JOT
JOURNAL OF
OBJECT TECHNOLOGY

Journal of Object Technology | **RESEARCH ARTICLE**

A Survey on the Relevance of the Performance of Model Transformations

Raffaela Groner*, Katharina Juhnke*, Stefan Götz*, Matthias Tichy*, Steffen Becker†, Vijayshree Vijayshree†, and Sebastian Frank†
*Ulm University, Germany
†University of Stuttgart, Germany

ABSTRACT

In Model-Driven Engineering (MDE) model transformation languages are used to describe important operations on models. Such domain-specific languages are specially developed to describe transformation rules, according to which an output model should be generated from an input model. In comparison to these domain-specific languages, techniques to analyze and improve the performance of programs written in a general-purpose language, such as Java or C, are well known. However, are such techniques also needed for model transformation languages? *Problem.* Since these languages are only used in certain domains; the first question is whether performance is at all relevant for model transformations and whether techniques similar to those used to analyze and improve the performance of general-purpose languages are needed. Research in the performance of model transformations focuses mainly on comparing the performance of different languages or different definition styles or optimizing the engine that executes the transformation. However, it is not clear to what extent these efforts can mitigate or prevent performance issues, and there is also a lack of studies that examine to what extend the performance of transformations is relevant. *Method.* In order to close this gap and to answer the initial question about the relevance of performance, we conducted an online survey. For this purpose, we developed a questionnaire and identified 649 authors as potential participants based on a Systematic Literature Review (SLR) on a selection of model transformation languages. Additionally, we were able to acquire four further potential participants by advertising our study. In total, 84 participants took part in our survey. We used statistical tests such as Kendall's τ_c, the Kruskal-Wallis-Test and the Mann-Whitney-U-Test to evaluate our hypotheses on relevant factors for the performance of model transformations. *Results.* The results show that specific performance is desired and that there is a willingness to improve performance. In this regard, we identified a need for insights necessary to better understand how a transformation is performed in order to be able to improve its performance. Furthermore, we investigated with the help of hypotheses tests the possible influencing factors that cause participants to try to analyze or improve the performance of model transformations. The main results of the hypotheses tests are that the satisfaction with the execution time, the size of the models used, the relevance of whether a specific execution time is not exceeded in the average case, and the knowledge of how a transformation engine executes a transformation are relevant factors.

KEYWORDS Model transformation; performance; survey; ATL; Henshin; QVTo; Viatra.

JOT reference format:
Raffaela Groner, Katharina Juhnke, Stefan Götz, Matthias Tichy, Steffen Becker, Vijayshree Vijayshree, and
Sebastian Frank. *A Survey on the Relevance of the Performance of Model Transformations*. Journal of Object Technology. Vol. 20, No. 2, 2021.
 http://dx.doi.org/10.5381/jot.2021.20.2.a5

1. Introduction

Model-Driven Engineering (MDE) is a valuable technology in different areas, particularly, in the area of cyber-physical systems (Liebel et al. 2018). Model transformations are a key technology in MDE as they provide means, e.g., to synchronize multiple models, to translate models in one formalism into models in another formalism as part of model transformations chains, and continuously update models@run.time in response to environment events in self-adaptive systems. Typically, model transformations are specified in transformation scripts executed by engines. The scope of this paper are model-to-model transformations excluding model-to-text transformations. However, for the sake of better readability, we generally use the term model transformations for model-to-model transformations in the following.

Consequently, a plethora of model transformation languages like the Atlas Transformation Language (ATL) (Jouault et al. 2008), Henshin (Strüber et al. 2017), QVTo ((OMG) 2016) or Viatra (D. Varró et al. 2016) have been developed in the past. Furthermore, there exist approaches to analyze model transformation scripts with respect to functional requirements by formal verification (cf. Amrani et al. 2015).

Unfortunately, empirical research on usage, advantages, and disadvantages of model transformation languages and corresponding tools is relatively scarce (Götz et al. 2020) as works similar to the empirical studies by Hutchinson et al. (2011) and Liebel et al. (2018) on model-based engineering in general are still missing. Such empirical studies could influence the research in the area of model transformations and may, consequently, positively affect the adoption of model transformations in industrial practice (Burgueño et al. 2019).

Götz et al. (2020) have identified 15 different categories of advantages and disadvantages of model transformations in their recent Systematic Literature Review (SLR). One of these 15 categories is the execution performance of model transformation. However, only five claims specifically about performance were found by Götz et al. This raises the question whether performance is relevant for developers of model transformations. Consequently, the aim of this paper is to shed light on this question by providing *quantitative* data and, in case performance is indeed important, reporting which information the developers of model transformations need to improve the performance of a model transformation's execution. We also consider the usage contexts of model transformations, since they may influence the relevance of performance.

This paper complements our recent paper (Groner et al. 2020a) that *qualitatively* investigates how developers deal with the performance of model transformation executions. In contrast to our work, most existing research on the performance of model transformations focuses on improving the engine executing the model transformations, e.g., by improving the application sequence of model transformation rules inside the engine (Fritsche et al. 2017; Fleck et al. 2015) or using parallel and/or concurrent execution engines (Sanchez Cuadrado et al. 2020; Benelallam et al. 2016). Another line of existing research investigates refactorings to improve the execution performance of model transformations (Mészáros et al. 2010; Bruni & Lluch Lafuente 2012).

Specifically, our research questions are as follows:

RQ1 What is the usage context of model transformation languages?

RQ2 How relevant is the performance of model transformations executions for developers?

RQ3 What information needs exist in regards to the performance of model transformations?

RQ4 What are differences between developers who have tried to analyze or improve the performance of model transformations and those who have not?

We conducted a systematic online survey to answer our research questions. Potential survey participants were selected based on a light-weight Systematic Literature Review (SLR) process. The SLR process focused on the model transformation languages ATL, Henshin, QVTo, and Viatra, and two publishers (IEEE Xplore and ACM Digital Library) resulting in 415 selected papers and 649 authors. Those 649 authors as well as four additional experts were invited to participate in our online survey, of which 84 participants completely answered the survey. In addition to presenting the survey answers descriptively, we also tested seven hypotheses about sub groups of answers using appropriate statistical tests.

With respect to **RQ1**, the survey results show that model transformations are used for many different tasks such as model analysis, model migration, model manipulation, respectively model refinement each have been reported by more than half of the survey respondents. Furthermore, respondents report models with more than 100,000 model elements.

With respect to **RQ2**, the average execution time of model transformation is important for 57% of the survey respondents and 40.5% of all respondents are rarely or only sometimes satisfied with the execution time.

With respect to **RQ3**, we asked the survey participants to rate whether a set of proposed execution data – which are not provided by existing model transformation approaches –, like the number of investigated objects and execution order or called rules, satisfies the information needs to improve the performance of model transformations. All proposed execution data were rated by more than half of the survey respondents as moderately important or higher.

With respect to **RQ4**, we conducted statistical procedures to check correlation between answers using Kendall's τ_c and to check significance between subgroups of the participants using the Kruskal-Wallis-Test and the Mann-Whitney-U-Test. Major results are that (unsurprisingly) participants who have tried to analyze or improve the performance tend to 1) have more knowledge about the engine, 2) are less satisfied with the execution time, 3) use large models, and 4) consider it more important that a certain execution time is not exceeded in the average case than participants who have not. An interesting result, in our data set, is that the size of models is *not* correlated with satisfaction of the execution time.

In summary, performance is an important and relevant quality-characteristic of the execution of model transformations. Specifically, our statistical analysis shows that those who are more knowledgeable about the engine also tend to analyze and try to improve the performance compared to non-experts whereas we did not find a significant difference in satisfaction with the execution performance between those groups. Furthermore, developers report that they require more additional information than currently available in model transformation approaches to be able to improve the performance of model transformation executions. Both results lead us to the conclusion that more tool support for non-experts, i.e., users of model transformation engines, should be developed and, subsequently, empirically evaluated.

In the next section, we present our methodology. Section 3 contains a detailed presentation and statistics on our survey results. After a comparison with related work in Section 4, we conclude and give an outlook on future work in Section 5.

2. Methodology

In this section we describe our study design. Figure 1 gives an overview of our methodological approach, in particular the steps we have taken to develop our study design.

Our approach is related to the theory-testing survey research process presented by Forza (2002). Along this process, we have also structured the following sections. We first stated our hypotheses and operationalized them, which we describe in Section 2.1. Then, as described in Section 2.2, we designed our questionnaire to measure the variables for our hypotheses and to gather further information to answer our research questions. We describe the followed search for potential participants and the execution of the study in Sections 2.3 and 2.4. Last, we analyzed our data using descriptive statistics and hypotheses testing, which we explain in more detail in Section 2.5. In Section 2.6, we discuss some of the major threats to validity of our study.

2.1. Hypotheses and Operationalization

In this section, we present the hypotheses that we have tested using the data collected through the questionnaire. Furthermore, we also sketch which data we collected for each hypothesis (with more detail in Section 2.5). We defined our hypotheses in order to answer **RQ4**. Based on our own experience with model transformations, we considered what could cause the performance of a transformation to be so poor that developers try to improve it and how much knowledge about the engine is needed to do so. Based on these considerations, we defined the following seven hypotheses:

$H0_1$ There is no correlation between the size of input models (variable *modelElement_WA*: number of objects in the model) used and the satisfaction (*satisfaction*: on a 5-point scale from *never* to *always*) with the execution time (Groner et al. 2021).

$H0_2$ There is no difference in satisfaction (*satisfaction*) with the execution time between the group of participants who have expert knowledge about the engine (*developer*), the group of participants who have limited knowledge about the engine (*researcher*) or the group of participants for whom the engine is a black box (*user*) (*role_rating*: distinguishing *developer*, *researcher*, and *user*) (Groner et al. 2021).

$H0_3$ There is no difference in possible expert knowledge about the engine (*role_rating*) between the group of participants who have already tried to analyze or improve performance and the group of participants who have never tried (*analyze*: distinguishing *yes* and *no*) (Groner et al. 2021).

$H0_4$ There is no difference in the satisfaction (*satisfaction*) with the execution time of a transformation between the group of participants who have already tried to analyze or improve the performance and the group of participants who have never tried (*analyze*) (Groner et al. 2021).

$H0_5$ There is no difference in the sizes of the input models used (*modelElement_WA*) between the group of participants who have already tried to analyze or improve the performance and the group of participants who have never tried (*analyze*) (Groner et al. 2021).

$H0_6$ There is no difference in the importance of not exceeding a certain execution time in the average case (*averageCase*: on a 7-point scale from *not at all important* to *extremely important*) between the group of participants who have already tried to analyze or improve performance and the group of participants who have never tried (*analyze*) (Groner et al. 2021).

$H0_7$ There is no difference in the importance of not exceeding a certain execution time in the worst case (*worstCase*: on a 7-point scale from *not at all important* to *extremely important*) between the group of participants who have already tried to analyze or improve performance and the group of participants who have never tried (*analyze*) (Groner et al. 2021).

The aim of **$H0_1$** is to find out whether respondents are more dissatisfied with the performance of their transformations when using larger models than respondents using smaller models. The aim of **$H0_2$** is to investigate whether respondents with more knowledge about the engine are more dissatisfied with the performance of their transformations than respondents who have limited or no knowledge about the engine. The aim of **$H0_3$** to **$H0_7$** is to investigate the differences between the respondents who have already tried to analyze or improve performance and those who have never done so.

2.2. Instrumentation

In this section we present the design of our questionnaire. The way we designed our questionnaire is inspired by the process presented by Saris & Gallhofer (2014). Therefore, we first identified high-level concepts (concepts–by–postulation) in which we are interested in. We have decomposed these concepts into easy to measure concepts (concepts–by–intuition) for which we

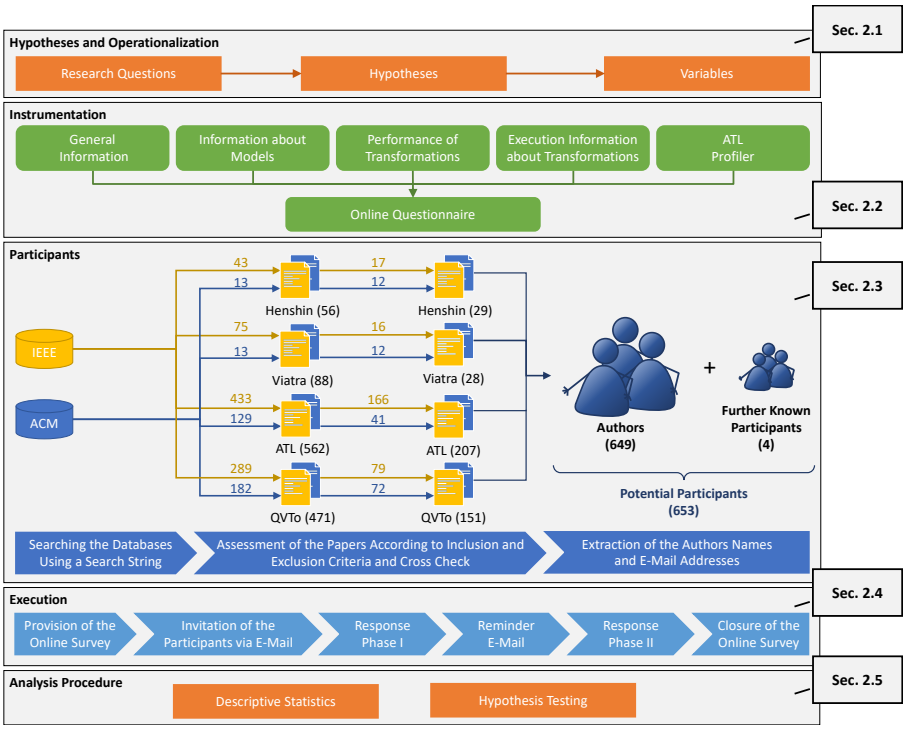

Figure 1 Overview of the methodological approach for the study design

formulated questions. We did not perform all the process steps of Saris & Gallhofer (2014), because while we broke down the concepts–by–postulation into concepts–by–intuition, we already came up with some suitable questions. During this step we also decided on how to measure the variables for our hypotheses. We used the guidelines from Malhotra (2006) to finalize the formulation of our questions and we used the suggestions from Vagias (2006) to select appropriate Likert scales.

Based on our research questions we identified the following five different concepts–by–postulation:

(A) *General Information*
In order to answer **RQ1** we are interested in general information, e.g., the number of years the participants have used a transformation language or what languages they use.

(B) *Information about Models*
Since models serve as input and output of transformations, they have a strong impact on performance. Therefore, their size and what information about a model is relevant are also important information to answer **RQ1** and **RQ3**.

(C) *Performance of Transformations*
In order to answer **RQ2** we are interested in different aspects of performance, e.g., whether the participants are satisfied with the execution time of their transformations or whether it is important for the participants that the transformation does not exceed a certain execution time.

(D) *Execution Information about Transformations*
In order to answer **RQ3** we are interested in what information the participants want to know about a transformation execution.

(E) *ATL Profiler*
In order to answer **RQ3** we are interested in whether the already available profiler in Eclipse for ATL transformations (Piers 2010) provides helpful information or not for the participants. So far, this is the only profiler known to us that presents information about the execution at transformation level, such as execution time or memory per operation. Therefore, it is of interest whether the provided information is already sufficient or not to fix performance issues.

In the following we go into detail about the individual concepts–by–intuition, into which we have divided the concepts–by–postulation and present a shortened version of the resulting questionnaire. For some questions, we have shortened the answer options by omitting explanations in order to be able to present our questionnaire as compactly as possible. In Tables 1 to 5, we marked the affected questions with an "*" after their ID. The complete questionnaire is available at (Groner et al. 2021, 2020b).

In order to operationalize *(A) General Information* we broke it down into general characteristics of a transformation developer and characteristics that can be influenced by our way of searching for potential participants. Due to our way to search for potential participants (cf. Sec. 2.3), we assume that the majority of them are active in research and use mainly ATL, Henshin, QVTo or Viatra. Therefore, we asked about the domain in which transformations are used (*Q1*) and different aspects about the transformation languages used, e.g., which language (*Q3*) has been used for how long (*Q4*) and with what intensity (*Q5–Q6*). To reduce the effort for the participants, we offered a

ID	Question and answer options
Q1	In which domain do you use transformations? Teaching / Research / Industry / Other: [Free text field]
*Q2**	In what role do you work with transformations? User / Developer / Researcher / Other: [Free text field]
Q3	Which of these transformation languages do you use? ATL / Viatra / Henshin / QVTo / AGG / eMoflon / ETL / Fujaba / GReAT / GrGen / JTL / Kermeta / QVTr / RubyTL / Tefkat / Other: [Free text field]
Q4	How many years have you been using these transformation languages? We used the same languages as answer options as in *Q3* and for each option a value x, with $x \geq 0$, can be assigned.
Q5	In a typical month, how many hours do you roughly spend to define transformations in any of these languages? We used the same languages as answer options as in *Q3* and for each option a value x, with ≥ 0, can be assigned.
Q6	In which languages are your transformations defined? Specify the distribution in percent. For each answer option ATL / Viatra / Henshin / QVTo / Other a value x, with $x \geq 0 \wedge x \leq 100$, can be assigned for the distribution in percent.
*Q7**	I use transformations for... Manipulation / Restrictive Query / Abstraction / Refinement / Analysis / Simulation / Model Generation / Migration / Optimization / Refactoring / Composition / Synchronization / Other: [Free text field]

* Explanations of the answer options are omitted. The full answer options are available at (Groner et al. 2021)

Table 1 Questions resulting from the operationalization of *(A) General Information*.

pre–selection of different languages we know and a free text field for further answers.

In order to measure the variable *role_rating* used in $\mathbf{H0_2}$ and $\mathbf{H0_3}$ we asked the participants in what role they use transformations (*Q2*). Another information we are interested in is the intention of a transformation execution (*Q7*), because depending on the usage the performance can be more or less relevant. We used a selection of the transformation intents presented by Amrani et al. (2012), to offer answer options which could be extended by the participants by a free text field. Table 1 lists the questions resulting from the operationalization of *(A) General Information*.

ID	Question and answer options
Q8	How big are the models you use? More precisely, how many objects (model elements excluding references and attributes) do your models contain? We offered the following six intervals #objects$\leq$10 / 10<#objects$\leq$100 / 100<#objects$\leq$1.000 / 1.000<#objects$\leq$10.000 / 10.000<#objects$\leq$100.000 / 100.000<#objects For each interval a value x, with $x \geq 0 \wedge x \leq 100$, can be assigned.
Q9	If you want to understand and improve the runtime of your transformations, which information do you think is important about the input or output models? For each answer option, the importance can be assessed using a 7–point Likert scale based on Vagias (2006): Number of objects per class in the meta-model / Average number and variance of attributes that objects of a specific class in the meta-model have / Average number and variance of references that objects of a specific class in the meta-model have
Q10	What further information about the input or output models is also important to understand and improve the runtime of your transformations? [Free text field]

Table 2 Questions resulting from the operationalization of *(B) Information about Models*

In order to operationalize *(B) Information about Models* we broke it down into characteristics that describe models. Since models are used as input and output of model transformations, their size is an important influencing factor for the performance of a transformation. To measure the associated variable *modelElement_WA* used in $\mathbf{H0_1}$ and $\mathbf{H0_5}$, we asked the participants about the number of model elements their input models consists of (*Q8*). We also discussed to gather other information about the models used to measure their size, but we came to the conclusion that the number of model elements is the easiest to estimate by the participants compared to number of references or branching. In order to identify the participants' information needs regarding models used, we asked them to assess the importance of metrics offered (*Q9*) and to add further information they consider interesting (*Q10*). To find suitable metrics we conducted one step forward snowballing and one step backward snowballing using Van Amstel et al. (2011) as starting point (details about the snowballing conducted is available at Groner et al. (2021)). We used Van Amstel et al. (2011) as starting point because it is the only paper known to us that focuses exclusively on metrics related to performance. However, we found mainly general metrics or metrics that only apply to one language, so after a discussion we decided to use the metrics listed in Table 2 for *Q9* in the questionnaire. Table 2 lists the questions resulting from the operationalization of *(B) Information about Models*.

ID	Question and answer options
Q11	Is it important for the execution that your transformations do not exceed a certain execution time in the average case and/or in the worst case? For each answer option, the importance can be assessed using a 7–point Likert scale based on Vagias (2006): The execution must not exceed a certain execution time in the average case / The execution must not exceed a certain execution time in the worst case
Q12	How satisfied are you with the execution time of your transformations? The question "How often are you satisfied with the execution time of your transformations?" should be answered, using a 5–point Likert scale to measure the frequency based on Vagias (2006).
*Q13**	Why are you unsatisfied with the execution time of your transformations? takes too long / fluctuates / Other: [Free text field]
Q14	Did you ever try to analyze or improve the execution time of your transformations? Yes / No

* Explanations of the answer options are omitted. The full answer options are available at (Groner et al. 2021)

Table 3 Questions resulting from the operationalization of *(C) Performance of Transformations*

In order to operationalize *(C) Performance of Transformations* we broke it down into different aspects that we can use to measure our variables and that indicate that the performance is relevant or not to the developers. One aspect, which indicates that performance is relevant for transformations, is the importance of not exceeding a certain execution time in the average case or in the worst case (*Q11*). Thus, we measure the associated variable *averageCase* used in $\mathbf{H0_6}$ and the variable *worstCase* used in $\mathbf{H0_7}$ on the basis of their importance to the participants. The satisfaction with the execution time is also an important performance aspect. To measure the associated variable *satisfaction* used in $\mathbf{H0_1}$, $\mathbf{H0_2}$ and $\mathbf{H0_4}$, we asked the participants about their satisfaction with the execution time (*Q12*). We decided to ask about the frequency of satisfaction, because this depends strongly on the executed transformation and we assume that only a minority of transformation developers are always or never satisfied with the execution time. In addition, we asked the participants who are not always satisfied with the execution time why they were dissatisfied (*Q13*). We assume that the performance is relevant, if a participant has already tried to analyze or improve the performance (*Q14*). We also measure the variable *analyze* used in $\mathbf{H0_3}$–$\mathbf{H0_7}$ with the help of *Q14*. Table 3 lists the questions resulting from the operationalization of *(C) Performance of Transformations*.

ID	Question and answer options
*Q15**	If you want to understand and improve the execution time of your transformations/transformation scripts, which information is important about the execution of a transformation? For each answer option, the importance can be assessed using a 7–point Likert scale based on Vagias (2006): Number of applied rules / Call hierarchy / Execution order of called rules / Execution time of each rule / Number of investigated objects / Number of potentially investigated objects in the worst case / Number of backtracking by the engine during the traversal of the input model / Number of created and deleted objects per applied rule / Number of created and deleted objects per class in the meta-model
Q16	What further information do you consider important about the execution of a transformation script/single transformation to understand and improve its runtime? [Free text field]

* Explanations of the answer options are omitted. The full answer options are available at (Groner et al. 2021)

Table 4 Questions resulting from the operationalization of *(D) Execution Information about Transformations*.

ID	Question and answer options
Q17	How many hours do you roughly use the ATL profiler in a typical month? A value x, with $x \geq 0$, can be assigned for the number of hours.
Q18	What is your opinion about the ATL profiler? For each answer option, the degree of agreement can be assessed using a 7–point Likert scale based on Vagias (2006): The ATL profiler helped me / The information about the total number of instructions executed helped me / The information about the total time helped me / The information about the used memory helped me / The information about the operation names helped me / The information about the number of calls helped me / The information about the execution time per operation helped / The information about the number of execution instructions helped me / The information about the memory per operation helped me / The information about how much percent of the execution time an operation needs helped me

Table 5 Questions resulting from the operationalization of *(E) ATL Profiler*

In order to operationalize *(D) Execution Information about Transformations* we also used the results from our forward and backward snowballing (details available at Groner et al. (2021)). We discussed which of the metrics are suitable to be used in our questionnaire. The metrics should be easy to understand and be applicable for several transformation languages. We asked the participants to assess the importance of the selected metrics (*Q15*) and to add further information they consider important (*Q16*). Table 4 lists the questions resulting from the operationalization of *(D) Execution Information about Transformations*.

In order to operationalize *(E) ATL Profiler* we examined the profiler offered by the ATL Eclipse plug-in (Piers 2010). We have formulated statements for each piece of information provided by the profiler to determine whether the profiler meets the participants' information needs (*Q18*). To examine the statements about the profiler in the context of the participants' experience, we also asked the participants how many hours per month they use the profiler (*Q17*). Table 5 lists the questions resulting from the operationalization of *(E) ATL Profiler*.

2.3. Participants

To conduct our survey, we needed participants with sufficient knowledge of model transformations, for example through their usage. In order to identify such potential participants, we conducted a Systematic Literature Research (SLR) (Kitchenham et al. 2009) with the aim of extracting authors and their e-mail addresses from included publications. The detailed documentation of how we performed our SLR, the inclusion and exclusion criteria used, the publications found, and their assessment are available at Groner et al. 2021.

The procedure of our SLR is based on a review protocol template from Booth et al. (2016). We searched for publications in the IEEE Xplore Digital Library and the ACM Digital Library, as they offer conference proceedings from several conferences on models and model transformations, such as MODELS, MODELSWARD, MiSE, or MoDeVVA (Groner et al. 2021, 2020a,b).

In order to find potential participants with sufficient knowledge about model transformations, we looked for publications dealing with the model transformation languages ATL, Henshin, QVTo, or Viatra. We chose these languages for the following reasons: 1) The Eclipse plug–in of ATL (Piers 2010) provides a profiler at the transformation level and offers appropriate support, 2) Henshin is a representative for a purely declarative language, 3) QVTo is widely used and is standardized by Object Management Group (OMG) (2016) and 4) the Viatra engine already provides many different concepts that improve the performance of a transformation (D. Varró et al. 2016) (Groner et al. 2020a).

Since we do not want to identify a specific use case regarding these transformation languages with our SLR, but simply publications that work with these modeling languages, our search strings turned out to be relatively simple as shown in Table 6. In order to find publications and therefore authors who work with Henshin and Viatra, our search strings consist only of the words "Henshin" and "Viatra", as they are unique names. This was not the case with ATL, because ATL is used as an abbreviation for several other terms, such as for alternating time logic. For example, the search string "ATL" leads to over 6,000 results in IEEE Xplore Digital Library. Therefore, we had to refine this search string for the search in the IEEE Digital Xplore Digital Library. Regarding QVTo, a more complex search string is required due to different existing notations. In addition, Table 6 shows the search results based on the search strings used for the respective digital libraries, in total 1,177 publications.

We applied inclusion and exclusion criteria to this set of found publications as reported in detail in our supplementary material (Groner et al. 2021). We only have included publications that contain any of the following: use or plan to use the named transformation languages to transform models, report on their development, analyze transformation scripts, or use tools based on these languages. We excluded publications that were not available in English or as full text, or of which we were the only authors. Since we used the SLR in order to find suitable participants for our survey and not to answer any research question, our criteria rather represent different contexts in relation to the work with transformations. Thus, meeting one inclusion criteria is already enough for a publication to be included. In addition, we excluded publications on ATL published before 2014 from the search, and we excluded publications on QVTo published before 2007. We considered a period of 5 years to be appropriate to identify authors who are still actively working with ATL or at least have sufficient knowledge about it. For QVTo, publications since 2007 were considered, as the Borland (Kurtev 2008) QVTo engine became available in 2007, which is later used as the QVTo engine in Eclipse. Our search for participants was split into two SLRs, one focused on publications about ATL, Henshin, and Viatra until 8th July 2019 and the other focused on publications about QVTo until 19th August 2019 (Groner et al. 2021, 2020a,b).

Overall, we included 415 papers[1]. Figure 1 and Table 6 show how many included papers we identified from which digital library and for which transformation language. Based on these papers we were able to identify 649 authors with a valid e-mail address as potential participants. If we found several e-mail addresses for an author, we used the one from the latest publication. Additionally, we invited four more participants who we acquired by talking to transformation developers and advertising our study and about whom we knew that they had problems with the performance of model transformations. In the end, we were able to identify a total of 653 potential participants.

2.4. Execution

We invited the collected 653 potential participants to our questionnaire in an online survey via e-mail. In order to increase the response rate, we sent a reminder e-mail to all participants who had not yet completed the questionnaire after an initial response phase (Groner et al. 2021, 2020a). In total, 84 participants have

[1] Due to a potential different interpretation of the inclusion and exclusion criteria during the crosscheck of some publications, we had to repeat this step for some publications resulting in more included publications and potential participants than reported in (Groner et al. 2020a,b). As a result, three additional participants completed our questionnaire.

	Henshin		Viatra		ATL		QVTo	
Libraries	IEEE	ACM	IEEE	ACM	IEEE	ACM	IEEE	ACM
Search strings	(Henshin)		(Viatra)		((model transformation) AND ATL)	(ATL)	((((((((((((((((QVTO) OR QVT-O) OR QVTo) OR QVT-o) OR qvtO) OR qvt-O) OR qvto) OR qvt-o) OR operational QVT) OR QVToperational) OR operational qvt) OR qvt operational) OR QVT-operational) OR operational-QVT) OR qvt-operational) OR operational-qvt)	(("QVTO","QVT-O", "QVTo", "QVT-o", "qvtO", "qvt-O", "qvto", "qvt-o", "operational QVT", "QVT operational", "operational qvt", "qvt operational", "QVT-operational", "operational-QVT", "qvt-operational", "operational-qvt"))
Search results	43	13	75	13	433	129	289	182
Included papers	17	12	16	12	166	41	79	72

Table 6 Search strings used to identify included publications and search results (Groner et al. 2021).

completed our questionnaire, resulting in a response rate of 12.9%.

Our response rate is low compared to the reported ones from other online surveys (Nulty 2008). But regarding our survey, it is not clear whether all invited potential participants (the authors from the collected papers) really belong to the target group of our questionnaire. For example, since we have invited all authors of an included publication, it may well be that some potential participants have never used a transformation due to being responsible for other aspects of the paper. It is also possible that the performance is only relevant for some of the transformation developers. Additionally, Singer et al. (2008) reported that they found a response rate of 5% in software engineering surveys, which means we are way above that with our 12.9%.

During the execution of the study, the answers were not anonymous, because we conducted follow-up interviews with a selection of suitable and willing participants based on the questionnaire, whose results have been published in Groner et al. (2020a). We then anonymized the data before performing our analysis, the results of which are presented in Section 3.

2.5. Analysis Procedure

For the analysis of the data collected through the questionnaire described in Section 2.2, we used both descriptive statistics, such as bar charts, ridgeline plots, box plots, and Likert plots as well as inferential statistics. For the latter we use statistical tests to test the hypotheses presented in Section 2.1.

In the following, we describe the variables used to test $\mathbf{H0_1}$ to $\mathbf{H0_7}$ and the statistical tests used. The same descriptions can also be found in our supplementary material at (Groner et al. 2021).

modelElement_WA:

Values: A value x with $x \in \{1,2,3,4,5,6\}$.

Obtained: The value is calculated based on the answers to *Q8*. We assign each interval (*interval*) of *Q8* an ascending number (*rank*) from 1 to 6 and then calculate $modelElement\_WA = \frac{\sum_{\forall interval} rank_{interval} \times w_{interval}}{100}$. The weight w is the percentage distribution that is answered by a participant for a given interval of *Q8*. In order to obtain valid results with respect to the ordinal scale, *modelElement_WA* is rounded depending on the first decimal place.

satisfaction:

Values: A value x with $x \in \{0,1,2,3,4\}$ based on a 5-point Likert scale with 0="Never" to 4="Always".

Obtained: Answer to the question *Q12*.

role_rating:

Values: A value x with $x \in \{1,2,3\}$.

Obtained: The value is calculated based on the answers to *Q2*. We assign the value 1 to the role *"user"*, because for most users the transformation engine is a black box and therefore users probably have the least expert knowledge about the engine. The value 2 is assigned to the *"researcher"* because they need to understand at least parts of the engine in order to develop analyses and the value 3 is assigned to the *"developer"* because they have a deep understanding

of the engine. This rating is based on our experience we gained during our mixed method study (Groner et al. 2020a). Based on this ranking, values are assigned to the answers for *Q2* and their maximum determines the value for *role_rating*. To calculate *role_rating* we ignore possible further answers in the free text field of *Q2*.

analyze:

Values: A value x with $x \in \{0,1\}$ with 0="No" or 1="Yes"

Obtained: Answer to the question *Q14*.

averageCase:

Values: A value x with $x \in \{0,1,2,3,4,5,6\}$ based on a 7-point Likert scale with 0="Not at all important" to 6="Extremely important".

Obtained: Answer to the option "The execution must not exceed a certain execution time in the average case" of question *Q11*.

worstCase:

Values: A value x with $x \in \{0,1,2,3,4,5,6\}$ based on a 7-point Likert scale with 0="Not at all important" to 6="Extremely important".

Obtained: Answer to the option "The execution must not exceed a certain execution time in the worst case" of question *Q11*.

In the following, we describe the statistical tests used and the reasons for their selection.

Hypothesis $\mathbf{H_1}$ represents a correlation hypothesis. For testing this hypothesis it is necessary to determine a correlation coefficient. Due to the scale levels of the variables *modelElement_WA* (ordinal scaled) and *satisfaction* (ordinal scaled) a non-parametric correlation must be used, such as Kendall's τ (Kendall 1938) or Spearman's ρ (Spearman 1910). The correlation coefficient Kendall's τ is more robust against outliers and more suitable if many scores have the same rank than Spearman's ρ. Furthermore, Kendall's statistics provide a better estimate of the correlation in the population (cf. Howell 2009). Hence, we conducted a Kendall correlation analysis. To be precise, we use Kendall's τ_c, also known as Stuart-Kendall's τ_c (Stuart 1953), as this is more suitable for analyzing data that are based on non-square contingency tables than τ_a or τ_b. With respect to the variables *modelElement_WA* and *satisfaction*, a 6×4 cross table is formed. While $\tau = 0$ means that there is no correlation between the variables examined, $\tau = 1$ indicates that the variables correlate perfectly and $\tau = -1$ expresses a perfect inversion.

For testing hypothesis $\mathbf{H_2}$, we consider three independent samples that we form using the variable *role_rating*. Using the variable *role_rating*, a distinction is made between participants with expert knowledge (group 1) and with limited knowledge about the engine (group 2), as well as participants for whom the engine represents a black box (group 3). It is tested whether these groups show a significant difference with respect to the variable *satisfaction*. Since the hypothesis testing is an analysis of variance with more than two groups with respect to an ordinal scaled variable, we use the Kruskal-Wallis-Test (Kruskal & Wallis 1952) to test hypothesis $\mathbf{H_2}$.

The statistical analysis of hypotheses $\mathbf{H_3}$ to $\mathbf{H_7}$ is based on a comparison of two independent samples with respect to one characteristic. These two independent samples are constructed based on the variable *analysis*: the group of participants who have already tried to analyze or improve the performance, and the group of participants who have never tried to analyze or improve the performance. Since the characteristics to be tested are non-metric variables, a non-parametric test must be used as the equivalent of the independent t-Test – the Mann-Whitney-U-Test (Mann & Whitney 1947). In detail the following ordinal scaled variables are tested: *role_rating* ($\mathbf{H_3}$), and *satisfaction* ($\mathbf{H_4}$) *modelElement_WA* ($\mathbf{H_5}$) *averageCase* ($\mathbf{H_6}$), and *worstCase* ($\mathbf{H_7}$).

If the test statistic is significant, we calculate the effect size as measure of the magnitude of the observed effect. For this purpose, we calculate the effect size based on the Z value as follows: $r = \frac{z}{\sqrt{N}}$ (Fritz et al. 2012). For interpreting the effect size, we refer to the definition according to Cohen (1992): $0.10 \leq r < 0.30$ (small effect), $0.30 \leq r < 0.50$ (medium effect), $r \geq 0.50$ (large effect). According to the calculation formula for the effect size, r can also be negative for a negative Z-Score, whereby a value of $r = -1$ indicates a perfect negative relationship and $r = 0$ indicates no linear relationship. Accordingly, negative effect sizes are to be interpreted as follows: $-0.10 \geq r > -0.30$ (small effect), $-0.30 \geq r > -0.50$ (medium effect), $r \leq -0.50$ (large effect) (Field 2009).

For all statistical tests we use the significance level $\alpha = 0.05$ (Field 2009).

2.6. Threats to Validity

In this section, we discuss the major threats to validity of our study.

Construction Validity: To identify survey participants by doing a literature survey, we followed best-practices. The inclusion and exclusion of each paper, and consequently its authors, was decided based on pre-defined explicit included and excluded criteria by two researchers independently. Divergence in assessment was solved for each paper by discussion between the two researchers and reaching consensus. Because we had to repeat parts of the SLR due to a potential different interpretation of the inclusion and exclusion criteria, we obtained additional authors. For these authors, we reran the study, resulting in three additional, fully answered questionnaires. We cannot say whether more of the subsequently invited authors would have participated in our study if we had invited them at an earlier time. Regarding the three additional fully answered questionnaires, we see no reason why the responses should be affected by the timing. Especially since these participants reported to only use QVTo and Acceleo and for both languages there is still no support to analyze the performance, which could have influenced the answers. In order to improve the construction validity of the survey, we conducted a pilot study for the questionnaire.

The pilot study was tested with two persons not involved in the design in order to identify unclear and/or biased questions. The questionnaire was improved based on the results of the pilot study. The survey answers were not collected anonymously (to contact survey participants for follow-up interviews whose results have been published in Groner et al. 2020a,b), but the answers were pseudonymized and were completely anonymized for analysis. The planned use of data was presented to the survey participants and they agreed upon them prior to the questions. This avoids evaluation apprehension. For our hypotheses $\mathbf{H_2}$ and $\mathbf{H_3}$, we defined the knowledge about the engine based on the role in which the participants work with transformations. This assumption is also supported by our study in Groner et al. (2020a), but it cannot be excluded that, e.g., a user has the same expert knowledge about the transformation engine as an engine developer.

Internal Validity: To avoid researcher bias, one of the authors was responsible for the statistical data analysis who was not part of the data collection activities. Furthermore, this author is also not affiliated with the funding project. This avoids that results of the data analysis may be biased by the goals and work packages of the funding project. We avoided 'fishing for results' by explicitly defining the hypotheses and the corresponding statistical procedures before starting the data analysis.

External Validity: Our method to identify participants focused on the IEEE Digital Library and ACM Digital Library. While we thus potentially have excluded survey participants that purely publish outside of those two publishers, we believe that the majority of authors which can provide insights to answer our research questions will have published at least one publication with those two publishers, e.g., due to publishing at MODELS or at MODELS workshops. Due to our method to identify participants, our questionnaire was mainly answered by transformation developers who use ATL, Viatra, Henshin or QVTo. While this poses a threat to generalize the results outside of the survey participants, the participants also report the use of other languages. Furthermore, our method to identify participants also influences that most of the participants are researchers and only 18% work in industry. Hence, it remains unclear to what extent the results can be generalized to industrial use of model transformation languages.

Conclusion Validity: In order to ensure the validity of the conclusions drawn from the statistical tests, we checked that requirements for the use of the tests prior to performing the statistical test. Some questions in our questionnaire offer answer options based on Likert scales. Although we have chosen the options based on the suggestions of Vagias (2006), we cannot guarantee that all participants interpret them in the same way. Some questions in our questionnaire use a Likert scale as response options to evaluate the importance of provided items. This can lead to participants rating each item as important. This could be circumvented by having participants rank the given items in terms of importance, but this would only give us the importance of each item relative to the others. Therefore, we decided against ranking the given item and to use a Likert scale.

Reliability: In order to improve the reliability of the study, we published all study materials, the raw data, and the SPSS project file for the statistical analysis as data publication (cf. Groner et al. 2021). This enables replication as well as independent assessment of the validity of our results. However, other researchers might come to different results since the presented results are based on the individual answers of the survey participants and replicating the data collection of the study will most likely result in different participants and different responses. Due to the fact that ACM launched on 1st January 2020 a completely new ACM Digital Library[2] our search within ACM cannot be fully reproduced anymore. However, at Groner et al. 2021 we provide the full documentation of the publications we found while performing our SLR in ACM for publications about ATL, Henshin, and Viatra until 8th July 2019 and the for the ones about QVTo until 19th August 2019.

3. Results

In this section, we present the results of our survey. We divide our results in three different categories. First, we descriptively present in Section 3.1 **general information** regarding the transformation languages used, the purposes for which transformations are used, and the sizes of the models used. Second, we descriptively present our findings in terms of **performance of model transformations** in Section 3.2. In this section we present to what extent the participants have already dealt with the performance of model transformations and we also present the performance related information the participants wish for. Third, we present in Section 3.3 the **analysis** results of our hypothesis testing on the hypotheses formulated in Section 2.1, in which we mainly examine the differences between the participants who have already tried to analyze or improve performance and those who have not. Although we focus on model-to-model transformations, we also present results related to model-to-text transformations in the following, as these may be of interest to other researchers in the field.

In order to trace specific answers to the survey data (in ProcessedData.xlxs/csv at Groner et al. 2021), we reference a questionnaire completely filled out by a participant with [ID<ID>$^{<\text{Domain}>}$]. For example, [ID1$^{\text{T,R,I}}$] means that the answer is from the participant with the <*ID*> 1 who works with transformations in the domains teaching (T), research (R) and industry (I) (cf. *Q1*).

All raw data and processed data of the survey are available at Groner et al. (2021).

3.1. General Information

In this section we present the results of *Q1* to *Q8* that provide general information about the participants in order to characterize them and answer **RQ1**, but also to investigate to what extent the selection process of potential participants has influenced our sample.

Figure 2 shows the different domains in which the participants use model transformations (cf. *Q1*). The majority of the participants is obviously active in research, due to the way we searched for participants (cf. Section 2.3). Approximately

[2] https://libraries.acm.org/training-resources/new-dl-features, Accessed: 05.07.2021

96% (81 out of 84 participants) have stated to work in research with transformations. In comparison, the shares for teaching with 54% and industry with 18% are much smaller. Since a participant may be working with transformations not only in one domain, multiple answers are possible, and therefore the sum of the percentages exceeds 100. We also asked the participants about in what role they work with transformations (cf. *Q2*), and 71% answered to be users, 54% answered to be engine developers or transformation language developers, and 67% answered to be researchers with transformations as object of research (see Figure 3). Since a participant can also work with transformations in several roles, the sum of the percentages exceeds 100.

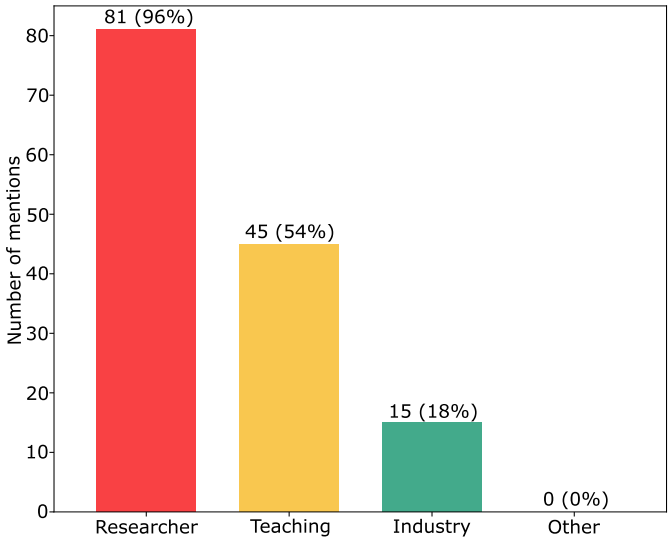

Figure 2 Distribution of domains (cf. *Q1*)

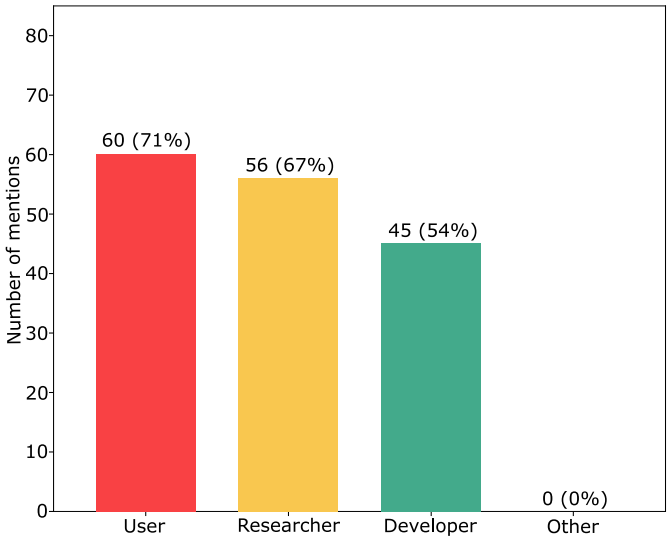

Figure 3 Distribution of roles (cf. *Q2*)

Figure 4 shows that transformations are used for completely different transformation intents (cf. *Q7*). In particular, more than half of the participants used transformations for analysis, migration, manipulation and refinement. In an additional free text, code generation was also mentioned once as a purpose to use transformations.

Figure 5 shows the distribution of model transformation languages used by the participants (cf. *Q3*). The way we looked for participants has influenced the model transformation languages used, which can be seen in the large number of ATL and QVTo users. However, a wide range of different model transformation languages are used by the participants. Additionally mentioned languages are 8 times Java (or Java+EMF, cf. Steinberg et al. 2008), 5 times Acceleo (Eclipse Foundation, Inc. 2019) and 4 times Xtend (Bettini 2016). The languages C++, C# , CoqTL (Cheng et al. 2020), EOL (Kolovos et al. 2006), Groove (Kastenberg & Rensink 2006), JetBrains MPS (Pech et al. 2013), Reactions and Mappings (Vitruvius project) (Klare 2018), Scala, TXL (Cordy 2006), UML-RSDS (Lano 2014), Velocity (Gradecki & Cole 2003), Xpand (Eclipse Foundation, Inc. 2020), XSLT (W3C 2020), Xtext (Bettini 2016), and one in-house development were only mentioned once.

Figure 6 shows the distribution of the hours per month in which a transformation language is used (cf. *Q5*) split into two plots. There are two box plots per language, where the orange one shows the distribution of hours of use, from participants who have never tried to analyze or improve the performance of their transformations, and the green one shows the distribution of hours of use, from participants who have already tried to analyze or improve the performance (cf. *Q14*). In this plot, we have omitted the languages GReAT, GrGen and Tefkat because none of the participants use them. It is noticeable that many participants who use a language for several hours per month have not analyzed the performance. It is also noticeable that the performance of a transformation is more often analyzed when another language is used. This could possibly be because, e.g., to analyze transformations in Java one can use tools, like JProfiler (cf. EJ-Technologies 2020), which are not available for many transformation languages.

We also asked the participants to specify the distribution in percent of the sizes of their used models (cf. *Q8*). This question was answered by 83 out of the 84 participants, since one participant answered all intervals with 0. The participants with the response IDs 51, 60 and 182 have distributed less than 100% over the given intervals, so we scaled their responses accordingly to present them in the following plot (Groner et al. 2021). The answers to question *Q8* are illustrated in Figure 7. On the x-axis the given intervals of model sizes are shown and on the y-axis for each participant the given distribution. The ridges describe how many percent the participant entered for each interval. For example, the first ridge line at the bottom of Figure 7 shows the answers of a participant who has about 80% models of less than 10 model elements (objects) and about 20% ones with more than 10 and less than 100 model elements. We see at the bottom left in orange/red that many participants work mainly with very small input models. However, the figure shows in the upper third that some participants use larger input models. At the top right in green, we can see that there are also some participants who mainly use models that are larger than 100,000 model elements. Hence, Figure 7 shows that the participants

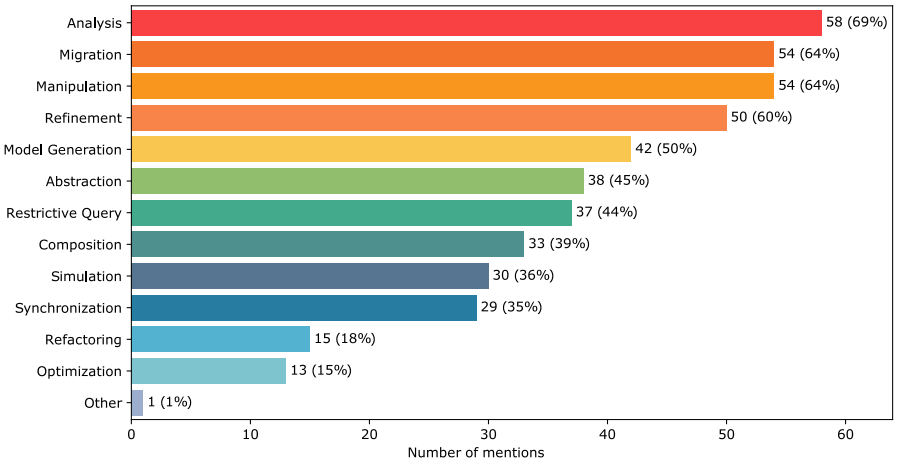

Figure 4 Tasks that are solved with model transformations (cf. *Q7*)

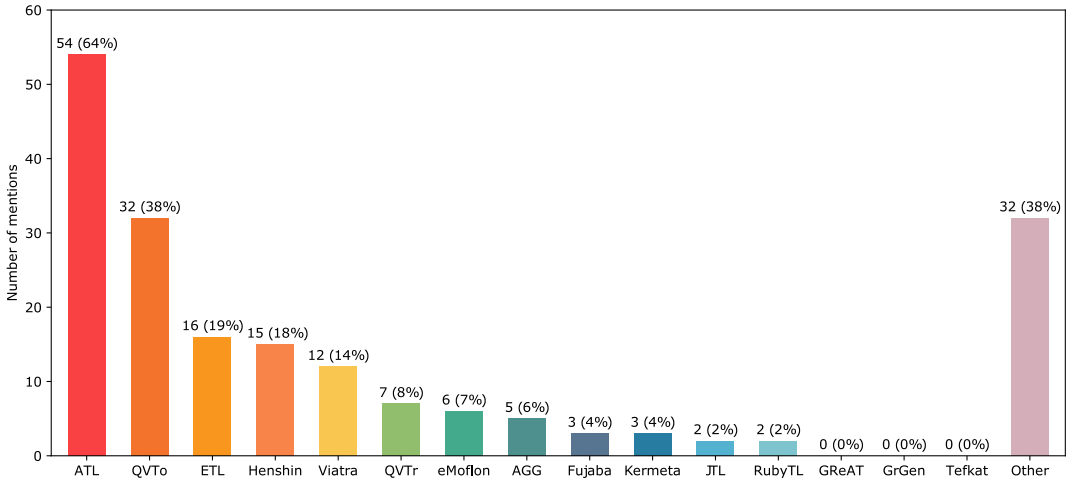

Figure 5 Model transformation languages used (cf. *Q3*)

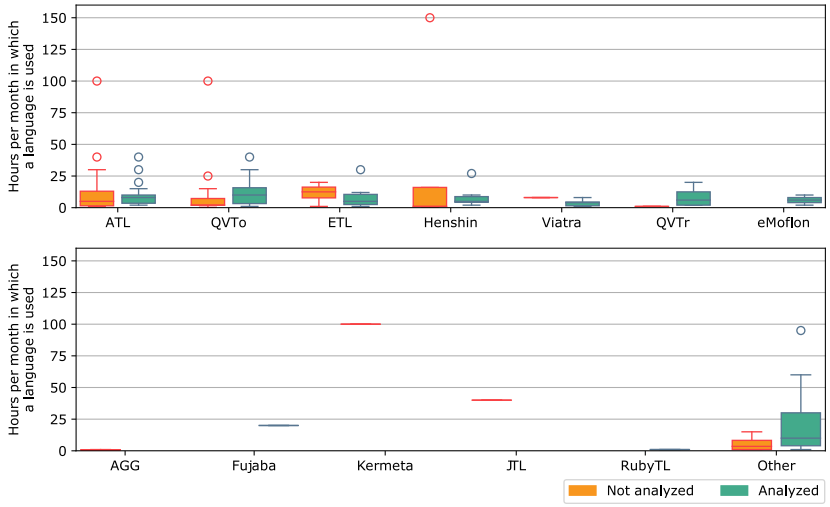

Figure 6 Distribution of the hours per month in which a transformation language is used (cf. *Q5*)

use a wide range of input models of different sizes and large models exists for which performance of model transformations is relevant (cf. Groner et al. 2020a).

Summary: The way we have searched for potential participants has an influence on characteristics of the participants. This can be seen in the fact that about 96% stated that they are involved in research with transformations (see Figure 2). The high number of participants using ATL (54 participants) and QVTo (32 participants) is also a result of our search for potential participants, but it also reflects the fact that there are far more publications about these two languages in IEEE Xplore Digital Library and the ACM Digital Library than about Henshin and Viatra. Nonetheless, the diversity of mentioned model transformation languages indicates that we cover a wide range of users of model transformation languages (see Figure 5). With regard to the models used, we see that models with up to 100 model elements are frequently used, but we also see that many participants use large models with 1,000 model elements and more (see Figure 7).

3.2. Performance of model transformations

In this section we describe our results of *Q9* to *Q18*, which are related to the performance of model transformations. In Section 3.2.1 we describe the extent to which the performance of model transformations is taken into account by the participants and in Section 3.2.2 we present what information about models and their execution is considered relevant in terms of performance and its improvement.

3.2.1. Performance in the context of model transformations

In this section we present to what extend performance is relevant for the participants and thus answer **RQ2**.

Figure 8 shows the results of the participants' assessment of how important it is for the execution of their transformations that a certain execution time is not exceeded in the average case or in the worst case (cf. *Q11*). The percentage declaration on the left of the bar chart in Figure 8 is the percentage of participants who stated that it is *"not at all important"*, *"low importance"* or *"slightly important"* that a certain execution time in the average case or the worst case is not exceeded. The percentage declaration on the right shows the percentage of answers with *"moderately important"*, *"very important"* or *"extremely important"*. Figure 8 shows that 57% (out of 84) answered that it is at least moderately important that their transformations must not exceed a certain execution time in the average case and 47% stated the same for the worst case (cf. Groner et al. 2020a).

We also asked the participants how often they are satisfied with the execution time of their transformations (cf. *Q12*) and, as Figure 9 shows, only 15.5% (out of 84) participants are *"always"* satisfied. Many of the participants are *"often"* satisfied, but 40.5% participants are only *"sometimes"* or *"rarely"* satisfied with the execution time. We asked the 71 participants who are not always satisfied about the reasons (cf. *Q13*) and 30 are dissatisfied because the execution takes too long. 25 gave as a reason that the execution time fluctuates, for example depending on the input model, and 10 gave both as reasons. Further reasons which were stated are "Transformation engines typically do a bad job w.r.t. possible optimisations" [ID51^{T,R,I}], "often due to a poor implementation of the transformation module" [ID167^{T,R}], "scalability issues on big models" [ID86^{R}] and "consumes too many resources, e.g., memory" [ID123^{R,I}] (cf. Groner et al. 2020a).

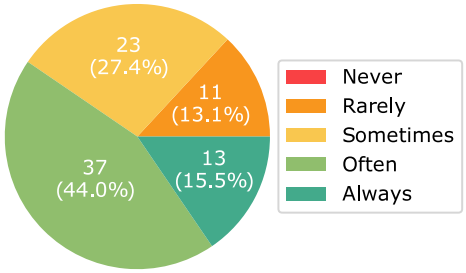

Figure 9 Frequencies of satisfaction with the execution time (cf. *Q12*) (Groner et al. 2020a)

Summary: The results show that the participants are concerned with the performance of model transformations. On the one hand, the execution time seems to be important in some applications (see Figure 8) and on the other hand, only some participants are satisfied with the execution time of their transformations (see Figure 9). All in all, it seems like a certain performance is desired, but not always achieved.

3.2.2. Performance related information

In this section we present what information participants consider important to understand and improve the performance of transformations, and thus answer **RQ3**.

When executing a transformation, the input model might have an influence on its execution time. An indicator for this relationship is, for example, that 49% of the participants who are not always satisfied (cf. *Q12*) stated that the execution time fluctuates, for example, depending on the input model (cf. *Q13*) (cf. Groner et al. 2020a).

Figure 10 shows the results of the assessment of the importance of some information about a model to understand and improve the performance of a transformation (cf. *Q9*). The additional bar chart on the right shows for each information the percentage of participants who answered *"Don't know"* or *"No answer"*. In particular the average number and variance of references and the number of objects (model elements) per class are at least moderately important for 66% of the participants. We can also see that the number of objects per class was rated by more participants as extremely important than the other two pieces of information.

We also asked the participants what further information about models they consider important (cf. *Q10*) and we could identify two categories of information: **traceability information** and **structural information**. Table 7 categorizes and summarizes the answers given. Some answers were split and/or paraphrased and abridged for presentation purpose. The complete,

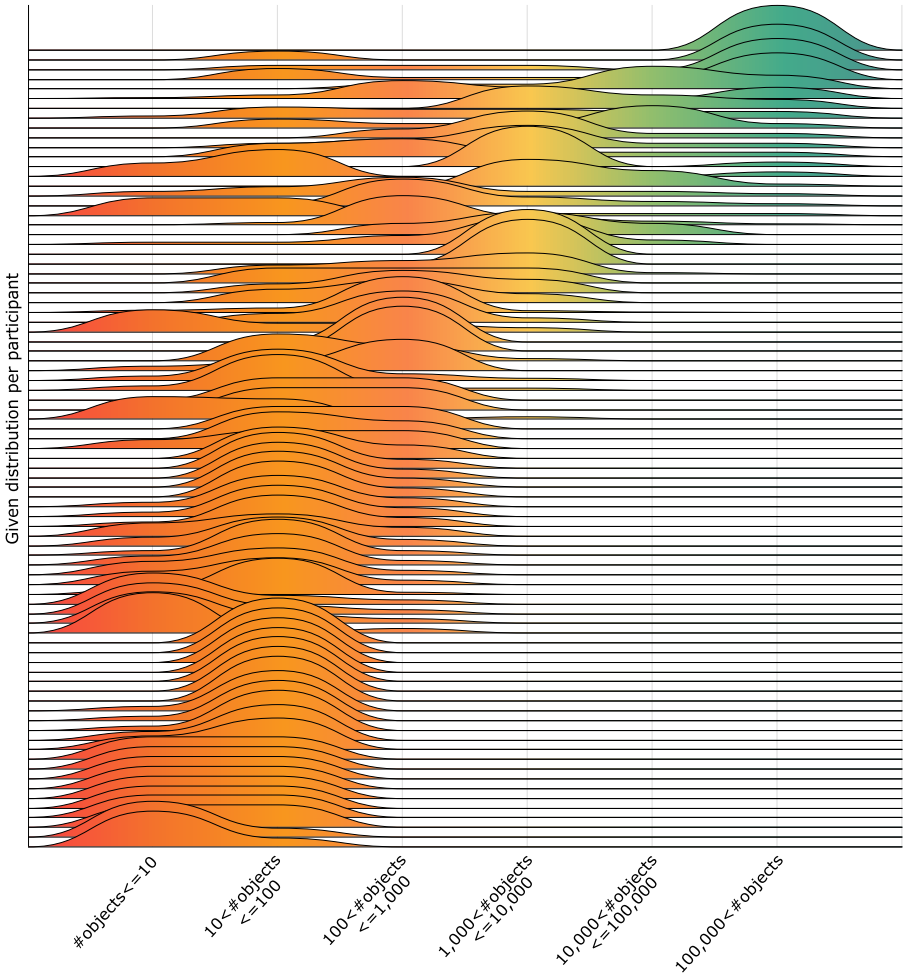

Figure 7 Distribution of model sizes per participant (cf. *Q8*)

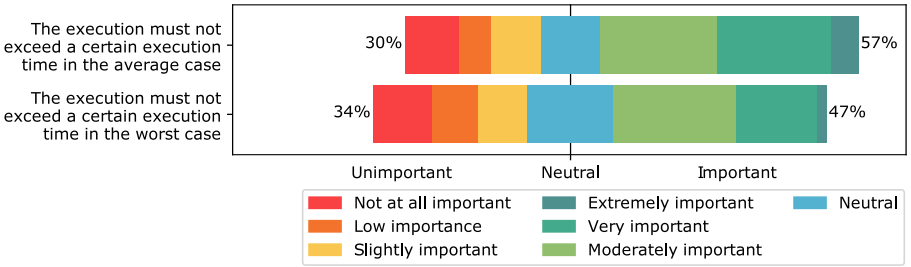

Figure 8 Assessment of the importance of the average case and the worst case (cf. *Q11*)

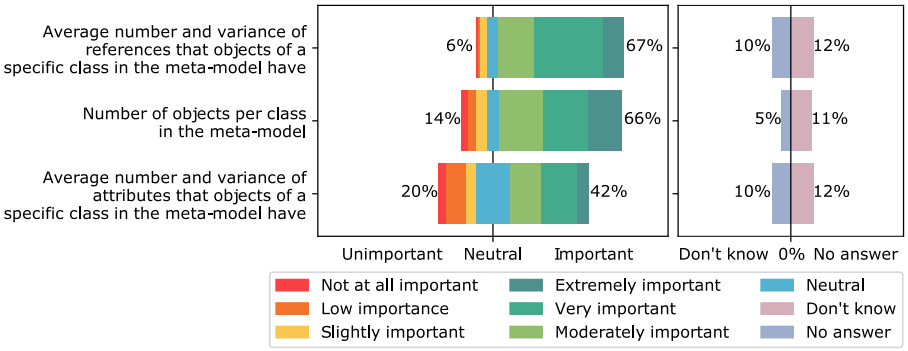

Figure 10 Results of the assessment of information about models (cf. *Q9*)

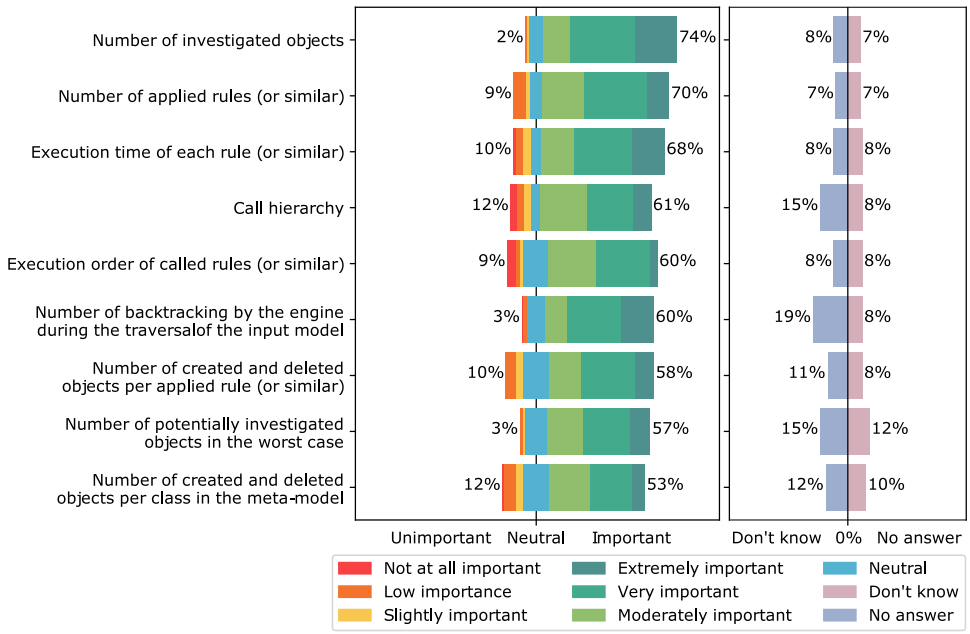

Figure 11 Assessment of the importance of characteristics of a transformation execution (cf. *Q15*)

anonymized answers can be found in the survey data (cf. Groner et al. 2021).

Table 7 shows that two different types of traceability are wanted. Firstly, to see the relationships between the input model and the output model and secondly, to understand which elements of the input model were involved in a transformation. With regard to the structure of a model, most participants are interested in its hierarchy/depth, also branching/connectivity, cycles, and constraints were mentioned twice.

Some answers are not related to models, but they are nonetheless interesting in the context of performance of model transformations. Therefore, we do not include them in the table, but list them below:

- "[...] well-formedness rules [...]" [ID100^{R,I}]
- "The key aspect is how many units of translations, and how complex is the algorithm, and what are the slow operations. I do not think this depends universally on number of objects or variation of values." [ID137^{T,R}]
- "Cyclic dependencies in networks of transformations" [ID157^{T,R}]
- "[...] it is/was crucial to understand the implementation of the transformation engine itself in order to understand and improve the runtime of our transformations [...]" [ID172^{R}]
- "Matched rules with conditions to check before model transformation" [ID11^{T,R}]

In our questionnaire the participants were also asked to assess the importance of different characteristics of a transformation execution to improve and understand performance (cf. *Q15*). Figure 11 shows the assessments and it is noticeable that each characteristic is considered at least *"moderately important"* by the majority of the participants. Especially the number of investigated objects is important to the participants, since 74% have rated them at least *"moderately important"* and moreover this characteristic has the most *"extremely important"* ratings.

We also had a free text field in our questionnaire for further information (cf. *Q16*), which the participants consider important about the execution of a transformation script/single transformation to understand and improve its execution time. Table 8 summarizes the given answers, some of them being split and/or paraphrased and abridged for presentation purposes. The complete, anonymized answers can be found in the survey data (cf. Groner et al. 2021). The answers given could only be categorized superficially. This may be due to the fact that information about a transformation execution also depends strongly on the language used, since each engine also uses its own concepts during a transformation execution.

The free text answers in Table 8 clearly show that the participants want information on different levels. On the one hand, they want to have a kind of overview of which transformations were applied in which order. On the other hand, they want more detailed information about how the engine applied a single transformation. Based on the answers, however, we see that the participants not only have an information need regarding the execution of a transformation, but also want support to improve the performance.

We asked the 54 participants using ATL (cf. *Q3*) to what extent they agree with statements that pieces of information provided by the profiler in Eclipse (cf. Piers 2010) for ATL transformations are helpful (cf. *Q18*). Figure 12 shows the results of this assessment, with over 55% of participants using ATL not answering this question. In order to examine this distribution of answers more closely, we have created a combined plot in Figure 13 that shows how many hours the participants roughly use the profiler per month (cf. *Q17*) and whether they have assessed the statements about the profiler or not (cf. *Q18*). We have divided the hours in which the participants use the

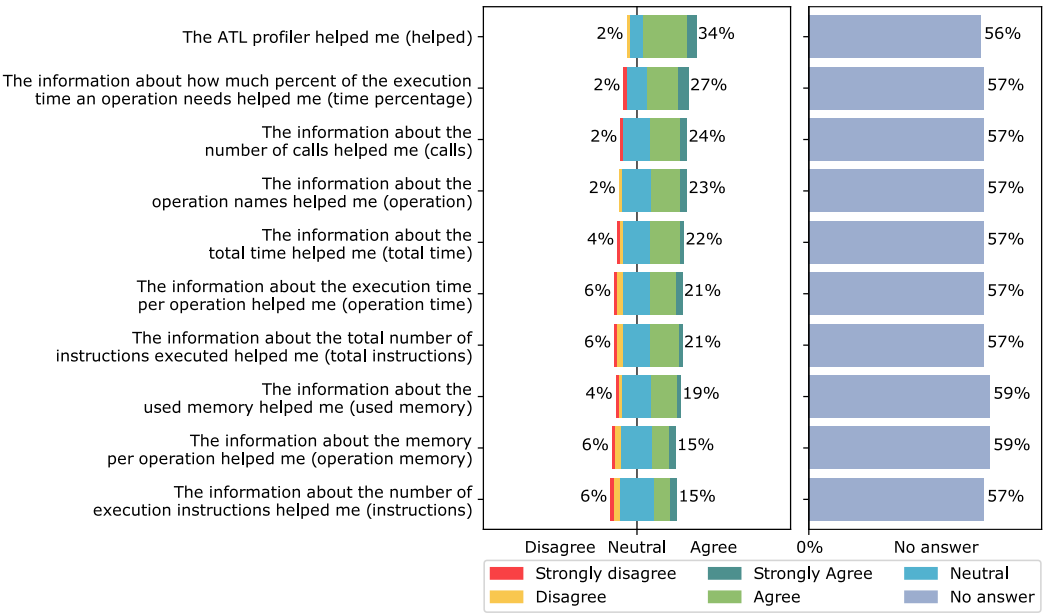

Figure 12 Assessment in how far the participants using ATL agree with statements about the helpfulness of the Eclipse ATL profiler (cf. *Q18*)

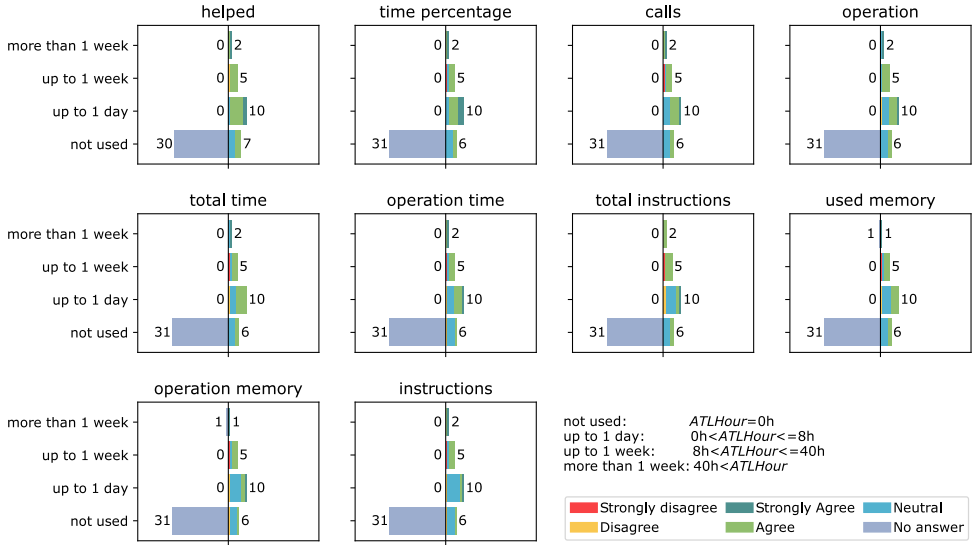

Figure 13 Distribution of the hours per month (*ATLHour*) in which the ATL Profiler is used (cf. *Q17*), broken down by statement to be assessed and whether or not it has been assessed (cf. *Q18*)

Traceability	**Between input model and output model:** • "[...] difference between the size of input and output models (for traceability)" [ID118^{T,R,I}] • "The ways to uniquely identify related parts of the elements (both in source and target model elements alone, and how easy is to figure out the traceability between them)." [ID123^{R,I}] • "Understand the mapping between the source model elements and target model elements (i.e., what target model elements are generated from what source model elements)." [ID166^{T,R}] **Between input model and transformation:** • "Which elements are touched by which rules." [ID51^{T,R,I}] • "Number of matches of a graph pattern [...]" [ID118^{T,R,I}]
Structural	**Hierarchy/Depth:** • "Hierarchical composition structures, [...]" [ID84^{T,R,I}] • "Depth of hierarchy." [ID116^{T,R}] • "Maximum depth of the model, as in the number of references from the root." [ID122^{T,R}] • "containment hierarchies" [ID162^{T,R}] **Branching/Connectivity:** • "[...] average branching factor of references [...]" [ID118^{T,R,I}] • "Connectivity between elements." [ID94^{T}] **Cycles:** • "[...], and existence of cycles of references." [ID84^{T,R,I}] • "Cyclicity of references, [...]" [ID100^{R,I}] **Constraints:** • "The ocl constraints" [ID37^{R,I}] • "The number of constraints that objects of a specific class in the meta-model have" [ID67^{T,R}] **Other:** • "In my opinion and based on the research developed, the input model needs to contain architectural details, [...] a paper precisely to approach this question: what was the most expressive and efficient model for performing harmonic mappings?, and in many contexts, in my case, I found that it was Digital TV [3] and Pervasive Computing [4]. Additionally, there was an experimental study [...]. In this study, ADL emerged as a better option to represent the architecture and to be the input model in transformations. [...] [3] M. Satyanarayanan. "Pervasive Computing: Vision and Challenges" IEEE Personal Communication pp. 10-17 Aug. 2001. [4] Ginga - http://www.ginga.org.br/pt-br/sobre [...]" [ID41^{T,R}] • "Structural patterns" [ID125^{T,R,I}] • "Number of classes in the meta-model -> typically results in higher number of transformation rules that need to be checked" [ID127^{R,I}]

Table 7 Further important information about a model (cf. *Q10*)

profiler (ATLHour) into four intervals, assuming an 8 h working day and a 40 h working week: (1) *"not used"* includes all ATL users who indicated that they use the profiler for 0 h per month. (2) *"up to 1 day"* includes all ATL users who use the profiler more than 0 h and maximum 8 h per month and (3) *"up to 1 week"* includes all ATL users who use the profiler more than 8 h and maximum 40 h per month. (4) *"more than 1 week"* includes all who use the profiler more than 40 h per month.

Figure 13 shows for each statement about the ATL profiler in Figure 12 one plot. In order to identify which plot from Figure 13 belongs to a statement in Figure 12 we use the abbreviations in parentheses after each statement. The horizontal bars in light blue, which grow to the left, show how many participants using ATL have not answered the question for the

Overview information	• "The thing I noticed is that the ark rules are not really executed in parallel and this really matters in execution time" [ID37^{R,I}] • "Number of applications per rule" [ID53^{R}] • "Iterations of a loop unit or similar concepts." [ID52^{T,R}] • "Duplicated executions of operations/rules which can be eliminated by adopting caching or similar features (available in ATL and UML-RSDS)." [ID84^{T,R,I}]
Detailed information	• "Logic of transformation rule" [ID67^{T,R}] • "How the pattern-matching is done. If there are multiple inputs which require joins, etc." [ID116^{T,R}] • "Order of objects considered by the matching engine/search plan [...]" [ID52^{T,R}] • "[...] What element/subgraph of a rule took long time or many tries to be matched. Not just the number of investigated/backtracked objects but a graphical view of the rule with elements colored according to their matching time would be nice to get an impression about hardly matchable subgraphs. [...]" [ID52^{T,R}] • "[...] A graphical debugger showing the progress of the matching process. So that a user can easily follow, e.g., the backtracking visually. Could be combined with the aforementioned bullet to show live development of how expensive it is to match a specific element/subgraph." [ID52^{T,R}] • "navigation paths [...]" [ID52^{T,R}] • "[...] complexity of the expressions in the rules [...]" [ID59^{T,R,I}] • "[...] and, most importantly, the existence of NACs in the conditions of the rules." [ID59^{T,R,I}]
Support	• "Possibility to optimize rule calling order." [ID94^{T}] • "Appropriate setup of model indexes: too few result in slower execution, too many consumes a lot a memory and initialization time increases." [ID123^{R,I}] • "recursive call modified by a loop [...]" [ID132^{R}] • "[...] optimization of the algorithm implemented by the transformation [...]" [ID132^{R}] • "[...] possibility of splitting the input model [...]" [ID132^{R}] • "Performance analysis of the actual rule interpreter using profiling tools" [ID127^{R,I}] • "Is it possible to perform incremental transformation instead of re-doing a complete transformation from scratch when the source model changes?" [ID166^{T,R}]
Other	• "In the case of rule-based languages, the execution time of OCL navigation expressions, in particular in helpers." [ID51^{T,R,I}] • "Memory usage" [ID54^{T,R,I}] • "Architectural information to specify input models." [ID41^{T,R}] • "The implementation of the transformation engine itself [...]" [ID172^{R}]

Table 8 Further important information about a transformation execution (cf. *Q16*)

corresponding statement. The horizontal bars growing to the right show how many participants using ATL have answered the question for the corresponding statement and their color indicates how they answered the question. The answers given are divided into four time intervals based on the duration of use per month, which are shown on the y-axis. The annotated numbers indicate the absolute numbers of participants using ATL per time interval who have answered the question or not. Using Figure 13, we see that, with very few exceptions, the assessments of the statements are from participants who use ATL and the profiler more often and participants who use the profiler about 0 hours per month have not answered the question.

Based on Figure 13, we can therefore say that the assessments in Figure 12 mainly express real experience, since they are mainly from participants who use ATL and the profiler more than 0 h per month. The majority of the participants using ATL who assessed the statements are neutral towards them or agree with them. Only a few participants using ATL disagree with the statements.

Summary: There are different information needs for performance related information, which on the one hand involve the models used and on the other hand involve the execution of transformations. With regard to models, their size (number of model elements) and structure is important for a number of participants, as well as the traceability between input and output model, but also between input model and transformation (see Figure 10 and Table 7). It is also clear from the answers that the participants also consider more detailed information about the execution of a transformation to be important, e.g., number of investigated objects (see Figure 11 and Table 8). This could also explain the cautious evaluations of the statements about the ATL Profiler in Eclipse, which consist mainly of *"Neutral"* or *"Agree"* (see Figure 12), since this profiler only provides overview information about a transformation execution and only few details about the internal execution in the transformation engine, e.g., number of executed instructions. It is also interesting that the participants not only want more information about models and transformation executions, but also want support to improve the performance of a model transformation (see Table 8).

3.3. Analysis

In addition to the descriptive statistics presented in the previous sections, we present hypothesis testing statistics in this section and answer **RQ4**. In the following, we present the results for testing our hypotheses $\mathbf{H0_1}$ to $\mathbf{H0_7}$ (see Section 2.1) using the statistical tests described in Section 2.5. We used IBM SPSS to calculate the statistical tests. We provide the variables, the SPSS project file and the outputs under Groner et al. (2021).

$\mathbf{H0_1}$ There is no correlation between the size of input models (*modelElement_WA*) used and the satisfaction (*satisfaction*) with the execution time.

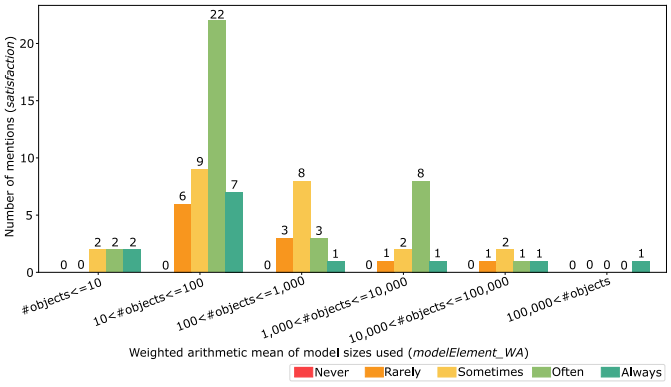

Figure 14 Number of mentions in terms of satisfaction (*satisfaction*) with performance per weighted arithmetic mean of model size used (*modelElement_WA*)

We ran a Kendall's τ_c correlation test to test $\mathbf{H0_1}$. The result shows that there is no significant correlation between the size of the input models and the satisfaction with the execution time, $n = 83$, $\tau_c = -0.086$, $p = 0.304$. Based on the test result we cannot reject the null hypothesis $\mathbf{H0_1}$.

Due to the fact that one participant did not answer *Q8*, we can only use the remaining 83 answers to test $\mathbf{H0_1}$. The negative correlation coefficient indicates that there might be a correlation between larger models and decreasing satisfaction, but this trend is not noticeable in Figure 14 and since the significance is greater than the defined significant level $\alpha = 0.05$, the correlation is not significant.

$\mathbf{H0_2}$ There is no difference in satisfaction (*satisfaction*) with the execution time between the group of participants who have expert knowledge about the engine, the group of participants who have limited knowledge about the engine or the group of participants for whom the engine is a black box (*role_rating*).

The Kruskal-Wallis-Test was used to test $\mathbf{H0_2}$. The result shows that there is no significant difference in satisfaction between the group of participants with expert knowledge about the engine (Developer: $M_{Rank} = 41.51$), the group of participants with limited knowledge about the engine (Researcher: $M_{Rank} = 40.85$) and pure users (User: $M_{Rank} = 49.92$), $H = 1.471$, $p = 0.484$, $n = 84$. Based on the test result we cannot reject the null hypothesis $\mathbf{H0_2}$.

	role_rating	*N*	Mean Rank
satisfaction	User	12	49.92
	Researcher	27	40.85
	Developer	45	41.51
	$\sum$	84	

Table 9 Ranks for $H0_2$

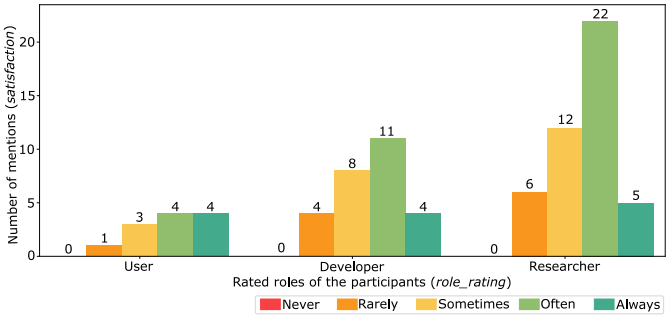

Figure 15 Number of mentions in terms of satisfaction (*satisfaction*), divided by the roles of the participants (*role_rating*)

Table 9 shows that the mean of ranks is higher for participants who are pure users and therefore have mainly no or very limited expert knowledge about the engine. This means that participants who do not have expert knowledge about the engine answered that they are more often satisfied with the execution time than participants who tend to have more knowledge about the engine. This trend is not indicated in Figure 15 and the difference is not significant.

H0$_3$ There is no difference in the distribution of possible expert knowledge about the engine (*role_rating*) between the group of participants who have already tried to analyze or improve performance, and the group of participants who have never tried (*analyze*).

The Mann-Whitney-U-Test was used to test **H0$_3$**. The result shows that there is a significant difference in possible expert knowledge between the group of participants who have already tried to analyze or improve performance ($M_{Rank} = 48.77$) and the group of participants who have never tried ($M_{Rank} = 35.60$), $U = 604.00, Z = -2.746, p = 0.006, n = 84, r = -0.299$. Based on the test result we have to reject the null hypothesis **H0$_3$**. Since r is between -0.30 and -0.10, this is just a small effect.

	analyze	*N*	Mean Rank	Sum of Ranks
role_rating	No	40	35.60	1424.00
	Yes	44	48.77	2146.00
	$\sum$	84		

Table 10 Ranks for H0$_3$

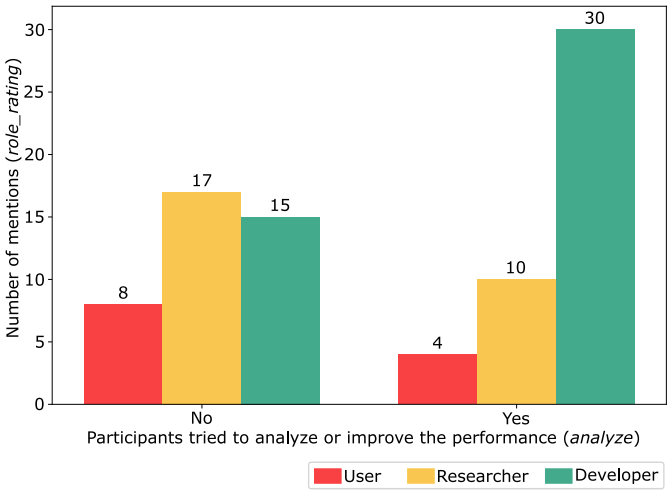

Figure 16 Number of mentions in terms of the expected knowledge about the engine based on the role (*role_rating*), divided by the group of participants who have tried to analyze or improve the performance and those who have not (*analyze*)

Table 10 shows that the mean of ranks and the sum of ranks are higher for participants who have tried to analyze or improve the performance of their transformations. This means that participants who have tried to analyze or improve the performance tend to have more expert knowledge about the engine than those who have not. This trend is also indicated in Figure 16, e.g., 30 out of 45 participants who belong to the group of engine developers have already tried to analyze or improve the performance.

H0$_4$ There is no difference in the distribution of satisfaction (*satisfaction*) with the execution time of a transformation between the group of participants who have already tried to analyze or improve the performance and the group of participants who have never tried (*analyze*).

The Mann-Whitney-U-Test was used to test **H0$_4$**. The result shows that there is a significant difference in satisfaction between the group of participants who have already tried to analyze or improve performance ($M_{Rank} = 37.50$) and the group of participants who have never tried ($M_{Rank} = 48.00$), $U = 660.00, Z = -2.091, p = 0.037, n = 84, r = -0.228$. Based on the test result we have to reject the null hypothesis **H0$_4$**. Since r is between -0.30 and -0.10, this is just a small effect.

	analyze	*N*	Mean Rank	Sum of Ranks
satisfaction	No	40	48.00	1920.00
	Yes	44	37.50	1650.00
	$\sum$	84		

Table 11 Ranks for H0$_4$

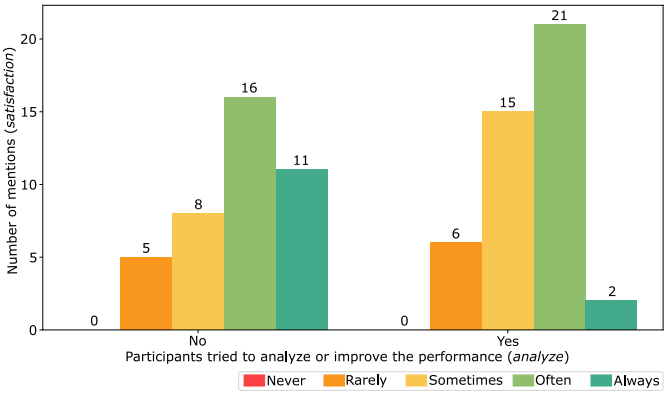

Figure 17 Number of mentions in terms of satisfaction (*satisfaction*), divided by the group of participants who have tried to analyze or improve the performance and those who have not (*analyze*)

Table 11 shows that the mean of ranks and the sum of ranks are higher for participants who have never tried to analyze or improve the performance of their transformations. This means that participants who have tried to analyze or improve the performance tend to be less satisfied than participants that have not tried to analyze or improve the performance. This trend is also indicated in Figure 17, since more participants who are sometimes or rarely satisfied have tried to analyze or improve the performance.

H0$_5$ There is no difference in the distribution of the sizes of the input models used (*modelElement_WA*) between the

group of participants who have already tried to analyze or improve the performance and the group of participants who have never tried (*analyze*).

The Mann-Whitney-U-Test was used to test **H0$_5$**. The result shows that there is a significant difference in the sizes of input models between the group of participants who have already tried to analyze or improve performance ($M_{Rank} = 51.15$) and the group of participants who have never tried ($M_{Rank} = 32.16$), $U = 466.50, Z = -3.909, p < 0.001, n = 83, r = -0.429$. Based on the test result we have to reject the null hypothesis **H0$_5$**. Since r is between -0.50 and -0.30, this is a medium effect.

	analyze	*N*	Mean Rank	Sum of Ranks
modelElement_WA	No	40	32.16	1286.50
	Yes	43	51.15	2199.55
	∑	83		

Table 12 Ranks for H0$_5$

Due to the fact that one participant did not answer *Q8*, we can only use the remaining 83 answers to test **H0$_5$**. Table 12 shows that the mean of ranks and the sum of ranks are higher for participants who have tried to analyze or improve the performance of their transformations. This means that participants who have tried to analyze or improve the performance tend to use larger models than participants that have not tried to analyze or improve the performance. This trend is also indicated in Figure 18, since more participants who use models larger than 1,000 model elements have tried to analyze or improve the performance.

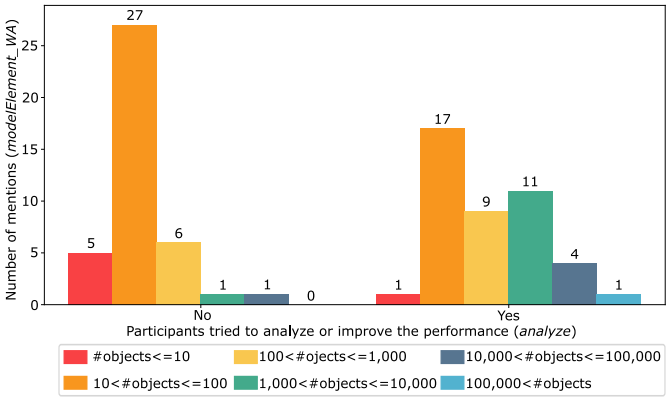

Figure 18 Number of mentions in terms of the weighted arithmetic mean of model sizes used (*modelElement_WA*), divided by the group of participants who have tried to analyze or improve the performance and those who have not (*analyze*)

H0$_6$ There is no difference in the distribution of the importance of not exceeding a certain execution time in the average case (*averageCase*) between the group of participants who have already tried to analyze or improve performance and the group of participants who have never tried (*analyze*).

The Mann-Whitney-U-Test was used to test **H0$_6$**. The result shows that there is a significant difference in the importance of not exceeding a certain execution time in the average case between the group of participants who have already tried to analyze or improve performance ($M_{Rank} = 49.56$) and the group of participants who have never tried ($M_{Rank} = 34.74$), $U = 569.50, Z = -2.837, p = 0.004, n = 84, r = -0.310$. Based on the test result we have to reject the null hypothesis **H0$_6$**. Since r is between -0.50 and -0.30, this is a medium effect.

	analyze	*N*	Mean Rank	Sum of Ranks
averageCase	No	40	34.74	1389.50
	Yes	44	49.56	2180.50
	∑	84		

Table 13 Ranks for H0$_6$

Table 13 shows that the mean of ranks and the sum of ranks are higher for participants who have tried to analyze or improve the performance of their transformations. This means that participants who have tried to analyze or improve performance tend to be more interested in not exceeding a certain execution time in the average case than participants who have not tried to analyze or improve performance. This trend is also indicated in Figure 19, since more participants who consider it at least *"very important"* that a certain execution time is not exceeded in the average case have tried to analyze or improve the performance.

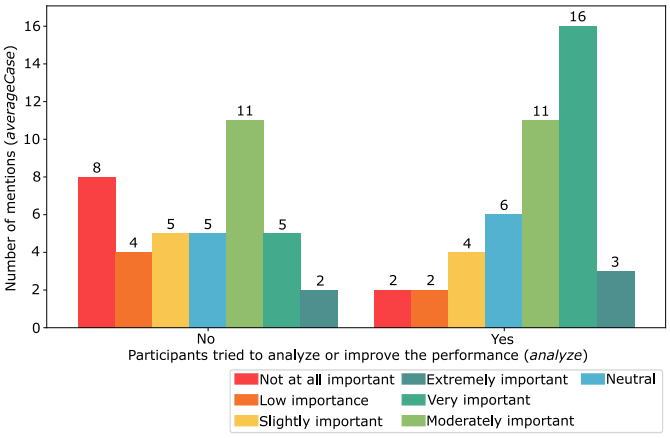

Figure 19 Number of mentions in terms of the importance that a certain execution time is not exceeded in the average case (*averageCase*), divided by the group of participants who have tried to analyze or improve the performance and those who have not (*analyze*)

H0$_7$ There is no difference in the distribution of the importance of not exceeding a certain execution time in the worst case

(*worstCase*) between the group of participants who have already tried to analyze or improve performance and the group of participants who have never tried (*analyze*).

The Mann-Whitney-U-Test was used to test **H0$_7$**. The result shows that there is no significant difference in the importance of not exceeding a certain execution time in the worst case between the group of participants who have already tried to analyze or improve performance ($M_{Rank} = 46.05$) and the group of participants who have never tried ($M_{Rank} = 38.60$), $U = 724.00, Z = -1.424, p = 0.155, n = 84$. Based on the test result we cannot reject the null hypothesis **H0$_7$**.

	analyze	*N*	Mean Rank	Sum of Ranks
worstCase	No	40	38.60	1544.00
	Yes	44	46.05	2026.00
	∑	84		

Table 14 Ranks for H0$_7$

Table 14 shows that the mean of ranks and the sum of ranks are higher for participants who have tried to analyze or improve the performance of their transformations. This means that participants who have tried to analyze or improve performance tend to be more interested in not exceeding a certain execution time in the worst case than participants who have not tried to analyze or improve performance. This trend is slightly indicated in Figure 20, but the difference is not significant.

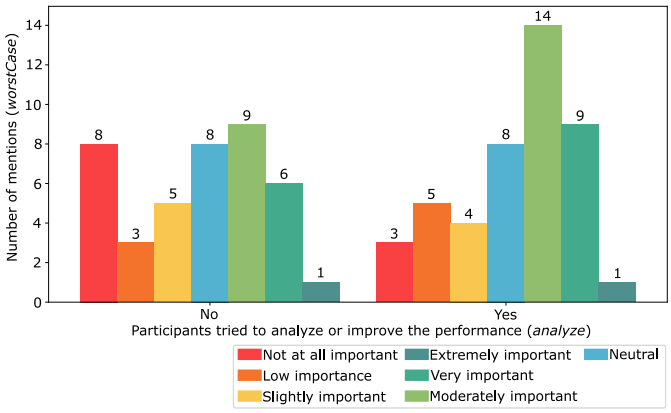

Figure 20 Number of mentions in terms of the importance that a certain execution time is not exceeded in the worst case (*worstCase*), divided by the group of participants who have tried to analyze or improve the performance and those who have not (*analyze*)

It is noticeable that there is no significant difference between the two groups of participants in terms of the importance of not exceeding a certain execution time in the worst case. This may be due to the fact that the worst case, is less important to the participants than the average case (see Figure 8). If we compare the two cases, we can see that the difference in ratio is not very big. For example, of the 48 participants who rated the average case at least *"moderately important"*, 63% also have tried to analyze or to improve the performance and of the 40 participants who rated the worst case at least *"moderately important"*, 60% also have tried to analyze or improve the performance.

Summary: Table 15 summarizes the results of the hypothesis testing. Based on our test results, we can identify three significant differences between the participants who have tried to analyze or improve the performance and those who have not. Participants who have tried to analyze or improve the performance tend to 1) have more knowledge about the engine, 2) are less satisfied with the execution time, 3) use large models, and 4) consider it more important that a certain execution time is not exceeded in the average case.

4. Related Work

In this section, we discuss related work, which, with respect to the performance of model transformations, mainly consists of works that achieve performance improvements by adjusting the transformation engine or by adjusting the definition of a transformation. Due to the amount of available publications in this area, we will only give an overview of some examples.

There is a line of research mainly focusing on improving the transformation engine, that executes transformations e.g., G. Varró et al. 2015; Boronat 2018; Vizhanyo et al. 2004; Giese et al. 2009; Veit Batz et al. 2008. Such works try to improve the performance by improving the algorithms within the engine. For example, Fritsche et al. (2017) present a look-ahead strategy to prevent unnecessary applications of transformations which have to be undone later. Fleck et al. (2015) also presents an approach to determine the most efficient sequence of transformation applications.

Another line of research focuses on developing different approaches to execute transformations to improve their performance, e.g., by parallel, incremental or distributed execution of transformations (cf. Benelallam et al. 2016; Burgueño et al. 2016; Jouault & Tisi 2010; Szárnyas et al. 2014; Tisi et al. 2013; D. Varró et al. 2016).

Other works investigate how the definition of a transformation or the model transformation language used affects performance. For example, Wimmer et al. (2012) present refactorings for transformations defined in ATL and also examine the performance changes after application of the refactorings, concluding that some improve the performance. Taentzer et al. (2012) also presents refactorings that improve the performance of transformations defined in Henshin. The work of Mészáros et al. (2010) demonstrates how performance improvements of up to 70% can be achieved by manually optimizing the definition of a transformation. Bruni & Lluch Lafuente (2012) examine the performance differences of different definitions of transformations and present additionally guidelines to improve the performance by changing the transformation definition. Van Amstel et al. (2011) not only examine the performance differences of different ways to define a transformation in ATL, but also compare the

	Meaning	N	τ_c	p^a
H0$_1$	There is no significant correlation between the size of the models used and how often the participants are satisfied with the performance.	83	−0.086	0.304

	Meaning	N	H	p^a	r^b
H0$_2$	There is no significant difference in satisfaction between the group of participants who have expert knowledge about the engine, the group of participants who have limited knowledge about the engine or the group of participants for whom the engine is a black box.	84	1.471	0.484	–

	Meaning	N	U	Z	p^a	r^b
H0$_3$	Participants who have tried to analyze or improve the performance tend to have more knowledge about the engine.	84	604.00	−2.746	0.006*	−0.299
H0$_4$	Participants who have tried to analyze or improve the performance tend to be less satisfied with the execution time.	84	660.00	−2.091	0.037*	−0.228
H0$_5$	Participants who have tried to analyze or improve the performance tend to use larger models.	83	466.50	−3.909	< 0.001*	−0.429
H0$_6$	Participants who have tried to analyze or improve performance tend to consider it more important that a certain execution time is not exceeded in the average case.	84	569.50	−2.837	0.004*	−0.310
H0$_7$	There is no significant difference in the assessment of the importance of not exceeding a certain execution time in the worst case between participants who have tried to analyze or improve performance and those who have not.	84	724.00	−1.424	0.155	–

N: Sample Size, H: Test Statistic for Kruskal-Wallis-Test, U: Test Statistic for Mann–Whitney test, Z: Z-Score, p: Significance of the Test, r: Effect Size

a Significant p-values are marked with *.

b Effect sizes are only given if the test was significant. If effect sizes are negative, this is due to the order of the compared groups, which is negligible.

Table 15 Summary of the results of the hypothesis testing

performance differences between the transformation languages ATL, QVTo, and QVTr.

The online survey presented in this paper was part of a mixed method study and we have already described the qualitative part of this study in Groner et al. (2020a). We present, in Groner et al. (2020a) the results of our semi-structured interviews on how transformation developers deal with performance issues, what causes they found and how they tried to fix performance issues. We also used some of the results from our online study in Groner et al. (2020a), in order to motivate the relevance of performance of model transformations and to gain background information to find suitable interviewees. Through this study we were able to identify different strategies to find or prevent performance issues in model transformations. We were also able to identify different causes and solutions. In addition, we were able to compile a list of the interviewees' ideas, which in their opinion can help to find causes or solve performance issues.

Other empirical studies in the field of model-driven software development and model transformations often focus on whether they help to develop software more effectively, such as Hutchinson et al. (2011) and Liebel et al. (2018). To our knowledge, apart from our own study in Groner et al. (2020a), there is no other empirical study that systematically investigates the experience of transformation developers in regard to performance of model transformations.

5. Conclusion and Future Work

In this paper, we present the quantitative part of our mixed-methods study consisting of an online survey about the relevance of performance of model transformations. We collected the following three different types of information in our questionnaire: 1) general information about the participants, 2) information to assess whether the performance of model transformations is

relevant, and 3) what information need exists.

In total, we have three different results regarding the performance of model transformations:

1) The performance of model transformations seems to have a relevance for some of the participants. Our results in Section 3.2.1 indicate that a certain performance is desired, but not always achieved.

2) There is an information need about the execution of model transformations and the models used to understand and improve the performance. In addition, the participants would also like to get hints on how to improve performance (see Table 8, category **Support**).

3) Participants who have tried to analyze or improve the performance tend to have more knowledge about the engine, are less satisfied with the execution time, use large models and consider it more important that a certain execution time is not exceeded on average.

It is noticeable that, on the one hand, in the group of participants who have already tried to analyze or improve the performance, there are significantly more participants who have expert knowledge about the engine. On the other hand, there is a need for more detailed information about the transformation execution. This indicates that there is a lack of support for analyzing or improving the performance of model transformations without detailed knowledge about the engine, although there is a need for support, since a certain performance is desired.

In our future work, we aim to fill this gap by developing an approach that helps to analyze and improve the performance of transformations. For this purpose, we will use the results of this study, in particular the information about the transformation execution and the models used, as well as the hints on how to improve performance mentioned by the participants, as a basis for our approach.

Acknowledgments

This work was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – Ti 803/4-1 and BE 4796/3-1

References

Amrani, M., Combemale, B., Lucio, L., Selim, G. M. K., Dingel, J., Traon, Y. L., … Cordy, J. R. (2015). Formal Verification Techniques for Model Transformations: A Tridimensional Classification. *Journal of Object Technology*, *14*(3), 1:1–43. doi: 10.5381/jot.2015.14.3.a1

Amrani, M., Dingel, J., Lambers, L., Lúcio, L., Salay, R., Selim, G., … Wimmer, M. (2012). Towards a Model Transformation Intent Catalog. In *Proceedings of the first workshop on the analysis of model transformations (amt'12)* (pp. 3–8). doi: 10.1145/2432497.2432499

Benelallam, A., Tisi, M., Cuadrado, J. S., de Lara, J., & Cabot, J. (2016). Efficient Model Partitioning for Distributed Model Transformations. In *Proceedings of the 2016 acm sigplan international conference on software language engineering (sle'16)* (pp. 226–238). New York, NY, USA: ACM. doi: 10.1145/2997364.2997385

Bettini, L. (2016). *Implementing Domain-Specific Languages with Xtext and Xtend* (2nd ed.). Birmingham, UK: Packt Publishing.

Booth, A., Sutton, A., & Papaioannou, D. (2016). *Systematic Approaches to a Successful Literature Review* (2nd ed.). SAGE.

Boronat, A. (2018). Expressive and Efficient Model Transformation with an Internal DSL of Xtend. In *Proceedings of the 21th acm/ieee international conference on model driven engineering languages and systems (models'18)* (pp. 78–88). New York, NY, USA: ACM. doi: 10.1145/3239372.3239386

Bruni, R., & Lluch Lafuente, A. (2012). Evaluating the Performance of Model Transformation Styles in Maude. In *Proceedings of the 8th international workshop on formal aspects of component software (facs'11)* (pp. 79–96). Berlin, Heidelberg: Springer. doi: 10.1007/978-3-642-35743-5_6

Burgueño, L., Cabot, J., & Gérard, S. (2019, July). The Future of Model Transformation Languages: An Open Community Discussion. *Journal of Object Technology*, *18*(3), 7:1-11. (The 12th International Conference on Model Transformations) doi: 10.5381/jot.2019.18.3.a7

Burgueño, L., Wimmer, M., & Vallecillo, A. (2016). A Linda-based platform for the parallel execution of out-place model transformations. *Information and Software Technology*, *79*, 17–35. doi: 10.1016/j.infsof.2016.06.001

Cheng, Z., Tisi, M., & Douence, R. (2020). CoqTL: a Coq DSL for rule-based model transformation. *Software and Systems Modeling*, *19*(2), 425–439. doi: 10.1007/s10270-019-00765-6

Cohen, J. (1992). A Power Primer. *Psychological Bulletin*, *112*(1), 155–159. doi: 10.1037/0033-2909.112.1.155

Cordy, J. R. (2006). The TXL Source Transformation Language. *Science of Computer Programming*, *61*(3), 190–210. (Special Issue on The Fourth Workshop on Language Descriptions, Tools, and Applications (LDTA'04)) doi: 10.1016/j.scico.2006.04.002

Eclipse Foundation, Inc. (2019). *What is Acceleo?* Retrieved from https://www.eclipse.org/acceleo/overview.html (Accessed: 28.08.2020)

Eclipse Foundation, Inc. (2020). *Eclipse Xpand.* Retrieved from https://projects.eclipse.org/projects/modeling.m2t.xpand (Accessed: 28.08.2020)

EJ-Technologies. (2020). *JProfiler: The Award-Winning All-In-One Java Profiler.* Retrieved from https://www.ej-technologies.com/products/jprofiler/overview.html (Accessed: 19.08.2020)

Field, A. (Ed.). (2009). *Discovering statistics using SPSS: (and sex and drugs and rock 'n' roll)* (3. ed. ed.). Los Angeles, London, New Delhi, Singapore, Washington DC: SAGE.

Fleck, M., Troya, J., & Wimmer, M. (2015). Marrying Search-based Optimization and Model Transformation Technology. In *Proceedings of the 1st north american search based software engineering symposium (nasbase'15)* (pp. 1–16). Retrieved from https://publik.tuwien.ac.at/files/PubDat_237899.pdf

Forza, C. (2002). Survey research in operations management: a process-based perspective. *International journal*

of operations & production management. doi: 10.1108/01443570210414310

Fritsche, L., Leblebici, E., Anjorin, A., & Schürr, A. (2017). A Look-Ahead Strategy for Rule-Based Model Transformations. In *Proceedings of the 11th international workshop on models and evolution (me)* (pp. 45–53). CEUR-WS.org. Retrieved from http://ceur-ws.org/Vol-2019/me_1.pdf

Fritz, C. O., Morris, P. E., & Richler, J. J. (2012). Effect size estimates: Current use, calculations, and interpretation. *Journal of Experimental Psychology: General*, *141*(1), 2–18. doi: 10.1037/a0024338

Giese, H., Hildebrandt, S., & Seibel, A. (2009). Improved Flexibility and Scalability by Interpreting Story Diagrams. *Electronic Communications of the EASST*, *18*, 1–12. doi: 10.14279/tuj.eceasst.18.268

Götz, S., Tichy, M., & Groner, R. (2020). Claimed Advantages and Disadvantages of (dedicated) Model Transformation Languages: A Systematic Literature Review. *Software and Systems Modeling*, 1–35. doi: 10.1007/s10270-020-00815-4

Gradecki, J. D., & Cole, J. (2003). *Mastering Apache Velocity*. Indianapolis, Indiana: John Wiley & Sons.

Groner, R., Beaucamp, L., Tichy, M., & Becker, S. (2020a). 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* (p. 308–319). New York, NY, USA: Association for Computing Machinery. doi: 10.1145/3365438.3410950

Groner, R., Beaucamp, L., Tichy, M., & Becker, S. (2020b). *An Exploratory Study on Performance Engineering in Model Transformations : Data of the mixed method study.* Open Access Repositorium der Universität Ulm und Technischen Hochschule Ulm. doi: 10.18725/OPARU-32365

Groner, R., Juhnke, K., Götz, S., Tichy, M., Becker, S., Vijayshree, V., & Frank, S. (2021). *A Survey on the Relevance of the Performance of Model Transformations: Data of the Participant Search and the Questionnaire.* Open Access Repositorium der Universität Ulm und Technischen Hochschule Ulm. doi: 10.18725/OPARU-38188

Howell, D. C. (2009). *Statistical Methods for Psychology* (7th ed.). Belmont, CA: Wadsworth, Cengage Learning.

Hutchinson, J. E., Rouncefield, M., & Whittle, J. (2011). Model-Driven Engineering Practices in Industry. In *Proceedings of the 33rd international conference on software engineering (icse'11)* (pp. 633–642). ACM. doi: 10.1145/1985793.1985882

Jouault, F., Allilaire, F., Bézivin, J., & Kurtev, I. (2008). ATL: A Model Transformation Tool. *Science of Computer Programming*, *72*(1-2), 31–39. doi: 10.1016/j.scico.2007.08.002

Jouault, F., & Tisi, M. (2010). Towards Incremental Execution of ATL Transformations. In *Proceedings of the 3rd international conference on theory and practice of model transformations (icmt'10)* (pp. 123–137). Springer. doi: 10.1007/978-3-642-13688-7_9

Kastenberg, H., & Rensink, A. (2006). Model Checking Dynamic States in GROOVE. In *Proceedings of the 13th international spin workshop on model checking software (spin'06)* (pp. 299–305). Berlin, Heidelberg: Springer. (Lecture Notes in Computer Science, vol 3925) doi: 10.1007/11691617_19

Kendall, M. G. (1938). A New Measure of Rank Correlation. *Biometrika*, *30*(1-2), 81–93. doi: 10.1093/biomet/30.1-2.81

Kitchenham, B., Brereton, O. P., Budgen, D., Turner, M., Bailey, J., & Linkman, S. (2009). Systematic literature reviews in software engineering–A systematic literature review. *Information and software technology*, *51*(1), 7–15. doi: 10.1016/j.infsof.2008.09.009

Klare, H. (2018). *Welcome to the Vitruv Wiki.* Retrieved from https://github.com/vitruv-tools/Vitruv/wiki (Accessed: 28.08.2020)

Kolovos, D. S., Paige, R. F., & Polack, F. A. C. (2006). The Epsilon Object Language (EOL). In A. Rensink & J. Warmer (Eds.), *Model driven architecture – foundations and applications* (pp. 128–142). Berlin, Heidelberg: Springer Berlin Heidelberg. doi: 10.1007/11787044_11

Kruskal, W. H., & Wallis, W. A. (1952). Use of Ranks in One-Criterion Variance Analysis. *Journal of the American Statistical Association*, *47*(260), 583–621. doi: 10.1080/01621459.1952.10483441

Kurtev, I. (2008). State of the Art of QVT: A Model Transformation Language Standard. In A. Schürr, M. Nagl, & A. Zündorf (Eds.), *Applications of Graph Transformations with Industrial Relevance* (pp. 377–393). Berlin, Heidelberg: Springer Berlin Heidelberg. doi: 10.1007/978-3-540-89020-1_26

Lano, K. (2014). *The UML-RSDS Manual* (Tech. Rep.). Department of Informatics, King's College London. doi: 10.13140/RG.2.1.1052.0487

Liebel, G., Marko, N., Tichy, M., Leitner, A., & Hansson, J. (2018). Model-based engineering in the embedded systems domain: an industrial survey on the state-of-practice. *Software and Systems Modeling*, *17*(1), 91–113. doi: 10.1007/s10270-016-0523-3

Malhotra, N. K. (2006). The Handbook of Marketing Research: Uses, Misuses, and Future Advances. In (pp. 83–94). Sage Publications Thousand Oaks, CA.

Mann, H. B., & Whitney, D. R. (1947). On a Test of Whether one of Two Random Variables is Stochastically Larger than the Other. *The Annals of Mathematical Statistics*, *18*(1), 50–60. doi: 10.1214/aoms/1177730491

Mészáros, T., Mezei, G., Levendovszky, T., & Asztalos, M. (2010). Manual and automated performance optimization of model transformation systems. *International Journal on Software Tools for Technology Transfer*, *12*(3), 231–243. doi: 10.1007/s10009-010-0151-0

Nulty, D. D. (2008). The adequacy of response rates to online and paper surveys: what can be done? *Assessment & evaluation in higher education*, *33*(3), 301–314. doi: 10.1080/02602930701293231

(OMG), O. M. G. (2016). *Meta Object Facility (MOF) 2.0 Query/View/Transformation Specification Version 1.3.* Retrieved from https://www.omg.org/spec/QVT/1.3/PDF (Accessed: 02.09.2020)

Pech, V., Shatalin, A., & Voelter, M. (2013). JetBrains MPS as a Tool for Extending Java. In *Proceedings of the 2013 international conference on principles and practices of programming on the java platform: Virtual machines,*

languages, and tools (pppj'13) (pp. 165–168). ACM. doi: 10.1145/2500828.2500846

Piers, W. (2010). ATL 3.1–Industrialization improvements. In *Proceedings of the 2nd international workshop on model transformation with atl (mtatl'10)* (pp. 34–38). CEUR-WS.org. Retrieved from http://ceur-ws.org/Vol-711/paper4.pdf

Sanchez Cuadrado, J., Burgueno, L., Wimmer, M., & Vallecillo, A. (2020). Efficient execution of ATL model transformations using static analysis and parallelism. *IEEE Transactions on Software Engineering*, 1-1. doi: 10.1109/TSE.2020.3011388

Saris, W. E., & Gallhofer, I. N. (2014). *Design, Evaluation, and Analysis of Questionnaires for Survey Research* (2nd ed. ed.). Hoboken, NJ: J. Wiley & Sons. Retrieved from https://learning.oreilly.com/library/view/-/9781118634554/?ar (1 online resource (1 v.))

Singer, J., Sim, S. E., & Lethbridge, T. C. (2008). Software Engineering Data Collection for Field Studies. In F. Shull, J. Singer, & D. I. K. Sjøberg (Eds.), *Guide to Advanced Empirical Software Engineering* (pp. 9–34). London: Springer London. doi: 10.1007/978-1-84800-044-5_1

Spearman, C. (1910). Correlation Calculated from Faculty Data. *British Journal of Psychology, 1904-1920, 3*, 271–295. doi: 10.1111/j.2044-8295.1910.tb00206.x

Steinberg, D., Budinsky, F., Paternostro, M., & Merks, E. (2008). *EMF: Eclipse Modeling Framework* (2nd ed.). Upper Saddle River, N.J.: Addison-Wesley.

Strüber, D., Born, K., Gill, K. D., Groner, R., Kehrer, T., Ohrndorf, M., & Tichy, M. (2017). Henshin: A Usability-Focused Framework for EMF Model Transformation Development. In *Proceedings of the 10th international conference on graph transformation (icgt'17)* (pp. 196–208). Springer. (Lecture Notes in Computer Science, vol 10373) doi: 10.1007/978-3-319-61470-0_12

Stuart, A. (1953). The Estimation and Comparison of Strengths of Association in Contingency Tables. *Biometrika*, *40*(1/2), 105–110. doi: 10.2307/2333101

Szárnyas, G., Izsó, B., Ráth, I., Harmath, D., Bergmann, G., & Varró, D. (2014). IncQuery-D: A Distributed Incremental Model Query Framework in the Cloud. In *Proceedings of the 17th international conference on model-driven engineering languages and systems (models'14)* (pp. 653–669). Cham: Springer. doi: 10.1007/978-3-319-11653-2_40

Taentzer, G., Arendt, T., Ermel, C., & Heckel, R. (2012). Towards Refactoring of Rule-Based, In-Place Model Transformation Systems. In *Proceedings of the 1st workshop on the analysis of model transformations (amt'12)* (pp. 41–46). ACM. doi: 10.1145/2432497.2432506

Tisi, M., Martínez, S., & Choura, H. (2013). Parallel Execution of ATL Transformation Rules. In *Proceedings of the 16th international conference on model-driven engineering languages and systems (models'13)* (pp. 656–672). Berlin, Heidelberg: Springer. (Lecture Notes in Computer Science, vol 8107) doi: 10.1007/978-3-642-41533-3_40

Vagias, W. M. (2006). Likert-Type Scale Response Anchors. *Clemson International Institute for Tourism & Research Development, Department of Parks, Recreation and Tourism Management. Clemson University*.

Van Amstel, M., Bosems, S., Kurtev, I., & Pires, L. F. (2011). Performance in Model Transformations: Experiments with ATL and QVT. In *Proceedings of the 4th international conference on theory and practice of model transformations (icmt'11)* (pp. 198–212). Berlin, Heidelberg: Springer. doi: 10.1007/978-3-642-21732-6_14

Varró, D., Bergmann, G., Hegedüs, Á., Horváth, Á., Ráth, I., & Ujhelyi, Z. (2016). Road to a reactive and incremental model transformation platform: three generations of the VIATRA framework. *Software & Systems Modeling*, *15*(3), 609–629. doi: 10.1007/s10270-016-0530-4

Varró, G., Deckwerth, F., Wieber, M., & Schürr, A. (2015). An algorithm for generating model-sensitive search plans for pattern matching on EMF models. *Software & Systems Modeling*, *14*(2), 597–621. doi: 10.1007/s10270-013-0372-2

Veit Batz, G., Kroll, M., & Geiß, R. (2008). A First Experimental Evaluation of Search Plan Driven Graph Pattern Matching. In *Proceedings of the 3rd international symposium on applications of graph transformations with industrial relevance (agtive'07)* (pp. 471–486). Berlin, Heidelberg: Springer. (Lecture Notes in Computer Science, vol 5088) doi: 10.1007/978-3-540-89020-1_32

Vizhanyo, A., Agrawal, A., & Shi, F. (2004). Towards Generation of Efficient Transformations. In *Proceedings of the 3rd international conference on generative programming and component engineering (gpce'04)* (pp. 298–316). Berlin, Heidelberg: Springer. (Lecture Notes in Computer Science, vol 3286) doi: 10.1007/978-3-540-30175-2_16

W3C. (2020). *XSLT – Transformation.* Retrieved from https://www.w3schools.com/xml/xsl_transformation.asp (Accessed: 28.08.2020)

Wimmer, M., Martínez, S., Jouault, F., & Cabot, J. (2012). A Catalogue of Refactorings for Model-to-Model Transformations. *Journal of Object Technology*, *11*(2), 1–40. doi: 10.5381/jot.2012.11.2.a2

About the authors

Raffaela Groner is a Ph.D. student at Ulm University. Her research is focused on the performance of model transformations. Prior, she studied Computer Science at the Ulm University, where she received her M. Sc. in Computer Science. You can contact the author at raffaela.groner@uni-ulm.de.

Katharina Juhnke is a PostDoc at Ulm University. She received her M. Sc. in Computer Science at Leipzig University of Applied Sciences (HTWK Leipzig) and worked as an IT Consultant, Requirements & Usability Engineer. During her doctoral studies she worked at Mercedes-Benz Passenger Car Development focusing on improving the quality of test case specifications that are relevant for system and system integration testing of embedded systems. Her research interests are primarily embedded software testing, domain specific languages as well as usability engineering and empirical research methods. You can contact the author at katharina.juhnke@uni-ulm.de.

Stefan Götz is a Ph.D. student at Ulm University. His research is focused on topics surrounding the development and evaluation of model transformation languages. Prior to his work as a Ph.D. student he was a student of Software Engineering at Ulm University where he received his M.Sc. You can contact the author at stefan.goetz@uni-ulm.de.

Matthias Tichy is full professor for software engineering at Ulm University and director of the Institute of Software Engineering and Programming languages. His main research focuses on model-driven software engineering, particularly for cyber-physical systems. He works on requirements engineering, dependability, and validation and verification complemented by empirical research techniques. He is a regular member of program committees for conferences and workshops in the area of software engineering and model driven development. He is co-author of over 110 peer-reviewed publications. You can contact the author at matthias.tichy@uni-ulm.de.

Steffen Becker is full professor for Software Quality and Architecture at the University of Stuttgart. His main research areas are software architecture, software performance and model-driven quality analyses, mainly for Cloud and IoT systems. He is known for being a core researcher in the Palladio component model project — a simulator for different software qualities based on the software's architecture. He is a member of the steering committee of the International Conference on Software Architecture (ICSA) and a PC member of various conferences related to performance, software quality and model driven software development. You can contact the author at steffen.becker@iste.uni-stuttgart.de.

Vijayshree Vijayshree is a Ph.D. student at the University of Stuttgart. Her research is focused on the Performance analysis of model transformation. She has received her M.Tech in Computer Science at Visvesvaraya Technological University (VTU) India. You can contact the author at vijayshree.vijayshree@iste.uni-stuttgart.de.

Sebastian Frank is a Ph.D. student at the University of Stuttgart. His research is focused on software architecture optimization and requirements engineering in the context of microservice-based software systems. He received his M.Sc. at the University of Stuttgart. You can contact the author at sebastian.frank@iste.uni-stuttgart.de.