

A Formal Approach to Patterns in MDE

Yngve Lamo

21 September 2010

Diagram Predicate Framework (DPF) [1] is a generic graph-based specification framework that tends to adapt first-order logic and categorical logic to software engineering needs. DPF is used as a formal diagrammatic approach to (meta)modelling, model transformation and version control in model-driven engineering (MDE) [3, 4, 5, 2, 6].

In this presentation, we will briefly introduce DPF and discuss how it can be used as a formal approach to the specification of patterns in MDE. In particular, we will discuss pattern enforcement and pattern discovery in view of DPF. We will also give examples to illustrate patterns in structural models, model-to-model transformation and model refactoring.

References

- [1] Bergen University College and University of Bergen. *Diagram Predicate Framework (DPF)*. <http://dpf.hib.no/>.
- [2] Alessandro Rossini, Adrian Rutle, Yngve Lamo, and Uwe Wolter. A Formalisation of the Copy-Modify-Merge Approach to Version Control in MDE. *Journal of Logic and Algebraic Programming*, 2010.
- [3] Adrian Rutle. *Diagram Predicate Framework: A Formal Approach to MDE*. PhD thesis, Department of Informatics, University of Bergen, Norway, To appear.
- [4] Adrian Rutle, Alessandro Rossini, Yngve Lamo, and Uwe Wolter. A Diagrammatic Formalisation of MOF-Based Modelling Languages. In Manuel Oriol and Bertrand Meyer, editors, *TOOLS Europe 2009: 47th International Conference on Objects, Components, Models and Patterns*, volume 33 of *LNBIP*, pages 37–56. Springer, 2009.
- [5] Adrian Rutle, Alessandro Rossini, Yngve Lamo, and Uwe Wolter. A Formalisation of Constraint-Aware Model Transformations. In David Rosenblum and Gabriele Taentzer, editors, *FASE 2010: 13th International Conference on Fundamental Approaches to Software Engineering*, volume 6013 of *LNCS*, pages 13–28. Springer, 2010.
- [6] Adrian Rutle, Alessandro Rossini, Yngve Lamo, and Uwe Wolter. A Formal Approach to the Specification and Transformation of Constraints in MDE. *Journal of Logic and Algebraic Programming*, Submitted.