Karlsruhe Institute of Technology

Institute of Information Security and Dependability (KASTEL)
Application-oriented Formal Verification
Prof. Dr. Bernhard Beckert

Praxis der Forschung, Master Thesis

Lenses for Models, Categorically

Models in Model-driven development (MDD) are used to manage the complexity of system development: each model describes a different purpose-oriented aspect of the system. However, such models are never fully separate: dependencies and overlap abound between them.

Usually models are given in languages like UML or SysML, by discarding their concrete syntax they can be formalised as merely (attributed, typed) graphs. Such graphs have received extensive categorical treatment in the literature, and in particular ideas like *graph grammars* or the *unstructured calculus* (UnCAL) imbue them with compositional structure.

Lenses are an idea from the field of software engineering and functional programming: a lens consists of a $get : M \to V$ and a $put : V \times M \to M$ operation which satisfy the "lens laws":

$$put(get(m), m) = m \qquad get(put(v, m)) = v$$

Thus a lens can be thought of as an (editable!) *view* onto a larger structure which allows focussing on a (not necessarily syntactic) subcomponent, which makes them relevant to considering dependencies between models.

However, lenses are commonly constructed only over tree-like data structures (XML, HTML, algebraic data types, ...), which allow for easy (recursive) definitions of a lens's operations.

Several topics could be studied as a PdF/thesis:

1. **Lens construction:** Lenses on trees are often defined using *combinators*, via the tree's structure. Using the structure of a graph grammar, can similar combinators be defined for models (formalised as attributed typed graphs)?
2. **Lens evolution:** Models, by their nature, are subject to change during the larger system's development process. Investigate the consequences of such changes for lenses between them.
3. **Vitruvius as an HLR System:** Vitruvius is a tool for co-developing consistent models. Provide a categorical formulation to study termination and confluence of propagated changes.

Your Profile. You should have an interest in category theory and abstract reasoning, and be curious about their applications in (model-driven) engineering.

Contact
Terru Stübinger stuebinger@kit.edu Office: 50.34, R247