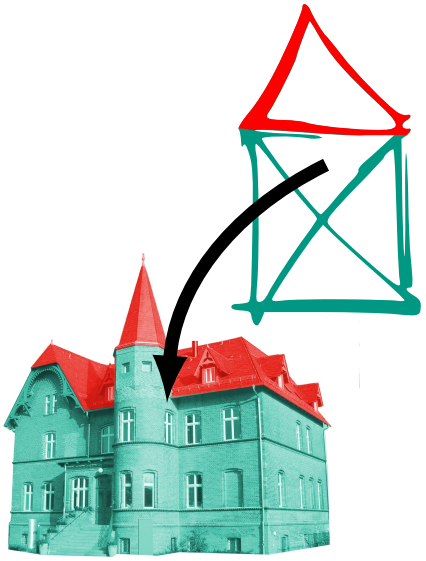
Karlsruher Institut für Technologie

Institut für Informationssicherheit und Verlässlichkeit (KASTEL)
Anwendungsorientierte Formale Verifikation
Prof. Dr. Bernhard Beckert

Praxis der Forschung

Algorithmic Refinement for KeY



Background. Formal verification of programs is challenging. One of the reasons for this is that real-world programs intertwine the abstract ideas of data structures and algorithms with features of the programming language, for example for handling memory access or exceptions.

Refinement is a technique (used for example in the B method/Event-B or the refinement framework of Isabelle) to build systems starting from an abstract model and refining it step by step into a more and more concrete one.

Vision. We envision a refinement technique for Java programs where the data structures are modelled via abstract/algebraic data types (ADTs), while the algorithms working on these structures are described with specification-only programs, so called “ghost” code. These entities are then open to treatment with efficient tools that work well on such abstract levels. Proofs on that level can then be refined to the Java level (suitable for the tool KeY developed at our chair), where only the additional language specific part as well as the coupling has to be proven.

Your Task. The goal is to develop such a technique for refinement of abstract algorithms to concrete Java programs, such that the algorithmic part of the program is correct by construction and only the language specific part, i.e., the handling of features of the concrete programming language has to be proven on the concrete Java program.

Your Profile. You should have a background in formal systems (e.g. from the respective lectures of the KIT curricula). Ideally, you have attended “Formale Systeme 2: Anwendung”, where an introduction to refinement with Event-B is given.

Contact
Wolfram Pfeifer wolfram.pfeifer@kit.edu Office: 50.34, R228