Durchführung der Arbeit benötigte Kenntnisse: Java

Hilfreiches Wissen: Codegenerierung, z.B. mit XText, Modellbasierte Softwareentwicklung mit Ecore

### Motivation — English

Our framework Codeling<sup>1</sup> synchronizes models and code by eliminating the need for models as separate artifacts. The tool translates models into source code patterns and back. This allows models such as state machines to be reliably extracted from the source code when needed, and changes in the extracted models can be propagated to the code. The generated code has well-defined interfaces for interaction with its contextual code, i.e. the rest of the system. For model/code mappings, the framework describes translations, code libraries and stubs for executing the models at runtime. The development of these artifacts in this framework is time consuming and error-prone. By describing explicit mappings the generation of the artifacts can be automated and thus significantly improved. In this master thesis you will investigate how mappings between software design models and code structures can be specified for the framework Codeling, so that the mentioned artifacts can be generated automatically.

In this master thesis you will pursue the following research questions:

- How can model/code mappings be formulated so that translations, libraries and runtime stubs can be generated?
- How can symbolic mappings be implemented in this context?

Knowledge required to carry out the work: Java

Helpful knowledge: Code generation, e.g. with XText, model-based software development with Ecore

---

<sup>1</sup><https://codeling.de>

## Organisatorisches

Kontakt:

Dr. Marco Konersmann <konersmann@uni-koblenz.de>