STSM Report: Evaluating the efficiency in use of search-based automated model merge technique

Ankica Barišić
NOVA Laboratory for Computer Science and Informatics
Departamento de Informática, Faculdade de Ciências e Tecnologia,
Universidade Nova de Lisboa, Portugal
Email: a.barisic@campus.fct.unl.pt

I. Purpose of the Visit

This report contributes to WG2, by evaluating the adequacy of the technology for model differencing and merging, which resulted from the challenge of the increasing demand for collaboration features in industrial applications of model-driven engineering (MDE).

The FP7 project MONDO, developed at Budapest University of Technology and Economics, aims to tackle the challenge of scalability in MDE in a comprehensive manner by developing the theoretical foundations and an open-source implementation of a platform for scalable modeling and model management. This technique uses rule-based design space exploration to search the space of solution candidates that represent conflict-free merged models. The approach allows engineers to easily incorporate domain-specific knowledge into the merge process to provide better solutions.

We systematically evaluate the efficiency of the technique from the user point of view using a reactive experimental software engineering approach. In particular, we asked users to merge the different versions of same model. These empirical tests include the involvement of the intended end users (i.e. engineers), which are expected to confirm the impact of design decisions. The experiment participants were observed while performing the tasks of different complexity. Evaluation took place at a the Budapest University of Technology and Economics.

Achieving scalability in modelling and MDE involves being able to construct large models and domain-specific languages in a systematic manner, enabling teams of modellers to construct and refine large models in a collaborative manner, advancing the state-of-the-art in model querying and transformations tools so that they can cope with large models (of the scale of millions of model elements), and providing an infrastructure for efficient storage, indexing and retrieval of large models.

To address these challenges, MONDO brings together partners with a long track record in performing internationally-leading research on software modelling and MDE, and delivering research results in the form of robust, widely-used and sustainable open-source software, with industrial partners active in the fields of reverse engineering and systems integration, and a global industry consortium including more than 400 organisations from all sectors of IT.

A. Technique

Industrial applications of MDE to develop large and complex systems resulted in an increasing demand for collaboration features. However, use cases such as model differencing and merging have turned out to be a difficult challenge, due to

- the graph-like nature of models, and
- the complexity of certain operations (e.g. hierarchy refactoring) that are common today.

MONDO European FP7 project [14] aims to tackle the challenge of scalability in MDE in a comprehensive manner by developing the theoretical foundations and an open-source implementation of a platform for scalable modelling and model management.

The tool support developed within this project at Budapest University of Technology and Economics, named DSE Merge, presents a novel search-based automated model merge [11] which builds on off-the-shelf tools for the model comparison step, but uses guided rule-based design space exploration (DSE) [10] for merging models. In general, rule-based DSE aims to search and identify various design candidates to full certain structural and numeric constraints. The exploration starts from an initial model and systematically traverses paths by applying operators. In this context, the results of model comparison will be the initial model, while a target design candidates will represent the conflict-free merged model.

While many existing model merge approaches detect conflicts statically in a preprocessing phase, this DSE technique carries out conflict detection dynamically, during exploration time as conflicting rule activations and constraint violations. Then multiple consistent resolutions of conflicts are presented to the domain experts. This technique allows to incorporate domain-specific knowledge into the merge process by additional constraints, goals and operations to provide better solutions.

B. Evaluation approach

Practitioners are still experiencing problems in order to adopt modeling techniques, in practice. Among other factors,

developers seem to underestimate the importance of really aligning the domain-specific support with the needs of their end users. We argue that for this kind of techniques the measure of success has to be captured by assessing the impact of using the technique, in a realistic context of use, by its target domain users. Investment into this assessment, commonly called Usability evaluation, is justified by reduction of development costs and increased revenues for other software products, brought by an improved effectiveness and efficiency by their end users [13].

Existing Experimental Software Engineering techniques [9] combined with Usability Engineering techniques [15] can be adopted in order to support this evaluations. This includes application of reactive experimental approaches, based on which the support should be tested empirically with humans using systematic techniques to confirm the impact of design decisions on usability of approach.

The proposed evaluation approach is illustrated by a real life case study of the usability evaluation of a domain-specific language (DSL) for the High Energy Physics [8]. It is also applied in the context of iterative development of a DSL for humanitarian campaigns flow specification (FlowSL) [6]. Finally, the approach is being applied in a context of DSL summer schools and in several master theses developed at NOVA University at Lisbon, involving industrial partners, among which we can highlight the example of developing and evaluating DSL that is meant to enable the children to program the robots [12].

II. Work Carried out during STSM

The experiment preparation started immediately upon receiving the positive answer from STSM committee. After obtaining the information about possible availability of participants in the period from 1-15 December, planned 1-week visit was more convenient to take place during second week of December (6-13 of December). First days of visit applicant got introduced to host team and worked on validating and improving materials needed for experiment. Pilot session took place on 10th of December in the morning, while experiment itself was scheduled for 11th December in afternoon. The collected data on machines that were used during experiment was delivered by the 17th December. The development team from Budapest University rated the success of delivered projects and STSM applicant performed other result analysis during the first week of January 2016. Finally, the report was conducted and submitted by the 13 of January.

A. Experiment Preparation

The host institution provided the subjects with different level of the modelling expertise that were to participate in experiment execution. Based on participant expertise in the domain, the availability questionnaire was conducted in advance in order to profile experiment subjects and get idea about availability for experiment (see Figure 1). Meanwhile, the development team was preparing the demo for DSE Merge tool, the tasks and training material, and finally the virtual machine environment. All provided materials were verified and improved during the STSM visit. The materials were evaluated during the pilot session that took place before the experiment execution.

The participants of the pilot session were two academics that are part of the development team, although did not participate in development of the evaluated tool.

Before starting the experiment, decisions have to be made concerning the context of the experiment, the hypotheses under study, the set of independent and dependent variables that will be used to evaluate the hypotheses, the selection of subjects participating in the experiment, the experiment's design and instrumentation, and also an evaluation of the experiment's validity. Only after all these details are sorted out should the experiment be performed. The outcome of planning is the experimental evaluation design, which should encompass enough details in order to be independently replicable.

B. Experiment Objective

The goal of experiment is to answer the following research question:

- *How usable is a proposed technique for performing the model merge operations when compared to alternative?*

In particular we tested following hypothesis:

- *H1:* By using DSE Merge engineers can perform model merge operations more effectively when compared to alternative.
- *H2:* By using DSE Merge engineers can perform model merge operations more efficiently when compared to alternative.
- *H3:* By using DSE Merge engineers can perform model merge operations more satisfactory when compared to alternative.
- *H4:* By using DSE Merge engineers can perform model merge operations with less cognitive effort when compared to alternative.

C. Experiment Context

The planning of experiment started by defining explicitly the context of use for technology under evaluation, namely DSE Merge tool.

The alternative, i.e. baseline support for model merge problem that is suitable for experimental comparison is identified to be following:

- Diff Merge [3] shows all the changes to user where the changes have to be applied manually one by one. Its strength is the user-friendly UI which is very intuitive for the novice users.
- EMF Compare [2] is default comparison and merge tool in the Eclipse environment. In each steps, the tool show only a subset of the changes that the user has to apply into the merge model. Its strength is the capability of handling very complex impacts of changes.

The alternative solutions are meant to support software engineers during model merge process. The additional benefit

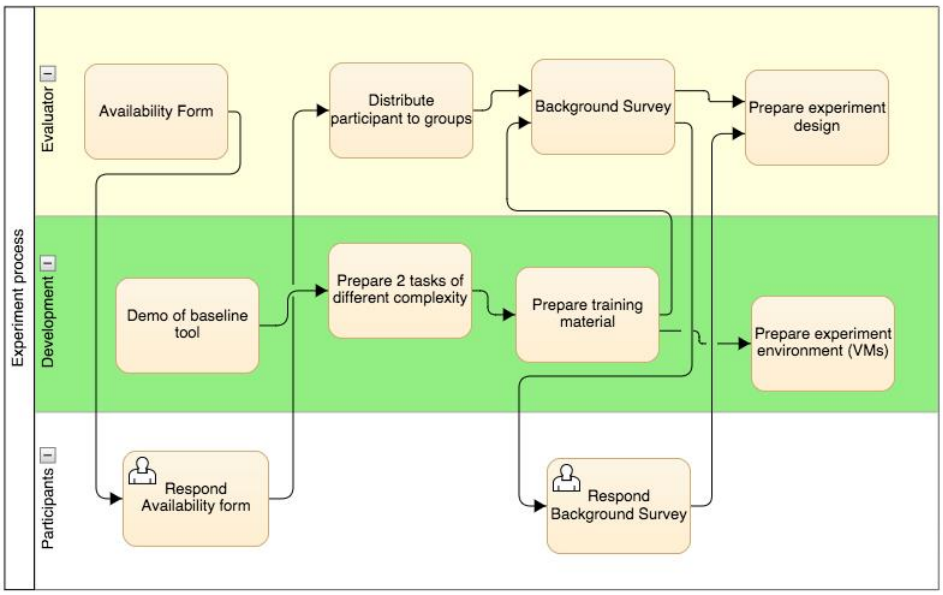

Fig. 1. Experiment preparation

claimed for the DSE Merge tool is its power to support domain experts in same process without requiring from these experts a high level of programming expertise. DSE Merge is claimed to empower incorporation of domain-specific knowledge explicitly into merge process. However, these two benefits can only be evaluated afterwards. This experiment was scoped to the similar context as alternative supports, to confirm its benefits in familiar context described as follows:

- *User Profile* - target users for this experiment are expected to be software engineers
- Technology - all three tools are ruining over Eclipse IDE. OS during evaluation was Windows 7 on Desktop computer (Intel(R) Core(TM) i5 650@3.2GHz, 8 GB RAM, 19") or Lenovo Thinkpad T61p laptop (Intel T7700@2.4GHz, 4GB RAM, 15.4").
- *Social and Physical environment* - the environment in which the tool is expected to be used reflects the typical office environment, where the user is working individually by the desk using laptop or desktop computer. Interaction is performed by use of the mouse, keyboard and the monitor.
- *Domain* - the domain meta-model that was chosen for the experiment reflected the Wind Turbine domain problem.
- *Workflow* - due to existence of the 2 different versions representing the same instance model, the user need to find the best merge solution. The problem is more complex depending on the number of the conflicts between the models. The task T0, described in the DSE-MergeWT.pdf [7] was taken as representative to problem reasoning based on domain example.

D. Training materials

Teaching session was expected to start with an *Introduction Session* to the Wind Turbine meta-model and model merge problem, that was followed with a practical reminder on basic functionality of Eclipse modeling environment. During *Introduction Session* the participants are allowed to ask questions. This session was supported by:

- Wind Turbine Control System printed document containing meta-model.
- EMF-models demo video describing use of eclipse and model merge problem.

The *Introduction Session* was followed by the *Tool Session* during which participants were not allowed to ask any question until the session is finished, for each evaluated tool. The produced materials for all three tools, DSE Merge, Diff Merge and EMF Compare were the following:

- Demo video describing the use of the tool trough presentation of the task T0 that was defined in experimental workflow context.
- Printed document containing explanations and screen shots presented in the demo video.

During the pilot session the participants were asked to give the feedback about training directly in the printed materials. The training materials were improved and can be found in folder Teaching [7]. Time was estimated to be 10 minutes for *Introduction Session*, while 5 minutes for *Tool Sessions*.

E. Experiment instruments and measurements

The experiment instruments and measurement factors are presented in Table 1.

The data for calculating the *Profile* factor was collected trough Availability and Background questionnaire. The *Profile* is influenced by following Experience factors:

- education + programming
- modelling
- EMF Compare tool

- Diff Merge tool
- DSE Merge tool
- Wind Turbine meta-model

For each Experience factor participant rated themselves by 5 point Likert scale and justify their answer by open end question. The final *Profile* score, scaling from 0-5 was calculated as average of all six Experience factors, to which it was added the value of 1 in a case that person had relevant Industry experience. In other case the person was assumed to be Academic.

The *Time* reflects the actual time taken to solve the tasks and was captured trough video analysis.

TABLE I
INSTRUMENTS AND SCALES

	Instrument	*Value*
Profile	Availability Form, Background Questionnaire	[0-5]
Time	Video recording	mm:ss
Success	Eclipse project delivery	[0-1]
Cognitive Effort	NASA TLX Scale	[0-1]
Satisfaction	Satisfaction Questionnaire	[(-1)-1]
Preference	Feedback Questionnaire	0 or 1

Success Factor is defined by following values

- 1 - if the project reflect set of correct solution and is delivered with success
- 0.5 - project delivered but is not reflecting the set of correct solutions
- 0 - no project delivery.

Quality Factor is described in following section, as it is defined specifically for each Task. The *Success* reflects the is multiplication of the Success Factor and Quality Factor.

The *Cognitive Effort* reflects the participants workload during solving task and is measured by a NASA TLX Scale [5].

The *Satisfaction* scale is reflecting average values in range (-1) strongly disagree till (1) strongly agree on a 5-point Likert scale regarding following factors:

- Easy to Use
- Confidence
- Readability and Understandability of User Interface
- Expressiveness
- Suitability for complex problems
- Learnability

The *Preference* is a factor reflecting explicit preference (marked 1) toward one of the tools used based on subset of Satisfaction criteria, that is annulled if in conflict with same factor collected using Satisfaction Questionnaire.

All defined instruments were used during the pilot session, after which trough interview the evaluator collected the suggestions and doubts regarding the surveys developed for the purpose of the experiment, and can be found in folder Instruments [7].

F. Tasks

The representative tasks, of different level of complexity (see Table II), were defined and analysed to be used during experiment execution and are documented in a Task folder [7].

TABLE II
TASK COMPLEXITY

Task	*Model Size*	*Change Size*	*Solutions*
T1	Small	4	2
T2	Small	12	8
T3	Big	6	2
T4	Big	54	¿mil

Quality Factor is defined for each task separately.

- Task 1.
  - 1 - One of the two possible solution is delivered
  - 0.75 - Only one of the two conflict resolved well.
  - 0.5 - None of the two conflicts are resolved correctly.
  - 0.25 - Other part of the model is modified.
- Task 2.
  - 1 - One of the 8 possible solution is delivered
  - 0.75 - Conflicts are resolved, but non-conflicting changes are missing.
  - 0.5 - Conflicts are not resolved, but non-conflicting changes are applied.
  - 0.25 - Other part of the model is modified.
- Task 3.
  - 1 - One of the two possible solution is delivered
  - 0.75 - Only one of the two conflict resolved well.
  - 0.5 - None of the two conflicts are resolved correctly.
  - 0.25 - Other part of the model is modified.
- Task 4.
  - 1 - At least 10 local and 10 remote changes are applied
  - 0.75 - At least 5 local and 5 remote changes are applied.
  - 0.5 - Only local or only remote changes are applied.
  - 0.25 - Other part of the model is modified.

The pilot session showed that cognitive effort is similar for each task (see Table III), probably due to impact of learning trough previous problem participants were able to solve more complex task by having similar workload. Avg time was ranging between 3-5min, while success rate was high and was a bit lower for more complex tasks.

TABLE III
TASK PILOT VALIDATION

Task	*Cognitive Effort*	*Time*	*Success*
T1	25.83	3:32	1
T2	28.61	4:59	1
T3	20.55	3:18	0.88
T4	24.02	4:27	0.83

G. Experiment Flow

The experiment took place on 11th December at the Budapest University of Technology and Economics. The general experimental process is presented in Fig.2, starting by Learning session, during which the subjects filled the Background questionnaire. After this they continue by to solve the exercises

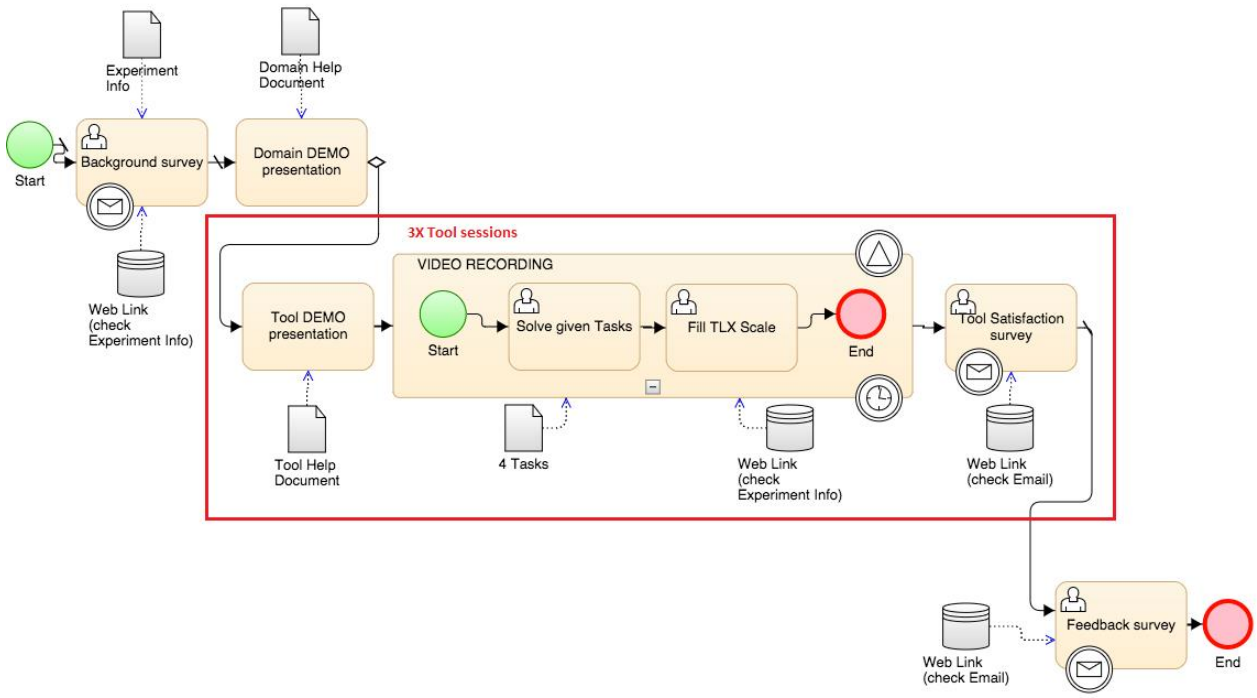

Fig. 2. Experiment treatments

during Task session, that was video recorded. Finally, during Feedback session participants filled final questionnaire rating tools that they have used. The Figure 2 except reflecting flow of activities during the experiment, explicitly shows documents and treatments that were provided to participants, as well as the instruments that were used to collect the data.

During Learning sessions the subjects learned about domain and tool. Subjects were invited to ask questions and to ask for help only after presentation. During the Task session the subjects were not limited with time to solve this tasks. They were not allowed to ask for help, except if they experienced some technical or connection problem.

During the pilot session the cognitive effort for each task was estimated to be similar, the TLX scale is decided to be used just once for each tool that is being evaluated, in the end of Video Session. Based on obtained results and opinions of the participants during Pilot Session, it was found that Diff Merge is rated as more competitive alternative for DSE Merge. The experimental groups were divided in two (G1, G2), G1 receiving first *Tool Session* for Diff Merge and then DSE Merge, while G2 opposite sequence. Finally the EMF Compare *Tool Session* was lefted to be final and was evaluated just by G1.

III. Main Results

In this section we present obtained results of the experiment.

A. Subjects background

In the Table IV we present the number of subjects and obtained Profile score and industry experience. There was total

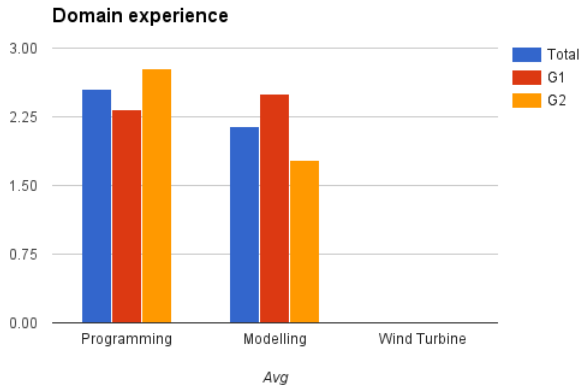

Fig. 3. Domain Experience

of 15 participants with. Among them around half were industry and other half academics.

TABLE IV
Subject Background

	Total	*G1*	*G2*
Number of participants	15	6	9
Profile	1.65	1.92	1.39
Industry	56%	67%	44%

We can see the Experience score in Figure 3 and 4. The majority of participants had the high experience in programing and modeling. No one had experience with Wind Turbine

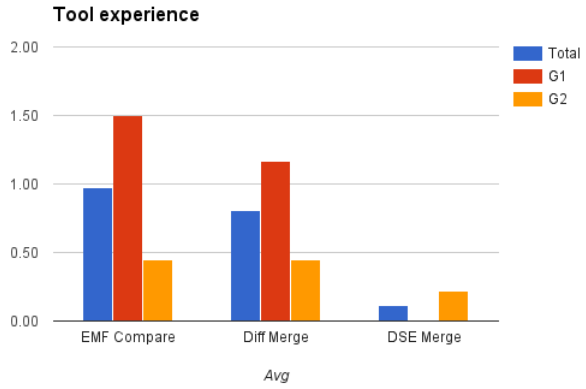

Fig. 4. Tool Experience

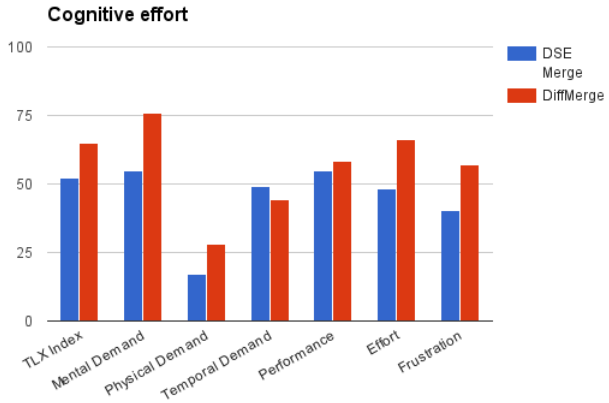

Fig. 5. Cognitive Effort

domain. Some participant had previous experience with alternative tools, while just one participant had a little knowledge about DSE Merge.

B. Comparative results

Table V presents obtained results for all three tools, for Group 1 (G1). We can see that the results confirmed that EMF Compare is candidate indicating lowest score, even as it was evaluated last, when subjects were already having high understanding of the merge process, domain and tasks and they had relevant previous experience with this tool (see Figure 4).

TABLE V
G1 RESULTS

	DSE Merge	*Diff Merge*	*EMF Compare*
Experience	0	1.17	1.5
Time	12:14	22:50	17:06
Success	0.88	0.97	0.63
Preference	5/6	1/6	0/6
TLX Index	46.65	62.83	84.93
Satisfaction	0.33	0.14	-0.36

Due to fact that Diff Merge was the object of first Task Session, while the DSE Merge of the second session, we can observe that there was a much longer time necessary to execute the tasks with Diff Merge. Success rate is higher for Diff Merge, but we can also observe that the participants did have a relative experience with this tool. On other hand they present lower cognitive effort, higher satisfaction rating and explicitly preference toward DSE Merge.

In regard to both groups, we present comparative results for DSE Merge and Diff Merge in Table VI. In total DSE Merge scored with lower time indicating a slightly better efficiency. Also, DSE Merge indicated slightly higher success rate and explicit preference by 11/15 participants, which contributes to possibility of accepting hypothesis H2. However, we could observed that there was a tendency to give the preference to the same, although it was not justify by ratings given during Tool Satisfaction Survey. This preferences were not considered. Also there were subjects that were indifferent and did not express the significant preference based on the ratings described before.

TABLE VI
DSE MERGE V.S. DIFF MERGE

	DSE Merge	*Diff Merge*
Experience	0.11	0.8
Time	20:19	23:02
Success	0.92	0.85
Preference	11	1

Concerning cognitive effort (see Figure 5), in total subjects rated with higher workload for Diff Merge regarding all factors, observing significantly higher Mental Demand and Frustration in comparison to which they experienced with DSE Merge.

We analyze more in detail the Satisfaction rating based on predefined factors in Figure 6. DSE Merge scored very high regarding easiness of use, expressiveness and learnability. Confidence was positive and better than with Diff Merge, while suitability to solve the given tasks even rated negatively for Diff Merge. User Interface, namely its readability and understandability, seems to be most important factor to be improved in order to provide better usability of the DSE MErge.

C. Threats to validity

The results presented are good indicator that DSE Merge is good enough, in regard to its purpose for people with high programming and modeling expertise. However, as it is meant to be used by the domain experts, that often are not advanced in programming, it will be necessary to evaluate it with more novice programmers, and preferably with real domain experts domain experts from a few domains, to validate the target scope of its use. Another threat was that the subjects were mostly in some way related to projects developed by the same team, which could influence their preference and satisfaction scores a bit toward DSE Merge.

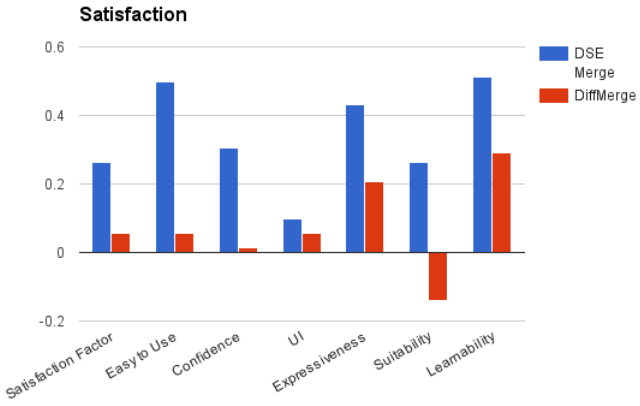

Fig. 6. Satisfaction

D. Conclusion

The most valuable contribution that resulted from this STSM visit is the experiment design, instrumentation and metrics that we believe can be easily repeated and reused for similar evaluations of new techniques for multi-paradigm modeling of cyber-physical systems. This experiment design takes deeper analysis of subject profiles, technology, social and physical environment and targeted workflow scenarios, that are defined explicitly and incorporated in a data collection instruments and reflected in hypothesis.

IV. Future Collaboration

In order to obtain significant data to confirm the experiment objective, the plan is to continue a collaboration with following goals:

1) Run provided experimental design over virtual machines, that are to be created at virtual portal of Budapest University. The undergraduate students and other possible participants will be invited to participate in experiment over portal, until we collect enough data to make a statistically relevant report. For this purpose the metrics will be standardized and calculated automatically for provided instruments. On other hand, the quality of demo videos should be improved and supported by textual explanation. Time for solving the task is to be captured trough the eclipse plug in, and experiment flow is to be preserved by use of some e-learning techniques.

2) Reusing the provided design in assessment for different tool, or the improved version of same tool with different evaluation objectives or subject profiles. This can help us to identify reusable parts, and provide scripts that can help in automatizing result analysis.

V. Foreseen publications

We plan to publish results after running the planed experiment over virtual machines in February, 2016. Target venues that we were considering are MODELS[4] and ASE[1].

References

[1] Automaed Software Engineering conference 2016, http://ase16.org/, last accessed in 11 January 2016.

[2] EMF compare, https://www.eclipse.org/emf/compare, last accessed in 11 January 2016.

[3] EMF Diff/Merge, http://eclipse.org/diffmerge/, last accessed in 11 January 2016.

[4] Model Driven Languages and Systems conference 2016, http://models2016.irisa.fr/, last accessed in 11 January 2016.

[5] NASA-TLX - Task Load Index, https://en.wikipedia.org/wiki/nasa-tlx, last accessed in 11 January 2016.

[6] Ankica Barišić, Vasco Amaral, Miguel Goulão, and Ademar Aguiar. Introducing usability concerns early in the dsl development cycle: Flowsl experience report. In *MD^2P^2 2014–Model-Driven Development Processes and Practices Workshop Proceedings*, page 8, 2014.

[7] Ankica Barišić. Dse merge stsm experiment - https://goo.gl/Kq3R1G, last accessed in 11 January 2016.

[8] Ankica Barišić, Vasco Amaral, Miguel Goulão, and Bruno Barroca. Quality in use of domain-specific languages: a case study. In *Proceedings of the 3rd ACM SIGPLAN workshop on Evaluation and usability of programming languages and tools*, PLATEAU '11, pages 65–72, 2011.

[9] Victor R. Basili. The role of controlled experiments in software engineering research. In Victor R. Basili, Dieter Rombach, Kurt Schneider, Barbara Kitchenham, Dietmar Pfahl, and Richard Selby, editors, *Empirical Software Engineering Issues. Critical Assessment and Future Directions*, LNCS, pages 33–37. Springer Berlin / Heidelberg, 2007.

[10] Abel Hegedus, Akos Horváth, István Ráth, and Dániel Varró. A model-driven framework for guided design space exploration. In *Proceedings of the 2011 26th IEEE/ACM International Conference on Automated Software Engineering*, pages 173–182. IEEE Computer Society, 2011.

[11] Marouane Kessentini, Wafa Werda, Philip Langer, and Manuel Wimmer. Search-based model merging. In *Proceedings of the 15th annual conference on Genetic and evolutionary computation*, pages 1453–1460. ACM, 2013.

[12] Pedro Leonardo. Child programming : An adequate domain specific language for programming specific robots. *MSc dissertation, Universidade Nova de Lisboa*, 2013.

[13] Aaron Marcus. The ROI of usability. In Bias and Mayhew, editors, *Cost-Justifying Usability*. North- Holland: Elsevier, 2004.

[14] MONDO. Scalable modelling and model management on the cloud, project.

[15] Jakob Nielson. *Usability Engineering*. AP Professional, 1993.