

Supporting the Reflective Learning of Conceptual Modeling Using Interactive Timelines and Replays

Marcel Schmittchen
Ruhr-University Bochum, Germany
marcel.schmittchen@ruhr-uni-bochum.de

Abstract

It is not a trivial task to teach “good” conceptual modeling in a structured way. One problem we encountered in our previous exploratory studies is the predominant reflection of the quality of finished models and less of the modeling process itself. As a result, many phenomena like the emergence of errors or the coordination and collaboration phases in group modeling usually get out of focus and are subsequently not thoroughly considered in teaching. Thus, we have developed a tool-supported timeline method, which allows us to review and discuss the collaborative development process of models after they have been finished or submitted. We evaluated the method and employed a variety of tool features in our regular undergrad courses. Considering the perspective and experiences of teachers and learners alike, we discovered that teachers can use our method to better analyze strategies and made mistakes when enough time to prepare the reflection, even if this increases the time distance to the event it is reflected on. Learners were motivated to share and reflect on their actions. Furthermore, we identified further steps to facilitate our reflective modeling teaching method.

1. Introduction

The use of conceptual models is an integral part of the software engineering process [1]. In addition, it helps to define and communicate complex ideas and systems [2, 3]. Novel applications and concepts always attempt to improve higher education methods in software engineering education, and the need for good supporting software tools and computer-supported collaborative learning (CSCL) benefits are already established [4].

Over the last years, we conducted exploratory studies to test different group-focused methods, hardware, and tools in our software engineering and IT-project management undergrad courses [5]. The taught notations are predominantly Unified Modeling

Language-based techniques [6, 7, 8] like use case diagrams or activity diagrams but also include notations from adjacent domains like entity-relationship diagrams, business process modeling [9], schedule modeling (e.g., precedence diagrams) or sociotechnical process modeling [10]. These studies allowed us to identify three critical requirement areas for improving collaborative group modeling exercises: Easy point of entry, coordination support, and evaluation support [11]. In their literature review on collaborative modeling, Renger [12] assessed that much research is focused on the quality and complexity of the resulting model and less on the participants’ reasoning about their modeling decisions and shared understanding of the resulting model.

However, conflicts, compromises and drafts might be vital to reflect problems and learnings in coordination and collaboration phases. For a necessary reflection of modeling process issues such as subsequently fixed errors need to be preserved.

The SARS-CoV-2 pandemic of 2020 forced many learning settings to switch to online and remote formats, which imposed additional constraints in evaluating and supporting group work tasks. Thus, we have focused our implementation and studies on our evaluation support method “timeline”.

The timeline’s basic idea is to preserve the model’s creation process by logging snapshots of the various development stages and presenting them along a timeline. This feature allows teachers and learners to browse through a modeling process later. Providing intuitive comparison views of these stages should help to trigger reflection on errors or decision points. Like sportscasts’ slow-motion replays and strategic analysis views, this type of presentation and method should help identify how mistakes and conflicts arose and were later solved. This method can also help to encourage the modelers to reflect on choices they have made and the shared understanding of their final models.

Farah and Lethbridge [13] already implemented and evaluated a similar approach with a temporal exploration of changes made to a software (design) over

time and whether it could improve the ability to reengineer and maintain it quickly. We focused our approach on conceptual modeling and the learning thereof through reflective learning. The main research questions for this paper were: “How can a timeline-based method for encouraging reflection on collaborative modeling be implemented? How well does it work in real educational settings, and what are limitations and aspects for improvement?”

We will discuss related work concerning reflective learning and conceptual modeling learning research in the following section. Afterward, we present our developed method and prototype, followed by our two study designs for synchronous group work and asynchronous homework settings.

2. Related work

In the following, we review the different approaches and applications for computer-supported reflection (Section 2.1) and conceptual modeling (Section 2.2)

2.1. Computer-supported reflection

Huge benefits from reflective learning are researched and identified in different areas including but not limited to: computer-supported collaborative work (CSCW) in co-located teams [14, 15, 16, 17]; as well as in asynchronous online communities [18, 19]; design thinking and designing of business models [20, 21]; in computer-supported collaborative learning (CSCL) [22, 23] and (teacher) education [24, 25, 26]. These various approaches and concepts are unified in their shared insights on the importance of reflective learning and the challenge to create effective facilitation for each domain’s unique requirements for reflection support.

Based on core concepts by Schön [27], Lynch and Metcalfe [28] distinguish two types of reflection processes: Reflection in action happens when action and reflection happen almost simultaneously. The user draws knowledge from experience to react intuitively, while reflection on action covers a retrospective consideration of their decisions and consequences. The latter can and should be supported in coached teaching as much and often as possible. In addition, reflection can only arise from subjective experiences and not from objective content and should therefore be practiced repeatedly [15, 28]. In order to increase the value of reflection, it should also move away from an introspective endeavor [15] and promote exchange in (peer) groups whenever possible. Knipfer et al. [15] “consider team reflection to be an iterative cycle of individual and joint reflective activity, both of which are closely intertwined. Furthermore, (preliminary)

outcomes are fed back into the reflection process, thereby enhancing and enriching an individual’s and a team’s understanding of an experience.”

Therefore, peer observation and feedback methods play an important role that teachers and learners should be familiarized with early on [25].

2.2. Teaching conceptual modeling

Bork [29] created a framework for teaching conceptual modeling in which they defined cognitive process and knowledge dimensions for (meta-) modeling tasks and mapped these on several courses. However, they found that their own tasks lacked especially in the metacognitive knowledge dimension (strategical thinking, self-knowledge, reflection on experiences) and in evaluation tasks (checking and critiquing other models) [29] MacCreery and Tenbergen [30] addressed the difficulty that conceptual modeling is often seen as a by-product in software engineering courses and attempted to define quantifiable data on conceptual modeling quality. They evaluated the quality of student models in three dimensions: syntactic (e.g., are correct notational elements used), semantic (e.g., do models correspond with the semantic domain of the notation), and pragmatic (e.g., is the correct notation used for the purpose). While their early modeling phases contained many errors in the first two dimensions, the pragmatic quality was consistently high showing that a shared understanding and communication was possible even when the first two dimensions were faulty [30]. The initial motivation of students to learn conceptual modeling is often low, especially if they are already experienced in programming. However, all students use the diagram sooner or later as a tool to communicate with others [30]. This suggests a need for more reflective learning approaches to promote (self-) evaluation and communicating skills and practice them as early and often as possible.

Other concepts to improve conceptual modeling teaching include inverted classroom concepts that encourage students to be more independent and that allows for more time for other activities [31]. These should foremost be rooted in practical examples close to real-world problems [32] and combined with continuous feedback [33].

3. Timeline concept and tool features

Although previous works by other researchers studied the quality of conceptual modeling and their education techniques, they are focused on the results and finished models. E.g. MacCreery and Tenbergen [30] found not many syntactic errors at the end of their studied courses, but they observed that the modeling

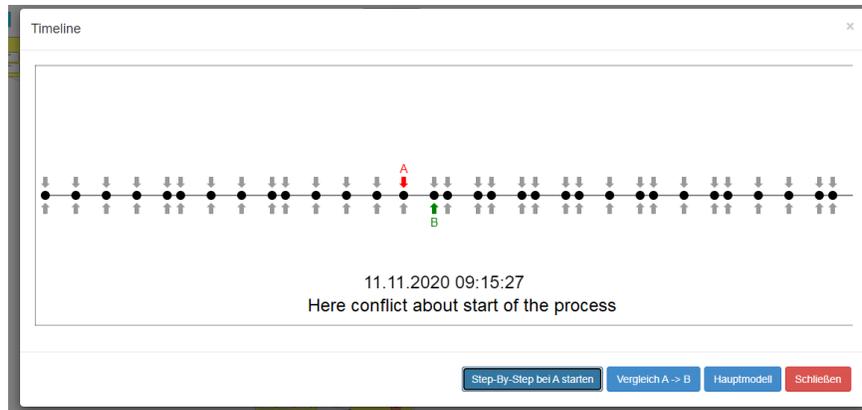


Figure 1. Timeline control window with commented snapshot selection

process itself was riddled with difficulties. Therefore, we want to make these processes and struggles more visible and, above all, more reflective.

While building on the experiences of our previous exploratory studies [5, 11], we also tried to incorporate the primary design considerations to scaffold learner reflection by Lin et al. [34]: To support the teachers to organize reflective activities the timeline method provides options for teachers to manage and prepare reflection sessions. To offer opportunities for learners to compare their results the timeline feature uses a comparison notation and function to compare models from different implementation phases. Also, conceptual modeling is a sufficiently complex task when used in tasks that mirror real-world tasks and case studies. A simple use case example for the timeline method is as follows: A group or individual creates a conceptual model. Using the timeline function the modeler(s) or a coach/teacher is able to see essential changes made to the model on a timeline. If desired, they can manage the snapshots on the timeline and use them to view a comparison between different versions of the model, highlighting the changes. This is intended to be used by groups, individuals, or in educational settings to reflect on the modeling and design process.

3.1. Prototype implementation

The first timeline concept prototype features were implemented as a plugin for an already existing web-based modeling tool with collaborative features. The snapshots are stored on the server-side. Depending on the selected replay mode, two snapshots are compared, and the derived comparison model is displayed (see fig 2). One challenge was to find a style for the comparison notation that makes the changes between snapshots intuitively and easily comprehensible without compromising the semantics of the model's notation.

Through iterative pretests, we arrived at a comparison notation rule set (see table 1).

Table 1. Comparison notation rules

Change (from A to B)	Display-Rule
Element was added	A green frame around the element is displayed
Element was deleted	A red frame around the element is displayed
Relation was added	The relation is displayed green
Relation was deleted	The relation is displayed red
Element was re-embedded	The element is displayed at the previous position with a red frame and at the new position with a green frame
Text change	The former text is displayed in red the new text in green. (Due to technical constraints implemented with brackets instead of color)

Another challenge was to determine which actions were “snapshot-worthy” with respect to their relevance for reflection. During first pretests, the many interactions of joint modeling quickly overloaded the servers' capacities (with well over 1000 actions in 15 minutes). Secondly, the timeline was overcrowded with markers that were impossible to distinguish or select properly. Therefore, we explored a balance between essential actions needed to track relevant changes in the model and a manageable number of snapshots.

In the end, we defined the following actions as “snapshot-worthy”: deleting and creating elements and

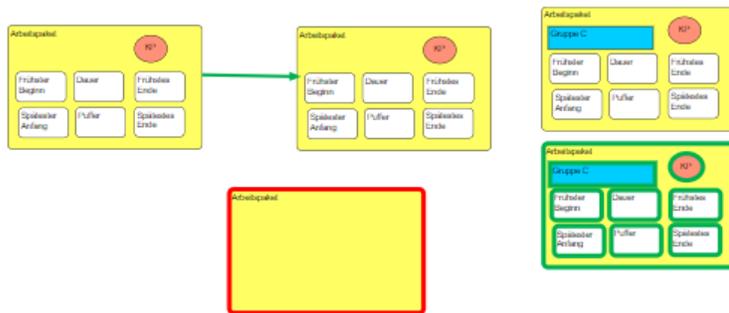
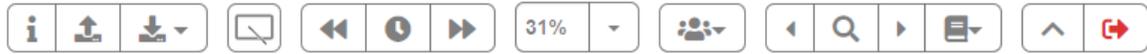


Figure 2. Comparison view of two model snapshots

relations, changing the label of an element, and re-embedding an element (an essential part of the used modeling tool). In addition, a timer is used to collect all changes made in a five-second interval and creating one snapshot collecting all changes made in this time per snapshot. This way, there is always a gap of at least 5 seconds between snapshots on the timeline to still be easily distinguished and selected while keeping all changes retrievable.

3.2. Comparison view modes

There are two ways of viewing replays with the tool: 1) A-B comparison: For this, the user selects any two snapshots from the timeline (marked as A and B, see fig. 1). The tool now creates a view that compares these two snapshots. Using arrow symbols, the B-marker can be moved chronologically back and forth through the timeline without opening the overview.

2) Step-By-Step. In this variant, the user just clicks on the arrow symbols without selecting a marker first. In this view, the comparison is always derived from the current and the previous snapshot. The comparing view is displayed on the main modeling canvas using the comparison rules (see fig. 2).

3.3. Timeline management

The timeline can be accessed via a button in the upper main toolbar. The Timeline shows all saved snapshots of the model as points with their distances on the bar in rough relation to the time intervals of their creation (see fig. 1) By hovering the cursor over one point, its timestamp and comment (if available) are displayed. In order to curate the timeline presentation, it

is possible to delete snapshots from the timeline or label them with a comment.

4. Study design

We conducted an exploratory study with the implemented timeline concept, to find out whether the concept is feasible to initiate and support reflection, and which factors influence the success of reflection in which way. The study was run in two different ways: A *synchronous group work variant* and an *asynchronous homework variant*. The study design focused on the applicability in traditional university teaching settings and on the quality of the feedback made by teachers and the students' understanding of the learning content. in-depth research questions are described in the respective subsections.

4.1. Task and participants

Instead of the usual UML diagrams, the precedence diagram method (PDM) [35] was used as the modeling notation. This method is used to schedule activities or tasks throughout a project. Activities are mapped with their dependencies and durations on paths for which starting/ending dates and slack are calculated. The critical path method (CPM) can then be used to identify parts of projects which are especially prone to create delays. This notation was used because its implementation has an unambiguous correct solution and thus differences in the creation process of the model or errors made in the calculation of values can be more easily identified and tracked.

We adapted an exercise that we have already used in previous studies [5] for a co-located group setting for synchronous group work. In this exercise the group of

nine learners is separated into three subgroups of three learners. Each subgroup takes on the role of a fictional department and receives an individual task sheet consisting of five work packages with their dependencies and workloads. Now each group models the precedence diagram of their department (containing 5 elements). Finally, they have to calculate and add the necessary values to their model and identify the critical path.

Once all three groups have completed their task, they come together again. Another task sheet is given to the group containing new dependencies intersecting the departments' work packages. The group is asked to merge their diagrams from the first task into a single and complete precedence diagram (containing 15-17 elements). Hereby, the positions and relations of individual elements change. This also changes many of the previously calculated values which must be adjusted by the learners in order to find the common critical path.

For the asynchronous homework variant, the same task was used with a small variation: Since the homework is done alone and not in a group, the two tasks were combined, i.e., the learners received a task sheet with all work packages and all intersecting dependencies for all three fictional departments right away. Only the final model was to be created.

The learners were computer science students recruited from our undergrad software-engineering and project management courses. They were treated with an inverted classroom learning unit consisting of a learning video and an interactive presentation slide about a week before the study to familiarize themselves with the notation and calculation rules of the elements. At the time of the study, the learners had not yet modeled precedence diagrams themselves. The participation was voluntary and not graded. Participants received compensation in form of a fixed sum of extra points for their courses upon completing the study regardless of results in the study.

The teachers were research assistants from our department with at least two years of experience in teaching PDM in said undergrad courses.

Due to health concerns during the SARS-CoV-2 pandemic in 2020/2021, all studies took place as remote online meetings. learners and teachers communicated via a video meeting tool. The synchronous exercise was performed two times with a total of 18 individual learners. The asynchronous variant was performed four times with a total of four individual students. No student was allowed to partake in both variants. Three individual teachers including the author conducted all studies.

The structure of the learning exercises and the qualitative evaluation methods of both variants are described below.

4.2. Synchronous group work variant

The research questions for this variant were: "How can the teacher(s) facilitate a reflection phase with this method?" and "How do learners benefit from the reflection phase?"

For this, in contrast to the earlier versions of this task, we extended this exercise now with a reflection phase instead of a third group task. In this reflection phase, the group should discuss difficult situations or possible mistakes made during the tasks. The discussion should be supported and moderated by an experienced (modeling) teacher to stimulate the students' reflection and lead the group to think about ways to improve their modeling as well as their collaboration. This can be supported by showing the learners the corresponding moments from the timeline presentation including the highlighted changes made between selected snapshots. The learners are also encouraged to browse through the timeline to showcase or find specific snapshots they want to talk about.

To facilitate this reflection phase, two teachers with distributed roles (*group-teacher*, *reflection-teacher*) were active in our study. The *group-teacher* was responsible for the tasks, i.e., explaining the exercise, handing out the task sheets, and answering the participant's questions. During the two task phases, it was the job of the *reflection-teacher* to observe the group processes and to note situations that might be important for subsequent discussions based on the timeline representation. In the reflection phase, the *reflection-teacher* takes over the moderation of the group and uses his notes and marked snapshots in the timeline to stimulate the group reflection. During both tasks, the *group-teacher* can also note observed situations or errors and can either mark them with the timeline tool or send a short message to the *reflection-teacher* via a backchannel (e.g. an internal chat) so that they can mark it for them.

In order to evaluate the method, the learners were then interviewed about the tasks and the individual processing phases. After the learners were dismissed, the two teachers were interviewed about the opportunities and challenges in applying the method and using the tool.

4.3. Asynchronous homework variant

The research questions for this variant were: "Can the teacher(s) make correct assumptions on the model creation process?" and "How does the learner perceive the feedback and assumptions?"

The learners received the task and an individual link to the web-based modeling tool via e-mail. In addition, they were informed that they had five days to complete

the task and would then receive written feedback and a questionnaire.

After the deadline, two teachers met in an online video meeting to review the participant's submission. In the first step, they reviewed the final model and wrote a short feedback paragraph. On average, this review found one or two errors, mainly a mislabeled element or a wrong value in the calculated values.

After this fairly basic homework review, the teachers switched to the timeline tool. Now, the teachers went through the steps of the model's creation discussing and marking noteworthy moments. Special attention was paid to two aspects: a) the general strategy used to approach the modeling and calculation tasks by the participant and b) finding the moment when the mistakes found in the first review were made and what might have led to it. From these notes, the teachers then generated a second feedback paragraph that included tips on avoiding these mistakes in the future.

The two variants of feedback, as well as excerpts from the teachers' notes, were included in the questionnaire sent to the learners. The questionnaire used 5-point Likert items and free text fields. In addition to questions about prior knowledge of the modeling method, understanding of the task, and problems with the web tool, the learners were asked to evaluate the quality of the feedback. First, the learners were asked to describe whether they had made any mistakes in the course of the homework that they noticed later and how they had proceeded. This was followed by the feedback written by the teachers, which were rated in terms of their comprehensibility and usefulness. Afterward, the learners were confronted with the notes of the teachers and asked to reflect and comment on the assumptions made by the teachers. In the end, they were asked to rate how accurate the assumptions about their general strategy and made errors were. In this method, the teachers were observed conducting the review and then interviewed.

5. Results and observations

In the following subsections, results from our study variants are presented

5.1. Results and observations of the synchronous variant

In two cases, a significant coordination problem occurred during the merging phases: two or more people worked on the same part of the model, resulting in duplicate elements without being aware of this overlapping of their activities. The group noticed this problem later and solved it by coordinating their actions

from then on out. Other special situations were limited to the calculation of the values to be entered into the diagram. At this stage, all groups had started to discuss a clear division of tasks. Learners who were not involved in the calculation intervened whenever their group colleagues made a mistake.

Although these situations led to exciting discussions about group work and coordination processes, the teachers could not find or present suitable situations or snapshots on the timeline tool for the addressed cases in time. Especially the A-B comparison view was used significantly less than the step-by-step view (see 3.2). The teachers stated in the evaluation that the workload of the regular exercise supervision and the spontaneous parallel preparation of the reflection phase was greatly underestimated in the planning. In addition, especially the reflection-teachers wished for a longer and better familiarization with the use of the tool and a better feeling for the kind of comparisons the timeline tool generates from the selected snapshots. The step-by-step view was deemed more intuitive under the given conditions. Some learners also indicated that they had not paid attention to the displayed comparisons during the reflection phase. Thus, it seems that the possibility of using the timeline representation for reflection support remained below the projected benefits, and needs to be further improved.

Besides wishes made by the participants in regard to general usability most desired features to improve the method and prototype included a heatmap style differentiation of snapshots with different amount of changes made and the highlighting of unusual gaps that often represent phases where considerations, decisions, or discussions took place.

The learners positively evaluated the synchronous group variant and especially the reflection phase in the interviews. In particular it was remarked that "communication and coordination challenges are often ignored in regular exercises"

5.2. Results and observations of the asynchronous variant

In the asynchronous homework variant, the timeline representation was used more in accordance with the intentions of its implementation. In this specific task, several strategies and their manifestations were observed. One significant difference in strategies was creating all needed elements before arranging them in a sequence vs. modeling the sequence and creating elements "just in time" when they had to be inserted into the diagram. Other differentiating strategies were identified in the order in which the values are calculated (left to right was given by the method, but the further

calculation order was sometimes column-wise, row-wise, or in sequence order).

By searching for errors in the final model beforehand, the teachers' observations naturally focused on finding the origin of the errors. The teachers found possibilities for drafting their feedback to be more responsive and meaningful regarding finding incorrect procedures rather than being limited to simple "right/wrong" assessments. Similar to the synchronous version, time was the biggest hurdle in implementation. The corrections of this minimal modeling task were about 90 minutes per participant/homework, well above the amount of time an application in a natural learning setting would allow. Yet the teachers' noted that it was "fascinating how well individual strategies and mistakes can be seen".

The evaluation of the received feedback, i.e. the correctness of the assumptions made by the teachers, and of the method itself were very positive in the questionnaires (see table 2). All learners gave at least one self-description of a mistake made on their own. The detailed comments on the assumptions of the teachers resulted in an average of 4.5 comments per participant. Of these, only a small portion (11%) were corrections of a wrong assumption, and a high proportion (56%) were an additional reflective comment by the participant. I.e., the participants agreed with the teacher's observation adding their reasons, connected problems, or further considerations (see fig. 3). All in all a much more active level of reflective discussion was observed between the teachers and learners compared to regular homework/review processes.

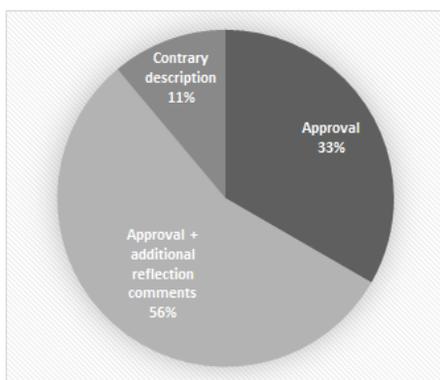


Figure 3. Comment types regarding teacher's assumptions

The learners rated the method itself with an average of 1.5 based on the German grade system (American Letter Grade "A"). Qualitative comments were in favor of the method, noting in particular that it could be used in assigning partial points for solution paths or to make cheating more difficult since the teachers will see when solutions are directly copied into the diagram. One

participant described the accuracy of the assumptions and the idea of the teacher viewing the whole modeling process as "creepy," expressing the fear that this form of reviewing could lead to insecurities, for example, when learners with unconventional strategies would be punished even when the final solution is correct.

Table 2. Results from the asynchronous variant questionnaire

Item	Median	Mean
Comprehensibility of feedback (1= Very well comprehensible, 5= Not at all comprehensible)	1	1.25
Correctness of the assumptions (1= Very correct, 5= Not at all correct)	1	1.25
Rating of a modeling homework with reflective feedback in school grades (1=A, 5=F)	1.5	1.5
Rating of analyzing the homework's creation process by the teacher in school Grades (1= A, 5=F)	1	1.5

5.3. Limitations and Discussion

Unfortunately, the synchronous execution suffered some technical problems. Especially in terms of usability and stability of the shared web-based modeling software, there were some delays and problems with the prototype.

Furthermore, we were only able to recruit a small number of participants (n=22) for our studies in the last semester. Since we needed nine learners and two teachers to implement one synchronous group work variant, only one internal pilot test and two runs with this variant were carried out. Interested students for whom a full group size was not achieved participated in the asynchronous homework variant. This small sample size prevents a meaningful quantitative evaluation. Nevertheless, the qualitative observations and interviews with learners and teachers in both study variants were very informative and served as a basis for further concept ideas and studies.

Even with our limitations, the comparison of the two variants shows that the timeline method in its current form can support a *reflection on action*. Strategies, errors in the modeling process and causes of errors were successfully identified by the teachers and translated into feedback that was rated as helpful by learners. On the other hand, learners were also very often able to reflect on their reasons, decisions, and mistakes in answering the teachers' assumptions.

For *reflection in action*, the current form of the method and tool is not yet sufficiently developed. While teachers and learners used the Step-By-Step comparison view intuitively, the A-B comparison view lacked clarity and thus the opportunity for meaningful use in action. Especially, the reflection-teachers would have needed the time to select and decide on certain snapshots from the timeline tool. The learners should also get a chance to explore their process independently from the teacher with the timeline presentation before the reflection phase starts. Because of the continuous recording of snapshots, a larger time gap between action and reflection [28] should be possible without problems.

The biggest hurdle of our method in both variants was the unsuitable ratio of time expenditure and results. While the reflection results are quite promising, they are of little use in teaching if they cannot be applied in normal teaching settings and timeframes. In the synchronous variant, the biggest error was also due to our inadequate planning, with both learners and teachers attest that a break between the processing of the task and the reflection would have helped both groups better prepare for the reflection phase.

6. Conclusion and Outlook

We created a method and a prototype that can be used to view and compare the iterative work on a broad range of modeling notations. We suggested two variants of application in an educational setting and showed how it can support the reflective learning of groups and individual learners. All in all, it turns out that the success depends on the socio-technical way of how the features of the timeline prototype are intertwined with the method of how the initiation and support of reflection as well as the interplay of the involved roles are organized.

An encountered problem is the experience required by the teacher to moderate the reflection process and to identify reflection-worthy moments in the group work. Reflections follow paths even if these are not yet clearly identified [19]; thus, further research should be done to identify the domain-specific reflection paths of conceptual modeling.

One of the clearest benefits of our timeline method was the high quality of the assumptions made by the teachers about the learners' behavior in the asynchronous homework variant. Furthermore, the comprehensiveness of the learners' written reflections appears as successful. This verbalization of thoughts on the modeling process and one's own mistakes was significantly better than expected by teachers compared to typical homework tasks.

Based on our studies, another touch-optimized prototype is already being developed, which, in addition to the comparing views and snapshots, will provide a

function for adding additional annotations to snapshots in a dialog-based mode. With this feature, we want to integrate the text-based reflection from the questionnaires of the asynchronous version into the timeline presentation itself and allow the learners to use the reflection support for peer feedback sessions.

However, in order to compensate for the most prominent shortcomings in the support provided by our tool, further research is needed to identify which additional features and information the timeline should incorporate in order to allow a better reflection in action. For example, we encountered the wish to add a heatmap to the timeline.

The automated highlighting of potentially more interesting moments in the modeling process would be a primary goal. That of course needs more insight into typical problematic situations or strategies of the modeling process. These in turn are better to achieve with a reflective analysis of the modeling process the timeline method can provide.

If through reflective learning and analysis of strategies and errors it will be possible to (semi-) automate the management and presentation of the timeline (e.g. with the additional help of machine learning), we could significantly lower the resources and time needed for supporting reflective learning of conceptual modeling.

Acknowledgements. The author would like to thank Prof. Dr.-Ing. Thomas Herrmann (Department for Information and Technology Management, Ruhr-University Bochum) for his support and contributions.

7. References

- [1] Tamai, T., "How to teach software modeling", Proceedings of the 27th international conference on Software engineering, Association for Computing Machinery, 609-610, 2005
- [2] Patel, A., M. Sim, and R. Weber, "Stakeholder Experiences with Conceptual Modeling: An Empirical Investigation" pp. 7, 1998
- [3] Ebert, C., „Systematisches Requirements Engineering: Anforderungen ermitteln, dokumentieren, analysieren und verwalten“, dpunkt.verlag, Heidelberg, 2019
- [4] Knutas, A., J. Ikonen, and J. Porras, "Computer-Supported Collaborative Learning" in *Software Engineering Education: A Systematic Mapping Study*, 2019
- [5] Schmittchen, M., and A. Avdullahu, "Evaluating Portable Touch Projectors in the Context of Digital Education", in *Learning and Collaboration Technologies. Designing, Developing and Deploying Learning Experiences*, Springer International Publishing, 169–178, 2020
- [6] OMG, „Unified Modeling Language (UML), Version 2.5.1“, December 2017.
<https://www.omg.org/spec/UML/2.5.1/PDF> Accessed: 6. June 2021, 2017

- [7] Fowler, M., "UML distilled: a brief guide to the standard object modeling language", Addison-Wesley, Boston, 2004
- [8] Booch, G., J. Rumbaugh, and I. Jacobson, "The unified modeling language user guide", Addison-Wesley, Reading Mass, 1999
- [9] Association of Business Process Management Professionals, "BPM CBOK: version 4.0.", 2019
- [10] Herrmann, T., Hoffmann, M., Kunau, G., & Loser, K. U. "A modelling method for the development of groupware applications as socio-technical systems". In *Behaviour & Information Technology*, 23(2), 119-135. (2004)
- [11] Schmittchen, M., A. Avdullahu, and R. Beerermann, "Prototyping a Touch-Optimized Modeling Tool for Co-located and Inverted Classroom Group Modeling Scenarios", in *Learning and Collaboration Technologies* Springer International Publishing (2020)
- [12] Renger, M., G.L. Kolfshoten, and G.J.D. Vreede, "Challenges in collaborative modelling: a literature review and research agenda", in *International Journal of Simulation and Process Modelling* 4(3/4), pp. 248, 2008
- [13] Farah, H., and T.C. Lethbridge, "Temporal Exploration of Software Models: A Tool Feature to Enhance Software Understanding", in *14th Working Conference on Reverse Engineering (WCRE 2007)*, IEEE (2007), 41–49., 2007
- [14] Herrmann, T., Nolte, A., & Prilla, M., "Awareness support for combining individual and collaborative process design in co-located meetings", in *Computer Supported Cooperative Work (CSCW)*, 22(2), 241–270. <https://doi.org/10.1007/s10606-012-9179-x>, 2013
- [15] Knipfer, K., D. Wessel, and U. Cress, "Reflection as a catalyst for organizational learning", in *Studies in continuing education* 35(1), pp. 30–48, 2013
- [16] Krogstie, B.R., M. Prilla, and V. Pammer, "Understanding and Supporting Reflective Learning Processes in the Workplace: The CSRL Model", In D. Hernández-Leo, T. Ley, R. Klamma and A. Harrer, eds., *Scaling up Learning for Sustained Impact*. Springer Berlin Heidelberg, 151–164, 2013
- [17] Balzert, S., P. Fettke, and P. Loos, "A Framework for Reflective Business Process Management", in *45th Hawaii International Conference on System Sciences*, IEEE), 3642–3651, 2012
- [18] Herrmann, T., & Kienle, A. "Context-oriented communication and the design of computer supported discursive learning" in *International Journal of Computer Supported Collaborative Learning*, 3(3), 273–299. 2008
- [19] Prilla, M., O. Blunk, and I.-A. Chounta, "How Does Collaborative Reflection Unfold in Online Communities? An Analysis of Two Data Sets", in *Computer Supported Cooperative Work (CSCW)* 29(6), pp. 697–741, 2020
- [20] Schoormann, T., J. Hofer, and R. Knackstedt, "Software Tools for Supporting Reflection in Design Thinking Projects", in *53rd Hawaii International Conference on System Sciences*, 2020
- [21] Schoormann, T., D. Behrens, and R. Knackstedt, "The Noblest Way to Learn Wisdom is by Reflection: Designing Software Tools for Reflecting Sustainability in Business Models", ICIS, 2018
- [22] Lee, S., "Design and analysis of reflection-supporting tools in computer-supported collaborative learning", in *International Journal of Instructional Technology and Distance Learning* 2(3), pp. 49–56, 2005
- [23] Moore, B., C. Smith, A. Boardman, and A. Ferrell, "Using Video Self-Reflection to Support Collaborative Learning for Students With Learning Disabilities", in *TEACHING Exceptional Children* 53(1), pp. 52–59, 2020
- [24] Falter, M.M., and M.E. Barnes, "The Importance of the 'Comfort Zone' in Preservice Teachers' Evaluation of Video Analysis Sessions as a Tool for Enhanced Reflection", in *Teacher Education Quarterly*, 2020
- [25] Hammersley-Fletcher, L., and P. Orsmond, "Reflecting on reflective practices within peer observation", in *Studies in Higher Education* 30(2), pp. 213–224, 2005
- [26] Leinonen, T., A. Keune, M. Veermans, and T. Toikkanen, "Mobile apps for reflection in learning: A design research in K-12 education: Mobile apps for reflection in learning: a design research", in *British Journal of Educational Technology* 47(1), pp. 184–202, 2016
- [27] Schön, D.A., "The Reflective Practitioner: How Professionals Think in Action", Basic Books, USA, 1983
- [28] Lynch, M., and M. Metcalfe, "Reflection, Pragmatism, Concepts and Intuition", in *Journal of Information Technology Theory and Application (JITTA)*, 7:4, 1-10, 2006
- [29] Bork, D., "A Framework for Teaching Conceptual Modeling and Metamodeling Based on Bloom's Revised Taxonomy of Educational Objectives", in *52rd Hawaii International Conference on System Sciences*, 2019
- [30] MacCreery, J., and B. Tenbergen, "On the Syntactic, Semantic, and Pragmatic Quality of Students' Conceptual Models", in *52rd Hawaii International Conference on System Sciences*, 2019
- [31] Mason, G.S., T.R. Shuman, and K.E. Cook, "Comparing the Effectiveness of an Inverted Classroom to a Traditional Classroom in an Upper-Division Engineering Course", in *IEEE Transactions on Education* 56(4), pp. 430–435, 2013
- [32] Kuhrmann, M., D.M. Fernandez, and J. Munch, "Teaching software process modeling", in *2013 35th International Conference on Software Engineering (ICSE)*, IEEE, 1138–1147.18, 2013
- [33] Unkelos-Shpigel, N., J. Sheidin, and M. Kupfer, "Climb Your Way to the Model: Teaching UML to Software Engineering Students: Teaching Case", In H.A. Proper and J. Stirna, eds., *Advanced Information Systems Engineering Workshops*. Springer International Publishing, Cham, 40–46.29, 2019
- [34] Lin, X., C. Hmelo, C.K. Kinzer, and T.J. Secules, "Designing technology to support reflection", in *Educational Technology Research and Development* 47(3), pp. 43–62.30, 1999
- [35] Wiest, J.D., "Precedence diagramming method: Some unusual characteristics and their implications for project managers", in *Journal of Operations Management* 1(3), pp. 121–130, 1