

Process models: Neutral ground for collaboration, but power matters

Alexander Nolte¹, Michael Prilla¹

¹Information and Technology Management, University of Bochum, Germany

nolte@iaw.rub.de, prilla@iaw.rub.de

Abstract. Models and other process visualizations are common artifacts in organizations to visualize, analyze and sustain processes. They also serve as artifacts for communication. In these settings, models serve as neutral ground taking away anxieties usually arising when different parties work together. Models can also become tools of power enabling inferior participants to state their opinion or becoming tools superior participants want to control. Facilitation of model usage and development can give room to the positive aspects of this usage and diminish possible downsides. This paper deals with the question whether these effects can also be achieved in situations in which people use models on their own. As we found in a study, some of these effects are present without facilitation, but there is some work remaining to support all of them in practice.

Introduction

Visualizations of work such as process models are established tools in modern organizations. They support people in making perspectives explicit, understanding the work of others, jointly planning work and communicating about it (cf. Suchman 1995, Herrmann et al. 2004b, Prilla 2010). This is mirrored by many methods using models and other visualizations for the design of cooperation support (e.g. Beyer and Holtzblatt 1998, Conklin 2005, Herrmann 2009). Most of these methods rely on expert facilitators: Users do not use or manipulate visualizations directly, but their utterances are connected to visualizations by experts during or after the interaction. Thus, the usage of models by non-experts

depends on the availability of experts. Besides such settings, models are rarely used by other people (cf. Wand and Weber 2002, Prilla 2010). This slows down model development and prevents positive effects of models on cooperation.

People are capable of using models to support communication and manipulating them if they are given adequate means to do so (cf. Herrmann 2009, Prilla 2010, Prilla and Nolte 2010). Thus, adequate support of **self-directed interaction with models** (interaction without facilitation during model usage or manipulation) can diminish the problems of expert-driven model interaction and preserve the benefits of it. Thus, we created a prototype for such interaction and an experimental setting to explore users' interaction with process models. Through this we wanted to explore whether the benefits of models in expert-facilitated settings can also be reached in self-directed settings. In this paper, we report on results from this approach.

In what follows, we describe potentials and problems of model interaction. After that, we describe our experimental setting and the results stemming from our experiments. We then discuss our findings and elaborate on further work to be done for the implementation of self-directed interaction with models.

Potentials and pitfalls of collaborative model usage

There are several contributions from CSCW and related disciplines providing insights into potentials and pitfalls of the model usage we intend to support. Among others, we identified the following insights to be most influential for this:

- **Models for the exchange of perspectives and negotiation in grounding:** Models can be boundary objects (Star 1989), making perspectives explicit and support people in exchanging these perspectives and in negotiating common understanding (cf. Davies et al. 2004, Herrmann and Hoffmann 2005).
- **Models support communication:** Visualizations can make work visible to others (Suchman 1995), help designers from different backgrounds to find a common solution (Herrmann et al. 2004a), support communication about past activities and trigger communication (Fleck and Fitzpatrick 2006).
- **Models equalize politics and hierarchies:** Working with models can equalize differences in opinions and hierarchies among cooperators (Samarasan 1988, Herrmann et al. 2004b). However, in practice this work includes both the "artful crafting of peoples' stories" and political or hierarchical influences leading to "strategic manipulation of images" (Suchman 1995). Facilitation of group modeling can diminish unwanted influences (Samarasan 1988, Herrmann 2009).

The advantages described above stem from facilitated model usage. Therefore, we cannot take these benefits for granted in self-directed model interaction. Also, downsides such as unwanted influence may reoccur if we reduce the influence of

facilitators and let people use models on their own. Dealing with that needs exploring model interaction and analyzing it properly:

- Concerning its **applicability for negotiation processes**, we need to analyze model-related negotiation processes during self-directed model usage. For this, Beers et al. (2005) name primitives of negotiation such as contributing own perspectives, verifying the understanding of other perspectives, clarifying contributions and accepting or rejecting it.
- For the analysis of **communication about models** we need to look for model references in communication. Typical elements for this can be pointing to a model or referring to parts of a model during communication.
- In order to explore whether self-directed model usage has an **effect on political and hierarchical influences** on interaction, we need to analyze the conversations between actors using models according to arguments exchanged, decisions made and rationales behind them.

Setting: A prototype and environment for non-expert model interaction

The exploratory study was conducted with a prototype built based on experiences from prior work (c.f. Herrmann et al., 2010), which enables users to contribute to a process model without the need to be familiar with the respective process modeling language. It uses the SeeMe modeling language, which has been shown to be easily understood even by inexperienced people (cf. Herrmann et al. 2004a, Herrmann 2009). This prototype is coupled with an environment providing a large rear projection touch screen used to visualize process models and users' contributions to them as well as to manipulate resulting models via touch interaction (see Figure 1 for a glimpse of the environment). This environment provides an easy to use and intuitive interaction with models and is thus ideal for

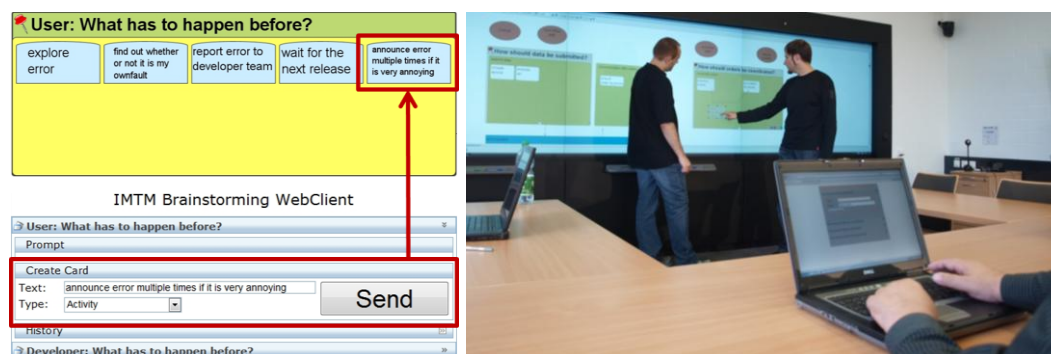


Figure 1: Contributing to a model from a web interface and transformation to a labeled model element (left) and self-directed interaction with process models on a large touch screen (right).

our purpose of exploring self-directed model interaction.

In our experiments, pairs of participants interacted with process models. We used scenarios of processes they were familiar with, which included two different roles (see Table 1). Each role was taken by one participant. We conducted five experiments (three covered scenario one) with two participants each, lasting about 30-45 minutes. We included different kinds of self-directed model usage into the experiments. First, participants were asked to add necessary parts of the process from the scenario to their own process model. After that, they had to explain the resulting models to each other and identify differences concerning both content and sequence of actions. After that, they were asked to articulate differences and similarities they found. During the experiment a facilitator guided the participants through the script of actions, but did not intervene in any model-related tasks.

The participants we worked with differed in terms of hierarchies between them (see Table 1). For two pairs, one participant was ranked significantly higher than the other and for the other three pairs, there was no big gap in hierarchies.

Table 1: Participants of the experiments and hierarchies between them.

Pair	Scenario	Participant 1	Participant 2	Hierarchy
P1	(1) Bug processing in software dev.	Project manager	Junior Developer	Yes
P2	(2) Book ordering in a library	Library owner	Library clerk	Yes
P3	(1) Bug processing in software dev.	Software user	Software developer	No
P4	(1) Bug processing in software dev.	Software user	Senior developer	No
P5	(2) Book ordering in a library	Library user	Library clerk	No

For analysis, we videotaped the workshops and an observer made notes. Afterwards, we analyzed this material according to the criteria described above.

Insights into self-directed model interaction

We observed models to support and influence the communication of participants in many ways. They oftentimes served as artifacts of common ground and reference. Unfortunately, we also observed influences of hierarchies. This shows that models can be used for grounding, but that power still matters in their usage. In what follows, we describe a selection of the most remarkable findings.

Models as means for the creation of neutral ground: In the experiments, the model-related tasks conducted by the participants fostered the creation of neutral ground. For example, we observed that visualizing the perspective of participants and communication about them fostered the understanding for each other's work. By e.g. pointing to models during discussion, the participants were able to identify differences and to cope with them on neutral ground and without the help of a

facilitator. In addition, participants told us that the preparation of models during the contribution of activities to their own model helped them “... *to create a compressed visualization of the own view...*” which made “...*the following discussion much easier...*” (developer from P3).

Models as a result of negotiation: During the discussion and – in absence of hierarchies – during the negotiation of differences, models proved to equalize gaps in different opinions. For example, the user of pair P4 (c.f. Table 1) criticized a lack of awareness on “...*the current state of a bug and the current priorities of development...*”. In contrast to that, the developer stated that he would “...*avoid giving feedback or even talking to the users...*” as this would distract him and slow him down, causing the bug to last longer. This discussion was triggered by the fact that during the comparison they had found that the user had included a *feedback*-activity into the process of bug processing whereas the developer had not. After a short discussion they agreed to a solution: The user would receive better feedback on bug processing while exact details would be left to the developer. This example shows how communication can be triggered in self-directed model interaction and how it can support the negotiation processes.

Models as a result of hierarchical decision: In contrast to the description above we experienced that hierarchy plays a decisive role in negotiation processes related to models. This was especially present in pairs P1 and P2, who had a huge difference in status. For P1, this resulted in the developer oftentimes instantly adopting the view of the user without any notable negotiation. When it came to a discussion about what is considered to be a bug, the user stated that “...*anything that does not work as expected is a bug...*” while the developer first considered a bug to be “...*a malfunction compared to how it is implemented...*”. However, after the user had explained his notion, the developer inclined to this view without any discussion possibly although he felt he was right. This shows that self-directed model interaction cannot prevent hierarchies from being an influence.

Summing up, we found all aspects discussed above in the observed interaction: self-directed work with models triggered communication and models were used as a reference in communication. Moreover, we found the benefit of models for perspective exchange and negotiation of common ground as well. For unwanted influences such as hierarchical decisions, we need to find solutions in order to consider self-directed model interaction to be an alternative to facilitated settings.

Conclusion and future work

In this paper, we report on an approach in enabling people to work with models on their own, preserving positive aspects of models for collaboration and diminishing possible problems. Results from our experiment indicate that – up to a certain extent – perspective exchange and negotiation about processes does not require

content related facilitation and can be done self-directed. Given the right means, users can express their perspectives on their own and are able to discuss and negotiate them. Furthermore, perspective exchange and discussion was not decisively influenced by hierarchies. However, when manipulation of process parts requires negotiation, hierarchy influences the outcome.

In the future we will conduct further experiments to gain more sustainable data on the insights described before – especially dealing with hierarchies will be part of this work. Currently, there are a lot of questions remaining for our work:

- How to compose models from different perspectives and negotiate them with special regard to hierarchy and how does group composition affect this?
- How is self-directed model usage and negotiation affected by the separation or intertwining of discussion and design with phases of assessment?
- To what extent are non-expert modelers capable of dealing with formalism and how can the functionality of a tool support them adequately?
- How much support can software provide for self-directed model interaction and when are modeling experts and facilitators required?

References

- Beers, P. J., H. P. A. Boshuizen, P. A. Kirschner and W. H. Gijssels (2005). Computer support for knowledge construction in collaborative learning environments. *Computers in Human Behavior*, vol. 21, pp. 623–644.
- Beyer, H. and K. Holtzblatt (1998). *Contextual design: defining customer-centered systems*. Morgan Kaufmann.
- Conklin, J. (2005). *Dialogue Mapping*. Wiley, Chichester.
- Davies, I., P. Green, M. Rosemann, S. Gallo, A. Brisbane and A. Ipswich (2004). Conceptual Modelling - What and Why in Current Practice: Conceptual modeling: ER 2004: 23rd International Conference on Conceptual Modeling. New York: Springer, pp. 668–681.
- Fleck, R. and G. Fitzpatrick (2006). Supporting collaborative reflection with passive image capture. In Hassanaly, P.; Herrmann, T.; Kunau, G.; Zacklad, M. (eds): *Cooperative Systems Design. Seamless Integration of Artifacts and Conversations - Enhanced Concepts of Infrastructure for Communication* Proceedings of COOP 2006. IOS Press.
- Herrmann, T., M. Hoffmann, G. Kunau and K. U. Loser (2004). A modelling method for the development of groupware applications as socio-technical systems. *Behaviour & Information Technology*, vol. 23. no. 2, pp. 119-135.
- Herrmann, Thomas (2009). Systems Design with the Socio-Technical Walkthrough. In Whitworth, Brian; Moor, Aldo de (eds): *Handbook of Research on Socio-Technical Design and Social Networking Systems*. Information Science Reference.
- Herrmann, Thomas and Marcel Hoffmann (2005). The Metamorphoses of Workflow Projects in their Early Stages. *Computer Supported Cooperative Work*, vol. 14. no. 5, pp. 399–432.

- Herrmann, Thomas, Gabriele Kunau, Kai-Uwe Loser and Natalja Menold (2004b). Sociotechnical Walkthrough: Designing Technology along Work Processes. In Clement, Andrew; Cindio, Fiorella; Oostveen, Anne-Marie; Schuler, Doug; van den Besselaar, Peter (eds): *Artful Integration: Interweaving Media, Materials and Practices* Proceedings of the eighth Participatory Design Conference 2004. ACM Press, pp. 132–141.
- Herrmann, Thomas, Alexander Nolte and Michael Prilla (2011). The Integration of Awareness, Creativity Support and Collaborative Process Modeling in Colocated Meetings. *International Journal on Computer-Supported Cooperative Work (IJCSCW)*. Special Issue on Awareness (accepted).
- Prilla, Michael (2010). Models, Social Tagging and Knowledge Management – A fruitful Combination for Process Improvement. In Rinderle-Ma, Stefanie; Sadiq, Shazia; Leymann, Frank (eds): *BPM International Workshops 2009*.
- Prilla, Michael and Alexander Nolte (2010). Fostering self-direction in participatory process design: Proceedings of the Participatory Design Conference 2010 (PDC 2010).
- Samarasan, D. (1988). Collaborative modeling and negotiation. *Conference on Supporting Group Work*, pp. 9-21.
- Star, Susan Leigh (1989). The structure of ill-structured solutions: boundary objects and heterogeneous distributed problem solving. In Gasser, Less; Huhns, Michael H. (eds): *Distributed Artificial Intelligence*. Morgan Kaufmann Publishers Inc., pp. 37-54.
- Suchman, L. (1995). Making work visible. *Communications of the ACM*, vol. 38. no. 9.
- Wand, Y. and R. Weber (2002). Research Commentary: Information Systems and Conceptual Modeling. *A Research Agenda. Information Systems Research*, vol. 13. no. 4, pp. 363–376.