COLLABORATIVE BUSINESS PROCESS MODELING ON INTERACTIVE TABLETOPS

Döweling, Sebastian, Tahiri, Tarik, Schmidt, Benedikt, SAP Research Darmstadt, Bleichstr. 8, 64283 Darmstadt, Germany, {firstname.lastname}@sap.com

Nolte, Alexander, University of Bochum, Information and Technology Management Group, Universitätsstr. 150, 44780 Bochum, Germany, nolte@iaw.ruhr-uni-bochum.de

Mohammadreza Khalilbeigi, Technische Universität Darmstadt, Telecooperation Group, Hochschulstr. 10, 64289 Darmstadt, Germany, khalilbeigi@tk.informatik.tu-darmstadt.de

Abstract

Recent research has established that successful collaborative modeling requires a more active role of domain experts in the modeling process. Yet, tool support for this endeavor is still lacking. Large touchscreens could serve as a platform for easy-to-use modeling tools that address this problem. However, these devices pose new challenges for interaction design – in particular because of parallel multi-user interaction. In this paper, we present coBPM, a system for collaborative modeling on large touchscreens, and four multi-user interaction techniques that we coherently integrated into it. Using a set of typical modeling tasks, we provide a first evaluation of their effects on working styles, ease of use, teamwork- and output quality. We conclude with a summary of our findings on involving non-expert modelers in actual modeling, and design implications for collaborative modeling tools.

Keywords: Modeling; Collaboration; Multi-touch; Large Displays.
1 Introduction

Visual models are common tools in modern organizations and serve for a variety of purposes, among others, the description, analysis and improvement of business processes. Building the respective models, however, is a complex task that requires the input of multiple stakeholders, as each person participating in a process is only knowledgeable about a small part of it (Rittgen, 2010). To exchange perspectives and reach a common understanding about a process, researchers from various domains propose to draft process models collaboratively. This typically happens in co-located modeling workshops, where participants verbally contribute their knowledge and experts conduct the actual modeling. While this approach is effective for negotiating perspectives, its efficiency is questioned by management and participants alike. Workshops are often time consuming, with long idle-times for participants (Prilla and Nolte, 2010). Also, limiting domain experts to verbal contribution may decrease motivation for active contribution in the workshop and identification with workshop results, and thus reduce buy-in for later process implementation (den Hengst and de Vreede, 2004).

Recent research (Prilla and Nolte, 2012) has shown that domain experts are capable of performing many typical modeling actions in a self-directed manner, e.g. identification of actors, logical sequences, and necessary resources. Yet, tool support for this endeavor is currently very limited: modeling tools are usually built with experts in mind and thus hard to use for users without modeling expertise; also, they are typically built for a single user and do not allow parallel input by multiple users, even if designed for collaborative modeling (Rittgen, 2010). To leverage the full potential of direct model interaction by domain experts, both limitations should be addressed. We argue that large touchscreens provide a great opportunity for this, as they allow intuitive interaction with content on the screen and support effective collaboration. We have thus developed an application for interactive tabletops named coBPM (collaborative Business Process Modeler) that features both direct interaction by non-expert modelers (typically process domain experts) and parallel input by multiple users. This software allows users to sketch and iteratively refine business processes using BPMN2; it currently runs on a Samsung SUR40 tabletop (cf. Figure 1). Notwithstanding the potential for other activities in business process management, our focus has been on model development as a pivotal part thereof. Furthermore, while our approach could also be useful for modeling activities in other contexts, e.g. decision processes, for first insights into the benefits of interactive tabletops for collaborative modeling, we have focused on business processes only.

In the remainder of this paper, we review existing tools for collaborative modeling that support direct input by domain experts, then describe coBPM’s functionality with emphasis on four interaction techniques addressing the challenges of parallel multi-user interaction, and subsequently report the results of a first evaluation of the technique’s effects on working style, ease of use, teamwork and output quality. We conclude with a summary of our findings and design implications for collaborative modeling tools.
2 Related Work

Depending on application domains, concrete approaches to collaborative modeling differ from one another. Nevertheless, there are a number of tasks that all approaches have in common:

- **Data collection** contributing elements of the process (actors, objects, activities, conditions and flows) – typically as a loose collection of artifacts.
- **Model creation** relating elements to each other through clustering and alignment, effectively creating a first draft of the model.
- **Model revision** evaluating and refining the process where necessary.

For our subsequent analysis of existing tools for collaborative modeling, we therefore review – in addition to support for the direct domain expert input and parallel user input – whether these tasks can be realized with the respective tool (we argue that a support for all three phases is beneficial, because it helps to a) minimize transition time between phases and b) does not force participants to familiarize themselves with different visualizations in different tools). In general, there are only a few collaborative modeling tools that allow each participant to directly interact with the model:

The COMA tool (Rittgen, 2010) allows all workshop participants to propose model parts that can then be discussed and integrated into the model. Communication is possible via text comments that can hold rationales for proposals, support for proposals or challenges of these. For settings where all workstations are co-located, verbal communication also is possible. However, the results of Hawkey et al. (2005) as well as practical experience showed that individual desktop PCs often cause users to focus on the screen in front of them, and not on verbal communication with others. In addition, COMA provides no specific support for initial data collection. Other tools for distributed collaborative diagram editing (e.g. the IDEF0 modeler (Dean et al., 1997)) suffer from the same limitations.

Both Chen et al. (2008) and de Vreede et al. (2009) explore the possibilities of interactive whiteboards for modeling purposes. The former, however, focuses mostly on software modeling and the model construction phase, and provides little support for model revision (there is also no specific support for data collection). The latter explores process aspects of modeling with interactive whiteboards, but does so for the technology in general, and not for a concrete tool for collaborative modeling at these devices. Both resort to turn-taking and do not consider parallel input by multiple users. Frisch et al. (2010) investigate diagram editing on interactive tabletops; however, they focus on elicitation of gestures for diagram editing instead of an actual tool for collaborative modeling.

Some tools in the making (e.g. by Nolte and Prilla (2013)) target non-expert modelers, but to our best knowledge none of them supports all three task types and parallel input at the same time.

3 System and Interaction Techniques

In order to effectively support collaborative modeling and provide support for all three of the discussed modeling task types, we designed coBPM (collaborative Business Process Modeler), a visual editor for process models, that implements graph sketching features with concurrent user interaction in mind. New nodes can be inserted (via drag&drop) from a palette menu. Node movement is possible via direct manipulation. Contextual menus offer additional operations which can be called with a tap – an invisible buffer makes tapping easier for edges. Among others, these menus offer the possibility to change the type of an element at any point in time, thus providing basic support for brainstorming on model elements. Selections can be created with a drag gesture on empty screen space. They are permanent (until canceled), allowing multiple selections to coexist on the screen. Additional features (loading/saving, auto-snapping for drag-and-drop) have also been implemented.

**Concurrent Text-Entry:** when multiple users want to enter text in parallel, the restriction to a single keyboard in current operating systems quickly becomes a bottleneck. A first solution using physical keyboards has been proposed by Hartmann et al. (2009). Physical keyboards, however, occupy a
significant portion of the screen (permanently). Also, a more precise mapping of keyboards to input elements is required for process models with many elements. Thus, we opted for virtual keyboards directly linked to input elements. By default, keyboards are linked to the element used to spawn them; however, the link can be changed with a simple drag gesture. For each element, we emulate a cursor and handle input from the associated keyboard as if it came from the system’s default cursor.

Object Handover: when models are consolidated or revised, items typically have to be rearranged. Without proper support, users are either forced to sequence their actions, or suffer from (physical) interference (Hornecker et al., 2008) and the need to reach into territories others consider private (Scott et al., 2004). To address this problem, we extended the technique developed by Liu et al. (2008) by implementing throwing and subsequent element movement following a naïve physics model, instead of simple movement by the user. While throwing is an easy and intuitive metaphor for handing over objects on touchscreens, it also suffers from inaccuracies, especially on large screens. We have thus added the possibility to scale the magnetic field Liu et al. (2008) introduced for easier handover via the well-known pinch and reverse-pinch gestures. For cases in which two users have created magnetic fields, thrown elements will be attracted by the field which they enter first.

Interference Reduction: during modeling sessions, users may need repeated access to a region of the screen in which another user is also active; if they want to maintain the overall structure of the process while doing this, handing over elements is not adequate to reduce interference. Individual devices would solve this problem, but were found to reduce workspace awareness and collaboration quality (Hawkey et al., 2005). We thus designed a personal viewport technique, which provides local live copies of remote areas of the screen (c.f. (Bezerianos and Balakrishnan, 2005)), and adapted existing designs for quick creation and dismissal as well as flexible drag&drop operations. Viewports can be created from a selection, can be moved, (handed over) and adjusted in size. Users can also change the content displayed in them (using two-finger scrolling). Drag&drop is possible both within and across viewport borders. Within viewports, users may also start with a drag gesture, then hold the respective object with one finger, while panning the viewport canvas with two others, and finally drop the object at a location not shown initially. Basic awareness information is provided by semi-transparent rectangles on the main canvas indicating areas shown in personal viewports.

Conflict Resolution: when multiple users manipulate a shared artifact, contradicting actions may occur. While this is more common for distributed systems in which spatial separation of users and network latency reduce and delay awareness about other’s actions, co-located modeling may also be affected. To address this, we incorporated a voting on changes mechanism (Morris et al., 2004) – the selection state of an element is used to infer conflicts, i.e. a selection is effectively interpreted as an incomplete operation, and manipulations to the respective element by other users are interpreted as a conflict. When this happens, a voting dialog is displayed for all affected users (the initiating user is informed about the ongoing voting process instead). Currently, modifications are only accepted if all users vote for their acceptance; different strategies (e.g. majority vote) would, however, be possible.

4 Evaluation

We evaluated coBPM and the presented techniques with the following research questions in mind:

- **Teamwork** What effect do the interaction techniques have on the quality of teamwork? What impact do they have on the working style (in particular: division of labor) of participants?
- **Output** Does the use of the interaction techniques affect the participants’ understanding of the process model? Does it affect the quality of the model?
- **Usability** What effect do the techniques have on performance and ease of use? Are they perceived as useful? Do they help to reduce interference phenomena?

While we used a lab study to allow for inferential statistics, we strived to match a real modeling process as close as possible. For that purpose, we used typical tasks of collaborative modeling sessions, i.e. brainstorming, model creation, and model revision (cf. section 2), and situated them in a
fictional process re-engineering attempt of a small pizza shop. A total of 27 participants took part in the study (2 female). The average age of participants was 30.41 years (SD = 6.13) – job roles ranged from student to manager. All participants were smartphone owners and familiar with touchscreen devices. On average, participants had little to some experience with modeling (mean = 2.78, SD = 0.97 on a 5-point scale).

Participants were grouped into teams of three (i.e. 9 groups in total). We used a within-group design, i.e. all groups performed two tasks of comparable complexity, with three sub-tasks each (corresponding to the modeling tasks discussed above) – interaction techniques were enabled for one task (WT), disabled for the other (NT). Both order of tasks and use of interaction techniques were counterbalanced. Sub-tasks were time-boxed to avoid groups getting “stuck” on sub-tasks, and intermediate results for later sub-tasks were prepared to ensure equal starting conditions for all teams. Data was mostly collected using observation, video recordings and questionnaires.

Normal distribution was confirmed for all questionnaire scales (using the Kolmogorov Smirnov test); they were analyzed simultaneously in a one-factorial multivariate analysis of variance (MANOVA) with repeated measures. Main results have been summarized in Table 2.

**Perceived Teamwork Quality** A slightly modified and shortened (21 items) version of the team work questionnaire (TWQ) (Hoegl and Gemuenden, 2001) was used to evaluate communication, coordination, balance of member contributions, mutual support and effort. The average score over all subscales of the TWQ was 3.93 (WT) and 3.94 (NT), i.e. the presence/absence of the interaction techniques did not impact the perceived quality of the teamwork. Also, none of the subscales showed a significant difference.

**Perceived Quality of End-Product/Model Correctness** To assess the participant’s perception of their modeling results, the 4-item 7-point subscale Perceived Quality of End-Product (PQEP) of the Collaborative Modeling Process Quality (CMPQ) questionnaire (Ssebuggwawo et al., 2010) was used. However, there was no significant difference (WT: 5.97, NT: 5.85). As results for sub-tasks were known in advance for this study, we also assessed objective model correctness using a self-developed point scale (1p for each node with correct type and label, 0.5p for each correct inbound and outbound edge). The total score was normalized (per sub-task), i.e. a fully correct model resulted in a score of 1.0. Yet, correctness did also not differ significantly between conditions for any of the three sub-tasks.

**Perceived Ease of Use** Perceived ease of use of the system was assessed by the 4-item 7-point subscale Ease of Use of Medium (EOUM), also taken from the CMPQ questionnaire. The results show a significant improvement (t(t(26)=2.29, p=0.030) when techniques are used (WT: 5.57, NT: 5.12).

**Working Style/Division of Labor** To assess how well participants were able to organize their work in terms of distributing sub-tasks in the team, we used a 1-item 7-point scale in which participants specified how they perceived support for division of labor. The result is significantly higher (t(t(26)=2.67, p=0.013) with techniques (5.78) than without (4.93).

<table>
<thead>
<tr>
<th>Scale</th>
<th>Scale Values</th>
<th>With Techniques</th>
<th>No Techniques</th>
<th>Significant</th>
</tr>
</thead>
<tbody>
<tr>
<td>Teamwork Quality (TWQ)</td>
<td>1 to 5</td>
<td>3.93</td>
<td>3.94</td>
<td>No</td>
</tr>
<tr>
<td>Quality of End-Product (PQEP)</td>
<td>1 to 7</td>
<td>5.97</td>
<td>5.85</td>
<td>No</td>
</tr>
<tr>
<td>Ease of Use of Medium (EOUM)</td>
<td>1 to 7</td>
<td>5.57</td>
<td>5.12</td>
<td>Yes</td>
</tr>
<tr>
<td>Division of Labor (DoL)</td>
<td>1 to 7</td>
<td>5.78</td>
<td>4.93</td>
<td>Yes</td>
</tr>
<tr>
<td>Model Understanding (MU)</td>
<td>0 to 10</td>
<td>7.07</td>
<td>7.63</td>
<td>No</td>
</tr>
<tr>
<td>Visual Interference (VI)</td>
<td>no. of events</td>
<td>2.18</td>
<td>1.33</td>
<td>No</td>
</tr>
<tr>
<td>Physical Interference (PI)</td>
<td>no. of events</td>
<td>1.85</td>
<td>1.96</td>
<td>No</td>
</tr>
</tbody>
</table>

*Table 2. Summary of evaluation results (TWQ assessed by questionnaire of Hoegl and Gemuenden (2001), PQEP and EOUM assessed by questionnaire of Ssebuggwawo et al. (2010), DoL and MU assessed using self-developed scales).*
**Understanding of Model** Understanding of the created models was assessed by requesting model-specific knowledge in the form of yes/no-questions subsequent to the sub-tasks 2 (6 questions) and 3 (4 questions). For these 10 comprehension questions per condition, we averaged the score, finding no significant difference (WT: 7.07, NT: 7.63).

**Interference Events** The assessment of interference events was based on observations. Orienting towards categories from existing studies, we distinguished between physical and visual interference. Interference events were counted for each sub-task and averaged a posteriori. The mean frequency of visual interference events per sub-task was 2.18 (WT) and 1.33 (NT). Yet, this difference closely misses the threshold for significance (t(26) = 1.82, p = 0.08). Physical interference events show no significant difference (WT: 1.85, NT: 1.96).

**Usefulness of the Techniques** A self-developed 4-item questionnaire with 7-point Likert scales was used to assess the perceived usefulness of the techniques. The keyboard technique was clearly most appreciated (6.27); yet, it was also the main reason for visual interference. Personal viewports were appreciated in general, but space considerations were named as a limitation (4.15). Conflict resolution was also rated only moderately useful (4.13), presumably because the high workspace awareness of the co-located setup allowed participants to avoid editing conflicts in most cases. Object handover was considered least useful and essentially unneeded (3.88); users stated that they would rather stretch or throw objects just roughly into the direction of the other user (expecting him/her to pick them up).

5 Discussion and Conclusion

Our results show that given suitable tools, direct participation of users with little modeling expertise is indeed possible and ensures a generally high degree of teamwork, satisfaction with the outcome of the modeling process and understanding of the model. Moreover, the significant increase in ease of use of the system underlines the need for specific multi-user support when designing for collaborative work on large touchscreens. Further support for the techniques comes from the significant improvements in the support for easier division of labor. While it could be suspected that the increased parallelization comes at the cost of teamwork or output quality, our results indicate that this is not the case.

However, not all techniques proved to be equally useful. In particular for the object handover technique, our results indicate that, for practical applications, techniques in this area (also referred to as distal access in literature) may be dispensable for currently available tabletop systems. User comments and observations indicate that users preferred to stretch or simply throw objects roughly into the right direction (expecting another user to pick up the respective object) instead. This may change as display size grows beyond the distance users can reach when bending over the table. Furthermore, our analysis shows a trend towards more visual interference when interaction techniques are available. We observed visual interference especially when the keyboard technique was used frequently; however, ratings and comments indicate that it also prevented a more frequent usage of the personal viewports technique. This implies that, despite the larger than average displays, screen space can quickly become rare for collaborative tasks at interactive tabletops. At some point, this may even hinder domain expert involvement, as users may perceive the interface as too cluttered and hard to use. Thus, designers should carefully balance element size between the needs of easy touch interaction and the risk of visual interference when many elements are displayed.

For future work, we will incorporate these findings into our design, tuning personal viewports and keyboards in particular. We also plan to conduct a larger study, in which we will provide both a more extensive analysis of the emerging interaction and collaboration patterns and a comparative analysis of “traditional” collaborative modeling (domain experts providing verbal input only) and collaborative modeling with direct domain-expert input to the model. Also, although we target the editing of business process models as our primary use case, we have also started to investigate how our observations may transfer into other domains (e.g. modeling in software engineering).
Acknowledgements  This work has partly been funded by the German Ministry for Research and Education under grant no. 13N10711 and by the European Union under grant no. FP7-314350.

References


