Tracing the Process of Process Modeling with Modeling Phase Diagrams

Jakob Pinggera[1], Stefan Zugal[1], Matthias Weidlich[2], Dirk Fahland[3],
Barbara Weber[1], Jan Mendling[4], and Hajo A. Reijers[3]

[1] University of Innsbruck, Austria
`jakob.pinggera|stefan.zugal|barbara.weber@uibk.ac.at`
[2] Hasso-Plattner-Institute, University of Potsdam, Germany
`matthias.weidlich@hpi.uni-potsdam.de`
[3] Eindhoven University of Technology, The Netherlands
`d.fahland|h.a.reijers@tue.nl`
[4] Humboldt-Universität zu Berlin, Germany
`jan.mendling@wiwi.hu-berlin.de`

Abstract. The quality of a business process model is presumably highly dependent upon the modeling process that was followed to create it. Still, there is a lack of concepts to investigate this connection empirically. This paper introduces the formal concept of a phase diagram through which the modeling process can be analyzed, and a corresponding implementation to study a modeler's sequence of actions. In an experiment building on these assets, we observed a group of modelers engaging in the act of modeling. The collected data is used to demonstrate our approach for analyzing the process of process modeling. Additionally, we are presenting first insights and sketch requirements for future experiments.

Key words: business process modeling, modeling phase diagrams, process model quality, empirical research, modeling process

1 Introduction

Considering the heavy usage of business process modeling in all types of business contexts, it is important to acknowledge both the relevance of process models and their associated quality issues. On the one hand, it has been shown that a good understanding of a process model has a positive impact on the success of a modeling initiative [1]. On the other hand, actual process models display a wide range of problems that impede upon their understandability [2]. Clearly, an in-depth understanding of the factors of process model quality is in demand.

The quality of process models can be evaluated along a wide spectrum of properties, such as syntactic correctness or semantic accuracy [3]. Most research in the field puts a strong emphasis on the *product* or *outcome* of the process modeling act [4, 5]. For this category of research, the resulting model is the object of analysis. Many other works—instead of dealing with the quality of individual models—focus on the characteristics of modeling languages [6, 7]. However, these studies put less emphasis on the fact that model quality is presumably dependent upon the modeling process that was followed to create it. While there is work on micro-management of creating models [8], there is a notable research gap on how the process of process modeling can be analyzed quantitatively.

In this paper, we address this specific problem. In particular, we focus on the *formalization phase* in which a process modeler is faced with the challenge of constructing a syntactically correct model that reflects a given domain description (cf. [9]). This appeals to one's *ability to model* [10], arguably the most important capability of a modeler according to its expected effect on the quality of the ensuing model. The formalization of process models—which can be considered a *process in itself*—is crucial for obtaining a good modeling result and to overcome quality problems right from the start [2].

Given this context, we introduce an analysis technique called *modeling phase diagram*. The technique supposes to record all modeling activities throughout the creation of a process model in a log. Our technique classifies the recorded modeling activities according to cognitive research; the classification, in turn, allows to visualize and analyze the process of modeling itself in a diagram. The technique has been implemented in a graphical modeling tool that logs a user's modeling activities in the background. We conducted a modeling session with graduate students to demonstrate the feasibility of our approach; we present first insights and outline requirements for further experiments.

The paper is structured accordingly. We continue with a discussion of the fundamental cognitive considerations about the process of process modeling. Then, Sect. 3 presents our general approach along with the corresponding algorithms to generate modeling phase diagrams. The setup of the modeling experiment and its results are described in Sect. 4, along with a discussion of lessons learned. Then, Sect. 5 discusses related research before Sect. 6 concludes the paper.

2 Cognitive Foundations of the Process of Modeling

Before investigating the process of process modeling, a discussion of its cognitive foundations is required. Sect. 2.1 introduces a basic model for understanding information processing within the human mind. In particular, the concept of "chunking" is introduced. The different phases of this process are described in Sect. 2.2, namely comprehension, modeling, and reconciliation.

2.1 A Model of the Mind

A central insight from cognitive research is that the human brain contains specialized regions that contribute different functionality to the process of *solving complex problems*. The *modal model* describes the mind as being separated into different types of memory, the most important for our research being *working memory*, the place where comparing, computing and reasoning takes place [11]. Although working memory is the main working area of the brain, it can store only a limited amount of information, which is forgotten after 20–30 seconds if not refreshed [12]. The working memory's span is measured in chunks, being able to store a more or less constant number of items [13]. Although this capacity is reduced while performing difficult tasks, the span of working memory can be increased by suitable organization of information [11]. For example, when asked to repeat the sequence “U N O C B S N F L”, most people miss a character or two as the number of characters exceeds the working memory’s span. However, people being familiar with acronyms might recognize and remember the sequence “UNO CBS NFL”, effectively reducing the working memory’s load from nine to three so-called “chunks” [11, 14, 15]. As modeling is related to problem solving [14], modelers with a better understanding of the modeling tool, the notation, or a superior ability of extracting information from requirements can utilize their working memory more efficiently when creating process models [16].

2.2 Process of Process Modeling — An Iterative Process

During the formalization phase process modelers are working on creating a syntactically correct process model reflecting a given domain description by interacting with the process modeling tool [9]. This modeling process can be described as an iterative and highly flexible process [14, 17], dependent on the individual modeler and the modeling task at hand [18]. At an operational level, the modeler’s interactions with the tool would typically consist of a cycle of the three successive phases of comprehension, modeling, and reconciliation.

Comprehension. In the comprehension phase modelers try to understand the requirements to be modeled as well as the model that has been created so far. Consequently, working memory is filled with knowledge extracted from the requirements and, if available, from the process model itself. The amount of information stored in working memory depends on the modeler’s abilities and her knowledge organization (cf. Section 2.1).

Modeling. The modeler uses the information acquired and stored in working memory during the previous comprehension phase for changing the process model. The process modeler’s utilization of working memory influences the number of modeling steps executed during the modeling phase before forcing the modeler to revisit the requirements for acquiring more information.

Reconciliation. After the modeling phase, modelers reorganize the process model (e.g., renaming of activities) and utilize the process model’s *secondary notation* (e.g., notation of layout, typographic cues) to enhance the process model’s understandability [19, 20]. However, the number of reconciliation phases in the process of process modeling is influenced by a modeler’s ability of placing elements correctly when creating them, alleviating the need for additional layouting. Furthermore, the factual use of secondary notation is subject to the modeler’s personal style [19]. The improved understandability supports the comprehension phase of the subsequent iteration, as the process model becomes more comprehensible for the modeler when coming back to it [19]. In particular, during the subsequent comprehension phase the modeler has to identify the part of the model to work on next. A better laid out model helps identifying a suitable area of the model, causing less distraction and therefore enables the modeler to store more information in working memory that can be incorporated in the process model.

3 Investigating the Process of Process Modeling

This section introduces a method to investigate the process of process modeling via *modeling phase diagrams*. Sect. 3.1 describes how the modeling process can be captured, providing the basis for its analysis in Sect. 3.2. Finally, Sect. 3.3 illustrates how the modeling process can be measured.

3.1 Capturing Events of the Process of Process Modeling

In order to get a detailed picture of how process models are created, we use the Cheetah Experimental Platform (CEP). CEP has been specifically designed for investigating the process of process modeling in a systematic manner [21]. In particular, we instrumented a basic process modeling editor within CEP to record each user's interactions together with the corresponding time stamp in an event log, describing the creation of the process model step by step.

When focusing on the process modeling environment, the development of process models consists of adding nodes and edges to the process model, naming or renaming these activities, and adding conditions to edges. In addition to these interactions a modeler can influence the process model's secondary notation, e.g., by laying out the process model using move operations for nodes or by utilizing bendpoints to influence the visualization of edges. A complete overview of the possible interactions is provided in Table 1.

3.2 Analyzing the Process of Process Modeling

By capturing all of the described interactions with the modeling tool, we are able to *replay* a recorded modeling process at any point in time without interfering with the modeler or her problem solving efforts. This allows for observing how the process model unfolds on the modeling canvas. A demonstration of the replay function is available at http://cheetahplatform.org. Fig. 1 illustrates the basic idea of our technique. Fig. 1a shows several states of a typical modeling process as it can be observed during replay. Fig. 1c shows the states of a different modeling process that nonetheless results in the *same* model. This replay functionality of CEP allows to observe in detail how modelers create the model on the canvas.

We postulate that observations made for the process of modeling at a syntactic level can be traced back to the various phases of the modeling process (cf. Section 2.2). Clearly, modeling manifests in the creation of model elements.

User Interaction	Description	User Interaction	Description
CREATE NODE	Create activity or gateway	RENAME	Rename an activity
DELETE NODE	Delete activity or gateway	UPDATE CONDITION	Update an edge's condition
CREATE EDGE	Create an edge connecting two nodes	MOVE NODE	Move a node
DELETE EDGE	Delete edge	MOVE EDGE LABEL	Move the label of an edge
CREATE CONDITION	Create an edge condition	CREATE/DELETE/MOVE EDGE BENDPOINT	Update the routing of an edge
DELETE CONDITION	Delete an edge condition		
RECONNECT EDGE	Reconnect edge from one node to another		

Table 1. User Interactions Recorded by Cheetah Experimental Platform

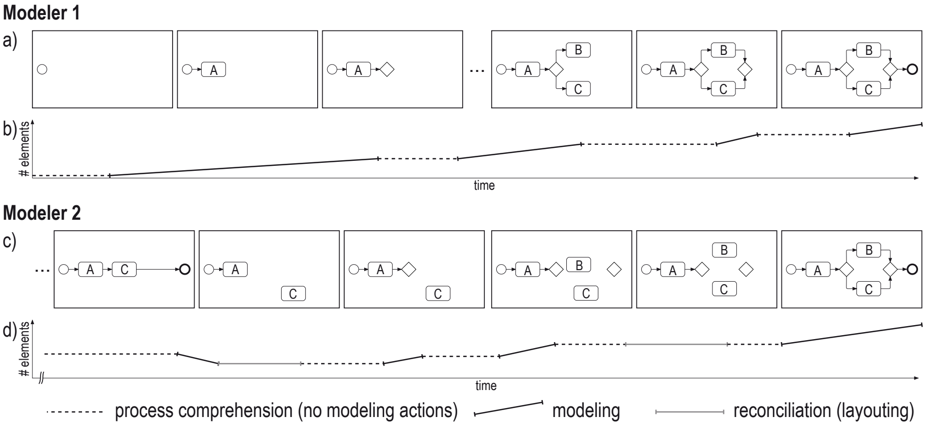

Fig. 1. Two Different Processes of Process Modeling to Create the Same Process Model

Hence, (1) a *modeling phase* consists of a sequence of interactions to create or delete model elements such as activities or edges. (2) A modeler usually does not create a model in a continuous sequence of interactions. She rather pauses after several interactions to inspect the intermediate result of her modeling and to plan the next steps. Syntactically, this manifests in reduced modeling activity or even inactivity. We refer to such a phase as a *comprehension phase*. (3) Besides modeling and thinking, a modeler also needs to *reorganize* the model. Reconciliation interactions manifest in moving or renaming model elements to prepare the next modeling interactions or to support her comprehension of the model. A sequence of such interactions is a *reconciliation phase*.

To obtain a better understanding of the modeling process and its phases, we supplement model replay with a *modeling phase diagram*. Such a diagram quantitatively highlights the three phases of modeling, comprehension, and reconciliation. It primarily depicts how the size of the model (vertical axis) evolves over time (horizontal axis), as can be seen in Fig. 1b and Fig. 1d for the modeling processes in Fig. 1a and Fig. 1c, respectively. The graph partitions the user interactions into the three phases, based on the kind of interactions and their frequencies in the modeling process.

So, we can read from Fig. 1b that the modeler created the model in a straightforward series of modeling steps interrupted by periods of comprehension. The modeling process in Fig. 1d shows a different approach. After some modeling, the modeler removes parts of the created model and moves an activity to make space for some control-flow constructs, as indicated by the reconciliation phase. Then, several model elements are placed and laid out before the model is completed. Note that the resulting models are identical. Yet, the phase diagrams show significant differences between both modeling processes. This illustrates the value of analyzing the modeling process in the described manner beyond the inspection of the process models themselves.

Phase	Identification Criteria
Comprehension	no interaction with the system for longer than a predefined threshold
Modeling	creating modeling elements (activities, gateways, edges), deleting modeling elements, reconnecting edges, adding/deleting edge conditions
Reconciliation	layouting of edges, moving of modeling elements, renaming of activities, updating edge conditions

Table 2. Identification of Phases of the Process of Process Modeling

3.3 Measuring the Process of Process Modeling

Based on the theoretical background regarding the process of process modeling, we developed an algorithm for automatically extracting *modeling phase diagrams* (cf. Fig. 1) from the logs created by CEP. For this purpose, the user interactions depicted in Table 1 are categorized into modeling and reconciliation interactions as listed in Table 2. Comprehension phases are determined by measuring the time when no interaction with the system is recorded.

Algorithm 1 Extracting the Process of Process Modeling

```
Require: $interactions\ [I_1, I_2, \ldots I_n]$
Require: $threshold_c, threshold_d$
 1: $phases \leftarrow$ [new comprehension phase]
 2: for all $i$ such that $1 \leq i \leq n$ do
 3:     if $i > 1$ and durationBetween$(I_{i-1}, I_i) > threshold_c$ then
 4:         add new comprehension phase to $phases$
 5:     $previousPhase \leftarrow last(phases)$
 6:     $upcomingPhase \leftarrow identifyUpcomingPhase(interactions, I_i)$
 7:     if $upcomingPhase = previousPhase$ then
 8:         add $I_i$ to $previousPhase$
 9:     else
10:         $durationOfUpcomingPhase \leftarrow duration(upcomingPhase)$
11:         if $durationOfUpcomingPhase > threshold_d$ then
12:             add $upcomingPhase$ to $phases$
13:         else
14:             add $I_i$ to $previousPhase$
```

Algorithm 1 shows the procedure for extracting the phases of the modeling process from the user interactions logged by CEP. Comprehension phases are identified in lines 3–4 of the algorithm by evaluating the time between interactions and comparing it to the minimal duration of a comprehension phase defined by $threshold_c$. Line 6 calculates the upcoming phase by integrating all following interactions that are of the same type as the current interaction until a different interaction type is found. The upcoming phase is subsequently compared to the previous phase and, in case they match, added to the previous phase. Otherwise, the duration of the upcoming phase is assessed by computing the time between the first interaction and the last interaction of the identified phase (line 10). If the duration is longer than $threshold_d$ a new phase is added to the list of identified phases (line 12). Otherwise, the interaction is added to the previous phase. Time periods between two phases, being shorter than $threshold_c$, are indicated as gaps in the phase diagrams as it cannot be determined whether the user was still in the first phase or already in the second one.

Additionally, comprehension phases which are interrupted by short modeling or reconciliation phases are merged, as users sometimes move single elements of the process model or add single elements (e.g. a start event) while making sense of the requirements. Using the phases extracted by Algorithm 1, Algorithm 2 identifies situations comprising two comprehension phases being separated by an intermediary modeling or reconciliation phase. If the duration of the intermediary phase is smaller than the $threshold_a$ the two comprehension phases and the intermediary phase are merged to a single comprehension phase.

Algorithm 2 Merging of Comprehension Phases

```
Require: phases $[P_1, P_2, \dots P_n]$
Require: $threshold_a$
for all $i$ such that $1 \leq i \leq n-2$ do
    if $P_i$ and $P_{i+2}$ are comprehension phases and $duration(P_{i+1}) < threshold_a$ then
        merge phases $P_i$, $P_{i+1}$ and $P_{i+2}$
```

4 Experimental Investigation

This section describes a modeling session conducted to collect modeling processes for demonstrating our technique. Sect. 4.1 introduces the setup used for data collection. Sect. 4.2 describes the execution while Sect. 4.3 presents two of the collected modeling processes. Sect. 4.4 presents first insights into the process of process modeling and outlines lessons learned.

4.1 Preparing the Experiment

The main goals of the described experiment have been (1) to investigate the process of creating a formal process model in BPMN from an informal description, and (2) to assess the applicability of the described approach. The object that was to be modeled is an actual process run by the "Task Force Earthquakes" of the German Research Center for Geosciences (GFZ). The task force runs in-field missions after catastrophic earthquakes [22]. Subjects were asked to model the "Transport of Equipment" process based on a structured description of how the task force transports its equipment from Germany to the disaster area[1].

To mitigate the risk that the modeling processes were impacted by complicated tools or notations [14], we decided to use a subset of BPMN for our experiment. In this way, modelers were confronted with a minimal number of distractions, but the essence of how process models are created could still be captured. A pre-test was conducted at the University of Innsbruck to ensure the usability of the tool and the understandability of the task description. This led to further improvements of CEP and minor updates to the task description.

---

[1] Material download: http://pinggera.info/experiment/ModelingPhaseDiagrams

4.2 Conducting the Experiment

The experiment was conducted in November 2009 with students of a graduate course on Business Process Management at Eindhoven University of Technology and students from Humboldt-Universität zu Berlin following a similar course. The modeling session at each university started with a demographic survey, followed by a modeling tool tutorial explaining the basic features of CEP. After that, the actual modeling task was presented in which the students had to model the above described "Transport of Equipment" process. This was done by 20 students in Eindhoven and 6 students in Berlin. By conducting the experiment during class and closely monitoring the students, we mitigated the risk of falsely identifying comprehension phases due to external distractions. No time restrictions were imposed on the students.

4.3 Modeling Phase Diagram Examples

This section presents two modeling processes and the corresponding modeling phase diagrams from the experiment[1]. Recall that in such a diagram the horizontal axis represents time and the vertical axis the number of elements in the process model. Differences in the number of elements in the process models can be attributed to superfluous activities, missing activities or different usage of gateways among our subjects. We explicitly connect each example to the modeling phase diagram, and to observations that we obtained by replaying each of the modeling process in CEP.

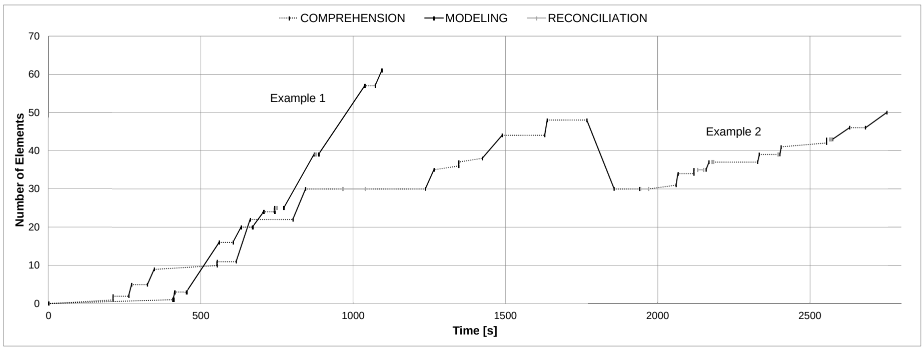

Fig. 2. Two Processes of Process Modeling

Example 1. The modeling phase diagram of Example 1 (cf. Fig. 2) shows a rather long initial comprehension phase after which alternating comprehension and modeling phases can be observed. All modeling phases are very long and steep, i.e., much model content is added per iteration. Virtually no reconciliation can be observed.

---

[1] We used $threshold_c = 30s$; $threshold_d = 2s$; $threshold_a = 4s$

Modeling Style. Replaying the modeling process in CEP shows that the modeler appears to have a clear conception of the model to be created. Elements are placed on the canvas in large chunks, while all elements are being placed to appropriate positions so that no movement of elements is required.
Modeling Result. The created process model moderately approximates the expected modeling outcome in terms of graph edit distance (cf. [23]), due to some superfluous activities. The created model is, however, free of syntax errors and behavioral anomalies, such as deadlocks.

Example 2. The modeling phase diagram of Example 2 is depicted in Fig. 2. This process starts very similar to Example 1 by adding model elements in large chunks after average comprehension phases. At around 800s, the process starts to deviate by a very long comprehension phase. After this phase, modeling continues similarly to the beginning of the process until a large part of the model is removed (falling iteration around 1800s). The modeling process completes in iterations with significantly longer comprehension phases, short modeling phases, and some time spent on reconciliation.
Modeling Style. The replay shows that the modeler started modeling with a clear idea of the model to be created in mind. However, the modeler used some BPMN modeling elements wrongly, i.e., start events as intermediate states. At about 2/3 of the model created, the modeler realizes the mistake, removes all intermediate states, inserts missing gateways and arcs, and completes the model.
Modeling Result. The model shows an above average similarity to the expected modeling result in terms of graph edit distance. While the model is syntactically correct, it contains two deadlocks due to a wrong pairing of gateways.

4.4 Lessons Learned

The main purpose of our experiment was to validate the feasibility of using modeling phase diagrams for gaining insights into the process of process modeling, more specifically into the formalization of process models. We could demonstrate that our technique allows to empirically investigate aspects of modeling that could not be observed or analyzed earlier. In particular, we witnessed different approaches on layouting the process models. Some of the modelers placed many of the key activities at strategic places on the canvas *right from the start*, without ever having to change their position again, alleviating the need for further reconciliation. Others were more careless when placing modeling elements on the canvas, but continuously invested into improving the process model's optical appearance, resulting in many short reconciliation phases. Interestingly, when replaying the modeling processes we observed that all modelers seemed to dislike activities disappearing from sight. By placing elements far apart, a modeler can in principle span up a 'virtual' canvas beyond the size of the physical dimensions of the computer display. However, many modelers—when reaching the bounds of the physical canvas—spent much time on reconciliation exactly to prevent such a situation. Besides these principle observations on layouting, we were also able to track when a modeler faced difficulties as this directly manifests in the modeling phase diagram, for instance in phases where elements are removed (cf. Fig. 2). A subsequent replay usually allowed us to understand very well the nature of the difficulty, such as an improper use of gateways. Observations like these would not have been possible without the specific setup employed, allowing us to investigate the process underlying the creation of the process model. Therefore, we conclude that modeling phase diagrams and CEP's replay feature constitute valuable assets for further research on understanding the factors influencing the process of creating process models.

5 Related Work

Our work is essentially related to three streams of research: model quality frameworks, research on the process of modeling, and insights into modeling expertise.

There are different frameworks and guidelines available that define quality for process models. Among others, the SEQUAL framework uses semiotic theory for identifying various aspects of process model quality [3], the Guidelines of Process Modeling describe quality considerations for process models [25], and the Seven Process Modeling Guidelines define desirable characteristics of a process model [26]. While each of these frameworks has been validated empirically, they rather take a static view by focusing on the resulting process model, but not on the act of modeling itself. Our research complements these works by providing the methodological means for tracing model quality back to different modeling strategies and competence.

Research on the process of modeling typically focuses on the interaction between different parties. In a classical setting, a system analyst directs a domain expert through a structured discussion subdivided into the stages elicitation, modeling, verification, and validation [9, 27]. The procedure of developing process models in a team is analyzed in [8] and characterized as a negotiation process. Interpretation tasks and classification tasks are identified on the semantic level of modeling. Participative modeling is discussed in [28]. These works build on the observation of modeling practice and distills normative procedures for steering the process of process modeling towards a good completion. Our tool-based approach focuses on the formalization of process models by generating fine-granular phase diagrams from event logs, inspired by process mining techniques. In a similar vein, the replay function of ProcessWave has been used to analyze the modeling collaboration support provided by BPM tools [29].

Finally, research has discussed various aspects of modeling expertise. Results of a survey on process modeling success establish modeler expertise as a critical success factor [30]. On the one hand, different experiments have shown that expertise is a key factor for comprehension performance [20, 24]. On the other hand, expert modelers spend much more time and dedicate more attention to an appealing layout of the models [19]. Our research design provides means for making modeling phases visible based on log data. In this way, this work offers a way to observe how variations in expertise translate into models of different quality by using different modeling strategies.

6 Summary, Conclusions and Future Work

In this paper, we motivated the need for a detailed understanding of the process of process modeling and provided the conceptual background to analyze the modeling process itself. Through the unique setup of the described modeling environment involving Cheetah Experimental Platform, we have been able to observe under experimental conditions the process of process modeling at close quarters. In particular, we presented a technique for extracting modeling phase diagrams and demonstrated the feasibility of our approach in an experiment from which we presented insights into the creation of two example process models.

Requirements for Experiments. In addition to presenting lessons learned with respect to the act of modeling, we collected requirements for future empirical experiments investigating the process of modeling. Besides standard requirements such as a significant number of participants with representative skills, some specific requirements can be postulated. While the modelers that participated in this study are not representative for the modeling community at large, the question can be raised whether experienced modelers exhibit the same style of modeling observable with our technique as skillful yet inexperienced modelers; observing differences may yield fruitful insights regarding teaching modeling. In addition, it has to be recognized that there are some aspects of the process of process modeling that cannot be measured using solely the modeling tool (e.g., sense making of an informal process description). For this purpose, think-aloud protocols and/or eye tracking technologies might be considered.

Future Work. In the short term, our follow up research will be concerned with collecting additional data, identifying different modeling practices and developing a categorization of different modeling styles. Furthermore, we are planning to develop measurements to quantify the modeling process that might be connected to the quality of the resulting process model. In addition, we are planning to extend the basis for our findings by involving modeling experts. In the longer term, our interest is with how superior modeling styles can be acquired or trained, if at all. Even if we understand that experts increase their cognitive capacity through a masterly organization of knowledge, i.e., chunking, a question with a high practical relevance is how such techniques can be developed and trained.

Acknowledgements. This research was funded by the Austrian Science Fund (FWF): P23699-N23.

References

1. Kock, N., Verville, J., Danesh-Pajou, A., DeLuca, D.: Communication flow orientation in business process modeling and its effect on redesign success: results from a field study. DSS **46** (2009) 562–575
2. Mendling, J.: Metrics for Process Models: Empirical Foundations of Verification, Error Prediction, and Guidelines for Correctness. Springer (2008)
3. Krogstie, J., Sindre, G., Jørgensen, H.: Process models representing knowledge for action: a revised quality framework. EJIS **15** (2006) 91–102

4. Van der Aalst, W., Ter Hofstede, A.: Verification of workflow task structures: A petri-net-baset approach. IS **25** (2000) 43–69
5. Gruhn, V., Laue, R.: Complexity metrics for business process models. In: Proc. ICBIS '10. (2006) 1–12
6. Siau, K., Rossi, M.: Evaluation techniques for systems analysis and design modelling methods-a review and comparative analysis. ISJ (2007)
7. Moody, D.L.: The "Physics" of Notations: Toward a Scientific Basis for Constructing Visual Notations in Software Engineering. IEEE Trans. Software Eng. **35** (2009) 756–779
8. Rittgen, P.: Negotiating Models. In: Proc. CAiSE '07. (2007) 561–573
9. Hoppenbrouwers, S., Proper, H., Weide, T.: A fundamental view on the process of conceptual modeling. In: Proc. ER '05. (2005) 128–143
10. Persson, A., Stirna, J.: Towards Defining a Competence Profile for the Enterprise Modeling Practitioner. The Practice of Enterprise Modeling (2010) 232–245
11. Gray, P.: Psychology. Worth Publishers (2007)
12. Tracz, W.: Computer programming and the human thought process. Software: Practice and Experience **9** (1979) 127–137
13. Miller, G.: The Magical Number Seven, Plus or Minus Two: Some Limits on Our Capacity for Processing Information. The Psychological Review **63** (1956) 81–97
14. Crapo, A.W., Waisel, L.B., Wallace, W.A., Willemain, T.R.: Visualization and the process of modeling: a cognitive-theoretic view. In: Proc. KDD '00. (2000) 218–226
15. Newell, A.: Unified Theories of Cognition. Harvard University Press (1990)
16. Shanteau, J.: How much information does an expert use? Is it relevant? Acta Psychologica **81** (1992) 75–86
17. Morris, W.T.: On the Art of Modeling. Management Sc. **13** (1967) B–707–B–717
18. Willemain, T.R.: Model Formulation: What Experts Think about and When. Operations Research **43** (1995) 916–932
19. Petre, M.: Why Looking Isn't Always Seeing: Readership Skills and Graphical Programming. Commun. ACM (1995) 33–44
20. Mendling, J., Reijers, H.A., Cardoso, J.: What Makes Process Models Understandable? In: Proc. BPM '07. (2007) 48–63
21. Pinggera, J., Zugal, S., Weber, B.: Investigating the process of process modeling with cheetah experimental platform. In: Proc. ER-POIS '10. (2010) 13–18
22. Fahland, D., Woith, H.: Towards Process Models for Disaster Response. In: Proc. PM4HDPS '08. (2008) 254–265
23. Dijkman, R., Dumas, M., García-Bañuelos, L.: Graph Matching Algorithms for Business Process Model Similarity Search. In: Proc. BPM '09. (2009) 48–63
24. Reijers, H., Mendling, J.: A study into the factors that influence the understandability of business process models. IEEE Trans. Sys. Man & Cybernetics, A (2011)
25. Becker, J., Rosemann, M., von Uthmann, C.: Guidelines of business process modeling. In: BPM. Volume 1806 of LNCS. Springer (2000) 241–262
26. Mendling, J., Reijers, H.A., van der Aalst, W.M.P.: Seven process modeling guidelines (7pmg). Information & Software Technology **52** (2010) 127–136
27. Frederiks, P., Weide, T.: Information modeling: The process and the required competencies of its participants. DKE **58** (2006) 4–20
28. Stirna, J., Persson, A., Sandkuhl, K.: Participative Enterprise Modeling: Experiences and Recommendations. In: Proc. CAiSE '07. (2007) 546–560
29. Hahn, C., Recker, J., Mendling, J.: An exploratory study of it-enabled collaborative process modeling. In: Proc. BPD '10. (2010)
30. Bandara, W., Gable, G., Rosemann, M.: Factors and measures of business process modelling: model building through a multiple case study. EJIS **14** (2005) 347–360