

Styles in business process modeling: an exploration and a model

Jakob Pinggera · Pnina Soffer · Dirk Fahland ·
Matthias Weidlich · Stefan Zugal · Barbara Weber ·
Hajo A. Reijers · Jan Mendling

Received: 30 September 2012 / Revised: 26 March 2013 / Accepted: 10 May 2013
© The Author(s) 2013. This article is published with open access at Springerlink.com

Abstract Business process models are an important means to design, analyze, implement, and control business processes. As with every type of conceptual model, a business process model has to meet certain syntactic, semantic, and pragmatic quality requirements to be of value. For many years, such quality aspects were investigated by centering on the properties of the model artifact itself. Only recently, the process of model creation is considered as a factor that influences the resulting model's quality. Our work contributes to this stream of research and presents an explorative analysis of the process of process modeling (PPM). We report on two large-scale modeling sessions involving 115 students. In these sessions, the act of model creation, i.e., the PPM, was automatically recorded. We conducted a cluster analysis on this data and identified three distinct styles of modeling. Further, we investigated how both task- and

modeler-specific factors influence particular aspects of those modeling styles. Based thereupon, we propose a model that captures our insights. It lays the foundations for future research that may unveil how high-quality process models can be established through better modeling support and modeling instruction.

Keywords Business process modeling · Process of process modeling · Modeling styles · Cluster analysis

1 Introduction

Considering the intense usage of business process modeling in all types of business contexts, the relevance of process models has become obvious. However, actual process models display a wide range of problems [20] falling into the quality dimensions of syntactic, semantic, and pragmatic quality of a model [17]. Syntactic and semantic quality relate to model

Communicated by Dr. Selmin Nurcan.

This research is supported by Austrian Science Fund (FWF): P23699-N23. The stay of Dr. Pnina Soffer in Innsbruck, Austria was funded by BIT School.

J. Pinggera (✉) · S. Zugal · B. Weber
University of Innsbruck, Innsbruck, Austria
e-mail: jakob.pinggera@uibk.ac.at

S. Zugal
e-mail: stefan.zugal@uibk.ac.at

B. Weber
e-mail: barbara.weber@uibk.ac.at

P. Soffer
University of Haifa, Haifa, Israel
e-mail: spnina@is.haifa.ac.il

D. Fahland · H. A. Reijers
Eindhoven University of Technology, Eindhoven, The Netherlands
e-mail: d.fahland@tue.nl

H. A. Reijers
email: h.a.reijers@tue.nl

M. Weidlich
Technion - Israel Institute of Technology,
Haifa, Israel
email: weidlich@tx.technion.ac.il

J. Mendling
Vienna University of Economics and Business,
Vienna, Austria
email: jan.mendling@wu.ac.at

H. A. Reijers
Perceptive Software, Apeldoorn,
The Netherlands

construction and address the correct use of the modeling language and the extent to which the model truthfully represents the real-world behavior, respectively. Pragmatic quality addresses the extent to which a model supports its usage for purposes such as understanding behavior and system development. Considering process models whose purpose is to develop an understanding of real-world behavior, pragmatic quality is typically related to the understandability of the model [15]. Clearly, an in-depth understanding of the factors influencing the various quality dimensions of process models is in demand.

Most research in this area puts a strong emphasis on the product or outcome of the process modeling act (e.g., [10, 48]). For this category of research, the resulting model is the object of analysis. The objective, for example, is to relate how structural characteristics of the model relate to its pragmatic quality. Instead of dealing with the quality of individual models, many other works focus on the characteristics of modeling languages (e.g., [26, 42]). Recently, research has begun to explore another dimension that presumably affects the quality of business process models by incorporating the process of creating a process model into their investigations (e.g., [33, 37, 43]). In particular, the focus has been put on the formalization phase in which a process modeler is facing the challenge of constructing a syntactically correct model reflecting a given domain description (cf. [14]). Our research can be positioned within the latter stream of research.

Earlier works observed the existence of genuinely different process modeling styles [37]. Moreover, it has been shown that certain characteristics of how a modeler creates a model correlate with the quality of the created model [6]. What has not taken place is a systematic investigation of which distinct modeling styles can be observed in reality, what characterizes these modeling styles, and which factors influence that a particular modeling style is followed. Answers to these questions form a prerequisite to a systematic understanding of how modeling influences model quality and how it can be improved, for instance, by providing adequate modeling environments and by addressing quality concerns when teaching how to model.

This paper identifies distinct modeling styles, together with the factors that are supposed to influence which particular modeling style is followed. In an explorative study, we conducted modeling sessions with 115 students solving two different modeling tasks. We recorded each modeler's interactions with a modeling tool that captures all details of how the actual modeling was done. We then applied data mining techniques to identify different modeling styles; a cluster analysis suggests the existence of three modeling styles. The modeling styles were subsequently analyzed using a series of measures for quantifying the process of process modeling (PPM) to validate differences between the three groups and between different tasks.

Our main findings are that three modeling styles can be distinguished in terms of a few simple measures. With these measures, we can characterize (1) modeling with high efficiency, (2) modeling emphasizing a good layout of the model, being created less efficiently, and (3) modeling that is neither very efficient nor very focused on layouting. We found that modelers may change their modeling style subject to modeler- and task-specific characteristics. As modeler-specific characteristics, we could identify modeling speed, the time needed to develop an understanding of the modeling task, and the inherent desire to invest into a good layout of the model. We observed that repairing mistakes as introduced during modeling is a separate issue that correlates with the perceived complexity of the modeling task. Also, we found that modelers who invest into good layout will persist in this intent even when they perceive the modeling task as difficult.

This paper extends the results of [34] in several ways. Most notably, we have reproduced the results of [34] in a new modeling task, thus confirming the existence of three genuinely distinct modeling styles. Further, we develop more refined measures to describe the modeling styles and factors that influence modeling styles. We have aggregated these into a first model explaining process modeling styles and their influence factors.

The remainder of this paper is organized as follows. Section 2 presents related work. Section 3 presents the PPM and how it can be measured. Section 4 develops the setup of our exploratory study based on insights into the PPM gained in earlier studies. The execution of the study is presented in Sect. 5. In Sect. 6, we describe insights into modeling styles gained by data mining; these insights are used in Sect. 7 to develop a number of hypotheses on influence factors on modeling styles. We test the hypotheses in Sect. 8 and compile the results into a model of process modeling styles and their influence factors in Sect. 9. In this section, we also discuss limitations. We conclude and discuss future work in Sect. 10.

2 Related work

Our work is essentially related to model quality frameworks and process model quality (cf. Sect. 2.1), research into the process of modeling (cf. Sect. 2.2), and the process of programming (cf. Sect. 2.3).

2.1 Quality frameworks and process model quality

Different frameworks and guidelines have been developed that define quality aspects in the context of process models. The SEQUAL framework uses semiotic theory for identifying dimensions of process model quality [15], including semantic, syntactic, pragmatic, and other types of issues. The Guidelines of Modeling (GoM) also elaborate on quality

considerations for process models [2] and prescribes principles such as correctness and clarity that should be considered during model creation. The ‘Seven Process Modeling Guidelines’ (7PMG) comprise a set of actions a process modeler may want to undertake to avoid issues with respect to the understandability of a process model and its logical correctness [22]. The 7PMG accumulate the insights from various empirical studies on the quality of process models [23,25]. Other studies have proposed, applied, and validated alternative, yet similar metrics to assess the quality of the model artifact itself, e.g., [1,5,10,40]. Besides, pragmatic quality, i.e., understandability, has been investigated based on insights from cognitive psychology, e.g., [51,53,54].

All of the mentioned works have in common that they start from an analysis or reflection on the quality of *the model itself*. Through the focus on both desirable and actual properties of the process model, prescriptive measures for the process modeler are derived. In our work, we aim to extend this perspective by including the viewpoint of the *modeling act itself*, i.e., the PPM. The idea is that by understanding the PPM, it will become possible to develop insights why process models lack the desired level of quality.

2.2 Process of modeling

Research into the process of modeling typically focuses on the interaction between different parties. In a classical setting, a system analyst interacts with a domain expert through a structured discussion, covering the stages of elicitation, modeling, verification, and validation [8,14]. The procedure of developing process models in a team is analyzed in [39] and characterized as a negotiation process. Interpretation tasks and classification tasks are identified on the semantic level of modeling. Participative modeling is discussed in [44].

These works build on the observation of modeling practice and distill normative procedures for steering the process of modeling toward a good completion. The focus is on the effective interaction between the involved stakeholders. Our work is complimentary to this perspective through its focus on the *formalization* part of the modeling process. In other words, we are interested in the modeler’s interactions with the modeling environment when creating the formal business process model.

2.3 Process of programming

A stream of research related to the PPM is conducted in the realm of understanding the process of computer programming, e.g., [4,11,19,46]. The development of a program can be considered a problem-solving task with an external representation, i.e., the source code, being a central artifact of the process [3]. Also, the process of software design can be seen as highly iterative, interleaved, and loosely

ordered [12]. Researchers have identified three phases of comprehension, decomposition, and solution specification in this process [3,11,46].

These works support the idea that an insight into the PPM is valuable. We adopt the notion of process modeling as a problem-solving task that is executed where an artifact, i.e., the process model, is created. Indeed, we have already observed phases similar to the ones in the programming process [37]. At the same time, it is still relevant to study the specific act of process modeling, instead of relying on existing insights from the area of programming. After all, writing a program in textual form and developing a process model using a graphical notation are different matters. In addition, process models—especially when they serve as a means for communication—should be understood not only by developers, as is the case in programming, but also by various stakeholders with varying backgrounds.

3 Backgrounds

We aim at establishing the existence of different styles in creating a process model and investigating the factors that influence the selection of a style. This section describes the necessary backgrounds in terms of cognitive foundations of the PPM (cf. Sect. 3.1) as well as its phases (cf. Sect. 3.2). Moreover, it explains both how the PPM can be captured (cf. Sect. 3.3) and be quantified using a series of measures (cf. Sect. 3.4).

3.1 Cognitive foundations of the process of process modeling

When creating a process model, the human brain as a “truly generic problem solver” [47] comes into play. Three different problem-solving “programs” or “processes” are known from cognitive psychology: search, recognition, and inference [16]. Search and recognition identify information of rather low complexity, i.e., locating an object or the recognition of patterns. Most conceptual models go well beyond the complexity that can be handled by search and recognition and require “true” problem solving in terms of inference. Cognitive psychology differentiates between *working memory* that contains information that is currently being processed and *long-term memory* in which information is stored for a long period of time [31]. Most severe, and thus of high relevance, are the limitations of the working memory. As reported in [24], the working memory cannot hold more than 7 ± 2 items at the same time, referred to as chunks. Due to these limits, problem-solving tasks are typically not solved as a whole, but rather broken down into smaller parts and addressed chunk-wise. How problem-solving tasks are

addressed, thus, depends on the problem-solving capacity of the problem solver.

By suitable organization of information, the span of working memory can be increased [9]. 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 “chunks” [7,9,28]. As modeling is related to problem solving [7], 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 [41].

Moreover, also the problem-solving task itself influences the development of the solution (cf. Cognitive Load Theory [45]). This influence is described as *cognitive load* for the person solving the task. The cognitive load of a task is determined by its *intrinsic load*, i.e., the inherent difficulty associated with a problem-solving task and its *extraneous load*, i.e., generated by the manner the task is presented [31]. The amount of working memory used to solve a task is referred to as *mental effort* [31]. As soon as a mental task, e.g., creating a process model, overstrains the capacity of the modeler’s working memory, errors are likely to occur [45] and may affect the modeler’s style.

3.2 The process of process modeling

The PPM refers to the formalization of a business process from a domain description. During the formalization phase process, modelers are creating a syntactically correct process model reflecting a given domain description by interacting with the process modeling environment [14]. This modeling process can be described as an iterative and highly flexible process [7,27], dependent on the individual modeler and the modeling task at hand [50]. At an operational level, the modeler’s interactions with the modeling environment typically consist of a cycle of three successive phases, (1) comprehension (i.e., the modeler forms a mental model of domain behavior), (2) modeling (i.e., the modeler maps the mental model to modeling constructs), and (3) reconciliation (i.e., the modeler reorganizes the process model) [37,43].

3.2.1 Comprehension

According to [29], when facing a task, the problem solver first formulates a mental representation of the problem and then uses it for reasoning about the solution and the selection of problem-solving methods. In process modeling, the task is to create a model which represents the behavior of a

domain. The process of forming mental models and applying methods for achieving the task is not done in one step for the entire problem. Rather, due to the limited capacity of working memory, the problem is broken into pieces that are addressed sequentially, chunk by chunk [37,43].

3.2.2 Modeling

Using the problem and solution developed during the previous comprehension phase, a modeler materializes the solution by creating or changing a process model [37,43]. The modeler’s utilization of working memory influences the number of executed modeling steps before the modeler is forced to revisit the problem for acquiring more information [37].

3.2.3 Reconciliation

After modeling, modelers typically reorganize the process model (e.g., rename activities) and utilize the process model’s *secondary notation* (e.g., the layout, typographic cues) to enhance the process model’s understandability [21,32]. However, the amount of reconciliation in a PPM instance is influenced by a modeler’s ability of placing elements correctly when creating them, alleviating the need for additional layouting [37].

3.3 Capturing events of the process of process modeling

To investigate the PPM, actions taken during modeling have to be recorded and mapped to the phases described above. Process modeling with dedicated tools consists of adding nodes and edges to the process model, naming or renaming activities, and adding conditions to edges. In addition, 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 routing of edges (cf. [37]). To capture modeling activities and obtain insights on how process models are created, we instrument a basic process modeling editor in the following way: each user interaction is captured together with the corresponding time stamp in an event log, thereby describing the process model creation step by step. By capturing all interactions with the modeling environment, we are able to *replay* a recorded modeling process at any point in time without interfering with the modeler or her problem-solving efforts. Cheetah Experimental Platform (CEP) [35] provides the features for model editing, event recording, and replay.

3.4 Quantifying the process of process modeling

Having recorded actions taken during model creation, the resulting log of modeling events allows for a quantitative analysis of PPM instances. As described in [37], compre-

hension (C), modeling (M), and reconciliation (R) phases are identified by grouping events. The PPM instance can then be divided into *modeling iterations*. One iteration is assumed to comprise a comprehension (C), modeling (M), and reconciliation (R) phase in this order. The iterations of a modeling process are identified by aligning its phases to the CMR-pattern. If a phase of this pattern is not present, the respective phase is skipped and the process is considered to continue with the next phase of the pattern. We use five measures to quantify the PPM.

3.4.1 Number of PPM iterations

This measure counts the modeling iterations in a PPM instance reflecting how often a modeler had to interrupt modeling for comprehension or reconciliation.

3.4.2 Iteration chunk size

Modelers can be assumed to conduct modeling in chunks of different sizes. The *iteration chunk size* is the average number of create and delete operations per PPM iteration and reflects the ability to model large parts of a model without the need to comprehend or reconcile.

3.4.3 Share of comprehension

In comprehension phases, a mental model of the problem and a corresponding solution is developed. Differences in the time spent on comprehension can be expected to influence modeling styles and the modeling result. We quantify this aspect as the ratio of the average length of a comprehension phase in a process to the average length of an iteration. We neglect the initial comprehension phase to avoid a bias from the time needed for reading the task description.

3.4.4 Reconciliation breaks

A steady process of modeling should be a sequence of iterations of the CMR-pattern. Reconciliation can sometimes be skipped if the modeler places all model elements directly at the right spot. However, we may observe iterations of CR-patterns, i.e., an iteration without a modeling phase, where a modeler interrupts the common flow of modeling for further reconciliation. We quantified this aspect by the relative share of iterations that comprise unexpected reconciliation (without modeling).

3.4.5 Delete iterations

From time to time, modelers are required to remove content from the process model. This might happen when modelers identify errors in the model that are resolved by removing

modeling constructs and implementing the desired functionality. This measure describes the relative number of iterations in a PPM instance that contains delete operations to the total number of iterations in that PPM instance.

4 Building a model for understanding modeling styles

When comparing the PPM instances of different modelers, who were creating a formal process model from the same informal process description, we observed that groups of PPM instances exposed similar characteristics and that different modelers exhibit genuinely distinct modeling styles [37]. However, it remained unclear what modeling styles can be found in practice, and more importantly, how the selection of a particular style is influenced.

Given the lack of an in-depth understanding of both the modeling styles and the influencing variables, we follow an *explorative* approach. Rather than addressing a defined set of hypotheses, our aim is to investigate whether distinct modeling styles exist, to explore what distinguishes them from one another, and to discover relations between them. The findings may form the basis for a model that ties together influence factors and modeling styles.

Building on the backgrounds introduced in Sect. 3, we summarize the most important aspects influencing process model creation as follows:

1. *Task-intrinsic characteristics*, the factual properties of the process that shall be modeled,
2. *Task-extraneous characteristics*, the way the factual properties of the process are presented and properties of the modeling tool and notation,
3. *Modeler-specific characteristics*, the modeler's cognitive abilities, but also preferences in terms of modeling and tool usage.

We discuss the first two categories in Sect. 4.1 and the modeler-specific characteristics in Sect. 4.2. In Sect. 4.3, we will then derive a setup that is suitable for building a model for understanding modeling styles.

4.1 Task-intrinsic and task-extraneous characteristics

Creating a formal process model from a given process description is influenced by characteristics of the concrete task. Section 3 discussed that the cognitive load of a task is determined by its *intrinsic load* and its *extraneous load* [31].

In our context, *intrinsic load* is determined by the model to be created. It can be characterized by the size (e.g., number of activities or control flow constructs) and complexity of the model structure and constructs. Yet, it is independent of the presentation of the modeling task to the modeler.

Extraneous load, by contrast, concerns the presentation of the task to the modeler. For instance, in [36], the modeler's performance was significantly influenced when restructuring the informal task description, even though no changes were made to the intrinsic load of the modeling assignment. If the cognitive load exceeds the modeler's working memory capacity, errors are likely to occur [45] and may affect the modeler's style. The extraneous load is part of the task-extraneous properties, which also include properties of the modeling tool and notation, which constrain the modeling process.

4.2 Modeler-specific characteristics

Modeler-specific characteristics consider cognitive characteristics and model interface preferences. The former are related to the capacity of the working memory, which can be expected to affect the cognitive load imposed by the task. Also, this category includes the modeler's expertise, e.g., the modeler's experience with the modeling notation, the modeling domain [31], or the modeling tool. In addition to cognitive and task-specific characteristics, distinct preferences of a modeler on how to create a model in terms of layouting and tool usage play a role. For instance, [37] describes on the one hand modelers who carefully place and arrange nodes and edges of a model to achieve an appealing layout. On the other hand, the study reports on modelers who carelessly put nodes on the canvas and draw straight connecting edges, mostly not influencing the visual appearance of the resulting process model. It was also recognized that several modelers seemed to dislike activities disappearing from sight. More specifically, when a model is about to get larger than what can be shown on the display, many modelers spend much time on reconciliation to free up space on the visible canvas and prevent model elements from disappearing. Most notably, reconciliation to free up space on the canvas seems to be independent of whether the modeler is interested in an appealing layout or not.

4.3 Designing an exploratory study for building a model

As outlined above, we believe that several factors influence the modeling style, namely the intrinsic and the extraneous load of a modeling task as well as modeler-specific characteristics. When designing the setup for the modeling sessions, we have to assume that these factors have mutually independent influences on the modeling styles. For a first exploratory study, we control two factors (task-intrinsic load and modeler-specific characteristics) and keep the remaining factor (task-extraneous characteristics) constant.

1. We control modeler-specific characteristics by conducting the exploratory study with a large number of partici-

pants (>100). Hence, it is reasonable to assume that the subjects are representative of the general population in terms of cognitive characteristics. The subjects' expertise (both modeling and domain knowledge) turned out to be quite uniform (cf. Sect. 5).

2. We control task-intrinsic load by giving each participant modeling tasks of two different processes in the form of a textual description. These processes are to be sufficiently distinct to ensure that the influence of task-specific characteristics materializes.
3. We keep the task-extraneous characteristics constant. Textual descriptions for both modeling tasks are given in the same style with respect to the process to be modeled. Also, the influence of tool and notation are kept constant by letting all participants model the process in the same editor featuring limited BPMN syntax and modeling functionality.

5 Data collection

Section 5.1 presents the planning of the exploratory study to investigate modeling styles. The execution of the study is described in Sect. 5.2.

5.1 Definition and planning

This section contains requirements regarding the subjects of the exploratory study as well as information on the developed materials and the data to be collected in this exploratory study.

5.1.1 Subjects

When investigating the PPM, one of the key challenges is to balance the difficulty of the modeling task to be executed with the knowledge of the participants. If the modeling task is too complicated, hardly any conclusions on modeling style can be drawn since most modelers would experience serious difficulties. By contrast, if the task is too easy, hardly any differences can be observed since challenging situations are a key ingredient of problem solving. Hence, the targeted subjects should be moderately familiar with business process management and imperative process modeling notations to avoid problems with the modeling notation, but still encounter some challenges when creating the process models of the given difficulty.

5.1.2 Objects

The study was designed to collect PPM instances of students with moderate process modeling skills creating a formal process model in BPMN from an informal description. Each student was asked to create two models. To control

task-intrinsic load and observe task-specific characteristics, the objects have to be sufficiently different. We accommodated for this aspect by considering processes of different domains, sizes, and structures.

The first modeling assignment is a process describing the activities a pilot has to execute prior to taking off with an aircraft. The process model consists of 12 activities and contains basic control flow patterns, such as sequence, parallel split, synchronization, exclusive choice, and simple merge [49].

The second process model to be created describes the process followed by the scouting department of a National Football League (NFL) team to acquire new players through the so-called NFL Draft. The process model was considerably smaller, consisting of eight activities, still incorporating the basic control flow patterns of sequence, parallel split, synchronization, exclusive choice, simple merge, and structured loop [49].¹

5.1.3 Response variables

To collect PPM instances of all participants, all details of the modeling process have been recorded. Further, we measured the modelers' perceived mental effort for each modeling task since mental effort provides a fine-grained measure for the modeler's performance [52]. The collected PPM instances are analyzed with data mining techniques to identify modeling styles (Sect. 6) and to reveal relevant response variables that govern modeling styles and their interplay with influence factors (Sect. 7).

5.1.4 Instrumentation and data collection

CEP was utilized for recording and analyzing PPM instances. CEP provides support for conducting experiments and case studies by providing means to define an experimental workflow for each participant. This reduces the risk of students accidentally deviating from the intended research design [35]. To limit extraneous cognitive load by complicated tools or notations [7], we used a subset of BPMN. 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. Based on a pretest at the University of Innsbruck, minor updates have been applied to CEP's functionality and the task descriptions.

5.2 Performing the exploratory study

This section describes the execution of the exploratory study.

5.2.1 Execution of exploratory study

The modeling sessions were conducted in November 2010 with students of a graduate course on Business Process Management at Eindhoven University of Technology and in January 2011 with 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 "Pre-Flight" process. After completing the first modeling task, students were asked to create the process model for the "NFL Draft" process. This was done by 102 students in Eindhoven and 13 students in Berlin. By conducting the modeling sessions during class and closely monitoring the students, we mitigated the risk of falsely identifying comprehension phases due to external distractions. Each modeling task was followed by a self-rating of the mental effort required for completing the modeling task on a seven-point Likert scale ranging from *Very Low* over *Medium* to *Very High*. Self-rating scales for mental effort have been shown to reliably measure mental effort and are thus widely adopted [30]. Students were not instructed about the research questions to be answered in the exploratory study prior to performing the modeling task. No time restrictions were imposed on the students. Participation was voluntary; data collection was performed anonymously.

5.2.2 Data validation

Similar to [21], we screened the subjects for familiarity with BPMN by asking them whether they would consider themselves to be very familiar with BPMN, using a Likert scale with values ranging from *Strongly disagree* (1) over *Neutral* (4) to *Strongly agree* (7). The familiarity with BPMN was slightly below *Neutral* ($M = 3.47$, $SD = 1.45$). For confidence in understanding BPMN models, the students reported a mean value slightly above *Neutral* ($M = 4.05$, $SD = 1.49$). Finally, for perceived competence in creating BPMN models, a mean value slightly below *Neutral* was reported ($M = 3.65$, $SD = 1.41$). We conclude that the subjects constituted a rather homogeneous group, reporting a familiarity close to average. Thus, the participants are well suited for investigating their modeling style when translating an informal description into a formal BPMN model.

Similarly, participants were indicating their familiarity with Pre-Flight processes and the NFL on the same Likert scale (Pre-Flight: $M = 2.40$, $SD = 1.27$; NFL Draft: $M = 3.45$, $SD = 1.91$). For the NFL Draft modeling task, modelers indicated a slightly higher domain knowledge. Still, for both tasks, the average familiarity is below *Neutral*, indicating that

¹ Material download: <http://bpm.q-e.at/experiment/ModelingStyles>.

Table 1 Classification of CEP's user interactions

Interaction	Classification	Interaction	Classification
CREATE NODE	Adding	RENAME ACTIVITY	Reconciliation
DELETE NODE	Deleting	UPDATE CONDITION	Reconciliation
CREATE EDGE	Adding	MOVE NODE	Reconciliation
DELETE EDGE	Deleting	MOVE EDGE LABEL	Reconciliation
RECONNECT	Adding/deleting	MODIFY EDGE	Reconciliation
EDGE		BENDPOINT	

modelers could hardly rely on prior domain knowledge for performing the task.

When investigating mental effort data, we observed a lower mental effort for the second modeling task (Pre-Flight: $M = 4.01$, $SD = 1.047$, NFL Draft: $M = 3.77$, $SD = 0.974$). The differences turned out to be statistically significant (Wilcoxon signed-rank test, $Z = -2.54$, $p = 0.011$), indicating that modelers perceived the second modeling task to be easier than the first one. This is consistent with the smaller size of the second modeling task. These results indicate that the two processes to be modeled are indeed different and, thus, allow for controlling task-intrinsic load (cf. Sect. 4).

6 Clustering

To investigate the existence of different modeling styles, we apply cluster analysis to the collected PPM instances and analyze whether groups of PPM instances exhibiting similar characteristics can be identified. The applied clustering procedure is described in Sects. 6.1 and 6.2. The identified clusters are then visualized and analyzed to determine whether they indeed represent different modeling styles. To check whether the identified modeling styles persist over tasks with different characteristics, clustering is applied to two tasks with different characteristics. Results of clustering the Pre-Flight task are discussed in Sect. 6.3, while the clustering results of the NFL Draft task are discussed in Sect. 6.4.

6.1 PPM profile for clustering

First and foremost, we need a representation suited for clustering for all collected PPM instances. Based on our previous experience, we decided to focus on four aspects: the addition of content, the removal of content, reconciliation of the model, and comprehension time, i.e., the time when the modeler does not work on the process model. To also reflect that modeling is a time-dependent process, we do not just look at the total amount of modeling actions and comprehension, but on their *distribution* over time. We sampled

every process into segments of 10 s length. For each segment, we compute its *profile* (a, d, r, c), i.e., the numbers a , d , and r of add, delete, and reconciliation events, and the time c spent on comprehension. The *profile* of one PPM is the sequence $(a_1, d_1, r_1, c_1)(a_2, d_2, r_2, c_2) \dots$ of its segments' profiles. The a , d , and r are obtained per segment by classifying each event according to Table 1. Adding a condition to an edge was considered being part of creating an edge. The comprehension time c was computed as follows. First, events were grouped to intervals, i.e., sequence of events where two consecutive events are ≤ 1 s apart. Second, the interval duration was calculated as the time difference between its first and its last event (intervals of one activity got a duration of 1 s). Comprehension time c is calculated as the length of the segment (10 s) minus the duration of all intervals in the segment. For example, if the modeler moved activity A after 3 s, activity B after 3.5 s, and activity C after 4.2 s the comprehension time would be 8.8 s. To give all PPM profiles equal length, we normalized profiles by extending them with segments of no interaction.

6.2 Performing the clustering

The PPM profiles were exported from CEP [35] and subsequently clustered using Weka.² The K-Means algorithm [18] utilizing an Euclidean distance measure was chosen for clustering as it constitutes a well-known means for cluster analysis. As K-Means might converge in a local minimum [13], the obtained clustering has to be validated. If the identified clusters exhibit significant differences with regard to the measures described in Sect. 3, we conclude that different modeling styles were identified. K-Means requires the number of clusters to be known a priori. Thus, we started with two expected clusters, gradually increasing the number of expected clusters. Similarly, several different values for the seed of the clustering were investigated.

6.3 Clustering of Pre-Flight task

For the first modeling task, we start by presenting the result of clustering. Then, we illustrate the clusters visually, conduct

² <http://www.cs.waikato.ac.nz/ml/weka>.

a statistical validation of the clustering, interpret their differences, and report on findings from replaying representative PPM instances.

6.3.1 Result of clustering

Setting the number of expected clusters to 2 resulted in only one major cluster. For a value of 3, we obtained two major clusters and one cluster of 2 PPM instances. Most promising results were achieved with a number of expected clusters of 4 and a seed of 10, returning three major clusters and one small cluster of 2 PPM instances. We considered these three major clusters for further analysis; increasing the number of

expected clusters only generated further small clusters. The three major clusters comprise 42, 22, and 49 instances, called C1, C2, and C3 in the sequel.

6.3.2 Cluster visualization

In order to visualize the obtained clusters, we calculate the average number of adding, of deleting, and of reconciliation operations per segment for each cluster. To have a smooth representation, we also calculate the moving average of six segments, presented in Figs. 1, 2, and 3 for clusters C1, C2, and C3. The horizontal axis denotes the segments derived by sampling the PPM instances. The

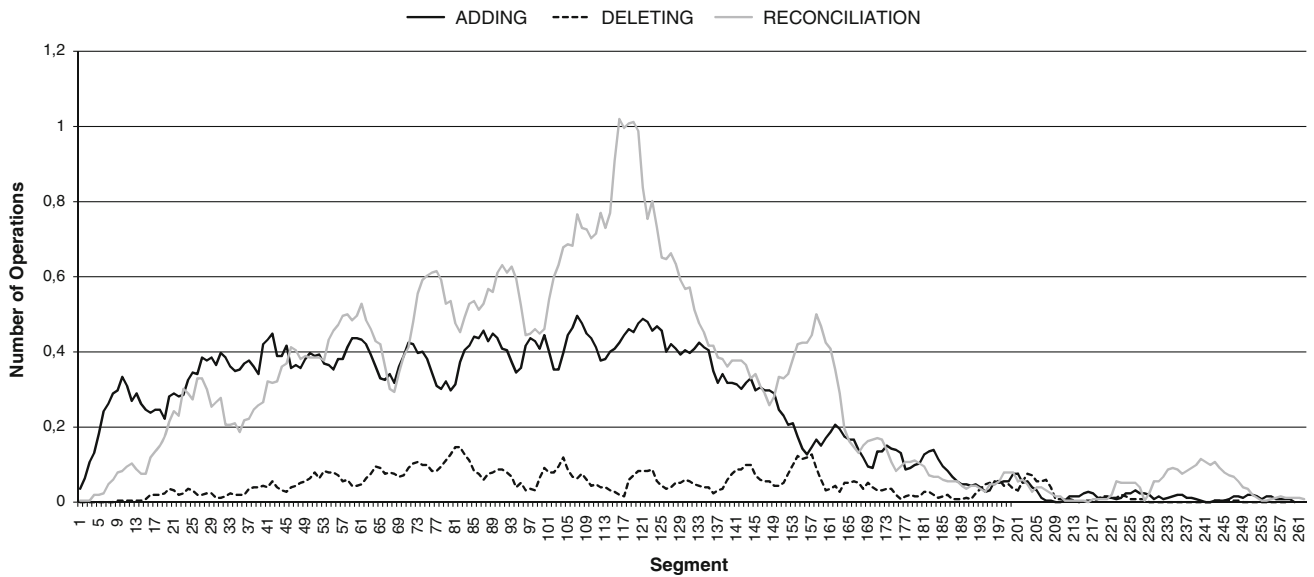


Fig. 1 Cluster C1 Pre-Flight task

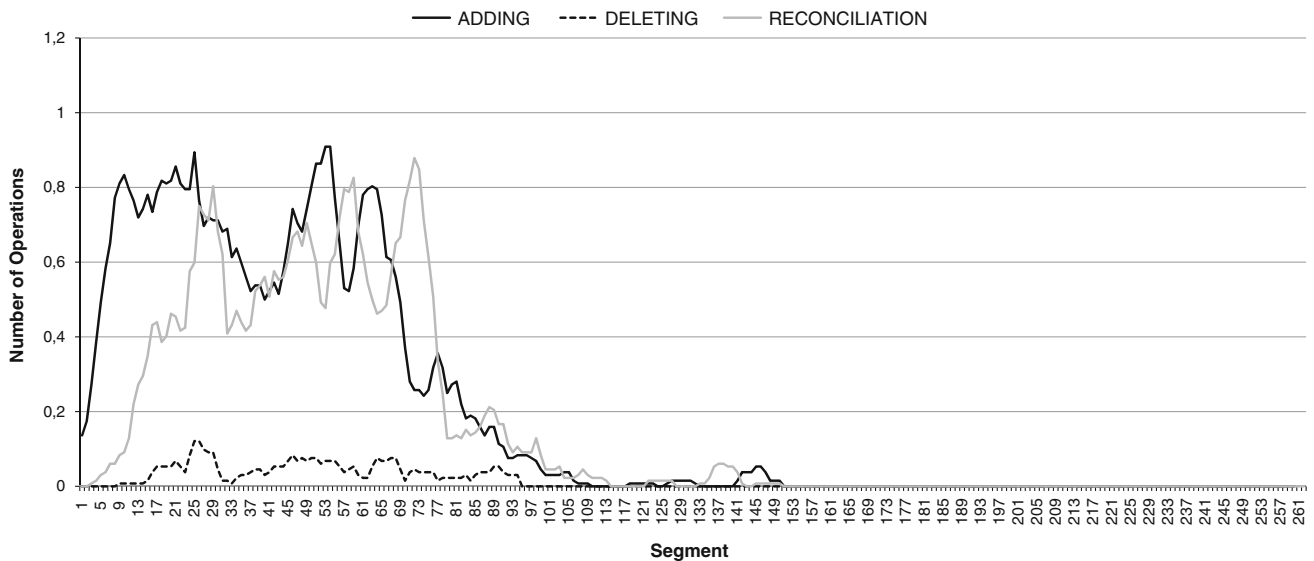


Fig. 2 Cluster C2 Pre-Flight task

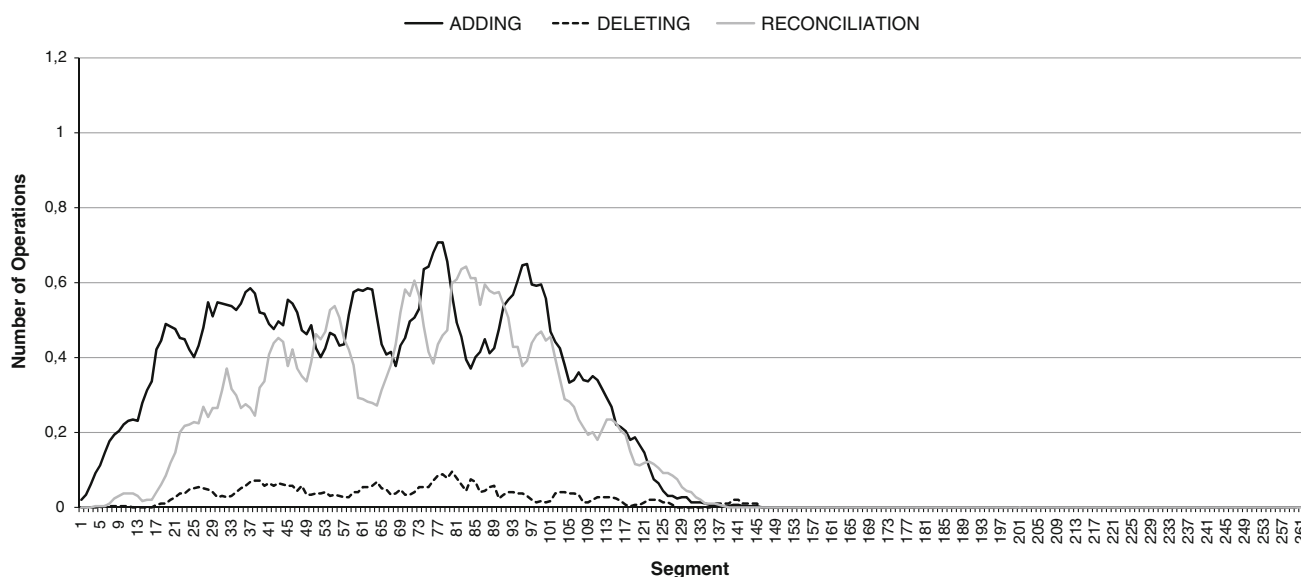


Fig. 3 Cluster C3 Pre-Flight task

vertical axis indicates the average number of operations that were performed per segment. For example, a value of 0.8 for segment 9 (cf. Fig. 2) indicates that all modelers in C2 averaged 0.8 adding operations within this 10 s segment.

C1 (cf. Fig. 1) is characterized by long PPM instances, as the first time the adding series reaches 0 is after about 205 segments. Additionally, the delete series indicates more delete operations compared to the other clusters. Several fairly large spikes of reconciliation activity can be observed, the most prominent one after about 117 segments.

C2, as illustrated in Fig. 2, is characterized by a fast start as a peak in adding activity is reached after 13 segments. In general, the adding series is most of the time between 0.5 and 0.9 operations, which is higher compared to the other two clusters. The fast modeling behavior results in short PPM instances as the adding series is 0 for the first timer after about 110 segments.

At first sight, C3 (cf. Fig. 3) seems to be somewhere between C1 and C2. The adding curve is mostly situated between 0.4 and 0.7, a little lower than for C2, but still higher compared to C1. Similar values can be observed for the reconciliation curve. The deleting curve remains below 0.1. The duration of the PPM instances is also between the duration of C1 and C2 as the adding series is 0 for the first time after about 137 segments.

6.3.3 Cluster validation

Next, we validated the clusters by testing whether they indeed expose significant differences.

Table 2 presents general statistics on the number of adding operations, the number of deleting operations, and the num-

Table 2 Statistics per cluster Pre-Flight task

Measure	C1	C2	C3
No. of instances	42	22	49
Avg. no. of adding operations	61.36	52.91	52.57
Avg. no. of deleting operations	10.81	3.91	4.55
Avg. no. of reconciliation operations	76.26	42.00	39.27

ber of reconciliation operations for each cluster. Modelers in C1 carried out more add and delete operations and, most notable, almost twice as many reconciliation operations compared to C2 and C3. The numbers for C2 and C3 appear to be similar.

We conducted the statistical analysis as follows. If the data were normally distributed and homogeneity of variances was given, we used one-way ANOVA to test for differences between the groups. Pairwise comparisons were done using the Bonferroni post hoc test. Note that the Bonferroni post hoc test uses an adapted significance level, so that p values < 0.05 are considered to be significant; i.e., there is no need to divide the significance level by the number of groups. In case a normal distribution or homogeneity of variance was not given, a nonparametric alternative to ANOVA, i.e., Kruskal–Wallis, was utilized to test for differences between the groups. Pairwise comparisons were done using the t test for (un)equal variances (depending on the data) if a normal distribution was given. If no normal distribution could be identified, the Mann–Whitney test was utilized. In either case, i.e., t test or Mann–Whitney test, the Bonferroni correction was applied; i.e., the significance level was divided by the number of clusters.

Table 3 Significant differences for statistics Pre-Flight task

Statistic	All groups		Pairwise comparison		
			1–2	1–3	2–3
No. of adding operations	Sig.	0.000 ^a	0.003 ^a	0.000 ^a	
	Test	One-way ANOVA	Bonferroni post hoc		
No. of deleting operations	Sig.	0.000 ^a	0.000 ^b	0.000 ^b	
	Test	Kruskall–Wallis	Mann–Whitney		
No. of reconciliation operations	Sig.	0.000 ^a	0.000 ^b	0.000 ^b	
	Test	Kruskall–Wallis	(un)equal variances <i>t</i> test		

^a $p < 0.05$; ^b $p < 0.05/3$

Table 4 Measures per cluster Pre-Flight task

Measure	C1	C2	C3
Avg. no. of PPM iterations	21.50	12.32	14.69
Avg. iteration chunk size	3.66	5.28	4.24
Avg. share of comprehension	49.88	39.28	45.02
Avg. reconciliation breaks	21.37	18.14	13.85
Avg. delete iterations	17.06	10.07	10.83

As shown in Table 3, we observe significant differences between C1 and C2 and C1 and C3, but not between C2 and C3. Only significant differences are reported in this and all following tables.

To further distill the properties of the three clusters, we calculated the measures described in Sect. 3.4 for each PPM instance. Table 4 provides an overview of the obtained average values. As indicated in Fig. 1, C1 constitutes the highest number of PPM iterations. Tightly connected to this observation is the average iteration chunk size. Modelers in C2 added by far the most content per iteration to the process model. Also, the number of iterations containing delete iterations is higher for C1 than for the other clusters. The amount of time

spent on comprehending the task description and developing the plan on how to incorporate them into the process model seems to be far larger for C1 compared to C2, which has the lowest share of comprehension, but also larger compared to C3. When considering reconciliation breaks C3 sets itself apart, posting the lowest number of reconciliation breaks. C1 has the highest number of reconciliation breaks.

The results of an statistical analysis of the differences between the groups are presented in Table 5. In contrast to the statistics presented in Table 3, we were able to identify significant differences between C2 and C3.

6.3.4 Interpretation of clusters

Our results clearly indicate that C1 can be distinguished from C2 and C3. Modelers in C1 had rather long PPM instances (cf. number of PPM iterations), spent more time on comprehension compared to C2, started rather slowly (cf. number of adding operations and chunk size), and showed a high amount of delete and reconciliation operations. This suggest that modelers in C1 were not as goal-oriented as their colleagues in other clusters, since they spent a great amount of time on comprehension, added more modeling elements which were

Table 5 Significant differences for measures Pre-Flight task

Measure	All groups		Pairwise comparison		
			1–2	1–3	2–3
Iteration chunk size	Sig.	0.000 ^a	0.000 ^b	0.000 ^b	0.007 ^b
	Test	Kruskall–Wallis	(un)equal variances <i>t</i> test		
No. of PPM iterations	Sig.	0.000 ^a	0.000 ^b	0.000 ^b	0.004 ^b
	Test	Kruskall–Wallis	(un)equal variances <i>t</i> test		
Share of comprehension	Sig.	0.000 ^a	0.000 ^a	0.036 ^a	0.045 ^a
	Test	One-way ANOVA	Bonferroni post hoc		
Delete iterations	Sig.	0.005 ^a	0.026 ^a	0.011 ^a	
	Test	One-way ANOVA	Bonferroni post hoc		
Reconciliation breaks	Sig.	0.005 ^a		0.004 ^a	
	Test	One-way ANOVA	Bonferroni post hoc		

^a $p < 0.05$; ^b $p < 0.05/3$

subsequently removed, and put significantly more effort into improving the visual appearance of the model.

Focusing on C2, we observe a very steep start of the adding curve in Fig. 2, indicating that modelers started creating the process model right away. The measures described in Sect. 3.4 further indicate high chunk sizes, a low number of PPM iterations, and little comprehension time. Thus, modelers of C2 appear to be focused and goal-oriented when creating the model. They are quick in making decisions about how to proceed and only slow down from time to time for some reconciliation.

The PPM instances of C3 are shorter compared to C1 and longer compared to C2. The reconciliation curve is close to the adding curve. Notably, there is no reconciliation spike once the number of adding operations decreases. Albeit close to C2, C3 is characterized by slower and more balanced model creation (larger chunk size, higher number of iterations, more comprehension time). Thus, C3 follows a rather structured approach to modeling.

6.3.5 Analysis of cluster representatives

We gained further insights in the cluster differences by manually comparing representative PPM instances. Clustering with K-Means yields *cluster centroids*, the mean for add, delete, reconciliation, and comprehension over all PPM profiles inside a cluster. For each cluster, we have chosen the PPM instance with the smallest distance to this centroid as a representative and compared them using the replay functionality of CEP [35]. Then, we repeated the procedure with the PPM instances showing the second-smallest distance to the centroids.

The representative for C1 is very volatile in terms of speed and locality of modeling. Adding elements is done in an unsteady way with intermediate layouting, conducted in short phases. The aspect of locality relates primarily to reconciliation. The modeler frequently touched not only the last elements added, but also distant parts of the process model. These observations are largely confirmed by the second representative for C1, which further shows long reconciliation phases to gain space on the canvas.

The representative for C2 follows a rather straight, steady, and quick modeling approach. A group of elements is placed first and only later connected by edges. There is little reconciliation since the layout appears to be considered when adding elements. If applied, reconciliation refers to the last added elements only. The second representative follows the same approach until two-thirds of the model have been created. Then, it deviates with a relayouting the model to gain space on the canvas.

For C3, the representative PPM instance is also steady, but slower than those investigated for C2. At most two elements are added at a time before they get connected. Reconcilia-

tion is done continuously, but restricted locally. Model parts that are distant from the last added elements are not changed. These observations are confirmed by the second representative.

In essence, the representatives of the clusters appear to be distinguished by two aspects in particular, the steadiness of the PPM instance in terms of adding elements, and the characteristics of the reconciliation phases. The latter are characterized by their length and their locality.

6.4 Clustering of NFL Draft task

To test whether the identified clusters persist over different modeling task, we repeated the cluster analysis procedure for the second modeling task.

6.4.1 Result of clustering

Again, we conducted the clustering by gradually increasing the number of expected clusters and investigating different seeds. The most promising results were obtained with a seed of 30 and 5 expected clusters. We obtained three major clusters of 30, 31, and 42 PPM instances. Two smaller clusters, 4 and 8 PPM instances, were not further considered.

6.4.2 Cluster visualization

The cluster visualizations are presented in Figs. 4, 5, and 6, respectively.

Figure 4 pictures Cluster C1, which is characterized by long PPM instances, exhibiting a slow start and a low adding curve. The adding curve is closely followed by a reconciliation curve indicating several spikes of reconciliation and much reconciliation after the adding curve starts to decrease. The deleting curve is generally higher compared to the other clusters.

Cluster C2 (cf. Fig. 5) shows short PPM instances and a high adding curve, showing a decrease after 60 segments before reaching 0 after 77 segments. Also, there is a fast increase right at the beginning of the modeling process. The reconciliation curve follows the adding curve with some additional reconciliation at the end. The deleting curve is rather low.

Cluster C3 (cf. Fig. 6) seems to be situated between cluster C1 and cluster C2. It does not exhibit the fast start of cluster C2, but shares similarities for the deleting curve. The PPM instances in C3 are considerably shorter than those in C1, but not as short as in cluster C2. Modelers in C3 show a rather slow start. After 10 segments, the adding curve is close to 0.2, which is similar to C1, but not to C2. Afterward, C3 outperforms C1 in terms of adding elements to the process model. The reconciliation curve follows the adding curve, not showing any major spikes in reconciliation activity.

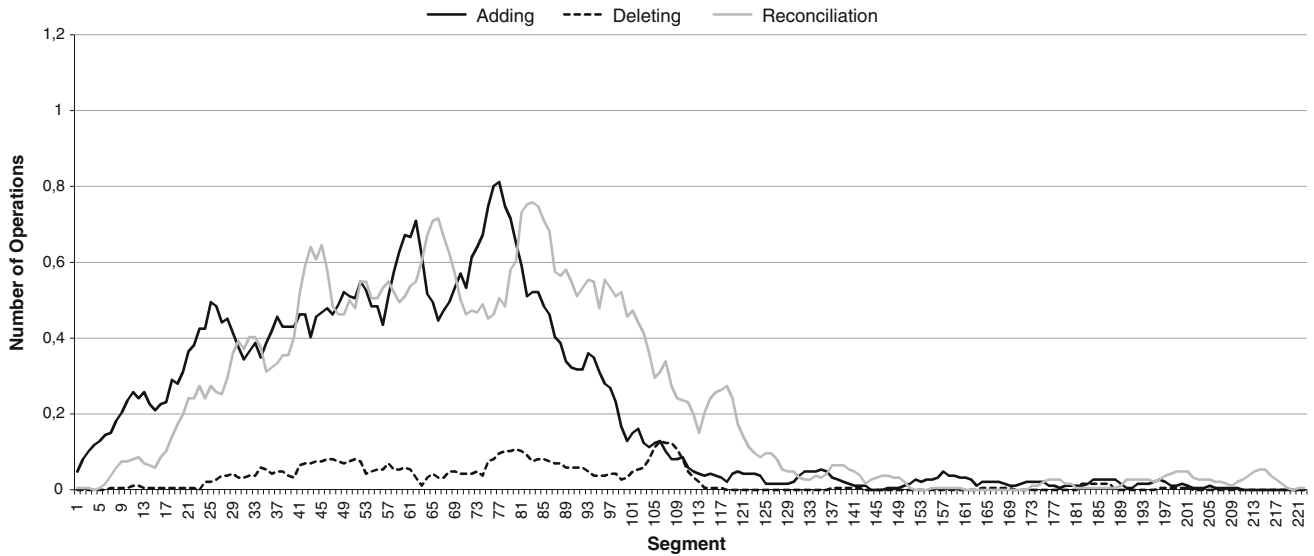


Fig. 4 Cluster C1 NFL Draft task

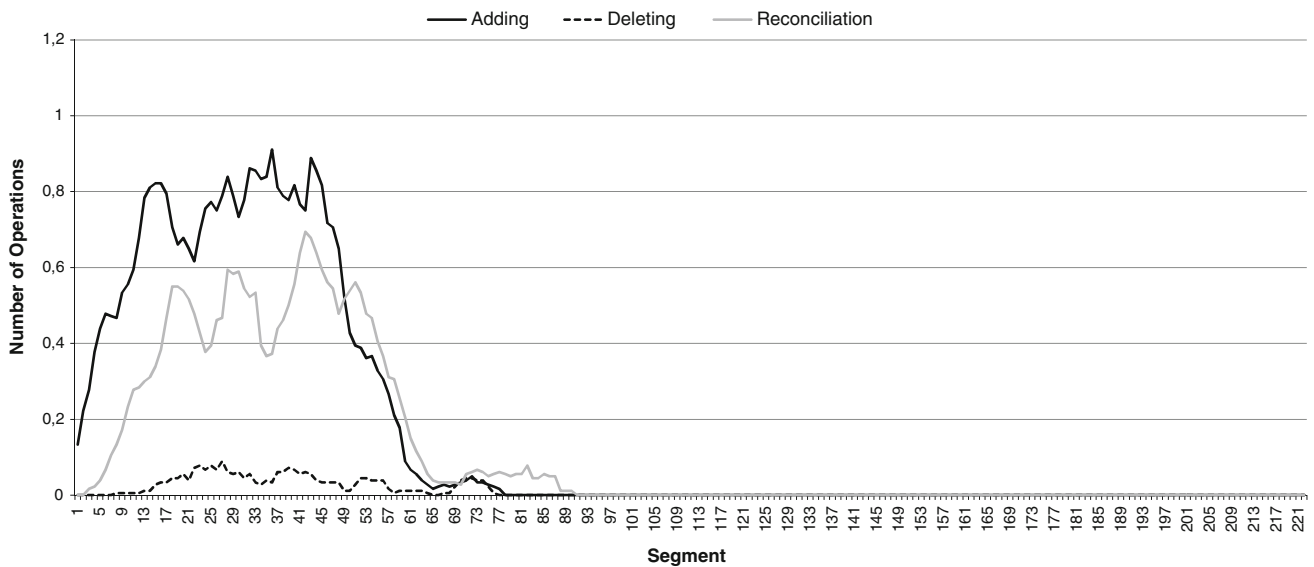


Fig. 5 Cluster C2 NFL Draft task

6.4.3 Cluster validation

The average number of adding operations, the average number of deleting operations, and the average number of reconciliation operations are presented in Table 6. As for the first modeling task, cluster C2 and cluster C3 exhibit similar values, while cluster C1 sets itself apart by the adding, deleting, and reconciliation operations. The statistical analysis illustrated in Table 7 supports this observation by indicating significant differences between C1 and C2 and C1 and C3, but not between C2 and C3.

The average values retrieved by calculating the measures introduced in Sect. 3.4 are listed in Table 8. The three clusters seem to be different when it comes to chunk size and the number of PPM iterations. C2 has the lowest number of PPM iterations and the highest chunk size. C1 is on the opposite side of the spectrum posting the highest number of PPM iterations and the lowest chunk size. The average share of comprehension is similar for all clusters. In terms of reconciliation breaks, C1 has the highest value and C2 posts the lowest value. Delete iterations do not hint at any difference.

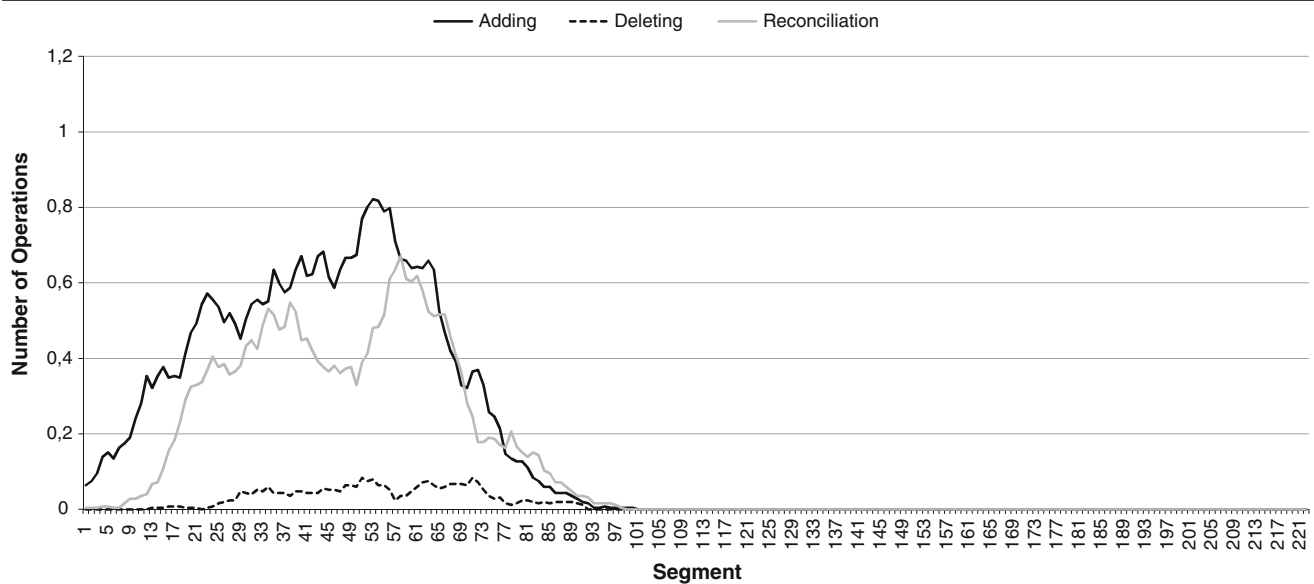


Fig. 6 Cluster C3 NFL Draft task

Table 6 Statistics per cluster NFL Draft task

Measure	C1	C2	C3
No. of instances	31	30	42
Avg. no. of adding operations	45.39	37.67	38.07
Avg. no. of deleting operations	5.61	2.40	3.00
Avg. no. of reconciliation operations	49.94	25.73	27.17

The corresponding statistical analysis is illustrated in Table 9, revealing significant differences between all clusters in terms of the number of PPM iterations. Similarly, chunk size is significantly different when comparing C1 and C2 and when comparing C2 and C3.

6.4.4 Interpretation of clusters

Similar to the clusters identified for the Pre-Flight process, C1 can be distinguished from C2 and C3 (adding operations, reconciliation operations, number of PPM iterations). Again, modelers in C1 seem to be less goal-oriented and spent a lot

of time on reconciliation. However, we could not identify the significant differences in terms of share of comprehension we have observed for the first modeling task.

As for cluster C2, we do obtain significant differences regarding C3 only for iteration chunk size and the number of PPM iterations. This is in line with the first modeling task and suggests that modelers in C2 were very focused on executing the modeling task.

The PPM instances in C3 are longer compared to C2, but not as long as the PPM instance in C1. Modelers in C3 do not share the high number of reconciliation operations and the high number of deleting operations with C1. The overall picture drawn for C3 is similar to the Pre-Flight task. Thus, modelers in C3 can be seen as following a balanced modeling approach that is situated between the other two clusters.

6.4.5 Analysis of cluster representatives

Analyzing the representative PPM instance for C1 showed that it is structured by phases in which a certain model part is

Table 7 Significant differences for statistics NFL Draft task

Statistic	All groups		Pairwise comparison		
			1–2	1–3	2–3
No. of adding operations	Sig.	0.000 ^a	0.000 ^b	0.000 ^b	
	Test	Kruskall–Wallis	Mann–Whitney		
No. of deleting operations	Sig.	0.003 ^a	0.002 ^b	0.005 ^b	
	Test	Kruskall–Wallis	Mann–Whitney		
No. of reconciliation operations	Sig.	0.000 ^a	0.000 ^a	0.000 ^a	
	Test	One-way ANOVA	Bonferroni post hoc		

^a $p < 0.05$; ^b $p < 0.05/3$

Table 8 Measures per cluster NFL Draft task

Measure	C1	C2	C3
Avg. no. of PPM iterations	14.61	8.57	10.79
Avg. iteration chunk size	4.08	5.49	4.46
Avg. share of comprehension	39.74	37.02	42.61
Avg. reconciliation breaks	24.29	13.66	19.13
Avg. delete iterations	12.25	11.40	9.83

added and phases in which parts of a model are reconciled. We observed long phases of layouting that mainly relate to edges. Also, at the end, the model is refactored and layouting is improved. Long adding and reconciliation phases are also visible in the second representative.

The representative for C2 showed a very quick model creation. Also, the process was steady and the rate of adding elements appears to be constant. The PPM instance features only sparse reconciliation. Reconciliation seems to be avoided by considering the model layout when adding an element. If applied, layouting focuses on the elements last added. The second representative for C2 shows very similar characteristics. The only difference is that large sets of elements are added before they get connected.

For C3, the representative PPM instance follows a steady approach, but slower than the one for cluster C2. Also, reconciliation is more prominent than for C2, whereas the reconciliation phases are shorter than observed for C1. Also, reconciliation relates to a rather large area of the canvas. The second representative follows the same approach.

These observations are largely in line with those obtained for cluster representatives for the Pre-Flight process. Again, the locality of operations appears to be important.

In sum, we were able to identify three significantly different clusters representing different modeling styles for each modeling task. Further, the cluster characteristics were similar in terms of number of adding operations, number of deleting operations, and the number of reconciliation operations for the two modeling tasks. Differences among the clusters in the number of iterations and chunk size were consistent over both modeling tasks.

Table 9 Significant differences for measures NFL Draft task

Measure	All groups	Pairwise comparison		
		1–2	1–3	2–3
Iteration chunk size	Sig.	0.003 ^a	0.000 ^b	0.012 ^b
	Test	Kruskall–Wallis	(un)equal variances <i>t</i> test	
No. of PPM iterations	Sig.	0.000 ^a	0.000 ^b	0.001 ^b
	Test	Kruskall–Wallis	(un)equal variances <i>t</i> test	

^a $p < 0.05$; ^b $p < 0.05/3$

7 Identification of variables/generation of hypotheses

In this section, we pick up the observations made during the analysis presented in the previous section to further characterize the three different modeling styles. Some of our observations are already covered by the existing measures. For instance, we observed modelers who were considerably faster in adding elements to the process model than others, which relates to measuring the iteration chunk size since it reflects the number of added elements per PPM iteration. Other observations, in turn, point to potential additional factors characterizing modeling styles.

Below, we present six measures to further discriminate modeling styles on a statistical basis. They are explicitly derived from the reported observations and complement the set of measures needed to characterize modeling styles.

Adding rate. Our analysis showed that clusters deviate from each other in the number of adding operations, see Tables 4 and 8. Also, steepness of the curves for adding operations in relation to PPM segments (Figs. 1, 2, 3, 4, 5, 6) is different for the clusters. Thus, to consider differences between modelers in the speed of adding elements to the canvas, we define the *adding rate*. It is calculated by counting the number of adding operations within modeling phases, i.e., Create Node, Create Edge, and dividing it by the total duration of modeling phases in seconds within a PPM instance.

Avg. iteration duration. When replaying the PPM instances, we observed differences of modelers in terms of modeling speed. To further relate the modeling style to the actual time spent, and characterize quick from slow modeling, we consider the *average iteration duration*. It indicates how long an average PPM iteration takes. Modelers largely ignoring reconciliation phases or modelers who are particularly fast in adding elements should have shorter modeling phases. For this purpose, all durations of a modeler’s PPM iterations are measured and the mean value is calculated.

Initial comprehension duration. When replaying the PPM instances using CEP, we observed differences in the time it took modelers to start working on the process model. Some started right away adding the first elements, while others invested more time in gaining an understanding of the mod-

eling task. To investigate the respective differences in modeling style, we defined the measure of *initial comprehension duration*. It captures the duration between opening the modeling editor and the beginning of the first modeling phase in milliseconds.

Reconciliation phase size. In both modeling sessions, cluster C1 sets itself apart from the other clusters in terms of reconciliation. Therefore, we consider this aspect further by the *reconciliation phase size*. It is calculated by counting the number of operations within a reconciliation phase. Then, the individual reconciliation phase sizes are aggregated by calculating the average size of a reconciliation phase and by calculating the maximum size of a reconciliation phase. These two measures are motivated by the replay of PPM instances in CEP. We observed that reconciliation may be done rather continuously or very focused at a certain point in time, e.g., for gaining additional space on the canvas or resolving a major problem. The former is addressed by the average reconciliation size, since it reflects the number of reconciliation operations throughout the PPM. The latter is considered by the maximum size, which indicates a large chunk of reconciliation.

Number of reconciliation phases. This measure also aims at gaining insights in the modelers' reconciliation behavior. C1 showed a higher number of reconciliation operations, but we did not know whether this was caused by more smaller reconciliation phases or by few larger ones. Therefore, this measure complements the reconciliation phase size measure by counting the number of reconciliation phases in a PPM instance.

Number of moves per node. The replay of PPM instances close to the cluster centroids also hinted at modelers placing model elements at strategic places, alleviating them from additional reconciliation. This aspect can be assumed to be reflected in the *number of moves per node*. We derive this measure by counting the number of move operations, i.e., `Move Node`, for each node within the process model, calculating the average number of move operations per node. The number of moves per node indicates how often a modeler touched a specific element. If modelers placed elements at strategic places, the average number of move operations should be considerable lower compared to modelers placing the model carelessly on the canvas and performing the layout operations later on.

8 Analysis: influencing factors and distinct modeling styles

Equipped with the additional measures introduced in Sect. 7, this section presents a statistical analysis of the influences on the PPM. First, Sect. 8.1 focuses on further characterizing the

Table 10 Additional measures Pre-Flight

Measure	C1	C2	C3
Adding rate	0.113	0.148	0.118
Initial comprehension duration (s)	149.0	73.3	154.2
Avg. iteration duration (s)	77.4	68.1	78.5
Avg. no. of moves per node	2.32	1.44	1.20
Avg. reconciliation phase size	6.68	5.79	5.32
Max. reconciliation phase size	21.38	15.32	12.74
No. of reconciliation phases	10.52	6.54	6.41

different modeling styles identified in Sect. 6. Subsequently, Sect. 8.2 addresses the question of which factors influence the modeling style.

8.1 Distinct modeling styles

Below, we apply the measures defined in Sect. 7 to each of the clusters of the two modeling tasks.

8.1.1 Pre-Flight

Table 10 illustrates the mean values for each cluster. C2 sets itself apart in terms of adding rate, the amount of time spent on initial comprehension, and the avg. duration of PPM iterations. For these particular measures, hardly any differences can be identified between C1 and C3. In terms of reconciliation measures, i.e., number of moves per node, avg. reconciliation phase size, max. reconciliation phase size, and number of reconciliation phases, C1 posts the highest values. In either case, C2 has the second highest value, followed by C3. The differences between C2 and C3 are relatively small though.

The statistical analysis presented in Table 11 supports most of the observations. C2 is indeed significantly different compared to C1 and C3 in terms of adding rate and initial comprehension duration. For average iteration duration, the difference is only significant when comparing C2 to C3. In terms of reconciliation measures, i.e., number of moves per node, max. reconciliation phase size, and number of reconciliation phases, the statistical analysis confirms the observation that C1 sets itself apart compared to the C2 and C3. No differences were observed in terms of reconciliation behavior between C2 and C3. No significant differences could be identified in terms of avg. reconciliation phase size.

8.1.2 NFL Draft

For the second modeling task, the statistics presented in Table 12 draw a similar picture. In terms of adding rate, C2 has the highest value. Similarly, C2 has the shortest ini-

Table 11 Significant differences for additional measures Pre-Flight

Statistic	All groups		Pairwise comparison		
			1–2	1–3	2–3
Adding rate	Sig.	0.000 ^a	0.001 ^b		0.000 ^b
	Test	Kruskall–Wallis	Mann–Whitney		
Initial comp. duration	Sig.	0.000 ^a	0.001 ^b		0.000 ^b
	Test	Kruskall–Wallis	Mann–Whitney		
Avg. iteration duration	Sig.	0.042 ^a			0.045 ^a
	Test	One-way ANOVA	Bonferroni post hoc		
Avg. no. of moves per node	Sig.	0.000 ^a	0.003 ^b	0.000 ^b	
	Test	Kruskall–Wallis	Mann–Whitney		
Avg. reconciliation phase size	Sig.				
	Test	One-way ANOVA			
Max. reconciliation phase size	Sig.	0.014 ^a			0.005 ^b
	Test	Kruskall–Wallis	Mann–Whitney		
No. reconciliation phases	Sig.	0.000 ^a	0.000 ^b	0.000 ^b	
	Test	Kruskall–Wallis	(un)equal variances <i>t</i> test		

^a $p < 0.05$; ^b $p < 0.05/3$

Table 12 Additional measures NFL Draft

Measure	C1	C2	C3
Adding rate	0.112	0.152	0.134
Initial comprehension duration (s)	160.5	71.8	127.2
Avg. iteration duration (s)	77.2	69.7	76.1
Avg. no. of moves per node	1.45	0.97	1.08
Avg. reconciliation phase size	5.89	5.15	5.05
Max. reconciliation phase size	14.23	9.93	9.76
No. of reconciliation phases	8.45	4.50	4.83

tial comprehension phase. When considering the differences between C1 and C3, we observe a difference compared to the first modeling task, indicating a considerable gap between C1 and C3 in terms of initial comprehension duration and adding rate. C3 seems to be between C1 and C2, a familiar picture throughout the data analysis. For iteration duration, C2 sets itself apart, while C1 and C3 post relatively similar values. In terms of the reconciliation statistics, similarities can be identified to the Pre-Flight modeling task, even though the differences are smaller, which might be caused by the smaller modeling task. Still, C1 posts the highest values in all reconciliation statistics.

The statistical analysis for adding rate shows significant differences between C1 and C2 (cf. Table 13). When using a *t* test for pairwise comparison, the difference between C1 and C3 is also significant. The difference between C2 and C3 is barely not significant ($t(70) = 2.10, p = 0.039$) since the Bonferroni correction dictates a significance level of $0.05/3 = 0.017$. Interestingly, when using the nonparametric Mann–Whitney test, the picture changes, indicating a significant difference between C2 and C3, but a barely nonsignif-

icant difference between C1 and C3 ($U = 438, p = 0.017$). For initial comprehension duration, the results for the first modeling task are replicated. C2 is significantly different compared to C1 and C3. The difference observed for the mean initial comprehension duration between C1 and C3 is not statistically significant. In terms of average iteration duration and the number of moves per node, the differences were not statistically significant. For avg. reconciliation phase size, max. reconciliation phase size, and the number of reconciliation phases, the results of the Pre-Flight task are replicated.

8.1.3 Interpretation

The statistical analysis for both tasks revealed several differences that complement the picture of the cluster characteristics.

We observe significant differences in terms of adding rate, meaning that adding of elements is done differently not only in absolute terms (number of iterations), but also relative over time. Modelers in cluster C2 seem to be faster in adding elements since they added more content in shorter modeling phases. Also, they started faster with adding content since the initial comprehension phases were significantly shorter compared to C1 and C3. Apparently, modelers in C2 were fast in making plans on how to create the process model and in using the modeling tool to convert the informal description into the formal model. No difference in initial comprehension duration and adding rate could be identified between C1 and C3.

Further, when investigating the reconciliation measures, differences for max. reconciliation size and the number of reconciliation phases could be identified for both modeling tasks pointing toward more reconciliation in C1. The

Table 13 Significant differences for additional measures NFL Draft

Statistic	All groups		Pairwise comparison		
			1–2	1–3	2–3
Adding rate	Sig.	0.000 ^a	0.000 ^b	0.002 ^b	
	Test	Kruskall–Wallis	(un)equal variances <i>t</i> test		
Adding rate	Sig.	0.000 ^a	0.000 ^b		0.016 ^b
	Test	Kruskall–Wallis	Mann–Whitney		
Initial comp. duration	Sig.	0.000 ^a	0.000 ^b		0.000 ^b
	Test	Kruskall–Wallis	(un)equal variances <i>t</i> test		
Avg. iteration duration	Sig.				
	Test	One-way ANOVA			
Avg. no. of moves per node	Sig.				
	Test	Kruskall–Wallis			
Avg. reconciliation phase size	Sig.				
	Test	Kruskall–Wallis			
Max. reconciliation phase size	Sig.	0.004 ^a		0.001 ^b	
	Test	Kruskall–Wallis	Mann–Whitney		
No. reconciliation phases	Sig.	0.000 ^a	0.000 ^a	0.000 ^a	
	Test	One-way ANOVA	Bonferroni post hoc		

^a $p < 0.05$; ^b $p < 0.05/3$

Table 14 Modelers in the same cluster

	C1	C2	C3	Overall
Modelers in cluster Pre-Flight	38	20	43	101
Modelers in same cluster NFL Draft	13	10	20	43
Modelers in same cluster (%)	34.21	50.00	46.51	42.57

measures presented in Sect. 7 provide us with additional insights in reconciliation differences that go beyond reconciliation breaks (cf. Sect. 3.4). Interestingly, the high number of reconciliation operations cannot be traced back to the average size of reconciliation phases since no significant differences could be identified. On the contrary, modelers in C1 had at least one significantly larger reconciliation phase compared to C3. This indicates phases of extensive layouting in the modeling process, which might have been caused by difficulties when creating the process model. The high number of reconciliation operations in C1 seems to be caused by a combination of longer PPM instances and phases of extensive layouting.

8.2 Factors influencing the modeling style

To understand which factors influence the modeling style and to establish to which extent certain factors are task-specific or modeler-specific, we first investigate the movement of modelers between different clusters over both modeling tasks. Second, we look at correlations of measures between the two modeling tasks to identify measures that were rather modeler-specific.

8.2.1 Cluster movement

When clustering the Pre-Flight process and the NFL Draft process, we obtained clusters with similar properties. Therefore, the question arises whether modelers in a specific cluster for the Pre-Flight process can be found in the corresponding cluster for the NFL Draft process. If all modelers are assigned to the same cluster for both modeling tasks, we can conclude that the modeler's style is entirely dependent on the modeler's personal preferences without any influence of the modeling task at hand.

Table 14 illustrates the number of modelers who stayed in the same cluster, e.g., 50.00 % of the modelers who were in C2 for the Pre-Flight process were also in C2 for the NFL Draft process.³ Overall, 42.57 % of the modelers remained in the same cluster. To test whether cluster moves reflect a random assignment or whether they are influenced by modeler-specific factors, we compute the expected number of moves under a null hypothesis of random cluster assignment and use the chi-square test for goodness of fit, rejecting the null hypothesis ($p = 0.009$). This points toward a combination of

³ Modelers who were assigned to clusters that were ignored for further analysis were also ignored for analyzing cluster movements.

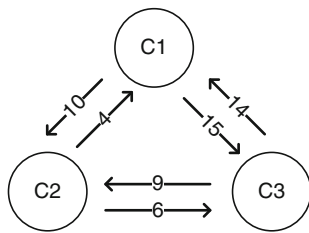


Fig. 7 Cluster movement

modeler and task-specific factors influencing the modeler's style. For instance, the modeling style might be influenced by modelers experiencing difficulties during the first modeling task. In the second task, a modeler might not face the same difficulties, resulting in a different modeling style and, thus, different cluster assignment.

Figure 7 illustrates the movement of modelers among the clusters. Modelers tended to move toward C2 for the second modeling task, which gained 19 additional modelers and lost only 10. On the contrary, C1 lost 25 modelers and gained only 18 additional modelers. For C3, the number of gained and lost modelers is similar, i.e., 21 gained and 23 lost. This could indicate that less modelers had problems with the second modeling task, which would be consistent with our finding that no significant differences among the clusters could be identified for the share of comprehension and delete iterations.

Going back to the measures defined in Sects. 3 and 7, we further investigate the cluster movements. The individual groups for cluster movement are relatively small, e.g., only four modelers moved from C2 in the Pre-Flight task to C1 in the NFL Draft task, making a detailed analysis difficult. Hence, we aggregate modelers into groups described in the sequel for analyzing cluster movement.

Our analysis indicated the following characteristics for the clusters.

- *Cluster C1.* more reconciliation/slower modeling
- *Cluster C2.* less reconciliation/faster modeling
- *Cluster C3.* less reconciliation/slower modeling

Since the largest differences in terms of our measures could be observed between C1 and C2, we assume them to be located toward the ends of a spectrum of modeling styles, while C3 can be placed in between. Based on this assumption, we perform the following aggregation of cluster movements.

- *Toward less reconciliation/faster modeling.* Modelers changing their modeling style toward faster modeling, i.e., C1 to C2, C1 to C3, and C3 to C2, were considered in this group. This group contains modelers who spent less time on reconciliation and might have experienced less difficulties in the second modeling task.

- *Toward more reconciliation/slower modeling.* This group contains modelers who slowed down their modeling endeavor during the second modeling task, i.e., C2 to C1, C2 to C3, and C3 to C1. Modelers in this group spent more time on reconciliation. Some of them might have experienced more difficulties in the second task.
- *Same.* This group contains modelers who were in the same cluster for both tasks.

For each modeler, we calculated the difference between the Pre-Flight task and the NFL task for each measure. Table 15 displays the results. Negative values indicate that the results for this measure decreased compared to the first modeling task. For example, we have established significant differences for the number of PPM iterations among all three groups in Sect. 6 with C2 posting the lowest values and C1 the highest, creating a spectrum of modeling styles in terms of PPM iterations. The aggregated cluster movement supports this impression, since modelers who moved toward less reconciliation/faster modeling showed an average decrease of 10.06 PPM iterations. On the contrary, modelers moving toward more reconciliation/slower modeling had only a mild average decrease of 0.5 PPM iterations (the NFL Draft modeling task was considerably smaller making a decrease in the number of PPM iterations likely). The measures in Table 15 draw a consistent picture of cluster movement. Modelers who moved toward less reconciliation/faster modeling needed less adding operations in a smaller number PPM iterations to create the process model in larger chunks. The number of reconciliation operations is even higher when modelers moved toward more reconciliation/slower modeling compared to the first modeling task, even though the second task was smaller. Mental effort indicates that modelers moving toward less reconciliation/faster modeling perceived the second task to be easier compared to the first one. Modelers moving to more reconciliation/slower modeling perceived the second task to be equally difficult compared to the first one, even though we observe a significant difference between both tasks for the whole population (cf. Sect. 5).

Summarized, we have observed a considerable number of modelers moving to different clusters when comparing the two modeling tasks. The initial set of measures (cf. Sect. 3) and the measures developed based on our observations (cf. Sect. 7) support our observation of placing the identified clusters on a spectrum of modeling styles. C1 represents more reconciliation and slower modeling, while C2 represents faster modeling and less reconciliation. Modelers in C3 seem to work slower with less reconciliation operations, representing a mixture of the characteristics of C1 and C2. The observed cluster movement points to the presence of task-specific factors influencing the modeler style. If the modeling style could be entirely attributed to the modeler's preferences, no cluster movement would be present. However, a consid-

Table 15 Measures for cluster movement

Measure	More rec./slower	Less rec./faster	Same
No. adding operations	-11.33	-22.41	-13.40
No. deleting operations	0.96	-8.21	-1.72
No. reconciliation operations	3.29	-40.41	-16.58
Avg. no. of PPM iterations	-0.50	-10.06	-4.58
Avg. iteration chunk size	-0.48	1.20	0.25
Avg. share of comprehension	1.48	-9.74	-6.16
Avg. reconciliation breaks	6.20	-3.08	1.69
Avg. delete iterations	2.61	-7.26	-0.34
Adding rate	-0.007	0.018	0.012
Initial comprehension duration (s)	19.93	-29.84	-10.44
Avg. iteration duration (s)	-4.53	0.95	2.06
Avg. no. of moves per node	0.05	-0.98	-0.39
Avg. reconciliation phase size	-0.13	-1.26	-0.50
Max. reconciliation phase size	-1.17	-9.91	-4.88
No. of reconciliation phases	-6.46	-15.18	-9.72
Mental effort	0.04	-0.41	-0.30

erable amount of modelers remained in the same cluster for both modeling tasks, pointing to task independent factors.

8.2.2 Correlations

To understand modeler-specific factors influencing the modeler's style, we introduce the notion of stability of measures among the two tasks. If a specific measure shows a high stability over the two modeling tasks, it indicates that there was only a limited influence of the modeling task. Therefore, a measure showing a high stability indicates a modeler-specific factor influencing the modeling style. For assessing the stability of the measures defined previously, we use correlational analysis. More specifically, we correlate all measures of the Pre-Flight task with the corresponding measure for the NFL Draft task. The results are shown in Table 16. It is interesting to note that several variables are highly correlated between both tasks. The number of reconciliation operations, adding rate, and average number of moves per node are strongly and significantly correlated. The same holds for the initial comprehension duration. Significantly, but less strongly correlated are the number of adding operations, average iteration chunk size, number of reconciliation phases, and average iteration duration. All these variables can be considered as stable across the two modeling tasks.

Beyond that, we were interested in how the measures relate to the mental effort perceived by the modelers. Correlating mental effort with the measures reveals that a significant correlation exists only for the average number of delete iterations. Note that the correlation was also almost equally strong in both task: 0.235 (0.012) for the first and 0.209 (0.025) for the second. This observation suggests that modelers perceive

Table 16 Pearson correlations for measures between task one and task two

Measure	Correlation	Sign.
No. of adding operations	0.202	0.031*
No. of deleting operations	0.062	0.508
No. of reconciliation operations	0.330	0.000**
Avg. no. of PPM iterations	0.153	0.103
Avg. iteration chunk size	0.214	0.022*
Avg. share of comprehension	0.006	0.951
Avg. reconciliation breaks	0.095	0.311
Avg. delete iterations	0.181	0.053
Adding rate	0.559	0.000**
Initial comprehension duration	0.354	0.000**
Avg. iteration duration	0.189	0.043*
Avg. no. of moves per node	0.467	0.000**
Avg. reconciliation phase size	0.125	0.189
Max. reconciliation phase size	0.114	0.231
No. of reconciliation phases	0.266	0.004**

* $p < 0.05$; ** $p < 0.01$

a modeling task as more difficult when the complexity of the tasks forces them to conduct delete operations.

In sum, we identified a considerable amount of movement among the clusters over the two modeling tasks, indicating that several characteristics of modeling style are indeed influenced by the modeling task. However, several measures showed strong and highly significant correlations between the two modeling tasks, pointing toward factors related to the individual modeler rather than to the modeling task.

9 Discussion and model building

Based on the presented analysis, Sect. 9.1 presents a first attempt to define a model that describes modeling styles and the factors that affect them. Then, we reflect on limitations of our study in Sect. 9.2.

9.1 Building a model

As discussed in Sect. 4, the PPM is influenced by task-specific characteristics and modeler-specific characteristics. The design of our exploratory study kept task-extraneous factors constant and focused on the modeler characteristics, i.e., cognitive properties and preferences, and on the task-intrinsic characteristics. We aimed at answering the following questions:

1. What aspects of the PPM constitute distinct modeling styles?
2. What aspects of modeling styles are affected by which factors and how?

Considering the first question, the cluster analysis revealed for both tasks three modeling styles, which can be distinguished by three main aspects of the modeling process. First, the layout behavior, which was pursued by modelers in C1 and resulted in considerably slower PPM instances. No such emphasis was observed for modelers in C2 and C3. Second, the extent to which the adding of content was streamlined and undisturbed, which we refer to as the efficiency of the modeling process. Modelers in C2 efficiently utilized their cognitive resources in large iteration chunks, the result of which was a focused and fast PPM. Finally, PPM clusters were also distinguished by evidence of difficulties encountered while modeling. These were mainly reflected when the modeler removed model parts (delete operations) and remodeled them (additional adding operations). Even though we observed delete operations in all clusters, C1 had a significantly higher amount of delete operations compared to C2 and C3, indicating that modelers in C1 experienced more difficulties. Issues while modeling also entail spending more time on comprehension (larger share of comprehension). Following this analysis, we have grouped the measures that were used in the data analysis to form three aspects of modeling style:⁴

Layout/Tool Behavior: operationalized by the measures number of reconciliation operations, number of reconciliation

phases, avg. number of moves per node, and max. reconciliation phase size.

Efficiency: the associated measures include avg. number of PPM iterations, iteration chunk size, share of comprehension, avg. iteration duration, adding rate, initial comprehension duration, number of reconciliation phases, max. reconciliation phase size, and reconciliation breaks.

Troubles: reflected by the measures of number of deleting operations, number of adding operations, share of comprehension, and delete iterations.

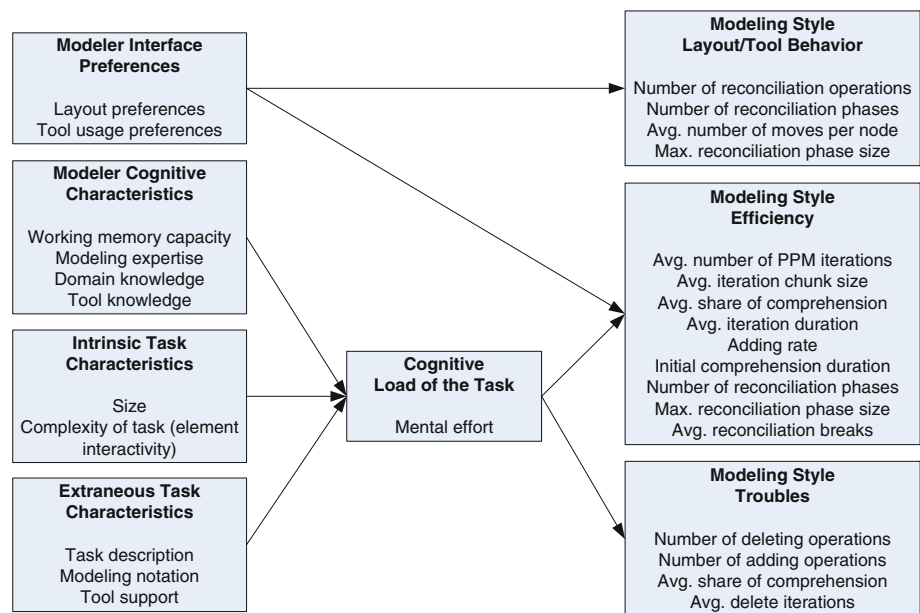
Considering the second question, it would seem reasonable to expect that some modeler-specific factors consistently affect the modeling style, regardless of the task at hand, while others would affect the modeling style in interaction with the task characteristics. A first look into this question is based on the cluster movement analysis. It was established that while modelers did not move arbitrarily between clusters, considerable movement has taken place, implying that the modeling style of a modeler is not fully consistent for different tasks. The cluster movement analysis indicated some relation between the modeling style and the perceived mental effort. More modelers were in C2 for the task whose mental effort was lower than for the task with higher mental effort. The cluster movement entailed consistent changes in measures of efficiency and troubles, as well as in layout behavior.

A better understanding of the consistency of specific aspects of the modeling style and the factors that might affect it is gained through the correlation analysis. It was established that several of the measures attributed to the reconciliation behavior indicate highly significant correlations along the two tasks. Only the correlation for max. reconciliation phase size is not significant, which is also part of the efficiency group. This might imply that this behavior is typical for an individual modeler, directly affected by the reconciliation preferences and independent of the modeling task at hand. In contrast, the measures related to the efficiency aspect of the modeling style exhibit different levels of correlation if any (e.g., adding rate was highly correlated, iteration chunk size correlated to a medium extent, and share of comprehension was not correlated at all).

This partial consistency, along with the findings from the cluster movement analysis, suggests that efficiency is affected by both the properties of the modeler and the properties of the task. The interaction of the task and the modeler's properties can be considered as the *cognitive load* imposed on the modeler by the specific task. It can be operationalized by the mental effort measure, which can explain some of the cluster movement findings. Cognitive load should also affect the trouble aspect of the modeling style. For the measures that reflect trouble, we did not find a significant correlation between the tasks. This seems reasonable, since modeling

⁴ Avg. reconciliation phase size was ignored since this measure showed significant results neither for the cluster analysis nor for the correlations.

Fig. 8 A model of factors influencing the modeling style



troubles are usually not consistently encountered. Furthermore, a significant correlation between number of delete operations (indicating troubles) and mental effort was found for both tasks, indicating that mental effort was perceived to be higher when troubles were encountered.

Summarizing this discussion, the model that emerges from our findings is depicted in Fig. 8. The model includes the three aspects of modeling style with their associated measures. The cognitive characteristics of the modeler, the intrinsic task characteristics, and the extraneous task characteristics affect cognitive load (operationalized by the mental effort measure), which in turn affects the efficiency and the trouble aspects of the modeling style, i.e., in case that cognitive load exceeds the modeler's working memory capacity, errors are likely to occur [45]. In contrast, the modeler's interface preferences directly affect both the layout/tool behavior and the efficiency aspect.

Several notes should be made about the proposed model. First, we designed the exploratory study to keep extraneous task characteristics constant. Hence, the effect of this factor on the cognitive load is merely an assumption that seems reasonable considering insights from cognitive load theory [31], yet currently not supported by the findings in this paper. Second, the effect of the interface preferences on efficiency is implied by the cluster analysis and is quite obvious, since extensive reconciliation operations reduce the efficiency of modeling. Third, the model does not include a relationship between the interface preferences and the cognitive load. Our findings suggest direct relationships between the interface preferences and the layout behavior and efficiency aspects. Still, there might also be an indirect relationship through cognitive load. It is possible that emphasized interface preferences cause increased cognitive load and thus

an additional effect on the efficiency and trouble aspects of the modeling style. However, establishing such an effect, as well as gaining a full understanding of the effects of the modeler interface preferences, requires additional research efforts.

Finally, emerging from exploratory findings, the proposed model cannot be considered a fully established theory. Rather, it serves as a research agenda and a platform for the derivation of hypotheses for further studies. Such studies can address factors which were kept constant in our study, such as modeling notation and tool, modeling expertise and domain knowledge of the modeler, and task description. Besides, future research should explore the individual parts of the model. For instance, troubles were only touched in this paper by considering the number of delete operations in a PPM instance. Research on problems arising during the creation of process models can be combined with cognitive load theory for the development of teaching materials. For this purpose, an in-depth understanding of not only the errors occurring during the PPM, but also the intrinsic and extraneous characteristics and the modeler's cognitive characteristics is in demand. For example, we might be able to establish the perceived difficulty of the various modeling constructs in order to focus on the most challenging parts when instructing our students in the craft of modeling. Future studies may also address possible correlations among the modeling style aspects and specific measures. The cluster analysis also suggests there could be correlations among metrics such as iteration chunk size, share of comprehension, and number of moves per node. These are not readily explained correlations. Future studies can establish what connections exist among different properties of the modeling process and offer theoretical explanations for them.

9.2 Limitations

The interpretation of our findings is presented with the explicit acknowledgment of a number of limitations to our study. First of all, our respondents represented a rather homogeneous and inexperienced group. Although relative differences in experience were notable, the group is not representative for the modeling community at large. At this stage, in particular, the question can be raised whether experienced modelers also exhibit the same modeling styles as skillful yet inexperienced modelers. In other words, will experienced modelers display similar characteristics of modeling style or can other styles be observed within their approaches? Therefore, we explicitly included the three factors of modeling expertise, domain knowledge, and tool knowledge in the model explaining the differences in modeling styles. The actual influence of these factors on the observed modeling style was beyond the scope of this work and has to be determined in future work. Note that we are mildly optimistic about the usefulness of the presented modeling styles on the basis of modeling behavior of graduate students, since we have established in previous work that such subjects perform equally well in process modeling tasks as some professional modelers [38].

Second, we cannot rule out that K-Means identified local minima, resulting in a suboptimal clustering. To counter this threat, we validated the clustering using a series of measures quantifying the PPM and identified significant differences among the three groups.

Third, our approach of using cluster analysis for identifying distinct modeling styles is based on the assumption that there exists one modeling style per PPM instance. Since it seems reasonable to assume that modelers may change their modeling style, e.g., when facing difficulties, this is a considerable limitation of our work. Still, the presented approach allowed us to gain initial insights into different modeling styles that can be extended toward including changes in modeling style during the PPM in future work.

10 Summary

This paper contributes to our understanding of how process models are created, as it constitutes the first systematic attempt to identify different modeling styles in the domain of business process modeling. We recorded and analyzed PPM instances of 115 students of courses on business process management in two modeling tasks. Using data mining techniques, we were able to identify three distinct modeling styles that occurred independently of the concrete modeling task. Each modeling style has specific characteristics that can be *measured* in terms of how the modeler acts on the modeling canvas.

Within the bounds of this exploratory study, we were able to observe three different modeling styles. We could distinguish (1) an “efficient modeling style” characterized by a limited time needed to think about the modeling task, and a fast rate of adding elements to the model; (2) a “layout-driven modeling style” which involves much time in creating a comprehensible layout while being less efficient in creating the model; and (3) an “intermediate modeling style” that is neither particularly efficient nor invests particularly into model layout. In addition, we found the choice for a particular modeling style to be subject to various factors. We observed that regardless of the modeling style, a modeler may face problems during modeling and may have to correct parts of the model. However, modelers following a “layout-driven modeling style” invested more work into correcting a model than modelers following other styles.

We observed modelers sticking to the same modeling style in all tasks, and we saw modelers following different modeling styles in different tasks. Thus, we contend that a particular modeling style depends on both modeler- and task-specific characteristics. We identified that (i) the time needed to think about the modeling task and (ii) the rate of adding elements to the model are more likely related to the modeler than to the task. Also, the amount of layouting invested during modeling is more related to the modeler than to the task.

These modeler-specific characteristics meet task-specific characteristics, which together determine the modeling style followed. Here, we found that the amount of layouting invested during modeling is *independent* of the perceived complexity of the task. This suggests that a modeler who prefers a good model layout will invest in this aspect even if the modeling task is difficult. We found that a modeler’s perception of a task as hard was correlated with the probability to face trouble during modeling and having to rework parts of the model. In contrast to this, the efficiency at which a model is created was largely independent of the perceived complexity of the task. All these insights are backed up by a number of *concrete measures* on the PPM, as formulated in a first model of the influence factors on process modeling. This model serves as basis for deriving hypothesis that can be investigated in future studies. Such studies might investigate factors influencing the modeling style that have not been addressed in this work, e.g., the modeler’s expertise. Additionally, researcher can use the model for identifying research areas demanding an in-depth understanding, e.g., troubles during the PPM. Therefore, the proposed model provides an agenda for future research on the PPM.

We believe these first insights regarding the PPM will be beneficial for future process modeling environments and will support teachers in mentoring their students on their way to become proficient process modelers by allowing them to measure differences in modeling styles. Additionally, this paper presented a viable experimental design for further

investigating the PPM, providing the ground for new investigations and for testing the hypotheses identified in this work.

The results of our study give rise to various future work. In addition to testing our model in additional experiments, we aim for a including changes in modeling style in our model, a more detailed investigation of the layouting behavior of modelers, a more fine-grained analysis of the influence of the concrete modeling task on the modeling style, and ultimately the influence of modeling style on modeling outcome.

Open Access This article is distributed under the terms of the Creative Commons Attribution License which permits any use, distribution, and reproduction in any medium, provided the original author(s) and the source are credited.

References

- Aguilar, E., Ruiz, F., García, F., Piattini, M.: Evaluation measures for business process models. In: Proceedings of the SAC '06, pp. 1567–1568 (2006)
- Becker, J., Rosemann, M., von Uthmann, C.: Guidelines of business process modeling. In: Proceedings of the BPM, vol. 1806 of LNCS, pp. 241–262. Springer (2000)
- Brooks, R.: Towards a theory of the cognitive processes in computer programming. *Int. J. Man-Mach. Stud.* **9**, 737–751 (1977)
- Cant, S., Jeffery, D., Henderson-Sellers, B.: A conceptual model of cognitive complexity of elements of the programming process. *Inf Softw Technol* **37**(7), 351–362 (1995)
- Cardoso, J.: Business process control-flow complexity: metric, evaluation, and validation. *Int. J. Web Serv. Res.* **5**(2), 49–76 (2008)
- Claes, J., Vanderfeesten, I.T.P., Reijers, H.A., Pinggera, J., Weidlich, M., Zugal, S., Fahland, D., Weber, B., Mendling, J., Poels, G.: Tying process model quality to the modeling process: the impact of structuring, movement, and speed. In: Proceedings of the BPM '12, pp. 33–48 (2012)
- Crapo, A.W., Waisel, L.B., Wallace, W.A., Willemain, T.R.: Visualization and the process of modeling: a cognitive-theoretic view. In: Proceedings of the KDD '00, pp. 218–226 (2000)
- Frederiks, P., Weide, T.: Information modeling: the process and the required competencies of its participants. *Data Knowl. Eng.* **58**(1), 4–20 (2006)
- Gray, P.: Psychology. Worth Publishers, UK (2007)
- Gruhn, V., Laue, R.: Complexity metrics for business process models. In: Proceedings of the ICBIS '10, pp. 1–12 (2006)
- Guindon, R.: Designing the design process: exploiting opportunistic thoughts. *Hum.-Comput. Interact.* **5**(2), 305–344 (1990)
- Guindon, R., Curtis, B.: Control of cognitive processes during software design: what tools are needed?. In: Proceedings of the CHI '88, pp. 263–268 (1988)
- Hamerly, G., Elkan, C.: Alternatives to the k-means algorithm that find better clusterings. In: Proceedings of the CIKM '02, pp. 600–607 (2002)
- Hoppenbrouwers, S., Proper, H., Weide, T.: A fundamental view on the process of conceptual modeling. In: Proceedings of the ER '05, pp. 128–143 (2005)
- Krogstie, J., Sindre, G., Jørgensen, H.: Process models representing knowledge for action: a revised quality framework. *EJIS* **15**(1), 91–102 (2006)
- Larkin, J.H., Simon, H.A.: Why a diagram is (sometimes) worth ten thousand words. *Cogn. Sci.* **11**(1), 65–100 (1987)
- Lindland, O.I., Sindre, G., Sølvsberg, A.: Understanding quality in conceptual modeling. *IEEE Softw.* **11**(2), 42–49 (1994)
- MacQueen, J.: Some methods of classification and analysis of multivariate observations. In: Proceedings of the Berkeley Symposium on Mathematical Statistics and Probability, pp. 281–297 (1967)
- McCracken, M.W.: Models of designing: understanding software engineering education from the bottom up. In: Proceedings of the CSEET '02, pp. 55–63, (2002)
- Mendling, J.: Metrics for process models: empirical foundations of verification, error prediction, and guidelines for correctness. Springer (2008)
- Mendling, J., Reijers, H.A., Cardoso, J.: What makes process models understandable? In: Proceedings of the BPM '07, pp. 48–63 (2007)
- Mendling, J., Reijers, H.A., van der Aalst, W.M.P.: Seven process modeling guidelines (7PMG). *Inf. Softw. Technol.* **52**(2), 127–136 (2010)
- Mendling, J., Verbeek, H., van Dongen, B., van der Aalst, W., Neumann, G.: Detection and prediction of errors in EPCs of the SAP reference model. *Data Knowl. Eng.* **64**(1), 312–329 (2008)
- Miller, G.: The magical number seven, plus or minus two: some limits on our capacity for processing information. *Psychol. Rev.* **63**, 81–97 (1956)
- Moody, D.L.: Theoretical and practical issues in evaluating the quality of conceptual models: current state and future directions. *Data Knowl. Eng.* **55**(3), 243–276 (2005)
- Moody, D.L.: The “physics” of notations: toward a scientific basis for constructing visual notations in software engineering. *IEEE Trans. Softw. Eng.* **35**(6), 756–779 (2009)
- Morris, W.T.: On the art of modeling. *Manag. Sci.* **13**(12), B-707–B-717 (1967)
- Newell, A.: Unified Theories of Cognition. Harvard University Press, Cambridge (1990)
- Newell, A., Simon, H.: Human Problem Solving. Prentice Hall, Englewood Cliffs (1972)
- Paas, F., Renkl, A., Sweller, J.: Cognitive load theory and instructional design: recent developments. *Educ. Psychol.* **38**(1), 1–4 (2003)
- Paas, F., Tuovinen, J.E., Tabbers, H., Gerven, P.W.M.V.: Cognitive load measurement as a means to advance cognitive load theory. *Educ. Psychol.* **38**(1), 63–71 (2003)
- Petre, M.: Why looking isn't always seeing: readership skills and graphical programming. *Commun. ACM*, **38**(6), 33–44 (1995)
- Pinggera, J., Furtner, M., Martini, M., Sachse, P., Reiter, K., Zugal, S., Weber, B.: Investigating the process of process modeling with eye movement analysis. In: Proceedings of the ER-BPM '12, pp. 438–450 (2013)
- Pinggera, J., Soffer, P., Zugal, S., Weber, B., Weidlich, M., Fahland, D., Reijers, H.A., Mendling, J.: Modeling styles in business process modeling. In: Proceedings of the BPMDS '12, pp. 151–166 (2012)
- Pinggera, J., Zugal, S., Weber, B.: Investigating the process of process modeling with cheetah experimental platform. In: Proceedings of the ER-POIS'10, pp. 13–18 (2010)
- Pinggera, J., Zugal, S., Weber, B., Fahland, D., Weidlich, M., Mendling, J., Reijers, H.: How the structuring of domain knowledge can help casual process modelers. In: Proceedings of the ER '10, pp. 445–451 (2010)
- Pinggera, J., Zugal, S., Weidlich, M., Fahland, D., Weber, B., Mendling, J., Reijers, H.A.: Tracing the process of process modeling with modeling phase diagrams. In: Proceedings of the ER-BPM '11, pp. 370–382 (2012)
- Reijers, H., Mendling, J.: A study into the factors that influence the understandability of business process models. *IEEE Trans. Syst. Man Cybern. Part A* **41**(3), 449–462 (2011)
- Rittgen, P.: Negotiating models. In: Proceedings of the CAiSE '07, pp. 561–573 (2007)

40. Rolón, E., Cardoso, J., García, F., Ruiz, F., Piattini, M.: Analysis and validation of control-flow complexity measures with bpmn process models. *Enterp. Bus.-Process Inf. Syst. Model.* **29**, 58–70 (2009)
41. Shanteau, J.: How much information does an expert use? Is it relevant? *Acta Psychol.* **81**(1), 75–86 (1992)
42. Siau, K., Rossi, M.: Evaluation techniques for systems analysis and design modelling methods—a review and comparative analysis. *Inf. Syst. J.* **21**(3), 249–268 (2011)
43. Soffer, P., Kaner, M., Wand, Y.: Towards understanding the process of process modeling: theoretical and empirical considerations. In: *Proceedings of the ER-BPM '11*, pp. 357–369 (2011)
44. Stirna, J., Persson, A., Sandkuhl, K.: Participative enterprise modeling: experiences and recommendations. In: *Proceedings of the CAiSE '07*, pp. 546–560 (2007)
45. Sweller, J.: Cognitive load during problem solving: effects on learning. *Cogn. Sci.* **12**(2), 257–285 (1988)
46. Tegarden, D.P., Sheetz, S.D.: Cognitive activities in OO development. *Int. J. Human-Comput. Stud.* **54**, 779–798 (2001)
47. Tracz, W.: Computer programming and the human thought process. *Softw. Pract. Exp.* **9**(2), 127–137 (1979)
48. Van der Aalst, W., Ter Hofstede, A.: Verification of workflow task structures: a petri-net-based approach. *Inf. Syst.* **25**(1), 43–69 (2000)
49. van der Aalst, W., ter Hofstede, A., Kiepuszewski, B., Barros, A.: Workflow patterns. *Distrib. Parallel Databases* **14**, 5–51 (2003)
50. Willemain, T.R.: Model formulation: what experts think about and when. *Oper. Res.* **43**(6), 916–932 (1995)
51. Zugal, S., Pinggera, J., Mendling, J., Reijers, H., Weber, B.: Assessing the impact of hierarchy on model understandability—a cognitive perspective. In: *Proceedings of the EESSMod '11*, pp. 123–133 (2011)
52. Zugal, S., Pinggera, J., Reijers, H., Reichert, M.U., Weber, B.: Making the case for measuring mental effort. In: *Proceedings of the EESSMod '12*, pp. 37–42 (2012)
53. Zugal, S., Pinggera, J., Weber, B.: Assessing process models with cognitive psychology. In: *Proceedings of the EMISA '11*, pp. 177–182 (2011)
54. Zugal, S., Soffer, P., Pinggera, J., Weber, B.: Expressiveness and understandability considerations of hierarchy in declarative business process models. In: *Proceedings of the BPMDS '12*, pp. 167–181 (2012)

Author Biographies



Jakob Pinggera is a Ph.D. candidate at the University of Innsbruck (Austria). Jakob is a member of Quality Engineering (QE) Research Group and member of the Research Cluster on Business Processes and Workflows at QE. Jakob received his M.Sc. degree from the Department of Computer Science, University of Innsbruck in 2009. His main research interest is the process of process modeling, i.e., how modelers create business process models. His research interests

further include business process modeling notations and business process quality. Jakob has published more than 30 refereed papers in international journals, conferences, and workshops.



Pnina Soffer is a senior lecturer in the Information Systems Department at the University of Haifa. She received her B.Sc (1991) and M.Sc (1993) in Industrial Engineering from the Technion and Ph.D in Information Systems Engineering from the Technion (2002). Her research deals with business process modeling and management, requirements engineering, and conceptual modelling, addressing issues such as goal orientation, flexibility, interoperability, and context-aware adaptation. Her research has appeared in journals such as *Journal of the AIS*, *European J of IS*, *Requirements Engineering*, *Information Systems*, and others. She has served as a guest editor of a number of special issues related to various business process topics such as business process flexibility, coordinated development of business processes and information systems, and business process design. Pnina has served in program committees of numerous conferences, including CAiSE, BPM, ER, and others. She is a member of the CAiSE steering committee and an organizer of the BPMDS working conference. She is a member of editorial boards of several journals including the *Journal of the AIS*.



Dirk Fahland is assistant professor at the Eindhoven University of Technology researching in the area of distributed systems. His research interests include distributed processes and systems built from distributed components for which he investigates modeling systems (using process modeling languages, Petri nets, or scenario-based techniques), analyzing systems for errors or misconformances (through verification or simulation), repairing erroneous/misconforming systems, discovering system models from event logs, and synthesizing systems and components from specifications.



Matthias Weidlich is a research fellow and adjunct lecturer at the Technion, The Israel Institute of Technology. He received his Ph.D. in Computer Science from the Hasso Plattner Institute (HPI), University of Potsdam, Germany, in 2011. His research focuses on processes as the basis of behavioral modeling and analysis with contributions related to process model matching, consistency analysis, process transformations, and compliance monitoring. Further,

his research interests include data interoperability and reasoning in event-processing environments. His results appeared in journals, such as *IEEE Transactions on Software Engineering*, *Information Systems*, *The Computer Journal*, and *Acta Informatica*. He is on the editorial board of Elsevier's *Information Systems* and a member of ACM and IEEE.



Stefan Zugal is a Ph.D. candidate at the University of Innsbruck (Austria). Stefan is a member of Quality Engineering (QE) Research Group and member of the Research Cluster on Business Processes and Workflows at QE. Stefan received his M.Sc. degree from the Department of Computer Science, University of Innsbruck in 2008. His main research interests are declarative business process models and the understandability of business process models. Stefan has published

more than 30 refereed papers in international journals, conferences, and workshops.



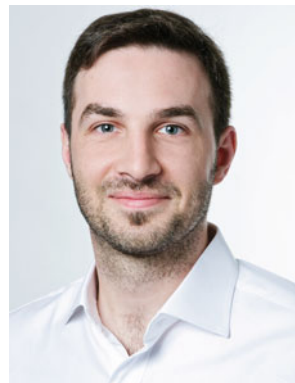
Barbara Weber is associate professor at the University of Innsbruck (Austria). Barbara is a member of the Quality Engineering (QE) Research Group and head of the Research Cluster on Business Processes and Workflows at QE. Barbara holds a Habilitation degree in Computer Science and Ph.D. from the University of Innsbruck. Barbara's research interests include process model understandability, process of process modeling, integrated process life cycle

support, change patterns, process flexibility, user support in flexible process-aware systems, and recommendations to optimize process execution. Barbara has published more than 100 refereed papers, for example, in *Data and Knowledge Engineering*, *Computers in Industry*, *Science of Computer Programming*, *Enterprise Information Systems* and *IET Software*, and co-author of the book "Enabling Flexibility in Process-aware Information Systems: Challenges, Methods, Technologies" by Springer. Moreover, Barbara is organizer of the successful BPI workshop series.



Hajo A. Reijers is a full professor in Information Systems at Eindhoven University of Technology (TU/e), as well as the head of Business Process Management (BPM) research at Perceptive Software. He received a Ph.D. in Computer Science (2002), an M.Sc. in Computer Science (1994), and an M.Sc. in Technology and Society (1994), all from TU/e. His research interests are business process modeling, workflow management, and process innovation. On these and

related topics, he published over 150 scientific papers, chapters in edited books, and articles in professional journals.



Jan Mendling is a full professor and head of the Institute for Information Business at WU Vienna. His research areas include Business Process Management, Conceptual Modelling, and Enterprise Systems. He received a Ph.D. degree from WU Vienna (Austria) and diploma degrees from the University of Trier (Germany) in 2003. After being a postdoc with QUT Brisbane (Australia) and a junior professor at HU Berlin (Germany), he moved back to WU in 2011. He

has published more than 200 research papers and articles, among others in *ACM* and *IEEE Transactions* journals, *Information Systems*, *Data and Knowledge Engineering*, and *Decision Support Systems*. He is member of the editorial board of three international journals, one of the founders of the Berlin BPM Community of Practice (www.bpmb.de), and board member of the Austrian Gesellschaft für Prozessmanagement.