

Clone Detection for Rule-Based Model Transformation Languages

Daniel Strüber,¹ Vlad Acretoaică,² Jennifer Plöger³

Keywords: model transformation; clone detection; quality assurance; Henshin; ATL

1 Abstract

We present our paper published in the Journal of Software and Systems Modeling [SAP17].

Model transformation is an enabling technology of Model-Driven Engineering, a software engineering paradigm in which models are routinely refined, translated, and optimized. To support the developers of a model transformation with advanced mechanisms for analysis, verification, and traceability, a large variety of dedicated model transformation languages has evolved [CH06]. Since many of these languages were originally developed in an academic setting, there is a lack of mature mechanisms for the reuse of existing transformations. As a result, the most frequently applied reuse mechanism remains cloning, i.e., copying and modifying existing transformations. Unfortunately, cloning presents severe maintainability challenges, which are well-known from the substantial body of research on *software clones* [Ko07]. Despite the availability of numerous clone detection techniques for programming and modeling languages, no specific ones have emerged for model transformation languages.

We present the first investigation of clone detection for model transformation languages. Our paper focuses on two main paradigms of model transformations that are related by a common denominator, the notion of rules: *graph-based* languages, in which transformations are expressed using graph rewriting rules, and *hybrid* languages, which combine declarative rule notions with imperative concepts. Specifically, we make the following contributions:

- A discussion of use-cases for clone detection for model transformation languages. Aiming to shed light on the indeed extensive scope of possible use-cases, we consider the refactoring of existing transformations towards suitable reuse mechanisms, quality

¹ Universität Koblenz-Landau, Universitätsstr. 1, 56070 Koblenz, Germany, strueber@uni-koblenz.de

² Frequentis Romania, Tetarom 1 Technology Park, Taietura Turcului 47, 400221 Cluj-Napoca, Romania, radu-vlad.acretoaică@frequentis.com

³ Philipps-Universität Marburg, Hans-Meerwein-Str. 1, 35032 Marburg, Germany, jennifer.ploeger@gmx.de

assessment, performance optimizations, and the identification of transformation design patterns. Based on the use-cases, we elicit five key requirements for clone detection for graph-based and hybrid model transformations.

- A customization strategy of existing model clone detection techniques, allowing us to address the identified requirements. We customized eScan [Ng09], an a-priori-based technique, and ConQAT [De08], a heuristic one, to the graph-based language Henshin [Ar10, St17] and the hybrid language ATL [JK05].
- An extensive experimental evaluation. For the graph-based case, we considered three rule sets; the largest one included 1,404 rules. For the hybrid case, we considered the ATL zoon, including 2,566 rules in total. A key finding is that ConQAT's accuracy was nearly optimal when using the results of eScan as a ground truth.

In the established taxonomy of software clones [Ko07], our investigation focuses on *Type I* and *Type II* clones, i.e. identical fragments and almost-identical ones (except for naming). Type I and II clones are typically created when model transformations are developed in a copy-and-paste manner. With our findings, we provide a first insight into the application of clone detection to model transformation languages. In the future, we aim to extend our work to Type III and IV clones and additional model transformation paradigms.

References

- [Ar10] Arendt, Thorsten; Biermann, Enrico; Jurack, Stefan; Krause, Christian; Taentzer, Gabriele: Henshin: advanced concepts and tools for in-place EMF model transformations. In: MODELS, pp. 121–135. Springer, 2010.
- [CH06] Czarnecki, Krzysztof; Helsen, Simon: Feature-based survey of model transformation approaches. IBM Systems Journal, 45(3):621–645, 2006.
- [De08] Deissenboeck, Florian; Hummel, Benjamin; Jürgens, Elmar; Schätz, Bernhard; Wagner, Stefan; Girard, Jean-Francois; Teuchert, Stefan: Clone Detection in Automotive Model-Based Development. In: ICSE. ACM, pp. 603–612, 2008.
- [JK05] Jouault, Frédéric; Kurtev, Ivan: Transforming Models with ATL. In: MODELS Workshops. Springer, pp. 128–138, 2005.
- [Ko07] Koschke, Rainer: Survey of research on software clones. In: Dagstuhl Seminar 06301: Duplication, Redundancy, and Similarity in Software, p. 24. GI, 2007.
- [Ng09] Nguyen, Hoan; Nguyen, Tung; Pham, Nam; Al-Kofahi, Jafar; Nguyen, Tien: Accurate and Efficient Structural Characteristic Feature Extraction for Clone Detection. In: FASE. Springer, pp. 440–455, 2009.
- [SAP17] Strüber, Daniel; Acrețoiaie, Vlad; Plöger, Jennifer: Model clone detection for rule-based model transformation languages. Software & Systems Modeling, Oct 2017.
- [St17] Strüber, Daniel; Born, Kristopher; Gill, Kanwal Daud; Groner, Raffaella; Kehrer, Timo; Ohrndorf, Manuel; Tichy, Matthias: Henshin: A Usability-Focused Framework for EMF Model Transformation Development. In: ICGT. Springer, pp. 196–208, 2017.