An Exploratory Study on Performance Engineering in Model Transformations

Raffaela Groner[1], Luis Beaucamp[1], Matthias Tichy[1], Steffen Becker[2]

Abstract: Model–Driven Software Engineering is used to deal with the increasing complexity of software, but this trend also leads to larger and more complex models and model transformations. While improving the performance of transformation engines has been a focus, there does not exist any empirical study on how transformation developers deal with performance issues. We used a quantitative questionnaire to investigate whether the performance of transformations is actually important for transformation developers. Based on the answers to the questionnaire, we conducted qualitative semi-structured interviews. The results of the online survey show that 43 of 81 participants have already tried to improve the performance of a transformation and 34 participants are sometimes or only rarely satisfied with the execution performance. Based on the answers from our 13 interviews, we identified different strategies to prevent or find performance issues in model transformations as well as different types of causes of performance issues and solutions to resolve them. We compiled a collection of tool features perceived helpful by the interviewees for finding causes. Overall, our results show that performance of transformations is relevant and that there is a lack of support for transformation developers without detailed knowledge of the engine to solve performance issues. This summary refers to our work, which was accepted for the Foundation Track of the ACM / IEEE 23rd International Conference on Model Driven Engineering Languages and Systems (MODELS) in 2020 [Gr20].

Keywords: Mixed Method Study; Model Transformation; ATL; Henshin; QVTo; Viatra; Performance Engineering

Due to the increasing size and complexity of software systems to be developed, the size and complexity of models used in Model–Driven Software Engineering (MDSE) is also increasing. Since these models serve as input for model transformations, this trend can lead to an increasing execution time of transformations until a performance issue occurs. Currently, there is a lot of research that deals with the performance of the transformation engine, for example by improving the engine [Fr17] or developing approaches that speed up the transformation execution by parallel execution [BWV16]. However, there is still a lack of research focusing on whether transformation developers have to deal with performance issues and, if they do, how they try to solve them. Therefore, we conducted an explorative mixed method study, consisting of a quantitative online survey and qualitative interviews.

The results show that 43 out of 81 survey respondents have already tried to analyze or improve the performance of their transformations. The issues mentioned are long execution

[1] Ulm University, Institute for Software Engineering and Programming Languages, James–Franck–Ring, D–89069 Ulm, Germany raffaela.groner@uni-ulm.de, luis.beaucamp@uni-ulm.de, matthias.tichy@uni-ulm.de
[2] University of Stuttgart, Institute for Software Engineering, Universitätsstraße, D–70569 Stuttgart, Germany steffen.becker@iste.uni-stuttgart.de

doi:10.18420/SE2021_14

times and high memory consumption. The transformation developers have developed techniques to prevent performance issues or to find causes. For example, they carry out performance tests to prevent issues. The strategies used to find causes vary depending on the knowledge of the engine and the available information. Since most of the languages do not provide any support for such an analysis at the transformation level, pure users can only try to find possible causes through trial–and–error. Developers with more knowledge of the engine, use tools, e.g. a Java profiler, at the engine level to analyze the performance. The developers have discovered different causes for performance issues that occur either at the engine level, the transformation level, or in the input model. Causes on the engine level include, e.g., properties of the engine, for instance in some engines, lists are copied multiple times to avoid side effects. In case of transformations, e.g., expensive OCL expressions can be the cause of performance issues. On the model level, certain structures, such as deep inheritance hierarchies, can be the cause. With regard to the solutions used, there are only a few that require adjustment of the engine. Instead, the developers try to improve the execution time, e.g., by defining their transformations more unambiguous through stricter preconditions or by using language concepts like caching. It also becomes clear that the developers want more information about a transformation execution at the transformation level. For example, they want not only information like execution time or memory consumption, but also tracing information about how a transformation is executed. They also want more tool support, e.g., profilers at the transformation level or analyses to detect expensive operations.

Some transformation developers have already developed techniques for analyzing and improving the performance, but these work mainly at the engine level. Therefore, these techniques are not applicable for pure users, although many solutions could be applied by them, since they do not require any engine customization. In order to address this problem, we want to develop an approach in our future work that will also help pure users to identify causes for performance issues and to fix them.

This work was funded by the Deutsche Forschungsgemeinschaft (DFG) - Ti 803/4-1.

Bibliography

[BWV16] Burgueño, Loli; Wimmer, Manuel; Vallecillo, Antonio: A Linda-based platform for the parallel execution of out-place model transformations. Information and Software Technology, 79:17 – 35, 2016.

[Fr17] Fritsche, Lars; Leblebici, Erhan; Anjorin, Anthony; Schürr, Andy: A Look-Ahead Strategy for Rule-Based Model Transformations. In: MODELS (Satellite Events). volume 2019 of CEUR Workshop Proceedings. CEUR-WS.org, pp. 45–53, 2017.

[Gr20] Groner, Raffaela; Beaucamp, Luis; Tichy, Matthias; Becker, Steffen: An Exploratory Study on Performance Engineering in Model Transformations. In: Proceedings of the 23rd ACM/IEEE International Conference on Model Driven Engineering Languages and Systems. MODELS '20, Association for Computing Machinery, New York, NY, USA, p. 308–319, 2020.