

Master Internship

Formal and tool-supported operator for multi-formalism modelling

Keywords:

Software Engineering, Domain Specific Languages, Metamodelling, and Model Composition.

Description:

Software-intensive systems become more and more complex, causing software engineering techniques to evolve constantly. The software engineering community tries to tackle this issue by encouraging people to use higher levels of abstraction and to separate concerns to ease the process of describing and building software. In this context, Model-Driven Engineering proposes frameworks and concrete good practices to assist people in designing and building software. Separation of concerns is carried out through the use of Domain-Specific Languages. Each concern is described with help of a single DSL and this DSL provides sets of dedicated concepts defined at the good level of abstraction.

Of course, separation of concerns is useful only if we are able to easily merge, synchronize, make those concerns interoperable, or to translate abstract models into concrete implementations (using usually complex chains of compilation). Each situation involves the definition of complex relationships between the software artefacts with regard to specific purposes. These relationships are usually described at the level of the languages in form of models operators. Nevertheless, these issues are currently considered as problem-specific operations and consequently lead to specific solutions which implementation process is tedious and time-consuming (e.g. [1, 2, 3, 4] in the Triskell team).

In order to help the designer in the implementation of complex operators on heterogeneous models, the candidate will propose a framework that can be easily specialized for a particular intent. For this purpose, the candidate will look upon the state of the art about model composition techniques. Then, he/she will propose a formal unified definition with an explicit identification of the variability in the existing approaches. Finally, the candidate will implement the associated tools helping the designer in the definition of complex relationship between languages.

Bibliography:

- [1] Mickael Clavreul, Olivier Barais, and Jean-Marc Jézéquel. Integrating legacy systems with mde. In ICSE'10: Proceedings of the 32nd ACM/IEEE International Conference on Software Engineering and ICSE Workshops, volume 2, pages 69--78, Cape Town, South Africa, May 2010.
- [2] François Fouquet, Olivier Barais, and Jean-Marc Jézéquel. Building a kermeta compiler using scala: an experience report. In Workshop Scala Days 2010.
- [3] Franck Fleurey, Benoit Baudry, Robert France, and Sudipto Ghosh. A generic approach for automatic model composition. In Aspect Oriented Modeling (AOM) Workshop, Nashville, USA, October 2007
- [4] Brice Morin, Franck Fleurey, Nelly Bencomo, Jean-Marc Jézéquel, Arnor Solberg, Vegard Dehlen, and Gordon Blair. An aspect-oriented and model-driven approach for managing dynamic variability. In Proceedings of ACM/IEEE 11th International Conference on Model Driven Engineering Languages and Systems (MoDELS 08), Toulouse, France, October 2008.

Working Environment:

- Laboratory:
 - [IRISA](#), [EPI Triskell](#) (Campus de Beaulieu, Université de Rennes 1, France)
- Scientific Advisors:
 - [Benoît Baudry](#) and [Benoît Combemale](#) (IRISA, Triskell)