Ghent University
Faculty of Economics and Business Administration
Department of Business Informatics and Operations Management

Eindhoven University of Technology
Department of Industrial Engineering and Innovation Sciences
Information Systems Group

Investigating the Process of Process Modeling and its Relation to Modeling Quality

The Role of Structured Serialization

Jan Claes

Supervisors: Prof. dr. Geert Poels, Prof. dr. Frederik Gailly
Ghent University, Belgium

Prof. dr. ir. Paul Grefen, Prof. dr. ir. Irene Vanderfeesten
Eindhoven University of Technology, the Netherlands

Submitted to Ghent University and Eindhoven University of Technology
in fulfillment of the requirements for the degree of Doctor
November 2015

PhD Series – Ghent University
Faculty of Economics and Business Administration
http://www.ugent.be/eb

UNIVERSITEIT GENT

TU/e Technische Universiteit
Eindhoven
University of Technology

Doctoral Jury

- Prof. dr. Ingrid Heynderickx (Dean School of Industrial Engineering & Innovation Sciences TU/e, The Netherlands)
- Prof. dr. ir. Paul Grefen (Supervisor TU/e, The Netherlands)
- Prof. dr. Geert Poels (Supervisor UGent, Belgium)
- Prof. dr. ir. Irene Vanderfeesten (Co-supervisor TU/e, The Netherlands)
- Prof. dr. Frederik Gailly (Co- supervisor UGent, Belgium)
- Prof. dr. Wijnand Ijsselsteijn (TU/e, The Netherlands)
- Prof. dr. Manu De Backer (UGent, Belgium)
- Prof. dr. Stefanie Rinderle-Ma (University of Vienna, Austria)

The mentioned academic titles conform to the Belgian standard

Something I learned on the road...

> *"You can't connect the dots looking forward; you can only connect them looking backwards. So you have to trust that the dots will somehow connect in your future. You have to trust in something - your gut, destiny, life, karma, whatever."*
>
> Steve Jobs

Acknowledgements

I didn't even apply for the job.

Six years ago, after having worked in industry for three years, I felt it was time for a new challenge. Allowing myself a 3-month gap, I was planning to finish some personal projects and not to focus on browsing job applications. Only, my attention was drawn to a vacancy at Ghent University College, for which I could not wait 3 months to apply, so I ended up breaking the promise I made to myself and I applied. Nevertheless, the college was looking for experienced researchers and my application was rejected after a first panel meeting. No worries, I was looking forward to my 3 months of quietness.

Two weeks later, I got an e-mail from Geert Poels, professor at Ghent University, who had been member of the selection committee of my application at Ghent University College. He explained to me that he was looking for someone to help with his teaching duties and suggested that I could start at Ghent University as a teaching assistant and PhD researcher under his supervision. Cautiously he added that he was counting on my teaching experience and took a small risk because he did not know what to expect from my research capabilities.

Today, I am incredibly thankful to Geert for this wonderful opportunity. Not only did he introduce me to the world of research that I fell in love with, he kept supporting me when I was exploring the most unusual research topics and methods. It must not have been easy to collaborate with such a stubborn and rebellious PhD student. Geert, thank you for not giving up on me, thanks for all the opportunities, advice, trust, independence, and all other kinds of support!

In 2012, I started collaborating with Eindhoven University of Technology, which was later formalized in the set-up of a joint PhD. So besides having had the pleasure to work with a fantastic supervisor, I now had the luxury to complete my PhD project under supervision of four excellent researchers (one supervisor and one co-supervisor at both universities). Irene, Frederik and Paul, thank you so much for all the feedback, for sharing work and laughter, for being there when I needed you!

Further, I have met so many brilliant minds from the academic world, which have helped shaping this work. Special thanks go to Irene V. for our continuing close collaboration that started in 2012, Hajo R. for inviting me to collaborate with the PPM researchers, to Barbara W. and Jakob P. for their help with the Cheetah tool, to Eric V. for all the help with the ProM tool, to Wil vdA. who keeps trying to convince me to shift my research focus to process mining, to Anne R. and Christian G. for just being so inspirational, to all my co-authors for their input and to the jury of my PhD defense for their critical judgment. Also thank you Anne R., Barbara W., Benoit D., Boudewijn vD. Carlos R., Carol B., Christel R., Christian G., Dirk dS., Dirk F., Eric V., Erik P., Hajo R., Jakob P., Jan M., Jan R., Jeffrey P., Jochen dW., Joel R., Joos B., Koen S., Manu dB., Markus D., Martin V., Matthias W., Michael zM., Mieke J., Mijalce S., Peter R., Raykenler H., Remco D., Rob vW., Roel W., Shirley G., Stefan Z., Stefanie R.M., Stijn V., Uzay K., Wijnand IJ., Wil vdA., and all the other researchers for the interesting conversations at multiple occasions!

A big thank you is also reserved for the people in companies that I have worked with during my PhD. Thanks Alain dL. (Volvo), Benoit C. (BNP Paribas Fortis), Bram vS. (Möbius and AE), Bruno dH. (Volvo), Daniël dS. (EROV), Eddy J. (Ideas@Work), Filip V. (Century 21), Frank vG. (Rabobank), Geert D. (EROV), Geert D. (Flexpack), Guillaume H. (Möbius), Guy M. (Daikin Europe), Ingmar R. (Volvo), Jan S. (Ideas@Work), Jeroen P. (EROV), Karolien M. (EROV), Katrien vG. (Stad Gent), Lieve dlR. (Century 21), Luc dlR. (Century 21), Olivier dB. (Ideas@Work), Piet dG. (UGent), Pieter-Jan P. (Ideas@Work), Roland R. (Capgemini), Tim M. (Möbius), Ward S. (Volvo) for your interest, support and/or collaboration!

Not only for sharing research ideas, but especially for all the fun and support, thank you to my colleagues! Thank you Adnan K., Amy vL., Ann vR., Annemarie vdA., Bart M., Ben R., Birger R., Charlotte C., Dirk S., Elien D, Elisah L., Frederik G., Gert L., Griet vdV., Ken D., Laleh R., Machteld S., Martine M., Maxime B., Michael V., Nadejda A., Renata P., Rob vW., Tarik K., Tom P., Sebastiaan vdW., Sofie dC., Steven M., Wim L., all the other colleagues of the faculty, and all the colleagues at IS group of IE&IS faculty of Eindhoven University of Technology!

Also, I would like to take the opportunity to thank all the students that I came in contact with. Sometimes you annoyed me, but mostly you were a welcome distraction, a true challenge, an inspiration, and an attentive audience!

One cannot accept a challenge as big as a PhD project without a net of support. Therefore, thank you for listening to my complaints, for sharing this rollercoaster adventure, for accepting my physical or mental absence, for being true friends! Thank you Agnes, S., Amira B., Bomma C., David dT., Dorian B., Enya H., Fien B., Fred B., Gust B., Jolien W., Joris C., Katrien C., Kimberley G., Kristof dW., Lien dC., Mathias V., Maxime vdM., Olivia H., Phoebe A., Vishvas P., all the monitors from summer camps at Red Cross Flanders and AJOK, and all members of my family!

Finally, a huge thank you goes to my brother and parents. Luk, I have learnt a lot from you and I thank a lot to you! Mom, dad, thank you for being so awesome, for raising me, for supporting me, for funding me, for providing me with the best chances in life, for being there for me, for everything! Without you, I would never have been able to start this PhD. Without you, I would never have been able to finish this PhD. Thank you!

Jan Claes

Gent, November 2015

Table of Contents

List of Figures

List of Tables

List of Definitions

List of Acronyms

BPI	Business Process Improvement
BPM	Business Process Management
BPMM	Business Process Maturity Model
BPMN	Business Process Model & Notation
BPMs	Business Process Management system
BPO	Business Process Orientation
BPR	Business Process Reengineering
CAiSE	Conference on Advanced Information Systems Engineering
ECIS	European Conference on Information Systems
EIS	Enterprise Information Systems
EPC	Event-Driven Process Chain
FEB	Faculty of Economics and Business Administration
IIA	Institute of Internal Auditors
LNBIP	Lecture Notes in Business Information Processing
LNCS	Lecture Notes in Computer Science
PAIS	Process-Aware Information System
PPM	Process of Process Modeling
RQ	Research Question
SPMT	Structured Process Modeling Theory
UML	Unified Modeling Language
WfMs	Workflow Management system

Summary (English)

Lately, the focus of organizations is changing fundamentally. Where they used to spend almost exclusively attention to results, in terms of goods, services, revenue and costs, they are now concerned about the efficiency of their business processes. Each step of the business processes needs to be known, controlled and optimized. This explains the huge effort that many organizations currently put into the mapping of their processes in so-called (business) process models.

Unfortunately, sometimes these models do not (completely) reflect the business reality or the reader of the model does not interpret the represented information as intended. Hence, whereas on the one hand we observe how organizations are attaching increasing importance to these models, on the other hand we notice how the quality of process models in companies often proves to be insufficient.

The doctoral research makes a significant contribution in this context. This work investigates in detail how people create process models and why and when this goes wrong. A better understanding of current process modeling practice will form the basis for the development of concrete guidelines that result in the construction of better process models in the future.

The first study investigated how we can represent the approach of different modelers in a cognitive effective way, in order to facilitate knowledge building. For this purpose the PPMChart was developed. It represents the different operations of a modeler in a modeling tool in such a way that patterns in their way of working can be detected easily. Through the collection of 704 unique modeling executions (a joint contribution of several authors in the research domain), and through the development of a concrete implementation of the visualization, it became possible to gather a great amount of insights about how different people work in different situations while modeling a concrete process.

The second study explored, based on the discovered modeling patterns of the first study, the potential relations between how process models were being constructed and which quality was delivered. To be precise, three modeling patterns from the previous study were investigated further in their relation with the understandability of the produced process model. By

comparing the PPMCharts that show these patterns with corresponding process models, a connection was found in each case. It was noticed that when a process model was constructed in consecutive blocks (i.e., in a structured way), a better understandable process model was produced. A second relation stated that modelers who (frequently) moved (many) model elements during modeling usually created a less understandable model. The third connection was found between the amount of time spent at constructing the model and a declining understandability of the resulting model. These relations were established graphically on paper, but were also confirmed by a simple statistical analysis.

The third study selected one of the relations from the previous study, i.e., the relation between structured modeling and model quality, and investigated this relation in more detail. Again, the PPMChart was used, which has lead to the identification of different ways of structured process modeling. When a task is difficult, people will spontaneously split up this task in sub-tasks that are executed consecutively (instead of simultaneously). Structuring is the way in which the splitting of tasks is handled. It was found that when this happens consistently and according to certain logic, modeling became more effective and more efficient. Effective because a process model was created with less syntactic and semantic errors and efficient because it took less time and modeling operations. Still, we noticed that splitting up the modeling in sub-tasks in a structured way, did not always lead to a positive result. This can be explained by some people structuring the modeling in the wrong way. Our brain has cognitive preferences that cause certain ways of working not to fit. The study identified three important cognitive preferences: does one have a sequential or a global learning style, how context-dependent one is and how big one’s desire and need for structure is. The Structured Process Modeling Theory was developed, which captures these relations and which can form the basis for the development of an optimal individual approach to process modeling. In our opinion the theory has the potential to also be applicable in a broader context and to help solving various types of problems effectively and efficiently.

Samenvatting (Dutch)

Er is een fundamentele verschuiving aan de gang van de focus van organisaties. Waar zij vroeger bijna uitsluitend aandacht hadden voor resultaten in termen van producten, diensten, opbrengsten en kosten, ligt men nu wakker van de bedrijfsprocessen. Men wil elke stap in het bedrijfsproces kennen, beheersen en optimaliseren. Dit verklaart de enorme inspanningen die veel organisaties vandaag leveren voor het in kaart brengen van hun processen in zogenaamde (bedrijfs)procesmodellen.

Helaas komen deze modellen soms niet (helemaal) overeen met de bedrijfsrealiteit of interpreteert de lezer van een model de voorgestelde informatie anders dan bedoeld. Waar we dus enerzijds constateren dat men in organisaties steeds meer belang gaat hechten aan deze modellen, stellen we anderzijds ook vast dat de kwaliteit van de procesmodellen in bedrijven dikwijls te wensen overlaat.

Het doctoraatsonderzoek levert een belangrijke bijdrage in deze context. Dit werk onderzoekt in detail hoe mensen procesmodellen maken en waarom of wanneer het fout gaat. Een beter begrip van de huidige manier van procesmodelleren ligt aan de basis voor het ontwikkelen van concrete richtlijnen die ervoor kunnen zorgen dat in de toekomst betere procesmodellen gemaakt zullen worden.

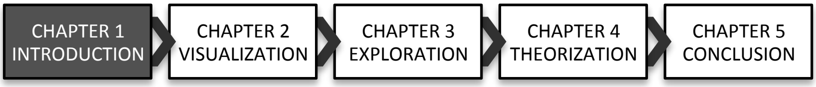
De eerste studie onderzocht hoe we de werkwijze van verschillende modelleurs op een cognitief effectieve wijze kunnen voorstellen, zodat het bouwen van deze kennis gemakkelijker wordt. Daartoe werd de PPMChart visualisatie ontwikkeld. Deze stelt de verschillende operaties van een modelleur in een modelleertool voor op zulke wijze dat gemakkelijk patronen ontdekt kunnen worden in hun manier van werken. Door de verzameling van data van niet minder dan 704 unieke modelleersessies (een gezamenlijke bijdrage van verschillende auteurs in het vakgebied) en door de ontwikkeling van een concrete implementatie van de visualisatie, werd het mogelijk om een grote hoeveelheid kennis te vergaren over hoe verschillende mensen te werk gaan in verschillende situaties bij het modelleren van een concreet proces.

De tweede studie verkende aan de hand van de ontdekte modelleerpatronen de mogelijke relaties tussen hoe procesmodellen gemaakt worden en welke kwaliteit daarmee geleverd werd. Concreet werden drie modelleerpatronen uit de vorige studie nader onderzocht in relatie met de verstaanbaarheid van het gemaakte procesmodel. Door het vergelijken van de PPMCharts die deze patronen vertonen, met de bijhorende procesmodellen, werd telkens een verband gevonden. Zo werd vastgesteld dat wanneer men het procesmodel in opeenvolgende blokken maakt (dus op een gestructureerde manier), een betere verstaanbaarheid van het resulterende procesmodel bekomen werd. Een tweede verband stelde dat modelleurs die (veel) elementen in het model (veel) verschuiven, doorgaans een minder verstaanbaar model creëerden. Het derde verband werd gevonden tussen de hoeveelheid tijd die men spendeert aan het maken van het model en een dalende verstaanbaarheid van het resulterende model. Deze verbanden werden grafisch vastgesteld op papier, maar werden bekrachtigd door een eenvoudige statistische analyse.

De derde studie selecteerde één van de verbanden uit de vorige studie, namelijk de relatie tussen gestructureerd modelleren en modelkwaliteit, en bestudeerde deze in meer detail. Opnieuw werd de ontwikkelde PPMChart visualisatie ingezet, wat leidde tot het identificeren van verschillende manieren van gestructureerd modelleren. Wanneer een taak moeilijk is, gaan mensen die spontaan opsplitsen in deeltaken die achtereenvolgens (in plaats van tegelijk) opgelost worden. Structureren gaat over de manier waarop men het opsplitsen aanpakt. Er werd vastgesteld dat wanneer dit op een consistente wijze en volgens een bepaalde logica gebeurt, het modelleren beter en gemakkelijker ging. Beter omdat een procesmodel werd gemaakt dat minder syntactische en semantische fouten bezat en gemakkelijker omdat hiervoor minder tijd en modelleeroperaties nodig waren. Toch merkten we dat het opsplitsen in deeltaken op een gestructureerde manier niet altijd tot een positief resultaat leidde. Dit kan verklaard worden doordat sommige mensen op een verkeerde manier structureren. Onze hersenen hebben immers cognitieve voorkeuren die ertoe leiden dat bepaalde manieren van werken niet bij ons passen. De studie identificeerde drie belangrijke factoren: heb je een sequentiële of globale leerstijl, hoe context-afhankelijk ben je en hoe groot is je verlangen en noodzaak naar structuur. De Structured Process Modeling Theory werd ontwikkeld die deze verbanden vastlegt en die de basis kan vormen van het ontwerpen van een optimale individuele werkwijze voor procesmodelleren. De theorie heeft volgens ons het potentieel om ook ruimer toegepast te kunnen worden en te helpen bij het effectief en efficiënt oplossen van allerlei soorten problemen.

1

Introduction



Summary. The introduction describes the research problem, context and design. Further, an overview of the structure of this dissertation is presented, as well as a list of the papers published during the doctoral project.

1.1. Research context

Organizations operate in an increasingly complex business context, which is reflected in the way they manage their activities (Håkansson & Snehota, 1989). The focus of organizations is therefore no longer only on the end product or service, but the whole business process of creation and delivery to the customer of this product or service is targeted for optimization (Mccormack, 2001; Willaert et al., 2007).

Because of the increased importance that organizations attach to their business processes, they nowadays put a lot of effort in documenting, analyzing and improving them (Burton-Jones et al., 2009). For this purpose, business process models are often constructed as a supporting instrument (Abecker et al., 2000; Davies et al., 2006; Kock et al., 2009; Xiao & Zheng, 2012). The models represent the relevant properties of the business processes under study in an orderly manner, aggregating information of different allowed process executions (Dumas et al., 2013).

The importance of (business) process models as a tool in gaining or retaining competitive advantage, requires a thorough understanding of the factors that impact the quality of process models (Krogstie et al., 2006; Mendling, 2008; Rittgen, 2010). Further, there is clearly a need to offer operational guidance on how models of high quality have to be created (Becker et al., 2000; Mendling, Reijers, et al., 2010). Hence, within the research stream of (business) process model quality two key research questions exist: (i) *what* makes a process model of high quality and (ii) *how* can process models be constructed that possess high quality. This dissertation is situated in the latter research stream, which studies the Process of Process Modeling (PPM).

First the essential concepts are defined on the next page, before the research context is described in more detail in Section 1.2, which positions the research on the intersection of two research domains, and in Section 1.3, which puts the research in the context of other PPM research. Next, Section 1.4 discusses the research design and provides an overview of the performed research studies. The structure of this dissertation is presented in Section 1.5 and Section 1.6 lists the articles that were published during the doctoral program.

- ***Definition 1: Business process***
  "A business process consists of a set of activities that are performed in coordination in an organizational and technical environment. These activities jointly realize a business goal." (Weske, 2007, p. 5)

- ***Definition 2: Business process model***
  We define a business process model as follows. Note how the term 'business' can be dropped to generalize this definition to any process model.
  "A business process model is a mostly graphical representation that documents the different steps that are or that have to be performed in the execution of a particular business process under study, together with their execution constraints such as the allowed sequence or the potential responsible actors for these steps."

- ***Definition 3: Process of process modeling***
  We define the process of process modeling as "the sequence of steps a modeler performs in order to translate his mental image of the process into a formal, explicit and mostly graphical process specification: the process model."

1.2. Positioning of the research

The research about the process of process modeling can be situated on a crossroad of two research domains. Process modeling is the part of Business Process Management that adheres to Conceptual Modeling. Both domains are described concisely below.

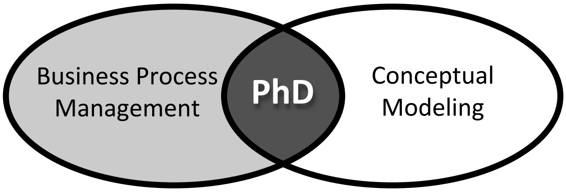


Figure 1.1. Positioning of the research on a crossroad of two research domains

Business Process Management

The primary research domain of the doctoral research is Business Process Management (BPM), marked in light grey in Figure 1.1. BPM is defined by Weske as follows.

- ***Definition 4: Business Process Management***
  "Business process management includes concepts, methods, and techniques to support the design, administration, configuration, enactment, and analysis of business processes." (Weske, 2007, p. 5)

In the late 19th century, management principles were developed within the manufacturing industries that pursued labor productivity based on empirical analyses, called Taylorism. A result of this development was that factory workers became pure specialists of only a part of a process, thus unconsciously introducing for each process the need for process management. Later, in the late 20th century Business Process Management emerged from these principles, as a discipline that targets processes over the whole business context, i.e., across the boundaries of functional units. (Dumas et al., 2013) A great deal of the innovations in the field was inspired by developments in quality management, which accelerated after the Second World War. Figure 1.2 presents a brief overview of BPM research developments, which are described below.

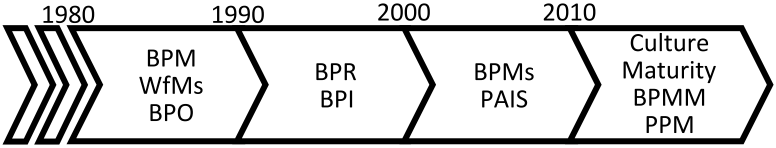


Figure 1.2. Brief overview of BPM research

Business Process Management foundations. In 1776 Adam Smith suggested the division of work in branches, each responsible for a series of tasks, which formed the basis for what is now considered a *business process* (A. Smith, 2014, originally published in 1776). In the early 1980s the basic concepts of *Business Process Management* (BPM) emerged. William Edwards Deming proposed the Plan-Do-Check-Act cycle for process control (Deming, 1982). The phases of this cycle can still be recognized in the *BPM lifecycle* (e.g., Analyze-Design-Implement-Manage-Improve) (Weske, 2007), which defines the activities included in Business Process Management, and their desired order.

Workflow Management and Business Process Orientation. In the late 1980s the developments in information technology facilitated the development of tool support for process execution. These tools were called *Workflow Managements systems* (WfMs) (Jablonski & Bussler, 1996). Improvement opportunities were revealed after studying process data that became available due to this widespread introduction of information technology (Drucker, 1988; Porter & Millar, 1985), which was later called *Business Process Orientation* (BPO) (Mccormack & Johnson, 2001).

Business Process Reengineering and Business Process Improvement. In the 1990s a fundamentally new view was taken on Business Process Management by Hammer & Champy, when they introduced *Business Process*

Reengineering (BPR), advocating to take the courage to radically redesign the processes in the company from scratch (Hammer & Champy, 1993). Thomas Davenport subscribed to this vision and provided more practical advise for companies concerning the related information technology challenges of process redesign (Davenport, 1993). James Harrington defined *Business Process Improvement* (BPI) (Harrington, 1991) as a way of applying novel quality management principles in services and processes (i.e., the Theory of Constraints, Lean, and Six Sigma).

Business Process Management Systems. In the 2000s, inspired by the principles of Total Quality Management, Champy recognized the need to involve all stakeholders in process management and improvement, including employees, suppliers, and customers (Champy, 2002). Driven by globalization of the economic context and by extensive customization, attention shifted towards process agility and automation (H. Smith & Fingar, 2003). Chang described how all these evolutions converged at a technical level to a need for dedicated tool support with an additional focus on management, i.e., Business Process Management systems (BPMs) (Chang, 2005) and Process Aware Information Systems (PAIS) (Dumas et al., 2005). Further, a more balanced approach arose that combined the principles of radical and incremental changes (Zhao & Cheng, 2005).

Business Process Culture and Business Process Maturity. The current evolutions in the field of Business Process Management lift the field to a more holistic understanding. The importance of culture is recognized by Jeston & Nelis (2008), who state that processes are *"the central core from which business is conducted, so long as they are supported by the people within the organization"* (p. 4). Business Process Maturity *"indicates how well an organization can perform based on its business processes"* (Van Looy et al., 2014, p. 188). Various Business Process Maturity Models (BPMM) were developed that represent the consecutive stages of maturity level and the prevailing improvement strategies for each level (Mccormack et al., 2009).

Process of Process Modeling. Almost as long as people are studying processes (i.e., from the late 19th century), some sort of process models were constructed (Marsh, 1975). The models followed the described trends from manufactory focused Gantt charts in the early 20th century (Gantt, 1913), over the flowcharts in the mid 20th century (Goldstine & von Neumann, 1947), towards the business process models from the late 20th century on. In the late 1990s particular studies investigated ways to measure and improve the quality of process models and in the 2010s researchers started to investigate the process of process modeling (see Section 1.3).

Conceptual Modeling

Within the BPM research field an important stream of research focuses on understanding and developing solutions for business process description and design. The description of an existing process and design of an envisioned process is mostly represented in a graphical model, which we previously referred to as the process model (see Definition 2). As a process model is a kind of conceptual model, our research is also related to the research domain of Conceptual Modeling. Conceptual modeling is defined by Mylopoulos as follows.

- ***Definition 5: Conceptual Modeling***
  "Conceptual modeling is the activity of formally describing some aspects of the physical and social world around us for purposes of understanding and communication." (Mylopoulos, 1992, p. 3)

(Business) process models. In Information Systems conceptual models thus represent the domain to be supported by the systems, independently of the technology that is or will be used (Olivé, 2007). The models describe the domain concepts, the properties of and the relations between these concepts. In the context of Business Process Management the domain is a business process. Hence, the models - (business) process models - represent the concepts of interest of a business process, the properties of these concepts, and the relations between the concepts. Examples of relevant concepts are the activities that are to be performed as part of a business process and the events that initiate these activities. Examples of relevant properties are the durations of the activities. Examples of relations between concepts are the sequence relations between the activities of a process. Figure 1.3 shows an example of a process model: it shows next process elements: activities (rectangles), events (circles), routing constructs (diamond), and the sequence in which these can occur (arrows).

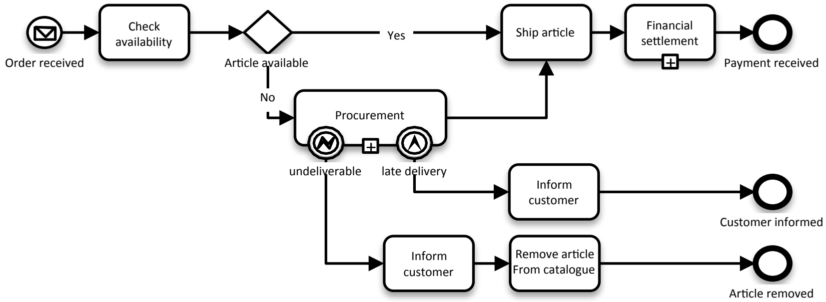


Figure 1.3. Example of a process model in BPMN notation (From BPMN Quick Guide, OMG, 2015)

Process model types. Different types of process models exist. Process models are usually *graphical* (e.g., BPMN), but there are also pure *textual* process models (e.g., LTL business rules). Further, the majority of process models represent mainly the *control flow*, i.e., the order of activities. Other process models target *data flow*, *organizational structure*, *process interactions*, etc. Next, whereas *imperative* process models describe all possible execution alternatives of the process explicitly, *declarative* process models describe the constraints that limit the possible alternatives (Goedertier et al., 2013).

Process model languages. These different model types are reflected in the variety of modeling languages that are used. A model language describes formally which concepts are allowed in the model and their meaning (i.e., the semantics), and which (combinations of) symbols can be used to represent them (i.e., the syntax). Process model language research evaluates and compares industry standards such as UML, BPMN and EPC in terms of ontological clarity and completeness, graphical quality, and cognitive efficacy (Börger, 2012; Figl et al., 2009; Moody, 2009). In 1962 Carl Adam Petri developed the Petri-net notation (Petri, 1962) as a modeling tool supporting both practitioners and theoreticians by combining graphical and mathematical elements (Murata, 1989). Petri-net research targets mainly the investigation of the usefulness of Petri-nets to formalize several aspects of process models through execution semantics, and the development of variants and extensions, such as colored Petri-nets, WF-nets and YAWL (Van der Aalst & Ter Hofstede, 2002, 2005).

Quality of conceptual models. Various quality dimensions and variables are proposed in literature. Quality of an artifact in general can be defined as fit-for-purpose (Juran & Gryna, 1988). In the context of conceptual modeling, quality is often divided into syntactic quality, semantic quality and pragmatic quality (Lindland et al., 1994). *Syntactic quality* indicates to which degree the symbols of the modeling language were used according to the rules of the language. *Semantic quality* indicates how adequate the model represents the modeled phenomenon in terms of correctness and completeness. *Pragmatic quality* indicates the extent to which the users of the model understand the model as intended by the modeler. Lately, more extensive quality frameworks were developed that incorporate for example the relation of the model with the knowledge of the modeler or the model reader: e.g., CMQF (Nelson et al., 2012), COGEVAL (Rockwell & Bajaj, 2005), and SEQUAL (Krogstie et al., 2006). Specifically for process models, many quality measures are defined that quantify (an approximation of) one or more quality dimensions. Instead of discussing a selection of these metrics here, we refer to rather complete literature reviews of research (Sánchez-González et al., 2013) and metrics (Mendling, 2008) of process model quality.

1.3. Process of process modeling

The aim of research into the Process of Process Modeling (PPM) is to improve process model quality by investigating how the process of creating process models can be improved. During this process, two parallel sub-processes are executed; (i) gathering knowledge about the process to form a mental image, and (ii) translating the mental image into a formal representation in the form of a process model (Hoppenbrouwers et al., 2005). Whereas originally both sub-processes were considered as the Process of Process Modeling (PPM) (e.g., Soffer et al., 2012), current research seems to target mainly the latter sub-process (see Definition 3). Below, an overview is provided of the current state-of-the-art in PPM research.

Measurement. PPM research accelerated when a tool was developed at the University of Innsbruck that was able to capture the operations of the modeler while modeling, i.e., Cheetah Experimental Platform (Pinggera, Zugal, & Weber, 2010). The tool records the creation, movement, deletion, and (re)naming of events, activities, gateways, and edges. It allows replaying (parts of) the modeling, and the collected data facilitated the empirical study of the PPM. Further efforts were made to diversify measurements with estimates for mental effort using self-rating scales (Pinggera et al., 2014) or based on eye-movement (Pinggera, Furtner, et al., 2013) or heart-rate (Zugal et al., 2012).

Visualization. In order to make it easier to get insights in the collected data, visualizations are developed. The PPM can be visualized by Modeling Phase Diagrams, which represent the course of three PPM phases: comprehension, modeling and reconciliation (Pinggera, Zugal, et al., 2012). An extended version is proposed that also displays mental effort during modeling (see Figure 1.4).

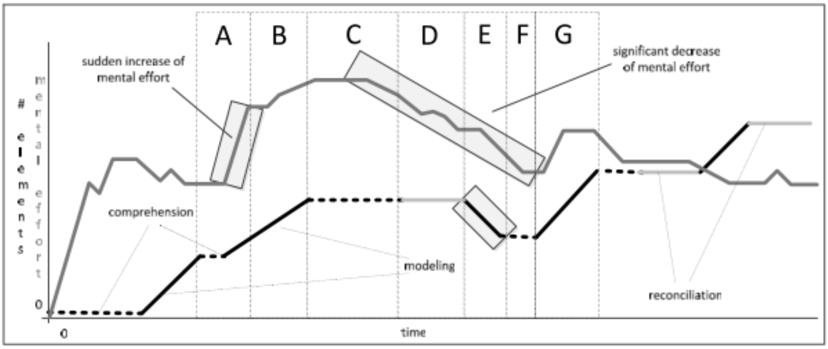


Figure 1.4. Modeling Phase Diagram with mental effort (from Pinggera et al., 2014)

Process modeling patterns. The measurement and visualization of PPM data assisted researchers in discovering modeling patterns, which we define as follows.

- ***Definition 6: Process modeling pattern***
  A process modeling pattern is the description of a recurring set of operations as part of the modeling process (e.g., creating split and according join gateways in pairs).

- ***Definition 7: Process modeling style***
  A process modeling style is a more high-level process modeling pattern that describes a particular way of creating an entire process model, focusing on multiple modeling aspects (e.g., fast modeling with few reconciliation operations).

Whereas modeling styles thus relate to complete approaches of constructing an entire process model, modeling patterns can also be more specific and can also describe particular modeling actions. Pinggera, et al. (2014) describe three modeling patterns, which they called PBPs (PPM Behavior Patterns). First, there is a difference between the times the modelers take before they start working on the model in the tool. Second, some modelers delete more of the model elements than others. Third, a difference is observed in how many phases the modelers use to layout the model. Next, Pinggera, et al. (2013) discovered three modeling styles, i.e., (i) slow modeling with more reconciliation, (ii) faster modeling with less reconciliation, and (iii) slow modeling with less reconciliation. They concluded that the applied style is partly dependent on the modeling task and partly on the modeler.

Relation between the PPM and process model quality. Previous research showed that certain aspects of process modeling that are relevant during the PPM can be linked to the quality of the produced process model, such as the structuring of the input document for the modeler (Pinggera, Zugal, Weber, et al., 2010), the mixture of textual and graphical elements of the modeling language (Recker et al., 2012), or the social distance of the modeler towards the modeling domain (Kolb et al., 2014). Also, a wide range of studies were performed to assess how tools can aid model understanding during or after modeling, e.g., syntax highlighting (Reijers et al., 2011), improving aesthetics of symbols or lay-out of the model (Figl et al., 2013; Purchase, 1997), hierarchical expansion of process models (Reijers & Mendling, 2008), and adding semantic annotations (Francescomarino et al., 2014).

Improving the process of process modeling. Concerning concrete process modeling instructions, Guidelines of Modeling (GoM) present general guidelines in terms of desired outcome (Becker et al., 2000). Similarly, Seven Process Modeling Guidelines (7PMG) provide more precise instructions about the model to produce (Mendling, Reijers, et al., 2010). Recently, studies emerged that investigate how process modelers would benefit of reusing process model fragments (Koschmider & Reijers, 2013; I. Wolf & Soffer, 2014) and how the PPM can be supported by providing change patterns (Reichert & Weber, 2013; B. Weber et al., 2008, 2014). Change patterns describe a set of modeling operations that together perform a high-level change to the model, such as replacement of a process fragment. Also, concrete step-by-step process modeling methods are used in practice (e.g., in Silver, 2011).

Overview. Table 1.1 provides an overview of current PPM research papers (i.e., papers that take a process view on modelers individually constructing one process model). It indicates their focus according to the discussed topics above, the publication type, and the stage of the research.

Table 1.1. Current PPM research papers
(excluding the papers that are part of this dissertation)

	Measurement	Visualization	Patterns	Relation with quality	Understanding	C: conference J: journal	Research stage (*)
(Francescomarino et al., 2014)				X		C	Observations
(Kolb et al., 2014)				X	X	C	Theory
(Pinggera, Zugal, & Weber, 2010)	X					C	Tool
(Pinggera, Soffer, et al., 2012)			X			C	Observations
(Pinggera, Zugal, et al., 2012)		X	X			C	Tool
(Pinggera, Furtner, et al., 2013)	X					C	Exploration
(Pinggera, Soffer, et al., 2013)			X		X	J	Theory
(Pinggera et al., 2014)	X		X		X	C	Idea
(Recker et al., 2012)			X	X	X	J	Theory
(Sedrakyan et al., 2014)	X		X			J	Observations
(Soffer et al., 2012)	X					C	Exploration
(B. Weber et al., 2013)					X	C	Exploration
(Zugal et al., 2012)	X					C	Evaluation
(Zugal & Pinggera, 2014)	X					C	Exploration
(B. Weber et al., 2014)					X	C	Theory

(*) Idea *(research proposal)* > Observations *(data interpretation)* > Exploration *(data analysis)* > Theory/Tool *(developed and evaluated knowledge/artifact) or* Evaluation *(of existing technique)*

1.4. Research design

As can be noticed from the overview of PPM papers in Table 1.1, the PPM research domain is a young domain that has been growing together with the research comprised in this dissertation. Because of the early stage of the research domain, the doctoral research started with explorative research and the overall objective was mainly curiosity-driven. As research progressed, the objectives evolved and became more specific. Therefore, the research design is discussed study per study. For each study first the objective of the study is discussed, followed by the research execution method, and concluding with an overview of the contributions of the study. A summary is presented at the end of the section in Figure 1.5.

Study 1. Visualization

The overall objective of the doctoral research is to provide scientific knowledge about the PPM, which will facilitate the development of techniques and tools that improve process modeling quality. Therefore, we were initially interested in how people construct process models.

- ***Research Objective 1*. Build knowledge about how people construct process models**

Already at an early stage of the research, it was noticed that there are different ways to approach modeling and the first goal was to try to reveal a number of these approaches in terms of concrete modeling patterns. One good way to detect such patterns is by visually representing the modeling approaches (Vessey, 1991). Because the existing visualization, i.e., Modeling Phase Diagram (Pinggera, Zugal, et al., 2012), already aggregates the data to the level of modeling phases, another visualization was searched for, which could represent the raw data that was collected by various researchers of the PPM domain. Inspiration was found in a process mining technique, called Dotted Chart, which was redesigned to support cognitive effective detection of process modeling patterns. The newly developed visualization, called PPMChart, represents the operations that a modeler performs in the modeling tool while constructing a single process model.

The design science research method was used to develop and evaluate the PPMChart visualization and to build the requested knowledge (Hevner et al., 2004; Peffers et al., 2007). The problem identification and solution objective definition were guided by 9 design principles for cognitive effective representations defined in literature (Moody, 2009). Next, the visualization was developed as a chart containing colored and shaped dots that represent the creation, movement, deletion and alteration of the model elements. The position of the dot in the visualization specifies the time of the operation and links the operation to a certain model element.

The usefulness of the PPMChart visualization was demonstrated by describing twenty two concrete process modeling patterns classified in ten categories (i.e., targeting ten different aspects of the PPM), which were discovered after studying a total of 357 different process modeling executions represented in PPMCharts. Thirteen general observations were described about the occurrence of the discovered patterns in the dataset. A qualitative evaluation of the PPMChart was performed through the observation and interviewing of six academic researchers with varying levels of research expertise (i.e., a subset from the intended users of the visualization) while working with the PPMChart implementation in ProM. It was concluded that the visualization is useful and more cognitive effective than existing alternatives.

- ***Contribution A*. PPMChart visualization**
- ***Contribution B*. Description of 22 process modeling patterns covering 10 aspects of PPM**
- ***Contribution C*. Description of 13 observations related to the 22 patterns**

Study 2. Exploration

After an initial understanding was formed of how people construct process models, the objective of the research was revised to a deeper understanding of not only how people construct process models, but also about the relation of their approach with the quality of the produced model.

- ***Research Objective 2*. Build knowledge about the relation between how people construct process models and the quality of the produced process model**

Therefore, the visualization was used in an explorative study that compared the identified modeling patterns with the corresponding produced process models. Three PPM aspects were selected for further investigation, i.e., structuredness, movement, and speed of process modeling. The 8 patterns related to these aspects were described in more detail and the link with process model quality was studied. It was argued that these aspects potentially influence that part of process model quality that relates to the cognitive functioning of the modeler during the PPM (as opposed to knowledge-related quality issues). Therefore, a metric was defined to measure this aspect of process model quality, i.e., the perspicuity metric. This metric is a binary metric that indicates if a process model has syntax errors caused by cognitive failure (disregarding those errors that originate in imperfect knowledge of the modeling syntax by the modeler).

Modeling patterns from 103 process modeling executions, represented by PPMCharts, were compared with their corresponding process models in order to discover potential links. This resulted in the selection of three potentially interesting links and the description of three concrete conjectures that link the quality of the constructed model to the structuredness of the modeling process, the amount and spread of move operations on model elements and the overall modeling speed. Simple statistics were performed on these relations, which confirmed empirically that the visually discovered links were indeed present in the data.

- ***Contribution D. Refinement of the 8 process model pattern descriptions from study 1 concerning structuredness, movement, and speed***
- ***Contribution E. Description of 3 conjectures about the relation between these patterns and process model quality***
- ***Contribution F. Definition of the perspicuity metric***

Study 3. Theorization

Keeping the overall objective in mind of building knowledge about the PPM aimed at process model quality improvement, the potential impact of investigating these conjectures was assessed. It was concluded that primarily the relation between structured process modeling and improved modeling quality deserved further attention. Potentially, a more structured approach helps avoiding mistakes during process modeling. Therefore, the research objective of this study is a further refinement of the previous ones.

- ***Research Objective 3. Build knowledge about why people make mistakes during process modeling and why structured process modeling can help avoiding mistakes***

A theory was developed according to the behavioral science research paradigm (Gregor, 2006; March & Smith, 1995). First, utilizing the PPMChart visualization, 118 process modeling executions were studied, which has lead to the identification of four concrete process modeling styles related to structuring of the modeling process. Further, we defined six general observations and three impressions about these styles and their relation with the effectiveness (i.e., model quality) and efficiency (i.e., modeling speed and effort) of modeling.

Based on the inductive knowledge represented by the observations and impressions, the theory was developed in a deductive way integrating theories from cognitive psychology and aiming to explain the observed behavior. The developed theory is called Structured Process Modeling Theory (SPMT). It states that process modeling is more effective and more efficient, when the modeler (i) serializes the modeling (ii) in a structured way (iii) that fits with his cognitive profile. Serialization means that the modeling process is divided in subtasks, which are executed consecutively rather than simultaneously.

A discussion of the potential utility of the theory was provided and an evaluation of consistency was performed by asserting that the theory could be used to explain additional observations from a new dataset containing 143 modeling executions. One additional process modeling style was described and its beneficial impact on modeling quality could indeed be explained by the SPMT.

- ***Contribution G. Description of 4 + 1 process modeling structuring styles***
- ***Contribution H. Description of 6 observations and 3 impressions about the 4 defined structuring styles and their relation to modeling efficacy***
- ***Contribution I. The Structured Process Modeling Theory***

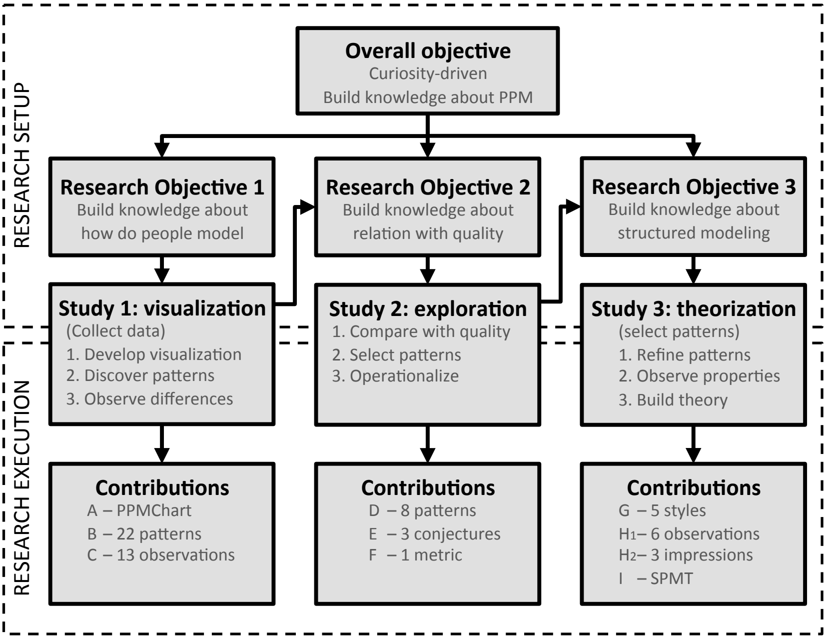


Figure 1.5. Research objectives, studies and contributions of the PhD

1.5. Structure of the PhD

This dissertation consists of three parts, i.e., the introduction (Chapter 1), the body of the doctoral dissertation (Chapters 2, 3, 4), and the conclusion (Chapter 5). Each chapter of the body of the PhD was written as a self-contained research paper. These chapters can be read independently from each other.

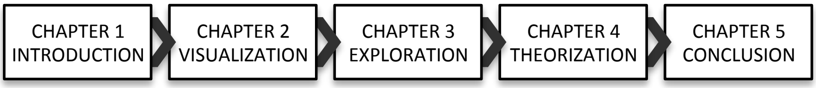


Figure 1.6. Structure of the doctoral dissertation

- **Chapter 1. Introduction.** *This introductory chapter describes and defines the research context, the research objectives, the studies and contributions, and provides an overview of publications published during the doctoral research program.*

- **Chapter 2. Visualization.** *This chapter addresses research objective 1 through the presentation of the design science research performed to develop the PPMChart visualization, which facilitates the detection of modeling patterns and building knowledge about how people construct models.*

- **Chapter 3. Exploration.** *This chapter addresses research objective 2. It is discussed which relations between the modeling process and the modeling result were revealed by comparing PPMCharts and process models of various modeling executions. We found empirical indications of a relation between structuredness, movement and speed of modeling on the one hand and process model quality on the other hand.*

- **Chapter 4. Theorization.** *This chapter addresses research objective 3 through the presentation of the behavioral science research that was conducted to develop the Structured Process Modeling Theory, which relates the degree, structuredness and fit of process model serialization with the effectiveness and efficiency of process modeling.*

- **Chapter 5. Conclusion.** *The conclusion summarizes and discusses the findings, reflects on the methodological aspects of the doctoral research, and provides an outlook on current and future work based on the limitations of the research.*

1.6. Publications

Parts of this dissertation have been presented at international conferences or have been published in international journals. Below is a list of all publications and conference contributions that were published in the course of the doctoral program (including articles not related to the doctoral research).

Publications in international journals

Indexed by Web Of Science

- *J. Claes, I. Vanderfeesten, F. Gailly, P. Grefen, G. Poels, The Structured Process Modeling Theory (SPMT) A cognitive view on why and how modelers benefit from structuring the process of process modeling, Information Systems Frontiers, Vol 17(6), 2015. [Chapter 4]*
- *J. Claes, I. Vanderfeesten, J. Pinggera, H.A. Reijers, B. Weber, G. Poels, A visual analysis of the process of process modeling, Information Systems and e-Business Management, Vol 13(1), p. 147-190, 2015. [Chapter 2]*
- *S. De Cnudde, J. Claes, G. Poels, Improving the quality of the Heuristics Miner in ProM 6.2, Expert Systems with Applications, Vol 41(17), p. 7678-7690, 2014. [not included]*
- *J. Claes, G. Poels, Merging Event Logs for Process Mining: A Rule Based Merging Method and Rule Suggestion Algorithm, Expert Systems with Applications, Vol 41(16), p. 7291-7306, 2014. [not included]*

Publications in national journals

Not peer reviewed

- *J. Claes, M. Jans, Process mining: get your processes out of the black box, The Internal Auditor Compass, IIA Belgium Magazine, 2012. [not included]*

Publications in international conference proceedings

Indexed by Web Of Science

- *J. Claes, F. Gailly, G. Poels, Cognitive Aspects of Structured Process Modeling, Proc. CAiSE '13 Workshops, LNBIP 148, Springer, p. 168-173, 2013. [Chapter 4]*
- *J. Claes, I. Vanderfeesten, H.A. Reijers, J. Pinggera, M. Weidlich, S. Zugal, D. Fahland, B. Weber, J. Mendling, G. Poels, Tying Process Model Quality to the Modeling Process: The Impact of Structuring, Movement, and Speed, Proc. BPM '12, LNCS 7481, Springer, 2012, p. 33-48. [Chapter 3]*
- *J. Claes, I. Vanderfeesten, J. Pinggera, H.A. Reijers, B. Weber, G. Poels, Visualizing the Process of Process Modeling with PPMCharts, Proc. BPM '12 Workshops, LNBIP 132, Springer, 2012, p. 744-755. [Chapter 2]*
- *J. Claes, G. Poels, Process Mining and the ProM Framework: An Exploratory Survey, Proc. BPM '12 Workshops, LNBIP 132, Springer, 2012, p. 187-198. [not included]*
- *J. Claes, G. Poels, Merging Computer Log Files for Process Mining: an Artificial Immune System Technique, Proc. BPM '11 Workshops, Part1, LNBIP 99, Springer, 2011: p. 99-110. [not included]*
- *W.M.P. Van der Aalst, et al., Process Mining Manifesto, Proc. BPM '11 Workshops, Part1, LNBIP 99, Springer, 2011: p. 169-194. [not included]*
- *J. Claes, G. Poels, Integrating Computer Log Files for Process Mining: a Genetic Algorithm Inspired Technique, Proc. CAiSE '11 Workshops, LNBIP 83, Springer, 2011: p. 282-293. [not included]*

Other peer reviewed

- *J. Claes, I. Vanderfeesten, H.A. Reijers, J. Pinggera, M. Weidlich, S. Zugal, D. Fahland, B. Weber, J. Mendling, G. Poels, Tying Process Model Quality to the Modeling Process: The Impact of Structuring, Movement, and Speed, Proc. EIS '12, 2012 (abstract). [Chapter 3]*
- *J. Claes, G. Poels, Merging Computer Log Files for Process Mining: an Artificial Immune System Technique, Proc. EIS '11, 2011 (abstract). [not included]*

Presentations at international events (excluding presentations of aforementioned conference publications)

PhD symposia related to international reviewed scientific conferences

- *Structured Process Modeling, Doctoral Consortium ECIS conference, 3 June 2013, Geetbets, Belgium.*
- *Business Process Modeling in Support of Supply Chain Applications, Doctoral Consortium CONFENIS conference, 19 September 2012, Ghent, Belgium.*
- *Business Process Modeling in Support of Supply Chain Applications, Doctoral Consortium BPM conference, 12 September 2010, Hoboken, New Jersey, USA.*

FEB PhD Day Contributions

- *Why do people struggle with the complexity of constructing a process model?, PhD Day FEB, 23 May 2014, Ghent, Belgium.*
- *Structured Process Modeling, PhD Day FEB, 24 May 2013, Ghent, Belgium.*
- *Merging Log Files for Process Mining, PhD Day FEB, 24 May 2011, Ghent, Belgium.*

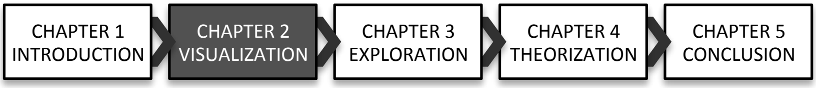
Other public presentations

- *The process of process modeling and process model quality, PhD Colloquium, 3 October 2013, Eindhoven, the Netherlands.*
- *Process Mining, City of Ghent, 25 September 2012, Ghent, Belgium.*
- *Process Mining, CONFENIS conference, 20 September 2012, Ghent, Belgium.*
- *Merging Event Logs in ProM, ProM meeting, 6 February 2012, Eindhoven, the Netherlands.*
- *Process Mining Open House Seminar, Ideas@Work, 1 August 2011, Brussels, Belgium.*

2

A visual analysis of the process of process modeling

Published as J. Claes, I. Vanderfeesten, J. Pinggera, H.A. Reijers, B. Weber, G. Poels, A visual analysis of the process of process modeling, Information Systems and e-Business Management, Vol 13(1), p. 147-190, 2015.



Summary. The construction of business process models has become an important requisite in the analysis and optimization of processes. The success of the analysis and optimization efforts heavily depends on the quality of the models. Therefore, a research domain emerged that studies the process of process modeling. This chapter contributes to this research by presenting a way of visualizing the different steps a modeler undertakes to construct a process model, in a so-called PPMChart. The graphical representation lowers the cognitive efforts to discover properties of the modeling process, which facilitates the research and the development of theory, training and tool support for improving model quality. The chapter contains an extensive overview of applications of the tool that demonstrate its usefulness for research and practice and discusses the observations from the visualization in relation to other work. The visualization was evaluated through a qualitative study that confirmed its usefulness and added value compared to the Dotted Chart on which the visualization was inspired.

Keywords. Business Process Management, Process Model Quality, Process of Process Modeling, Information Visualization.

2.1. Introduction

In the quest for knowledge about how to make process models of high quality, recent research focus is shifting from studying the quality of process models to studying the process of process modeling itself (PPM). The PPM is a phase in the process model development lifecycle where the mental view of the modeler on the process is formalized into a graphical process representation (Hoppenbrouwers et al., 2005) (see e.g. Figure 2.2). It encompasses the course of action taken by the modeler to design/construct a (business) process model consisting of start and end event(s), activities, gateways, edges, etc. Such a process model artifact is created by a stepwise design process; e.g., first putting a start event on the canvas, then an activity, then an arc connecting the start event and the activity, etc. The aim of PPM research is mainly to determine the characteristics of the process of process modeling that have a positive impact on process model quality. It is not about *what* is a good process model, but *how* can a good process model be created.

In order to be able to study current (implicit) modeling approaches of various modelers, since 2010 various datasets with data about such construction activities were collected in a series of observational modeling sessions executed at different universities in Europe (Pinggera, Furtner, et al., 2013; Weber, Pinggera, Zugal, & Wild, 2010; Weidlich et al., 2010)[1]. These data are being used in order to examine the PPM, mainly from a control flow perspective. The goal of the study presented in this chapter is to discover characteristics of various model constructions (i.e., instances of the PPM). Ultimately, these exemplar cases can be used to examine if more generic modeling patterns exist (e.g., Pinggera, Soffer, et al., 2013). The patterns can then be studied further to evaluate if certain modeling patterns impact process model quality in a positive way: i.e., the search for best practices that can be generalized in empirically validated process modeling guidelines and tool support (Claes et al., 2012).

This chapter describes the design of a tool to recognize and analyze these patterns in a cognitive effective way. The cognitive fit theory (CFT) states that a certain cognitive task can be optimally performed if the task material is represented appropriately (Vessey & Galletta, 1991). According to this theory, the proper instrument to discover relationships in datasets, are diagrams (Larkin & Simon, 1987). In other words, a visual representation of the data is believed to be a means to improve the efficacy of the cognitive task to discover patterns in the data (Fekete et al., 2008).

---

[1] An overview of these modeling sessions can be consulted at http://bpm.q-e.at/experiments.

Furthermore, humans excel in visual pattern recognition (Baird et al., 2003). Therefore, a visualization was designed to support researchers, practitioners and tool developers to get insights in the PPM to develop theory, training and tools for improving process model quality.

For the visualization described in this chapter, inspiration was drawn from the process mining research field (Van der Aalst, 2011). Process mining techniques make use of historical data of various process executions (i.e., process instances) to graphically represent and analyze a particular process (Weijters & Van der Aalst, 2001). The PPM is a typical type of process that also can be analyzed with process mining techniques. More specific, the visualization described here was based on the Dotted Chart (Song & Van der Aalst, 2007), which represents every recorded event of the different process instances in one diagram in such a way that patterns across multiple instances can graphically be discovered and at the same time one can zoom in on details about the events of a single instance. However, the Dotted Chart does not support the analysis of the PPM optimally. Hence, this chapter presents a modified implementation, i.e., the PPMChart.

To produce the PPMCharts, the existing Dotted Chart implementation was fully redesigned. Every property of the chart and configuration option of the tool was evaluated against cognitive principles to reassure cognitive effectiveness of the visualization and the tool in the context of PPM research. The visualization is then applied in different contexts, which results in an extensive list of 13 observations of which initial insights are discussed and that are linked to empirically tested hypotheses. A qualitative study shows that the visualization succeeds in its goal to be useful for the study of the PPM, with a higher cognitive effectiveness than the Dotted Chart.

The discovery of confirmed (causal) relations between the PPM and the quality of the resulting process model and the knowledge about the circumstances needed to take optimal advantage of these relations, can enable the efficient exploration of ways to help improve process model quality in general. Modelers can be trained to implement the optimal modeling strategy that maximizes their individual capacity of creating high quality models in a specific domain or situation. Tools can be complemented with the developed knowledge to excel in supporting modelers to increase model quality. The PPMChart is a cognitive effective instrument to explore the data in order to build the necessary knowledge for the development of such training and tool support.

This chapter reports on the development and application of the PPMChart visualization. The design science research method of Peffers, et al. is used to structure the chapter (Peffers et al., 2007). The *problem description* is described in Section 2.2. The details of the *developed* graphical representation and the implementation are the subject of Section 2.3. The extensive overview of applications exhibited in Section 2.4 serves as a *demonstration* of the usefulness of the visualization in concrete analyses. Section 2.5 presents the result of a qualitative *evaluation*. Limitations and implications are discussed in Section 2.6. Section 2.7 describes related work. Finally, Section 2.8 summarizes the chapter and discusses the need for and the value of this work.

2.2. Motivation

The goal of the research presented in this chapter is to develop a visualization that can assist the study of the process of process modeling (PPM) in a cognitive effective fashion. Cognitive effectiveness is defined as the speed, ease and accuracy with which a representation can be processed by the human mind (Larkin & Simon, 1987). The inspiration for the PPMChart visualization was drawn from the Dotted Chart (Song & Van der Aalst, 2007)[2]. It was perceived to have an optimal balance between representing information about the structure of the overall process and the timing and relation of individual events. First, the Dotted Chart is presented in Section 2.2.1. Next, Section 2.2.2 evaluates the Dotted Chart as a solution for graphical analysis of the PPM. It can be concluded that the Dotted Chart in its current implementation is inadequate for studying the PPM in a cognitive effective way and that an adapted visualization is needed.

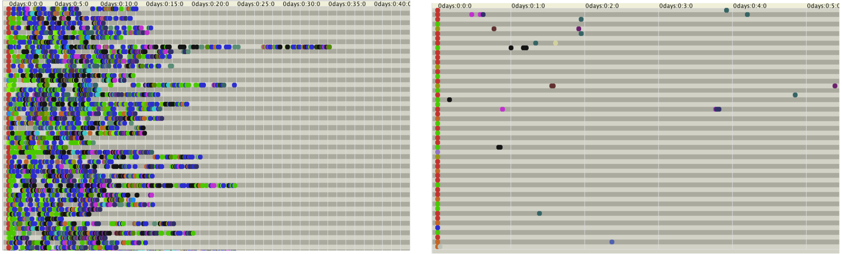
2.2.1. Dotted Chart

The Dotted Chart visualization displays the events of the instances of a process as colored dots on timelines. Each timeline corresponds with one particular execution instance of the analyzed process and each colored dot on the timeline corresponds with a specific event for that process instance. The color of the dot indicates which event happened, while the position of the dot on the timeline represents the time when the event occurred (see Figure 2.1a). It can be observed from Figure 2.1a that it is difficult to analyze a Dotted Chart that contains information of all recorded PPM instances. For example, it is not possible to know which dots represent different events on the same model element (e.g., creation, movement,

---

[2] See also Appendix A, which explains how our attention was directed to Dotted Chart to be used as inspiration.

renaming of a *particular* activity) without manually investigating attributes in the event log (i.e., without leaving the plug-in). Therefore, one could focus on a single PPM instance if the event log is split into multiple event logs (one for each instance). The events in these event logs can then be grouped per model element (rather than per PPM instance). In this case, a Dotted Chart taking one such event log as input represents the operations of only one PPM instance (see e.g. Figure 2.1b). We conclude, that it is possible to use the Dotted Chart for the visual study of the PPM. In the next section, we discuss the cognitive effectiveness of a visualization in general and of the Dotted Chart in particular.


(a) Full event log: multiple PPM instances

(b) Transformed partial event log: only one PPM instance

Figure 2.1. Example of a Dotted Chart for the full event log with multiple PPM instances and for an event log containing events of only one PPM instance (each line representing the operations on a different process model element).

2.2.2. Cognitive effectiveness of a visualization

Before the PPMChart visualization is described in more detail in Section 2.3, this section explains what is meant with a '*cognitive effective visualization*'. Moody collected nine concrete principles that can be applied to improve the cognitive quality of visual notations, which are presented and discussed below (Moody, 2009). The design principles are also used to evaluate the Dotted Chart (when used on an event log containing a single PPM instance). The conclusion is that even with a limited amount of data in the event log, the cognitive effectiveness of the Dotted Chart can be substantially improved for the study of the PPM.

Visual expressiveness

A graphical representation is visually expressive if it makes optimal use of the different graphical variables on which symbols can differ (Moody, 2009). Eight graphical variables are defined: shape, size, color, brightness, orientation, texture, horizontal position and vertical position (Bertin, 2010). Moreover, some graphical variables have a stronger impact on the cognitive load for interpreting the diagrams. Color is considered the most effective graphical variable (Lohse, 1993; Treisman, 1982; Winn, 1993), although in some situations it causes problems with visual perception (e.g., color blindness, black-and-white printers) (Moody, 2009). Therefore, other graphical variables should be used in combination with color.

The visual expressiveness of the Dotted Chart is rather low. With default settings, it makes no use of shape, size, brightness, orientation or texture. The various dots only differ in color and position. This can easily be improved by introducing the use of different shapes, brightness, size and/or texture of dots.

Perceptual discriminability

Perceptual discriminability advocates that symbols are clearly distinguishable and that the more two concepts differ from each other, the more the corresponding symbols should differ (Winn, 1990). In this context, the *visual distance* is determined by the number of graphical variables for which two symbols have different values and by the size of these differences (Moody, 2009).

The perceptual discriminability of the Dotted Chart is also rather low. The colors of the dots are assigned randomly to the event classes that are present in the event log, which means that two dots with similar colors (e.g. blue and violet), do not necessarily represent similar events. Therefore, to increase perceptual discriminability, it is proposed (i) to assign fixed colors to fixed operations, (ii) to choose colors in such a way that similar events get similar colors, and (iii) to introduce other graphical variables such as shape and brightness in order to distinguish more easily between different dots. Note that assigning fixed colors to the operations is only possible when there is a fixed, known set of possible occurring operations, which is the case in the study of the PPM.

Graphic economy

There is a limit on the amount of different *values* for each graphical variable to assure that these design principles increase cognitive effectiveness (Nordbotten & Crosby, 1999). The span of absolute judgment, the amount of distinct perceptual values for each graphical variable, is estimated at seven (Miller, 1956). Furthermore, the amount of different objects that can be distinguished at a glance, i.e., the span of attention, is estimated at six objects (Miller, 1956).

It can be observed from the examples in Figure 2.1 that the Dotted Chart violates the graphic economy principle. There are too many colors, which indeed hampers interpretation of the chart. In order to increase graphic economy, we propose to reduce the number of used colors, which can be easily compensated with the introduction of color shades (i.e., brightness) as discriminating variable.

Dual coding

While a graphical representation is a lot more effective than a textual representation (for processing information), the combination of both results in an even higher cognitive effectiveness (Paivio, 1990).

The Dotted Chart is divided in configurable time intervals of which *start and end date and time are textually displayed* on top of the chart. Each line also represents the *instance identifier at the beginning of the line*[3]. Information on selected dots is displayed in *tooltip text* when the mouse hoovers over the selected dot(s). We conclude that the Dotted Chart makes sufficient use of this design principle.

Semiotic clarity

Semiotic clarity means that every concept is represented by exactly one symbol and every symbol represents exactly one concept (Goodman, 1968).

Every event class of the event log is represented in a Dotted Chart as a dot with a unique color. Moreover, each dot has a unique position. Therefore, the Dotted Chart has maximal semiotic clarity. Nevertheless, it can sometimes be desired to introduce symbol deficit (i.e., use the same symbol for different concepts) to increase graphic economy (Moody, 2009).

Semantic transparency

A visualization has optimal semantic transparency if a novice would be able to guess the meaning of each symbol (Moody, 2009). This can be achieved through natural mappings: i.e., "*taking advantage of physical analogies and cultural standards*" (Norman, 2002, p. 23).

The Dotted Chart makes use of only three graphical variables: color, horizontal position and vertical position. When utilizing Dotted Charts for presenting the PPM, only one of these three variables is semantically transparent: the horizontal position indicates the timing of the corresponding event. The meaning of each color is not transparent to the reader of the chart (but can be derived from the color legend in a separate

---

[3] For readers that are familiar with process mining: this is the trace identifier in the event log.

tab page). Similarly, it is difficult to know which horizontal line corresponds to which instance in the event log. For this, the reader first needs to zoom in on the chart to reveal the instance identifier displayed on each line[4]. Then, the reader can use this identifier to look up the necessary information in the attributes in the event log.

Complexity management

To cope with perceptual and cognitive limits (Miller, 1956; Novak, 2002), it is encouraged to reduce complexity by modularization (divide the diagram in smaller subsystems) and hierarchical structuring (make separate diagrams of the same information at different levels of abstraction) (Moody, 2009; R. Weber, 1997).

We suggest to only represent one PPM instance at a time in each chart. This way, the timelines can represent the different model elements (as in Figure 2.1b), rather than different instances of the PPM (as in Figure 2.1a). This can be achieved by splitting the event log in multiple event logs, each containing information about only one PPM instance. Further, it is possible in the Dotted Chart to customize the chart to abstract from certain differences or tailor the view to a certain analysis. All the same, complexity can further be reduced by filtering the displayed information. This can optimally be managed from within the plug-in, rather than at event log level (which is currently the only option in the Dotted Chart implementation).

Cognitive integration

When different diagrams are used, explicit mechanisms should exist to support the integration of these diagrams (Hahn & Kim, 1999; Kim et al., 2000).

When different PPM instances are represented by different charts (rather than aggregating them in only one chart), the need for cognitive integration mechanisms emerges. A first step towards cognitive integration might be a *uniform color, shade and shape coding and time scaling* between different charts. Without uniform coding (as in the Dotted Chart), comparing or combining information of multiple charts is not cognitive efficient.

---

[4] Due to a misalignment, this is not visible in a typical Dotted Chart without zooming in (this might be an unintentional bug).

Cognitive fit

The optimal representation of data depends on the task it supports and on the user of the visualization. For the same representation, the cognitive load is greater for novices than for experts (experts in working with that particular visualization) (Vessey & Galletta, 1991). Similarly, depending on the particular analysis the user needs the representation to support, a different view on the data is desired.

The Dotted Chart has various *configuration options*, which facilitate to tailor the appearance of the chart according to the needs of the analysis. However, a mechanism to filter the information that is displayed in the chart is cumbersome, because it involves leaving the plug-in, using a filter plug-in of the framework on the event log, regenerating a chart from the filtered log and reconfiguring the chart to customize the view.

Conclusion

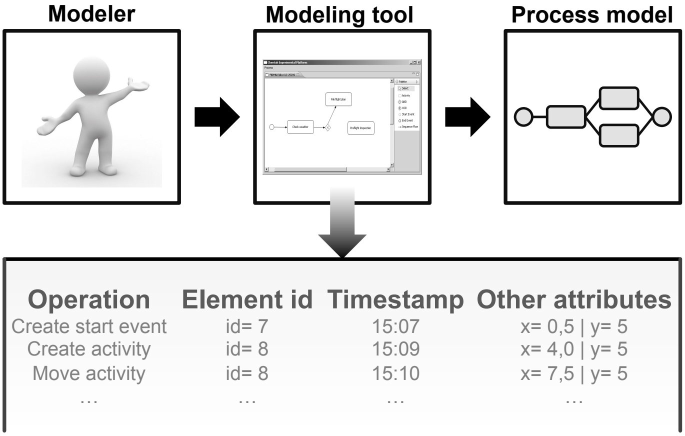
The Dotted Chart can be used in its current form to study the PPM. To maximize complexity management, it seems appropriate to represent only one PPM instance at a time. However, according to the presented design principles for visual notations, cognitive effectiveness of the Dotted Chart can be substantially improved in the specific context of this research (even if an event log containing data about a single PPM instance is provided to the plug-in). For 7 out of 9 principles concrete suggestions are formulated, which formed the basis for the extension of the Dotted Chart presented in this chapter. Note that every suggested improvement is specific for the PPM and could therefore not be incorporated in the original Dotted Chart plug-in. Because the improvements are specific for the context of the study of the PPM, it was decided to call the improved charts PPMCharts.

2.3. The PPMChart visualization

The PPMChart visualization graphically represents data of the process of process modeling (PPM) (see Section 2.3.1). The visualization uses timelines on which colored dots are associated with positions that correspond to the time when the corresponding PPM operations are recorded for the process model (see Section 2.3.2). The Dotted Chart Analysis plug-in was adapted and extended (see Section 2.3.3), which resulted in a new plug-in in the ProM tool that produces the PPMCharts (see Section 2.3.4).

2.3.1. Data requirements

The PPM is a human endeavor in which a modeler constructs a process model by drawing model elements such as activities, events, gateways, and edges on a canvas (see Figure 2.2). In order to be able to represent a PPM instance with the PPMChart visualization, data of the PPM instance need to be collected at a specific level (see Section 2.3.4). Therefore, it is convenient if a modeling tool with logging functionality is used for the construction of the process model. The PPMChart implementation is created under the assumption that it is possible to record data on every modeling operation on the canvas (e.g., create start event, create activity, move activity). Besides the *name* of each operation, the visualization needs two more attributes: the *identifier* of the model element on which the operation was performed and the *timestamp* of the execution of the operation. Possible other recorded attributes (such as the position of a model element on the canvas) are ignored by the visualization.



Operation	Element id	Timestamp	Other attributes
Create start event	id= 7	15:07	x= 0,5 | y= 5
Create activity	id= 8	15:09	x= 4,0 | y= 5
Move activity	id= 8	15:10	x= 7,5 | y= 5
…	…	…	…

Figure 2.2. The process of process modeling and attributes of the captured data

2.3.2. Visualization with PPMChart

The collected data about consecutive operations in a PPM instance are used to construct a PPMChart (see Figure 3.1). The horizontal axis represents a time interval of one hour by default. Vertically, each line represents one element of the process model, as it was present during modeling. The model element identifier is displayed at the beginning of the line. Each colored dot on the line represents one operation performed on the element.

- *The color of the dot corresponds with the type of operation: create (green), move (blue), delete (red) and (re)name (orange).*

- *The color shade and shape of the dot corresponds with the type of model element: activity (bright, box), event (very light, circle), gateway (dark, diamond) and edge (light, triangle).*
- *The position of the dot on the timeline corresponds with the time at which the operation was executed.*

The user can configure the order in which the timelines are presented. The default order of lines corresponds with the logical order from start event to end event of the elements in the process model (see Section 2.3.4).

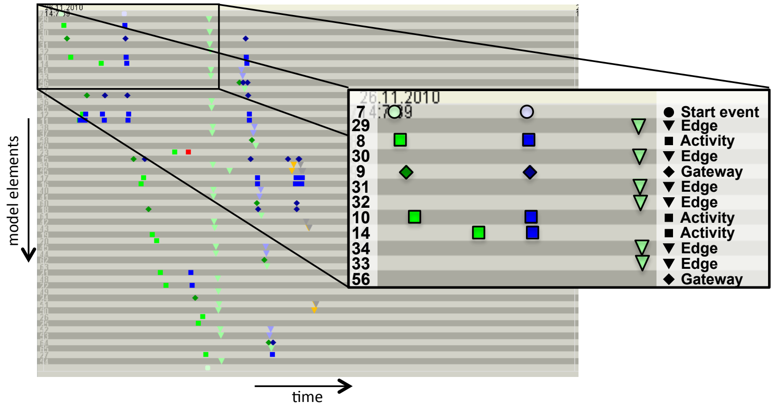


Figure 2.3. Visualization of the events in the construction of a single model [5]

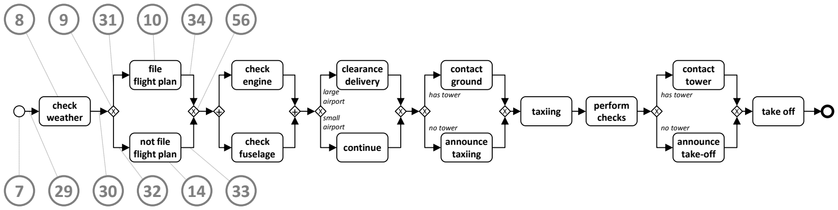


Figure 2.4. Process model as result of the modeling process in Figure 2.3 [6]
(Numbered circles indicate the model element id for comparison with highlight of Figure 2.3)

Figure 2.3 shows the different operations in the creation of the process model represented in Figure 2.4. The highlighted rectangle in Figure 2.3 displays the first operations on the left part of the model in Figure 2.4 (from start event to first XOR join gateway). On the first line, one can observe the creation of the start event (very light green circular dot). More to the right a very light blue circular dot represents a movement of that start event. The second line shows a light green triangle at the right that corresponds

---

[5] High resolution graphs in color of all figures in this chapter are available from http://www.janclaes.info/papers/PPMISeB.

[6] Download an animation showing how the PPMChart evolves during the model construction at http://www.janclaes.info/papers/PPMISeB.

with the creation of the edge that connects the start event and the first activity. Within the highlighted rectangle, different other light green triangles can be discovered on different lines. These dots represent the creation of other edges in the process model. The third line contains a bright green square shaped dot. This is the creation of an activity. More to the right, a blue square shaped dot indicates that this activity is moved later on. If one focuses on all colored dots in the rectangular selection, one can conclude that the creation of start event, activities and split gateway in the highlighted section is followed by an almost simultaneous movement of these model elements (vertical pattern of blue dots). Only later (i.e., more at the right), the edges that connect these model elements were created (light green triangles).

2.3.3. Differences with Dotted Chart

The PPMChart visualization differs from the classical Dotted Chart mainly in four ways:

(i) In contrast to the typical use of a Dotted Chart, the PPMChart displays information about only one process instance. Each timeline represents a particular model element of the constructed process model (i.e., activity, gateway, edge, etc.). The events in the PPM, represented by the colored dots, are the operations (i.e., creation, movement, deletion, etc.) performed on the particular model element represented by the timeline.

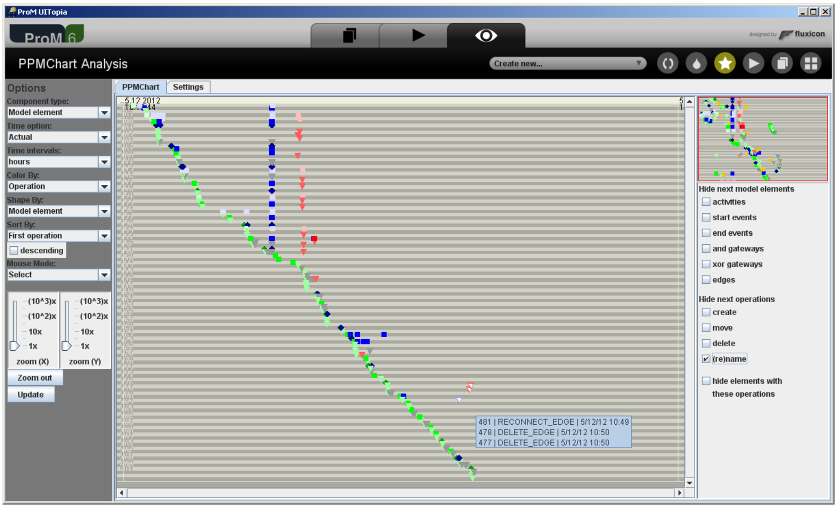
(ii) The PPM has a fixed set of possible operations and therefore, in a PPMChart, these operations are mapped on fixed default colors, color shades and shapes, which eases the visual comparison of different charts. For the same reason, every PPMChart shows initially information of the same timespan (i.e., one hour). Nevertheless, the option to change the color (shade) and shape coding of the dots and to zoom in or zoom out to influence the displayed timespan still exists.

(iii) For different graphical pattern analyses, the Dotted Chart can be sorted according to several sort options. Two sort options are added in the PPMChart implementation, which facilitate the study of the PPM with the charts.

(iv) The option to filter certain operations (individual dots) or model elements (individual timelines) enables the analyst to take different views at different abstraction levels on the data from within the PPMChart implementation. For example, one can focus on creation of model elements if other operation types are filtered out.

2.3.4. Tool support

The *PPMChart Analysis* plug-in[7] (see Figure 2.5) is an adapted version of the existing *Dotted Chart Analysis* plug-in in ProM. In the middle, the PPMChart is presented. At the left hand side, one can configure the view, and at the right one can filter the data. The previous version of this plug-in (Claes et al., 2013) remained closely to the Dotted Chart Analysis plug-in implementation in order to not confuse users that are familiar with that plug-in. The version described here was fully redesigned after feedback of various experts (i.e., participants to the BPM 2012 conference, international process modeling and visualization experts at various occasions). Every visualization property and tool setting was evaluated against principles of cognitive efficacy optimization.



Component type: Model element

Time option: Actual

Time intervals: hours

Color By: Operation

Shape By: Model element

Sort By: First operation

☐ descending

Mouse Mode: Select

481 | RECONNECT_EDGE | 5/12/12 10:49
478 | DELETE_EDGE | 5/12/12 10:50
477 | DELETE_EDGE | 5/12/12 10:50

Hide next model elements
☐ activities
☐ start events
☐ end events
☐ and gateways
☐ xor gateways
☐ edges

Hide next operations
☐ create
☐ move
☐ delete
☑ (re)name

☐ hide elements with these operations

Figure 2.5. Screenshot of the PPMChart window in ProM (Model 2012-184)

This section describes the data format for the PPMChart plug-in and the configuration options that were added to the Dotted Chart Analysis plug-in. A more detailed description about every configuration option of the implementation can be found in Appendix B.

Required format of the recorded data

For the sake of *complexity management* and *cognitive fit*, it was decided to define a PPMChart as a visualization to display information of a single process modeling effort. Therefore, the tool requires as input an event log containing data about a single PPM instance, grouped in traces per model

---

[7] The PPMChart Analysis plug-in for ProM 6 can be downloaded at http://www.janclaes.info/ppm.php.

element. Section 2.3.1 already explained which data of that PPM instance are needed. The required format of these data is described in this section.

In order to visualize the PPM, a fixed list was selected of possible operations in the construction of a process model (see Table 2.1). In our analyses and modeling sessions we build on a subset of the BPMN notation that can be used for the modeling. This subset was selected to correspond with the supported notation of our modeling tool Cheetah Experimental Platform (CEP) (Pinggera, Zugal, & Weber, 2010) and consists of six of the ten most used elements of BPMN according to (Zur Muehlen & Recker, 2008): start and end event, activity, XOR and AND gateway, and edge. It can be considered as the set of common constructs that are available in most process modeling languages (e.g., BPMN, UML Activity diagrams, Petri nets, Workflow nets, YAWL) (Marcello La Rosa et al., 2011).

Besides creation of these model elements, the visualization also includes changes in the model. Activities, events and gateways can be moved over the canvas or deleted. Edges can be deleted or reconnected, an edge can be rerouted through creation, movement and deletion of edge bend points, and the label of an edge can be moved. Finally, activities and edges can be named or renamed. Note that for the remainder of the chapter we assume only these modeling operations as part of the PPM (according to the recorded operations of the modeling tool), but our approach can easily be adapted for other modeling operation sets.

Table 2.1. Operations in the construction of a process model

Create	Move	Delete
CREATE_START_EVENT	MOVE_START_EVENT	DELETE_START_EVENT
CREATE_END_EVENT	MOVE_END_EVENT	DELETE_END_EVENT
CREATE_ACTIVITY	MOVE_ACTIVITY	DELETE_ACTIVITY
CREATE_XOR	MOVE_XOR	DELETE_XOR
CREATE_AND	MOVE_AND	DELETE_AND
CREATE_EDGE	MOVE_EDGE_LABEL	DELETE_EDGE
	RECONNECT_EDGE	
Other : NAME_ACTIVITY, RENAME_ACITIVTY, NAME_EDGE, RENAME_EDGE, CREATE_EDGE_BENDPOINT, MOVE_EDGE_BENDPOINT, DELETE_EDGE_BENDPOINT		

A plug-in for the well-known academic process mining framework ProM[8] was developed to facilitate the creation of PPMCharts. The input for most plug-ins in this tool is an event log. The xes file format for event logs for ProM is xml based and follows a certain hierarchical structure: a *process* consists of *traces* and each trace is a collection of *events*[9]. The process, traces

---

[8] The ProM tool can be downloaded at http://www.promtools.org.

[9] The xes file format of ProM is described at http://www.xes-standard.org.

and events can have attributes. In order to be able to create a PPMChart, the *events* in the log have to store information about operations on model elements and the events that correspond with operations on the same model element have to be bundled in one *trace*. The plug-in expects the *names* of events to correspond with the operations in Table 2.1. Further, each event should have an attribute that stores the *timestamp* of the execution of the event and an attribute *id* that matches with the name of its trace. It can be seen as the unique identifier of the model element.

Fixed default color, shade and shape coding

Visual expressiveness is increased by the addition of default shade and shape coding. In order to preserve *semantic transparency*, size, texture and orientation are currently not used as symbol discriminating factors. The selection of similar colors and shapes for similar operations improves *perceptual discriminability*. This introduces some redundant coding (i.e., the type of model element can be deduced from the color shade and from the shape) (Green & Swets, 1966), but increases perceptual pop-out (because almost each different operation has its unique color (shade), it is easy for the human brain to filter out specific operations/colors) (Quinlan, 2003; Treisman & Gelade, 1980).

The introduction of shades of colors reduces the amount of necessary colors, which adheres to the principle of *graphic economy*. The amount of used graphical variables raises (i.e., increased visual expressiveness), while the amount of *values* for each graphical variable can be limited to a lower number (i.e., graphic economy). Because the goal of the PPMChart is to reduce cognitive load and increase cognitive effectiveness for the study of data on the PPM, much importance was attached to this principle. The graphic economy of a representation with more than six different colors can be increased by (i) *increasing visual expressiveness*, (ii) *reducing semantic complexity*, and (iii) *introducing symbol deficit* (Moody, 2009). As a consequence, symbol deficit was introduced on operations on both types of events (i.e., start and end event) and both types of gateways (i.e., AND and XOR gateway). *Perceptual discriminability* advocates the choice for exactly these symbols to introduce symbol deficit, because they are most similar. However, the user can still change the color (shade) and shape of the dots such that both gateways and both events can be graphically distinguished.

Furthermore, *semantic transparency* is increased by selecting logical colors. Shades of green represent create operations and shades of red represent delete operations. For move operations, the third primary color was selected (i.e., blue). The selected shapes are similar to the BPMN symbol for the element type (square for activities, circle for events, diamond for gateways). Lastly, *cognitive integration* is facilitated by selecting fixed default coding. This resulted in the fixed default color (shade) and shape coding as presented in Table 2.2. However, the user can still modify the color (shades) and shapes in the *Settings* tab.

Table 2.2. Default colors, shades and shapes of the PPMChart Analysis plug-in.

Operation	Color	Model element	Shape	Color shade
Create	Green	Activity	Rectangle	Bright
Move	Blue	Event	Circle	Very light
Delete	Red	Gateway	Diamond	Dark
(Re)name	Orange	Edge	Triangle	Light

Fixed default time interval

By default, the time interval represented by a PPMChart is fixed, which makes it easier to compare time related issues between different PPMCharts (i.e., facilitating *cognitive integration*). The length of this interval is set to an arbitrary value of one hour, but this interval can be modified with the use of the zoom (X) button.

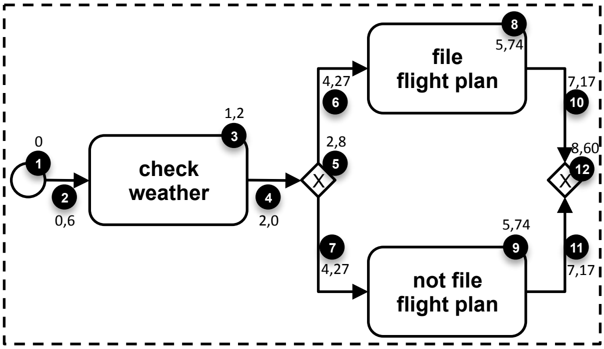
Sort options

Cognitive integration also promotes the addition of two sort options that help the user to find a link between the PPMCharts and the corresponding process model elements: '*Distance from start*' and '*Create order from start*' (see Figure 2.6). This subsection explains both additional sort options, as well as the option to sort by '*First operation*', which existed already in the Dotted Chart implementation and is used in two examples in this chapter.

Distance from start

The *Distance from start* sort option was added to sort the timelines according to the processing order of the corresponding model elements from start event to end event. This can be observed in Figure 2.6a, which represents the left part of the process model of Figure 2.4 and where the order of model elements is indicated with black circle annotations. For elements in parallel paths the order is rather arbitrary (see technical details below).

(a) Distance from start



(b) Create order from start

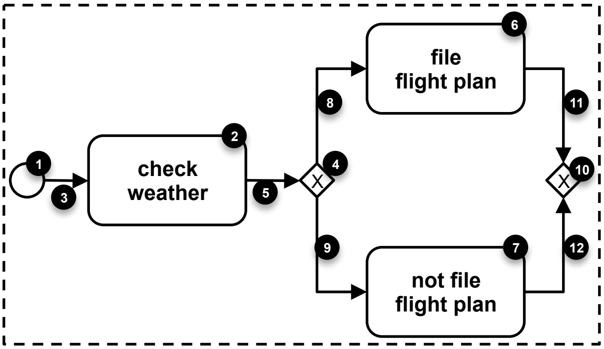


Figure 2.6. Additional sort order options in the PPMChart implementation

To determine this order technically, we used the notion of *length* of an arc, *path*, path *distance* and *shortest path*. We define the *length* of an arc as the graphical distance between start and end point of the arc, regardless the actual routing of the arc through possible bend points. A *path* is a route between two elements in the process model and summarizes the consecutive model elements that are to be passed when traversing the model from A to B through nodes and directed arcs that connect these nodes. The *distance* of a path is the sum of the lengths of the arcs in the path. If an arc is the first or last element in the path, only half of its length is included in the distance. The *shortest path* between two nodes is the path that starts in one of the nodes and ends in the other node with the lowest distance. The *distance from start* orders each model element according to the distance from the shortest path from the start event to the model element. The distance from start for each model element in Figure 2.6a is displayed right above or below the black circle annotation that indicates the order of the model elements. If two elements get the same distance value, the order of the related records in the log is preserved.

Create order from start

The *Create order from start* sort option was added under the assumption that in several modeling tools (e.g., ARIS, CEP) an edge connecting two nodes can only be created after the two nodes exist. Hence, it uses the same ordering mechanism as *Distance from start*, except that it puts each arc after the nodes it connects (see Figure 2.6b). This sort option was added to resemble (our interpretation of) the *logical* order of creating the model elements in a process model from start event to end event (i.e., from left to right in the example). Technically, while in the *Distance from start* ordering the value of the arc is the mean value of the nodes it connects, in the *Create order from start* ordering, it is the maximum value of the nodes it connects plus one.

First operation

The remainder of the article contains a lot of examples of PPMCharts. The majority of them apply one of the above sort options, but in two cases, it was useful to sort by *First operation*. This sort option orders the timelines according to the *actual* creation order of the model elements (the first operation of each element is its creation).

Filter options

In order to raise *cognitive fit* and for *complexity management*, the PPMChart can be filtered from within the plug-in to show or hide dots on the charts. Model element types can be selected to hide all operations on all model elements of that type (e.g., hide operations on edges). The timelines representing these model elements remain in the chart in order to preserve the positioning of the remaining timelines, but the concerned dots on these timelines are hidden. Next, it is also possible to hide specific operations by selecting the operation type in the filter panel (e.g., hide (re)name operations). Every dot that represents an operation of the selected type will be hidden. Finally, if the checkbox at the bottom is selected, all operations of model elements for which an operation exist of the selected type are hidden (e.g., hide deleted elements). Every dot on the timelines that contain at least one dot representing an operation of the selected type is removed from the chart.

Example

Figure 2.5 above shows an example of the implementation. The lines are vertically sorted by '*First operation*'. On the first line a circle indicates that a start or end event was created first. The last line represents the last created model element. It appears to be an edge (triangle). At one moment in time a lot of elements were moved simultaneously (perfectly vertical blue line formed by blue dots of different shapes), and somewhat later it can be observed that a lot of elements were deleted right after each other (almost vertical red line). The deleted elements were mainly events and edges (circles and triangles). Immediately after these operations, a high number of edges were created or reconnected (light green and gray rectangles below the red vertical line).

2.4. Application

To demonstrate the usefulness of the PPMChart visualization, it was applied on the collected data for many process of process modeling (PPM) instances of students. A PPMChart was generated for every PPM instance in the dataset. First, a description of the data collection is provided (see Section 2.4.1). Next, a list of observations is presented, followed by their interpretation and a number of possible explanations, illustrated by examples from the dataset[10]. Besides simple observations such as modeling time or amount of created model elements (see Section 2.4.2), more complex observations can be made from the study of patterns of operations (see Section 2.4.3). Furthermore, different charts can be compared to get additional insights (see Section 2.4.4).

The possible explanations were selected to optimally illustrate the usefulness of the PPMChart. It is not the purpose to be complete, but the focus is merely on demonstrating the usefulness of the visualization. A deeper understanding of the presented observations is discussed in Section 2.6. Note that the caption of each figure indicates which settings were used to produce the PPMChart in the plug-in.

2.4.1. Data collection

The observations below are based on the data of two observational modeling sessions conducted at Eindhoven University of Technology. The participants were international master students of three different educational programs (i.e., Operations Management & Logistics, Innovation Management, and Business Information Systems), which attended a course in Business Process Management. Participation was voluntarily and the students could decide to stop at any time without handing in a solution. They firstly completed a tool tutorial to get familiar with CEP and the modeling language. In this tutorial the user was presented with an explanation and a short movie on how to perform a certain operation in the tool. Only when the participant successfully imitated the example, the

---

[10] See also Appendix C, which explains how the observations were made.

tutorial continued. Next, based on a textual description they were asked to model the control flow of a certain business process. A survey was presented to collect additional information on the modelers (e.g., gender, age, familiarity with the case, etc.).

The first session was performed in November 2010 and 120 students participated. Each student constructed a process model in CEP for two cases (Pre-Flight and NFL case[11]). For the Pre-flight case, the modelers created models with 13 activities and used 120 recorded operations on average. The NFL models contained 9 activities and were constructed with 85 recorded operations on average. In December 2012, 117 students participated in the second modeling session. For this session, every student modeled only one case (Mortgage case[12]). Again, CEP was used to record the modeling operations. This session resulted in models of 27 activities and using 276 recorded operations on average. This indicates the mortgage case is more extensive than the other two cases.

2.4.2. Simple observations

A visualization of high quality supports the discovery of surprising insights in highly complex data, but should also support the easy derivation of simple characteristics. For the sake of brevity, this section is limited to a brief presentation of four of such rather simple observations.

Modeling speed

Observation: The width of the part of the chart that contains dots differs between charts from different modelers that modeled the same case.
Interpretation: Some modelers work faster than others.

Modeling pauses

Observation: Some charts have clear horizontal gaps: i.e., a non-trivial time interval where no line contains dots.
Interpretation: Some modelers pause their modeling operations at certain times.

---

[11] Both case descriptions can be downloaded from http://bpm.q-e.at/experiment/Pre-Flight.

[12] Case description can be downloaded from http://bpm.q-e.at/experiment/MortgageEindhoven.

Amount of model elements

Observation: The number of lines differs between charts from different modelers that modeled the same case.
Interpretation: Some modelers create more model elements.

Amount of modeling operations

Observation: The relative number of blue, red, and orange dots differs between charts from different modelers that modeled the same case.
Interpretation: Some modelers tend to move, delete, or rename elements relatively more than others.

2.4.3. Observations on patterns of operations on model elements

This section discusses a number of observations that relate to patterns of operations on model elements. These can be derived from the amount and position of dots of a certain type.

Patterns of delete operations

Observation: If the chart contains red dots, they are sometimes scattered around the chart (without a clear pattern), and sometimes a vertical line of red dots can be distinguished.
Interpretation: Some modelers delete model elements at various times, some modelers decide to delete a whole part of the model (i.e., multiple model elements) at the same time.

Possible explanations: In Figure 2.7a the red dots are scattered over the PPMChart. Possibly, this means the modeler occasionally changed her/his mind about the content of the model. When the PPMChart shows a vertical line of delete operations as can be observed in Figure 2.7b, the modeler threw away a whole part of the model at once. Possibly, the modeler wanted to start over and remodel that part or decided that that part was not necessary in the model (a closer inspection of the data about the operations after the deletion might reveal the exact cause).

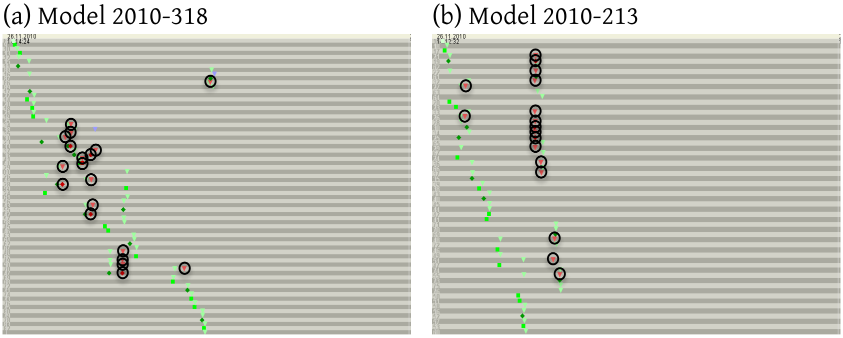


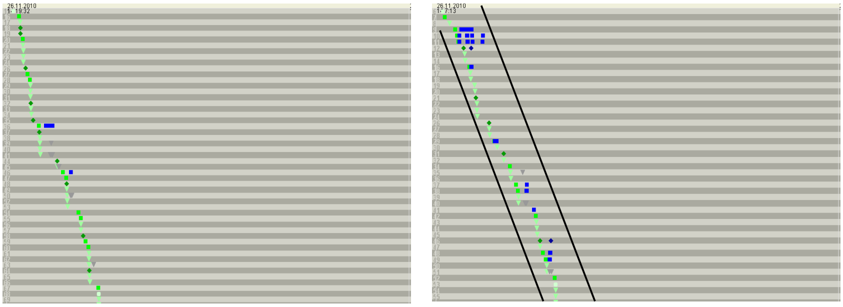
Figure 2.7. Scattered or simultaneous delete operations (charts hide moves and renames)

Patterns of move operations

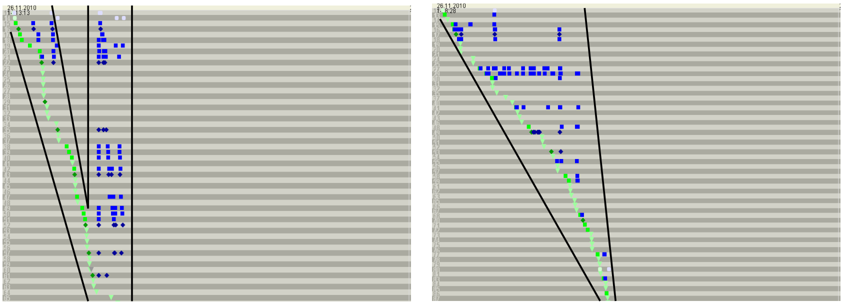
Observation: If the charts contains blue dots, sometimes they are positioned near to the green dot on the same line, sometimes they form a broad vertical line at the right, and sometimes the dots are scattered over the chart.

Interpretation: Some modelers hardly move any model elements, some modelers move the elements shortly after their creation, some modelers tend to move elements rather at the end of their modeling process, some modelers move model elements at various times.

Possible explanations: Some modelers do not move a lot of process model elements (see Figure 2.8a). In Figure 2.8b the movement pattern looks like a diagonal line. This means the modeler has moved elements only shortly after their creation and did never touch them again. Possibly, the modeler either has been pretty determined on the layout of the complete model, or the modeler did not bother to work on the layout of the model (the constructed process models might provide clarification). When the PPMChart shows a vertical line of blue dots at the right, the modeler has first created a number of model elements and then moved them simultaneously (see for example Figure 2.8c). Possibly, the elements needed to be moved to make room for a new part of the model, or to layout the model better. Finally, when the chart contains the triangle-like movement pattern as presented in Figure 2.8d, the modeler keeps moving elements that were created earlier in the modeling process. Possibly, this is caused by continuously layouting or creating space in the model, which might be derived from a closer review of the data.



(a) Model 2010-312 (b) Model 2010-210



(c) Model 2010-170 (d) Model 2010-228

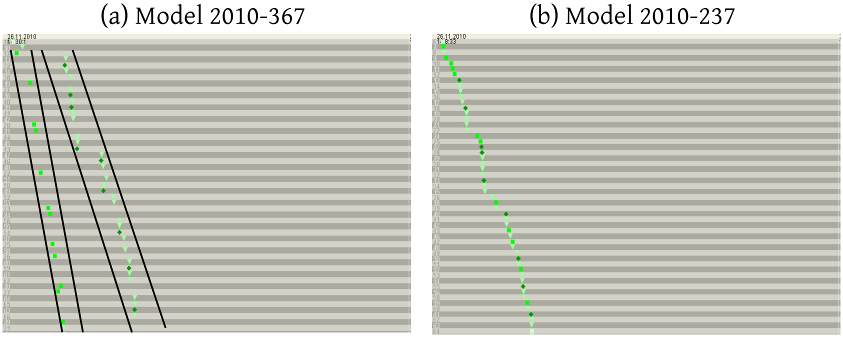
Figure 2.8. Timing of move operations: few (a), close to creation (b), at the end (c), scattered (d) (charts are sorted by First operation and moves and renames are hidden)

Note that the modeler may also combine these patterns; for instance, a modeler moves the elements shortly after their creation, does not touch them again until she/he at the end starts moving elements around again to work on the overall layout of the resulting model. This is for example displayed in Figure 2.8c where the PPMChart has, next to the vertical line of blue dots at the end, also a moving phase at the beginning of the modeling process.

Patterns of create operations: order of creating activities, gateways and edges

Observation: Some charts have non-crossing vertical lines formed by green, dark green and light green dots.
Interpretation: Some modelers delay creation of edges (and gateways) until all activities are put on the canvas.

Possible explanations: In some PPMCharts, activities are created first (green dots) followed by the edges (light green dots) (see Figure 2.9a), while other PPMCharts show a more divers order of creating activities, gateways and edges (see Figure 2.9b). Possibly, modelers either work aspect-oriented (i.e., they first focus on the content aspect by creating all activities in the model before connecting them with gateways and edges to fix the structure aspect), or flow-oriented (i.e., they first finish a logical part of the model by creating nodes and edges and then turn to another part of the model).


(a) Model 2010-367

(b) Model 2010-237

Figure 2.9. Order of creation of activities, gateways and edges (charts hide moves, deletes and renames)

Patterns of create operations: split and join gateways

Observation*:* When sorted by first event, it can be noticed that in some charts dark green dots come (mostly) in pairs, while in other charts they are interchanged with bright green dots.
Interpretation: Some modelers create the join gateway right after the creation of the split gateway, while others create the join gateway later on.

Possible explanations: Figure 2.10a presents a model where the creation of one gateway is mostly directly followed by the creation of another gateway. This modeler puts the split and join gateway right after each other on the modeling canvas, possibly to not forget to add the join gateway. He concentrates first on the correct structure of the model and then on the content (i.e., aspect-oriented). The modeler in Figure 2.10b follows a more flow-oriented approach. The study of the chart leaves the impression that the modeler constructed blocks of the model more linearly from start to end event. The join gateways are only created when all intermediate activities of the block structure are already in place. Note, that for both modelers the overall modeling process is flow-oriented (constructing part after part), but only within the creation of model blocks a different approach is observed.

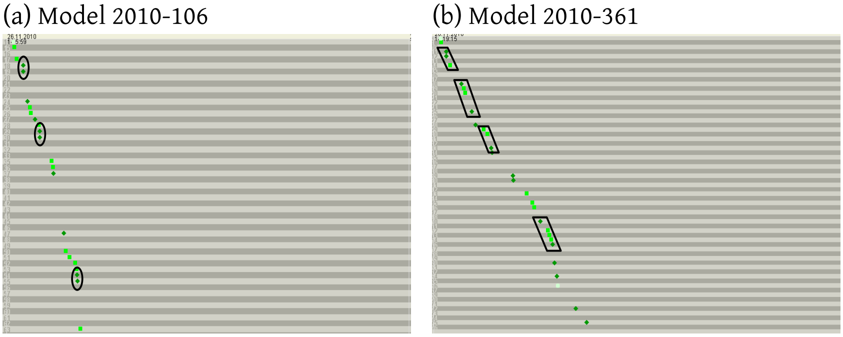


Figure 2.10. Order of creation of gateways and activities
(charts are sorted by First operation and hide edges, moves, deletes and renames)

Patterns of create operations: chunked modeling

Observation: Some charts contain groups of different shades of green delimited by pauses (see Section 2.4.2).
Interpretation: Some modelers work in chunks: i.e., they alternate between creating a group of activities, gateways and edges, and pausing.

Possible explanations: It is observed that some modelers group the creation of parts of the model. While the second observation in Section 2.4.2 concludes that modelers might take a pause at various times, we observe that some modelers seem to pause only after finishing a specific part of the process model consisting of gateways, activities and edges. Possibly, these parts correspond with process model blocks (i.e., part of the model consisting of a split and matching join gateway and all intermediate nodes). Because the parts delimited by pauses seem to represent deliberate parts of the process model, we call it chunked modeling. In Figure 2.11a the modeler seems to have constructed the process model in one chunk, while in Figure 2.11b the modeler worked in smaller chunks.

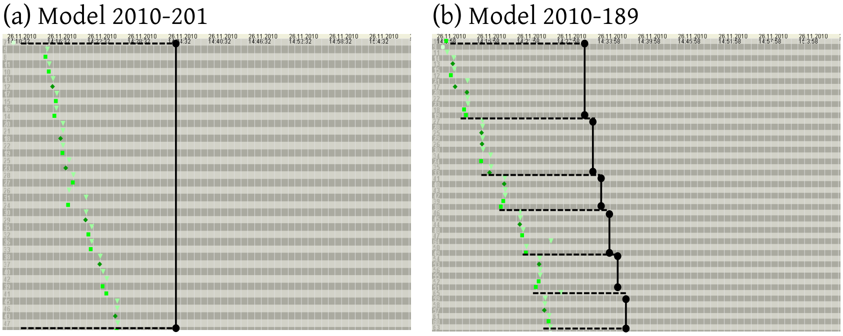
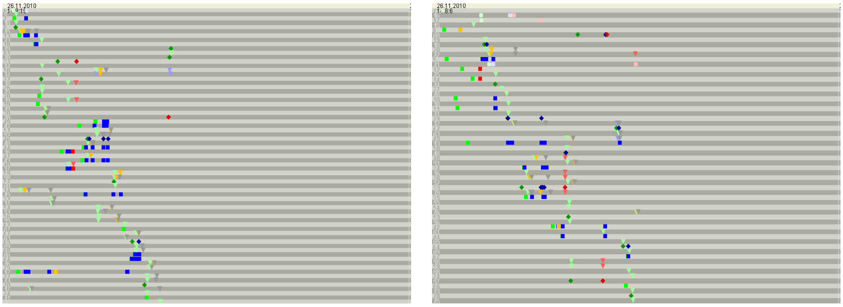


Figure 2.11. Chunked modeling
(Time intervals are set to minutes, charts hide moves, deletes and renames)

No clear patterns

Observation: In some charts no clear patterns are observed.
Interpretation: Some modelers have a more chaotic and less structured modeling process than other modelers.

Possible explanations: To be complete, it is important to notice that not every modeler has a clear, observable modeling approach. Some modelers work rather chaotically and tend to work on different parts of the model simultaneously. Possibly, they are not determined about how to construct the model. Indeed, it can be observed in the dataset that they usually have a lot of move and delete operations. Alternatively, some modelers might be more chaotic in nature regardless there level of experience. This might be checked if demographic data about the *need for structure* (Thompson et al., 1989) are collected. Figure 2.12 shows examples.


(a) Model 2010-258

(b) Model 2010-270

Figure 2.12. Chaotic process of process modeling (charts are generated with default settings)

2.4.4. Observations based on multiple diagrams

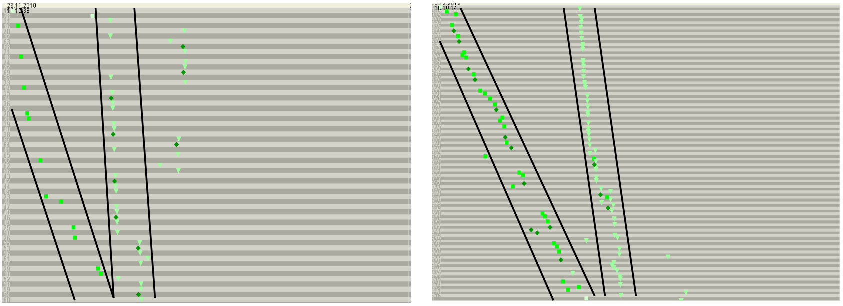
In this section, it is demonstrated how a deeper insight can be obtained through the comparison of a PPMChart with other PPMCharts of modeling sessions for more extensive cases, with other PPMCharts from the same modeler or with the constructed process model that corresponds to the PPMChart.

Comparison between PPMCharts of different cases

Observation: The same patterns (as described above) occur in charts for different cases.

Interpretation: The same modeling approaches can be observed for different cases.

(a) Model 2010-354 (b) Model 2012-156



(c) Model 2010-140 (d) Model 2012-136

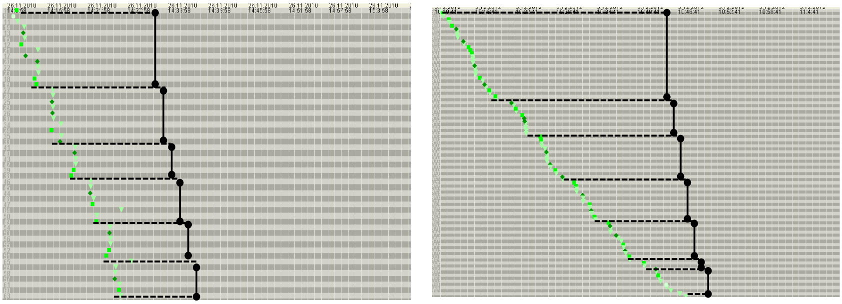


Figure 2.13. Similar patterns of creation of elements in simple (a, c) and extensive cases (b, d) (charts hide moves, deletes and renames; Time intervals of charts c-d is set to minutes)

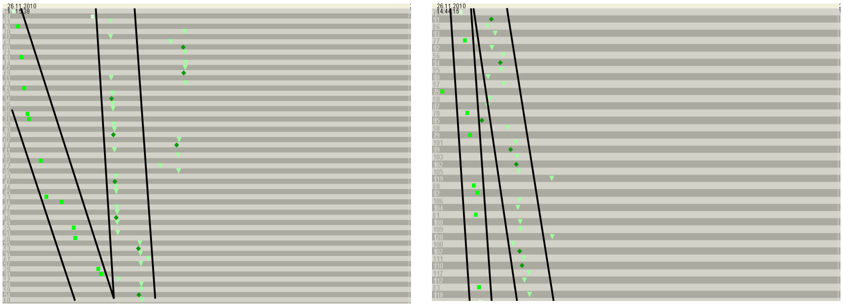
Possible explanations: Possibly, the discovered patterns correspond with general modeling approaches that are (rather) independent of the case to be modeled. Figure 2.13 shows two examples. The left column contains examples from the pre-flight case. The right column displays examples from the more extensive mortgage case. It can be observed that similar patterns exist in both datasets. An inspection of the corresponding process models can reveal insights on potential relations between certain patterns and process model quality (see further).

Comparison between PPMCharts of the same modeler modeling different cases

Observation: Sometimes, when the same modeler creates a process model for different cases, each chart for that modeler contains similar patterns. **Interpretation**: Some modelers used a certain modeling approach consistently.

Possible explanations: Possibly, these modelers have adopted a certain individual modeling approach that can be recognized in the PPM instances for different cases. Each PPMChart at the right in Figure 2.14 belongs to the same modeler as the corresponding chart at the left. Similar patterns can be observed.

(a) Model 2010-354 (b) Model 2010-355



(c) Model 2010-140 (d) Model 2010-141

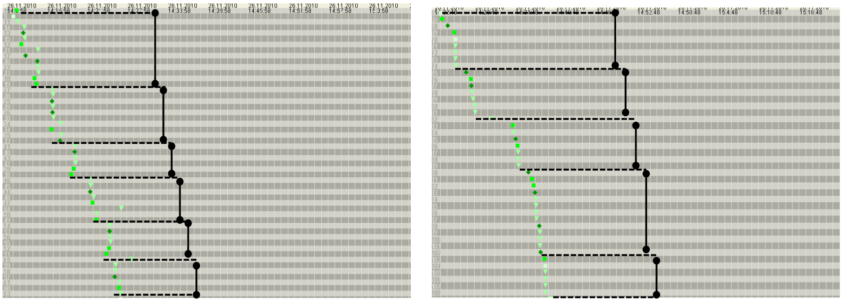
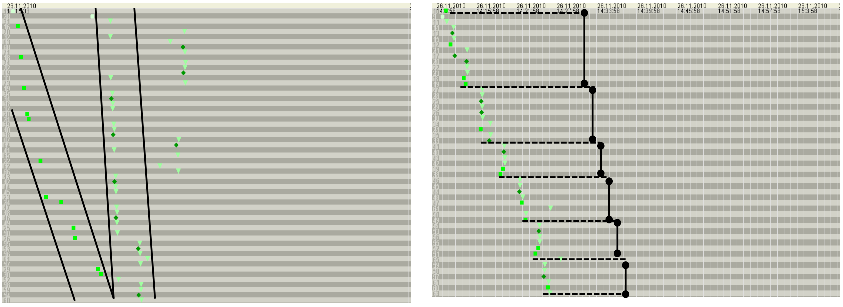


Figure 2.14. Similar patterns of element creation in a first (a, c) and second (b, d) modeling effort of the same modeler
(charts hide moves, deletes and renames; Time intervals of charts c-d is set to minutes)

Comparison between PPMCharts and process models

Observation: Some specific patterns seem to occur only in combination with particular process models.

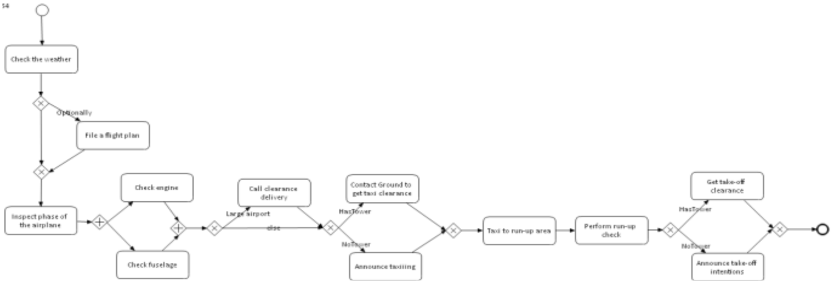
Interpretation: Modeling approaches can be discovered with the aid of PPMCharts that seem to have a relation with the properties of the modeling result (i.e., the constructed process model).



(a) Model 2010-354

(b) Model 2010-140

(c) Model 20120-354



(d) Model 2010-140

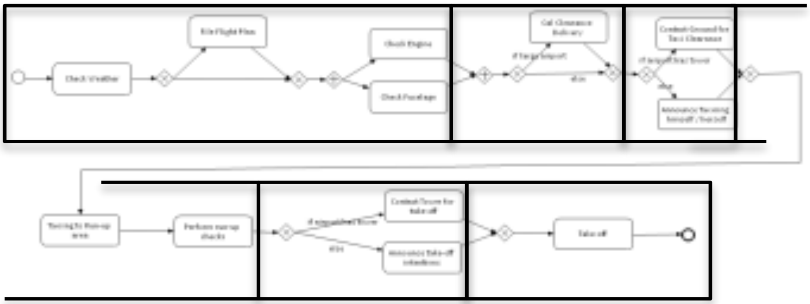


Figure 2.15. Patterns of creation of elements (a, b) and corresponding process models (c, d) (charts hide moves, deletes and renames; Time intervals of chart b is set to minutes)

Possible explanations: Figure 2.15 illustrates the comparison between PPMCharts and process models. The models at the bottom correspond to the charts at the top. The modeling approach in Figure 2.15a&c is aspect-oriented (i.e., first creating events and activities, only afterwards gateways and edges). The PPM instance depicted in Figure 2.15b&d shows chunked modeling. These are two different examples of a modeler that uses a clear modeling strategy that the modeler consistently applied. Note, how the corresponding process models also have a clear structure[13]. Because this particular relation can be observed in a high number of examples, this might indicate that having a clear modeling style (i.e., the patterns as described above are clearly recognizable), has an impact on the structuredness of the resulting process model.

2.5. Qualitative evaluation

This section reports on a qualitative study that confirms the usefulness of the PPMChart and the increase in cognitive effectiveness compared to the Dotted Chart. The evaluation was designed to collect mainly qualitative data about the use and usefulness of the visualization. Participants to the evaluation (see Section 2.5.1) were instructed to use a particular visualization (see Section 2.5.2) to graphically represent data about the process of process modeling (PPM) (see Section 2.5.3) and study the data for one hour (see Section 2.5.4). During the exercise qualitative and quantitative data were collected and before and afterwards a number of questions were posed to collect additional data (see Section 2.5.5). The results of the empirical evaluation are presented in Section 2.5.6.

2.5.1. Participants

The PPMChart visualization is in the first place intended to support explorative research into the PPM. The pool of potential participants therefore consisted of researchers that were familiar with process modeling, but that were not involved yet in research about the PPM. Six academic researchers with varying levels of experience volunteered to participate (three PhD students and one scientific programmer from Ghent University; and one PhD student and one assistant professor from Eindhoven University of Technology).

2.5.2. Visualization tools

Two visualizations were compared. The Dotted Chart in ProM 6.2 (see Section 2.2.1) was compared to the PPMChart (see Section 2.3.2). Each participant had to work with only one visualization (i.e., either Dotted

---

[13] We are aware of the fact the models are unreadable. This does not prevent to judge the structure of the models. The process models can be downloaded in high resolution from http://www.janclaes.info/papers/PPMISeB.

Chart or PPMChart). To prevent interference, one user that was familiar with the Dotted Chart was instructed to use the PPMChart. A random visualization was appointed to the other users, but in such a way that each visualization in the end would have been tested by an equal amount of users.

2.5.3. Input data

The evaluation involved five datasets that originate from a single observational modeling session. Every participant got access to all five datasets. These contained the information of the construction of models 2010-140 (see Figure 2.13c), 2010-213 (see Figure 2.7b), 2010-270 (see Figure 2.12b), 2010-354 (see Figure 2.13a), and 2010-361 (see Figure 2.10b). They were selected as outlier examples, which was explicitly mentioned to the participants. Each dataset consisted of a pre-loaded event log in ProM according to the format described in Section 2.3.4 and the corresponding process model on paper. It was explained to the participants that the data originated from a single modeling session where five different students had to model the same case using the same case description. No further information was provided about the modelers, the case, the quality of the process model, etc.

2.5.4. Protocol

The PPMChart visualization was developed to support explorative research. Therefore, no particular task was prepared, but the participants were simply instructed to use the visualization and the five event logs for one hour and to try to discover as much information as possible about how people construct models. Also, because of the explorative nature of the intended use, we selected researchers that had no experience with the visualization and we provided only minimal tool training.

The participants were asked to take place behind a computer on which ProM 6 and screen recorder software were pre-installed and running. The screen recording was activated as soon as they agreed with the capturing of data about their usage of the computer. Voice recording was activated accordingly after reception of their permission. First, it was briefly explained that the data was collected from observational modeling sessions with students and without treatment. The level of detail of the recorded data was described using some examples (e.g., creation of an activity, creation of an edge, movement of the activity, etc.). Next, ProM was introduced briefly. The five event logs were already imported in ProM. Using one random event log from the series of five, the assigned visualization was introduced. It was shown with the example event log how to generate the chart for a selected event log in ProM and which configuration options existed in the visualization plug-in.

Before starting the exercise, users were asked to think about the focus of their analysis. Using only the information of the brief introduction of the data and tool, they were instructed to elaborate on what kind of information they expected to be able to extract from the visualization. The introduction, instruction and focus question took no more than half an hour for each participant.

During the assignment, participants got one hour to explore the tool and the data and to reveal as much useful information about "how do people model" as possible. The initial exploration of the tool by the participant was included in the one hour duration, because the influence of the intuitiveness of the tool on the results was desired to be reflected in the results. However, to simulate real use conditions participants were allowed to ask the administrator for help if they did not understand a feature or did not know how to set a particular configuration. While they were working, participants were asked to think out loud and to clearly describe relevant observations. If they described an observation uncarefully, the administrator asked for a more precise description. Approximately every 15 minutes, when the administrator felt it would interrupt the participants the least, the session was paused for some minutes to ask two questions: (i) what is the most relevant observation so far, and (ii) from all the possible information that is present in the data and could be discovered with the visualization, how much percent do you think you already discovered. These questions served as a short mental break and were prearranged to keep the participant focused on the goal to derive as much insight as possible.

After exactly one hour the session was closed and the participants were asked to comment on the visualization. They were explicitly asked to be critical, to think of what bothered them, and to suggest improvements. When the participants could not think of any more feedback, the evaluation session was concluded with a short debriefing. They were explained that they used the developed or existing visualization and some information was given about our own initial insights into the PPM as presented in Section 2.4.

2.5.5. Measurements

Qualitative and quantitative data were collected during the evaluation session. Qualitative data includes the reported observations of the participants, observations of the administrator in how the participant uses the visualization, opinions of the participant about the data, visualization and tool (expressed during or after the assignment). Quantitative data includes per observation time and domain value (score from 1 to 5) assessed by an external BPM(N) expert. It was also coded by the authors whether an observation was correct, was directed or unexpected (based on the focus question answers prior to the assignment), was either focused on depth or breadth.

2.5.6. Results

To evaluate whether the PPMChart visualization is useful and cognitively more effective than the Dotted Chart, qualitative data was collected through participant observation and interviews during the evaluation session (Myers, 1997).

Usefulness

Section 2.4 clearly demonstrates the usefulness of the visualization for our own previous research. The question that remains is whether other users could draw meaningful conclusions from the charts as well.

It was observed that all three researchers that used the PPMChart visualization in the assignment indeed made similar observations than the ones described in this chapter. For example, they reported on patterns of deletion, movement and creation and discovered the styles that we labeled flow-oriented and aspect-oriented modeling. One of the participants even described a third similar style, which we did not discover in that dataset yet: one modeler appears to first have modeled one path from start to end and later on added all the exceptional paths using XOR gateways.

During the exercise and from the interview with the participants, perceptions about the tool were captured. The three researchers that tested the PPMChart visualization were mainly positive about the tool ("*very complete visualization*", "*contains a lot of useful options*", "*handy overview*", "*intuitive colors and shapes*"). On the question to name suggestions, participants proposed to "*display the model number somewhere on the screen*", "*reformulate filter options from 'hide' element to 'show' element*". Two of the participants mentioned that "*it was difficult in the beginning to use the tool* *because of its extensive options*". On the other hand, they both felt confident using the tool after less than an hour (i.e., before the end of the exercise).

Summarized, with the aid of the PPMChart visualization all the patterns and conclusions that are presented in Section 2.4 were discovered by the participants within one hour (using the five purposefully selected event logs). Also, some additional insights were derived from the charts (e.g., happy path first modeling style). The users perceive (parts of) the tool as being "*complete*", "*useful*", "*handy*" and "*intuitive*".

Increased cognitive effectiveness

We acknowledge that the Dotted Chart can be used for the study of the PPM, but we claim the PPMChart supports such research in a more effective way from a cognitive viewpoint. To evaluate this, the observations and feedback of the participants using both visualizations are compared.

First, it should be noted that we did not evaluate the choice of representing only one PPM instance at the time. Both participants appointed to the Dotted Chart and appointed to the PPMChart were provided with the same event logs containing one PPM instance each. This non-typical use of the Dotted Chart does however not cause a substantial bias, because none of the participants were familiar with the visualization they used. Therefore, they did not know that Dotted Chart is typically used with an event log containing multiple process instances.

All three researchers working with the Dotted Chart gradually started to change colors in the chart. Colors were picked to focus on a specific operation (i.e., because of a lack of semantic transparency), to clearly distinguish between different operations (i.e., to increase perceptual discriminability), or to give similar operations a similar color using different shades of the same color (i.e., to improve visual expressiveness, perceptual discriminability and graphic economy). They used an individual fixed color scheme in the end and started their last analysis by first changing the colors to their specific color scheme. In contrast, in the use of the PPMChart no colors were changed at all, which indicates the fixed color scheme of the PPMChart is intuitive, distinguishable and expressive (enough).

It was also observed that every Dotted Chart that was generated had an initial zoom level that did not fit the needs of the user (i.e., they changed it immediately after changing the colors). In PPMCharts zoom operations were only detected in the course of analysis to focus on a difficult zone.

In a PPMChart the timelines are sorted by the logical execution order of the process model from start event towards end event (i.e., by Distance from start). This makes it easier to connect the information to the process model on paper (i.e., cognitive integration). Only rarely the sort order of the PPMChart was changed by the users. In contrast, we recorded substantially more changes in the sort order for the Dotted Charts (i.e., to increase cognitive fit). This suggests the default sort order from the PPMChart can be considered to add value for the user.

Finally, it was noted that participants to the evaluation (i.e., using the Dotted Chart) occasionally changed the color of one particular type of event in a strikingly different color. We deduce that those participants wanted to focus only on this type of event, while disregarding the other events (i.e., to increase cognitive fit and for complexity management). In the PPMChart we did not observe this behavior, but two of the three participants used the filters in the plug-in for similar analysis tasks.

When we asked the participants about the Dotted Chart, the main suggestions pointed to "resolve weird zooming", "colors are different from previously generated chart", "hard to link back to the process model", "metric panel is useless", "drop the useless options", "use tones of colors", "zoom to fill the initial screen", "I can't tell which events are on the same model element". Other suggestions were to "work with predefined sets of configuration options", "use multi gesture zooming".

No substantial difference in the amount or quality of derived insights between the two visualizations was observed. But in the Dotted Chart users put a lot of effort in first configuring the tool to its maximal cognitive effectiveness to support their analysis, which we did not observe in the PPMChart. Participants reported on several cognitive drawbacks of Dotted Chart concerning for example cognitive integration ("*different colors*", "*hard to link*") and semantic transparency ("*which events are on the same model element*").

2.6. Limitations, implications and future research

2.6.1. Limitations of this study

The usefulness of the visualization is illustrated in Section 2.4. The presented examples mostly represent extremes. An informal analysis indicated that, although these extremes were present in the dataset, for a number of observations there appears to be a continuum of examples in between the presented extremes. Furthermore, these examples are based on data of modeling sessions with students only. They illustrate the usefulness of the visualization, but must not be considered to be representative for all modelers. Therefore, a profound study is needed to examine the circumstances and the generalizability of the observations and the possible explanations, which is out of the scope for this chapter, but it is partially addressed by the next chapters. Nonetheless, these examples provide a useful understanding of how the visualization can be applied for discovering properties of the process of process modeling (PPM).

The evaluation of the improved cognitive effectiveness of the PPMChart against Dotted Chart includes the study of qualitative data collected in an empirical evaluation study. Case study research could be performed to examine more in-depth and in a more realistic setting how the tool supports explorative PPM research in a cognitive effective way. Nevertheless, a summary of exemplar cases that show how the PPMChart could help or has helped researchers study the PPM is provided in Section 2.6.2 to exemplify the implications for research. In order to evaluate the difference in speed, amount and quality of derived insights between both visualizations, a quantitative approach would be desired. However, for a reliable quantitative evaluation a higher number of participants are required, which is cumbersome given the limited number of people in the intended target group. Moreover, the level of understanding of the reported insights from the point of view of the participant is largely lost when quantified (Kaplan & Maxwell, 2005), which makes a quantitative study less suitable to evaluate a tool for explorative research.

2.6.2. Implications for research and practice

The comparison of different diagrams and the discussion section clearly show that the PPMChart visualization can be a very helpful instrument in the exploration of data from observational modeling sessions. Concrete examples in Section 2.4.4 illustrate that the discovered patterns might relate to modeling approaches that are general (independent of the case), individual (dependent of the modeler) and that have an impact on the properties of the resulting process model. If this interpretation is correct, this means that the PPMChart can facilitate the study of the PPM significantly. In particular, this would be very useful for the research into modeling approaches that have a positive impact on process model quality. This way, it facilitates also the improvement of training or tool support to increase model quality (see Section 2.6.3). Due to the growing importance of process models in process analyses and optimization efforts, this is an important contribution to the research and practice of the business process management field.

Discovery of modeling phases and styles

When modelers work aspect-oriented, the PPM proceeds in phases. The PPMCharts for our dataset revealed that some PPM instances contain, besides the *creation phases*, also *deletion phases*, or *move phases*. Sometimes, the phases are interspersed with pauses, which can be easily detected graphically in the PPMCharts (see Section 2.4.2). Furthermore, the study of PPMCharts proposes that the *modeling phases* in one PPM instance might be aspect-oriented (i.e., during a modeling phase the modeler concentrates subsequently on different aspects: e.g., first content, then structure); or flow-oriented (i.e., the different modeling phases correspond with the creation of model chunks that are finished one after each other).

The existence of phases in the PPM was discovered before with the aid of Modeling Phase Diagrams (see Section 2.7.1) representing data from a different case than used for the datasets described in this chapter (Pinggera, Zugal, et al., 2012). Three phases are distinguished: a modeling phase in which the modeler mainly creates new elements, a comprehension phase in which the modeler pauses his modeling activities, and a reconciliation phase in which the modeler moves and deletes elements. Pinggera, et al. report on a study that analyses the modeler's eye movement in the model canvas and case description areas on screen (Pinggera, Furtner, et al., 2013). It was concluded that some modelers use the pause to reflect on their model so far and some modelers use the pause for reading the case description.

Next, similar patterns were observed between the PPM instances of the same modeler for different cases and between instances of different modelers and different cases. This suggests that common modeling styles exist, and that a modeler might adopt a rather fixed modeling style over time.

Pinggera, et al. used clustering techniques on the dataset from the modeling session in Eindhoven in 2010 and discovered three modeling styles: "*(i) modeling with high efficiency, (ii) modeling emphasizing a good layout of the model, being created less efficiently, and (iii) modeling that is neither very efficient nor very focused on layouting*" (Pinggera, Soffer, et al., 2013). The three styles can be characterized by the modeling speed (i.e., fast versus slow) and the amount of reconciliation operations (i.e., many versus few moves, deletes, renames).

This clearly demonstrates the usefulness of the PPMChart visualization. It can be used to make the same observations that have lead to the interesting insights presented in this section (i.e., the existence of phases and styles). Furthermore, concerning patterns of individual operations within phases, it provides more detailed information than for example Modeling Phase Diagrams.

Link with process model quality

While the previous section discusses how the PPMChart could have been used in other research about the PPM, this section describes how the PPMChart was actually used in PPM research. Because of the recent development of the PPMChart, only a limited part of our findings is already published in academic articles. The comparison of PPMCharts with the corresponding process models (see Section 2.4.4) revealed a potential link between certain PPM properties and process model quality. For example, we observed that modelers with a short PPM (see Section 2.4.2) tend to produce process models of better quality. It was also shown that a lot of move operations (see Section 2.4.2) often goes hand in hand with models of poorer quality. Finally, we discovered that modelers with a chunked modeling style (see Section 2.4.3) typically create better process models. These three conjectures were further studied and a statistical analysis of the data confirmed that these relations exist in the dataset (see Chapter 3).

2.6.3. Future research

To be able to reach the point of gaining knowledge of the connection between the characteristics of the PPM and the quality of the resulting process model, a substantial amount of research still needs to be performed. Concerning the graphical representation of the collected data in the PPMCharts presented in this chapter, the future work concentrates on improving the support for the inclusion of more data into the analysis (i.e., improve the cognitive integration of information). For example, from our experience in using the charts for research, we learned it would be valuable to simplify the way to link individual dots in charts with elements in the corresponding process models. Tools may also help in comparing different charts in a visual way. Research is needed to examine how this can be optimally implemented.

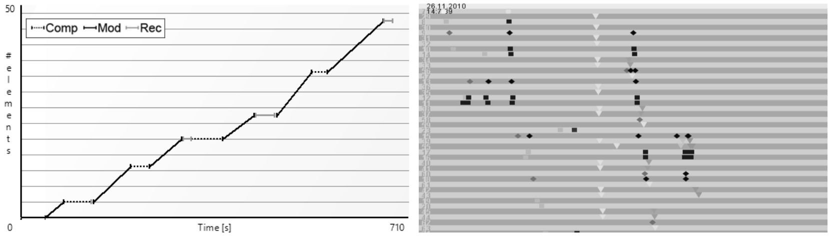
Furthermore, it might be interesting to include information about the *modeler* (e.g., is there a difference between the PPM for novice and expert modelers), *model language* (e.g., do certain modeling notations influence the way in which a modeler constructs the model), *modeling tool* (e.g., how do existing tool features help the modeler), *modeling case* (e.g., how much does the complexity of the case to be modeled influences the PPM), etc. in the charts in such a way that the additional information does not lower the cognitive advantages of the visualization. Future work may also include the extension of the configuration options, such as representing other operations than mentioned in Table 2.1, supplementary filter options, sort options, etc.

2.7. Related work

This section describes related work from several perspectives. First of all, related research in the area of visualizing the process of process modeling (PPM) is summarized (see Section 2.7.1). Next, the usage of traditional process modeling notations for the visualization of the PPM is discussed (see Section 2.7.2). Subsequently, a selection of other visualizations that focus on hierarchy and control perspectives is presented (see Section 2.7.3).

2.7.1. Visualizations of the process of process modeling

Pinggera, et al. (2012) use Modeling Phase Diagram visualizations to graphically represent the consecutive phases of a modeler in the construction of a process model. A Modeling Phase Diagram is a line chart representing the number of elements present in the partial process model during the model construction. The alternating line color (i.e., black or grey) and line style (i.e., solid or dotted line) indicate the consecutive phases of the modeling process (see Figure 2.16a). While this visualization is very useful to provide a consolidated graphical overview of these modeling phases, it does not allow zooming in on the singular recorded events of the PPM instance it represents. It is tailored to the study of consecutive phases in the PPM. Alternatively, the PPMChart visualization (see Figure 2.16b) was constructed to represent the captured data at a detailed level, which facilitates the analysis of singular operations performed by the modeler, rather than analyzing the characteristics of modeling phases. The result is a more comprehensive visualization than the Modeling Phase Diagrams. Moreover, the PPMChart displays the data as it was recorded and leaves the analysis and recognition of patterns, and the interpretation about the cause and meaning of specific operations to the reader of the chart. While the Modeling Phase Diagram is developed to support a very specific analysis at an aggregated level (i.e., of the model phases in the PPM), the PPMChart is designed to support the analysis of various aspects of the PPM at a detailed level.



(a) Modeling Phase Diagram

(b) PPMChart

Figure 2.16. Modeling Phase Diagram and PPMChart for the same PPM instance

2.7.2. Process models

A conceptual (process) model is a "*formal description of some aspects of the physical and social world around us for purposes of understanding and communication*" (Mylopoulos, 1992, p. 3). A combination of textual and graphical notations can be used for this formal specification (Engelen & Van den Brand, 2010). When the focus is on control flow (i.e., the order of activities and events), graphical representations are preferred over textual models (Weske, 2007). Various graphical process model notations exist (e.g., BPMN (OMG, 2011a), UML Activity diagrams (OMG, 2011b), Petri nets (Reisig & Rozenberg, 1998), Workflow nets (Van der Aalst, 1998), YAWL (Van der Aalst & Ter Hofstede, 2005), and Event-Driven Process Chains (Scheer, 1998)). To capture the complexity of processes, process models make use of the principles of structuring and abstraction (Polyvyanyy, 2012). For example, many process models do not show information at instance level, but summarize information of different executions, while often hiding infrequent behavior (Polyvyanyy et al., 2010; Reichert, 2013).

In terms of process model visualization, it should be noted that recent research studies how changes in the process model notation can help to lower the mental effort for the reader to interpret the models. For example, in order to bridge the gap between the traditional formal process model notations used by process modelers and the informal representations often used in practice (Barros & Ter Hofstede, 1998; Phalp, 1998), icons are introduced in existing model notations (Mendling, Recker, et al., 2010). Other research focuses on the addition of a third dimension to the model (Effinger, 2013) or represents the process model in a virtual world (Brown, 2010; Guo et al., 2013). Also, more creative approaches are used, such as sonification (Hildebrandt et al., 2012).

Although the different abstraction mechanisms and visual optimizations of the notations have success in supporting readers of the models to deal with the complexity of the represented process, they have in common that they hide details of individual instances. Therefore, classical process model notations are not suitable for the in-depth analysis at instance level of processes in general and the PPM in particular. In contrast, the PPMChart visualization represents only one PPM instance of which no details are hidden. This only shifts the complexity problem, because one still needs to compare a high number of charts for obtaining an overview over different instances, but this better fits with the goal of the exploratory research of this study, which aims at revealing in-depth information. Therefore, traditional process model notations are not appropriate to support this research.

2.7.3. Visualizations that concentrate on control flow and hierarchy

An optimal visualization technique for exploratory research about the characteristics of the PPM should show a lot of details about the recorded operations on model elements, focusing on timing and relative order and taking the hierarchical structure of the data into account. Therefore, this subsection presents a number of visualizations that concentrate on control flow or hierarchy and discusses their potential usefulness as an instrument to graphically detect characteristics of the PPM from the collected observational data.

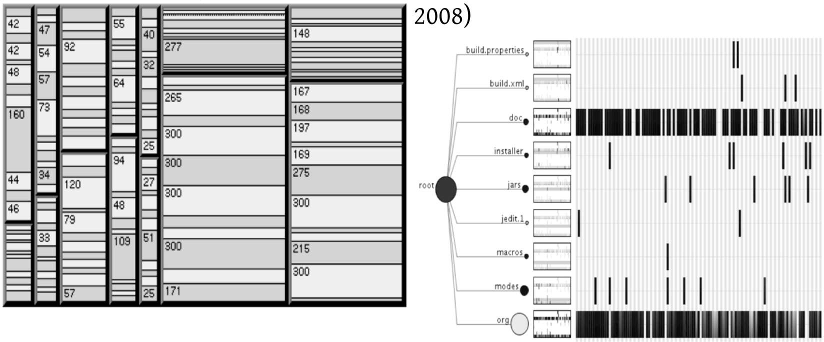
Different visualizations exist to represent *hierarchical* information (e.g., Treemaps (Johnson & Shneiderman, 1991), Timeline Trees (Burch et al., 2008), Arctrees (Neumann et al., 2005), Information Slices (Andrews & Heidegger, 1998), Sunburst diagrams (Stasko & Zhang, 2000)). These graphical representations display the details in such an hierarchical placement that relations between data elements can graphically be discovered.

Treemaps display information about elements as rectangular blocks of which position, width, height and color are the main properties to represent characteristics of the data (see Figure 2.17a). They make optimal use of the available space in the chart, because the whole chart is filled with information. However, the TreeMap visualization cannot optimally support the research into the PPM, because it focuses mostly on hierarchy and relative importance of the represented data elements. It has not been optimized to provide cognitive support for the recognition of patterns in the ordering of operations. Also Arctrees, Information Slices and Sunburst diagrams, which make use of a radial placement of information visualization elements, have the same shortcomings.

Timeline Trees do include an explicit representation of order and timing of data elements grouped in categories. The hierarchy of the data categories is represented by a textual tree, and the timing of transactions for each of the data elements in a category is visualized by a timeline for each of the leaves of the tree (see Figure 2.17b). The focus of the visualization is evenly spread over the hierarchy and the timing of data elements. For the PPM, this visualization can be used; although, in our opinion, the tree representation of the hierarchical structure of the data takes too much room and is distracting if the number of data categories increases.

Other well-known visualizations focus less on the representation of hierarchy, but more on the *timing* of or the *relation* between the data elements (e.g., Gantt charts (Gantt, 1913) and Railroad line diagrams (Tufte, 1983)). Gantt charts are used in project planning to analyze phases, dependencies and timing of projects (Wilson, 2003). The different phases of the project are mentioned beneath each other. Besides each phase, a horizontal bar indicates the planned timeframe for each phase on the time axis (see Figure 2.17c). The focus on the length of phases and dependencies between phases, and the lack of attention to optimally represented details of individual steps of each phase, makes them less suitable for the analysis of the PPM with regard to the discovery of useful insights at the level of patterns of individual operations of the modeler.

Next, different informal Railroad line diagram visualizations exist to represent the routes and hour schedules of trains. They can be considered variants of, for example, Marey's train schedule. The route of a train from one station to another is represented by a line that traverses the chart from left to right. Vertically the different stations are displayed and the horizontal axis represents a timeline (see Figure 2.17d). This visualization is suitable for the display of information about an object of which the properties change in two dimensions (i.e., place and time), but does not allow for adding more dimensions of information easily without decreasing its cognitive effectiveness substantially, which would have been necessary to be used for the analysis of the PPM.



(a) TreeMap (from Johnson, 1993)

(b) Timeline tree (from Burch et al., 2008)

(c) Gantt chart (from SE blog[14])

(d) E.J. Marey's train schedule (from Tufte, 1983)

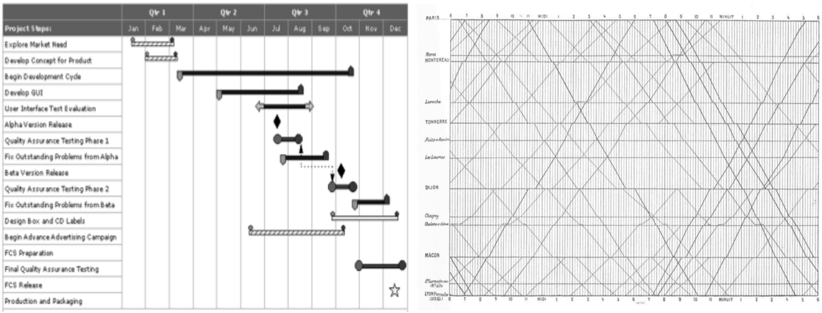


Figure 2.17. Process visualizations that are not considered as process models

2.8. Conclusion

The goal of the research described in this chapter was to design and implement a visualization that helps to study observational data about the process of process modeling (PPM) in a cognitive effective way. The visualization makes the characteristics of PPM instances explicit, which facilitates the development of theory, training and tool support for various aspects of the PPM, and especially in the context of increasing the quality of the resulting process models.

The PPMChart visualization of the PPM presented in this chapter displays modeling operations of one modeler in the construction of a single process model as colored and shaped dots in a chart. The dots are positioned on horizontal timelines that represent the model elements on which the operations are performed. The PPMChart is implemented in the process mining tool ProM in such a way that various options can be configured and that the data can be filtered from within the plug-in. This

---

[14] See http://software-document.blogspot.be/2010/07/activity-network-methods.html.

allows to effortlessly take different views at different levels of abstraction on the modeling operations. The chapter contains an extensive list of examples and observations to demonstrate the usefulness of this graphical representation in analyzing the PPM and the modeling behavior and styles of different modelers. A qualitative study confirms the usefulness of the PPMChart and the improved cognitive effectiveness compared to the Dotted Chart.

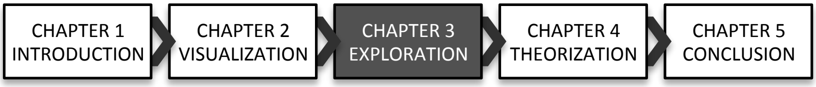
Two specific characteristics of the visualization need special attention. Firstly, the PPMCharts show raw, uninterpreted data. Each dot represents a clearly observable distinct modeling operation of the modeler. The interpretation of the meaning or cause of a specific operation is left up to the reader of the charts. Secondly, the visualization makes advantage of the same benefits that were originally present in the Dotted Chart Analysis plug-in. The presentation of operations makes use of dots that have a particular color, shade, shape and position, which means that all available data are presented in the visualization and that the reader can zoom in on details or at the same time take a so-called helicopter view on the whole chart. This is in contrast with classical process visualizations (i.e., process models) that mostly abstract from the data on individual cases and try to summarize the data. Both properties are beneficial for the explorative purpose the PPMCharts were developed for.

3

Tying Process Model Quality to the Modeling Process

The Impact of Structuring, Movement, and Speed

Published as J. Claes, I. Vanderfeesten, H.A. Reijers, J. Pinggera, M. Weidlich, S. Zugal, D. Fahland, B. Weber, J. Mendling, G. Poels, Tying Process Model Quality to the Modeling Process: The Impact of Structuring, Movement, and Speed, Proc. BPM '12, LNCS 7481, Springer, 2012, p. 33-48.



Summary. In an investigation into the process of process modeling, we examined how modeling behavior relates to the quality of the process model that emerges from that. Specifically, we considered whether (i) a modeler's structured modeling style, (ii) the frequency of moving existing objects over the modeling canvas, and (iii) the overall modeling speed is in any way connected to the ease with which the resulting process model can be understood. In this chapter, we describe the exploratory study to build these three conjectures, clarify the experimental set-up and infrastructure that was used to collect data, and explain the used metrics for the various concepts to test the conjectures empirically. We discuss various implications for research and practice from the conjectures, all of which were confirmed by the experiment.

Keywords. Business Process Modeling, Process Model Quality, Empirical Explorative Research, Process of Process Modeling.

3.1. Introduction

Business process modeling is utilized at an increasing scale in various companies. The fact that modeling initiatives in multinational companies have to rely on the support of dozens of modelers requires a thorough understanding of the factors that impact modeling quality (Krogstie et al., 2006; Mendling, 2008; Rittgen, 2010). One of the central challenges in this area is to provide modelers with efficient and effective training such that they are enabled to produce high-quality process models. There is clearly a need to offer operational guidance on how models of high quality are to be created (Becker et al., 2000; Mendling, Reijers, et al., 2010).

Recent research has investigated several factors and their influence on different measures of process model quality (Mendling et al., 2012; Reijers & Mendling, 2011). In essence, this stream of research identifies both process model complexity and the reader's modeling competence as the major factors among these. While these insights are in themselves valuable, they offer few insights into how we can help process modelers to create better models right from the start. In order to give specific hints to the modeler, we have to shed light on how good process models are typically created, and in which way this creation process differs from drawing process models of lower quality.

In this chapter, we look deeper into the modeling process in its relation to the creation of a high-quality process model. The research question we deal with, is whether it is possible to identify certain aspects of modeling style and model creation that relate to good modeling results. Our approach has been to leverage the Cheetah Experimental Platform (Pinggera, Zugal, & Weber, 2010), which allows for tracing the creation of process models on a detailed level. This permitted us to quantify the process of process modeling with respect to three different aspects. We also determined an objective measure for the quality of the resulting process models, putting the focus on the ease with which such models can be read. Based on an experiment with 103 graduate students following a process modeling course, we were able to demonstrate a strong statistical connection between three aspects of the modeling process on the one hand with our notion of model quality on the other. These findings have strong implications, as they pave the way for explicating and teaching successful modeling patterns.

Section 3.2 discusses cognitive concepts that are relevant for investigating the process of process modeling. In addition, we describe how the capabilities of the Cheetah Experimental Platform are conducive to document the process of process modeling in detail. Section 3.3 presents our research design. We explain how we developed three conjectures about process-related factors that result in better process models. Each of these three factors as well as the notion for process model quality is operationalized, such that the conjectures can be experimentally tested. Section 3.4 reports on the conduct and results of our experiment. We discuss the results and reflect upon the threats to their validity. The chapter closes with conclusions and an outlook on future research.

3.2. Background on the Process of Process Modeling

In this section, we revisit findings on process model quality and the process of process modeling. Section 3.2.1 summarizes prior research in this area, after which Section 3.2.2 discusses how the process of process modeling can be analyzed.

3.2.1. The Process of Process Modeling and Process Model Quality

There is a wide body of literature that centers on the quality of process models, ranging from high-level, comprehensive quality frameworks (e.g., Becker et al., 2000; Krogstie et al., 2006; Reijers et al., 2010) to a variety of metrics that pin down the quality notion in specific ways (e.g., Gruhn & Laue, 2006; Mendling, 2008; Vanderfeesten et al., 2007). Mostly, the process model is considered in these papers as a given, complete, and finished artifact. Recently, approaches are emerging that aim to connect the way that a process model has come into being with the properties of the ensuing model. In this context, various authors refer to the actual construction of a process model as *the process of process modeling* (Indulska et al., 2009; Pinggera, Zugal, & Weber, 2010; Soffer et al., 2012).

In general, modeling is often characterized as an iterative and highly flexible process (Crapo et al., 2000; Morris, 1967), dependent on the individual modeler and the modeling task at hand (Pinggera, Zugal, et al., 2012). A central element in the further understanding of the process of process modeling is the identification of the recurring activities or common phases that comprise this process. Inspired by views on problem solving, Soffer et al. (Soffer et al., 2012) distinguish between the phase in which a modeler forms a mental model of the domain and the phase in which the modeler maps the mental model to modeling constructs. The work presented in (Pinggera, Zugal, et al., 2012) is in line with this view by its explicit recognition of a *comprehension* phase and a *modeling* phase, yet extends it by the additional recognition of a *reconciliation* phase. During the latter phase, modelers may reorganize the process model at hand (e.g., rename activities) and utilize the process model's secondary notation (e.g., layout). While modeling and comprehension phases generally alternate, they may be interspersed with reconciliation actions (Pinggera, Zugal, et al., 2012). In the same work, a so-called *modeling phase diagram* is introduced that can be used to categorize a modeler's actions using these phases.

At this point, several preliminary insights exist that relate the modeling process with the modeling outcome, i.e., the business process model. First of all, the structure of the informal specification that is used as the basis for a process modeling effort seems to be of influence on the accuracy of the ensuing process model (Pinggera, Zugal, Weber, et al., 2010). The reason may be that pre-structuring such a specification lowers the mental effort for modelers, resulting in a process model that better reflects the actual domain. Another insight is that the specific reasoning tools that are at the disposal to the modeler, e.g., workflow patterns vs. behavioral patterns, seem to affect the mental model that the modeler creates of a domain and, in this way, influence the semantic quality of the process model (Soffer et al., 2012). Finally, in (Breuker et al., 2009) it is empirically shown that providing modelers in a distributed setting with specific model building blocks will minimize model quality issues such as variations in terminology and abstraction that individual modelers use.

The work that is presented in this chapter must be seen as an attempt to extend the list of factors that can be connected to the quality of a process model, in the spirit of (Breuker et al., 2009; Pinggera, Zugal, Weber, et al., 2010; Soffer et al., 2012). Another similarity with these works is that an empirical angle is taken to investigate conjectures about the influence of attributes of the modeling process.

3.2.2. Tracing the Process of Process Modeling with Cheetah Experimental Platform

The process of process modeling can be analyzed by recording editor operations as a sequence of modeling events. In this chapter, we rely on Cheetah Experimental Platform[15]. This platform has been specifically designed for investigating the process of process modeling in a systematic manner (Pinggera, Zugal, & Weber, 2010). In particular, the platform instruments a basic process modeling editor to record each user's

---

[15] For download and information we refer to http://www.cheetahplatform.org.

interactions together with the corresponding time stamp in an event log, describing the creation of the process model step by step.

When modeling with Cheetah Experimental Platform, the platform records the sequence of adding nodes, i.e., activities, gateways and events, and edges to the process model, naming or renaming activities, and adding conditions to edges. In addition, modelers can influence the process model's secondary notation, e.g., by laying out the process model using move operations for nodes or by utilizing bend points to influence the routing of edges (see Table 3.1 for an overview of all recorded operations). By capturing all of the described interactions with the modeling tool, we are able to replay a recorded modeling process at any point in time without interfering with the modeler or her problem solving efforts. This allows for observing how the process model unfolds on the modeling canvas. We refer to (Pinggera, Zugal, et al., 2012) for technical details.

Table 3.1 Recorded events in Cheetah Experimental Platform

Create	Move	Delete
CREATE_START_EVENT	MOVE_START_EVENT	DELETE_START_EVENT
CREATE_END_EVENT	MOVE_END_EVENT	DELETE_END_EVENT
CREATE_ACTIVITY	MOVE_ACTIVITY	DELETE_ACTIVITY
CREATE_XOR	MOVE_XOR	DELETE_XOR
CREATE_AND	MOVE_AND	DELETE_AND
CREATE_EDGE	MOVE_EDGE_LABEL	DELETE_EDGE
RECONNECT_EDGE (**)	CREATE_EDGE_BENDPOINT (*)	RECONNECT_EDGE (**)
	MOVE_EDGE_BENDPOINT (*)	
	DELETE_EDGE_BENDBPOINT (*)	
Other : NAME_ACTIVITY, RENAME_ACITIVTY, NAME_EDGE, RENAME_EDGE		

(*) create, move and delete edge bendpoint were considered as actions to move an edge
(**) reconnect edge was considered as deleting and creating an edge

3.3. Foundations of the Experimental Design

In this section we present the foundations of our experimental research design. Section 3.3.1 summarizes three conjectures that we derived from exploratory modeling sessions. Section 3.3.2 provides operational definitions for objectively measuring the process of process modeling. Section 3.3.3 builds an operational definition of quality for a resulting process model, which is suitable for our experimental setting.

3.3.1. Conjectures from Exploratory Modeling Sessions

To derive insights in the modeling process, we performed three small-scale experiments that involved 40 modelers in total. These were conducted at sites of the participating researchers throughout 2010. In these experiments modelers were asked to draw a process model[16] on the basis of a given informal description, which was the same at all sites. We analyzed the results of these experiments by visualizing the recorded data in PPMCharts (Claes et al., 2015).

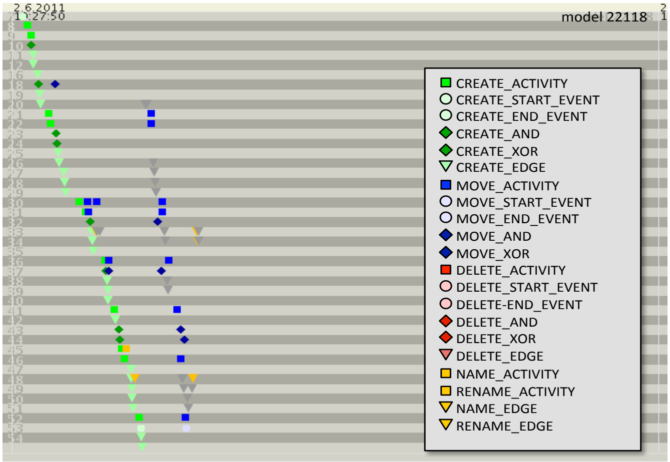


Figure 3.1. Visualization of the operations in the creation of a single model [17]

Figure 3.1 is an example of such a chart. The horizontal axis represents a time interval of one hour. Vertically, each line represents one object of the model as it was present during modeling. Each dot represents one action performed on the object; the color of the dot represents the type of action: create, move, delete or (re)name. The objects are vertically sorted by the time of the first action; the first action performed on each model object is its creation. In the example in Figure 3.1, we observe a short process (about 17 min) where most of the model objects were moved after creation (second dot on many lines). Furthermore, we see that the modeler has worked in 'blocks', i.e., two activities were created followed by gateways and edges. Figure 3.2 shows the clear and well-structured process model resulting from the creation process.

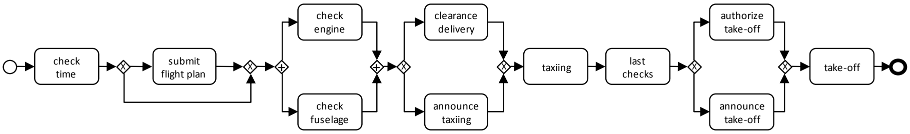


Figure 3.2. Process model as result of the modeling process in Figure 3.1

---

[16] The modeling language that was used in these experiments is presented in Appendix D.

[17] High resolution graphs are available from http://www.janclaes.info/papers/PPM.

The interesting point of our exploratory session was the variation that we could observe in the PPMCharts. Figure 3.3 shows different examples: Figure 3.3a shows a process where objects were barely touched after creation, while Figure 3.3b depicts a process with more actions, but mostly not long after the creation of the touched object. Figure 3.3c shows a process where move actions occurred after creation of *all* objects. Figure 3.3d visualizes a process with a rather chaotic actions pattern. Note that each PPMChart in Figure 3.3 visualizes the creation of a process model based on the *same* textual description. It can clearly be observed, therefore, that some modelers create more elements, take more time to create their model, or move around objects on the canvas more frequently than other modelers.

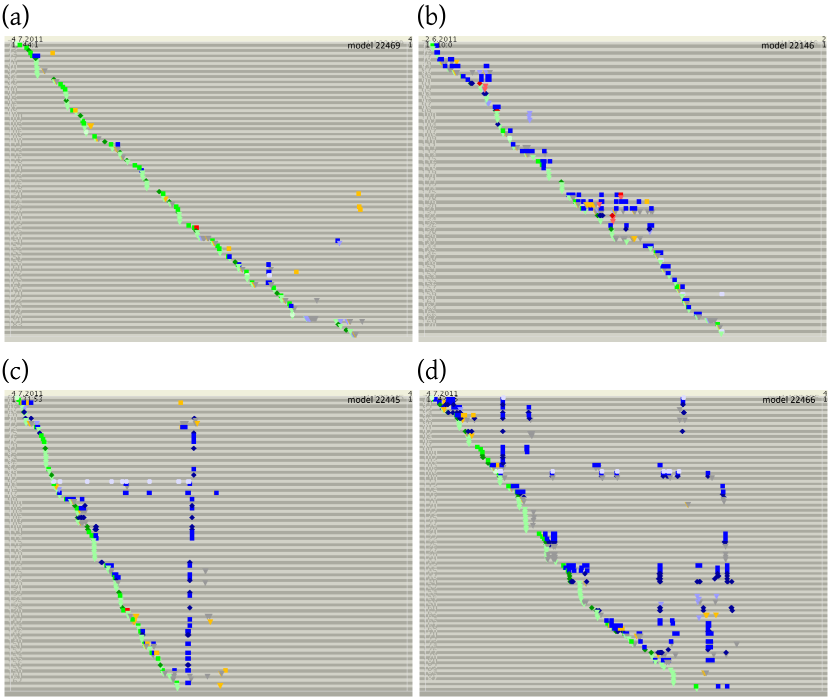

(a) (b) (c) (d)

Figure 3.3. More examples of PPMCharts

The utilization of PPMCharts helped us to identify patterns of modeling and connections between the process of process modeling and the quality of the resulting process models. More specifically, we found three conjectures:

Conjecture 1: Structured modeling is positively related with the understandability of the resulting model.

The conjecture is related to the limited amount of items that humans can hold in their *working memory* (Miller, 1956). Cognitive Load Theory suggests that problems arise when one's working memory is overloaded (Paas, Tuovinen, et al., 2003). We therefore surmise that working on the complete model at once will make overloading of the working memory more likely, as compared to working on calculable pieces of the model, one at a time. Conjecture 1 defines this style of working as *structured modeling*. In other words, we assume that focusing on a specific, bounded part of the model (e.g., a block as apparent in the modeling process in Figure 3.1) and finishing it before starting to work on another such part will help to reduce one's cognitive load. Hence, this style will result in better models.

Conjecture 2: *A high number of move operations is negatively related to the understandability of the resulting model.*

While studying the results of our exploratory experiments, we observed a notable difference in the structure of the modeling process across modelers. The data of the sessions suggest that modelers who frequently move model elements seem to have no clear idea in mind of how the process is supposed to be modeled. They will therefore potentially make more mistakes, which results models of lower quality.

Conjecture 3: *Slow modeling is negatively related to the understandability of the resulting model.*

Finally, we noticed a difference in the modeling speed of different modelers (i.e., in terms of the total time between the first and last recorded modeling actions). Presumably, modelers who are in doubt about the structure of the process or about the way to capture it, will spend more time thinking about the process, trying out different strategies to organize and re-organize the model. This will ultimately take more time to finalize the process model. We presume that the more time it takes the modeler to create the model, the lower the quality of the resulting model will be. Such an effect would be congruent with the result that faster programmers tend to deliver code with fewer defects than median or below-median performing programmers (Demarco & Lister, 1985).

3.3.2. Operational Measurement of Process-based Factors

The challenge arising for these conjectures relates to their operational definition. For **Conjecture 1**, we need to provide an operational definition for a structured style of modeling based on the notion of blocks.

- ***Definition 8: Process model block***
  In this context, a process model block consists of all involved model elements in two, or more, parallel or optional paths in the model.

Mostly, this will concern a structure that consists of one split gateway, some successive activities, and one join gateway to complete it. We consider the modeling process to be *structured* if the modeler is not working on more than one block at the same time. The degree of structured modeling is determined based on the replay of the modeling process as visually assessed by an expert. This assessment provides the values of two metrics for structured modeling.

- *MaxSimulBlock is the maximum number of blocks that were simultaneously in construction. A block was considered in construction from the time the first element was created until the time the last element was created. If a block was changed afterwards (e.g., deleting and creating an activity), it had no effect on this metric.*
- *PercNumBlockAsAWhole is the number of blocks that were made as a whole in relation to the total number of blocks. A block was considered to be made as a whole if no other elements (except for edges) were created between the creation of the first and last created element of the block.*

We observed many modelers positioning activities and gateways in a block structure while adding the edges much later. For this reason, we did not consider the edges to be part of the block when calculating these metrics. As we are interested in the timing of the *creation* of elements in a block, we did not consider changes after the original creation of a block. Therefore, only those elements that were present at the initial completion of a block (this is the point in time when its last element is added) were considered to be part of the block.

For **Conjecture 2**, we consider how many elements were moved and how many moves were performed on these elements. This was calculated by a program that determined which of the recorded actions are move actions according to the list presented in Table 3.1. We define the following two metrics.

- *AvgMoveOnMovedElements is the average amount of move operations on elements with at least one move operation.*
- *PercNumElementsWithMoves is the number of elements with move operations in relation to the total number of elements.*

For **Conjecture 3**, we also wrote a small program to calculate the time spent until the model was finished. As we observed many modelers moving lots of elements around after finishing the creation of all elements, we distinguish the time between first and last action and between first and last *create* action.

- *TotTime is the total time between the first and last recorded action of the modeling process.*
- *TotCreateTime is the total time between the first and last recorded create action of the modeling process.*

3.3.3. Operational Measurement of Process Model Quality

There is a wide body of literature available on quality measures for process models. In this chapter process model quality is defined as the ease with which the process model can be understood. In order to objectify this notion (and automate its assessment) we consider it from the structural correctness point of view (i.e., syntactic quality); not from the semantical point of view. Prior research has defined an extensive amount of formal, structural correctness criteria for process models (Van der Aalst et al., 2011). In the context of our experiments, we utilized BPMN as a modeling language. The problem with existing correctness criteria, such as soundness, is that they are not directly applicable to BPMN models because BPMN does not enforce a WF-net structure (Van der Aalst, 1998). Therefore, we consider a relaxed notion of quality, namely that the resulting process model should be *perspicuous*[18].

- ***Definition 9: Process model perspicuity***
  We operationalize the definition of a perspicuous model as "a model that is unambiguously interpretable and can be made sound with only small adaptations based on minimal assumptions on the modeler's intentions with the model".

To make our notion of *model quality* robust against the familiarity of a modeler with notational conventions, we translate each model to a syntactically correct BPMN model whenever the model structure strongly hints at the modeler's intentions. The resulting BPMN model is then transformed into a WF-net according to the mapping defined in (Dijkman et al., 2008). For such a WF-net, we checked soundness using LoLA (K. Wolf, 2007). A BPMN model is classified as being *perspicuous* if the respective WF-

---

[18] See Merriam-Webster at http://www.webster.com/dictionary/perspicuous.

net is sound; otherwise, it is classified as *non-perspicuous*. In the remainder of this section, we describe the transformation to derive a syntactically correct BPMN model that can be transformed into a WF-net based on structural characteristics. The transformation is inspired by the preprocessing discussed in (Dijkman et al., 2008) and applied in the presented order[19].

Handling of start and end events. Many modeling languages do not have specific symbols for the start or end of the process (e.g., Petri-nets and EPCs). Modelers who are not aware of these specific events in BPMN may, therefore, forget to include them in their model. In line with the BPMN specification, we normalize such models:

- *Transform a process that does not have a start or an end event into a process that does, by preceding each task without incoming flows by a start event and succeeding each task without outgoing flows by an end event. (Dijkman et al., 2008)*

Further, some modeling languages allow for several starting points in the model (e.g., EPC, BPMN), cf. (Decker & Mendling, 2009). Also, it is allowed or even required that each end point in the process model is indicated separately (e.g., EPCs, COSA, BPMN). Modelers may be familiar with this explicit modeling of each start or end point, so that a WF-net structure is obtained by the following transformations:

- *Transform a process that has multiple start (end) events by replacing all start (end) events with only one start (end) event succeeded (preceded) by an XOR-split (XOR-join) gateway, and connect this gateway to each activity that was preceded (followed) by one of the original start (end) events. (Dijkman et al., 2008)*
- *If we determine only one origin for the multiple flows, i.e., all starting (ending) paths join in (originate from) the same gateway, we use the sign (i.e., AND or XOR) of this gateway.*

Note that the latter rule, in particular, relates to the intention of a modeler and, therefore, is specific to the notion of a model being *perspicuous*. Figure 3.4 illustrates the transformations for exemplary cases.

---

[19] Note that these transformation rules may be generalized to any kind of modeling language.

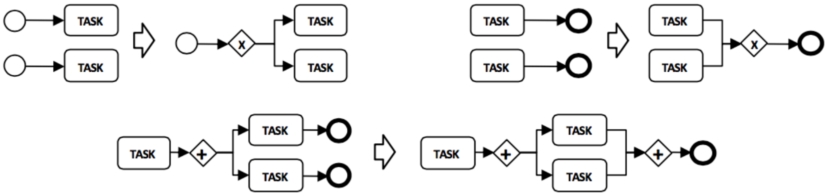


Figure 3.4. Transformations related to the handling of start and event events

Split and join semantics. BPMN allows for modeling nodes with more than one incoming or outgoing flow. To translate the BPMN model into a WF-net, we make those split and join semantics explicit:

- *Transform multiple incoming (outgoing) flows to an event or activity into one incoming (outgoing) flow, by preceding (following) the corresponding object with an XOR-join (AND-split) gateway that has all the incoming (outgoing) flows of the object. (Dijkman et al., 2008)*
- *If we determine only one origin (destination) for the multiple incoming (outgoing) flows, we use the sign of this gateway.*

Again, the latter transformation relates to the modeler's intentions. We deviate from the standard processing, if the model structure provides a strong hint to do so. Figure 3.5 illustrates the transformations. In the example in the lower half, none of the split gateways qualifies to induce the type of the join gateway, so that the default transformation applies.

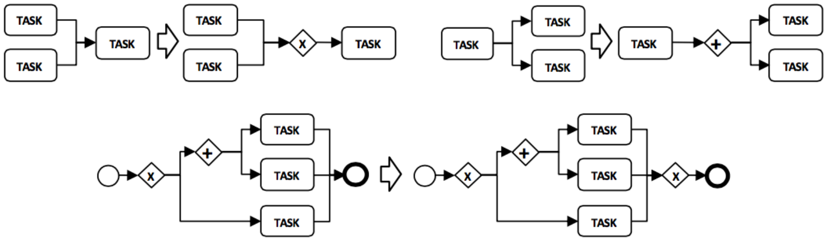


Figure 3.5. Transformations related to split and join semantics

Mixed gateways. BPMN allows for the specification of mixed gateways that combine split and join semantics. Those may be split up into a pair of a join and a split gateway of equal type (Dijkman et al., 2008). However, we do not adopt this transformation for several reasons. When building the conjectures based on preliminary studies, we observed that modelers would often be unsure about semantics of mixed gateways. In contrast to the handling of start and end events and split and join semantics mentioned earlier, however, the process model structure does not provide a strong hint on the modeler's intentions regarding a mixed gateway. As such, mixed gateways lead to a non-perspicuous model. Note that those considerations are in line with the recommendation of the BPMN specification not to use mixed gateways (OMG, 2011a, p. 288).[20]

3.4. Experimental Results

In this section we summarize the results of our experiment. Section 3.4.1 describes the experiment. Section 3.4.2 presents the results, while Section 3.4.3 provides a discussion.

3.4.1. Modeling Session in Eindhoven

In order to test our conjectures, we designed an experiment that would rely on the use of Cheetah Experimental Platform. The task in this experiment was to create a formal process model in BPMN from an informal description. The object that was to be modeled was the process of preparing the take-off of an aircraft[21]. We decided to use a subset of BPMN for our experiment and provided no sophisticated tool features (e.g. automated layout support or automatic syntax checkers) to prevent the modelers to become confused or overwhelmed with tool aspects (Crapo et al., 2000). A pre-test was conducted at the University of Innsbruck to ensure the usability of the tool and the understandability of the task description. This led to some minor improvements of Cheetah Experiment Platform and a few updates to the task description.

The modeling session was conducted in November 2010 with 103 students following a graduate course on Business Process Management at Eindhoven University of Technology. The modeling session started with a modeling tool tutorial, which explained the basic features of the platform. After that, the actual modeling task was presented according to which the students had to model the process shown in Figure 3.2. By conducting the experiment during class and closely monitoring the students, we mitigated the risk of external distractions that might otherwise have affected the modeling process. No time restrictions were imposed on the students.

3.4.2. Results

We used the collected data of the experiment to calculate the values of the six process-based metrics of Section 3.3.2 for the modeling process of each student. We also determined for each modeling process the value (0 or 1) for the perspicuity metric as a measurement of process model quality. As

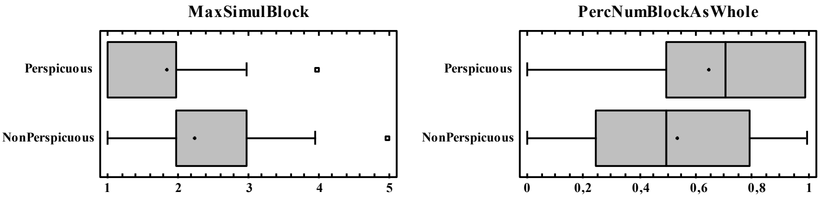
---

[20] More information on the measurement of perspicuity can be found in Appendix E.

[21] The case description is available at: http://bpm.q-e.at/experiment/Pre-Flight.

it turned out, 54 students (52%) managed to create a perspicuous model while the remaining 49 (48%) did not.

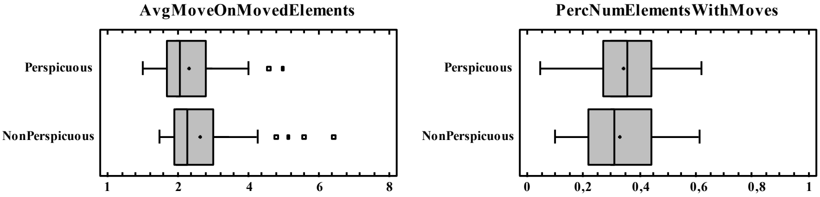
As a next step, we looked at the distribution of the metrical values. All distributions deviated from normality, being more skewed than a characteristic Bell-curve. Therefore, we turned to the representation of these distributions as boxplots (McGill et al., 1978). A *boxplot* (a.k.a. a *box and whisker plot*) consist of a *box*, which represents the middle 50 percent of the data. The upper boundary (also known as the *hinge*) of the box locates the 75th percentile of the data set, while the lower boundary indicates the 25th percentile. The area between these two boundaries is known as the *inter-quartile range* and this gives a useful indication of the spread of the middle 50 percent of the data. There is also a line in the box that indicates the *median* of the data (which may coincide with a box boundary) and a cross that indicates the *average value*. The *whiskers* of the box-plot are the horizontal lines that extend from the box. These indicate the minimum and maximum values in the dataset. If there are *outliers* in the data, shown as open rectangles, the whiskers extend to their maximum of 1.5 times the inter-quartile range. The boxplots for all metrics are shown in Figure 3.6, 7 and 8.


MaxSimulBlock

PercNumBlockAsWhole

Figure 3.6. Boxplots of the metrics for conjecture 1

What can be seen in Figure 3.6 is that people who created perspicuous models tend to simultaneously work on a *smaller* number of blocks (*MaxSimulBlock*) than people who delivered a non-perspicuous model. Overall, those who developed a perspicuous model tend to complete a *higher* percentage of blocks as a whole too (*PercNumBlocksAsWhole)*. Both aspects provide support to conjecture 1.


AvgMoveOnMovedElements

PercNumElementsWithMoves

Figure 3.7. Boxplots of the metrics for conjecture 2

In Figure 3.7 it can be seen that modelers of perspicuous models tend to *less frequently* move elements than the other modelers (*AvgMoveOnMovedElements*); this is in line with conjecture 2. The groups, however, do not seem to differ very much with respect to the overall *number* of elements being moved around (*PercNumElementsWithMoves*). This can be seen from the distributions that cover about the same area. So, this gives no additional support for conjecture 2.

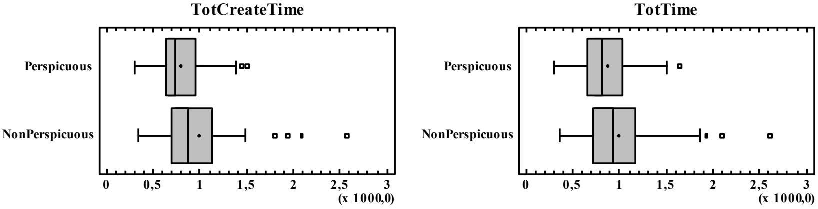


Figure 3.8. Boxplots of the metrics for conjecture 3

Finally, Figure 3.8 shows that the total time between the first and last recorded action of the modeling process (*TotTime*), as well as the total time between the first and last recorded *create* action of the modeling process (*TotCreateTime*), seem slightly lower for the group of modelers who created perspicuous models. It is this insight, i.e., that both distributions for modelers of perspicuous models cover a relatively lower range, that supports conjecture 3.

While these visual insights are promising, it is necessary to subject these to more rigorous testing. For this purpose, we carried out a t-test[22] for each of the six metrics in order to compare the respondents who created a perspicuous model with those who delivered a non-perspicuous model. The results are shown in Table 2.

What can be derived from these results is that there is a significant difference between the groups for all investigated metrics when assuming a 95% confidence interval (i.e., the P-values are lower than 0.05), except for *PercNumElementsWithMoves* (P-value equals 0.648 >> 0.05). In other words, the group of modelers who created a perspicuous model scored *significantly different* than the group who delivered non-perspicuous models with respect to *all our measures but one*, and in *exactly the direction* we conjectured. For example, the respondents who created a perspicuous model indeed were working on a lower maximum number of blocks simultaneously (*MaxSimulBlock*) and completed more blocks as one related whole

---

[22] In large samples, the t-test is valid for any distribution of outcomes (Lumley et al., 2002), even if we can not assume normality as is the case here.

(*PercNumBlockAsAWhole*) than the other group. From these results, we conclude that we have found strong support for conjectures 1 and 3 (i.e., through support for all related metrics), and mild support for conjecture 2 (i.e., via support for just one of the two related metrics).

Table 3.2. Results student t-test

Conjecture	Metric	T-value	df	P-value (sig.)
C1	MaxSimulBlock	-2.231	101	0.028*
	PercNumBlockAsAWhole	2.199	101	0.030*
C2	AvgMoveOnMovedElements	-1.984	101	0.049*
	PercNumElementsWithMoves	0.457	101	0.648
C3	TotTime	-2.183	101	0.031*
	TotCreateTime	-2.505	101	0.014*

(*) statistically significant values at the 95% confidence level

3.4.3. Discussion

Our findings warrant a reflection on their potential impact on research and practice. From a scientific point of view, our study confirms that the properties of a modeling process can be related to its outcome. Specifically, our work shows that aspects of a modeler's style can be operationalized and quantified, providing means to distinguish between more and less effective approaches to create a process model. As such, this work opens the venue towards a more sophisticated understanding of what makes someone a good modeler or, more precisely, what is a good modeling process. Values, beliefs, cognitive abilities, and personality traits may be as important in the field of process modeling as they are in the area of computer programming (see (Cegielski, 2006)). It is also noteworthy that the attractive aspect of structured modeling in particular echoes the large interest for the formal property of *structuredness* in the process modeling field (R. Laue & Mendling, 2008; Vanhatalo, 2007).

From a practical point of view, our findings suggest, cf. the support for conjecture 1, that an approach that emphasizes successive phases of thorough and localized modeling (i.e., within blocks) is more attractive than diverting one's attention across different parts of a model at the same time. Similarly, yet less pronounced via mild support for conjecture 2, excessive reshaping of a model and moving its elements around seem to be anathema to good modeling practice. These are both actionable items that can be shaped into modeling instructions, which can be incorporated in process modeling courses (beyond the more traditional syntactical and formal topics). Our insight with respect to modeling speed, cf. the support for conjecture 3, seems particularly relevant to distinguish more from less proficient modelers. Such an insight may be particularly useful when composing project teams (a fast modeler is an asset, both time- and quality-wise) or assigning modeling tasks to professionals (a faster modeler will deliver a readable model).

The interpretation of our findings is presented with the explicit acknowledgement 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 follow a similar approach to process modeling as that of skillful yet inexperienced modelers. Note that we are cautiously optimistic about the usefulness of the presented insights on the basis of modeling behavior of graduate students, since we have established in previous work that such subjects perform comparably in process modeling tasks as some professional modelers (Reijers & Mendling, 2011).

We cannot claim construct validity: In our approach we derive process metrics at the syntactical level of recorded actions of a modeler and we needed to make slight assumptions on the modelers' intentions to calculate our metrics. Nevertheless, we are hopeful that we can verify the results in later experiments, because the t-tests provided significant results (except for *PercNumElementsWithMoves*).

3.5. Conclusion

This chapter reports on research about the *process of process modeling* by examining relations between the modeling process and the modeling outcome (i.e., a process model). We have been particularly interested in the notion of understandability as a quality criterion for process models and searched for related properties of the modeling process that would ensure an *understandable* modeling result.

We formulated three conjectures, i.e., that (i) structured modeling ties to model quality, whereas (ii) lots of movement of modeling objects, and (iii) low modeling speed relate to low model quality. To validate or reject these conjectures, we performed an experiment with 103 modelers and recorded for each modeler all the actions performed with the modeling tool. This allowed us to measure the related concepts of our conjectures (i.e., structuredness, movement, speed, and understandability) in metrics on the modeling process and the modeling result. T-tests point at significant differences, in line with our conjectures about the quality of the model in terms of its perspicuity. We believe this provides firm empirical support for two of our conjectures and, to a lesser extent, for the remaining one.

This chapter forms a basis for a deeper understanding of the process of process modeling and its impact on the quality (*in casu* understandability) of the resulting process model. If we manage to better comprehend the factors that directly influence the result of the modeling process, we would be able to comprise this knowledge in training and tools supporting process modeling. This, in turn, could result in more understandable process models, as well as a more efficient modeling process.

In this chapter, we have limited ourselves to visual inspection of the distributions and t-tests to study three conjectures. The next chapter describes the identification of further factors describing the process of process modeling and the assessment of their influence on the quality of the resulting process model. Next to this further investigation of the collected data set, we have validated our observations in modeling sessions while varying the modeling task to be able to generalize our findings. In the future, we also wish to include modeling experts to be able to observe a more heterogeneous group of modelers during the act of modeling.

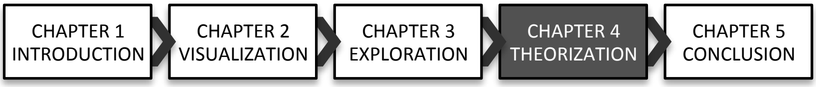
What is also open to further study is how effective modeling instructions can be developed on the basis of our findings. Beyond instruction, we expect that tool support may be another important ingredient in achieving good modeling practice.

4

The Structured Process Modeling Theory (SPMT)

A cognitive view on why and how modelers benefit from structuring the process of process modeling

Published as J. Claes, I. Vanderfeesten, F. Gailly, P. Grefen, G. Poels, The Structured Process Modeling Theory (SPMT) - A cognitive view on why and how modelers benefit from structuring the process of process modeling, Information Systems Frontiers, Vol 17(6), 2015.



Summary. After observing various inexperienced modelers constructing a business process model based on the same textual case description, it was noted that great differences exist in the quality of the produced models. The impression arose that certain quality issues originated from cognitive failures during the modeling process. Therefore, we developed an explanatory theory that describes the cognitive mechanisms that affect effectiveness and efficiency of process model construction: the Structured Process Modeling Theory (SPMT). This theory states that modeling accuracy and speed are higher when the modeler adopts an (i) *individually fitting* (ii) *structured* (iii) *serialized* process modeling approach. The SPMT is evaluated against six theory quality criteria.

Keywords. Business Process Modeling, Process of Process Modeling, Explanatory Theory, Structured Process Modeling, Cognitive Fit.

4.1. Introduction

For the design and analysis of information systems for organizations, analysts typically deal with the complexity of the organization by using conceptual models. These models abstract from specific instances and represent the generic properties of the modeled system. The focus in this dissertation is on process models, which are considered to be a specific kind of conceptual models. A process model is a mostly graphical representation that documents the different steps that are or that have to be performed in the execution of a particular process under study, together with their execution constraints such as the allowed sequence or the potential responsible actors for these steps (Dumas, 2013; Weske, 2007, Definition 2).

The recent developments in research about process models can be classified into three research streams. One stream studies the *application* of process models. For example, the construction of process models has shown to be a key success factor in process redesign (Kock et al., 2009; Xiao & Zheng, 2012), software development (Krishnan et al., 1999), and communication (Abecker et al., 2000; Davies et al., 2006). Therefore it is important that the quality of process models is high.

A second research stream is thus investigating the *quality* of process models. Traditionally, it is believed that the quality of the model has to be evaluated relative to the purpose of the model (Juran & Gryna, 1988; Lindland et al., 1994). An abundance of process model quality dimensions and metrics, targeted at various purposes, has thus been examined (Nelson et al., 2012; Vanderfeesten et al., 2007). For example, if the process model is created as a tool for communication about a particular process, the comprehensibility of the model by its intended readers can be regarded as an important quality dimension. In case the process model has to serve as input for a process-aware information system, syntactic correctness and semantic completeness may be considered to be more crucial. An extensive overview of quality dimensions and related metrics is presented by (Sánchez-González et al., 2013) in their systematic literature review on process model quality research.

Recently, a third stream of process model research originated that shifts the focus from investigating *what* are characteristics of a good process model towards the study of *how* good process models are constructed. For instance, (Brown et al., 2011) investigated how the use of virtual world technology increases modeler empowerment and consensual development during modeling in collaborative settings. Collaborative process modeling and how technology supports this activity is also the subject of (Recker et al., 2013). Further, (Pinggera, Soffer, et al., 2013) identified three process modeling styles relating to variations in modeling speed and model reconciliation. Lastly, (Claes et al., 2015) developed a visualization that represents how process models are created in terms of consecutive operations on the model in a modeling tool.

Similar trends of research shift existed already in the broad field of conceptual modeling (e.g., Hoppenbrouwers et al., 2005) and the even more general area of system analysis and development (e.g., Chakraborty et al., 2010; Nunamaker & Chen, 1990). The underlying assumption in all of these studies is that the quality of the product depends on the quality of the process that creates the product, at least to some extent. Based on the observations described in this chapter, we subscribe to this assumption and presume that certain quality concerns are caused during the modeling process. Therefore, we abstract from the different process model quality dimensions and study the cognitive mechanisms during the process of process modeling in which these quality issues originate.

We define the process of process modeling (PPM) as the sequence of steps a modeler performs in order to translate his mental image of the process into a formal, explicit and mostly graphical process specification: the process model (see Definition 3). The modeler forms a mental representation of the process based on direct observation and/or various descriptions of the real or intended process such as interview transcripts, whiteboard notes and requirements documents (Chakraborty et al., 2010). It should also be noted that mental models are rarely stable, they keep evolving as more information is processed (Rogers & Rutherford, 1992). Hence, the transformation of the (individual and dynamic) mental model into an explicit process model is a complex cognitive task. During this task the modeler iterates between shaping the mental model, evaluating the mental model, converting the mental model into a formal model, evaluating the formal model, adapting the mental model, etc.

Throughout this complex task the modeler is hindered by his cognitive limits, which results in cognitive ineffectiveness that can be manifested by a decrease of accuracy and speed (Rockwell & Bajaj, 2005; Sweller, 1988). Therefore, the end goal of our research is to help the modeler to reduce these negative effects by developing a method for process modeling that should warrant the optimal use of a modeler's cognitive functions. As advocated by (Avgerou, 2000; Naumann, 1986), a first, fundamental step for achieving this goal is the collection and description of the necessary knowledge that helps understand why, how and when cognitive failures occur during process modeling. Because such knowledge is not currently readily available, this chapter handles entirely on its development. The presented contribution is the Structured Process Modeling Theory (SPMT), which explains why the modelers that implement an optimal structuring approach towards modeling may deal with the complexity of the task in a more cognitively effective and efficient way.

The SPMT being an explanatory theory will indeed serve as a foundation for the development of a method that prescribes how to create process models in a cognitively optimal way. Furthermore, the SPMT brings together cognitive theories about learning and problem solving in a fundamental new way and has the potential to "give back" to the cognition research field, because of the novel view on (the combined application) of these theories. This chapter has practical significance by providing knowledge that can be used for process modeling training, for differentiated tool development, etc.

In Section 4.2, the methodology that was used to build and to test the SPMT is discussed. The theory was developed by adapting and combining cognitive theories to the context of process modeling in order to explain the varying success of different observed modeling approaches. The way these observations were collected is described in Section 4.3. Subsequently, Section 4.4 provides the theoretical background for the developed SPMT, which itself is presented in Section 4.5. Next, the SPMT is evaluated in Section 4.7. The context of the research is outlined in Section 4.8, which summarizes related work. Finally, Section 4.9 contains an extensive discussion and a brief conclusion is provided in Section 4.10.

4.2. Research methodology

Multiple research paradigms exist in Information Systems amongst which design science and behavioral science are prevalent paradigms (March & Smith, 1995). Design science is centered on the development of research artifacts such as constructs, models or methods, which have to possess value or utility (Hevner et al., 2004). Behavioral science is concerned with developing knowledge about human behavior, represented by theories (Simon, 1996). The selection of the research methodology depends on the research question. Based on descriptions in the introduction, this question can be phrased as:

- ***RQ.*** *Why do people struggle with the complexity of constructing a process model?*

The above research question asks for explanations of human behavior and thus an explanatory theory was developed that describes the cognitive leverages that play a role while constructing a process model. An explanatory theory *"provides explanations but does not aim to predict with any precision. There are no testable propositions."* (Gregor, 2006, p. 620). Other types of theories exist as well. A predictive theory for example does not provide explanations, but it does include testable propositions with predictable effects. Next to descriptive theories such as the explanatory or predictive theories, also prescriptive theories exist. Instead of only describing, explaining or predicting relations between constructs, they offer concrete prescriptions and relate the proposed actions to certain consequences. (Gregor, 2006)

4.2.1. Theory building

The input for theory development may include (objective) observations (Godfrey-Smith, 2009; Nagel, 1979), as well as (subjective) impressions (Popper, 2005). New theory can then be developed by searching for explanations for the observations and impressions (Weick, 1989). In order to collect observations and impressions about how modelers construct process models, exploratory modeling sessions were performed (see Section 4.3). An explanation for the observed relations between modeling approach and cognitive failures was searched for in cognitive literature. Section 4.4 explains how cognitive theories propose that the human brain is limited in handling complex tasks and if the brain gets overloaded, modelers tend to work slower and make more mistakes. These theories can explain the observed behavior and varying success of the modelers while constructing process models. We compiled and synthesized these theories into the central contribution of this chapter: the Structure Process Modeling Theory (SPMT), presented in Section 4.5.

4.2.2. Theory testing

For most theories, the actual value can only be measured on the long term, by evaluating its actual use by others (Weick, 1989). Nevertheless, in literature about theory in the information systems domain six assessable criteria for good (explanatory) theories were found: i.e., novelty, parsimony, consistency, plausibility, credibility, and transferability (Gregor, 2006; Grover et al., 2008; R. Weber, 2012; Weick, 1989). Section 4.7 elaborates on the assessment of the SPMT against these criteria. For the evaluation of consistency, a second series of observational modeling sessions was examined in order to assess to what extent the described theory can be used to explain the additional observations.

4.3. Problem exploration

In order to explore how people construct process models, structured sessions were performed in which participants were asked to construct a business process model based on a given textual case description. These observational sessions supported the collection of the data that were studied in order to collect the observations and impressions that served as input for the development of the Structured Process Modeling Theory (SPMT).

4.3.1. Data collection method: observational modeling sessions

During the exploratory modeling sessions it was observed how the modelers constructed a process model from a textual case description. The participants were instructed to aim for a high quality model. It was, however, not defined what was meant by ‘high quality model’.

Case. The case to be modeled described the steps in the request handling of mortgages by a bank[23]. A textual description was handed over to the participants and comprised two A4 format sheets excluding instructions. The process models that were built by the participants contained on average 27 activities and construction took on average 276 recorded modeling operations in the tool (see further). This size indicates the complexity of the case and the modeling task according to (Mendling, 2008), which will be further discussed in the following sections.

Participants. In order to gain knowledge about how inexperienced modelers deal with the complexity of a case throughout a process modeling endeavor, master students that attended a course in Business Process Management were selected as primary target group. The sessions were strategically planned after the lectures in which the students were introduced into process modeling, but before the training of specific modeling techniques or guidelines. This way a group was formed of participants that have enough maturity and knowledge about process modeling without possessing an abundance of modeling experience. The focus was on inexperienced modelers, because they did not yet consciously learn any technique to cope with the complexity of a modeling task, which we expected to result in more variety in the observations and a more open search for potential interesting modeling approaches. The observational

---

[23] Case description can be downloaded from http://bpm.q-e.at/experiment/MortgageEindhoven.

modeling sessions took place in December 2012 at Eindhoven University of Technology. The group of participants was composed of 118 master students in total, distributed over three different educational programs (i.e., Operations Management & Logistics, Innovation Management, and Business Information Systems). The mixture of educational profiles from technical-oriented to business-oriented students has the advantage of increasing the likelihood that a heterogeneous set of observations is obtained. Participation was voluntarily and the students could stop at any time without handing in a solution.

Modeling language. A simplified modeling language was used for the modeling sessions. It contained constructs representing the main *concepts* of a control flow model[24]: start node, end node, activity, sequence flow, parallel branch (split and join), and optional branch (split and join). These constructs were chosen because they are found in the majority of currently used process modeling languages (e.g., BPMN, EPC, Petri-Net, UML Activity Diagrams, Workflow Net, YAWL, etc.) Moreover, they are considered the most used constructs for process modeling (Marcello La Rosa et al., 2011; Zur Muehlen & Recker, 2008). The advantage of this approach is that the results can be transposed to existing or perhaps also future process model notations and the modeler could not be hindered by an abundance of model language constructs. The BPMN *symbols* for the constructs were used in order to be easily understood by the participants, who were familiar with the BPMN notation. This latter process model notation was used in a number of lectures of the BPM course in which the participants were enrolled.[25]

Supporting tool. The Cheetah Experimental Platform[26] (Pinggera, Zugal, & Weber, 2010) was used to support the data collection. This program was developed at the University of Innsbruck as an open source research platform to support experiments investigating the process of process modeling. The modeling sessions were entirely supported by this tool and consisted of three consecutive tasks. The *tool tutorial task* presented short videos together with a brief explanation to exemplify each feature of the modeling editor. To reassure that the tool features were sufficiently understood, the user had to mimic the actions of the video in the modeling editor correctly before the next feature was presented. Next, in the *process modeling task* the participants had to construct a process model for the

---

[24] A control flow model is a process model that mainly represents the sequential order of process steps (i.e., the control flow).

[25] More information about the used modeling language can be found in Appendix D.

[26] More information about the tool can be found at http://www.cheetahplatform.org.

given case description. Finally, the *survey task* had to be completed by answering a questionnaire.

Data collection. The experimental tool recorded each modeling operation automatically in an event log. A list of the different types of operations that were recorded, is presented in Appendix D. Besides the name of the recorded operation, the event records contained additional information such as the time of its occurrence, position on the canvas, source and target activities of edges, etc. These data can be used for a step-by-step replay of the model construction process or to feed mining algorithms that support analyses of this process (such as the PPMChart visualization). Furthermore, the tool captured the constructed process models, which allows for inspecting different properties of the produced models. Finally, the questionnaire (see Appendix F) was used to collect data about the demographics of the respondents, as well as domain knowledge, modeling language and method knowledge and general tool and language issues.

4.3.2. Data analysis

The answers to the demographic questions about the participants revealed that they are students between 20 and 28 years old, mainly male (93 out of 118). The majority of participants were non-native English speaking, but only 2 of them indicated to have some difficulties in reading or understanding English text. Table 4.1 presents an overview of the demographical data. Further, the students indicated to have had an average of 5.7 workdays of formal training on process modeling and 8.7 workdays of self-education. The mental effort was rated between 2 and 7 out of 10 (4.4 on average). Participants indicated they had no problem understanding the case description or working with the tool. More details about the prior knowledge of participants are provided in Appendix G.

Table 4.1. Demographic information of participants

Gender	Age	Native language	Current profession	Education program
93 Male 25 Female	1 age 20 7 age 21 36 age 22 38 age 23 22 age 24 11 age 25 2 age 26 1 age 28	99 Dutch 3 Chinese 2 English 2 Greek 2 Russian 1 Danish, French, German, Indonesian, Macedonian, Persian, Polish, Portuguese, Romanian	116 Student 1 Part-time student 1 PhD student	86 Operations Management & Logistics (OML) 25 Business Information Systems (BIS) 4 Innovation Management (IM) 3 Human-Technology Interaction (HTI) 1 Professional Doctorate 1 Doctorate (PhD)

PPMChart visualization

The PPMChart visualization represents the operations of the construction process of one modeler that produced a single process model (Claes et al., 2015). An example is shown in Figure 4.1. The chart consists of horizontal timelines, one for each model element that was present during modeling. The top-down ordering of these timelines is derived from the sequence flows in the process model. Each colored dot in the graph represents one operation on one model element on the modeling canvas with the following characteristics:

- *The line of the dot represents the model element on which the operation was performed (the identifier of the model element is displayed at the beginning of the line).*
- *The position of the dot on the line represents the time when the operation occurred (the default width of a PPMChart is one hour).*
- *The color of the dot represents the type of operation (i.e., green for creation, blue for movement, red for deletion, orange for (re)naming, and grey for reconnection of edges).*
- *The shape of the dot represents the type of model element of the operation (i.e., circle for events, square for activity, diamond for gateways, triangle for edges).*

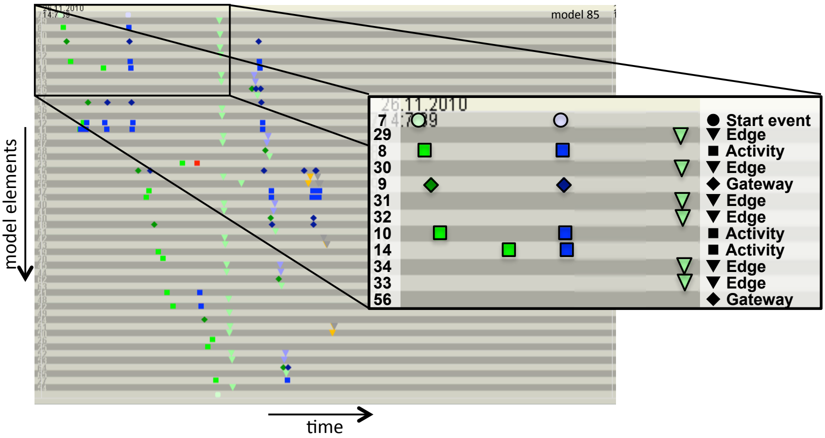


Figure 4.1. PPMChart visualization representing one process modeling instance

For example, in the annotated highlight of Figure 4.1 it can be observed that the first created element (i.e., the far most left green dot) was the start event (i.e., a circular dot on the first line). Next, an activity was put on the canvas somewhat later (i.e., a green square dot on another line, slightly more to the right). After the creation of some elements (i.e., left vertical zone of green dots), an almost simultaneous movement of all existing elements can be observed (i.e., vertical blue line of dots). Only much later, the edges that connect these elements were created (i.e., line of green triangular dots at the right).

Process model quality

(Lindland et al., 1994) define three main quality dimensions of conceptual models: (i) *syntactic quality* indicates to which degree the symbols of the modeling language were used according to the rules of the language, (ii) *semantic quality* indicates how adequate the model represents the modeled phenomenon in terms of correctness and completeness, and (iii) *pragmatic quality* indicates the extent to which the users of the model understand the model as intended by the modeler.

In the study of the observational modeling sessions, only syntactic quality was evaluated to form impressions of the quality of the produced models. This dimension was selected because a measurement can be determined easily and objectively on the basis of the modeling language specification. It was assumed that the syntactic quality provides a sufficient insight at this stage of the research. Moreover, both semantic quality and pragmatic quality include facets that are beyond the scope of the research as defined in Definition 3. Indeed, besides the course of the modeling process, the information gathering process in which the modeler forms a mental image of the process to be modeled also determines semantic quality. Pragmatic quality is also determined by characteristics of the model reader.

Furthermore, a distinction was made between errors that originate in a lack of knowledge of the modeling language and errors that originate in cognitive failure. Especially the latter type of error is interesting for investigating our research question. In the remainder of the chapter, the term *'mistake'*[27] is used to identify those syntactic errors in the process models, which did not clearly arise from a lack of knowledge of the process modeling language. A list of the observed syntax errors is included in Appendix E, together with their classification in *'mistakes'* and other syntactic errors.

---

[27] The term '*mistake*' as used in this chapter is not related to the well-known classification of human errors in slips and mistakes by Reason (1990). Our distinction is based on the cause of the error, while the distinction of Reason is based on the type of error.

Observations and impressions about the modeling process

PPMCharts allow for zooming in on specific operations (i.e., on individual dots in the charts), as well as on aggregated modeling phases and patterns (i.e., combinations of dots in the charts). Different PPMCharts were compared to extract patterns that reflect identifiable modeling approaches. This section presents a selection of such observations together with our impressions about the relation between these approaches and the properties of the resulting process models.

Serializing the modeling process

- ***Definition 10: Serialization***
  When tasks are complex, people tend to deal with task complexity by splitting up the task in implicit subtasks that are executed sequentially (De Jong, 2010). This complexity management technique is called serialization.

Because this technique needs some cognitive administration, it results in frequent pauses during the modeling process in which no visible activities occur. From the 118 recorded modeling sessions, it was observed that in all but one of the sessions pauses were observed in the modeling replays as a timespan in which no operations occurred. These pauses were also evidenced in the PPMCharts as a vertical zone in which no dots occur.
Observation 1: *All but one of the modelers paused frequently during the modeling process.*

Of course, different events can have caused these pauses. Potentially the modeler was distracted, the modeling tool was lagging, the modeler was reading the case description, the modeler was thinking about the previous or the next steps, etc. Because often a high concentration of dots was observed right after a pause, the pauses seemed to us to be deliberate interruptions of the modeling pace in which the previous and/or future modeling operations were considered. It gave us the impression that the modelers needed to serialize the modeling process.
Impression 1: *Modelers are in need of serializing the modeling process to deal with its complexity.*

Structuring the modeling process

- ***Definition 11: Structured serialization***
  Whereas serialization is defined as splitting up a task in sequentially executed subtasks, structuring can be defined as the extent to which a consistent strategy is applied for defining those subtasks.

The way of (not) structuring the modeling process can be recognized in the PPMChart by the patterns that can (not) be clearly discovered in the arrangement of the dots in the chart.

- ***Definition 12: Flow-oriented process modeling***
  Flow-oriented process modeling means that the modeler constructed parts of the process model according to the control flow structure of the process. Once a part of the model was considered complete, these modelers did not change that part of the model anymore.

The analysis of the PPMCharts revealed that 33 of the 118 modelers (28%) have built the process model in a flow-oriented way. In the PPMChart, because of the sorting of lines according to the sequence flows in the model, this style was observed as a diagonal zone of operations (see Figure 4.2).

Observation 2: *A large group of the sessions can be categorized as "flow-oriented process modeling".*

- ***Definition 13: Aspect-oriented process modeling***
  Aspect-oriented process modeling is observed when the modeler consecutively directs attention to different aspects of modeling[28]. They may for example first focus on the content of the model (i.e., placing every activity and gateway on the canvas), then on the sequence flow of the activities (i.e., connecting the elements with sequence flow arrows), and finally on the layout of the model (i.e., moving and aligning elements).

Conversely, 10 other modelers (8%) organized the modeling process in an aspect-oriented way. In Figure 4.3 aspect-oriented process modeling can be observed as several non-overlapping zones each enclosing similar operations.

Observation 3: *A smaller group of the sessions can be categorized as "aspect-oriented process modeling".*

---

[28] The term aspect-oriented process modeling must not be confused with aspect-oriented modeling. The former is our description of splitting up the modeling *process* according to the aspects that are targeted sequentially. The latter is a way of splitting up the model itself in sub-models that each represents another aspect of the system to be modeled. Both terms are derived from aspect-oriented programming, a technique for splitting up the programming process as well as the program code according to the different aspects to be programmed.

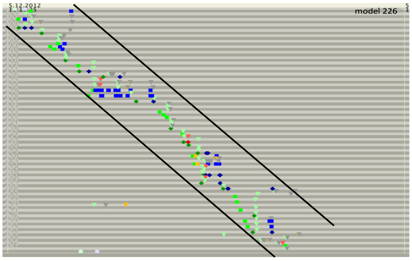


Figure 4.2. Example of flow-oriented process modeling

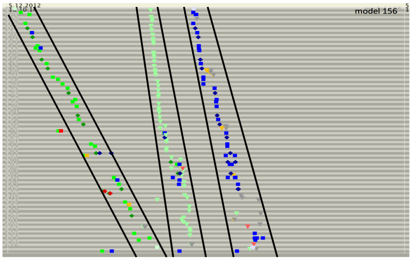


Figure 4.3. Example of aspect-oriented process modeling

Whereas many situations were discovered in which a flow-oriented or aspect-oriented organization of the modeling process was used consistently, it should be noted that also combinations were observed in 33 of the 118 cases (28%). Figure 4.4 shows an example where process model elements are *created* in a flow-oriented manner, but overall the modeler alternated between dedicated phases of working on different modeling aspects. This can be observed by a diagonal zone of green dots followed by a number of zones of limited height that each consist of similar dots.

Observation 4*: Another large group of the sessions used a combination of "flow-oriented process modeling" and "aspect-oriented process modeling".*

It should also be noted that not every modeler seemed to implement a particular way of organizing the modeling process, as could be concluded from Figure 4.5. No clear pattern of dots was discovered in the charts. A subset of 12 of the 118 instances (10%) was labeled "undirected process modeling". The term "undirected" is preferred over "unstructured", because it is physically impossible for most people to perform actions without any form of structured approach.

Observation 5*: Another small group of the sessions can be categorized as "undirected process modeling".*

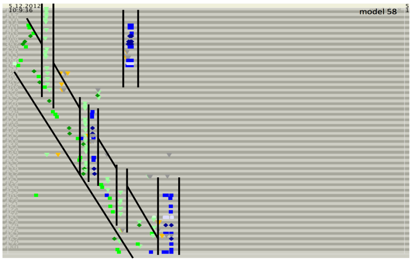


Figure 4.4. Example of a combination of flow- and aspect-oriented process modeling

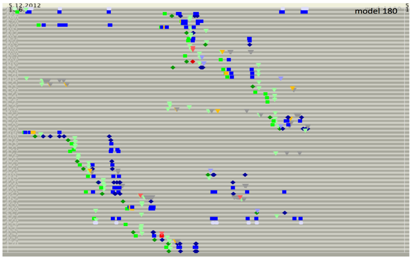


Figure 4.5. Example of undirected process modeling

So far, three structuring strategies for serialization were observed: flow-oriented process modeling, aspect-oriented process modeling, and a combination of both approaches. We also observed undirected process modeling.

The remaining 30 sessions (25%) could not clearly be categorized as structured or undirected. They were labeled "uncategorized" and were left out of scope for further analysis.

In order for a process model to be syntactically correct, no syntax errors may exist in the model. The number of '*mistakes*' in each model was assessed by the authors. In their assessment, for certain syntactical errors they had to make a subjective decision whether the error should be classified as a '*mistake*' or not (see classification of errors in Appendix E). From the models constructed according to a structured serialization strategy (i.e., flow-oriented, aspect-oriented, and combination of both approaches), a bigger proportion seemed to contain no '*mistakes*' than models originated from the undirected modeling approach. The impression arose that serializing the modeling process in a structured way helps avoiding these errors caused by cognitive failure.
Impression 2*: Structured serializing of the modeling process helps avoiding 'mistakes'.*

However, not every model that was created in a structured way ended up containing no '*mistakes*'. This could for example be explained if factors exist that counter the effect of the structured approach. Nevertheless, no factors hindering the modeler were observed and for that reason we got the impression that the structuring did not help every modeler in the same way.
Impression 3*: Structured serializing does not support every modeler to avoid 'mistakes' to the same extent.*

Speed of the modeling process

Finally, it was observed how in some PPMCharts the zone that contains dots is narrower than in other charts (e.g., compare Figure 4.4 with Figure 4.5). This means that some modelers took less time to construct the process model. A comparison of time distribution of the four defined serialization strategies revealed that the modeling sessions of the category "undirected" lasted clearly longer than the three other categories (see Figure 4.6). Independent t-tests indicated that the mean modeling time of the three structured approaches was significantly different from that of the undirected approaches ($p_{FO\text{-}UD}$=0,023, $p_{AO\text{-}UD}$=0,012, $p_{C\text{-}UD}$=0,000). The structured approach seems not only to help reducing the number of '*mistakes*'; it may also speed up the modeling process.

Observation 6: *The sessions labeled "undirected process modeling" lasted longer than the other approaches.*

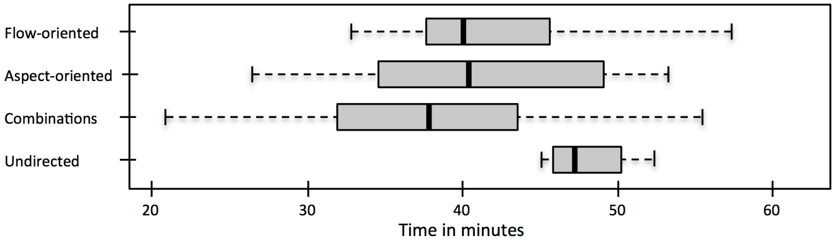


Figure 4.6. Boxplot of modeling time for each of the observed serialization styles

Overview

An overview of the data regarding Observations 1 to 6 is presented in Table 4.2. The outcomes of the exploratory study are the six observations and the three impressions, summarized in Table 4.3. As proposed by (Godfrey-Smith, 2009; Nagel, 1979; Popper, 2005) both observations and related impressions can then be used as input to build a theory, which is described extensively in Sections 4.4 and 4.5.

Table 4.2. Observed serialization strategies and their measured properties

serialization strategy	number of cases	number of serialized cases	mean modeling time
Flow-oriented modeling	33/118 (28%)	33/33 (100%)	42,80 ± 9,56 min.
Aspect-oriented modeling	10/118 (8%)	10/10 (100%)	40,32 ± 9,44 min.
Combined modeling	33/118 (28%)	33/33 (100%)	37,37 ± 7,92 min.
Undirected modeling	12/118 (10%)	11/12 (99%)	49,87 ± 6,63 min.
Uncategorized cases	30/118 (25%)	30/30 (100%)	-

Table 4.3. Overview of the defined observations and impressions

Observations	
Observation 1	All but one of the modelers paused frequently during the modeling process.
Observation 2	A large group of the modeling sessions can be categorized as "flow-oriented process modeling".
Observation 3	A smaller group of the sessions can be categorized as "aspect-oriented process modeling".
Observation 4	Another large group of the sessions used a combination of "flow-oriented modeling" and "aspect-oriented modeling"
Observation 5	Another small group of the sessions can be categorized as "undirected process modeling".
Observation 6	The sessions labeled "undirected process modeling" lasted longer than the other approaches.

Impressions	
Impression 1	Modelers are in need of serializing the modeling process to deal with its complexity.
Impression 2	Structured serializing of the modeling process helps avoiding '*mistakes*'.
Impression 3	Structured serializing does not support every modeler to avoid '*mistakes*' to the same extent.

4.4. Theoretical background

Different cognitive theories can be combined to provide explanations for the observed modeling approaches and their relation with modeling accuracy and speed. In the next section, the theoretical background presented here is used to formulate a theory for explaining modelers' cognitive strategies for dealing with complexity.

4.4.1. Kinds of human memory

The literature on cognition describes three main kinds of human memory. *Sensory memory* is very fast memory where the stimuli of our senses are stored for a short period (Sperling, 1963). During this instant the information that is unconsciously considered relevant is handed over to working memory (Sperling, 1963). Next, the information in working memory is complemented with existing knowledge that is retrieved from *long-term memory* (Sweller et al., 1998). This latter kind of memory is slow but virtually unlimited (Sweller et al., 1998). Information is stored in long-term memory as cognitive schemas composed of patterns of connected elementary facts (Sweller et al., 1998). Relevant information for process modeling that is retrieved from long-term memory includes domain knowledge, and modeling language and modeling method knowledge. In *working memory* the information is organized and processed in order to initiate certain performances (e.g., to put an activity on the modeling canvas with a mouse click) or to complement the knowledge in long-term memory (e.g., to complement the mental model of the case with new insights from a line of text that was read) (Atkinson & Shiffrin, 1968). Because working memory has a limited capacity (Cowan, 2010; Miller, 1956) and information can only be stored in this memory for a short period (Van Merriënboer & Sweller, 2005), it is important to use it effectively when dealing with highly complex tasks, such as process modeling.

4.4.2. Types of cognitive load

Process modeling requires input information to be absorbed and complemented with knowledge from long-term memory such as domain knowledge, in order to be processed in working memory leading to the actions of constructing the process model. The center of this complex task are the operations in working memory (Atkinson & Shiffrin, 1968; Sweller et al., 1998). The necessary information fills up working memory and is subdivided in three types of cognitive load (Sweller & Chandler, 1994). *Intrinsic cognitive load* is the amount of information that needs to be loaded in working memory for deciding how to conduct a particular task. It mainly depends on the properties of the task and the amount of relevant prior knowledge of the performer of the task (i.e., knowledge about the domain, about the modeling language and about the modeling method). *Extraneous cognitive load* is the load that is raised for processing and interpreting the input material of the task such as descriptions or direct observations of the process to be modeled. This type of cognitive load depends on the representation of the input material as well as the fit of this representation with the task it has to support and with the characteristics of the interpreter of the material (Vessey & Galletta, 1991) (see also Section 4.4.4). Finally, during the execution of a task humans usually are able to reserve some load in working memory for building, restructuring and completing cognitive schemas to be stored in long-term memory. This will help reducing cognitive load for performing similar tasks in the future. This activity is called learning and the associated load is the *germane cognitive load*. Furthermore, a distinction can be made between the *overall* cognitive load (i.e., the total amount of information sequentially loaded in working memory for performing a specific task) and *instantaneous* cognitive load (i.e., the amount of information that is loaded in working memory at a certain point in time) (Paas, Tuovinen, et al., 2003).

4.4.3. Cognitive Load Theory

The capacity of *working memory is limited*. In the past, researchers have tried to define how much information can be loaded at the same time in this kind of memory. Miller estimated the amount of information that can be remembered in short term memory at about 7 units (Miller, 1956). More recent research concludes that only 3 to 4 units of information can be activated and processed in working memory at the same time (Sweller et al., 1998; Van Merriënboer & Sweller, 2005). Although there appears to be a limit on the amount of units that can be loaded simultaneously in working memory, there seems to be no constraint on the size and complexity of these units of information (Sweller et al., 1998). More specifically, it is believed that one unit of information loaded in working memory (often referred to as 'information chunk') corresponds with one cognitive schema in long-term memory (Sweller et al., 1998). This can explain why a person seems to be able to store more information in working memory for tasks in which he is experienced, because for such tasks he was able to build up larger and stronger cognitive schemas in the past.

Therefore, for complex tasks or tasks in which a person is not adequately experienced, it is imaginable that the limited capacity of the working memory is not sufficient for the (maximum instantaneous) load that is needed to accomplish the task. This is called *cognitive overload* (Sweller, 1988). The Cognitive Load Theory states that when working memory is overloaded, there is *no room for learning* (i.e., schema building) and *accuracy and speed* of information processing *decrease* (Rockwell & Bajaj, 2005; Sweller, 1988). In other words, cognitive overload has a negative impact on the effectiveness and efficiency of the modeling performance.

4.4.4. Cognitive Fit Theory

The *Cognitive Fit Theory* states that humans are able to solve problems more effectively and efficiently if the *representation of the input material of a certain task 'fits' with the task itself* (Vessey, 1991). For example, when a task involves exploring relationships between data a visual representation such as a diagram is preferred. For more statistical purposes such as determining the average of a series of numbers a textual representation in the form of a list or table is more cognitive efficient (Vessey, 1991). Whereas the focus of Cognitive Fit Theory is on the match between problem representation and task, a secondary effect is described as the *match between the task and its performer*. For example, most people excel in either graphical or logical tasks (Pithers, 2002). The former type of people probably needs less effort to work on the layout of the model, whereas the latter may find it easy to warrant the semantic correctness of the model. For the development of our theory, we focused mainly on this secondary relation between task and performer. Since the initial publication of the theory in 1991, the work is refined and concepts of domain knowledge, method knowledge and problem solving tools are taken into account as well (Khatri, Vessey, Ram, et al., 2006; Khatri, Vessey, Ramesh, et al., 2006; Shaft & Vessey, 2006; Sinha & Vessey, 1992; Vessey & Galletta, 1991).

4.4.5. Overview

Figure 4.7 provides an overview of the reviewed cognitive theories and integrates them into a conceptual framework displaying the causal relations that might explain the phenomenon of cognitive overload in working memory during task performance. The task under consideration is the construction of a process model. The central construct of the constructed theoretical framework is cognitive overload, which depends on the modeler's working memory capacity and the cognitive load that the task requires. This cognitive load is composed of extraneous, intrinsic and germane cognitive load.

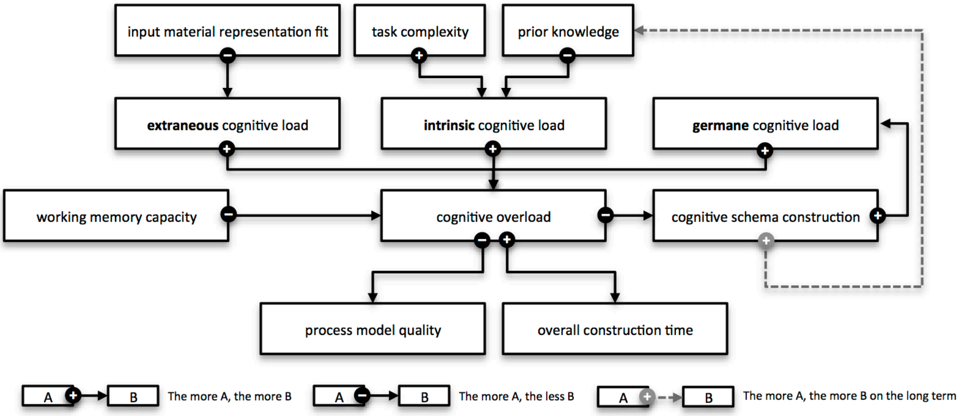


Figure 4.7. Causal model centered on cognitive overload in working memory

Extraneous cognitive load mainly depends on the input material representation fit with the task and the modeler. A higher fit requires a lower cognitive load. For process modeling the input material includes any descriptive process information such as verbal or oral transcripts of interviews with process managers/workers and existing documents describing the process.

The intrinsic cognitive load increases for more complex tasks and decreases in case the modeler possesses more relevant prior knowledge. Differences between various process modeling tasks are mainly related to the complexity of the case to be modeled. Prior knowledge incorporates domain knowledge, and modeling language and method knowledge.

Germane cognitive load is caused by loading information in working memory for the construction of cognitive schemas, which is not a prerequisite for the task, but rather the result of learning. This can only occur if during previous processing of information the working memory was not overloaded.

If the sum of these three types of cognitive load at a certain point in time transcends working memory capacity, cognitive overload occurs. This has a negative effect on process model quality (i.e., more '*mistakes*' are made), speed of modeling, and learning. Note that learning means that the set of cognitive schemas of the modeler is broadened and strengthened, which gradually improves the useful knowledge of the modeler for future similar tasks.

4.5. The Structured Process Modeling Theory (SPMT)

In order to explain the observations and impressions presented in Section 4.3, the cognitive theories listed in Section 4.4 were integrated and transformed into the newly developed Structured Process Modeling Theory (SPMT). Three key concepts were extracted from the observations and impressions: serialization, structuring and individual differences. Therefore the SPMT consist of three parts, each targeting one of these concepts.

4.5.1. Part 1: Serialization of the process modeling task can reduce cognitive overload

We observed that inexperienced modelers use a serialization approach to construct the process model (i.e., Observation 1). Our impression was that the serialization appears to help these modelers dealing with the complexity of the modeling task (i.e., Impression 1). Cognitive theories also recognize the concept of cognitive serialization to deal with cognitive overload. If a task requires too much information to be stored in working memory simultaneously, then it is advised to load the information sequentially (Bannert, 2002; De Jong, 2010; Gerjets et al., 2004; Paas, Renkl, et al., 2003; Pithers, 2002; Pollock et al., 2002; Van Merriënboer et al., 2003). This means that intrinsic cognitive load can be spread out over a longer period, which reduces the probability of instantaneous cognitive overload (De Jong, 2010).

On the other hand, serialization causes more intrinsic cognitive load for integration and for administration of the sequentially processed and produced information (Gerjets et al., 2004). In other words, extra load is created to aggregate the information of the separate parts of a solution and for building the modeling strategy (Gerjets et al., 2004; Van Merriënboer et al., 2003). The modeling strategy determines how to divide the modeling task in subtasks, in which order to proceed, how to execute each subtask, how to aggregate the different partial results, etc. The extra load for aggregation and strategy building results in a total *overall* intrinsic cognitive load that can be higher in case of serialization. But if the intrinsic load for aggregation and strategy building can be kept low, the maximum *instantaneous* load decreases together with the probability of cognitive overload.

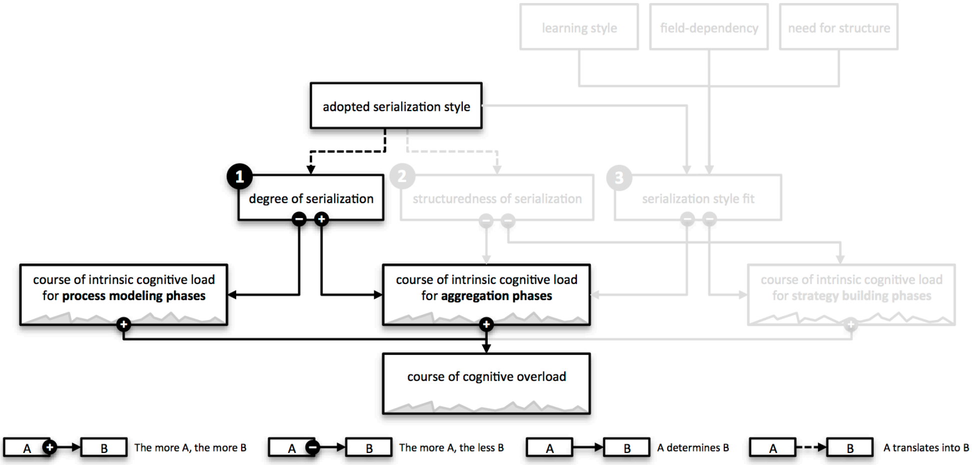


Figure 4.8. The effect of serialization on cognitive overload

Figure 4.8 shows these relations graphically. The adopted serialization style symbolizes how the model construction was serialized (e.g., flow-oriented, undirected, etc.). The *degree* of serialization indicates how much the modeling was subdivided. The *structuredness* of the serialization indicates the consistency of the implemented serialization strategy. According to the aforementioned phases of model parts creation, information aggregation and modeling strategy building, the intrinsic cognitive load is artificially subdivided into three subtypes of intrinsic load, but it is practically impossible to distinguish between those three types (Gerjets et al., 2004). Whereas the degree of serialization mainly impacts the intrinsic cognitive load for process modeling and aggregation, the structuredness of the serialization determines further how much load the serialization poses on aggregation and strategy building. Because the effect of structuring is explained in Part 2 of the SPMT, the effect of serialization on strategy building is not included in Part 1 of the SPMT which centers only on the degree of serialization.

We conclude that serialization of the process of process modeling helps reducing the probability of instantaneous cognitive overload if the important condition is met that aggregation of the partial solutions (and modeling strategy building) do not consume the freed resources in working memory.

4.5.2. Part 2: Structured process modeling reduces cognitive overload

As discussed in the previous subsection, the benefits of serializing complex cognitive tasks can only be realized if the accompanying additional cognitive effort for strategy building and aggregation does not surpass the gain of serializing. Observations 2-5 state that different serialization approaches exist. Observation 6 reports that the observed structured approaches (i.e., flow-oriented, aspect-oriented or a combination of these) were faster than the undirected approach. In Impression 2 our perception is expressed that the structured approaches also help to reduce the occurrence of '*mistakes*'. Structuring the process modeling approach seems to increase the effectiveness and efficiency of the construction of process models, which can be explained if these techniques ensure that the cognitive load for strategy building and aggregation is limited. Hence, Part 2 of the SPMT provides the theoretical support for this conclusion (see Figure 4.9).

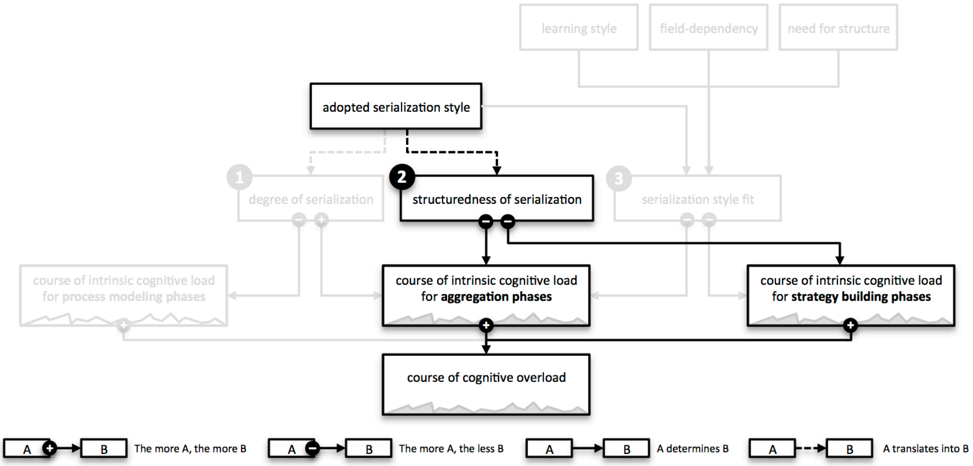


Figure 4.9. The effect of structuredness of the serialization on cognitive overload

A more structured serialization approach towards process modeling makes it easier to keep track of the progress of the modeling endeavor (Van Merriënboer et al., 2003). This in turn lowers the effort to evaluate and adjust the modeling strategy (Van Merriënboer, 1997). By structuring the serialization process, also the outcome of this process (i.e., the process model) will probably be more structured, which facilitates the aggregation of the separately developed parts of the process model (Kim et al., 2000).

Therefore, part 2 of the SPMT states that structuring the (serialized) approach towards process modeling lowers the intrinsic cognitive load for aggregation of the partial solutions and modeling strategy building and thus reduces the probability of instantaneous cognitive overload.

4.5.3. Part 3: Serialization style fit is a prerequisite for cognitive overload reduction

Nevertheless, based on Impression 3, it is proposed that a third factor has to be considered. Besides the degree and structuredness of the serialization, also the fit of the adopted serialization style with the characteristics of the problem solver plays an important role in the cognitive load that the problem imposes on the modeler (Vessey, 1991). This is represented in Figure 4.10.

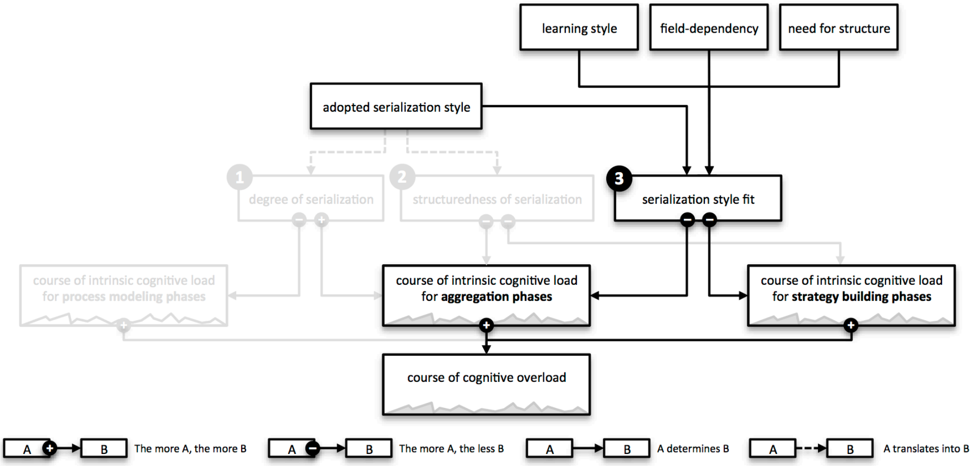


Figure 4.10. The effect of serialization style fit on cognitive overload

Cognitive literature suggests that each human being has a specific intrinsic learning style (Felder & Silverman, 1988). One of the defined dimensions of learning style is called ‘global/sequential understanding’ (Felder & Silverman, 1988). It specifies to what extent a learner needs the material to be processed sequentially. For example, we hypothesize that the flow-oriented modeling method is better suited to a sequential learner, because it builds up the model in a sequential manner. The aspect-oriented approach starts from a global view of the content of the model and drills down into the different details (aspects) of the process model to be constructed. Therefore, it is matched with the global learning style.

Similarly, the Field Dependence-Field Independence Theory (Pithers, 2002; Witkin & Goodenough, 1981) states that some people are better in abstract reasoning than others (i.e., they do not need to load a lot of contextual information in memory). Field dependent modelers find it harder to break up the model in smaller parts that they will construct separately and without considering its context (Pithers, 2002), which means they may prefer the aspect-oriented style for structuring the modeling process, because for each aspect that is targeted sequentially the whole process model is considered before turning to the next aspect.

The Need for Structure scale defines to what extent the performance of a person depends on the structuredness of the adopted solution method (Neuberg & Newsom, 1993). Therefore it is hypothesized that modelers with a high need for structure will benefit most of structuring the modeling process according to the one or the other structuring style.

In summary, based on cognitive theories, we suggest that the load for aggregation of the partial solutions and for modeling strategy building can be kept low if the serialization of the process modeling task is conducted in a structured way that fits with the characteristics of the modeler.

4.6. Summary of the SPMT

The SPMT is summarized below in (i) a theoretical model, which graphically represents the included constructs and their relation; (ii) the propositions, which describe the produced knowledge in a textual format; and (iii) a brief description of the boundaries of the theory.

4.6.1. Theoretical model

As stated before, the intrinsic cognitive load together with the extraneous and germane cognitive load that is needed to solve a certain problem can exceed working memory capacity in which case cognitive overload occurs. When this happens, a negative effect on modeling accuracy and modeling speed results in a decrease of effectiveness and efficiency of the overall modeling endeavor. The SPMT explains how the technique of *individually fitting structured serialized process modeling* can lower the course of intrinsic cognitive load (and thus also the chance of cognitive overload) for a given case complexity and prior knowledge (see Figure 4.11).

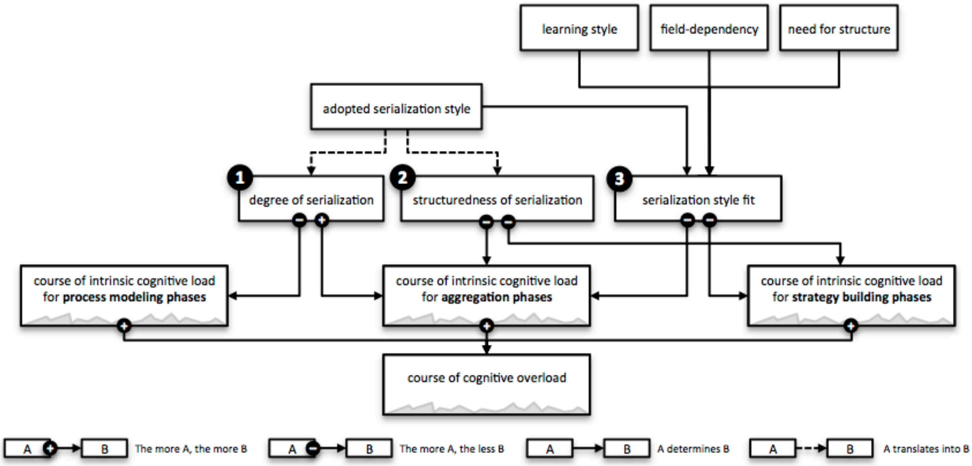


Figure 4.11. Theoretical model of the Structured Process Modeling Theory (SPMT)

4.6.2. Propositions

Based on the three parts of the research model, the research model of the SPMT can be complemented with three propositions.

Proposition 1: Serialization

When the construction process of process models is serialized, the instantaneous intrinsic cognitive load for modeling can be kept lower. If this reduction is greater than the accompanying increase of instantaneous intrinsic cognitive load for aggregating and for strategy building, the total cognitive load is decreased.

Proposition 2: Structured serialization

When the serialized process modeling approach occurs in a structured fashion, the increase in instantaneous intrinsic cognitive load for aggregating and for strategy building can be reduced.

Proposition 3: Individually fitting structured serialization

If the structured serialization approach (e.g., aspect-oriented or flow-oriented process modeling) fits with the characteristics of the modeler (i.e., learning style, need for structure and field-dependency) the increase in instantaneous intrinsic cognitive load for aggregating and for strategy building can be further reduced.

4.6.3. Boundaries

The SPMT was based on observations and impressions in a specific setting. The observed subjects were master students. They served as a proxy for inexperienced modelers. The observed task was the construction of a control flow model in a simplified modeling language. Therefore, the SPMT applies at least for control flow modeling by inexperienced modelers. However, the SPMT is composed of constructs and relations that were found in literature. The only boundary of these existing theories is that they describe cognitive properties, processes or relations of human beings. The SPMT may thus apply to more generic situations.

4.7. Evaluation of the Structured Process Modeling Theory (SPMT)

In this section the six criteria for evaluating an explanatory theory mentioned in Section 4.2.2 are applied to the SPMT: *novelty*, *parsimony*, *consistency*, *plausibility*, *credibility*, and *transferability*. These criteria were found in various academic articles about theory testing (Gregor, 2006; Grover et al., 2008; R. Weber, 2012; Weick, 1989). Nevertheless, we found no concrete guidelines on how to assess the SPMT against these criteria. In this chapter, where the emphasis is on the theory building, logical arguments rather than empirical data are used to evaluate the criteria (Whetten, 1989). The section concludes with a brief discussion of two other important theory testing criteria that we consider currently not feasible to evaluate: *falsifiability* and *utility* (Bacharach, 1989).

Novelty

There are different ways in which a theory can be novel: (i) it describes constructs or associations that were not established before, (ii) it describes well-known constructs or associations in a fundamental new way, (iii) it makes important changes to existing theory (R. Weber, 2012). The SPMT is novel because it combines several existing cognitive theories in a fundamental new way. The consideration of the first part of the SPMT - describing how serialization of the modeling effort helps reducing intrinsic cognitive load - has been touched before (Rockwell & Bajaj, 2005; Soffer et al., 2012). Yet, the idea of structuring the construction process of the process model (i.e., the second part of the SPMT) seems more original, although structuredness of the outcome of such a construction process is well studied (Ralf Laue & Mendling, 2010; Zugal et al., 2013). Also in software there are many studies about the structuredness of program code (e.g., procedural versus object-oriented code (Wiedenbeck & Ramalingam, 1999)).

The real novelty of the SPMT lies in the third part. The technique of serialization is described in cognitive literature as "cognitive sequencing" (De Jong, 2010). Different structured sequencing strategies are defined: e.g., simple-to-complex sequencing, part-whole sequencing (similar to flow-oriented modeling), simplified whole tasks or whole-task sequencing (similar to aspect-oriented modeling), and modular presentation (Gerjets et al., 2004; Van Merriënboer et al., 2003). However, while the notions of cognitive fit where already published in 1986 (Vessey & Weber, 1986), the principle of cognitive fit is not considered in literature when advising which of these sequencing strategies to use. For example whole-task sequencing is considered to always outperform part-whole sequencing (Van Merriënboer et al., 2003) and modular presentation in turn was presented as an improvement of whole-task sequencing (Gerjets et al., 2004). Nevertheless, we propose that cognitive fit should be considered for selecting the appropriate sequencing technique, as is stated in part 3 of the SPMT.

Parsimony

A theory is considered parsimonious if it uses only a small number of constructs and associations to accurately describe their focal phenomena (R. Weber, 2012). Still, a high amount of relevant constructs and associations are presented throughout this chapter. Most of them however are used to describe existing knowledge that constitutes the context of the SPMT. When the three parts of the SPMT themselves are considered, only a small number of constructs and associations is used. The number of constructs and associations of the separate parts and the whole of the SPMT are summarized in Table 4.4.

Table 4.4. Number of constructs and associations in the SPMT

	Part 1	Part 2	Part 3	SPMT
Constructs	5	5	8	11
Associations	5	5	8	15

The artificial distinction between the three types of intrinsic cognitive load (i.e., load for process modeling, for aggregating and for strategy building) and between the two attributes of selected serialization style (i.e., degree and structuredness) could have been omitted. This would reduce the total amount of constructs to 7 and the amount of associations to 8 (i.e., still distinguishing between a positive and a negative effect of adopted serialization style on intrinsic cognitive load). Nevertheless, this would - in our opinion - also significantly diminish the explanatory power and the understandability of the theory.

Consistency

A theory is consistent if various observations can be explained with the same theory. Therefore, other available datasets with recorded data about the process of process modeling were examined for supplementary observations about complexity handling during the modeling activities. Another set of observational modeling sessions in 2013 contained such additional observations. Participants were master students of Business Engineering at Ghent University. They have a similar background and are enrolled in a similar educational program as the students from the exploratory modeling sessions in Eindhoven. 143 additional modeling sessions were recorded. The case to be modeled described a process about collecting fines[29].

A new way of structuring the modeling process was observed in this additional dataset. 12 of the 143 modelers (8%) used a way of structuring that was labeled "happy path first modeling" (see Figure 4.12). They all ended up with a process model without '*mistakes*' that took far less time to construct than the 11 undirected ones in the dataset.

- ***Definition 14: Happy path first process modeling***
  The modelers seemed to have first modeled the main process behavior (i.e., the happy path), and afterwards they modeled an exceptional route. This is called happy path First process modeling.

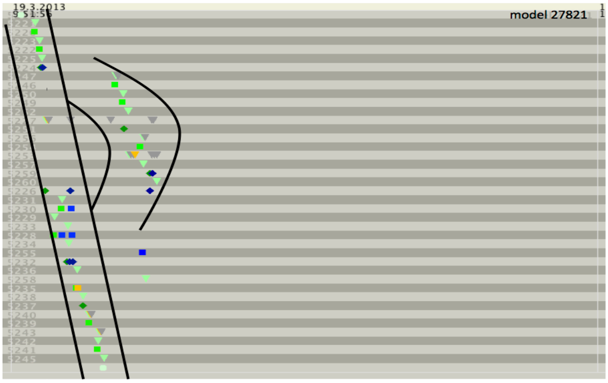


Figure 4.12. Example of happy path first process modeling

---

[29] The case description can be downloaded at http://www.janclaes.info/papers/PPMISF.

The SPMT was developed to describe and explain the observations of Section 4.3.2, but it was also intended to be applicable in a broader sense. As a consequence, it can also be used to explain this additional observation about a previously not discovered way of structuring the modeling process. The modelers clearly serialized the modeling process and used a structured approach (i.e., the happy path first modeling). Potentially, the use of this particular structuring style fitted more to the modeler. For example, a sequential learner that can be classified as field independent, would prefer the flow-oriented approach towards modeling the happy path, but may like to abstract from exceptional behavior at first. According to the SPMT, this can explain why they appeared to have made fewer '*mistakes*' and were faster than the modelers from the "undirected process modeling" subset. Structuring their approach to modeling in a way that fitted with their characteristics has helped them avoid cognitive overload, which has increased their modeling accuracy and speed.

In other words, the SPMT can be used in a consistent way to explain this observation. There was no need to adapt or complement the SPMT in order to be used to explain why the happy path first modeling approach helped these particular modelers. Moreover, a retrospective examination of the modeling sessions in the dataset described in Section 4.3, showed that 15 of the 118 sessions (13%) could have been labeled "happy path first modeling". It was also noticed that all but one of these instances were currently labeled 'uncategorized'.

Plausibility & credibility

The real observed behavior, pronounced in the observations and impressions in Section 4.3.2, was explained based on established theory. Existing cognitive theories were used to provide all the constructs and associations that make up the theory. This theory building methodology warrants both plausibility and credibility. The SPMT is plausible, because it explains accurately and profoundly the effects that were observed in reality. It is also credible, because it uses only constructs and associations from established existing theories to explain those effects.

Transferability

A good theory is transferable to other research contexts. The SPMT was developed as a mid-range theory (R. Weber, 2012) with the observations and impressions in the context of process modeling in mind. Nevertheless, the constructs and associations that constitute the theory were taken from general cognitive literature. Therefore, the SPMT has the potential to be transferred beyond the process modeling domain. It can apply also in other domains such as conceptual modeling in general, programming, text writing, etc., which would make it a macro-level theory (R. Weber, 2012). In order to establish the real rather than the potential theory level and transferability, the theory needs to be applied and tested in various domains, which is addressed as future work in Section 4.9.3.

Falsifiability and utility

According to (Bacharach, 1989) a theory should be evaluated against two other primary criteria: falsifiability and utility. We acknowledge this point of view, but evaluating our theory against these criteria is considered infeasible at this point and therefore out of the scope of this chapter. The evaluation of falsifiability of the SPMT is explicitly addressed as future research in Section 4.9.3, because this requires the propositions of the theory to be operationalized into testable hypotheses. The best way of evaluating the utility of a theory is to measure how much it is actually used for practical and academic purposes, which is off course only possible on a longer term.

4.8. Related Work

Although the constructs of serialization, structuredness and cognitive fit, the three parts of the Structured Process Modeling Theory (SPMT), were not considered together before, they were studied separately in various contexts. In this section, related work is presented that takes a cognitive view on general conceptual modeling or process modeling in particular with a focus on serialization, structuredness or cognitive fit.

Serialization

(Rockwell & Bajaj, 2005) propose the COGEVAL framework that consists of a collection of 8 propositions about modeling complexity and model readability based on cognitive theories. One of the propositions presents chunking as a technique in conceptual modeling to improve modeling effectiveness and efficiency. It is not clear if the term 'chunking' refers to splitting up the model in smaller subparts, or splitting up the modeling process in smaller subparts[30]. If the latter applies, this is similar to part 1 of the SPMT, but without considering the increase in cognitive load for aggregation and strategy building. Next, the process of constructing process models is described by (Soffer et al., 2012) as a sequence of two

---

[30] In literature the term 'chunk' has different meanings: a part of a process, a part of an artifact, a collection of information in memory. Therefore, the term was used sparsely in this chapter. Splitting up a process in parts was named 'serialization' and a collection of information is stored in memory as a 'cognitive schema'.

phases. A modeler first builds a mental model of the process to be represented in the diagram and then the mental model is mapped onto the constructs of a formal process modeling language in order to build the process model. The focus of the paper is on optimizing the formation of the mental model as a prerequisite to increase semantic quality of the process model. It is advised to lower cognitive load by building this mental model chunk by chunk. Furthermore, the paper suggests to examine the impact of model structuredness on domain understanding. Is does not consider however a structured approach towards the chunking.

Structuredness

Most cognition inspired literature on structuredness in conceptual modeling describes a relation between structuredness *of the model* and some other characteristic of the model. A model is considered well-structured if every branch of a split of a certain type is joined in a single join construct of the same type. For example, well-structuredness is proposed to have an impact on correctness because it makes it easier for the modeler to navigate through the model that was build so far, which reduces the chance on introducing errors (Ralf Laue & Mendling, 2010). Further, also nesting depth of split and join constructs is an aspect of structuredness and a greater nesting depth is proposed to imply greater model complexity (Gruhn & Laue, 2006). Finally, (Zugal et al., 2013) describe the effect of hierarchical structuring (i.e., decomposing the model in sub-models) on expressiveness and understandability. It is proposed that hierarchical models suffer from two opposing effects: (i) abstraction decreases mental effort[31] by hiding information and supporting pattern recognition, but (ii) fragmentation increases mental effort because of attention switch and integration effort. The opposing effects of abstraction and fragmentation are described in part 1 of the SPMT. Serialization of the modeling process allows focusing on one part of the model at a time (abstracting from the other parts), but there is a cost of aggregating the different parts (integration effort).

Except for structuredness of the model, there is also literature about structuredness of the input (e.g., a textual case description). (Pinggera, Zugal, Weber, et al., 2010) propose that a breadth-first ordering of text was best suited to yield good results. Breadth-first ordering was defined as *"begins with the start activity and then explains the entire process by taking all branches into account"* (p. 448). It corresponds with the flow-oriented

---

[31] Whereas mental load is defined as the amount of information *needed to store* in working memory at a certain time to perform a task, mental effort can be regarded as the amount of information that is *actually stored* in working memory during the execution of a task.

approach to modeling described in this chapter (whereas depth-first can be matched with the happy path first modeling style). Cognitive fit however, was not considered in their work.

Cognitive fit

In their summarizing framework of cognition variables for conceptual modeling, it is proposed in (Stark & Esswein, 2012) that problem-solving skills of the modeler have to match with the task of modeling and that this (mis)match can cause effects on the resulting conceptual model. Regrettably, this was not further investigated or tested. Further, (Agarwal et al., 2000, 1996a, 1996b) propose that an object-oriented representation is not universally more usable or less usable than other representations. Cognitive fit and prior method knowledge should be considered to evaluate the usability of object-oriented representations. This is fully in line with part 3 of the SPMT, but the focus is not on object-oriented modeling as a process (which would be similar to structured modeling), though it is on object-oriented representations. Therefore the research centers on extraneous load, rather than intrinsic load (as is the case for the SPMT). Lastly, the understandability of a process model is proposed to be more impacted by personal factors, than by model factors (Reijers & Mendling, 2011). This work also recognizes the need for studying cognitive fit, albeit in the context of model reading.

Guidelines for modeling

Most of the work mentioned above describes causal effects between various variables. The emphasis is on predicting, rather than explaining. (Gregor, 2006) states that both theories for explaining and theories for predicting can be used as input for a theory for design and action. The ambition of the SPMT is also to describe the necessary knowledge in order to build a prescriptive theory for process modeling. Two of such prescriptive theories were found already in literature. (Mendling, Reijers, et al., 2010) propose seven process modeling guidelines (7PMG) that are based on strong empirical evidence and are simple enough to be used by practitioners. Guideline 4 proposes to model as structured as possible. The guidelines of modeling (GOM) presented in (Becker et al., 2000) are less concrete guidelines that claim to assure the quality of process models beyond syntactical aspects. Both prescriptive theories, however, provide recommendations about desired process model properties that can be guarded during modeling without considering the cognitive fit of the recommendation with the characteristics of the modeler.

4.9. Discussion

The research described in this chapter is limited in several ways. Nevertheless, the SPMT can be valuable in practice and for research. The limitations and implications of the presented research are discussed below. In order to work on the limitations and to increase its usefulness, future research is described in this section as well.

4.9.1. Limitations

Limited ecological validity

The observations and impressions that were used as input for building the Structured Process Modeling Theory (SPMT) stem from modeling sessions with master students. Furthermore, they were given an artificial case description. In real life modeling sessions the modelers seldom start from a structured case description such as the one that was used for the observations. They rather use direct observation, interview transcripts, notes and pictures from whiteboard sessions, etc. Finally, only syntactic quality was considered when evaluating the produced models. Because the Structure Process Modeling Theory (SPMT) was only inspired by these observations and impressions, but it was compiled from existing cognitive theories that apply widely, there is no reason to suspect that the SPMT does not apply in a more realistic setting. However, the limited ecological validity of the observations and impressions may have hindered the disclosure of all relevant effects of serialization on cognitive load.

Limited content and construct validity

The SPMT and its constructs and associations may have limited content validity. First, only (structured) serialization was investigated (in accordance with the observations), no other general problem solving techniques were considered. Second, the assessment of syntactic quality that formed the base of Impressions 2 and 3, was partly subjective. It is possible that the impressions are not entirely accurate, which may have hindered the disclosure of certain relevant effects of serialization on cognitive load. The credibility of the theory however, is guaranteed by the deductive approach, which builds on existing, established theories. Additional observations in several different settings can help to assess the content validity of the SPMT in the future. Third, although the constructs are clearly described in the SPMT, some of them may be hard to transform into a variable that can be measured properly (i.e., with high construct validity). For example, to date there are no known metrics that measure intrinsic cognitive load separately from extraneous or germane cognitive load, not to mention metrics for the artificially separated constructs of intrinsic cognitive load for modeling, for aggregating and for strategy building of the SPMT.

4.9.2. Implications

Implications for practice

We have experienced that in practice a lot of modelers (experienced and inexperienced) often struggle with the complexity of the case at hand. Although it was observed how some inexperienced modelers automatically turned to a structuring approach and although the structuring techniques are not particularly hard to apply, other modelers do not seem to structure their modeling processes. A slower constructed and lower quality process model was observed, that – according to the SPMT – can be a consequence of applying an undirected process modeling strategy. The SPMT will help building the knowledge that is necessary to (i) be aware of suboptimal modeling conditions (e.g., when modelers apply a structuring technique that does not fit with the task and with their characteristics as a problem solver), (ii) train the modelers to use an *individually fitting structured serialization technique* for process modeling in order to raise effectiveness and efficiency, (iii) provide the means to better support the modelers in handling complexity (e.g., differentiated or adaptive tools that support structured process modeling in accordance to different modeling approaches or with changing features for consecutively modeling phases).

Implications for research

The SPMT is novel in its recognition of cognitive fit between modeling task and modeler characteristics of the proposed modeling structuring technique for optimal effectiveness and efficiency. This fundamental focus point can inspire researchers in other research domains to develop adaptive techniques as well. The SPMT can be applied in a broader context and can add to the existing cognitive theories about serialization as a generic problem solving technique. Furthermore, within the domain of process modeling, the (descriptive) SPMT is considered as a first, necessary step towards the development of a prescriptive theory that will further extend our knowledge about the effect and applicability of structuring and individual fit during process modeling.

4.9.3. Future work

More extensive evaluation of the SPMT

The SPMT needs to be tested more profoundly. The propositions will be converted into empirically testable *hypotheses*, accurate *metrics* need to be developed for each of the involved variables of these hypotheses and new series of *observational modeling sessions* will be performed in which these variables are measured and correlations are calculated.

Because of limited ecological validity of observations and impressions that were used as input for the development of the SPMT, the *external validity* of the SPMT itself needs to be examined further. Current observations were made on master student behavior where the prior knowledge of existing modeling techniques is assumed to be very low. Therefore, one of the factors to examine is how much this prior knowledge of experienced modelers influences the observed effects. Cognitive theories suggest, that retraining an experienced modeler to use a different technique than the ones he is used to, consumes a lot of germane load, which is expressed in an initial decrease of performance (this is called the Expertise Reversal Effect, Kalyuga et al., 2003).

Development of prescriptive theory and a method for cognitive effective and efficient process modeling

Furthermore, in order to convert the SPMT, which is a (descriptive) theory for explaining, towards a (prescriptive) theory for design and action, next actions still need to be undertaken.

First, it should be examined if modelers can be trained to apply the three aspects of the SPMT. This requires the development of a *method* (i.e., prescribing how to construct the process model according to the *individually fitting structured serialized process modeling* principle of the SPMT) and a *treatment* (i.e., describing how to train modelers to use that method). Subsequently, the degree of *treatment adoption* in an experimental context can be measured.

Second, it should be examined if the positive effect on load, overload and by consequence accuracy and speed manifests itself indeed when modelers are trained to apply the developed method based on the three aspects of the SPMT (i.e., *testing causality*). This requires reformulating the hypotheses into *causal* relations between the variables and the set-up of a *controlled comparative experiment* to isolate the effect of the treatment in the measurements of these causal relations.

4.10. Conclusion

In experimental modeling sessions with master students that were instructed to construct a process model based on the same textual description, we noted various differences in the produced process models. For example different syntactical errors were found in the models. Some errors were made consistently and can be caused by a lack of knowledge, whereas other errors seem to be a result of cognitive failures during process modeling. For the development of tools to help modelers to reduce the latter type of errors, knowledge is needed about why, how and when these failures occur and impact the accuracy and speed of the modeling process. This knowledge was not readily available and therefore it is provided in this chapter in the form of an explanatory theory.

The developed theory is called the Structured Process Modeling Theory (SPMT) and consists of three parts. Based on observations and impressions, and on explanations from cognitive literature, it describes how the probability of cognitive overload can be reduced by (i) *serializing* the modeling process, and (ii) *structuring* that serialization (iii) *in a way that fits* with the characteristics of the modeler. The research methodology of theory building based on observations and impressions and using components of existing theories should warrant the utility of the newly developed theory. However, a brief evaluation of the theory and a discussion on the limitations are described in Sections 4.7 and 4.9.

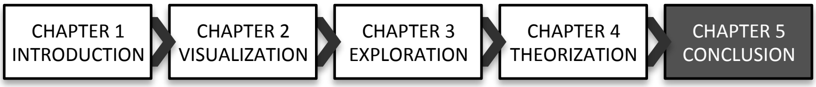
This chapter is important on three levels. Firstly, it provides new *knowledge* on the relation between serializing, structuring and fit of the process modeling approach on the one hand and cognitive effectiveness and efficiency on the other hand. It explains why some modelers struggle (more than others) with the complexity of constructing a process model. This knowledge in itself is useful because it facilitates the selection of suitable modelers or modeling approaches for concrete projects.

Secondly, it is a step towards the development of a *method* that aims at supporting modelers to select and implement an optimized process modeling strategy that fits with the task at hand and with the characteristics of the modeler. If the theory is true, if a modeler can be trained to modify his modeling technique and if this change of approach preserves the described effects, the SPMT has the potential to significantly and positively impact the quality of future process modeling projects.

Lastly, the knowledge and the method can be used to develop *tool support* for process modeling that is differentiated (i.e., the features of the tool can differ according to the use(r) of the tool) or adaptive (i.e., the features of the tool change during the modeling process, for example to support consecutive phases of modeling). Tools can ease a modeler's transition to an improved process modeling technique and can aid the application of such a technique.

5

Conclusion



Summary. In this concluding chapter the research results are summarized. The implications for researchers and for practitioners are discussed. Next, we reflect on our experience with empirical analysis of master student data. Finally, the limitations are presented, together with an outlook on current and future research.

5.1. Research results

This section discusses the contributions of the doctoral research about the Process of Process Modeling (PPM), which is a young research area (see Section 1.3). During the doctoral research, this area evolved and so did our understanding of this research domain. Therefore, we first present our insights in the cohesion of the domain in the form of an overview of four identified knowledge gaps that are addressed by PPM research. Next, the contributions of the three doctoral studies are reviewed and positioned in the PPM research field. In particular, a summary of the doctoral research and the corresponding contributions is provided and it is discussed how the doctoral research has helped shaping the PPM research domain in terms of addressing these four research gaps.

Knowledge gaps addressed by the PPM research domain

Evaluating the current state-of-the-art, four knowledge gaps can be retrospectively defined, which together express the aim to provide the knowledge that is necessary to improve process modeling quality. Below is an explanation of these research gaps, together with an evaluation of the current work addressing each gap (excluding the doctoral research, which is discussed in the next subsection).

- ***Knowledge gap 1**. Lack of knowledge about how people currently construct process models*
  RQs. Which process modeling patterns exist? When are they applied? By who are they applied?

The first identified knowledge gap builds on the observation that different approaches were used to tackle the same process modeling assignment (see chapter 2). Therefore, it is interesting to investigate how process models are typically constructed. A considerable effort was made to develop metrics and data collection and analysis tools, which help answering the research questions addressing this gap (Pinggera et al., 2014; Pinggera, Zugal, & Weber, 2010; Pinggera, Furtner, et al., 2013; Sedrakyan et al., 2014; Soffer et al., 2012; Zugal et al., 2012; Zugal & Pinggera, 2014). Moreover, several observed process modeling patterns are described by different studies (Pinggera et al., 2014; Pinggera, Soffer, et al., 2012, 2013; Pinggera, Zugal, et al., 2012; Recker et al., 2012; Sedrakyan et al., 2014), but we do not consider them to have provided a complete answer to the research questions mentioned above.

- ***Knowledge gap 2**. Lack of knowledge about the intrinsic quality of modeling approaches*
  RQs. How much time does it take to apply the identified process modeling patterns? How much effort does it take to apply them? How good is the resulting process model when they are applied?

It was also observed how the quality of the produced models and the efficiency of modeling differ between modeling executions for the same modeling assignment (see chapter 3). There is a lack of knowledge about how the applied modeling patterns intrinsically relate to the effectiveness and efficiency of modeling. Initial insights exist about how certain process modeling patterns relate to process model quality (Francescomarino et al., 2014; Kolb et al., 2014; Recker et al., 2012; B. Weber et al., 2013, 2014), but we believe that more research is needed.

- ***Knowledge gap 3**. Lack of knowledge about how people should construct process models*
  RQs. How to know which process modeling pattern(s) to apply in a specific context? When (not) to apply each pattern? Why (not) to apply each pattern?

Besides knowledge about which patterns perform intrinsically better than others, observations suggest that the context may also determine how much the advantages and disadvantages of each pattern influence the modeling process (see Chapter 4). Therefore, in order to know which process modeling approach should be applied in a specific context, research is needed to investigate the impact of the context on the efficacy of using process modeling patterns. Currently the relevant context is mainly defined in terms of task-specific characteristics (such as task complexity and task representation) and modeler-specific characteristics (such as modeling expertise, domain knowledge and cognitive characteristics of the modeler) (Pinggera et al., 2014). Also, initial attempts are made to study the effect on process modeling of using change patterns in the PPM. It was concluded that the current implementations are most successful for modeling sessions of low task-complexity (B. Weber et al., 2013, 2014).

- ***Knowledge gap 4**. Lack of knowledge about how people should change their modeling approach*
  RQs. How to apply each process modeling pattern? How to teach each process modeling pattern? When is learning of a new process modeling pattern hindered?

Furthermore, in our current (unpublished) work (see Section 5.4), we observed that additional research will also be needed to determine how (best) to teach optimal modeling approaches to the modelers. There is currently no research in the PPM domain that investigates how optimally to change one's modeling behavior, although it is known that this may be challenging. Cognitive psychology for example suggests that when people are used to a certain way of working, at first they will not improve when they learn any new technique. Applying a 'better' technique will only produce better results on a longer term. This initial decrease is known as the Expertise Reversal Effect (Kalyuga et al., 2003; Kalyuga, 2007).

Figure 5.1 presents the current PPM literature in relation to the four identified knowledge gaps. In the center, the four gaps are displayed, while the PPM research articles in this overview are positioned around and related to these research gaps.

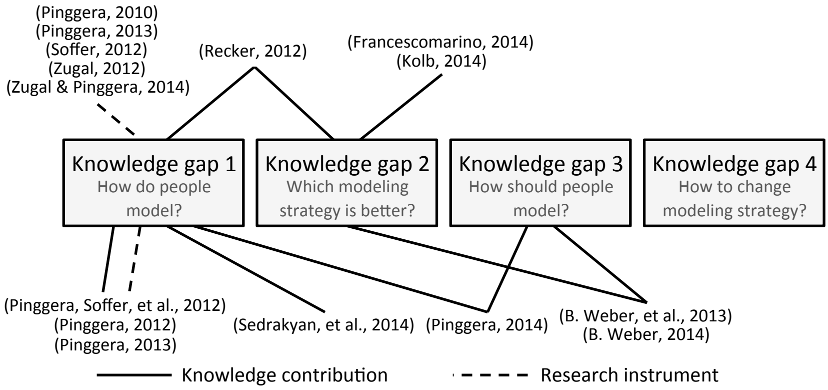


Figure 5.1. PPM literature in relation to the identified knowledge gaps

In summary, when the four mentioned research gaps are sufficiently reduced by future research in the PPM domain, scientific evidence will have been produced that supports the development of a successful concrete process modeling method. The method will be successful because it builds upon knowledge about existing process modeling approaches, their quality, when to use which approach and how to learn the appropriate approach. The method will be concrete, because this knowledge about process modeling approaches is formulated in terms of concrete and observable process modeling patterns and thus the steps of the method can also be formulated in this concrete terminology. This will help solve a relevant practical problem, because currently a lot of quality problems are reported about the used process models in organizations (Mendling et al., 2008), while at the same time the use of process models has increased radically during the last years (Dumas et al., 2005; Kock et al., 2009).

Research evaluation

This subsection discusses - study per study - the contributions of the doctoral research and relates each contribution to the identified knowledge gaps presented in the previous subsection.

Study 1. Visualization

The aim of the first study was to develop a cognitive effective visualization that presents detailed information about the process of process modeling, such that it supports the quick and easy discovery of knowledge about how exactly people construct process models. The developed visualization – the PPMChart – orderly displays the raw data that was collected by the modeling framework, i.e., every modeling operation of the modeler in the tool. The visualization is based on the Dotted Chart technique (introduced in Section 2.2.1), but was optimized for PPM research and for cognitive effectiveness.

Cognitive effectiveness was assessed theoretically against nine visualization principles from cognitive literature and was evaluated with a qualitative analysis (see Section 2.2.2). The visualization was perceived by six academic researchers (i.e., potential users of the visualization) as a useful tool and it was shown to be more cognitive effective than the Dotted Chart technique (i.e., observation generation was faster and more detailed). Therefore, the PPMChart, being a cognitive effective research instrument, contributes to research addressing Knowledge gap 1. It displays more detail than the Modeling Phase Diagrams (introduced in Section 1.2) and it is more cognitive effective than the default Dotted Chart technique.

Furthermore, the visualization was used to study 357 modeling executions, which has led to the documentation of twenty-two process modeling patterns. In addition, thirteen observations were made regarding differences in modeling time, in the amount of pauses, in the number of created elements and in the number of modeling operations. It was also observed that differences exist between how and when modelers delete elements (i.e., never, occasionally, in dedicated phases) and how and when modelers moved elements on the canvas (i.e., never, only shortly after creation, in a dedicated phase at the end of modeling). Next, the charts revealed that a number of modelers first create only certain kinds of elements, such as activities and gateways, while postponing the creation of other elements, such as the arrows connecting the activities and gateways. Further, whereas some modelers create gateways in pairs (i.e., they create the matching join gateway right after creation of a split gateway), other modelers consistently work in blocks (i.e., they create a split gateway, afterwards all paths of the model block, and finally the join gateway that closes the block). These patterns of process modeling behavior were discovered over different process modeling executions of the same modeler, and over different modelers that were modeling the same process.

Although the observations concern a specific set of modelers (i.e., 237 master students spread over two different European universities) and no quantitative analysis was performed to assess how many times each pattern was observed in the dataset, they provide preliminary insights in how people create process models. Hence, the study also provides a knowledge contribution that addresses Knowledge gap 1.

Study 2. Exploration

The aim of the second study was to explore how the discovered modeling patterns from study 1 relate to the quality of the produced process model. The comparison of the PPMCharts with the corresponding process models of 40 modeling executions uncovered three interesting conjectures about this relation. First, it was observed how certain modelers built the model block by block and these 'structured' modeling executions seemed to have resulted in better process models. Second, the modelers that moved a lot of process elements on the canvas during modeling tended to produce worse process models. Third, slow modeling was associated with process models that have low quality.

The conjectures were then statistically tested on a dataset from 103 process modeling executions. For each of the three conjectures convincing statistical support was found in the dataset. We conclude that this study can thus be regarded as contributing to address Knowledge gap 2.

Furthermore, to be able to perform statistical tests on the data, the conjectures had to be operationalized in measurable statements. We defined the notion of a process model block, which is a part of a model containing all the elements involved in two or more optional or parallel paths in the process model (see Definition 8). Structured modeling was defined in this study as a way of modeling where each block was finished first before working on another block. This way, the description of the 8 patterns about structuredness, movement and speed were refined, which extends the contribution of the previous study related to Knowledge gap 1.

Next, process model quality was assessed with the perspicuity metric, which we defined. This metric is a binary metric that indicates if a model contains no cognitive syntax errors, which are errors against the modeling notation syntax that have a cognitive origin, rather than being caused by a lack of knowledge of the modeling language (see Definition 9). The definition of this concept helps assessing the effectiveness of process modeling approaches and thus provides a research instrument to address Knowledge gap 2.

Study 3. Theorization

The aim of the third study was to search for a deeper understanding and an explanation of the first conjecture of study 2, i.e., the proposed link between structured process modeling and improved model quality. First, new insights from 118 modeling executions were used to elaborate the observations about structured process modeling from the previous studies.

Four different ways of process modeling related to structuredness were discovered and documented as process modeling styles. Flow-oriented process modeling is when modelers create the model in consecutive parts (e.g., in process model blocks). Once a part is considered complete, the modeler does not modify it anymore (see Definition 12). Aspect-oriented process modeling is when the modeler creates the process models in multiple iterations, each focusing on another aspect of the model. For instance, they first create all events and activities (i.e., content); then they add gateways and arrows (i.e., structure); and finally they complete the model by working on its lay-out (see Definition 13). Further, Combined process modeling is when both styles are combined. Finally, Undirected process modeling is defined as a chaotic pattern of modeling operations with no recognizable structure. These identified process modeling styles further extend the contribution to address Knowledge gap 1.

Next, six observations and three impressions were listed, which describe how much each style was observed, that the undirected style was associated with a longer modeling time, and that the other styles were mostly matched to process models of higher quality, but not in every case. This knowledge addresses Knowledge gaps 1 and 2.

We then developed the Structured Process Modeling Theory (SPMT), which offers an explanation for these observations. The theory is based on Cognitive Load Theory and Cognitive Fit Theory. The former theory describes how human's working memory capacity is limited and that when too much information needs to be processed, the working memory gets overloaded, which causes a decrease in effectiveness, efficiency, and learning. Therefore, it was investigated when cognitive overload occurs during modeling. The SPMT explains that overload can be associated with the amount of information to be stored simultaneously in memory during modeling. This amount can be kept low if people serialize their modeling process (i.e., divide the modeling task in sub-tasks that are executed sequentially rather than simultaneously, see Definition 10). Moreover, because the integration of the results of the different sub-tasks may occupy the freed resources in working memory, it is important that the serialization happens in a structured way (see Definition 11) and that this structured way fits with the cognitive preferences of the modeler (see Chapter 4).

The theory was evaluated through the assessment of its essential characteristics, i.e., novelty, parsimony, consistency, plausibility, credibility, and transferability (Gregor, 2006; Grover et al., 2008; R. Weber, 2012; Weick, 1989). The consistency of the theory was demonstrated by ascertaining that additional observations from a dataset with 143 modeling executions could be explained with the proposed SPMT. The SPMT, being an explanatory theory about why people create sub-optimal process models, contributes to the knowledge requested by Knowledge gaps 2 and 3. It defines the relation between modeling approach and model quality (cf. gap 2) and explains when mistakes are avoided (cf. gap 3).

An overview of the research contributions in relation to the four research gaps is presented in Figure 5.2.

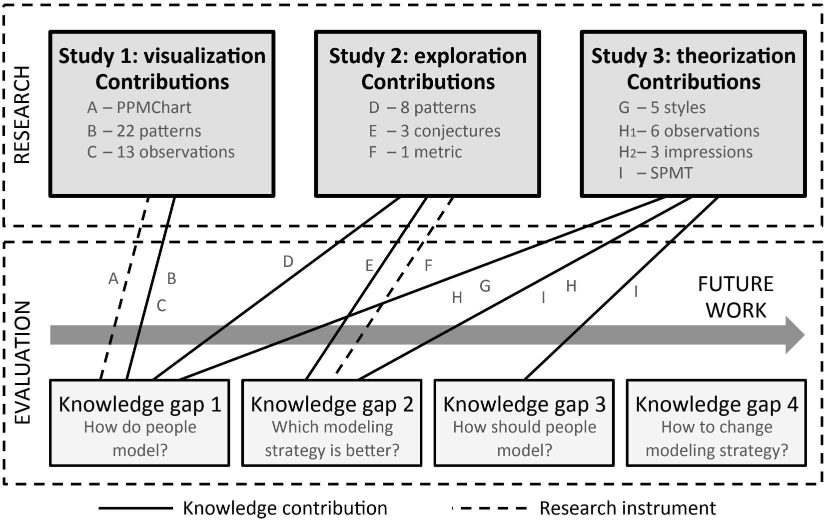


Figure 5.2. Contributions of the research in relation to the identified knowledge gaps

5.2. Implications

The results of the doctoral research have implications for researchers and practitioners. Researchers can use the developed research instruments and the developed knowledge to support their research. The contributions are also useful for practitioners to take into account when modeling, recruiting modelers, training modelers or developing tool support for modelers.

Implications for researchers

First, the research produced two concrete research instruments, i.e., the PPMChart and the perspicuity metric.

PPMChart. The PPMChart is currently the only PPM visualization that shows extensive detailed data of the PPM and yet is still considered cognitive effective. Moreover, it comes with an open source implementation that provides wide-ranging configuration and filtering options (see Appendix B). Since its original development presented in Chapter 2, features have been added, such as bulk generation of PPMCharts from an event log containing multiple process executions, or model chunk detection based on sudden differences in modeling speed, in region of modeling activity in the model, and in amount of model elements under construction. This makes the PPMChart visualization an advanced research instrument that is useful for PPM researchers. It has already proven its utility in the subsequent studies of this doctoral thesis, which were both supported by this visualization.

Perspicuity metric. In Study 2 the perspicuity metric is defined as *"a model that is unambiguously interpretable and can be made 'sound' with only small adaptations based on minimal assumptions on the modeler's intentions with the model"* (see Definition 9). Although certain design decisions in the calculation of the metric still need to be examined further (e.g., decision to not tolerate combined gateways), this metric can be useful for researchers that want to assess unambiguity of models that will be used by humans (who are capable to effortlessly ignore the disregarded small syntax errors). For example, when syntactic correctness is related to understandability of a model, it makes sense to use the perspicuity metric instead of the strict soundness to measure syntactic correctness.

Second, the developed knowledge can be used by researchers to support their work. This knowledge can be classified in knowledge about modeling patterns, knowledge about the relation with process modeling quality, and explanatory knowledge that describes why mistakes are made during modeling and how they can be reduced.

Process modeling patterns. Various process modeling patterns were described in the studies of this doctoral dissertation. These patterns are similar to or perhaps have inspired discovered patterns by other researchers. More specifically, Pinggera, et al. describe three modeling patterns, i.e., Planning (initial delay before modeling in the tool), Detours (occurrence of rework in the form of deletion and re-creation of elements), and Layout behavior (timing of move operations), and three modeling styles, i.e., slow modeling with more reconciliation, faster modeling with less reconciliation, and slow modeling with less reconciliation (Pinggera et al., 2014; Pinggera, Soffer, et al., 2013). Further, the knowledge about how modelers construct process models (i.e., the identified patterns) is useful to ground the development of broader PPM knowledge and to build advanced tools. For example, the development of the SPMT was based on observations about the occurrence of the identified modeling styles. Also, research has shown that the applied modeling approach is influenced by task-specific and by modeler-specific characteristics (Pinggera et al., 2014; Pinggera, Soffer, et al., 2013).

Link with process model quality. The second study revealed a link between structuredness, movement, and speed of modeling on the one hand and process model quality on the other hand. The SPMT links degree, structuredness and fit of serialization of the modeling process to process model quality and efficiency of modeling. This knowledge can be used to elaborate existing knowledge about process model quality and potential causes for quality issues.

Explanatory knowledge. The SPMT provides an explanation about why certain modeling styles may result in better process model quality than others. As such, it can be used to explain the outcomes of existing studies. For example, it is observed that structuring domain knowledge helps casual process modelers in creating more accurate process models (Pinggera, Zugal, Weber, et al., 2010). If the assumption is correct that structuring the domain knowledge promotes a more structured approach towards modeling, the SPMT explains why more accurate models are constructed. Furthermore, based on the explanatory knowledge of the SPMT, other types of theories can be developed, such as predictive theories or prescriptive theories. In order to transform the SPMT into a predictive theory, the relative weight of each causal relation needs to be better understood. A predictive theory can be used to predict (more accurately) already during modeling certain properties of the process models that is being constructed, such as syntactic and semantic correctness. The SPMT can also be used to develop a prescriptive theory that provides concrete guidelines about how to approach process modeling in order to produce a process model of higher quality or to need less time and effort for modeling. The development of such a prescriptive theory addresses Knowledge gap 3 and is described as ongoing and future research in Section 5.4.

Implications for practitioners

The practitioners that benefit from the doctoral research are process modeling lecturers, process modeling tool developers, process modeler recruiters, and process model users (e.g., business analysts).

Process modeling lecturers. Although the produced knowledge is limited and not much is known about the external validity of the results, it contains initial insights that can be used by people who have to train novice modelers. Moreover, the PPMChart and SPMT can be used to assess a modeler's current process modeling capabilities, which is useful for evaluation of the modeler or for the development of a targeted training program. Further, the doctoral research forms the basis for the development of a process modeling method that can be trained to novice modelers in order to improve modeling efficacy (see also Section 5.4).

Modeling tool developers. The knowledge about how people construct process models and which modeling approaches are associated with promising results can be used by tool developers to build modeling tools that are adaptive and differentiated. An adaptive tool is a tool that changes the offered features depending on the situation. For a modeler who applies an aspect-oriented process modeling style, and by consequence works in consecutive modeling phases each targeting another aspect, such a tool can be helpful. It can offer different support for each phase. Differentiated tools are tools that change the offered features depending on the user. Global learners, for example, may prefer to see at all times an overview of their model so far. Modelers with a high need for structure may desire specific structuring support.

Process modeler recruiters. An increasing number of organizations include extensive practical and psychological tests in their job application procedures. It can be imagined that such companies would like to use the PPMChart to visualize how the candidate approaches a modeling task, and the SPMT to assess candidate modelers in terms of their modeling capabilities. Furthermore, the SPMT may help during modeling already to estimate to some extent the quality of the result. Unstructured modeling,

modeling with many move operations and slow modeling are associated with less understandable results. A low degree of serialization, a low structuredness of serialization, and a low fit of the serialization style cause higher cognitive load, and thus also increases the probability of overload, which would result in a decrease of process modeling effectiveness and efficiency. This makes it easier for recruiters to assess the modeler already in the early stages of a modeling assignment.

Process modelers. Process modelers can use the PPMChart and SPMT as self-assessment tools that can help shed light on their current modeling capabilities and reveal potential future improvement points. They will obviously also benefit from the improved training programs and modeling tools, and from developed process modeling methods (see also Section 5.4).

5.3. Reflections

In this section at the end of the dissertation, we reflect on some research design aspects relevant to the doctoral research.

On the focus on the specific domain of process modeling instead of on the more general domain of problem solving

In the doctoral research a combination of deduction, induction and abduction (see Table 5.1) was used to generate the Structured Process Modeling Theory (SPMT).

Table 5.1. Comparison between deduction, induction and abduction

Concept	Logical reasoning					Reveals	Result
Deduction	Rule	+	Case	→	Result	Effect	Certainty
Induction	Case	+	Result	→	Rule	Mechanism	Probability
Abduction	Result	→	Rule	←	Case	Cause	Possibility

Deduction means that one uses a general rule and applies it to a specific case, which allows for logical derivation of the result. For example, we used theories form Cognitive Psychology to generate the relations in the SPMT from degree, structuredness, and fit of process modeling serialization towards cognitive load, overload, and modeling quality. The 'rules' are specified in the cognitive theories, the 'case' is the applied process modeling approach, and the derived 'result' is the effect on load and overload of a modeler, and finally also on the resulting modeling quality.

Induction means that one uses a certain specific case and observes a certain result, which allows for logical derivation of a proposed rule (i.e., an hypothesis). For example we instructed students to create models from a given description, collected data about how they worked and performed, and used these data to formulate observations and impressions. The 'case' is the applied process modeling approach. The 'result' is the produced process model. The derived 'rules' are formulated in the identified observations and impressions.

Abduction means that one uses results for which one invents possible causes formulated by rules, which then are evaluated with specific cases. For example we observed a difference in process model quality of modelers that all applied the same structured modeling approach and we suggested that structuring does not help every modeler to the same extent (see Impression 3). We proposed three factors to explain why sometimes an approach, even if it is structured, does not fit with a modeler (i.e., learning style, field dependency, and need for structure). The 'result' is our observation expressed in Impression 3. From cognitive theories we derived that one of the reasons for the observed differences may be a misfit of the structuring approach with the modeler's cognitive characteristics (i.e., deduction). Subsequently, the 'rules' obtained through abductive reasoning are formed by the specific cognitive causes for (mis)fit that we proposed. The collected data about the modelers and their applied process modeling approach (i.e., the 'case') were then used to evaluate if these possible explanations should be altered, which was not the case. Currently we have no data that suggests there are other relevant causes for (mis)fit of the structuring approach than the ones formulated in the SPMT.

As a result, the SPMT was constructed with a combination of these three logical reasoning techniques. Each concept and relation of the SPMT originated in inductive or abductive reasoning starting from the observational data. But the formalization of these concepts and relations was performed based on deductive reasoning, which limits the risk of errors in the theory to the misinterpretation or wrong application of the rules expressed by the existing theories that were used for the deduction.

Further, a big difference can be noticed between the generality of the cognitive psychology domain and the specificity of the process modeling domain for which we developed knowledge in the form of the SPMT. Why did we not target the more general conceptual modeling or problem solving domain for the knowledge development? In hindsight we believe that the inductive part of the research, which was the driving force behind the whole project so far, was facilitated by the selection of the particular domain of process modeling.

This can be explained as follows. The inductive part of the research involved visually recognizing patterns in the collected data. By recording data from a domain where it is obvious how to sort the data (i.e., the lines in the PPMCharts are sorted according to the order of the corresponding elements in the produced process model), pattern recognition became easier. Indeed, think about how it is less evident to determine a desired order of the construction actions of an arbitrary conceptual model such as an ER model. Even for programming, where there is certainly a notion of order at run time, it does not seem easy to specify the logical order of producing the code at design time. When the data in the visualization is not ordered logically, it is much harder to detect patterns (see for example Figure 2.1b and our observations about sort order described near the end of Section 2.5.6).

Thus, on the one hand, the focus on the particular application of process modeling has facilitated knowledge building because of the eased pattern detection. On the other hand, only trusting on the deductive aspects of the research, it can be concluded that the SPMT should apply at the more generic level of problem solving. Hence, we plan in future work to investigate if this is indeed the case (see Section 5.4).

On empirical research with student observation data

As can be derived from the overview of data collection sessions in Appendix H, the vast majority of data that was collected at different phases in the doctoral research concerns observations of master students while creating one or more process models. Although we were aware of the many threats of drawing conclusions based on student behavior, we deliberately chose to study this particular group for more than only practical reasons.

The use of student observations as input for theory development may limit the external validity of the theory. This argument has been used frequently to object to research based on student data (Moody, 2005). However, we consider our work to be crucially different from a lot of the related research, because we study the cognitive flaws of a modeler during modeling, rather than targeting knowledge-related issues. We have no reason to assume that a master student uses its cognitive functions fundamentally different than an experienced modeler. Whereas the point of cognitive overload definitely differs between students and modeling experts, the same causes and consequences apply when cognitive overload happens.

Therefore and in contrast to other studies, we rather consider the lack of prior knowledge and expertise of students as a unique advantage for our work. When modelers have built considerable expertise in using a certain modeling language or tool, they tend to first perform worse when confronted with another language or tool. This is known as the Expertise Reversal Effect (Kalyuga, 2007). In contrast, the students were less influenced by modeling traditions at the time of the observations; which were strategically planned after they have learned about the modeling language, but before they learned any modeling technique. Therefore, the simplistic modeling notation and editor that were used in the data collection sessions did not hinder them as much as it would be the case for modeling experts.

The advantages of observing master students in the context of this doctoral research can be summarized as: they are representative participants (i.e., human beings with mature cognitive reasoning skills), they do not suffer from the Expertise Reversal Effect, they are a homogeneous group (which is beneficial for between–subjects comparisons), they still provided a heterogeneous set of observations (because of differences in cognitive strategies), and the point of overload is easier to reach than with modeling experts.

On empirical research in Business Process Modeling

Process modeling is a complex problem-solving task. There are a lot of factors that influence the quality of the process and product of modeling. A distinction can be made between personal factors (e.g., cognitive capabilities, prior knowledge and cognitive preferences) and task-specific factors (e.g., case complexity, tool support and situational factors). Moreover, a high number of these variables are not constant during the modeling process (e.g., cognitive load, mental image complexity, attention). This makes it extremely hard to identify, measure and control the confounding variables during empirical process modeling analyses. Some researchers ignore this problem, assume that these factors can be kept constant during experimental exercises, or assume that their influence can be minimized. This poses a considerable threat to the validity of the results.

In our research however, we tried not to ignore the confounding variables and explicitly include them in the analyses. Also, we operated under the open-world assumption (Reiter, 1978). Under this assumption no statement can be made about information that is not known, other than that it is not known. The assumption implies that conclusions have to be considered incomplete, unless it is proven that they are complete.

Conversely, the closed-world assumption presumes that if one attempts but fails to observe a particular relation, it can be concluded that the relation (probably) does not exist (Motro & Smets, 1996). As a consequence of not drawing conclusions from inconclusive results, research progress tends to be more accurate, but slower under the open-world assumption (Moore & Pham, 2015). This supports the decision to leave the development of the process modeling method out of the scope of this dissertation.

Another strategy that we applied to deal with the difficulties of empirical research in the multifaceted context of process modeling is to elaborate the more quantitative approach with depth-oriented research activities that try to look for explanations for the observed behavior. During post-interviews and based on observations of individual modeling process replays, we gained insights in the underlying cognitive mechanisms, which helped us to identify confounding variables extending the research model of the SPMT. This way, our work could rise beyond the explorative level. In future work we will continue this trend through the development of a differentiated process modeling method that incorporates the complex knowledge comprised in the SPMT (see Section 5.4).

On the nature of interdisciplinary research

The interdisciplinary research described in this dissertation builds on and contributes to different research disciplines (see Figure 5.3). In Section 1.2, we positioned the research in the domain of Business Process Modeling, which is on the crossroad of the Business Process Management domain and the Conceptual Modeling domain. We have contributed to the existing Process of Process Modeling (PPM) research by developing two research artifacts (i.e., the PPMChart and the SPMT) and the related knowledge (i.e., pattern definitions, observations, and conjectures).

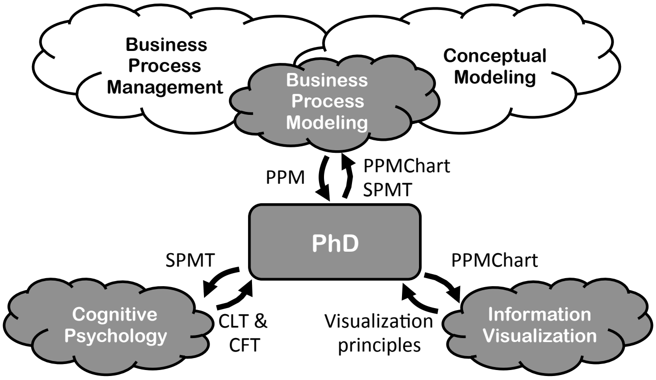


Figure 5.3. Research domains in relation to the PhD research

Further, the doctoral research was nurtured by the research disciplines of Cognitive Psychology and Information Visualization. Concerning Cognitive Psychology, the Cognitive Load Theory (CLT) and Cognitive Fit Theory (CFT) formed the basis for the development of the Structured Process Modeling Theory (SPMT). In turn, the theory can be used to explain human behavior through cognitive mechanisms. Similarly, the nine Information Visualization principles presented in Section 2.2.2 were used to develop the PPMChart, which in turn can be used to visualize information about the PPM.

Performing interdisciplinary research is challenging. First, the researcher needs to gain expertise concerning the current knowledge and research methods in each discipline (Golde & Gallagher, 1999). Then, he needs to resolve conflicts and to find a balance between these domains and methods. He has to construct and explain a new viewpoint and to adapt existing methods to this interdisciplinary viewpoint (Golde & Gallagher, 1999). Such a novel viewpoint is illustrated by the reflections in this section.

For instance, while Cognitive Psychology is considered to be a reference discipline, Business Process Modeling is traditionally considered to be an applied discipline. It builds on other, more established research disciplines (Baskerville & Myers, 2002). Therefore, Cognitive Psychology has a tradition of inductive, observation-driven research whereas Business Process Management comprises relatively more deductive research (i.e., deducing knowledge from the reference disciplines). As discussed in the first reflection of this section, we have combined the deductive and inductive approach in the research in order to try to cumulate their benefits. We consider the deductive side of the research to warrant the external validity and the inductive part introduces the depth of the observations to the resulting theory development.

Furthermore, in the Business Process Modeling field a difference between study subjects is mostly expressed in terms of their varying levels of prior knowledge and expertise (i.e., novices versus experts). Conversely, in the field of Cognitive Psychology a distinction is often made according to the cognitive preferences and capabilities of human beings. As discussed in the second reflection of this section, we have chosen to follow the latter view for the inductive part of our research. The focus on students as cognitive mature (enough) observation subjects has the benefits of studying a homogeneous group with similar skills and expertise, but yet different cognitive capabilities. Compared to experts, the point of cognitive overload is easier to reach and they do not suffer from the Expertise Reversal Effect.

Next, typical data-driven research of Business Process Modeling and Cognitive Psychology tries to isolate the mutual effects of a very limited amount of constructs by abstracting from other relevant factors. In contrast, the goal of the Information Visualization field is to provide ways to effectively deal with large amounts of interrelated data. Therefore, as discussed in the third reflection of this section, we tried to include a relatively large amount of variables in our model and deal with the accompanying uncertainty and complexity by subscribing to the open-world assumption.

5.4. Limitations and future work

In this section the limitations of the research are discussed, followed by an outlook on ongoing and future research, which target certain limitations of the current work.

Limitations of the research

The knowledge produced by the doctoral research includes the description of modeling patterns and the relation of these with cognitive preferences and model quality of the constructed process model. This knowledge was derived from observations (i.e., mainly inductively) and theory (i.e., deductively). The observations clearly have limited ecological validity as they describe students creating a process model from a given case description. In practice, modelers usually have more prior domain knowledge and modeling expertise. Also, they rarely start from a structured case description such as the ones used in the data collection sessions. This could limit external validity, which was countered by the triangulation via the deductive research (as explained in Section 5.3). Relying on these observations however, may still have restricted the disclosure of all relevant variables.

Furthermore, the identified variables of the SPMT are defined conceptually, but no validated measurements exist for all of them. This limits the current potential use of the produced knowledge. Nevertheless, research has started to overcome this limitation. Therefore, this conclusion ends with an outlook on current and future work that targets these shortcomings, but that is not included in the scope of this dissertation.

Measure development

As described before, the general aim of PPM research is to develop knowledge that is necessary to develop an effective process modeling method. The doctoral research contributes to this knowledge mainly via the Structured Process Modeling Theory (SPMT). Nevertheless several studies still have to be completed before such a method can be developed based on the SPMT. First, the SPMT needs to be operationalized in measurable variables and relations in order to discover knowledge about the relative importance of each proposed relation in the theory. For this analysis, accurate measurements need to be developed to quantify the concepts of the SPMT (see Figure 4.11).

The three cognitive preference variables of the SPMT, i.e., learning style, field-dependency, and need for structure, have validated metrics associated with them, which were found in cognitive literature.

The measurements of the degree, structuredness, and fit of the modeling serialization are currently under development. These concepts are split into sub-dimensions that measure certain aspects such as consistency or overlap of modeling operations, and for which we defined initial indicative measurements.

The measurement of cognitive load is challenging. Self-rate scales are believed to have a high validity (Paas, Tuovinen, et al., 2003), but they can only be used to measure overall cognitive load, while we are interested in the load at a certain time during modeling. Dual task measures can be used to have more fine-grained measures of cognitive load, but they may influence the modeling process, because they require the user to react on external signals during modeling (the reaction time or quality indicates the amount of load at a certain time) (Paas, Tuovinen, et al., 2003).

For measuring syntactic and semantic quality of the resulting process model, we are developing metrics that try to distinguish between knowledge-related and cognition-related errors in the process model.

Initial results revealed interesting new insights, which we are currently investigating. For example, apparently many modelers do not apply a matching modeling style automatically, which raises expectations for the potential gain of a good process modeling method.

Prescriptive theory

Next, the developed measurements can help expand our knowledge. The SPMT has to be transformed from a pure explanatory theory towards a prescriptive theory (i.e., addressing Knowledge gap 3). When the relative importance of each relation in the theoretical model of the SPMT is better understood, concrete guidelines can be developed, which will form a prescriptive theory. Currently, knowledge is lacking about the relative importance between the degree, structuredness and fit of serialization, but also about the relative importance of learning style, field-dependency and need for structure in the cognitive fit of the applied serialization technique. Inspired by specialized cognitive literature and by insights obtained with the new measurements, we are currently working on this step. However, initial results are not conclusive.

Modeling method

Finally, in future work a differentiated process modeling method will be developed and evaluated, i.e., the Structured Process Modeling Method (SPMM). First, the cognitive profile of a modeler will be determined with the developed measurements. Second, the prescriptive theory will derive the best-suited modeling style for the modeler. Third, a training instrument has to be developed, which will instruct the modeler how to construct the process model (cf. Knowledge gap 4).

Generalization

After the development of the SPMM, we intend to investigate if the SPMT and the SPMM can be generalized towards conceptual modeling in general or even towards the problem solving domain. For example, similar to flow-oriented and aspect-oriented process modeling, one can produce a large text in a flow-oriented or aspect-oriented manner. Flow-oriented text writing can be described as producing text chapter by chapter, where the writer immediately works on all aspects of the chapter such as the content, structure and formatting. Only when a chapter is considered complete, the writer continues with the next chapter. Other people may produce better results when working in an aspect-oriented way. They can first think about the structure of the text, then write the whole text, next format the document, add illustrations, etc. Similar analogies can be drawn with drawing (e.g., completing a picture of a face part by part or first sketching the main lines and gradually adding details, colors, shadows, etc.), programming (completing class per class or first preparing the class bodies, next adding method and variable definitions, then writing the program code, providing code documentation, etc.), and so on.

References

Abecker, A., Bernardi, A., & Hinkelmann, K. (2000). Context-aware, proactive delivery of task-specific information: The KnowMore project. *Information Systems Frontiers*, *2*(3-4), 253–276.

Agarwal, R., De, P., Sinha, A. P., & Tanniru, M. (2000). On the usability of OO representations. *Communications of the ACM*, *43*(10), 83–89.

Agarwal, R., Sinha, A. P., & Tanniru, M. (1996a). Cognitive fit in requirements modeling: A study of object and process methodologies. *Journal of Management Information Systems*, *13*(2), 137–162.

Agarwal, R., Sinha, A. P., & Tanniru, M. (1996b). The role of prior experience and task characteristics in object-oriented modeling: An empirical study. *International Journal of Human-Computer Studies*, *45*(6), 639–667.

Andrews, K., & Heidegger, H. (1998). Information slices: Visualising and exploring large hierarchies using cascading, semi-circular discs. In *Proceedings of the 4th Symposium on Information Visualization (InfoVis'98), Research Triangle Park, NC, USA, October 19-20, 1998* (pp. 9–12). IEEE Computer Society Press.

Atkinson, R., & Shiffrin, R. (1968). *Human memory: A proposed system and its control processes*. *Psychology of Learning and Motivation* (Vol. 2).

Avgerou, C. (2000). Information systems: What sort of science is it? *Omega*, *28*(5), 567–579.

Bacharach, S. (1989). Organizational theories: Some criteria for evaluation. *Academy of Management Review*, *14*(4), 496–515.

Baird, H. S., Coates, A. L., & Fateman, R. J. (2003). PessimalPrint: A reverse Turing test. *International Journal on Document Analysis and Recognition*, *5*(2-3), 158–163.

Bannert, M. (2002). Managing cognitive load: Recent trends in cognitive load theory. *Learning and Instruction*, *12*(1), 139–146.

Barros, A. P., & Ter Hofstede, A. H. M. (1998). Towards the construction of workflow-suitable conceptual modelling techniques. *Information Systems Journal*, *8*(4), 313–337.

Baskerville, R. L., & Myers, M. D. (2002). Information systems as a reference discipline. *Mis Quarterly*, *26*(1), 1–14.

Becker, J., Rosemann, M., & Von Uthmann, C. (2000). Guidelines of business process modeling. In W. M. P. Van der Aalst, J. Desel, & A. Oberweis (Eds.), *Business Process Management. Models, Techniques, and Empirical Studies. Part I.* (Vol. LNCS 1806, pp. 30–49). Springer-Verlag Berlin Heidelberg.

Bertin, J. (2010). *Semiology of graphics: Diagrams, networks, maps.* California, USA: Economic & Social Research Institute.

Börger, E. (2012). Approaches to modeling business processes: A critical analysis of BPMN, workflow patterns and YAWL. *Software & Systems Modeling*, *11*(3), 305–318.

Breuker, D., Pfeiffer, D., & Becker, J. (2009). Reducing the variations in intra-and interorganizational business process modeling: An empirical evaluation. In *Proceedings of the 6th Internationale Tagung Wirtschaftinformatik (WI '09), Vienna, Austria, February 25-27* (pp. 203–212).

Brown, R. A. (2010). Conceptual modelling in 3D virtual worlds for process communication. In S. Link & A. Ghose (Eds.), *Proceedings of the 7th Asia-Pacific Conference on Conceptual Modelling (APCCM '10), Brisbane, Australia, January 18-21, 2010* (Vol. 110, pp. 25–32). Australian Computer Society.

Brown, R. A., Recker, J. C., & West, S. (2011). Using virtual worlds for collaborative business process modeling. *Business Process Management Journal*, *17*(3), 546–564.

Burch, M., Beck, F., & Diehl, S. (2008). Timeline trees: Visualizing sequences of transactions in information hierarchies. In *Proceedings of the 8th Working Conference on Advanced Visual Interfaces (AVI '08), Naples, Italy, May 28-30, 2008* (pp. 75–82). ACM Press.

Burton-Jones, A., Weber, R., & Wand, Y. (2009). Guidelines for empirical evaluations of conceptual modeling grammars. *Journal of the Association for Information Systems*, *10*(6), 495–532.

Cegielski, C. (2006). What makes a good programmer? *Communications of the ACM*, *49*(10), 73–75.

Chakraborty, S., Sarker, S., & Sarker, S. (2010). An exploration into the process of requirements elicitation: A grounded approach. *Journal of the Association for Information Systems*, *11*(4), 212–249.

Champy, J. (2002). *X-engineering the corporation: reinventing your business in the digital age*. Warner business books.

Chang, J. F. (2005). *Business process management systems: Strategy and implementation*. Taylor & Francis.

Claes, J., Vanderfeesten, I., Pinggera, J., Reijers, H. A., Weber, B., & Poels, G. (2013). Visualizing the process of process modeling with PPMCharts. In M. La Rosa & P. Soffer (Eds.), *Proceedings of the 10th International Conference on Business Process Management - Workshops (BPM '12), Tallinn, Estonia, September 3, 2012* (Vol. LNBIP 132, pp. 744–755). Springer.

Claes, J., Vanderfeesten, I., Pinggera, J., Reijers, H. A., Weber, B., & Poels, G. (2015). A visual analysis of the process of process modeling. *Information Systems and E-Business Management*, *13*(1), 147–190.

Claes, J., Vanderfeesten, I., Reijers, H. A., Pinggera, J., Weidlich, M., Zugal, S., … Poels, G. (2012). Tying process model quality to the modeling process: The impact of structuring, movement, and speed. In A. Barros, A. Gal, & E. Kindler (Eds.), *Proceedings of the 10th International Conference on Business Process Management (BPM '12), Tallinn, Estonia, September 3, 2012* (Vol. LNCS 7481, pp. 33–48). Springer.

Cowan, N. (2010). The magical mystery four: How is working memory capacity limited, and why? *Current Directions in Psychological Science*, *19*(1), 51–57.

Crapo, A. W., Waisel, L. B., Wallace, W. A., & Willemain, T. R. (2000). Visualization and the process of modeling: A cognitive-theoretic view. In R. Ramakrishnan, S. Stolfo, R. Bayardo, & I. Parsa (Eds.), *Proceedings of the 6th International Conference on Knowledge Discovery and Data Mining (KDD '00), Boston, MA, USA, August 20-23, 2000* (pp. 218–226). ACM Press.

Davenport, T. H. (1993). *Process innovation: Reengineering work through information technology*. Harvard Business School Press.

Davies, I., Green, P., Rosemann, M., Indulska, M., & Gallo, S. (2006). How do practitioners use conceptual modeling in practice? *Data & Knowledge Engineering*, *58*(3), 358–380.

De Jong, T. (2010). Cognitive load theory, educational research, and instructional design: Some food for thought. *Instructional Science*, *38*(2), 105–134.

Decker, G., & Mendling, J. (2009). Process instantiation. *Data & Knowledge Engineering*, *68*(9), 777–792.

Demarco, T., & Lister, T. (1985). Programmer performance and the effects of the workplace. In *Proceedings of the 8th International Conference on Software Engineering (ICSE '85), London, UK, August 28-30, 1985* (pp. 268–272). IEEE Computer Society Press.

Deming, W. E. (1982). *Out of the crisis: Quality, productivity and competitive position*. MIT PRess.

Dijkman, R. M., Dumas, M., & Ouyang, C. (2008). Semantics and analysis of business process models in BPMN. *Information and Software Technology*, *50*(12), 1281–1294.

Drucker, P. F. (1988). The coming of the new organization. *Harvard Business Review*, *66*(1), 45–53.

Dumas, M., Rosa, M. La, Mendling, J., Reijers, H. A., & La Rosa, M. (2013). *Fundamentals of business process management*. Springer Berlin Heidelberg.

Dumas, M., Van der Aalst, W. M. P., & Ter Hofstede, A. H. M. (2005). *Process-aware information systems: Bridging people and software through process technology*. *Process-Aware Information Systems: Bridging People and Software through Process Technology*. Hoboken, NY: John Wiley & Sons.

Effinger, P. (2013). A 3D-navigator for business process models. In M. La Rosa & P. Soffer (Eds.), *Proceedings of the 10th International Conference on Business Process Management - Workshops (BPM '12), Tallinn, Estonia, September 3, 2012* (Vol. LNBIP 132, pp. 737–743). Springer.

Engelen, L., & Van den Brand, M. (2010). Integrating textual and graphical modelling languages. *Electronic Notes in Theoretical Computer Science*, *253*(7), 105–120.

Fekete, J. D., Van Wijk, J. J., Stasko, J. T., & North, C. (2008). The value of information visualization. In A. Kerren, J. T. Stasko, J.-D. Fekete, & C. North (Eds.), *Information Visualization. Human-Centered Issues and Perspectives*. (Vol. LNCS 4950, pp. 1–18). Dagstuhl Castle, Germany: Springer-Verlag Berlin Heidelberg.

Felder, R., & Silverman, L. (1988). Learning and teaching styles in engineering education. *Engineering Education*, *78*(June), 674–681.

Figl, K., Mendling, J., & Strembeck, M. (2009). Towards a usability assessment of process modeling languages. In *Proceedings of the 8th Workshop Geschäftsprozessmanagement mit Ereignisgesteuerten Prozessketten (EPK 2009), Berlin, Germany, November 26-27, 2009* (Vol. CEUR-WS 55, pp. 118–136). CEUR-WS.

Figl, K., Recker, J. C., & Mendling, J. (2013). A study on the effects of routing symbol design on process model comprehension. *Decision Support Systems*, *54*(2), 1104–1118.

Francescomarino, C. Di, Rospocher, M., Ghidini, C., & Valerio, A. (2014). The role of semantic annotations in business process modelling. In *Proceedings of the 18th International Conference on Enterprise Distributed Object Computing Conference (EDOC '14), Ulm, Germany, September 1-5, 2014* (pp. 181–189). IEEE Computer Society Press.

Gantt, H. L. (1913). *Work, wages, and profits* (2nd ed.). New York: Engineering Magazine Company.

Gerjets, P., Scheiter, K., & Catrambone, R. (2004). Designing instructional examples to reduce intrinsic cognitive load: Molar versus modular presentation of solution procedures. *Instructional Science*, *32*(1-2), 33–58.

Godfrey-Smith, P. (2009). *Theory and reality: An introduction to the philosophy of science*. Chicago: University of Chicago Press.

Goedertier, S., Vanthienen, J., & Caron, F. (2013). Declarative business process modelling: Principles and modelling languages. *Enterprise Information Systems*, *9*(2), 161–185.

Golde, C. M., & Gallagher, H. A. (1999). The challenges of conducting interdisciplinary research in traditional doctoral programs. *Ecosystems*, *2*(4), 281–285.

Goldstine, H. H., & von Neumann, J. (1947). *Planning and coding of problems for an electronic computing instrument*. Institute For Advanced Study Project.

Goodman, N. (1968). *Languages of art: An approach to a theory of symbols*. Hackett Publishing.

Green, D. M., & Swets, J. A. (1966). *Signal detection theory and psychophysics*. John Wiley & Sons.

Gregor, S. (2006). The nature of theory in information systems. *MIS Quarterly*, *30*(3), 611–642.

Grover, V., Lyytinen, K., Srinivasan, A., & Tan, B. C. Y. (2008). Contributing to rigorous and forward thinking explanatory theory. *Journal of the Association for Information Systems*, *9*(2), 40–47.

Gruhn, V., & Laue, R. (2006). Complexity metrics for business process models. In W. Abramowicz & H. C. Mayr (Eds.), *Proceedings of the 9th International Conference on Business Information Systems (BIS 2006), Klagenfurt, Austria, May 31-June 2, 2006* (Vol. LNI 85, pp. 1–12). Gesellschaft für Informatik (GI).

Guo, H., Brown, R. A., & Rasmussen, R. K. (2013). A theoretical basis for using virtual worlds as a personalised process visualisation approach. In X. Franch & P. Soffer (Eds.), *Proceedings of the 25th Conference on Advanced Information Systems Engineering - Workshops (CAiSE '13), Valencia, Spain, June 17-21, 2013* (Vol. LNBIP 148, pp. 229–240). Springer.

Hahn, J., & Kim, J. (1999). Why are some diagrams easier to work with? Effects of diagrammatic representation on the cognitive integration process of systems analysis and design. *ACM Transactions on Computer-Human Interaction*, *6*(3), 181–213.

Håkansson, H., & Snehota, I. (1989). No business is an island: The network concept of business strategy. *Scandinavian Journal of Management*, *5*(3), 187–200.

Hammer, M., & Champy, J. (1993). *Reengineering the corporation: A manifesto for business revolution*. Harper Business Paperbacks.

Harrington, H. J. (1991). *Business process improvement: The breakthrough strategy for total quality, productivity, and competitiveness*. McGraw-Hill Education.

Hevner, A. R. R., March, S. T., Park, J., & Ram, S. (2004). Design science in information systems research. *MIS Quarterly*, *28*(1), 75–105.

Hildebrandt, T., Kriglstein, S., & Rinderle-Ma, S. (2012). Beyond visualization: On using sonification methods to make business processes more accesible to users. In M. A. Nees, B. N. Walker, & J. Freeman (Eds.), *Proceedings of the 18th International Conference on Auditory Display (ICAD '12), Atlanta, GA, USA, June 18-22, 2012* (pp. 248–249). Georgia Institute of Technology.

Hoppenbrouwers, S. J. B. A., Proper, H. A., & Van der Weide, T. P. (2005). A fundamental view on the process of conceptual modeling. In L. Delcambre, C. Kop, H. C. Mayr, J. Mylopoulos, & O. Pastor (Eds.), *Proceedings of the 24th International Conference on Conceptual Modeling (ER '05), Klagenfurt, Austria, October 24-28, 2005* (Vol. LNCS 3716, pp. 128–143). Springer.

Indulska, M., Recker, J. C., Rosemann, M., & Green, P. (2009). Business process modeling: Current issues and future challenges. In P. Van Eck, J. Gordijn, & R. Wieringa (Eds.), *Proceedings of the 21st International Conference on Advanced Information Systems Engineering (CAiSE '09), Amsterdam, The Netherlands, June 8-12, 2009* (Vol. LNCS 5565, pp. 501–514). Springer.

Jablonski, S., & Bussler, C. (1996). *Workflow management: Modeling concepts, architecture and implementation*. International Thomson Computer Press London.

Jeston, J., & Nelis, J. (2008). *Management by process: A roadmap to sustainable business process management*. Elsevier/Butterworth-Heinemann.

Johnson, B. (1993). *Treemaps: Visualizing hierarchical and categorical data*. University of Maryland.

Johnson, B., & Shneiderman, B. (1991). Tree-maps: A space-filling approach to the visualization of hierarchical information structures. In G. M. Nielson & L. Rosenblum (Eds.), *Proceedings of the 2nd Conference on Visualization '91 (VIS '91), San Diego, CA, USA, October 24-26, 1991* (pp. 284–291). IEEE Computer Society Press.

Juran, J. M., & Gryna, F. M. (1988). *Juran's quality control handbook*. (J. M. Juran & F. M. Gryna, Eds.) (4th ed.). McGraw-Hill.

Kalyuga, S. (2007). Expertise reversal effect and its implications for learner-tailored instruction. *Educational Psychology Review*, *19*(4), 509–539.

Kalyuga, S., Ayres, P., Chandler, P., & Sweller, J. (2003). The expertise reversal effect. *Educational Psychologist*, *38*(1), 23–31.

Kaplan, B., & Maxwell, J. (2005). Qualitative research methods for evaluating computer information systems. In J. G. Anderson & C. E. Aydin (Eds.), *Evaluating the Organizational Impact of Healthcare Information Systems, Part I* (pp. 30–55). Springer-Verlag Berlin Heidelberg.

Khatri, V., Vessey, I., Ram, S., & Ramesh, V. (2006). Cognitive fit between conceptual schemas and internal problem representations: The case of geospatio-temporal conceptual schema comprehension. *IEEE Transactions on Professional Communication*, *49*(2), 109–127.

Khatri, V., Vessey, I., Ramesh, V., Clay, P., & Park, S.-J. (2006). Understanding conceptual schemas: Exploring the role of application and IS domain knowledge. *Information Systems Research*, *17*(1), 81–99.

Kim, J., Hahn, J., & Hahn, H. (2000). How do we understand a system with (so) many diagrams? Cognitive integration processes in diagrammatic reasoning. *Information Systems Research*, *11*(3), 284–303.

Kock, N. F., Verville, J., Danesh-Pajou, A., & DeLuca, D. (2009). Communication flow orientation in business process modeling and its effect on redesign success: Results from a field study. *Decision Support Systems*, *46*(2), 562–575.

Kolb, J., Zimoch, M., Weber, B., & Reichert, M. (2014). How social distance of process designers affects the process of process modeling: Insights from a controlled experiment. In S. Kim, C.-C. Hung, & J. Hong (Eds.), *Proceedings of the 29th Annual ACM Symposium on Applied Computing (SAC '14), Gyeongju, South Korea, March 24-28, 2014* (pp. 1364–1370). ACM Press.

Koschmider, A., & Reijers, H. A. (2013). Improving the process of process modelling by the use of domain process patterns. *Enterprise Information Systems*, *9*(1), 29–57.

Krishnan, M., Mukhopadhyay, T., & Zubrow, D. (1999). Software process models and project performance. *Information Systems Frontiers*, *1*(3), 267–277.

Krogstie, J., Sindre, G., & Jørgensen, H. (2006). Process models representing knowledge for action: A revised quality framework. *European Journal of Information Systems*, *15*(1), 91–102.

La Rosa, M., Reijers, H. a., Van Der Aalst, W. M. P., Dijkman, R. M., Mendling, J., Dumas, M., & García-Bañuelos, L. (2011). APROMORE: An advanced process model repository. *Expert Systems with Applications*, *38*(6), 7029–7040. http://doi.org/10.1016/j.eswa.2010.12.012

Larkin, J. H., & Simon, H. A. (1987). Why a Diagram is (Sometimes) Worth Ten Thousand Words. *Cognitive Science*, *11*(1), 65–100.

Laue, R., & Mendling, J. (2008). The impact of structuredness on error probability of process models. In *Information Systems and e-Business Technologies* (pp. 585–590). Springer.

Laue, R., & Mendling, J. (2010). Structuredness and its significance for correctness of process models. *Information Systems and E-Business Management*, *8*(3), 287–307.

Lindland, O. I., Sindre, G., & Solvberg, A. (1994). Understanding quality in conceptual modeling. *Software, IEEE*, *11*(2), 42–49.

Lohse, G. L. (1993). A cognitive model for understanding graphical perception. *Human-Computer Interaction*, *8*(4), 353–388.

Lumley, T., Diehr, P., Emerson, S., & Chen, L. (2002). The importance of the normality assumption in large public health data sets. *Annual Review of Public Health*, *23*, 151–169.

March, S. T., & Smith, G. F. (1995). Design and natural science research on information technology. *Decision Support Systems*, *15*(4), 251–266.

Marsh, E. R. (1975). The harmonogram of Karol Adamiecki. *Academy of Management Journal*, *18*(2), 358–364.

Mccormack, K. (2001). Business process orientation: Do you have it? *Quality Progress*, *34*(1), 51–60.

Mccormack, K., & Johnson, W. C. (2001). *Business process orientation: Gaining the e-business competitive advantage*. CRC Press.

Mccormack, K., Willems, J., Van Den Bergh, J., Deschoolmeester, D., Willaert, P., Štemberger, I., … Valadares, P. (2009). A global investigation of key turning points in business process maturity. *Business Process Management Journal*, *15*(5), 792–815.

McGill, R., Tukey, J., & Larsen, W. A. (1978). Variations of box plots. *The American Statistician*, *32*(1), 12–16.

Mendling, J. (2008). *Metrics for process models: Empirical foundations of verification, error prediction and guidelines for correctness. LNBIP 6.* (J. Mylopoulos & N. M. Sadeh, Eds.). Berlin Heidelberg: Springer Science & Business Media.

Mendling, J., Recker, J. C., & Reijers, H. A. (2010). On the usage of labels and icons in business process modeling. *International Journal of Information System Modeling and Design*, *1*(2), 19.

Mendling, J., Reijers, H. A., & Van der Aalst, W. M. P. (2010). Seven process modeling guidelines (7PMG). *Information and Software Technology*, *52*(2), 127–136.

Mendling, J., Sánchez-González, L., García, F., & La Rosa, M. (2012). Thresholds for error probability measures of business process models. *Journal of Systems and Software*, *85*(5), 1188–1197.

Mendling, J., Verbeek, H. M. W., Van Dongen, B. F., Van der Aalst, W. M. P., & Neumann, G. (2008). Detection and prediction of errors in EPCs of the SAP reference model. *Data & Knowledge Engineering*, *64*(1), 312–329.

Miller, G. A. (1956). The magical number seven, plus or minus two: Some limits on our capacity for processing information. *Psychological Review*, *63*(2), 81–97.

Moody, D. L. (2005). Theoretical and practical issues in evaluating the quality of conceptual models: Current state and future directions. *Data & Knowledge Engineering*, *55*(3), 243–276.

Moody, D. L. (2009). The “physics” of notations: Toward a scientific basis for constructing visual notations in software engineering. *IEEE Transactions on Software Engineering*, *35*(6), 756–779.

Moore, P., & Pham, H. Van. (2015). On context and the open world assumption. In *Proceedings of the 29th International Conference on Advanced Information Networking and Applications - Workshops (WAINA ’15), Gwangju, Korea, March 24-27, 2015* (pp. 387–392). IEEE Computer Society Press.

Morris, W. T. (1967). On the art of modeling. *Management Science*, *13*(12), B707–B717.

Motro, A., & Smets, P. (1996). *Uncertainty management in information systems: From needs to solutions*. US: Springer.

Murata, T. (1989). Petri nets: Properties, analysis and applications. In *Proceedings of the IEEE* (Vol. 77, pp. 541–580). IEEE Computer Society Press.

Myers, M. D. (1997). Qualitative research in information systems. *MIS Quarterly*, *21*(2), 241–242.

Mylopoulos, J. (1992). Conceptual modelling and Telos. In P. Loucopoulos & R. Zicari (Eds.), *Conceptual Modelling, Databases and CASE: An Integrated View of Information Systems Development* (pp. 1–20). John Wiley & Sons New York.

Nagel, E. (1979). *The structure of science*. Indianapolis: Hackett Publishing Company.

Naumann, J. D. (1986). The role of frameworks in MIS research. In *Proceedings of the 18th DSI National Meeting, Honolulu, Hawaii, November 23-25, 1986* (pp. 569–571).

Nelson, H. J., Poels, G., Genero, M., & Piattini, M. (2012). A conceptual modeling quality framework. *Software Quality Journal*, *20*(1), 201–228.

Neuberg, S., & Newsom, J. (1993). Personal need for structure: Individual differences in the desire for simpler structure. *Journal of Personality and Social Psychology*, *65*(1), 113–131.

Neumann, P., Schlechtweg, S., & Carpendale, S. (2005). Arctrees: Visualizing relations in hierarchical data. In K. Brodlie, D. Duke, & K. Joy (Eds.), *Proceedings of the 7th Joint Eurographics / IEEE VGTC Conference on Visualization (EUROVIS '05), Leeds, UK, June 1-3, 2005* (pp. 53–60). Eurographics Association.

Nordbotten, J. C., & Crosby, M. E. (1999). The effect of graphic style on data model interpretation. *Information Systems Journal*, *9*(2), 139–155.

Norman, D. A. (2002). *The design of everyday things*. New York, USA: Basic Books.

Novak, O. (2002). *Visualization of large graphs (study report)*. Prague: Czech Technical University.

Nunamaker, J. F., & Chen, M. (1990). Systems development in information systems research. In *Proceedings of the 23th Annual Hawaii International Conference on System Sciences (HICSS '90), Kailua-Kona, HI, USA, January 2-5, 1990* (pp. 631–640). IEEE Computer Society Press.

Olivé, A. (2007). *Conceptual modeling of information systems*. Springer-Verlag New York Inc.

OMG. (2011a). *Business Process Model and Notation (BPMN) version 2.0*.

OMG. (2011b). *Unified Modeling Language, superstructure (UML) version 2.4.1*.

OMG. (2015). BPMN Quick Guide. Retrieved June 8, 2015, from http://www.bpmn.org/

Paas, F. G. W. C., Renkl, A., & Sweller, J. (2003). Cognitive load theory and instructional design: Recent developments. *Educational Psychologist*, *38*(1), 1–4.

Paas, F. G. W. C., Tuovinen, J. E., Tabbers, H., & Van Gerven, P. W. M. (2003). Cognitive load measurement as a means to advance cognitive load theory. *Educational Psychologist*, *38*(1), 63–71.

Paivio, A. (1990). *Mental representations*. Oxford University Press.

Peffers, K., Tuunanen, T., Rothenberger, M. A., & Chatterjee, S. (2007). A design science research methodology for information systems research. *Journal of Management Information Systems*, *24*(3), 45–77.

Petri, C. A. (1962). *Kommunikation mit automaten*. Universität Bonn.

Phalp, K. T. (1998). The CAP framework for business process modelling. *Information and Software Technology*, *40*(13), 731–744.

Pinggera, J., Furtner, M., Martini, M., Sachse, P., Katharina, R., Zugal, S., & Weber, B. (2013). Investigating the process of process modeling with eye movement analysis. In M. La Rosa & P. Soffer (Eds.), *Proceedings of the 10th International Conference on Business Process Management - Workshops (BPM '12), Tallinn, Estonia, September 3, 2012* (Vol. LNBIP 132, pp. 438–450). Springer.

Pinggera, J., Soffer, P., Fahland, D., Weidlich, M., Zugal, S., Weber, B., … Mendling, J. (2013). Styles in business process modeling: An exploration and a model. *Software & Systems Modeling*, (published online, May 2013).

Pinggera, J., Soffer, P., Zugal, S., Weber, B., Weidlich, M., Fahland, D., … Mendling, J. (2012). Modeling styles in business process modeling. In I. Bider, T. Halpin, J. Krogstie, S. Nurcan, E. Proper, R. Schmidt, … S. Wrycza (Eds.), *Proceedings of the 13th International Conference on Business Process Modeling, Development, and Support (BPMDS '12) and the 17th International Conference on Exploring Modelling Methods for Systems Analysis and Design (EMMSAD '12), Gdansk, Poland, June 25-* (Vol. LNBIP 113, pp. 151–166). Springer.

Pinggera, J., Zugal, S., & Furtner, M. (2014). The modeling mind: Behavior patterns in process modeling. In I. Bider, K. Gaaloul, J. Krogstie, S. Nurcan, H. A. Proper, R. Schmidt, & P. Soffer (Eds.), *Proceedings of the 15th International Conference on Business Process Modeling, Development, and Support (BPMDS '14) and the 19th International Conference on Exploring Modelling Methods for Systems Analysis and Design (EMMSAD '14), Thessaloniki, Greece, Ju* (Vol. LNBIP 175, pp. 1–16). Springer.

Pinggera, J., Zugal, S., & Weber, B. (2010). Investigating the process of process modeling with Cheetah Experimental Platform. In B. Mutschler, J. Recker, R. Wieringa, J. Ralyte, & P. Plebani (Eds.), *Proceedings of the 1st International Workshop on Empirical Research in Process-Oriented Information Systems (ER-POIS '10), Hammamet, Tunisia, June 8, 2010* (Vol. CEUR -WS 6, pp. 13–18). CEUR-WS.

Pinggera, J., Zugal, S., Weber, B., Fahland, D., Weidlich, M., Mendling, J., & Reijers, H. A. (2010). How the structuring of domain knowledge helps casual process modelers. In J. Parsons, M. Saeki, P. Shoval, C. Woo, & Y. Wand (Eds.), *Proceedings of the 29th International Conference on Conceptual Modeling (ER '10), Vancouver, Canada, November 1-4, 2010* (Vol. LNCS 6412, pp. 445–451). Springer.

Pinggera, J., Zugal, S., Weidlich, M., Fahland, D., Weber, B., Mendling, J., & Reijers, H. A. (2012). Tracing the process of process modeling with Modeling Phase Diagrams. In F. Daniel, K. Barkaoui, & S. Dustdar (Eds.), *Proceedings of the 9th International Conference on Business Process Management - Workshops (BPM '11), Clermont-Ferrand, France, August 28-September 2, 2012* (Vol. LNBIP 99, pp. 370–382). Springer.

Pithers, R. T. (2002). Cognitive learning style: A review of the field dependent-field independent approach. *Journal of Vocational Education & Training*, *54*(1), 117–132.

Pollock, E., Chandler, P., & Sweller, J. (2002). Assimilating complex information. *Learning and Instruction*, *12*(1), 61–86.

Polyvyanyy, A. (2012). *Structuring process models*. University of Potsdam.

Polyvyanyy, A., Smirnov, S., & Weske, M. (2010). Business process model abstraction. In J. Vom Brocke & M. Rosemann (Eds.), *Handbook on Business Process Management 1* (pp. 149–166). Springer-Verlag Berlin Heidelberg.

Popper, K. (2005). *The logic of scientific discovery*. Taylor & Francis.

Porter, M. E., & Millar, V. E. (1985). How information gives you competitive advantage. *Harvard Business Review*, *63*(4), 149–160.

Purchase, H. (1997). Which aesthetic has the greatest effect on human understanding? In G. DiBattista (Ed.), *Proceedings of the 5th International Symposium on Graph Drawing (GD '97), Rome, Italy, September 18-20, 1997* (Vol. LNCS 1353, pp. 248–261). Springer.

Quinlan, P. T. (2003). Visual feature integration theory: Past, present, and future. *Psychological Bulletin*, *129*(5), 643–73.

Reason, J. (1990). *Human Error*. Cambridge University Press.

Recker, J. C., Mendling, J., & Hahn, C. (2013). How collaborative technology supports cognitive processes in collaborative process modeling: A capabilities-gains-outcome model. *Information Systems*, *38*(8), 1031–1045.

Recker, J. C., Safrudin, N., & Rosemann, M. (2012). How novices design business processes. *Information Systems*, *37*(6), 557–573. http://doi.org/10.1016/j.is.2011.07.001

Reichert, M. (2013). Visualizing large business process models: Challenges, techniques, applications. In M. La Rosa & P. Soffer (Eds.), *Proceedings of the 10th International Conference on Business Process Management - Workshops (BPM '12), Tallinn, Estonia, September 3, 2012* (Vol. LNBIP 132, pp. 725–736). Springer.

Reichert, M., & Weber, B. (2013). Process Change Patterns: Recent Research, Use Cases, Research Directions. In J. Bubenko, J. Krogstie, O. Pastor, B. Pernici, C. Rolland, & A. Sølvberg (Eds.), *Seminal Contributions to Information Systems Engineering. 25 Years of CAiSE* (pp. 397–404). Berling Heidelberg: Springer-Verlag.

Reijers, H. A., Freytag, T., Mendling, J., & Eckleder, A. (2011). Syntax highlighting in business process models. *Decision Support Systems*, *51*(3), 339–349.

Reijers, H. A., & Mendling, J. (2008). Modularity in process models: Review and effects. In M. Dumas, M. Reichert, & M.-C. Shan (Eds.), *Proceedings of the 6th International Conference on Business Process Management (BPM '08), Milan, Italy, September 1-4, 2008* (Vol. LNCS 5240, pp. 20–35). Springer.

Reijers, H. A., & Mendling, J. (2011). A study into the factors that influence the understandability of business process models. *IEEE Transactions on Systems, Man, and Cybernetics, Part A: Systems and Humans*, *41*(3), 449–462.

Reijers, H. A., Mendling, J., & Recker, J. C. (2010). Business process quality management. In J. Vom Brocke & M. Rosemann (Eds.), *Handbook on Business Process Management 1* (Vol. 1, pp. 167–185). Springer-Verlag Berlin Heidelberg.

Reisig, W., & Rozenberg, G. (1998). *Lectures on petri nets I: Basic models, advances in petri nets, LNCS 1491*. Springer Berlin Heidelberg.

Reiter, R. (1978). On closed world data bases. In H. Gallaire & J. Minker (Eds.), *Logic and Data Bases* (pp. 55–76). Springer-Verlag Berlin Heidelberg.

Rittgen, P. (2010). Quality and perceived usefulness of process models. In *Proceedings of the 25th Symposium on Applied Computing (SAC '10), Sierre, Switzerland, March 22-26, 2010* (pp. 65–72). ACM Press.

Rockwell, S., & Bajaj, A. (2005). COGEVAL: Applying cognitive theories to evaluate conceptual models. *Advanced Topics in Database Research*, *4*, 255–282.

Rogers, Y., & Rutherford, A. (1992). *Models in the mind: Theory, perspective, and application*. Londen, UK: Academic Press.

Sánchez-González, L., García, F., Ruiz, F., & Piattini, M. (2013). Toward a quality framework for business process models. *International Journal of Cooperative Information Systems*, *22*(01), 1350003.

Scheer, A.-W. (1998). *Business process engineering: Reference models for industrial enterprises*. Springer-Verlag.

Sedrakyan, G., Snoeck, M., & De Weerdt, J. (2014). Process mining analysis of conceptual modeling behavior of novices: Empirical study using JMermaid modeling and experimental logging environment. *Computers in Human Behavior*, *41*(December), 486–503.

Shaft, T. M., & Vessey, I. (2006). The role of cognitive fit in the relationship between software comprehension and modification. *MIS Quarterly*, *30*(1), 29–55.

Silver, B. (2011). *BPMN: Method & style* (2nd editio). Cody-Cassidy Press.

Simon, H. A. (1996). *The sciences of the artificial*. MIT Press.

Sinha, A. P., & Vessey, I. (1992). Cognitive fit: An empirical study of recursion and iteration. *IEEE Transactions on Software Engineering*, *18*(5), 368–379.

Smith, A. (2014). *The wealth of nations. Reprint, originally pulbished in 1776*. Createspace Independent Pub.

Smith, H., & Fingar, P. (2003). *Business process management: The third wave. Meghan-Kiffer Press, Florida* (Vol. 1). Meghan-Kiffer Press.

Soffer, P., Kaner, M., & Wand, Y. (2012). Towards understanding the process of process modeling: Theoretical and empirical considerations. In F. Daniel, K. Barkaoui, & S. Dustdar (Eds.), *Business Process Management Workshops. BPM 2011 International Workshops, Clermont-Ferrand, France, August 29, 2011, Revised Selected Papers, Part I.* (Vol. LNBIP 99, pp. 357–369). Springer Berlin Heidelberg.

Song, M. S., & Van der Aalst, W. M. P. (2007). Supporting process mining by showing events at a glance. In K. Chari & A. Kumar (Eds.), *Proceeding of the Seventeenth Annual Workshop on Information Technologies and Systems (WITS'07)* (pp. 139–145).

Sperling, G. (1963). A model for visual memory tasks. *Human Factors*, *5*(1), 19–31.

Stark, J., & Esswein, W. (2012). Rules from cognition for conceptual modelling. In P. Atzeni, D. Cheung, & S. Ram (Eds.), *Proceedings of the 31st International Conference on Conceptual Modeling (ER '12), Florence, Italy, October 15-18, 2012* (Vol. LNCS 7532, pp. 78–87). Springer.

Stasko, J. T., & Zhang, E. (2000). Focus+context display and navigation techniques for enhancing radial, space-filling hierarchy visualizations. In *Proceedings of the 6th Symposium on Information Visualization (InfoVis '00), Salt Lake City, UT, USA, October 9-10, 2000* (pp. 57–65). IEEE Computer Society Press.

Sweller, J. (1988). Cognitive load during problem solving: Effects on learning. *Cognitive Science*, *12*(2), 257–285.

Sweller, J., & Chandler, P. (1994). Why some material is difficult to learn. *Cognition and Instruction*, *12*(3), 185–233.

Sweller, J., Van Merriënboer, J. J. G., & Paas, F. G. W. C. (1998). Cognitive architecture and instructional design. *Educational Psychology Review*, *10*(3), 251–296.

Thompson, M. M., Naccarato, M. E., & Parker, K. E. (1989). Assessing cognitive need: The development of the Personal Need for Structure and Personal Fear of Invalidity scales. In *Proceedings of the 48th Annual Meeting of the Canadian Psychological Association, Halifax, Canada*.

Treisman, A. M. (1982). Perceptual grouping and attention in visual search for features and for objects. *Journal of Experimental Psychology: Human Perception and Performance*, *8*(2), 194–214.

Treisman, A. M., & Gelade, G. (1980). A feature-integration theory of attention. *Cognitive Psychology*, *12*(1), 97–136.

Tufte, E. R. (1983). *The visual display of quantitative information*. Graphics Press.

Van der Aalst, W. M. P. (1998). The application of petri nets to workflow management. *Journal of Circuits Systems and Computers*, *8*(1), 21–66.

Van der Aalst, W. M. P. (2011). *Process mining: Discovery, conformance and enhancement of business processes*. Springer.

Van der Aalst, W. M. P., & Ter Hofstede, A. H. M. (2002). Workflow patterns: On the expressive power of (petri-net-based) workflow languages. In *Proceedings of the Fourth Workshop on the Practical Use of Coloured Petri Nets and CPN Tools (CPN 2002)* (Vol. 560, pp. 1–20). Citeseer.

Van der Aalst, W. M. P., & Ter Hofstede, A. H. M. (2005). YAWL: Yet Another Workflow Language. *Information Systems*, *30*(4), 245–275.

Van der Aalst, W. M. P., Van Hee, K. M., Ter Hofstede, A. H. M., Sidorova, H., Verbeek, H. M. W., Voorhoeve, M., & Wynn, M. T. (2011). Soundness of workflow nets: Classification, decidability, and analysis. *Formal Aspects of Computing*, *23*(3), 333–363.

Van Looy, A., De Backer, M., & Poels, G. (2014). A conceptual framework and classification of capability areas for business process maturity. *Enterprise Information Systems*, *8*(2), 188–224.

Van Merriënboer, J. J. G. (1997). *Training complex cognitive skills: A four-component instructional design model for technical training*. Englewood Cliffs, New Jersey: Educational Technology.

Van Merriënboer, J. J. G., Kirschner, P. A., & Kester, L. (2003). Taking the load off a learner's mind: Instructional design for complex learning. *Educational Psychologist*, *38*(1), 5–13.

Van Merriënboer, J. J. G., & Sweller, J. (2005). Cognitive load theory and complex learning: Recent developments and future directions. *Educational Psychology Review*, *17*(2), 147–177.

Vanderfeesten, I., Cardoso, J., Mendling, J., Reijers, H. A., & Van der Aalst, W. M. P. (2007). Quality metrics for business process models. In L. Fisher (Ed.), *BPM and Workflow handbook* (pp. 179–190). Future Strategies.

Vanhatalo, J. (2007). Faster and more focused control-flow analysis for business process models through SESE decomposition. In B. J. Krämer, K.-J. Lin, & P. Narasimhan (Eds.), *Proceedings of the 5th International Conference on Service-Oriented Computing (ICSOC '07), Vienna, Austria, September 17-20, 2007* (Vol. LNCS 4749, pp. 43–55). Springer.

Vessey, I. (1991). Cognitive fit: A theory-based analysis of the graphs versus tables literature. *Decision Sciences*, *22*(2), 219–240.

Vessey, I., & Galletta, D. (1991). Cognitive fit: An empirical study of information acquisition. *Information Systems Research*, *2*(1), 63–84.

Vessey, I., & Weber, R. (1986). Structured tools and conditional logic: An empirical investigation. *Communications of the ACM*, *29*(1), 48–57.

Weber, B., Pinggera, J., Torres, V., & Reichert, M. (2013). Change patterns in use: A critical evaluation. In S. Nurcan, H. A. Proper, P. Soffer, J. Krogstie, R. Schmidt, T. Halpin, & I. Bider (Eds.), *Proceedings of the 14th International Conference on Business Process Modeling, Development, and Support (BPMDS '13) and the 18th International Conference on Exploring Modelling Methods for Systems Analysis and Design (EMMSAD '13), Valencia, Spain, June 25* (Vol. LNBIP 147, pp. 261–276). Springer.

Weber, B., Pinggera, J., Zugal, S., & Wild, W. (2010). Handling events during business process execution: An empirical test. In B. Mutschler, J. Recker, & R. Wieringa (Eds.), *Proceedings of the 1st International Workshop on Empirical Research in Process-Oriented Information Systems (ER-POIS '10), Hammamet, Tunisia, June 8, 2010* (Vol. CEUR-WS 60, pp. 19–30). CEUR-WS.

Weber, B., Reichert, M., & Rinderle-Ma, S. (2008). Change patterns and change support features: Enhancing flexibility in process-aware information systems. *Data & Knowledge Engineering*, *66*(3), 438–466.

Weber, B., Zeitelhofer, S., & Pinggera, J. (2014). How advanced change patterns impact the process of process modeling. In I. Bider, K. Gaaloul, J. Krogstie, S. Nurcan, H. A. Proper, R. Schmidt, & P. Soffer (Eds.), *Proceedings of the 15th International Conference on Business Process Modeling, Development, and Support (BPMDS '14) and the 19th International Conference on Exploring Modelling Methods for Systems Analysis and Design (EMMSAD '14), Thessaloniki, Greece, Ju* (Vol. LNBIP 175, pp. 17–32). Springer.

Weber, R. (1997). *Ontological foundations of information systems*. Melbourne, Victoria, Australia: Coopers & Lybrand.

Weber, R. (2012). Evaluating and developing theories in the information systems discipline. *Journal of the Association for Information Systems*, *13*(1), 1–31.

Weick, K. E. (1989). Theory construction as disciplined imagination. *Academy of Management Review*, *14*(4), 516–531.

Weidlich, M., Zugal, S., Pinggera, J., Weber, B., Reijers, H. A., & Mendling, J. (2010). The impact of change task type on maintainability of process models. In B. Mutschler, J. Recker, & R. Wieringa (Eds.), *Proceedings of the 1st International Workshop on Empirical Research in Process-Oriented Information Systems (ER-POIS '10), Hammamet, Tunisia, June 8, 2010* (Vol. CEUR-WS 60, pp. 1–12). CEUR-WS.

Weijters, A. J. M. M., & Van der Aalst, W. M. P. (2001). Process mining: Discovering workflow models from event-based data. In B. Kröse, M. De Rijke, G. Schreiber, & M. Van Someren (Eds.), *Proceedings of the 13th Belgium-Netherlands Conference on Artificial Intelligence (BNAIC '01), Maastricht, the Netherlands, October 25-26, 2001* (pp. 283–290).

Weske, M. (2007). *Business process management: Concepts, languages, architectures* (1st ed.). Springer Berlin Heidelberg.

Whetten, D. (1989). What constitutes a theoretical contribution? *Academy of Management Review*, *14*(4), 490–495.

Wiedenbeck, S., & Ramalingam, V. (1999). A comparison of the comprehension of object-oriented and procedural programs by novice programmers. *Interacting with Computers*, *11*(3), 255–282.

Willaert, P., Van Den Bergh, J., Willems, J., & Deschoolmeester, D. (2007). The process-oriented organisation: A holistic view developing a framework for business process orientation maturity. In G. Alonso, P. Dadam, & M. Rosemann (Eds.), *Proceedings of the 5th International Conference on Business Process Management (BPM '07), Brisbane, Australia, September 24-28, 2007* (Vol. LNCS 4714, pp. 1–15). Springer.

Wilson, J. M. (2003). Gantt charts: A centenary appreciation. *European Journal of Operational Research*, *149*(2), 430–437.

Winn, W. (1990). Encoding and retrieval of information in maps and diagrams. *IEEE Transactions on Professional Communication*, *33*(3), 103–107.

Winn, W. (1993). An account of how readers search for information in diagrams. *Contemporary Educational Psychology*, *18*(2), 162–185.

Witkin, H. A., & Goodenough, D. R. (1981). *Cognitive styles: Essence and origins, field dependence and field independence*. Michigan: International Universities Press.

Wolf, I., & Soffer, P. (2014). Supporting BPMN model creation with routing patterns. In L. Iliadis, M. Papazoglou, & K. Pohl (Eds.), *Proceedings of the 26th International Conference on Advanced Information Systems Engineering - Workshops (CAiSE '14), Thessaloniki, Greece, June 16-20, 2014* (Vol. LNBIP 178, pp. 171–181). Springer.

Wolf, K. (2007). Generating petri net state spaces. In J. Kleijn & A. Yakovlev (Eds.), *Proceedings of the 28th International Conference on Applications and Theory of Petri Nets and Other Models of Concurrency (ICATPN '07), Siedlce, Poland, June 25-29, 2007* (Vol. LNCS 4546, pp. 29–42). Springer.

Xiao, L., & Zheng, L. (2012). Business process design: Process comparison and integration. *Information Systems Frontiers*, *14*(2), 363–374.

Zhao, J. L., & Cheng, H. K. (2005). Web services and process management: A union of convenience or a new area of research? *Decision Support Systems*, *40*(1), 1–8.

Zugal, S., & Pinggera, J. (2014). Low-cost eye-trackers: Useful for information systems research? In L. Iliadis, M. Papazoglou, & K. Pohl (Eds.), *Proceedings of the 26th International Conference on Advanced Information Systems Engineering - Workshops (CAiSE '14), Thessaloniki, Greece, June 16-20, 2014* (Vol. LNBIP 178, pp. 159–170). Springer.

Zugal, S., Pinggera, J., Reijers, H. A., Reichert, M., & Weber, B. (2012). Making the case for measuring mental effort. In *Proceedings of the 15th International Conference on Model Driven Engineering Languages and Systems (MODELS '12), Innsbruck, Austria, September 30-October 5, 2012* (pp. 37–42). ACM Press.

Zugal, S., Soffer, P., Haisjackl, C., Pinggera, J., Reichert, M., & Weber, B. (2013). Investigating expressiveness and understandability of hierarchy in declarative business process models. *Software & Systems Modeling*, (published online, June 2013).

Zur Muehlen, M., & Recker, J. C. (2008). How much language is enough? Theoretical and practical use of the Business Process Modeling Notation. In Z. Bellahsène & M. Léonard (Eds.), *Proceedings of the 20th International Conference on Advanced Information Systems Engineering (CAiSE '08), Montpellier, France, June 16-20, 2008* (Vol. LNCS 5074, pp. 465–479). Springer.

Index

A

Appendix A Design process of the PPMChart

In this appendix we explain how the PPMChart was designed and why we started from the Dotted Chart Analysis plug-in in ProM to do so.

After the Cheetah Experimental Platform tool was developed and a great amount of observational data was collected with the tool, we wanted to investigate if and how process mining techniques could be used to analyze the process of process modeling with the collected data. At first, we experimented with process discovery techniques in combination with manipulating the data to change the level of detail or to switch between absolute and relative timing. Figure A.1 and Figure A.2 show an example of how we used Heuristics Miner and Fuzzy Miner to facilitate gaining insights in the Process of Process Modeling (PPM). It was difficult to get useful results because we had problems finding a good balance between representing enough detail and not getting lost in the complexity of the results. Therefore, we turned a first time to Dotted Chart (see Figure A.3), which is known for being able to represent detailed information in such a way that the user can easily zoom in on details or zoom out to take a helicopter view on the process (Song & Van der Aalst, 2007).

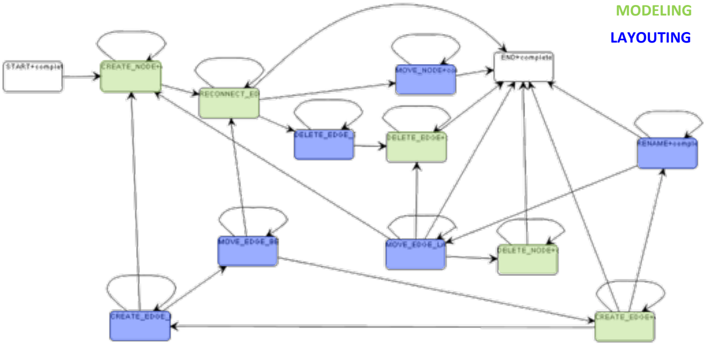


Figure A.1. Example of how we used Heuristic Miner to gain insights in the PPM

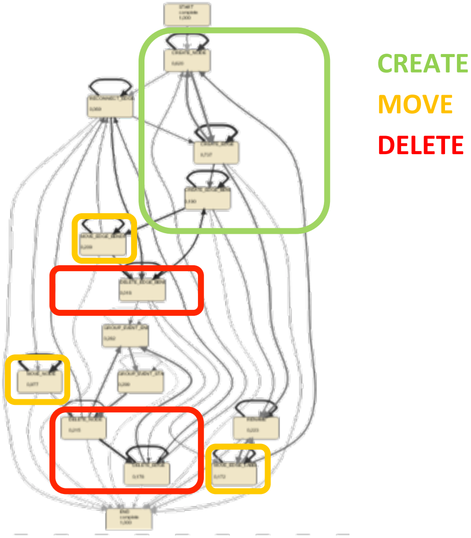


Figure A.2. Example of how we used Fuzzy Miner to gain insights in the PPM

Options

Component type: Instance ID

Time option: Actual

Time sort (chart): hours

Color By: Task ID

☑ Size shows # of events

Size/Shape By: None

Sort By: Number of Events

☐ descending

Mouse Mode: Select

Time sort (metrics): hours

Component Type: Instance ID
# of components: 0

items	values
time(first)	1/01/11 0:00
time(end)	1/01/11 23:59
avg spread	0.0
min spread	0.0
max spread	0.0

Component ID: 77
# of dots: 1

items	values
time(first)	1/01/70 1:00
time(end)	1/01/70 1:00

Set colors

CREATE:	push to change
DELETE:	push to change
END:	push to change
MOVE:	push to change
OTHER:	push to change
RENAME:	push to change
START:	push to change

Figure A.3. Example of how we used Dotted Chart Analysis to gain insights in the PPM

Inspired by these process mining techniques, but disappointed by their results in the context of our goal to use them in our explorative study, we decided to develop our own technique. In order to gain deeper insights in the data, we believed that an information visualization technique was needed. After some iterations of sketching, comparing and evaluating different visualizations (see for example Figure A.4 and Figure A.5), we realized how our attempts evolved to something that was similar to the Dotted Chart, which we had considered before, but now representing information of one process instance only (as discussed in Chapter 2).

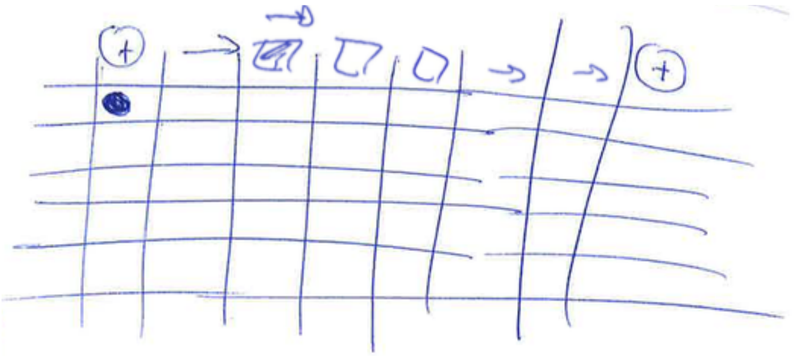


Figure A.4. Sketch of a table where the top line contains the process model elements and below a dot would be used to indicated the order of creating or altering these elements

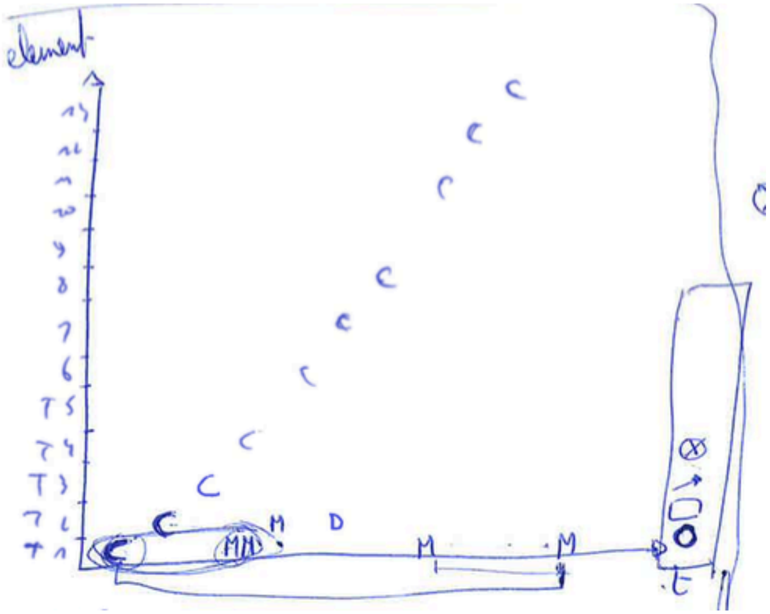


Figure A.5. Sketch of a chart where each line represents the events of one trace in the event log. At the right it is shown how these traces correspond with an element of the model (C = create, M = move, D = delete)

At first we used the existing Dotted Chart Analysis implementation in ProM to generate the PPMCharts (see Figure A.6), but afterwards we decided to redesign the visualization entirely to make it fit the needs of our study. This process and the resulting PPMChart implementation is described extensively in Chapter 2.

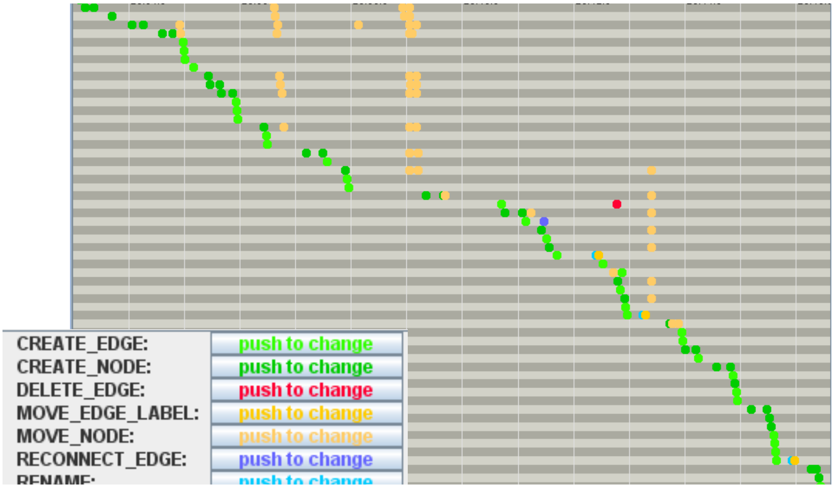
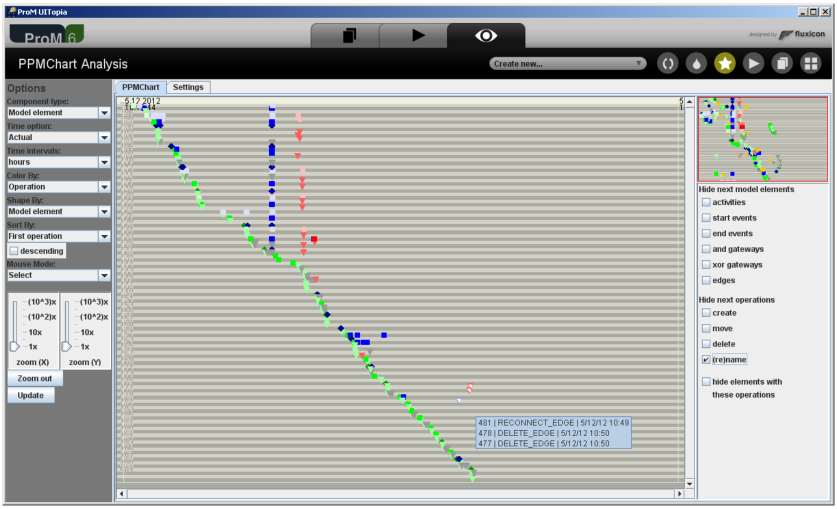


Figure A.6. Example of the first PPMChart created using the Dotted Chart implementation

B

Appendix B

Parameter settings of the PPMChart Analysis plug-in



Options

Component type: Model element

Time option: Actual

Time intervals: hours

Color By: Operation

Shape By: Model element

Sort By: First operation

☐ descending

Mouse Mode: Select

Hide next model elements
- ☐ activities
- ☐ start events
- ☐ end events
- ☐ and gateways
- ☐ xor gateways
- ☐ edges

Hide next operations
- ☐ create
- ☐ move
- ☐ delete
- ☑ (re)name

☐ hide elements with these operations

```
481 | RECONNECT_EDGE | 5/12/12 10:49
478 | DELETE_EDGE | 5/12/12 10:50
477 | DELETE_EDGE | 5/12/12 10:50
```

Figure B.1. Screenshot of the PPMChart window in ProM

Configuration

At the left hand side, the view can be configured (see Figure B.1).

The **Component type** indicates which dimension is used to define the unit of the timelines. In contrast to the Dotted Chart Analysis plug-in, this option cannot be configured. The fixed value for this option in the PPMChart implementation is:

- ***Model element (default)****: select this option to view a timeline per model element. Each dot on the timeline represents an operation on the model element represented by the timeline (e.g., create, move, (re)name, delete of a particular XOR gateway).*

The **Time option** can be configured to zoom in on the timing of the operations. Next three options can be selected:

- ***Actual (default):*** *select this option to view the dots positioned according to the real time of execution of the corresponding operation.*
- ***Relative (Time):*** *select this option to shift every time line in such a way that the first operation on each line is set to the beginning of the time interval of the PPMChart.*
- ***Relative(Ratio):*** *select this option to stretch every timeline in such a way that the first operation on each line is set to the beginning of the time interval and the last operation on each line is set to the end of the time interval (if at least two operations exist on the line).*

Vertical time intervals are marked according to the **Time intervals** configuration parameter. There are 13 different options.

- ***L-1, L-10, L-100, L-500:*** *select these options to divide the chart in time intervals of 1, 10, 100, or 500 milliseconds respectively. Time intervals are indicated with white vertical lines starting at the time of the first operation in the chart. It is necessary to zoom in on the chart to be able to analyze the chart at millisecond level.*
- ***Seconds, Minutes, Half hours, Hours (default):*** *select these options to divide the chart in time intervals of seconds, minutes, half hours, or hours respectively.*
- ***Days, Week, Months, Years:*** *select these options to divide the chart in time intervals of days, weeks, months, or years respectively. It is necessary to zoom out on the chart to be able to analyze the chart at a level greater than one hour.*

The option **Color by** indicates if the dots have to be color-coded or not. The PPMChart in principle uses a fixed default color coding (if turned on), but the colors can be changed by the user in the *Settings* tab (see below). Next two options can be selected:

- ***None:*** *select this option to remove color coding. Each dot will have the same color, which allows the user to focus on shape and position of the dots (in order to abstract from the type of operation).*
- ***Operation (default):*** *select this option to apply color coding. By default, create operations will be colored in green, move operations in blue, delete operations in red, and (re)naming in orange. A detailed legend of the default colors is displayed in Table B.1.*

Table B.1. Default color (shade) and shape coding of events

		Create	Move	Delete	(Re)name
		Green	Blue	Red	Orange
Start event	Circle ●	CREATE_START_EVENT Very light green circle	MOVE_START_EVENT Very light blue circle	DELETE_START_EVENT Very light red circle	NAME_ACTIVITY RENAME_ACTIVITY Orange square
End event	Circle ●	CREATE_END_EVENT Very light green circle	MOVE_END_EVENT Very light blue circle	DELETE_END_EVENT Very light red circle	
Activity	Square ■	CREATE_ACTIVITY Green square	MOVE_ACTIVITY Blue square	DELETE_ACTIVITY Red square	
XOR	Diamond ◆	CREATE_XOR Dark green diamond	MOVE_XOR Dark blue diamond	DELETE_XOR Dark red diamond	
AND	Diamond ◆	CREATE_AND Dark green diamond	MOVE_AND Dark blue diamond	DELETE_AND Dark red diamond	
Edge	Triangle ▼	CREATE_EDGE Light green triangle	MOVE_EDGE_LABEL Grey triangle	DELETE_EDGE Light red triangle	NAME_EDGE RENAME_EGE Orange triangle
		RECONNECT_EDGE Light purple triangle	CREATE_EDGE_BENDPOINT MOVE_EDGE_BENDPOINT DELETE_EDGE_BENDPOINT Dark grey triangle	RECONNECT_EDGE Light purple triangle	

Use the **Shape by** setting to configure if the dots have to be shape-coded or not. The PPMChart in principle uses a fixed default shape coding (if turned on), but the shapes can be changed by the user in the *Settings* tab (see A.3. below). Next two options can be selected:

- ***None:*** *select this option to turn off dot shaping. Each dot will be displayed as a circle, which allows the user to focus on color and position of dots (to abstract from the model element type of the operation).*
- ***Model element (default):*** *select this option to turn on dot shaping. Operations on activities will be displayed with rectangles, event operations with circles, gateway operations with diamonds, and edges with triangles. A detailed shape legend is displayed in Table B.1.*

Sort by can be used to influence the order in which the timelines are sorted (vertically). If *descending* is selected, the sort order is reversed. Next eight options can be selected:

- ***None:*** *select this option to select no ordering. The order of the data in the event log will be used.*
- ***Model element:*** *select this option to sort the lines by the model element identifier. The lines will be sorted according to the identifiers of the model elements represented by the timelines.*
- ***Number of operations:*** *select this option to sort the lines by the number of operations displayed on each line. Use this option to graphically observe differences between lines with fewer operations (top part of the chart if sorted according to this option) and lines with more operations (bottom part of the chart).*
- ***Duration:*** *select this option to sort the lines according to their duration. The duration is defined as the timespan between the first and the last operation on the line. This option allows to compare lines with shorter versus longer durations.*
- ***Distance from start (default):*** *select this option to sort the lines according to the traversing order of the corresponding model elements from the start event towards the end event (see description in Section 2.3.4).*
- ***Create order from start:*** *select this option to sort the lines according to the logical order of creation of the corresponding elements from start event to end event (see description in Section 2.3.4).*
- ***First operation:*** *select this option to sort the lines according to the time of the operation of the first dot on the line. This option facilitates to zoom in on the actual order of creation of model elements.*
- ***Last operation:*** *select this option to sort the lines according to the time of the operation represented by the last dot on the line. This option facilitates to zoom in on parts of the process model that are (not) touched towards the end of the modeling process.*

Configure the **Mouse mode** to set the way the mouse behaves in the plug-in. Next three options can be selected:

- ***Select (default):*** *select this option to be able to select different dots. Click on a dot or make a rectangular selection to indicate of which dots to display information in a tooltip.*
- ***Zoom in:*** *select this option to be able to easily zoom in on parts of the PPMChart. Make a rectangular selection on the screen to indicate the area you want to zoom in on.*

- ***Drag:*** *select this option to be able to bring a different area of the chart into the displayed rectangle if zoomed in. Drag the chart under the displayed rectangle to show other parts of the chart.*

The sliders **zoom (X)** and **zoom (Y)** can be used to zoom in horizontal or vertical dimension respectively on a logarithmical scale. The **Zoom out** button restores the zoom level to 1 x 1. The **Update** button needs to be pressed after changing one or more of previous options before the PPMChart is repainted on the screen.

Filtering

At the right-hand side the user can customize the view by filtering on specific operations or model elements (see Figure 2.5). The top part represents a small view on the unfiltered PPMChart. Below, one can configure next three filter options:

- ***Hide next model elements:*** *choose to hide specific element types (e.g., hide edges). All dots that represent operations on an element of the selected type are removed from the chart. However, no timelines are removed. This might result in a PPMChart with a number of empty timelines (i.e., without any dot on the line).*
- ***Hide next operations:*** *choose to hide specific operation types (e.g., hide (re)name operations). All dots that represent operations of the selected types are removed from the chart. Again, only dots are removed from the chart, not timelines. Empty timelines may originate from this option if the model element represented by the timeline has only operations that are selected to be hidden.*
- ***Hide all elements with these operations:*** *hide elements with a specific operation (e.g., hide deleted elements). All dots that represent any operation on a model element that contains at least one operation of the selected operation type are removed from the chart. Again, only dots are removed from the chart, not timelines.*

Settings

Use the **Settings** tab page to change the color and shape coding of elements. Simply click on the button to change the color or shape for the corresponding operation.

C

Appendix C
Method of pattern detection with PPMCharts

Section 2.4 lists a selection of patterns that was discovered in the PPMCharts of various modeling executions. This appendix briefly explains the method that was used to detect these patterns.

As described in Section 2.4.1, data was collected about students constructing two process models each. At first we used three datasets, i.e., the two datasets described in Section 2.4.1 and a set containing data from 14 modelers with professional modeling experience. Later, we decided to drop the expert set from the analysis because we had the impression that there were too many anomalies in this dataset. For example, two modelers found the experimental tool confusing and messed up their first modeling assignment, for which we thus have no data. Potentially, this was caused by the Expertise Reversal Effect discussed on page 142.

PPMCharts were generated for every modeling execution in the three datasets. Then, a random sample of 14 participants of both student sessions was drawn in order to compare an equal amount of charts for each of the three datasets. The PPMCharts for the two modeling executions of each participant in the sample were put on a wall in random order (but grouped per dataset).

Next, we organized a brainstorm session to try to discover as much patterns as possible, which resulted in the set of observations presented in section 2.4 (see Figure C.1). The three researchers performing this brainstorm - Irene Vanderfeesten, Hajo Reijers, and Jan Claes - were familiar with all the details of the data collection process (i.e., characteristics of the participants, tasks performed by the participants, organization of the sessions, etc.) Therefore, they possessed all the information needed to correctly interpret the PPMCharts.

Figure C.1. Irene Vanderfeesten (left) and Jan Claes (right) discovering patterns in PPMCharts

D

Appendix D
Modeling language and recorded operations in CEP

The modeling language of the process modeling editor in the Cheetah Experimental Platform (CEP) consists of activities, AND gateways and XOR gateways that can be used as parallel and exclusive split or join constructs, start events and end events and sequence flows (see Palette at the right in Figure D.1). In order to avoid learning difficulties and the Expertise Reversal effect (see Section 5.3), the modeling language was designed with the aim to limit the amount of constructs to the minimum. The provided constructs are the necessary and most used constructs for control flow modeling (Marcello La Rosa et al., 2011; Zur Muehlen & Recker, 2008). The used symbols are the symbols of the OMG process modeling standard BPMN (i.e., rectangles, diamonds with + or x annotation, circles and arrows). Because the participants were familiar with the syntax of BPMN, no further explicit explanation was given about the modeling language that the participants had to use. We assumed that the legend in the tool and the completion of the tutorial (see below) sufficed to properly introduce the modeling language to the participants.

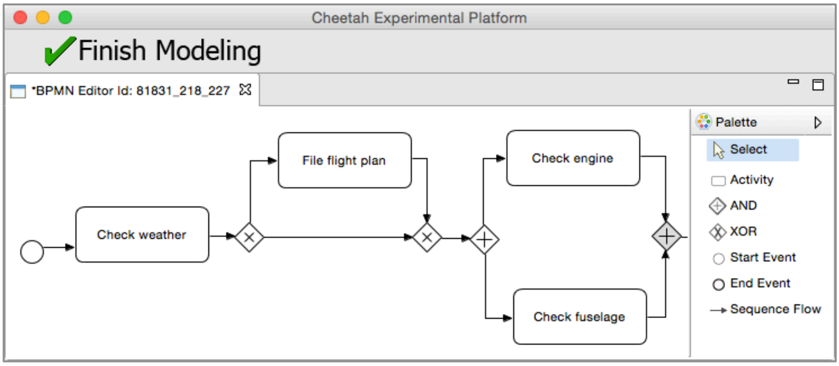
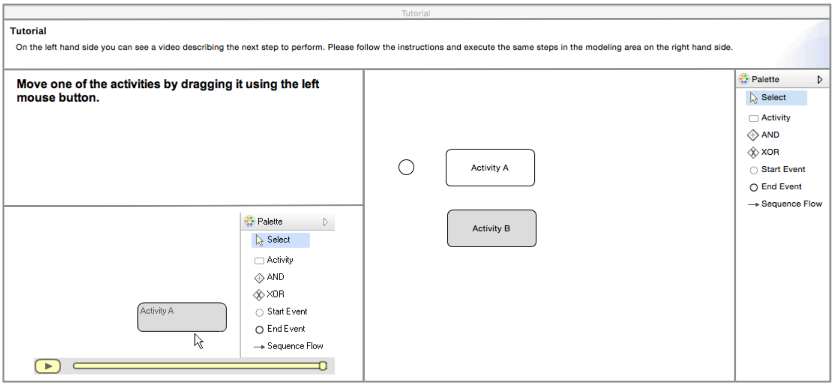


Figure D.1. Process modeling editor of Cheetah Experimental Platform

Before participants started the modeling assignment of the observational sessions, they had to complete the editor tutorial to make sure that they sufficiently understood how to use the editor. In this tutorial they had to mimic the modeling actions that were described in text and that were displayed in short videos (see Figure D.2). Only when the user correctly copied the presented modeling action, the tutorial continued with the next explanation.



Tutorial

On the left hand side you can see a video describing the next step to perform. Please follow the instructions and execute the same steps in the modeling area on the right hand side.

Move one of the activities by dragging it using the left mouse button.

Figure D.2. Editor tutorial of Cheetah Experimental Platform

During modeling, CEP records every interaction with the tool. Figure D.3 displays a list that contains the subset of the recorded operations that are represented in the PPMChart visualization, together with the used symbols in the visualization. The events that are ignored in the PPMChart have to do with selection of model elements and scrolling to change the part of the model that is currently visible on the modeling canvas.

Figure D.3. Model interactions recorded in CEP and visualized in the PPMChart

E

Appendix E

Determining ambiguity and *'mistakes'* in a process model

In Chapter 3 we defined a perspicuous process model as a model that is unambiguously interpretable and can be made sound with only small adaptations based on minimal assumptions on the modeler's intentions with the model (see Definition 9). For the measurement of perspicuity, the process model is transformed into a workflow net (see Section 3.3.3). This transformation takes the assumed intentions of the modeler into account.

The intentions of the modeler are derived from the origin or destination of parallel or exclusive paths. When multiple paths are joined in an implicit manner (i.e., without gateway construct) and they originate in a single split gateway, the intention of the modeler is derived from that split gateway (i.e., the sign of the split gateway is used as indicator of the modeler's intention when joining the paths). Conversely, when multiple paths are split in an implicit manner and they merge in a single join gateway, the intention of the modeler is derived from that join gateway.

In order to determine perspicuity of models, the authors marked the syntactical issues[32] in the models. Table E.1 lists the observed issues and their classification in issues that do or do not influence perspicuity. They correspond to the transformation rules described in Section 3.3.3.

---

[32] We use the term 'issue', because not every listed observation is a syntactical error.

Table E.1. Observed syntactic issues in the observational modeling sessions

Syntactic issue	Type
Contains activities or split gateways without incoming arc	No influence
Contains activities or join gateways without outgoing arc	No influence
Contains multiple start events starting paths that are joined correctly or in a single construct	No influence
Contains multiple start events starting paths that are joined incorrectly in multiple constructs	Influence
Contains multiple end events ending paths that are splitted correctly or at a single construct	No influence
Contains multiple end events ending paths that are splitted incorrectly at multiple constructs	Influence
Contains activities with multiple outgoing arcs that are joined correctly or in a single construct	No influence
Contains activities with multiple outgoing arcs that are joined incorrectly in multiple constructs	Influence
Contains activities with multiple incoming arcs that are splitted correctly or at a single construct	No influence
Contains activities with multiple incoming arcs that are splitted incorrectly at multiple constructs	Influence
Contains gateways with multuple incoming and outgoing arcs	Influence

Considering the way we determined in Chapter 3 the assumed intentions of the modeler, we realized that syntactical errors (whether they influence perspicuity or not) could be caused by two different factors. So in Chapter 4 we make a distinction between errors that originate in a lack of knowledge of the modeling syntax by the modeler, and errors that are caused by cognitive failure of the modeler. In the latter case the modeler knows the right way of modeling, but due to a lack of cognitive resources (i.e., lack of working memory capacity or attention), he unintentionally introduces an error into the model. We called this type of errors *'mistakes'* (see Section 4.3.2).

Below, we explain how we operationalized the measurements of the amount of *'mistakes'*, i.e. the amount of errors with a cognitive origin. Most errors were considered to be potentially caused by cognitive failure, whereas the *consistent* absence of (a certain kind of) gateways was assumed to originate in a lack of knowledge of the modeling language (i.e., it is assumed the modeler did not know a gateway is needed in those situations). Table E.2 lists the observed syntactic errors in the produced models of the observational modeling sessions, together with the used codes and their classification in knowledge-related (K) and cognitive-related errors (C). When the type is C/K, the final classification depends on whether the error was made consistently (K) throughout the model or not (C).

Table E.2. Observed syntactic errors in the observational modeling sessions

Syntactic error	**Code**	**Type**
Contains no end event (but does contain a start event)	0	C
Contains an end event in the middle (outgoing edge on end event)	B	C/K
Some, but not all of the paths are not closed (missing end event?)	P	C
Contains no split gateways at all	S	K
Contains no XOR split gateways at all	Sxor	K
Contains no AND split gateways at all	Sand	K
Forgot some, but not all split gateways	F	C
Forgot an XOR split gateway	Fxor	C/K
Forgot an AND split gateway	Fand	C/K
Contains no join gateways at all	J	K
Contains no XOR join gateways at all	Jxor	K
Contains no AND join gateways at all	Jand	K
Forgot some, but not all join gateways	G	C
Forgot an XOR join gateway	Gxor	C/K
Forgot an AND join gateway	Gand	C/K
One gateway combines a join and split feature	C	C/K
Wrong type of join combined with a certain split	W	C
Gateway with only one ingoing and one outgoing edge	1	C
Wrong nesting of gateways	N	C
AND and XOR are joined together in one join gateway	T	C

C = Cognitive, K = Knowledge ↑

First, in order to avoid overseeing errors, the syntactical errors in the models were marked and coded independently by three people. Next, the authors combined the three assessments into one, more complete list of errors per model. When an error occurred multiple times, also the number of occurrences was noted.

For a particular process model, the full code could for example look like this: Jxor(3C + 2Gxor) + P. This modeler used three gateways that combined split and join semantics, did not use a gateway to join paths of two XOR splits and had one activity without outgoing arc. Because the modeler never used join gateways (Jxor), the combined gateways (C) and missing XOR gateways (Gxor) are considered knowledge-related errors and so we only counted the missing outgoing arc (P) as a *'mistake'*.

F

Appendix F

Observational modeling sessions: questionnaire

1. What is your gender? (M/F)
2. How old are you?
3. What is your native language?
4. What is your current profession? (Student/Other)
   If you have selected 'Other' please specify which profession.
5. How many years ago did you start process modeling?
6. How many process models have you analyzed or read within the last 12 months? (A year has about 250 workdays. In case you read one model per day, this would sum up to 250 models per year)
7. How many process models have you created or edited within the last 12 months?
8. How many activities did all these models have on average?
9. How many workdays of formal training on process modeling have you received within the last 12 months? (This includes e.g. university lectures, certification courses, training courses. 15 weeks of a 90 minutes university lecture is roughly 3 work days)
10. How many workdays of self-education have you made within the last 12 months? (This includes e.g. learning-by-doing, learning-on-the-fly, self-study of textbooks or specifications)
11. Which education program are you following?
    (OML/BIS/IM/CSE/Other)
12. If you have selected 'Other' please specify which education program.

13. Which process modeling languages have you used before?
(Aris (express) / BPEL / BPMN / BPM|one / Petri Nets-Colored Petri Nets-CPN Tools / Tibco-COSA / Workflow Nets-WoPeD / Other)
14. If you have selected 'Other' please specify which modeling languages.
15. I consider myself being a process modeling expert. (*)
16. I have troubles reading English texts. (*)
17. I have troubles understanding English text. (*)
18. I am familiar with processes in the financial domain. (*)
19. I am familiar with mortgage approval processes (*)
20. I have created process models in the financial domain before. (*)
21. I have created process models for a mortgage process before. (*)
22. Overall, I am familiar with the BPMN modeling language. (*)
23. I feel confident in understanding process models created with the BPMN modeling language. (*)
24. I feel competent in using the BPMN modeling language for process modeling. (*)
25. How many months ago did you start using BPMN? (The first version of BPMN stems from May 2004, i.e. 60 months until May 2009)
26. Have you completed the assignment? (yes/no)
27. How would you assess the mental effort for completing the modeling task? (1-10)
28. I have experienced language difficulties when reading and understanding the case description. (*)
29. I found the case description easy to understand. (*)
30. I found the case description complex and difficult to follow. (*)
31. The case description was ambiguous. (*)
32. The case was clearly described. (*)
33. Have you experienced any disturbances during the execution of the modeling task? (yes/no)
34. If yes, which (kind of) disturbances have you experienced?
35. Learning to use the BPMN modeling language is easy for me. (*)
36. I find it easy to get the BPMN modeling language to do what I want it to do. (*)
37. My interaction with the BPMN modeling language is clear and understandable. (*)
38. I find the BPMN modeling language to be flexible to interact with. (*)
39. It would be easy for me to become skillful at using the BPMN modeling language. (*)
40. I find the BPMN modeling language easy to use. (*)
41. Learning to operate the modeling tool is easy for me. (*)
42. I find it easy to get the modeling tool to do what I want it to do. (*)
43. My interaction with the modeling tool is clear and understandable. (*)
44. I find the modeling tool to be flexible to interact with. (*)
45. It would be easy for me to become skillful at using the modeling tool. (*)
46. I find the modeling tool easy to use. (*)

(*) This item had to be scored on a 7-point Likert scale
(strongly agree – agree – somewhat agree – neutral – somewhat disagree – disagree – strongly disagree)

G

Appendix G

Prior knowledge and modeling experience

Figure G.1 and Figure G.2 show information about the prior knowledge and modeling experience of the participants of the observational modeling session described in Section 4.3.

Prior domain knowledge (PDK)

- *I am familiar with processes in the financial domain. (PDK01)*
- *I am familiar with mortgage approval processes (PDK02)*
- *I have created process models in the financial domain before. (PDK03)*
- *I have created process models for a mortgage process before. (PDK04)*

Prior modeling knowledge (PMK)

- *Overall, I am familiar with the BPMN modeling language. (PMK01)*
- *I feel confident in understanding process models created with the BPMN modeling language. (PMK02)*
- *I feel competent in using the BPMN modeling language for process modeling. (PMK03)*

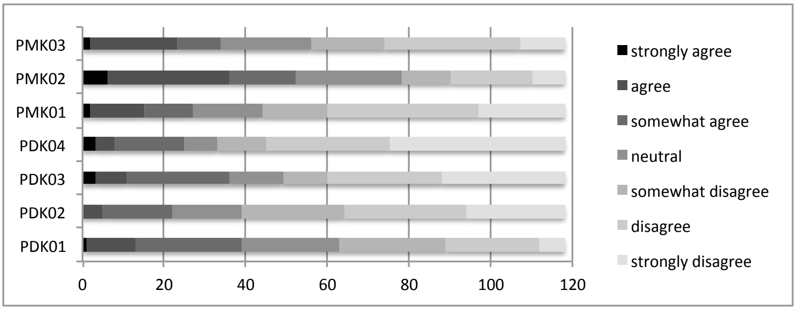


Figure G.1. Indicated prior domain knowledge (PDK) and prior modeling knowledge (PMK)

Modeling experience (ME)

- *How many years ago did you start process modeling? (ME01)*
- *How many process models have you analyzed or read within the last 12 months? (A year has about 250 workdays. In case you read one model per day, this would sum up to 250 models per year) (ME02)*
- *How many process models have you created or edited within the last 12 months? (ME03)*
- *How many activities did all these models have on average? (ME04)*
- *How many workdays of formal training on process modeling have you received within the last 12 months? (This includes e.g. university lectures, certification courses, training courses. 15 weeks of a 90 minutes university lecture is roughly 3 work days) (ME05)*
- *How many workdays of self-education have you made within the last 12 months? (This includes e.g. learning-by-doing, learning-on-the-fly, self-study of textbooks or specifications) (ME06)*
- *How many months ago did you start using BPMN? (ME07)*

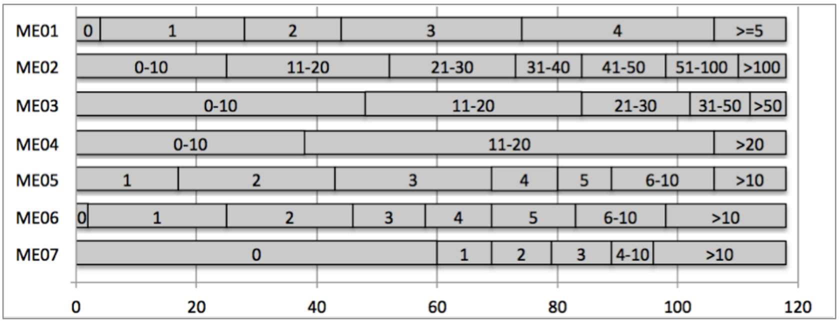


Figure G.2. Indicated modeling experience (ME)

H

Appendix H
Data collection overview

This appendix contains an overview of the data collection sessions related to the doctoral research. First, the experimental modeling tool is presented. Next, a table summarizes the properties of each performed data collection session. Finally, we provide an overview of the modeling cases.

Cheetah Experimental Platform (CEP)

The data collections sessions were all performed with the support of Cheetah Experimental Platform (CEP). CEP was developed as an open source tool by the University of Innsbruck to support data collection for Process of Process Modeling research (Pinggera, Zugal, & Weber, 2010). An experimental flow of tasks can be defined, which the participants have to execute. Typical tasks in an experimental workflow are a tool-tutorial task, a process modeling task, and a survey task. They are included in the tool. The process modeling task is supported by a basic process modeling editor with limited constructs and BPMN-like symbols (see Figure H.1). The editor logs every operation of the modeler in the tool, which can be used afterwards to replay (parts of) the modeling or to use process-mining techniques to analyze the data in the event log.

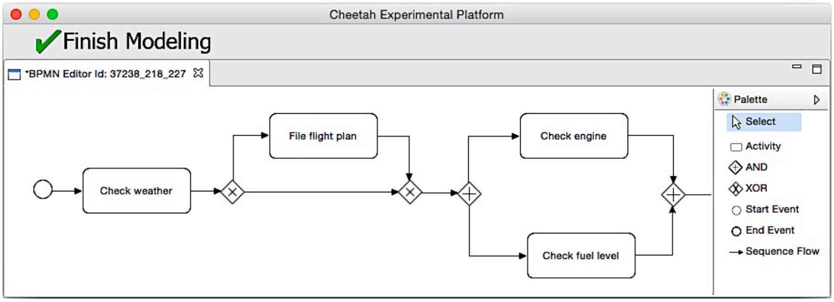


Figure H.1. Process modeling editor of the Cheetah Experimental Platform

Data collection sessions

November 2010 – Eindhoven University of Technology

Initiative	Eindhoven University of Technology University of Innsbruck
Goal	General observational modeling session
Participants	120 master students from one of these programs - Operations Management & Logistics - Innovation Management - Business Information Systems
Tasks	1. Tool tutorial 2. Pre-flight case 3. NFL case 4. Survey
Used in	Chapter 2 & 3

December 2010 – Humboldt University of Berlin

Initiative	Humboldt University of Berlin University of Innsbruck
Goal	General observational modeling session
Participants	14 master students
Tasks	1. Tool tutorial 2. Pre-flight case 3. NFL case 4. Survey
Used in	Chapter 3

June 2011 – Eindhoven University of Technology

Initiative	Eindhoven University of Technology University of Innsbruck
Goal	General observational modeling session
Participants	14 process modeling practitioners (3-10 years of process modeling experience)
Tasks	1. Tool tutorial 2. Pre-flight case 3. NFL case 4. Survey
Used in	Chapter 3

December 2012 – Eindhoven University of Technology

Initiative	Eindhoven University of Technology
Goal	Observational modeling session, focus on expertise
Participants	118 master students from one of these programs - Operations Management & Logistics - Innovation Management - Business Information Systems
Tasks	1. Tool tutorial 2. Mortgage case 3. Survey
Used in	Chapter 2 & 4

March 2013 (2 parts) – Ghent University

Initiative	Ghent University
Goal	Observational modeling session (part 1) Experiment to test first draft of SPMM (part 1 + part 2)
Participants	146 master students in Business Engineering
Tasks	1.1. Survey 1 1.2. Tool tutorial 1.3. Visa case 1.4. Survey 2 2.1. Technique tutorial 2.2. Fines case 2.3. Survey 3
Used in	Chapter 4

Cases

Throughout the different data collection sessions, participants had to construct a process model based on a given case description. Most cases were used in multiple sessions. A brief overview of the cases is presented here.

Pre-flight case

This case describes the process executed before an aircraft can take off. (Download from http://bpm.q-e.at/experiment/Pre-Flight)

NFL case

This case describes the process of the scouting department of the National Football League to acquire new players.
(Download from http://bpm.q-e.at/experiment/Pre-Flight)

Mortgage case

This case describes the handling of a mortgage request by a bank.
(Download from http://bpm.q-e.at/experiment/MortgageEindhoven)

Visa case

This case describes the process of visa granting at Australia's airports. (Download from http://www.janclaes.info/experiments/2015SPMM)

Fines case

This case describes the defaulter handling of fines by Dutch government. (Download from http://www.janclaes.info/experiments/2015SPMM)