Bachelor/Master Thesis Topics

Software product lines are families of related software systems that are developed by taking variability into account during the complete development process. In model-based refinement methods (e.g., ASM, Event-B, Z, VDM), systems are developed by stepwise refinement of an abstract, formal model.

In our research, we develop concepts to combine model-based refinement methods and software product lines. On the one hand, this combination aims to improve the cost-effectiveness of applying formal methods by taking advantage of the high degree of reuse provided by software product lines. On the other hand, it helps to handle the complexity of product lines by providing means to detect defects on a high level of abstraction, early in the development process.

Survey of Model-Based Refinement Methods

Several model-based refinement methods (e.g., VDM, RAISE, B, Event-B, Z, ASM) have been proposed in the literature. These methods apply the idea of stepwise refinement to formal specifications: an abstract specification is successively refined to a more concrete specification, and finally to executable code. On each refinement level, the system can be analysed and potential errors be found. The different methods differ in many aspects such as their underlying formalism, their notion of refinement, their support for different analysis techniques.

The goal of this thesis is to survey these methods and compare them regarding their expressiveness, i.e., what kind of systems can be modelled, their decomposition mechanisms used to structure systems, their concept of refinement, their possibilities for analysis and verification and regarding existing tool support. Ideally, the advantages and disadvantages of the different methods should be exemplified using a common example.

Variability-Encoding for Abstract State Machines (Bachelor/Master)

In Feature-Oriented Software Development (FOSD), individual products of a product line can be automatically generated for a given feature selection, either by composing feature modules or by extracting the relevant parts of an annotated code base. Due to the possibly massive amount of possible products, it is not feasible to analyse each product separately. Variability-Encoding is a technique in which compile-time variability is transformed to runtime variability, i.e., a meta-product is created that simulates the variable behaviour of the complete product line. This meta-product can be analysed efficiently to draw conclusions about all products.

In our research, we have developed techniques for feature-oriented development of formal specifications based on Abstract State Machines. The goal of this thesis is to develop a concept for variability encoding of Abstract State Machines, implement a prototype, and evaluate it by performing a case study.