1st International Workshop on

Empirical Research in Process-Oriented Information Systems

ER-POIS 2010

Hammamet, Tunisia, 8 June 2010

In conjunction with the CAISE'10
22nd International Conference on Advanced Information Systems Engineering

Workshop Proceedings

Editors

ER-POIS 2010 Chairs

Bela Mutschler
University of Applied Sciences Ravensburg-Weingarten, Germany

Jan Recker
Queensland University of Technology, Australia

Roel Wieringa
University of Twente, The Netherlands

CAiSE'10 Workshop Chairs

Jolita Ralyté
University of Geneva, Switzerland

Pierluigi Plebani
Politecnico di Milano, Itally

CAiSE 2010 Workshop ER-POIS

Proceedings

This volume contains the original articles presented at the 1st International Workshop on Empirical Research in Process-Oriented Information Systems – ER-POIS'10. The workshop was held in conjunction with the 22nd International Conference on Advanced Information Systems Engineering, in Hammamet, Tunisia, June 8, 2010.

CEUR Workshop Proceedings, CEUR-WS.org, ISSN 1613-0073.

Preface

Since the advent of business process reengineering in the 1990s, modeling and optimization of business processes has become a critical factor in business as well as in non-profit and government organizations. Today, with many forms of interorganizational cooperation in the form of e-business, outsourcing, and other kinds of value webs, business process management and coordination process management are crucial for the survival of organizations. As a result, information systems are required to manipulate not only data but also processes. *Process-oriented information systems* (POIS) are systems that can provide data support as well as process support to organizations. Examples are workflow management systems, ERP systems, and systems for cross-organizational coordination management. Some of the challenges facing POIS include requirements management, performance monitoring, service level agreements and governance. The ***1st International Workshop on Empirical Research in Process-Oriented Information Systems*** (ER-POIS 2010) aims to provide a platform for exchanging empirical research results about POIS problems and solutions.

April 2010

Bela Mutschler
Jan Recker
Roel Wieringa

Workshop Organization

Workshop Co-Chairs

Bela Mutschler
University of Applied Sciences Ravensburg-Weingarten, Germany
Jan Recker
Queensland University of Technology, Australia
Roel Wieringa
University of Twente, The Netherlands

Program Committee

Richard Baskerville, USA
Sjaak Brinkkemper, The Netherlands
Eric Dubois, Luxemburg
Marta Indulska, Australia
Matthias Jarke, Germany
Pontus Johnson, Sweden
Kalle Lyytinen, USA
Jan Mendling, Germany
Bela Mutschler, Germany
Andreas Opdahl, Norway
Jinsoo Park, Korea
Oscar Pastor, Spain
Yves Peigneur, Switzerland
Jan Recker, Australia
Manfred Reichert, Germany
Hajo Reijers, The Netherlands
Stefanie Rinderle-Ma, Austria
Matti Rossi, Finland
Marcus Rothenberger, USA
Camille Salinesi, France
Keng Siau, USA
Roel Wieringa, The Netherlands
Barbara Weber, Austria

Table of Contents

Process Owners in the Wild: Findings from a Multi-method Descriptive Study

Hajo A. Reijers[1] and Roel L.J.L Peeters[2]

[1]Eindhoven University of Technology, School of Industrial Engineering,
Den Dolech 2, 5600 MB Eindhoven, The Netherlands
h.a.reijers@tue.nl
[2]YNNO, Orteliuslaan 9, 3528 BA Utrecht, The Netherlands
roel.peeters@ynno.com

Abstract. Process ownership is widely considered as a key element in process-oriented organizations. However, no consistent view on this role can be found in the literature and only a limited insight exists into its fulfillment within industrial practice. This paper reports on the findings from a descriptive research study into process ownership. These findings are gathered through a survey and two in-depth case studies. A main conclusion is that tasks and responsibilities of process owners have a different focus for organizations at an early stage of BPM maturity compared to organizations at more progressed levels. Furthermore, the formal and actual fulfillment of this role may vary considerably. In this paper, we reflect on the implications of these findings for practice and research.

Keywords: Process ownership, process roles, BPM, BPM maturity

1 Introduction

The most visible difference between a process enterprise and a traditional organization is arguably the existence of *process owners* [8]. In a traditional organization, a geographical or functional manager oversees both the operations and the people performing them. In a process-oriented organization, it is the process owner who is responsible for the effective and efficient execution of a process [20].

Process ownership is broadly recognized as a crucial element in the effectiveness of process-oriented organizations. For example, Hammer describes a case of a newly designed order-fulfillment process where "the process owner didn't have the authority to force unit heads to implement it, so the effort floundered" [6]. Rummler and Brache refer to the interfaces between functional departments as "white spaces" and simply state that "without a process owner, the white spaces will be ignored" [18]. Also, in current frameworks for assessing Business Process Management (BPM) maturity of organizations, the existence of process owners and other BPM-related roles are considered as a major element of the governance structure [6;15;16].

Given the importance attributed to process ownership, actually *very little* is known about what process owners do or are supposed to do. There is consensus about

B. Mutschler, J. Recker, R. Wieringa, J. Ralyté, and P. Plebani (Eds.):
CAiSE 2010 Workshop ER-POIS, Hammamet, Tunisia, pp. 1-12, 2010.

process owners being responsible for the management of processes across the various phases of its lifecycle (see e.g. [2;8;20]), but little beyond that. It is the aim of this paper to shed light on these and other issues that relate to how process owners operate in *industrial practice*. The contribution of this paper is that it provides a contemporary, descriptive view on process ownership. This view may help organizations to reflect on their process ownership fulfillment. Our work may be used as the academic starting point for creating and supporting an informed, uniform, and prescriptive view on this subject.

To gather the insights we desire, we conducted both a survey and two case studies in The Netherlands with a *descriptive* research design. An important principle that guided our design is that process ownership is probably not a static role. In one of the few studies that empirically investigates process ownership, it was established that organizations that are well progressed with BPM or, in other words, display a high level of BPM maturity (BPMM) appoint a higher proportion of process owners compared to organizations in early stages of BPM adoption [14]. Also, but then from a prescriptive point of view, it has been argued that the role of the process owner *must* change as the organization's BPM initiative matures [5]. Therefore, our investigation of process ownership goes hand-in-hand with determining the maturity that organizations display with respect to their BPM initiatives.

The structure of the paper is then as follows. In the next section we will provide a review of the literature on process ownership. The review will particularly highlight the omissions in the existing body of knowledge. Next, we discuss in Section 3 our research design, followed by a presentation of the results from the survey and case studies in Section 4. Section 5 contains a discussion of the results and our conclusions.

2 Literature review

Process owners are broadly recognized as being important in a process-oriented approach to manage business operations. Hammer describes no less than four phases of process ownership in his tool to assess the maturity of business processes and enterprises [6]. Similarly, process owners – next to other process-related roles – are also considered as part of the governance structure in a recent BPM maturity model [16;17]. It should be noted, however, that support for the importance of process ownership is mostly anecdotic, as in e.g. [8].

The only empirical work known to us that considers the process owner role both methodologically and empirically is [14]. The authors' main insight is that organizations that are well progressed with BPM appoint a higher portion of process owners and that these are more often to be found at both a senior level and supervisory / frontline level than is the case for early stage organizations. It should be mentioned that the survey at the basis of these findings was carried out in 1996, more than a decade ago. That there *is* a notable uncertainty on process ownership in industrial practice was reported more recently in [10]. As the authors put it "often there is no explicit or implicit agreement of process ownership [within an organization]".

Except that process owners are considered important, little consensus exists on what process owners really are. Hammer & Champy [7] use the notion to identify the person responsible for the reengineering of a specific process, including establishing the standards of performance. As summarized in [10], Hammer *cum suis* state that the process owner must have the end-to-end accountability for a process. Rather than Hammer's powerful technocrat, Siemieniuch and Sinclair see process owners implementing an *administrative function* with as main responsibilities and tasks the documentation of a process and the evaluation and approval of process changes [19]. A more reflective, observing interpretation of the process owner role can also be found in [20]. Hardjono and Bakker describe a more elaborate role for the process owners as fulfilling a *management* and *control* function [9]. But in contrast to Hammer's strong emphasis on the reengineering phase of a process, Hardjono and Bakker clearly link the process owner to *all* phases of the process lifecycle. Furthermore, they argue that process ownership should be assigned as *low* in the organization as possible, to encourage 'a spirit of entrepreneurship'. The additional benefit of this level is that this is beneficial for the organization's customer focus. Clearly, this sharply contrasts with the examples provided by Hammer in [6] and [8] where process owners are senior managers. Hardjono and Bakker's view is also not consistent with the evolving nature of process owners in [14].

The only attempt that we are aware of to arrive at a prescriptive view on the tasks and responsibilities of a process owner from a more or less methodological requirements analysis is given in [10]. However, this attempt specifically aims at the situation where process ownership is implemented in an *inter-organizational* context and the role is closely linked to the implementation of IT.

In summary, the literature provides limited insight into process ownership. Firstly, in most publications where process ownership is addressed, the topic is treated superficially – almost as if authors take the subject for granted. Secondly, only little empirical research has been conducted in this area. Thirdly, most articles in which process ownership is touched are prescriptive in nature, but are in disagreement in many respects. This leads to a situation in which little consensus exists on the *preferable* fulfillment of process ownership and no insight at all into how organizations *actually* implement this role.

3 Research design

The main research question that we address with this work is: How is process ownership fulfilled in practice across different levels of BPM maturity? This question is considered at two levels of abstraction. In the first place, we are interested in the *organizational* level. At this level we consider how organizations formally give shape to the role of the process owner. Secondly, we consider process ownership at the *individual* level: How do process owners really act, either within or perhaps beyond the formal limits of their role?

Since the concept of process ownership is well-entrenched in the managerial discourse without, however, an exact understanding of its specific aspects, we addressed our research question with a *descriptive* design. Two complementing data collection methods were selected: a *survey*, which aims at gathering quantitative data, and two in-depth *case studies*, aimed at gaining a qualitative insight. A multi-method model of research like this one is not common in IS research, although the case for combining qualitative and quantitative research methods is strong [4]. In the application of both research methods the two concepts of interest are addressed, i.e. process ownership and BPM maturity (see Figure 1).

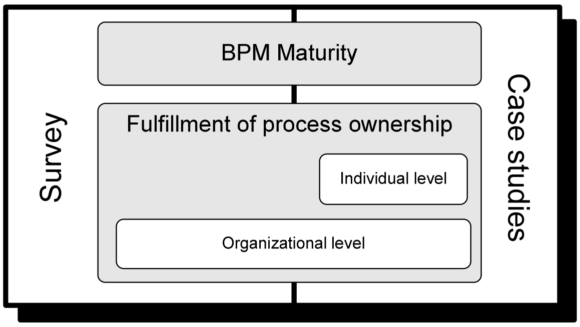


Figure 1: Multi-method approach.

In order to ensure comparable data was gathered in the research, a theoretical framework has been developed on the basis of literature on job (performance) analysis. The framework is based on Boyatzis' dynamic interaction model [1] and amended with BPM-related contextual factors to get insight into both the organizational implementation and individual fulfillment of process ownership. For more details, see Table 1.

Table 1: Variables of the analysis framework.

Organizational Environment	Organizational demands / actual fulfillment	Competences
Organizational demographics	Hierarchical position	Characteristics
Organizational strategy	Ownership hierarchy	Capacities
BPM drivers / strategy	Full-time / part-time	
BPM structure	Responsibilities	
BPM maturity	Authorities	
BPM governance	Tasks	

A further specification was made for the content-related aspects in the category 'organizational demands / actual fulfillment': responsibilities, authorities and tasks (see the second column in Table 1). Our review of the literature revealed two main *responsibilities*: responsibility (1) for process performance and (2) for process improvement. Also, the following *authorities* are taken into account: (1) decision-rights on e.g. process design [8], (2) establishing or distributing budget for realizing process goals and process improvement [8], (3) assigning capacity to e.g. process improvement projects, and (4) access to management information. To apply a structural approach investigating process owners' *tasks*, the managerial practices from [22] were extended with tasks assigned by Mintzberg's to his 'figurehead' and 'entrepreneur' role [11].

3.1 Survey

The general purpose of a descriptive survey is to find out what situations, events, attitudes, or opinions are occurring in a population [13]. For this research, our interest is with a particular situation: the implementation of the process owner role in a specific organization. Because of the necessary precaution with limiting the size of a survey to minimize non-response, we only addressed process ownership at the *organizational* level in our survey. For the same reason, we did not use one of the instruments available to assess an organization's BPM maturity level, as they are rather extensive. Instead, we used De Bruin's *BPM structure* variable as a proxy for BPM maturity since it appears to provide a reasonable cluster to distinguish an organization's evolution with respect to BPM over time [3]. This variable distinguishes whether an organization carries out BPM initiatives either in an *ad hoc* fashion, in the form of projects, coordinated from a BPM team or centre of excellence, or within the setting of an enterprise-wide program.

Overall, the descriptive survey contained four sections. The first section contained a statement of confidentiality and anonymity (to encourage truthful responses and minimize non-response) and an explanation of important concepts, such as BPM and BPM maturity. The second section queried the respondent for several details, such as the name of the organization and its type of industry. The third section addressed the organization's type of BPM undertaking and structure, while the last section dealt with the organizational implementation of process ownership. The exact aspects addressed in the survey can be seen in the second column in Table 1.

The survey was distributed at three professional conferences on process management in The Netherlands during 2007 and 2008 and made available online to industry contacts of the authors. In total, approximately 130 individuals were invited to fill in the survey, which resulted in 56 complete responses. From these we filtered out the responses from consultancy firms. Their answers generally did not refer to a specific organization, which made them unreliable for our purposes. Also, we filtered out organizations without any BPM initiative. As a result, we arrived at 22 responses.

3.2 Case studies

In contrast to a survey, case studies provide the opportunity to get an in-depth, qualitative insight of the subject of study and allow for retaining the holistic and meaningful characteristics of real-life events. Following [4], case studies can aid in capturing the richness of organizational behavior. For this research project this means investigating both the *organizational* fulfillment of the process owner role and the *individual* fulfillment of process ownership in the case setting. The organizational fulfillment is addressed similarly as in the survey, but as we recall from Figure 1 the individual fulfillment is only addressed in the case studies.

The methodology applied in the case studies follows the case study method as described by [21]. Potential case organizations were targeted via personal interaction at three professional conferences on process management in The Netherlands. Several criteria for participation were determined: (a) the organization needed to undertake some form of BPM initiative or program, (b) the organization needed to have process owners, and (c) the organization allowed for the application of our entire theoretical framework to ensure comparable data collection. From the three organizations willing to participate, two met all criteria; the study of these cases is further described in this paper. Note that since the exact approaches to BPM are of strategic value to both organizations, they are made anonymous in this paper.

Within the case studies, three sources of data were used: (1) documentation; (2) an online survey, and (3) semi-structured interviews. The desk research was primarily aimed at gaining insight into the organizational environment and the organizational fulfillment of the process owner role. The survey was applied as a first investigation of the individual fulfillment of the process owner role and for assessing the organization's level of BPM maturity. For triangulation purposes, the latter topics were also covered in the semi-structured interviews.

4 Results

4.1 Survey

With our descriptive survey we investigated the *organizational* implementation of the process owner role across various levels of BPM maturity. As the survey outcomes support the research of [14] – where it was found that organizations already progressed with BPM fulfill process ownership differently than those in the early stages of BPM – the outcomes are presented for each of these two groups separately. To this end, we combined responses from organizations at the two lowest levels of the BPM structure variable into an "early stage" group and the remaining organizations into a "progressed stage" group. Note that the number of responses did not allow for a further split-up into groups. Also note that the presented outcomes are the scores on the elements in the second column of Table 1. The following are the findings that are *independent* of the organizational level of BPM maturity:

a) The vast majority (77%) of organizations with a BPM program assign process owners, implicating that this role is a common aspect in BPM.
b) Three-quarters of the respondents report that their BPM program can best be described as either 'ad hoc' or 'project-based' – the two early stages of BPM maturity adoption – whereas one-quarter describes program as more progressed (BPM team / enterprise-wide). This points at a rather immature BPM landscape in The Netherlands.

Table 2: Comparison of survey outcomes per level of BPM maturity

BPM maturity level	**Early Stage (n=16)**	**Progressed stage (n=6)**
Process owner assignment		
Not assigned	31%	0%
Part-time occupation	69%	67%
Full-time occupation	0%	33%
Organizational level		
Ownership in the board	6%	50%
Ownership in staff	38%	33%
Ownership in line management	63%	50%
Process owners assigned on multiple levels (hierarchy)	25%	50%
Responsibilities		
Process performance	75%	67%
Process improvement	50%	67%
None of these responsibilities	6%	33%
Authorities		
Decision rights	63%	83%
Budget	38%	33%
Capacity	56%	33%
Access to management information	50%	50%

Comparing the survey outcomes of both groups, several differences come to light. In both groups, approximately two-thirds of the organizations *assign* process ownership as a part-time role. However, whereas the remaining one-third in the early stage group concerns organizations that have not assigned process owners at all, the remaining one-third in the progressed group has full-time process owners. This implies that progressed organizations recognize the added value of and necessity for process owners.

Comparing the organizational levels on which process owners operate, two differences can be distinguished. Firstly, only a quarter of the early stage organizations has process owners on multiple levels in the organization, versus half of the progressed organizations. This indicates that process ownership is more common and elaborate in progressed organizations as part of a process-based governance structure. Secondly, there is a big difference of in the assignment of process owners on a board-level in favor of the progressed organizations. This finding supports earlier research: It is in line with [3], where support was found for the importance of *executive ownership / commitment*. It also supports the research of [14], where it was found that process owners are more often positioned at the executive level in progressed organizations.

Only half of the early stage organizations report that process owners have a *responsibility* for process improvement versus two-third of the organizations in the second category. It seems that the progressed organizations have their business processes under control and shift their focus towards process improvement. Another substantial difference concerning responsibilities is that one-third of the respondents in the progressed group indicates that process owners have a responsibility beyond process performance and process improvement vs. 6% in the early stage group. It is an open issue to what these responsibilities relate, but it would be highly interesting to investigate this further.

Regarding *authorities* that process owners are facilitated with, no connection with a difference in BPMM was found, which is a rather surprising outcome. Basically, process owners have similar a similar authority at both levels.

Comparing the *tasks* carried out by process owners on both BPMM levels (not shown in the table), it could be established that in the progressed group, the process owner role is extended with ‘external’ tasks such as ‘representing the process’ compared to the role in early stage organizations. Also, the task ‘initiating process improvement’ is much more often mentioned by respondents in the progressed group, which is in line with our earlier finding that process owners have a responsibility for process improvement in progressed organizations.

4.2 Case studies

While the survey outcomes had a focus on the organizational level, we like to recall that the case studies allowed for investigating both the *organizational* implementation of the process owner role and the *actual* fulfillment of the role at an individual level.

Case A

Case organization A is a financial service provider primarily servicing small and medium enterprises. The organization established a BPM department, which indicates that their BPM initiative can be classified as ‘progressed’. The aim of the BPM department is to optimize all processes within the organization in order to increase the *control* and *level of standardization*, and to decrease *operational risk*. A year prior to our research, the BPM department implemented a BPM governance structure in one from its many ‘process chains’ as a pilot-test for the implementation of BPM within the business. Table 4 displays the organizational implementation of process ownership for this process chain, of which the primary process concerns the collection of money from cash-dispensers by customers. This is a intensively automated process, which strongly leans on the support from various involved departments. Considering the content of Table 4, it can be seen that process owners in case A are responsible for process performance and budget allocation (for process improvement). Since BPM governance is still in a pilot phase in this organization, all aspects concerning budgets only exist on paper. The process owner tasks are: enabling process design, coordinating process improvement, ensuring correct process measures (KPIs), and chairmanship of process chain meetings, where performance issues are addressed among others. The authorities that process owners have are (joint) decision making regarding process design and the determination of process requirements (KPIs).

Table 3: Organizational fulfillment process ownership cases A and B

Element	**Case A**	**Case B**
Hierarchical position	Line management (product manager)	Line manager (responsible for a part of the process)
Individual/ team role	Individual role	Individual role
Ownership hierarchy	Limited to process owner and work process managers	Domain owner, process owner, process manager
Full-time / part-time	Part-time	Part-time
Responsibility	Process performance (on KPIs) Budget monitoring	Ensuring effective and efficient process design Ensuring compliance with legislation Ensuring synchronization with stakeholders Validating and approving process changes
Tasks	Enabling process design Translating strategy into KPIs Coordination / conduct of process improvement Chairmanship of process chain meetings Reporting to stakeholders	Translating strategy into process requirements Keeping process documentation up-to-date Assessing for the need for process improvement Communication with stakeholders
Authorities	Design process (within requirements of Marketing) Determine KPIs Allocating budget	Process design

The *individual* fulfillment of the process owner role differs substantially from the organizational fulfillment as described above. The tasks we found to be most important for the process owners are ‘monitoring process improvement’, ‘problem solving’ and ‘initiating process improvement’. Clearly, these tasks are in line with the formal responsibility for managing process performance. But as the process owner – as one of the very few in the entire organization – has gained the knowledge and complete overview of the entire process chain, this person has become a central contact point for all kinds of issues regarding the process chain. And because of the centralization of responsibilities to the process owner role, a start is being made with the development policies to the benefit of the entire chain in areas on which policies were lacking prior to the implementation of the BPM governance structure. For example, in the situation before the implementation of the BPM governance structure local branches could request the installation of a cash-dispenser from the IT department. In the current situation, such requests are assessed against a policy that maximizes overall organizational profit. Finally, the process owner carries out various ‘external’ tasks such as forming a contact point for all kind of process chain-related issues, which do not follow from the formal description of the role. Summarizing, the actual process owner role can best be summarized as that of *process chain manager.*

Case B
Case organization B is a maintenance, repair and overhaul service provider in the airline industry. The organization has established a BPM department which aims to *support* the business in reaching its goals by a *continuous optimization* of process, organization and information. One of the instruments of the BPM department in reaching their goals is the roll-out of an enterprise-wide BPM governance structure. The existence of a BPM-department and an enterprise-wide approach classifies this organization's BPM program as 'progressed'. The process domain under consideration in case B is the end-to-end process of servicing engines. The organizational implementation of process ownership can also be seen in Table 4. As the table shows, the *organizational* fulfillment of process owner role in case B contains aspects of process performance, process improvement, and process documentation. The process owner role is primarily assigned to line managers who are responsible for the performance and improvement of their *part* of the business process. This is odd for two reasons. In the first place, the idea of process owners is that it is a cross-functional role. Secondly, there is no distinction between the hierarchal manager and the process owner in this set-up. Therefore, it is questionable if this part of the organizational fulfillment of the role truly represents process ownership or whether this is a nominal indication only.

Regarding the *individual* fulfillment of process ownership, the process owners indicated that not much has changed by the implementation of the BPM governance structure. This is perhaps not surprising given the organizational fulfillment that was just discussed. Process owners report that their function has only been extended with designing the process and validating potential process changes, or in their own words with “keeping the process documentation up-to-date”. The image on the actual fulfillment of process ownership that emerges here, is only a modest role resembling the one in [19], where BPM is approached from a knowledge management perspective. As the organization can be described as a ‘Machine Bureaucracy’, one of Mintzberg’s organizational archetypes [12] which relies heavily on the standardization of rules, procedures and work processes, the process owner role in case B resembles an *administrative function* but not someone who has an end-to-end accountability for a process.

5 Discussion and conclusion

The first important insight from our research, in particular from our survey, is that process ownership is a role that seems to progress with an organization’s level of BPM maturity. This coincides with an early insight from [14]. On the basis of our descriptive research, it is not possible to say whether this is intrinsically a good or a bad thing. Nonetheless, it seems sensible that organizations that want to move towards a higher level of BPM maturity should assign their process owners with the task to look for process improvement opportunities *beyond* their regular operational duties in managing a process.

The second insight is that the fulfillment of the process owner role on the individual level can be very different from the organizational level. In other words, process owners do different things than what they are supposed to do. This insight follows most clearly from our case studies, where distinctive discrepancies were noted between the organizational and individual levels. It seems prudent that process ownership should be assigned to the best and most motivated people in an organization, as they may be expected to look for the maximal leverage they can get out this position. Our case study A clearly shows what such a ‘soul of fire’ can achieve beyond the formal duties that he or she is assigned with. Another suggestion we like to make is to clearly detach process ownership from conventional managerial roles. Our case study B shows the risk that otherwise process ownership will not be seen as something new and therefore not as something of value. In this particular case study, process owners become mere “process clerks”.

The scientific contribution of this paper is that it gives an empirical and contemporary insight into the fulfillment of the process ownership role. Our work provides an indication for the importance of (executive) process ownership, which justifies further research in this area. As our research was limited to describing how process ownership is fulfilled in practice, future research may aim to explore *what* makes the fulfillment of process ownership and BPM governance successful / effective across various levels of BPM maturity.

The main limitations surrounding the survey are related to the respondents, the number of responses, and the survey content. The respondents are not sampled from a completely random group, as they represent attendants to professional BPM conferences as well as contacts from our industrial network. This group gives our research a bias towards frontrunners in this field. Also, there is a clear bias towards the regional and cultural area of The Netherlands. Finally, the number of responses that was usable is limited to 22. Limitations surrounding the conduct of the case studies relate mainly to the generalization of its outcomes. Both cases studies concern organizations that are progressed with respect to their BPM maturity, so the noted differences between the organizational and individual level of process ownership should be clearly seen in this context.

Building on these preliminary insights, the next step is to broaden the empirical basis for this research both in numbers of respondents and in other geographical and cultural zones. If we rely on the assumption that increasing BPM maturity will lead to improved effectiveness cf. [6], the connection between process ownership fulfillment and BPM maturity should then be investigated in more detail. In the end, we would hope for sufficient insights and evidence to advice organizations to become truly process-oriented, to the merit of all their stakeholders.

Acknowledgements. This research is supported by the Technology Foundation STW, applied science division of NWO, and the technology programme of the Dutch Ministry of Economic Affairs.

References

[1] Boyatzis, R. E.: The Competent Manager: A Model for Effective Performance. Wiley (1982).

[2] Davenport, T. H.: Process Innovation: Reengineering Work Through Information Technology. Harvard Business School Press (1993).

[3] de Bruin, T.: Insights into the Evolution of BPM in Organisations, in: Toleman, M., Cater-Steel, A., and Roberts, D. (eds.) Proc. ACIS 2007, pp. 632-642, 2007.

[4] Gable, G. G.: Integrating Case Study and Survey Research Methods: an Example in Information Systems. EJIS, 3(2), 112-126 (1994).

[5] Hammer, M.: Beyond Reengineering. Harper Business (1996).

[6] Hammer, M.: The Process Audit. HBR, 85(4), 111-123 (2007).

[7] Hammer, M. and Champy, J.: Reengineering the Corporation. Harper Business (1993).

[8] Hammer, M. and Stanton, S.: How Process Enterprises Really Work. HBR, 77(6), 108-118 (1999).

[9] Hardjono, T. W. and Bakker, R. J. M.: Management Van Processen: Identificeren, Besturen, Beheersen En Vernieuwen. Kluwer (2002).

[10] Larsen, M. H. and R. Klischewski: Process Ownership Challenges in IT-Enabled Transformation of Interorganizational Business Processes, Proc. HICSS 2004, pp. 11, 2004.

[11] Mintzberg, H.: The Nature of Managerial Work. Prentice-Hall (1980).

[12] Mintzberg, H.: Structure in Fives: Designing Effective Organizations. Prentice-Hall (1983).

[13] Pinsonneault, A. and Kraemer, K. L.: Survey Research Methodology in Management Information Systems: an Assessment. JMIS, 10(2), 75-105 (1993).

[14] Pritchard, J. P. and Armistead, C.: Business Process Management: Lessons From European Business. BPM Journal, 5(1), 10-32 (1999).

[15] Reijers, H. A.: Implementing BPM Systems: the Role of Process Orientation. BPM Journal, 12(4), 389-409 (2006).

[16] Rosemann, M. and de Bruin, T.: Towards a Business Process Management Maturity Model. Proc. ECIS 2005, (2005).

[17] Rosemann, M., de Bruin, T., and Hueffner, T.: A Model for Business Process Management Maturity. Proc. ACIS 2004, (2004).

[18] Rummler, G. A. and Brache, A. P.: Improving Performance. Jossey-Bass (1990).

[19] Siemieniuch, C. E. and Sinclair, M. A.: On Complexity, Process Ownership and Organisational Learning in Manufacturing Organisations, From an Ergonomics Perspective. Applied Ergonomics, 33(5), 449-462 (2002).

[20] Vanhaverbeke, W. and Torremans, H.: Organizational Structure in Process-Based Organizations. Knowledge and Process Management, 6(1), 41-52 (1999).

[21] Yin, R. K.: Case Study Research: Design and Methods. Sage (2003).

[22] Yukl, G. A.: Leadership in Organizations. Prentice Hall (1994).

Investigating the Process of Process Modeling with Cheetah Experimental Platform – Tool Paper –

Jakob Pinggera[1], Stefan Zugal[1], and Barbara Weber[1]

[1]Quality Engineering Research Group, University of Innsbruck, Austria
{Jakob.Pinggera, Stefan.Zugal, Barbara.Weber}@uibk.ac.at

Abstract. When assessing the usability of BPM technologies enterprises have to rely on vendor promises or qualitative data rather than on empirical or experimental research. To address this need Cheetah Experimental Platform (CEP) has been developed fostering experimental research on business process modeling. CEP provides components that are frequently used in controlled experiments and allows their assembly to experimental workflows. CEP supports experimental execution by mitigating risks endangering data validity through better user guidance. Additionally, CEP provides richer evaluation techniques compared to paper based experiments fostering the experiment's data analysis.

1 Introduction

Providing effective IT support for business processes has become an essential activity of enterprises in order to stay competitive in today's market [1]. Unfortunately, when assessing the usability of BPM technologies enterprises have to rely on vendor promises or qualitative data rather than on empirical or experimental research [2]. This is rather surprising as these research methods have been successfully applied in similar research areas like software engineering (e.g., [3, 4]). In order to facilitate empirical research in the context of business process modeling we developed Cheetah Experimental Platform (CEP) providing means for effectively and efficiently conducting controlled experiments.

During our experimental research (e.g., [5–8]) we identified several typical problems in the different phases of experiments that might be addressed by appropriate tool support. In the *experimental design* phase the setup has to be defined, including the definition of objects, subjects and the execution order of different tasks. Providing components that are frequently used in controlled experiments (e.g., surveys, tutorials, process modeling tools) facilitate researchers the creation of experimental designs. Still, a successful experimental design largely depends on the experimenter's experience and knowledge of the domain. The second phase, *experimental execution*, highly benefits from rich tool support as many risks endangering data validity can be mitigated through better user guidance (e.g., avoiding that subjects do not follow the experimental setup). Finally, tool support can also be beneficial in the *experimental analysis*

B. Mutschler, J. Recker, R. Wieringa, J. Ralyté, and P. Plebani (Eds.):
CAiSE 2010 Workshop ER-POIS, Hammamet, Tunisia, pp. 13-18, 2010.

phase as richer data evaluation techniques are available compared to paper based experiments (e.g., replaying the modeling process).

The remainder of this tool paper is structured as follows. Section 2 introduces a running example, which will be used in Section 3 for describing CEP. Finally, Section 4 concludes the paper with a summary and outlook on future work.

2 Example

To illustrate the functionalities of CEP, we introduce a typical experimental design as a running example (cf. Fig. 1). Let us assume that the goal of the experiment is to investigate whether secondary notations (cf. [9]), for example, layout of a process model has an influence on the quality of a change conducted on that process model. To investigate this question, the subjects (participants of the experiment) are divided into two groups. The first group is asked to conduct a change on a process model with good layout, whereas the second group has to perform the *same* change task on the *same* process model, this time with poor layout. As the subjects' modeling capabilities might differ and therefore influence their modeling performance, the research team wants to collect demographical data of each subject (e.g., experience in business process modeling). In addition, it should be ensured that the lacking knowledge about *how* to use the modeling tool does not influence the results, i.e., the impact of learning how to use the tool should be minimized. Consequently, the research team decides to include a process modeling tutorial in the experiment. Besides, the mental effort necessary for conducting the process change should be documented. For this, a survey on cognitive load should be presented to subjects.

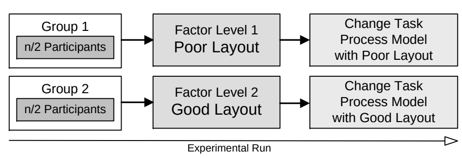


Fig. 1. Exemplary Experimental Design

3 Cheetah Experimental Platform

This section describes CEP. In particular, Section 3.1 illustrates how the platform can be used to support the design of experiments. Then, Section 3.2 deals with the actual operation of the experiment. Finally, Section 3.3 discusses how CEP fosters data analysis.

3.1 Experimental Design

Even tough the creation of experimental designs is a task highly relying on researcher's experience and domain knowledge, tool support can be beneficial in this phase. The majority of controlled experiments consists of a series of tasks that have to be executed by the experiment's subjects, referred to as *Experimental Workflow*. CEP enables experimenters to quickly assemble experimental workflows from components that have proven to work well in several experiments. In particular, CEP offers a set of frequently used components, including surveys, tutorials and *Cheetah Modeler* for creating business processes (cf. Section 3.2).

The exemplary experimental workflow described in Section 2 is supported by CEP as illustrated in Fig. 2. Depending on the number of different groups several branches are available in the experimental workflow configuration. At the beginning of the experiment, subjects are provided with assignment sheets containing an introductory text, instructions for performing the modeling tasks and a *group code*. Irrespective of the code the subjects entered, each participant has to fill out a demographic survey before working through an interactive tutorial. Based on the group code the respective branch of the experimental workflow is entered, presenting subjects with a change task for a process model with good/bad layout. Finally, participants are asked to fill out a survey about the cognitive load of the performed change task. All activities of the experimental workflow are handled using components provided by CEP.

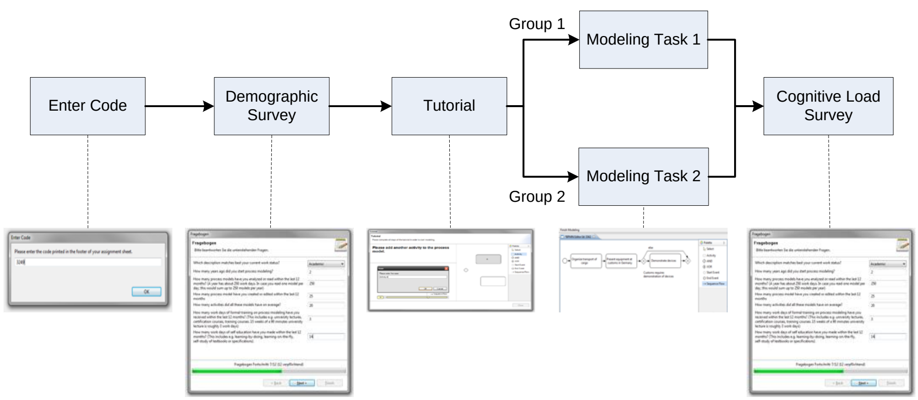


Fig. 2. Cheetah Experimental Workflow

3.2 Experimental Execution

Experimental Workflow When executing the experimental workflow configuration CEP guides the user through the experiment ensuring that the setup is followed. Furthermore, data collected when executing the experimental workflow is stored on a central database server, giving researchers the possibility to check whether all activities were completed and to restore the experiment to a specific state (e.g., in case of a crashed system). If the database server cannot be accessed a local copy is created and the user is asked to send it to the experiment's supervisor via email.

The experiment described in Section 2 is supported by CEP as follows. After entering the code identifying the group, the upcoming survey is collecting the user's demographic data. The survey ensures that all questions marked as mandatory are answered before the user continues with the next step in the experimental workflow. Before starting the actual modeling task the experimental workflow contains an interactive tutorial explaining the functionalities of Cheetah Modeler to make sure the used notation is well understood and participants know how to utilize the tool to change the process model. Therefore, each important functionality is presented by a screencast and users have to perform the corresponding modeling step. Depending on the entered code users are presented with process models with good/bad layout serving as a basis for the change task. Afterwards, a final survey assessing the mental effort for performing the change task is displayed.

Cheetah Modeler In order to enable the investigation of how process models are created, CEP offers Cheetah Modeler, which is a rather simple modeling component providing only basic modeling functionalities for simulating a pen and paper modeling session using a subset of BPMN (cf. Fig. 3). The focus was put on developing a tool facilitating the investigation of how process models are created, rather than providing a full fledged modeling suite. Currently, BPMN is the only process modeling language supported by CEP. Nevertheless, support for other notations was kept in mind when designing CEP and can easily be integrated.

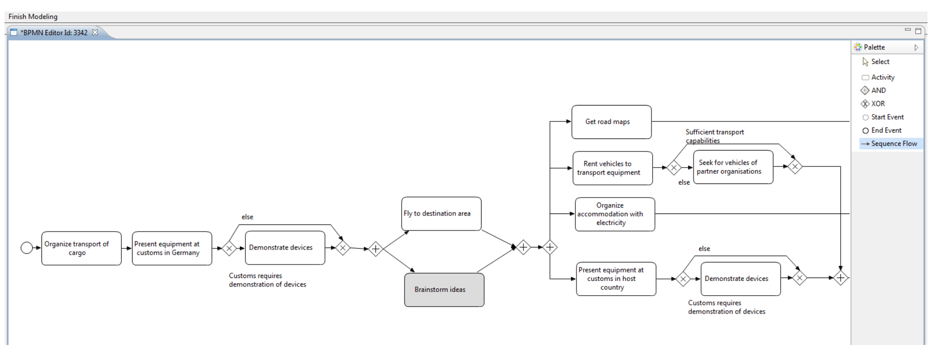


Fig. 3. Cheetah Modeler

Logging: Besides monitoring the experiment's correct execution and gathering the results of surveys, the collection of data on how users create process models was one of the main objectives when implementing Cheetah Modeler.

Consequently, every change to the process model (e.g., add/delete/move activity, add/delete/move edge) and the corresponding timestamp is automatically recorded and stored in a separate process log, offering the possibility for detailed investigations concerning the process of modeling (cf. Section 3.3).

3.3 Experimental Analysis

In addition to efficiently executing and monitoring experiments, data analysis was one of the main objectives when developing CEP. This section sketches the provided functionalities of *Cheetah Analyzer*, offering various data export features and means for replaying process models.

Experimental Workflow To be able to analyze data collected when executing the experimental workflow an export system is in place. By providing the option to export data as Comma-Separated Values (CSV) files, several tools for performing statistical analysis can be addressed (e.g., SPSS, Excel).

Process of Process Modeling One of the main advantages of using CEP is the possibility of replaying process models created with Cheetah Modeler. Recording all modeling steps enables researches to investigate *how* business process models are really created. For this purpose Cheetah Analyzer was implemented allowing for a step by step execution of modeling processes (cf. Fig. 4). Additionally, researches can export modeling processes using the Mining XML (MXML) format, allowing them to apply process mining techniques using ProM [10].

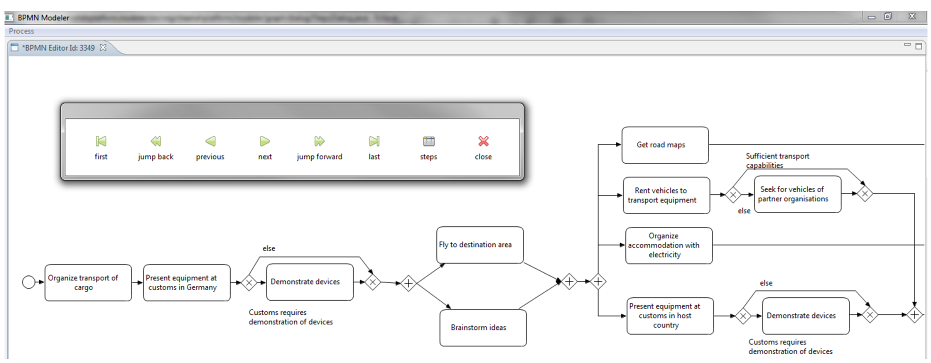


Fig. 4. Cheetah Analyzer

In context of the experiment presented in Section 2 researchers can have a detailed look on *how* the given process models were changed and if the layout had an influence on the change process. For example, it might be possible that users presented with a bad process layout rearranged activities before performing the actual change.

4 Summary and Outlook

Cheetah Experimental Platform, described in this tool paper, supports researches in conducting controlled experiments on business process modeling. In particular, CEP provides a repository of typical components (e.g., surveys, tutorials, process modeling tools) which can be used for assembling experimental workflows. Furthermore, the risk of producing invalid data is mitigated as the user is guided throughout the experiment's execution, reducing the number of accidental errors. In addition, richer analysis of data is possible compared to paper based experiments.

Future developments include a graphical experimental workflow and survey builder to further facilitate the creation of experimental designs as well as a dashboard simplifying the supervision of experiments. Furthermore, we would like to investigate the influence of collaborative modeling on *how* process models are created. For this purpose, CEP is currently extended toward collaborative modeling support.

Acknowledgements: We thank Dirk Fahland, Jan Mendling, Hajo A. Reijers, Matthias Weidlich and Werner Wild for their much appreciated feedback when developing Cheetah Experimental Platform.

References

1. Antonucci, Y.L.: Using workflow technologies to improve organizational competitiveness. Int'l. J. of Management **14** (1997) 117–126
2. B. Weber, B.M., Reichert, M.: Investigating the effort of using business process management technology: Results from a controlled experiment. Science of Computer Programming **75** (2010) 292–310
3. Myers, G.J.: A controlled experiment in program testing and code walkthroughs/inspections. Commun. ACM **21** (1978) 760–768
4. Lott, C.M., Rombach, H.D.: Repeatable Software Engineering Experiments for Comparing Defect-Detection Techniques. Empirical Software Engineering **1** (1996) 241–277
5. Weber, B., Reijers, H.A., Zugal, S., Wild, W.: The declarative approach to business process execution: An empirical test. In: CAiSE. (2009) 470–485
6. Pinggera, J., Zugal, S., Weber, B.: Alaska simulator supporting empirical evaluation of process flexibility. In: WETICE. (2009) 231–233
7. Zugal, S.: Agile versus Plan-Driven Approaches to Planning - A Controlled Experiment. Master's thesis, University of Innsbruck (2008)
8. Pinggera, J.: Handling Uncertainty in Software Projects A Controlled Experiment. Master's thesis, University of Innsbruck (2009)
9. Green, T.R.: Cognitive dimensions of notations. In: Proc. BCSHCI '89. (1989) 443–460
10. van der Aalst, W.M.P., Reijers, H.A., Weijters, A.J.M.M., van Dongen, B.F., de Medeiros, A.K.A., Song, M., Verbeek, H.M.W.E.: Business process mining: An industrial application. Inf. Syst. **32** (2007) 713–732

Handling Events During Business Process Execution: An Empirical Test

Barbara Weber[1], Jakob Pinggera[1], Stefan Zugal[1], and Werner Wild[2]

[1]Quality Engineering Research Group, University of Innsbruck, Austria
{Barbara.Weber,Jakob.Pinggera,Stefan.Zugal}@uibk.ac.at,
[2]Evolution Consulting, Innsbruck, Austria
Werner.Wild@evolution.at

Abstract. Declarative approaches have been proposed to counter the limited flexibility of the imperative modeling paradigm, but little empirical insights are available into their actual strengths and use. Our previous work has shown that end-users can effectively model and execute a declarative process with a considerable spectrum of constraints. However, what is still unclear is how effectively end-users are able to handle unforeseen events that can occur during run-time. This paper describes the design, execution, and results of a controlled experiment in which subjects have to execute a process with varying levels of events. The results suggest that our subjects, while being able to effectively handle constraints, have difficulties to handle unforeseen events during run-time. This outcome supports the argument that declarative processes require more experienced people, especially when dealing with unforeseen events.

1 Introduction

In today's dynamic business environment the economic success of an enterprise depends on its ability to react to change, like shifts in customers' attitudes or the introduction of new laws [1]. Process-aware information systems (PAISs) offer a promising perspective on shaping this capability, resulting in a growing interest to align information systems in a process-oriented way [2]. Yet, a critical success factor when applying a PAIS is the option to flexibly deal with process changes [3]. To address the need for flexible PAISs, competing paradigms enabling process changes and process flexibility have been developed, e.g., adaptive processes [4], case handling [5], declarative processes [6], and late binding and modeling [7] – for an overview see [8]. All of these approaches relax the strict separation of build-time (i.e., modeling or planning) and run-time (i.e., execution), which is typical for traditional workflow management systems following the imperative paradigm. However, by closely interweaving planning and execution the above mentioned approaches allow for a more agile way of planning. In particular, users are empowered to defer decisions regarding the exact control-flow to run-time, when more information is available. Depending on the concrete approach, planning and execution are interwoven to different degrees, resulting in different levels of decision deferral. The highest degree of decision deferral is enabled

B. Mutschler, J. Recker, R. Wieringa, J. Ralyté, and P. Plebani (Eds.):
CAiSE 2010 Workshop ER-POIS, Hammamet, Tunisia, pp. 19-30, 2010.

by declarative processes, which describe activities that can be performed as well as constraints preventing undesired behavior [8]. A declarative approach, therefore, seems to be particularly promising for highly dynamic processes [6, 9]. The support for partial workflows [9] allowing users to defer decisions to runtime [8], the absence of over-specification [6], and more maneuvering room for end-users [6] are all advantages commonly attributed to declarative processes.

Although the benefits of declarative approaches seem rather evident, such approaches are not yet widely adopted in practice. In addition, there is a lack of empirical evidence on how well declarative approaches perform in real-world settings. In our previous work we have shown that end-users can effectively model and execute declarative processes even with a considerable spectrum of constraints, especially when appropriate tool support is in place [10]. However, it is still unclear how well end-users can handle unforeseen events during the execution of declarative processes.

The goal of this paper is to pick up on the demand for more empirical insights into the use of declarative approaches. Specifically, we aim to investigate how the occurrence of exceptional situations may impede end-users' success when using a declarative approach for handling a particular business case (i.e., process instance). Proponents of declarative approaches argue that they are especially suited to support dynamic processes and that handling of unforeseen events is one of the strengths of the declarative approach [6, 9]. Due to its high flexibility the declarative approach provides maneuvering room for end-users to react upon unforeseen events without necessarily having to deviate from the process model. However, following literature on agile methods one could also argue that talent and skills are among the critical people-factors [11, 12] and declarative processes tend to necessitate a richer mix of higher-skilled people than traditional imperative approaches.

This paper reports on the results of a controlled experiment investigating how well inexperienced users can handle unforeseen events during the execution of declarative processes. Its findings are based on an experiment conducted in December 2008 at the Management Center Innsbruck with 20 students. The structure of this paper is as follows. After providing necessary background information in Section 2, Section 3 describes the experimental definition and Section 4 deals with the execution of the experiment and presents the results. Related work is listed in Section 5, Section 6 concludes the paper with a summary and an outlook.

2 Background

This section introduces declarative processes as well as the software used for the experiment, the Alaska Simulator.

2.1 Declarative Processes

There is a long tradition of modeling business processes in an *imperative* way. Process modeling languages supporting this paradigm, like BPMN, BPEL and

UML Activity Diagrams, are widely used. Recently, *declarative* approaches have received increased interest and suggest a fundamentally different way of describing business processes [13]. While imperative models specify exactly how things have to be done, declarative approaches only focus on the logic that governs the interplay of actions in the process by describing (1) the activities that can be performed, as well as (2) constraints prohibiting undesired behavior. An example of a constraint in a travel process would be that between a `Diving` activity and a `Flightseeing` activity there must be a resting period of two days to prevent aeroembolism. Imperative models take an 'inside-out' approach by requiring all execution alternatives to be explicitly specified in the model. Declarative models, in turn, take an 'outside-in' approach: constraints implicitly specify execution alternatives, as all valid alternatives have to satisfy the constraints [14]. Adding more constraints means discarding some execution alternatives (cf. Fig. 1). This results in a coarse up-front specification of a process, which can then be refined iteratively during run-time. Typical constraints described in literature can be roughly divided into three classes (e.g., [7, 13]): constraints restricting the *selection* of activities (e.g., the minimum or maximum occurrence of activities, mutual exclusion, co-requisite), the *ordering* of activities (e.g., pre-requisite or response constraints) and the use of *resources* (e.g., execution time of activities, time difference between activities, budget, etc.).

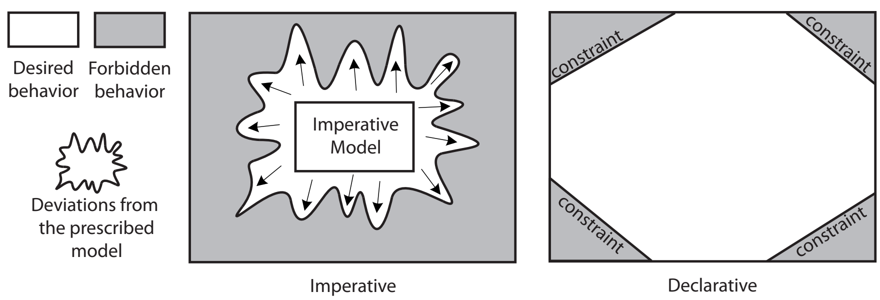


Fig. 1. Imperative vs. Declarative Approaches to Process Modeling (adapted from [14])

2.2 The Alaska Simulator

The Alaska Simulator[1] [15] fosters the comparison of different approaches to process flexibility, e.g., declarative processes, by using a *journey* as metaphor for a *business process*. The similarities being exploited here are that regardless of whether a journey or a business process is executed, various steps must be planned and carried out, even if the actual execution of those steps may be different from what is initially foreseen. Furthermore, journey planning is an attractive context for many people to become engaged in, which highly improves their willingness to participate in experiments.

The actions of a journey, like travel activities, routes and overnight stays correspond to activities in the business process. When conducting a journey, the

[1] Developed at the University of Innsbruck, http://www.alaskasimulator.org

goal is to maximize the travel experience (i.e., the overall "business value" of the journey), typical goals for business processes are the minimization of cost, cycle time or the optimization of quality or customer satisfaction. For optimizing the execution of a particular business case, information about the benefits (i.e., business value), cost and duration of activities is essential. Furthermore, both journeys and highly flexible business processes are characterized by incomplete information prior to execution. The business value for executing a particular activity within a business process is usually uncertain, likewise the outcome of a travel activity is not predefined and varies with the *weather conditions* encountered. The degree of variation is defined by the activity's *reliability*, i.e., low reliability indicates that the outcome of the activity is highly weather dependent. The overall business value of a journey (i.e., a numeric value representing the travel experience) is calculated as the sum of business values of all performed activities. Prior to performing the journey only the expected business value for each activity as well as its reliability (see below) are known. During the journey the activity's actual business value is calculated based on the weather conditions encountered. In addition to changing weather conditions, *unforeseen events* (e.g., a traffic jam resulting in delays) create uncertainty in a journey, while changing requirements or new laws complicate the modeling and execution of business processes. When composing a concrete business case, different constraints like *selection constraints*, *ordering constraints* or *resource constraints* have to be considered (cf. Section 2.1), similar constraints also exist when planning a journey (e.g., mandatory activities, dependencies between activities). To assess the last responsible moment for committing to an action, users must consider both its *availability* and *reliability*. By firmly booking an action its availability can be guaranteed, but the cost of the action must be paid immediately. If the booking is canceled during the journey, a cancelation penalty might apply, thus making too early commitments costly. Furthermore, booking is only possible up to a certain time before executing the action, as specified by the booking deadline.

Fig. 2 depicts the graphical user interface of the Alaska Simulator. Users can compose their individual travel plan by dragging available actions from the Available Actions View (3) onto the Itinerary (1). Actions are only available at a particular location on the Map (4). Existing constraints are displayed in the Constraint Overview (2) and have to be considered when composing a concrete journey. After each user (inter-)action, the journey is validated and the user is informed about any constraint violations and inconsistencies in the plan (5).

3 Experimental Definition and Planning

The main goal of the experiment is to evaluate the effects of unforeseen events on process outcomes (i.e., business value and number of failed journeys). Section 3.1 describes the setup of the experiment, the design is elaborated in Section 3.2. Finally, Section 3.3 discusses possible risks threatening the validity of the experiment as well as countermeasures taken.

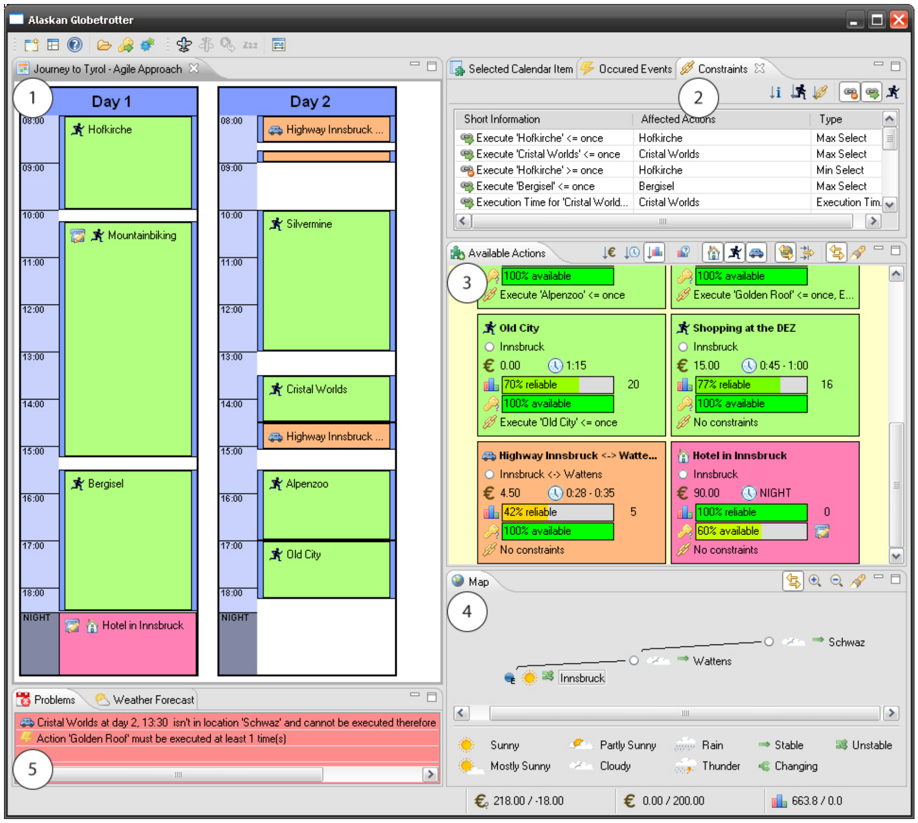


Fig. 2. Screenshot of the Alaska Simulator

3.1 Experimental Setup

Subjects: We conducted the experiment with 20 students of the study program "Management, Communication & IT" at the Management Center Innsbruck.

Objects: Two journeys representing two different business processes are used as objects, subsequently referred to as Configuration California (ConfCA) and Configuration Alaska (ConfAK). The configurations define the journey settings, like actions to be executed, constraints restricting their execution, events that might occur during run-time and weather conditions (cf. Section 2). For each of the configurations, two variants are created: A and B, differing in the number of events only. While Variant A contains no events, Variant B comprises many unforeseen events (e.g., event increasing the action's duration, temporary road closure). An overview of the different variant characteristics is given in Fig. 3.

Factor and Factor Levels: The number of unforeseen events that occur during run-time is the considered factor with levels "no events" and "many events". Variant A of a configuration corresponds to factor level "no events" and variant B to factor level "many events".

Variant	Events	Constraints
Alaska A		One budget contraint, one end-location constraint, three execution time constraints, three mandatory actions, one constraint requiring A to be followed by n times B and a final C, one pre-requisite constraint, one constraint requiring a minimum delay between two actions, one mutual exclusion constraint
Alaska B	Three events increasing the action's duration, two events increasing the action's duration and business value	
California A		One budget contraint, one end-location constraint, three mandatory actions, two execution time constraints, two constraint requiring a minimum delay between two actions
California B	Three events increasing the action's duration, one event increasing the action's duration and business value, one event closing a road for a certain period of time	

Fig. 3. Characteristics of the Configuration Variants

Response Variable: The achieved *business value* when planning and executing a given configuration with a given level of events is the response variable (cf. Section 2 for a description on how the business value is calculated). In addition, the *number of failed journeys*, i.e., journeys which could not be completed without constraint violations, is considered. To ensure comparability of results, weather conditions are the same for each subject.
Hypothesis Formulation: Goal of the experiment is to investigate the impact of unforeseen events on the response variables *business value* and *number of failed journeys*. Accordingly, we postulate the following hypotheses:

- **Null Hypothesis** $H_{0,0}$: There is no significant difference in the mean business values between configurations irrespective of the number of events.
- **Null Hypothesis** $H_{1,0}$: There is no significant difference in the number of failed journeys between configurations irrespective of the number of events.

Instrumentation: To ensure precise measurement of business value, the Alaska Simulator provides a mechanism for logging each relevant step the user undertakes while planning and executing a journey.

3.2 Experimental Design

The experimental setup is based on the guidelines for designing experiments in [16]. Following these guidelines a *randomized balanced single factor* experiment is conducted with *repeated measurements*. The experiment is called *randomized*, since subjects are assigned to groups randomly. We denote the experiment as *balanced* as each factor level is used by each subject, i.e., each student plans and executes two journeys, one without and one with events. As only a single factor is manipulated (i.e., the level of events), the design is called *single factor*. Due to the balanced nature of the experiment, each subject generates data for both factor levels and thus provides *repeated measurements*. Fig. 4 depicts the design following the aforementioned criteria. The subjects are randomly assigned to two groups of equal size, subsequently referred to as Group 1 and Group 2. To provide a balanced experiment with repeated measurements, the overall procedure consists of two runs. In the first run Group 1 applies factor level *no events* to object ConfCA, Group 2 factor level *many events* to the same object. In the second run, factor levels are switched and Group 1 applies factor level *many events* on ConfAK, Group 2 factor level *no events* to the same object. Since no subject deals with an object more than once, this design avoids learning effects.

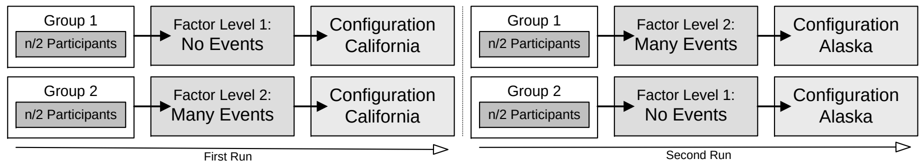


Fig. 4. Employed Experimental Design

3.3 Risk Analysis and Mitigation

In this section risks threatening the validity of the experiment are discussed.

Individual Planning Experience: Differences of participating students in respect to planning experience and productivity might have an impact on the students' performance, i.e., the business value achieved. This issue can be balanced by conducting the experiment with a sufficiently large and representative set of students or by replicating the experiment. The relatively low number of subjects (19 out of 20 could be used for data analysis) certainly constitutes a threat to the validity of this experiment.

Suitability of Metaphor: Whether or not the results of this experiment can be generalized to business process modeling and execution highly depends on the suitability of the chosen metaphor. However, due to the similarities of business process modeling and traveling planning and their respective execution (cf. Section 2), we assume the suitability of the metaphor. To further increase confidence in our view we plan testing whether travel planning serves as a good proxy for business process modeling and execution in future experiments.

Students instead of Professionals: In our experiment undergraduate students with limited planning experience were the subjects for investigating how well inexperienced users are able to model and execute declarative processes with varying numbers of events. While students can be regarded as suitable proxies for inexperienced users [17], it is arguable whether the results of this experiment can be generalized to professionals with significant planning experience. When replicating the experiment with experienced professionals we expect them to clearly obtain better process outcomes (i.e., higher business value and lower number of failed journeys).

Choice of Object: To be able to generalize results gained from this experiment, the configurations must be representative for a wide range of business process settings. Although the configurations used in this experiment do not have the complexity of real-world processes, they range well beyond the size of toy examples and include 22 and 26 activities, each of them varying in terms of expected business value, reliability and availability as well as their constraints and events.

Team Planning: In our experiment planning and execution was done on an individual basis, not in teams. Since planning often involves interactions among domain experts, system analysts and stakeholders, it has to be investigated how far our results can be transferred to team planning. For this we plan to replicate the experiment in a team setting.

4 Performing the Experiment

Section 4.1 describes the experiment's preparation and execution. Then, Section 4.2 presents the analysis of data for our experiment, followed by a discussion of the results in Section 4.3.

4.1 Experimental Operation

Experimental Preparation: The preparation of the experiment included the elaboration of the experimental design, implementation of the Alaska Simulator and devising the two travel configurations, i.e., ConfCA and ConfAK. To ensure that each configuration is correct and can be executed in the available amount of time, we performed pre-tests with several persons of different backgrounds.
Experimental Execution: The experiment was conducted in December 2008 at the Management Center Innsbruck. For organizational reasons, the execution of the experiment was split into two distinct sessions with 10 students each. At the beginning of each session, an introductory lecture was given to familiarize everyone with the Alaska Simulator and to clarify the experiment's rules and goals. For this, the students received a "starter kit" consisting of screencasts explaining the main features of the Alaska Simulator. Having watched the screencasts, the students were then randomly assigned to one of the two groups. As pointed out in Section 3.2, the experiment was executed in two subsequent runs, each taking about one hour. During the first 25 minutes of each run students could explore the configuration (i.e., ConfCA for the first run, ConfAK for the second run) to gather relevant domain knowledge. In the remaining 35 minutes students had to plan and execute the journey with the goal of optimizing the business value.
Data Validation: After having conducted the experiment, logged data was analyzed. We discarded data from one student since he did not follow the experiment setup (i.e., he adopted the same planning approach twice). Thus, 19 subjects remained for data analysis.

4.2 Data Analysis

In the following we describe the analysis and interpretation of data.

Descriptive Analysis: Based on data obtained from the logs of the Alaska Simulator, descriptive statistics for response variables *business value* and *number of failed journeys* were calculated.

Configuration	Approach	N	Failed Journeys	MinBV	MaxBV	MeanBV	Standard Deviation BV
California	No Events	9	0	5925	7492	6581	573
California	Many Events	10	4	4815	7289	5883	693
Alaska	No Events	10	1	2895	5497	4114	827
Alaska	Many Events	9	6	1045	4907	2826	1179

Fig. 5. Descriptive Statistics for Response Variables

Fig. 5 shows that for both ConfCA and ConfAK, Variant A (no events) yields a higher mean business value and a lower number of failed journeys compared

to Variant B (many events). The question is whether the observed differences in mean business values and number of failed journeys are statistically significant.
Data Plausibility: Fig. 6 shows a *box-whisker-plot diagram* as used for analyzing data plausibility. It visualizes data distribution and detects outliers. For ConfAK (many events) a single outlier exists. Since this is the only outlier, plausible data distributions seem to be in effect.

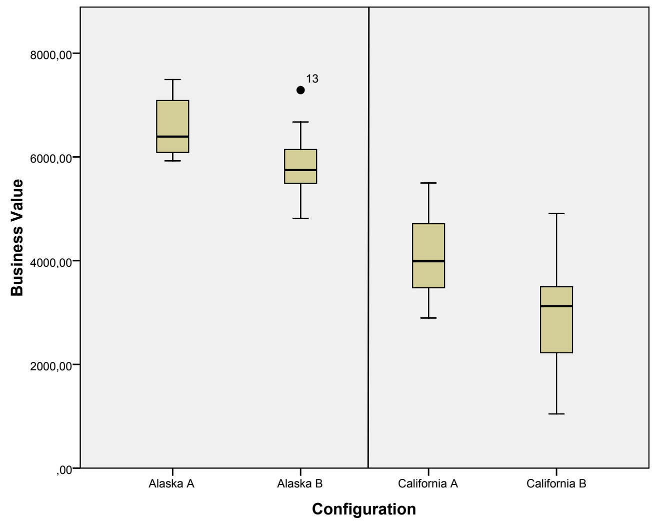


Fig. 6. Data Distribution (Box-Whisker-Plot Diagram)

Testing for Differences in Business Value: Since the expected business values of ConfAK and ConfCA differ, hypothesis testing is performed for each configuration separately. For ConfCA preconditions for the t-test for homogeneous variances are fulfilled (i.e., data is normally distributed and the Levene test confirmed equal variances). With an obtained significance of 0.013 (< 0.05) hypothesis $H_{0,0}$ can be rejected at a confidence level of 95%. ConfAK also fulfills all prerequisites for the t-test for homogeneous variances. The resulting significance of 0.030 (< 0.05) also leads to a rejection of hypothesis $H_{0,0}$ at a confidence level of 95%.
Testing for Differences in Number of Failed Journeys: To test for differences in the number of failed journeys we used Fisher's exact test [18]; the Chi Square test was not applicable due to the small sample size. For ConfCA, with an obtained significance of 0.017 (< 0.05) hypothesis $H_{1,0}$ can be rejected at a confidence level of 95%. For ConfAK, in turn, the obtained p-value of 0.054 ($>$ 0.05) is slightly above the cut-off point. Consequently, hypothesis $H_{1,0}$ cannot be rejected for ConfAK.

4.3 Discussion of Results

The major finding from our data analysis is that unforeseen events have a statistically significant impact on the outcome of journeys. Furthermore, the results obtained in this experiment seem to confirm the findings reported in [10], suggesting that handling constraints causes little difficulties for end-users, especially if appropriate tool support is provided. The constraints used in our experiment show a similar level of complexity as those used in [10]. As only one out of 19 journeys without unforeseen events was not completed successfully, we conclude that only when combined with events, constraints have a significant impact on the business value of a journey.

A manual analysis of our data showed that some events were more critical for the the process outcome (i.e., business value and number of failed journeys) than others. Especially, events that occurred in combination with mandatory actions were causing difficulties. For example, one of the events our subjects had to handle was a traffic jam on a route to a location with a single mandatory activity. This mandatory activity had a low availability and a restricted execution time. To handle this event, our subjects could not simply move the action, but usually had to rearrange major parts of the journey to fulfill the constraints and use the remaining time as effectively as possible.

Our results also support the argument that talent and skills are among the critical people-factors for agile methods as enabled by declarative processes [11, 12]. In fact, only few students were able to effectively handle events and to fully exploit the flexibility provided by the declarative approach. As stated in [10], tool support was essential for the successful dealing with constraints. Analogously, we assume that tools supporting the user's decision making, e.g., recommendation systems [19], might be beneficial for the journey's outcome.

5 Related Work

Most existing work about flexibly dealing with exceptions, changes, and uncertainty in the context of PAISs and related technologies is strongly *design-centered*, i.e., aiming at the development of tools, techniques, and methodologies. For overviews and discussions of these approaches, see [8, 20, 21].

Only few empirical investigations exist that aim to establish the suitability of the various proposed artifacts. Closely related to this paper is our previous work, which investigates how well end-users can cope with the gained flexibility provided by declarative approaches, especially when processes become rather complex [10]. While [10] focuses on the impact of constraints, this paper investigates the impact of unforeseen events. Also closely related is our work on the comparison of agile and plan-driven approaches to business process modeling and execution [22]. A theoretical discussion on declarative versus imperative approaches is provided in [23]. In [24], the results of a controlled experiment comparing a traditional workflow management system and case-handling are described. The systems are compared with respect to their associated implementation and maintenance efforts. In turn, the impact of workflow technology on PAIS development and PAIS maintenance is investigated in [25]. However, these works primarily focus on traditional workflow technology, while this paper puts its emphasize on declarative approaches. Other empirical works with respect to PAISs mainly deal with establishing their contribution to business performance improvement, e.g. [26, 27], and the way end-users appreciate such technologies, e.g., [28, 29].

6 Summary and Outlook

The advantages attributed to declarative processes are manifold, e.g., support for partial workflows allowing users to defer decisions to run-time, the absence of over-specification as well as more room for end-users to maneuver. However, their practical application requires the ability to resolve uncertainty and exploit learning outcomes. This raises the question whether inexperienced users are able to execute declarative processes especially when unforeseen events occur during run-time. This work picks up on this demand and contributes a controlled experiment comparing the process outcome for inexperienced users depending on the number of events.

While previous work shows that end-users can effectively handle varying levels of constraints when executing a declarative process, this paper demonstrates that unforeseen events are more problematic. The major result of our experiment is that process outcomes of inexperienced planners are significantly affected by unforeseen events which have to be handled during run-time. In particular, the combination of constraints and events turned out to be challenging for our subjects. These findings support the argument that declarative approaches require experienced users to fully exploit its benefits.

For further research we aim to investigate different techniques for improving understandability and maintainability of declarative process models to facilitate their application by less experienced users. Furthermore, we plan to run experiments in settings where planning is done in small teams, not individually, and replicate the experiment with more experienced users.

References

1. Poppendieck, M., Poppendieck, T.: Implementing Lean Software Development: From Concept to Cash. Addison-Wesley (2006)
2. Weske, M.: Business Process Management: Concepts, Methods, Technology. Springer (2007)
3. Mutschler, B., Reichert, M., Bumiller, J.: Unleashing the effectiveness of process-oriented information systems: Problem analysis, critical success factors and implications. IEEE Trans. on Systems, Man, and Cybernetics **38** (2008) 280–291
4. Reichert, M., Dadam, P.: ADEPT$_{flex}$ – Supporting Dynamic Changes of Workflows Without Losing Control. JIIS **10** (1998) 93–129
5. Van der Aalst, W., Weske, M., Grünbauer, D.: Case handling: A new paradigm for business process support. Data and Knowledge Engineering. **53** (2005) 129–162
6. Pesic, M., Schonenberg, M., Sidorova, N., van der Aalst, W.: Constraint-Based Workflow Models: Change Made Easy. In: Proc. CoopIS'07. (2007) 77–94
7. Sadiq, S., Sadiq, W., Orlowska, M.: A Framework for Constraint Specification and Validation in Flexible Workflows. Information Systems **30** (2005) 349 – 378

8. Weber, B., Reichert, M., Rinderle-Ma, S.: Change patterns and change support features -enhancing flexibility in process-aware information systems. Data and Knoweldge Engineering (2008) 438–466
9. Wainer, J., Bezerra, F., Barthelmess, P.: Tucupi: a flexible workflow system based on overridable constraints. In: Proc. SAC '04. (2004) 498–502
10. Weber, B., Reijers, H.A., Zugal, S., Wild, W.: The declarative approach to business process execution: An empirical test. In: Proc. CAiSE'09. (2009) 470–485
11. Highsmith, J., Cockburn, A.: Agile Software Development: The Business of Innovation. IEEE Computer **34** (2001) 120–122
12. Lindvall, M.: Empirical Findings in Agile Methods. In: Proc. XP/Agile Universe '02. (2002) 197–207
13. van der Aalst, W., Pesic, M.: DecSerFlow: Towards a Truly Declarative Service Flow Language. Technical report, BPMcenter.org (2006)
14. Pesic, M.: Constraint-Based Workflow Management Systems: Shifting Control to Users. PhD thesis, Eindhoven University of Technology (2008)
15. Weber, B., Zugal, S., Pinggera, J., Wild, W.: Experiencing Process Flexibility Patterns with Alaska Simulator. In: Proc. BPMDemos '09. (2009)
16. Wohlin, C., Runeson, R., Halst, M., Ohlsson, M., Regnell, B., Wesslen, A.: Experimentation in Software Engineering: an Introduction. Kluwer (2000)
17. Runeson, P.: Using students as experiment subjects: An analysis on graduate and freshmen student data. In: Proc. EASE'03. (2003) 95–102
18. Fleiss, J.L.: Statistical Methods for Rates and Proportions. Wiley (1981)
19. Schonenberg, H., B. Weber, B.F. van Dongen, W.v.d.A.: Supporting flexible processes through recommendations based on history. In: Proc. BPM'08. (51-66) 2008
20. Kammer, P., Bolcer, G., Taylor, R., Hitomi, A., Bergman, M.: Techniques for Supporting Dynamic and Adaptive Workflow. Computer Supported Cooperative Work **9** (2000) 269–292
21. Reijers, H., Rigter, J., van der Aalst, W.: The case handling case. International Journal of Cooperative Information Systems. **12** (2003) 365—391
22. Weber, B., Reijers, H.A., Zugal, S., Wild, W.: The Declarative Approach to Business Process Execution: An Empirical Test. In: Proc. CAiSE '09. (2009) 270–285
23. Fahland, D., Mendling, J., Reijers, H.A., Weber, B., Weidlich, M., Zugal, S.: Declarative versus Imperative Process Modeling Languages: The Issue of Understandability. In: Proc. EMMSAD '09. (2009) 353–366
24. Mutschler, B., Weber, B., Reichert, M.: Workflow management versus case handling - results from a controlled software experiment. In: Proc. SAC'08. (2008) 82–89
25. Kleiner, N.: Supporting usage–centered workflow design: Why and how?. In: Proc. BPM'04. (2004) 227–243
26. Oba, M., Onoda, S., Komoda, N.: Evaluating the quantitative effects of workflow systems based on real case. In: Proc. HICSS'00. (2000)
27. Reijers, H., van der Aalst, W.: The effectiveness of workflow management systems: Predictions and lessons learned. International Journal of Information Management **25** (2005) 458–472
28. Bowers, J., Button, G., Sharrock, W.: Workflow from within and without: technology and cooperative work on the print industry shopfloor. In: Proc. CSCW'95. (1995) 51–66
29. Poelmans, S.: Workarounds and distributed viscosity in a workflow system: a case study. ACM SIGGROUP Bulletin **20** (1999) 11–12

On Business Process Model Reviews

Alexander Grosskopf and Mathias Weske

Hasso Plattner Institute, University of Potsdam, Germany
{alexander.grosskopf,mathias.weske}@hpi.uni-potsdam.de
bpt.hpi.uni-potsdam.de

Abstract. In process reviews, domain experts validate the model against reality. In general, reviews are conducted in an iterative manner. Better reviews can build consensus faster and save iterations, i.e. time and money.
In an exploratory study, student clerks were asked to provide feedback to models from their domain. In this paper, we report on the study and the review performance. We explore typical issues raised in reviews and derive implications for practitioners and further studies. We identified education as the most influential factor on review performance in our sample set.

Key words: Process Modeling, Reviews, Performance, BPMN, t.BPM

1 Introduction

Visualized process models serve as a communication vehicle in business process management. Moreover, they become the blueprint for software implementations. On the path from the initial business process elicitation to software support, review cycles are required. Models are created once and get iterated several times. Iterations typically involve feedback cycles with domain experts. They have to ensure that the domain knowledge is properly represented in the model. Better review performance promises less iterations, which in turn translates to time and money saved on projects. But what can you expect from domain expert's reviews? How can you influence the performance of the reviewing task?

Empirical research on business process modeling has largely investigated the roles of models [1, 2] and modelers [3]. Condensed findings from empirical research even led to modeling guidelines [4]. Process reviews have not been addressed comparably.

We did a pre-study to assess the experiment setup for t.BPM [5]. It is a tangible toolkit to enable BPMN process modeling on a table. As part of this, university freshmen were introduced to BPMN and filled in a feedback test about a given model (adopted from [2]). It contained a graph with 14 tasks and five block structured exclusive and parallel sections. Activities were labelled with *A,B,C...* The students easily passed a test on understandability (also adopted from [2]). It turned out that all of them, had a strong formal background. Even though the freshmen were barely educated in process modeling, they could map the process

B. Mutschler, J. Recker, R. Wieringa, J. Ralyté, and P. Plebani (Eds.):
CAiSE 2010 Workshop ER-POIS, Hammamet, Tunisia, pp. 31-42, 2010.

semantics to known concepts from mathematics and physics, such as logical equations and circuit diagrams. This obviously influenced their performance. We concluded that, to get meaningful data about process reviews, a more realistic setup is needed.

In this paper, we first introduce our study design in Section 2. The data is evaluated and discussed in Section 3. Additional insights are drawn from investigating a related study in Section 4. We close the paper in Section 5 with a discussion of the findings and implications.

2 Study Design

2.1 Setup

The sample population, used in research studies, should be representatives of the population to which the researchers wish to generalize [6]. Thus, we wanted domain experts to provide feedback to domain specific processes. From interviews with BPM consultants we identified a typical scenario in which process consultants give workshops to elicit the domain knowledge, model the processes, and send them out as email attachment. Domain experts are asked to provide feedback. The model gets iterated. Part of the workshops with the consultant would be reserved to educate the participants about the goal of BPM and the notation used for process modeling.

To emulate best practices in the field, we designed the following exploratory study for subjects at the trade school in Potsdam. Students there are learners to become office or industrial clerks. They get practical training on the job and theoretical background for their profession at the trade school. As clerks, we consider them to be representatives of the population to be generalized on. We chose the domain processes *Moving to a new flat* and *Getting a new job*. The seventeen students (18-22 years) are considered to be domain experts, meaning they do know the context and can comment on the processes. Additionally, we designed a two page introduction into BPM and a one page modeling sample (topic: *Making Pasta*). The sample page contained a legend of the BPMN elements used. On that same page four pragmatical hints for to process modeling were provided. In particular, we suggested the balanced use of gateways, an eighty percent rule for relevance to set granularity, verb-object style activity labels as suggested by Mendling et al. [4] and a notational convention for conditions at gateways.

The introduction and sample sheet were designed to condition the subjects. They replace the guidance provided by the modeling experts in the workshops. The written form enforced the same type of treatment for all subjects. This was embedded in a larger experiment design to test the effect of t.BPM [5] on subjects. The hypotheses were that t.BPM modeling would yield positive effects on individuals, including more feedback in process model reviews. While the experiment result are to be published, this study explores partial data with focus on feedback performance. The experiment procedure is depicted in Figure 1.

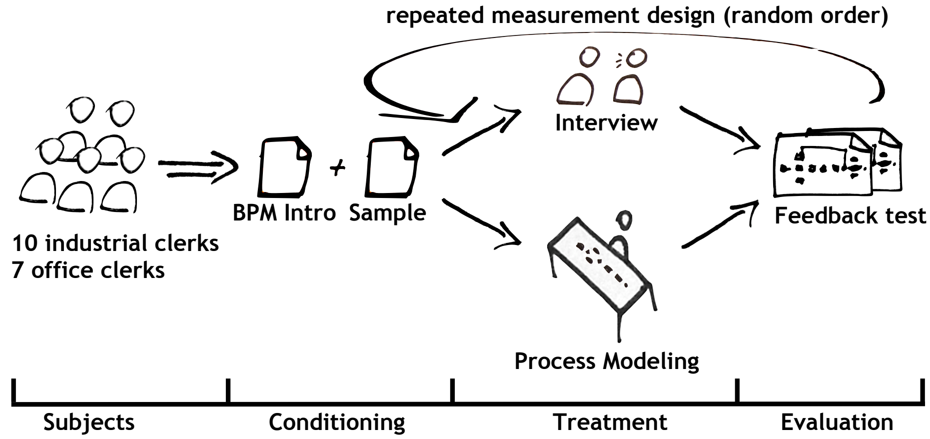


Fig. 1: Study Design

Each student got the BPM introduction and the sample. In general, time was not limited but tracked for each stage. Students were then randomly assigned to do either a structured interview or model their process on the table using BPMN elements. In that treatment step, they were asked to describe procurement processes, such as purchasing expensive hardware. Afterwards subjects were randomly given one of the process models for feedback. The treatment was repeated for each subject. In the second run, they got the alternative treatment and use the alternative feedback test.

In other words, the setup was a repeated measurement design in which all subjects get the same treatment in different orders. Subjects were assigned randomly. All subjects did interviews and process modeling. And all subjects did get both feedback tests, again randomly assigned.

2.2 Process Models used in Reviews

The process models used in the study are depicted in Figure 2. The models are annotated with the issues which we intentionally built into them.

Issues were chosen to belong to the area of *language* or *domain*. For example, two deadlocks were built into each process. This can be found by formal language analysis and requires no knowledge about the process domain. Nevertheless, we built-in these language related problems as indicators for the subjects' semantical understanding of the modeling language. The focus of this study are issues linked to the domain. They can only be interpreted if context information is available. Within the domain we consider three main categories: *labeling*, *information granularity* and *logical mismatch*. Labeling covers unsuitable naming of process elements, i.e. activity labeled with states not actions. Two obviously unsuited labels were build into the model (see Figure 2). Information granularity deals with too much or missing information in the process model. We left out an obvious activity and document per process model. Finally, logical mismatch describes wrong information in the model which contradicts the reality. We misplaced an activity to generate an issue of this category. An overview of the built-in problems is given in Table 1 when we report on the review performance.

One sample issue, a missing control flow connector, was marked up in the model to indicate how to give feedback. We asked reviewers very broadly to "provide feedback". We assume that guiding questions and a clear focus, e.g. communicating the goal of the modeling effort, would have steered the reviews. Our goal is to explore. Therefore, neither guiding questions nor a goal were provided.

2.3 Variables

For this investigation *feedback* is the **dependent variable**. We quantify feedback by counting the number items provided in a review. Feedback is distinguished into intentionally built-in problems and additional comments. Categorization and quantification of feedback items was done by expert reviews. We refer to the sum of all issues raised as feedback. While the quality of feedback matters most, we start with quantity for our exploration. The initial assumption was that variation in the amount of feedback could be explained by the treatment method (t.BPM vs. interviews). Data analysis revealed no influence by treatment method (details in Section 3.4).

Thus, we decided to explore other available information to explain the variance in the data set. In Section 3.4 we investigate the *time*, the participant's *education*, and *sex* as **independent variables**. Guiding questions and modeling goal were consciously excluded as variables from this study.

3 Data Exploration

3.1 Data Analysis Instruments

The data in the sample set was tested and is normally distributed[1]. Significance was tested with a one-tailed t-test, abbreviated here with p. Correlation between variables was calculated using Pearson's correlation coefficient r. It is a normalized measure of dependence between two quantities where 0 indicates no correlation, -1 is a perfect negative correlation and 1 is a perfect positive correlation. In Section 3.4 we use Multi Regression Analysis to explain variation with significantly influential factors. The coefficient of determination R^2 describes the proportion of variation in the data set that can be explained with the regression model. As an example, $R^2 = .30$ means that thirty percent of the data variation can be explained by a particular regression model.

Based on the repeated measurements design we treat each test as an independent sample (n=34). We keep in mind that pairs of samples result from a single person, but we'll see that splitting them up yields no negative effect on the data analysis. In summary,

[1] True for Kolmogorov-Smirnov and Shapiro-Wilk test

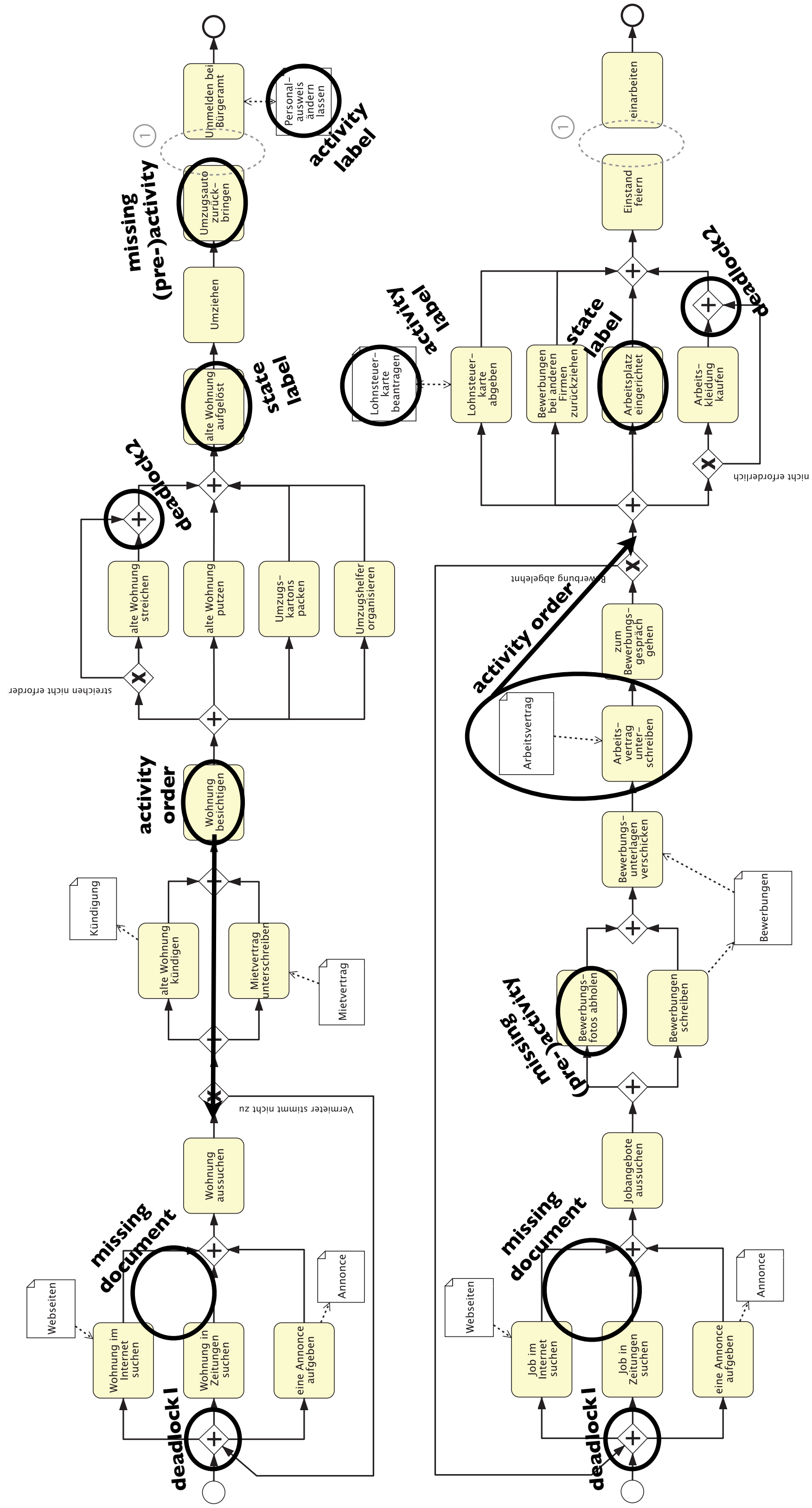


Fig. 2: Models used for feedback tests. Content is concerned with *Moving to a new flat* and *Getting a new job*

- data is normally distributed
- $r = [-1..1]$ describes the correlation of two quantities
- p is a one-tailed t-test, $p < .05$ is considered as significant
- $R^2 = [0..1]$ is the explained variance in the regression model
- all numbers are based on a sample set of $n = 34$

3.2 Review Performance

The review performance of the subjects was quite poor. Most problems were not found with an overall success rate of less then thirty percent. Table 1 lists the built-in issues and shows how often it was by a single reviewer in one or both of the feedback tests.

While the issue of a *wrong activity order* was always found, by all reviewers in all feedback tests, the opposite is the case for the *deadlock1* which results from a loop back. If reviewers found an issue only once, it indicates that they did not systematically checked for this type of issue.

Category	**Built-in Issues**	**Always Found**	**Found Once**	**Never Found**
Execution Semantics (Language)	deadlock1 (back loop)	0	0	**17**
	deadlock2 (bad block)	1	3	13
Labels (Domain)	activity labeled as state	1	2	14
	data object labeled as activity	2	2	13
Information Granularity (Domain)	missing document	4	0	13
	missing activity	3	2	12
Logical Mismatch (Domain)	wrong activity order	**17**	0	0

Table 1: Built-in issues in the two review models, to be found by reviewers

Investigating the individual performance, we found that review performance varies between one and six reported built-in issues. On average, only two were found per person and indeed, in nineteen of the thirty four cases, only one built-in issue, the *wrong activity order*, was found. Reviewers gave 2.2 additional comments. In summary, over all tests, subjects reported back 4.2 feedback items on average. It shows that, although only a few problems were found (69 out of 238 in total), the reviewers still had a lot to share about the process with 75 items delivered as additional feedback.

3.3 Distribution of comments on topics

The role of comments is to capture additional issues which were not intentionally built-in. Domain specific issues with processes are not necessarily modeling mistakes, they might be a conscious decision to capture a certain aspect, or not. If an issue was arguable, it was counted as a comment. Almost all comments were counted, except for two. They were dropped as questions about the notation, not comments on the process. Comments were aggregated if they centered around one single issue.

Category	Addtionally Reported Issues	Amount
Labels (Domain)	activity not labeled verb-object style*	1
Information Granularity (Domain)	missing document	6
	missing activity	20
	superfluous document	1
	superfluous activity	10
	missing event label*	4
	process scoping*	1
Logical Mismatch (Domain)	wrong activity order	12
	sequentialize parallel activities	9
	parallelize sequential activities	2
	lacking decision point	7
Ineffective Process	optimization potential	2

Table 2: Additionally reported issues in 34 reviews

Simply counting comments was not meaningful, so we categorized them, see Table 2. Two researchers reviewed each comment, negotiated the type and category. Despite the potential experimenter bias, the advantage of this qualitative approach is to discover new issues and categories.

As shown in Table 2, most comments seek to inject additional information into the process model (30 of 75), rather than leaving them out (only 11).

Parallelizing or sequentializing activities was surprisingly popular (11 of 75 comments). As an example, subjects commented for the process "*Moving to a new flat*" that they would not clean until the are done with painting or that changing the address with the authorities should be started earlier in the process (in parallel). In our opinion it indicates, that the reviewers understood the semantics of the model as well as the domain. Two reviewers found fundamental optimization potential. As an example, if multiple offered flats were researched early on, we do not need to loop back and start over with research all the time. This observation is acknowledged by introducing a new issue and category. However, one might argue that this new category relates to process design (to-be situation) whereas feedback is typically focussed on validation (as-is situation).

Most interesting to note are the three **issues marked with *** in Table 2. Those three categories originate from four reviewers. Two of them criticized that start and end events were not properly labelled. One argued that the activity "*Moving*" should be a word-object style label. One reviewer raised the question for process scoping. In particular, he commented, that the process *Moving to a new flat* should be completed after the rental contract was signed. The subsequent activities are not in the scope of the process. While the authors do not agree with this opinion, it brings up process scoping as an issue addressed in reviews. Notes taken during the pre-study interview indicate that all four subjects were involved in process modeling activities within their company. We conclude, that they brought in additional process knowledge which was not part of the conditioning for this study. With nine out of eleven issue types being new, this qualitative assessment of feedback widened our or repertoire of issues addressed in process model reviews.

3.4 Influential factors

The initial assumption was that t.BPM modeling influences the reviewer's performance, which did not happen. Indeed, subjects performed quite stable in both feedback tests independent of treatment order or type, see Table 3 for details.

We even found that the amount of feedback does not significantly differ between the first and the second feedback test. For that reason we decided to treat all thirty-four feedbacks as independent samples ($n = 34$). We also compared the mean scores for the two different feedback models. They do not significantly differ which indicates that both models were equally hard or easy to understand. We therefore conclude that *model type, treatment and order have no influence* on the reviewer's performance.

Sex, education, and time taken to conduct the review had a significant influence on the performance. For education and sex the results are depicted in Table 3. Education emerges as the most dominant factor with the highest effect size and strongest significance.

Influence Factor (independent variable)	**Alternatives**	**Effect Size** ($\overline{x}$ feedback)	**Significance** (one sided t-test)
Treatment order	1st t.BPM / 2nd interview	4.5556 / 4.7778	.354
	1st interview / 2nd t.BPM	3.875 / 3.75	.418
Treatment type	t.BPM / interview	4.1765 / 4.3529	.331
Model type	moving / job finding	4.0588 / 4.4706	.15
1st/2nd Test	1st / 2nd	4.2353 / 4.2941	.442
Education	office / industrial	2.50 / 5.45	.000019
Sex	male / female	4.95 / 2.92	.001

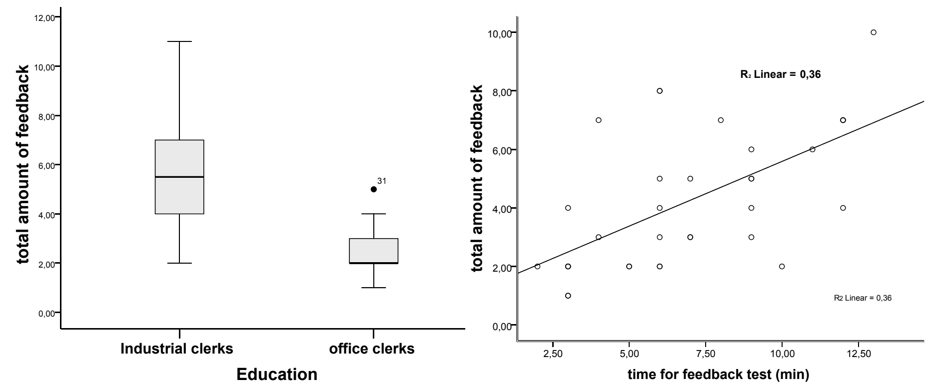
Table 3: Influential factors, individually tested for effect size and significance

To explain the significant ***influence of education***, we had post-study interviews with the principal of the trade school. We were informed that office clerks undergo a much stricter selection procedure and have better school achievements. On their job, they switch departments more easily and are often involved in supply chain optimization. Therefore, education for industrial clerks at the trade school does also include process notations, although in a very limited scope. Some students are also involved in process elicitation and modeling at their companies.

A boxplot in Figure 3a depicts the scattering of feedback for both groups. It visualizes that industrial clerks have much more to say about the process, up to eleven items in a single feedback test, while office clerks typically give two (at most five) items in a feedback test. In numbers, eight of fourteen tests done by office clerks reported one or two issues as feedback.

This dramatic difference due to education puts new light on our pre-study experience with HPI freshmen students. It raises the general question for transportability of empirical findings, if there is such a big gap between rather close professions.

The influence of sex is likewise significant with a considerable effect size. However, there is also a large overlap of sex and education in our sample set. Out



(a) Boxplot comparing education: office clerks only provide 1-5 items with a median of 2

(b) Scatterplot and regression curve: The longer a feedback test takes, the more feedback is gathered

Fig. 3: The influence of education and time on feedback visualized

of eleven industrial clerks in the study, eight were male and three were female. Whereas out of six office clerks, two were male and four were female.

We conducted a hierarchical multi regression analysis to determine the actual influence of this variable. This multi regression model has a coefficient of determination of $R^2 = .508$, which means that it can explain 50.8 % of the variance in the data[2]. In that model, the contribution of sex boils down to explain 0,1% of the overall variance ($R^2 = .001$). The standardized multi regression equation is:

$$FEEDBACK = .404 * education + .346 * time_{feedback} - .093 * time_{intro} + .043 * sex$$

The influence of time was determined using Pearson's correlation coefficient r. The time taken to complete the feedback test correlates significantly positive with the amount of feedback given ($p = .00008, r = .6$). That means, subjects that take more time for the feedback test, give more feedback. Figure 3b depicts the correlation in a scatterplot with a linear regression line. In the hierarchical multi regression model $time_{feedback}$ is the second strongest influence and contributes 10,4 % to the explanation of variance. While office clerks take about five minutes on average to complete the feedback test, industrial clerks take 8.3 minutes on average. We assume that subjects with less understanding have less to contribute and therefore need less time. Alternatively, subjects that investigate the process more deeply, find more issues but this of course needs more time. Similarly, people that need more time to read the BPM introduction perform worse in the feedback test ($p = 0.0375, r = -.39$). However, this has only a minor contribution of 0,9 % to the overall explanation.

Concluding the review of the influential factors, we can explain 50.8% of the overall variation using a Hierarchical Multi Regression Analysis which considers the four variables $education$, $time_{feedback}$, $time_{intro}$, sex. The main influential factor is education with the highest significance and effect. Education alone can

[2] $R^2_{education} = .393129 \quad R^2_{time_{feedback}} = .104 \quad R^2_{time_{intro}} = .009216 \quad R^2_{sex} = .001225$

explain 39.3% of the data in the Multi Regression Model. The significance and effect size found for sex (see Figure 3), diminishes in the Multi Regression Model.

3.5 Limitations Discussion

The validity of this explorative study is limited by the decisions taken for its practical implementation. In particular, one might argue that the domain processes from a private background might limit the transportability of findings to business domains (external validity). And of course, the definition of an "issue" as well as its categorization is subjective (internal validity).

The small sample set, with the influence factors reported earlier, also limits the generalizability of findings. Larger sets with more controlled variables should be used for hypotheses testing. In this exploratory study, the small sample set enabled us to look deeply into the reviews (qualitative research). Thereby, we identified new issues that we did not see before.

Throughout the study and its evaluation we took the following counter-measures to limited the experimenter bias:

- We standardized conditioning for the subjects using written documents.
- Two researchers coded the feedback and negotiated categories.

4 Related Work on Reviews

In 2002, Moody et. al. assessed a quality framework for conceptual models using process modeling [7, 8]. The subjects were 194 third year students in Information Systems (IS) which had to model a process and then peer review three processes modeled by others. A set of 20 process models and their reviews was qualitatively investigated.

Category	**defects**	**affected models**
Syntax (Language)	missing flows	50%
	wrongly specified decision point	35%
Labels (Domain)	poor naming of tasks	27%
Information Granularity (Domain)	missing roles	50%
	missing ressources	44%
	missing activity	25%
Logical Mismatch (Domain)	lacking decision point	30%
	wrong activity order	19%

Table 4: Defects in process models created by third year IS-students — In peer reviews "*64% of the defects went unreported.*" [7]

The authors state that "Many of the models were of quite poor quality, and counting the number of errors did not give interesting results." [7]. Thus the "errors" were classified and the reviews were assessed. He uses the notion of defects to summarize the issues. Table 4 shows the defects.

Interestingly, subsequent expert reviews found 6.6 defects per model of which 2.4 got reported by the reviewers. In other words, "on average, 64% of the defects went unreported" [7]. These numbers compare well with our seven intentionally built-in problems of which 2 were found (success rate $< 29\%$) on average.

While this is the nearest known relative to our study, several fundamental differences hamper a proper comparison of numbers from both studies. To name the most important ones,

- The review reported in Table 4 relates to modeling defects. In the study, reviews are evaluated by reporting true/false negatives/positives.
- The notion of defect used by [7] is much stronger than our notion of issues.
- Quality and defects per model did vary in [7], while we had a stable set of pre-defined issues per model.
- IS students have a very different education. They are method experts rather than domain experts.

Nevertheless, we learn from this study defect types that can be build into models for review tests. This further extends our set of feedback issues. Most important, we learn that proper education does not guarantee good process reviews. Thus, further research is needed to de-mystify the task of reviewing.

5 Discussion

Reviews by domain experts are a critical part of model validation and need more scientific investigation. Better review performance can avoid additional iterations needed in process analysis and design. This equals money and time saved on a project. We conducted an explorative study using qualitative and quantitative methods.

Findings from this study are the issue types and the influence factors. The identified issue types can be used to create better models for review tests with a larger variety of built-in issues. The distribution of issues raised by reviewers is also a finding. It can be used as a starting point to guide reviewers in their task. In other words, issues that are often missed might be worth a hint. Thus, reviewers can systematically check for them. A guideline for reviewers was out of scope for this work.

By statistical evaluation, we found education, sex and time as influential factors in the sample set. In particular, education dominated our findings. Although we had similar previous experiences with university freshmen, we did not anticipate education to be as influential within office and industrial clerks. We conclude that the subject group should be as homogeneous as possible to exclude those influences on the data set in future investigations. At the same time the model should involve a large variety of issues. Thus, it is possible to create the variance needed for insightful results. Our findings are limited by the small sample set and the dominance of education as an influential factor.

Implications for practitioners are phrased as suggestions to process modelers that do review cycles with domain experts. We suggest to,

- choose your reviewers wisely (huge differences in review performance)
- one reviewer per model is not enough (on avg. $> 60\%$ of issues not found)

Further research can built on the findings from this study to build a proper controlled experiment. In particular, the influential factors identified here should be fixed to rule them out. When designing models for review experiments, future research can take advantage of the domain related problems identified in this study. That can help to create models with a larger variety of problems built in.

In this study, we left out the aspects of a modeling goal and guiding review questions. We assume that they significantly influence the performance of reviewers. For example, guiding questions can link to frequently unreported issue types. In future work, we intend to investigate the influence of these aspects on reviewing performance.

Acknowledgements

We gratefully acknowledge the support of Karin Telschow and Markus Guentert to setup and conduct the t.BPM experiment series. Special credits to Karin for her great support during the data exploration phase.

References

1. Holschke, O., Rake, J., Levina, O.: Granularity as a Cognitive Factor in the Effectiveness of Business Process Model Reuse. In: Proceedings of the 7th International Conference on Business Process Management, Springer (2009) 260
2. Melcher, J., Mendling, J., Reijers, H., Seese, D.: On measuring the understandability of process models (experimental results). In: 1st Int. Workshop on Empirical Research in Business Process Management (ER-BPM). (2009)
3. Recker, J., Dreiling, A.: Does it matter which process modelling language we teach or use? In: 18th Australasian Conference on Information Systems. (2007) 356–366
4. Mendling, J., Reijers, H., van der Aalst, W.: Seven process modeling guidelines. Information and Software Technology (IST) (2009)
5. Grosskopf, A., Edelman, J., Weske, M.: Tangible business process modeling - methodology and experiment design. In Mutschler, B., Wieringa, R., Recker, J., eds.: 1st Int. Workshop on Empirical Research in Business Process Management (ER-BPM'09). (September 2009) 53–64
6. Cooper, D., Schindler, P.: Business Research Methods. 10 edn. McGraw-Hill Higher Education (2008)
7. Moody, D., Sindre, G., Brasethvik, T., Sølvberg, A.: Evaluating the quality of process models: empirical analysis of a quality framework. In: 21st Int. Conference on Conceptual Modeling–ER. (2002)
8. Sindre, G., Moody, D., Brasethvik, T., Solvberg, A.: Introducing peer review in an IS analysis course. Journal of Information Systems Education **14**(1) (2003) 101–120

The Impact of Sequential and Circumstantial Changes on Process Models

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

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

Abstract. While process modeling has become important for documenting business operations and automating workflow execution, there are serious issues with efficiently and effectively creating and modifying process models. While prior research has mainly investigated process model comprehension, there is hardly any work on maintainability of process models. Cognitive research into software program comprehension has demonstrated that imperative programs are strong in conveying sequential information while obfuscating circumstantial information. This paper addresses the question whether these findings can be transferred to process model maintenance. In particular, it investigates whether it is easier to incorporate sequential change requirements in imperative process models compared to circumstantial change requirements. To address this question this paper presents results from a controlled experiment providing evidence that the type of change (sequential versus circumstantial) has an effect on the accuracy of process models. For performance indicators *modeling speed*, *correctness*, and *cognitive load* no statistically significant differences could be identified.

1 Introduction

The increasing use of business process models has sparked a discussion on usability and quality issues. Large companies use business process modeling as an instrument to document their operations, typically resulting in several thousand process models which are partially created by staff members with limited modeling expertise. Therefore, analyzing factors that influence the usability of process models is a promising approach for securing success of process modeling initiatives [2].

Prior research has mainly investigated process model comprehension as a prerequisite for usability. Among others, modeling expertise and process model

B. Mutschler, J. Recker, R. Wieringa, J. Ralyté, and P. Plebani (Eds.):
CAiSE 2010 Workshop ER-POIS, Hammamet, Tunisia, pp. 43-54, 2010.

complexity have been identified as factors of comprehension [15]. Yet, comprehension captures only a partial dimension of usability. Process models in current process modeling initiatives are subject to frequent changes and a considerable amount of staff members are involved in updating process models. For this reason, investigating *process model maintainability* bears the potential to improve current process modeling practice.

Up until now, there is hardly any work on maintainability of process models beyond research into complexity metrics [3]. In this paper, we analyze to what extent cognitive research into software program comprehension can be transferred to process model maintenance. We feel that insights from the domain of software engineering are potentially valuable for process models given the high degree of similarities between software programs and process models (see [9, 21] for discussions of these similarities). Work on the cognitive dimensions framework has established a relativist view on usability [6, 8, 7]. In particular, it was demonstrated that imperative programs are strong in conveying sequential information while obfuscating circumstantial information. In this context, *sequential* information explains how input conditions lead to a certain outcome, and *circumstantial* information relates to the overall constraints that hold when that outcome is produced. We challenge this hypothesis for imperative process models in BPMN and test whether maintainability is influenced by the type of change requirement. Accordingly, we conduct an experiment that checks if sequential change requirements are easier to implement for a BPMN model than circumstantial change requirements. The results of this experiment foster research on maintainability factors of process models.

The remainder of the paper is structured as follows. Section 2 discusses the background of our research, namely sequential and circumstantial change requirements. Section 3 describes the setup for our experiment, which builds on a realistic modeling task taken from the disaster management domain. Section 4 covers the execution and the experiment's results. Finally, Section 5 discusses related work, followed by a conclusion.

2 Background

The central subject to maintainability considerations is the notion of a process change. A *process change* is the transformation of an initial process model S to a new process model S' by applying a set of change operations. A change operation modifies the initial process model by altering the set of activities and their order relations [12]. Typical change primitives are *add node*, *add edge*, *delete node*, or *delete edge* [23]. Figure 1 shows a BPMN process model from the domain of earthquake response, which is a simplified version of a process run by the "Task Force Earthquakes" of the German Research Center for Geosciences (GFZ). The main purpose of the task force is to coordinate the allocation of an interdisciplinary scientific-technical expert team after catastrophic earthquakes worldwide [5].

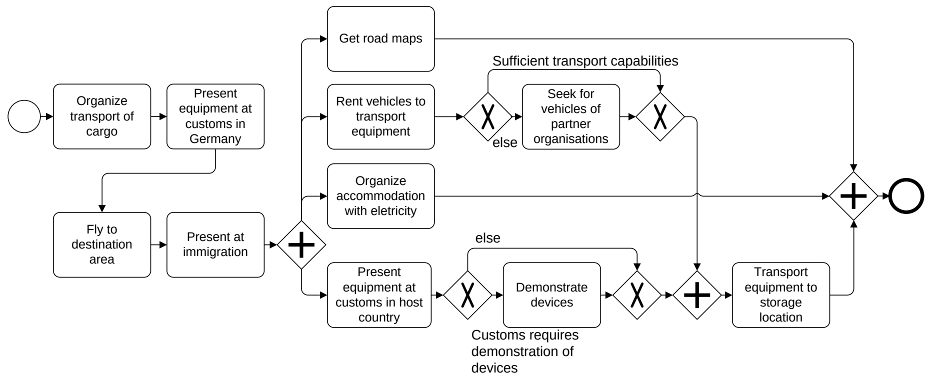


Fig. 1. BPMN Model for Transport of Equipment

According to considerations on cognitive software program analysis, not all change requirements are equally difficult (cf., [6]). Here, we call a change requirement *sequential* if an activity has to be added, deleted, or moved directly before or behind another activity. For example, once arrived in the host country, the taskforce has to demonstrate the devices to customs (cf., Fig. 1). In contrast to the model of Fig. 1, customs might not clear the equipment which requires additional activities. A concrete change might be to insert an activity "Negotiate with customs" in the process after "Demonstrate devices." Such a sequential change requirement describes whether a pair of activities is in a specific structural or behavioral order relation. In contrast, a *circumstantial* change requirement involves adding or moving an activity such that a general behavioral constraint is satisfied. Such a constraint might be given in terms of temporal operators like 'always', 'eventually', 'until', and 'next time'. As an example, consider a change requirement to execute "Demonstrate devices" eventually in each case. The region in the process model that needs to be changed cannot be deduced from the change requirement directly. Consequently, sequential changes tend to be rather *local* in the process model, whereas circumstantial changes tend to affect the process model *globally*. Two realistic change requirements are given in Appendix A.

How do these observations on process models relate to established theories? Adapting a software program to evolving needs involves both *sense-making tasks* (i.e., to determine which changes have to be made) and *action tasks* (i.e., to apply the respective changes to the program) [8]. We can discuss the problem of changing a process model in a similar vein. When process designers are faced with a change requirement, they have to consider two things: 1) they need to determine which change operations have to be used to modify the process model; and 2) they have to apply the respective changes to the process model. Consequently, the effort needed to perform a particular process model change is on the one hand determined by the cognitive load to decide which changes have to be made to the model, which is a comprehension and sense-making task. On the other hand, the effort covers the number of edit operations required to conduct these changes, which is an action task.

In the cognitive dimensions framework, an important result – regarding sense making of information artifacts – relates to the difference between the tasks of looking for sequential and circumstantial information in a software program. Transferring this result to process models reads as follows: circumstantial changes are more difficult to perform on a flow chart diagram like BPMN [8]. Consequently, we would expect that process designers show a better performance in applying sequential change requirements. We challenge this hypothesis in an experimental setup.

3 Research Setup

In this section we describe the design of an experiment that investigates the influence of different change types on modeling performance.

Subjects: In our experiment, the subjects are 15 students in Software Engineering of a graduate course on Business Process Management at the Hasso Plattner Institute. Participation in the study was voluntary.

Objects: The object of our experiment is a process model along with two descriptions of a change that have to be applied to the model. The process model used in our experiment describes an actual process run by the "Task Force Earthquakes" of the German Research Center for Geosciences (GFZ) [5]. In particular, we used a model of the "Transport of Equipment" process similar to the one shown in Fig. 1, which specifies how the transport of scientific equipment from Germany to the disaster area is handled by the task force. The two change descriptions require changes of this process if standard processing is not possible. On the one hand, it might happen that the transport of the equipment is delayed as customs might not clear the equipment immediately. On the other hand, equipment transport capacity might not be available right away. For both cases, the process of transporting the equipment has to be changed accordingly.

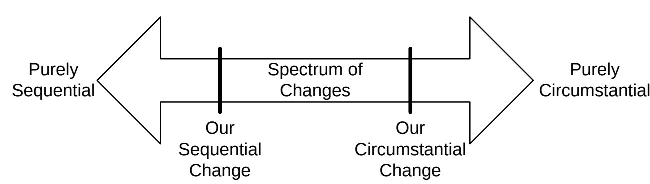


Fig. 2. Change Types

Factor and Factor Levels: The considered factor in our experiment is the type of the change task with factor levels *sequential* and *circumstantial*. It is important to note that the two change tasks used in the experiment are not strictly sequential and circumstantial. However, when compared to each other, one change is clearly more sequential, or circumstantial, respectively, than the other (cf., Fig. 2). We also ensured that both changes require the same effort in terms of graph-edit distance (i.e., the minimal number of atomic graph operations needed to transform one graph into another, it can be leveraged to assess the similarity of two process models [4]). For both changes, the graph edit distance between the original model and the changed model, i.e., the number of operations needed to perform the change is around 40 atomic change operations.

Response Variables: As response variables we consider the *modeling speed* of conducting the modification tasks, the *accuracy* of the change, the *correctness* of the resulting model as well as the perceived *cognitive load* of conducting the modification tasks. *Modeling speed* is measured as time (in seconds) needed for conducting a change task. For assessing the *accuracy* we utilize a set of 12 key properties for each change type, which are derived directly from the corresponding change description. For instance, "in the meantime" indicates parallel execution, whereas explicit naming of activities in the text indicates that respective activities should also be present in the process model. One point is rewarded for each fulfilled property in the solution model (e.g., existence of parallel execution). In addition, accuracy also includes penalty points for negative key properties (e.g., superfluous activities). Consequently, students are able to gather at most 12 points per change, allowing us to quantify their models in terms of accuracy. *Correctness*, in turn, is assessed in terms of model syntax as well as execution semantics. That is, whether syntactic requirements imposed by the BPMN specification are met, and whether the model is free of behavioral anomalies such as a deadlock or a lack of synchronization. To this end, we applied the well-known soundness criterion [20]. For obvious reasons, soundness checking is done solely for syntactically correct models. Finally, subjects are asked to assess their *cognitive load* (i.e., the perceived difficulty of conducting a change task) on a 7-point Likert scale.

Hypothesis Formulation: The goal of the experiments is to investigate whether the type of change influences *modeling speed*, *accuracy*, *correctness*, and *cognitive load*. Accordingly we postulate the following hypotheses:

- **Null Hypothesis** $H_{0,1}$: There is no significant difference in the speed of modeling a process change with respect to the type of change.
- **Null Hypothesis** $H_{0,2}$: There is no significant difference in the accuracy of the resulting models with respect to the type of change.
- **Null Hypothesis** $H_{0,3}$: There is no significant difference in the correctness of the resulting models with respect to the type of change.
- **Null Hypothesis** $H_{0,4}$: There is no significant difference in the perceived cognitive load with respect to the type of change.

Instrumentation: The participants conducted the modeling using the Cheetah BPMN Modeler [17], which is a graphical process editor. The editor provides only basic drawing functionality for creating, moving, and deleting nodes and edges of a single BPMN diagram; the modeling constructs were limited to tasks, start and end events, gateways (AND, XOR), and control flow edges. The reduced functionality mimics a flexible "pen and paper" setting. To be able to trace the actual modeling process, we extended the BPMN Modeler with a logging function, which automatically records every modeling step and allows us to derive performance characteristics (e.g., modeling time, number of syntactical errors, number of events) for each model, and a function to replay a modeling log.

Experimental Design: The experimental setup is based on literature providing guidelines for designing experiments [24]. Following these guidelines a *randomized balanced single factor* experiment is conducted with *repeated measurements*. The experiment is called *randomized* because subjects are assigned to groups randomly. We denote the experiment as *balanced* as each factor level is used by each subject, i.e., each student works on a sequential and circumstantial change task. As only a single factor is manipulated (i.e., the change type), the design is called *single factor*. Due to the balanced nature of the experiment, each subject generates data for both factor levels and thus provides *repeated measurements*. Figure 3 depicts the design following the aforementioned criteria. The subjects are randomly assigned to two groups of equal size, subsequently referred to as Group 1 and Group 2. To provide a balanced experiment with repeated measurements, the overall procedure is divided into two runs. In the first run Group 1 works on a *sequential* change task, Group 2 on a *circumstantial* one. In the second run factor levels are switched and to Group 1 the *circumstantial* factor level is applied, to Group 2 the *sequential* factor level. Since no subject deals with an object more than once, this design avoids learning effects.

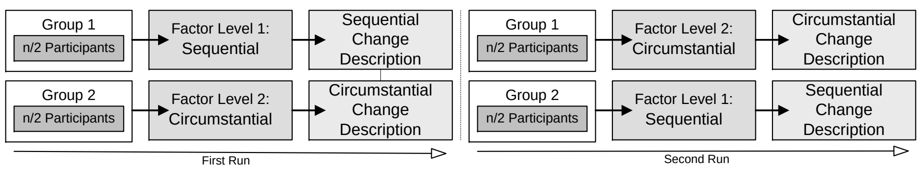


Fig. 3. Employed Experimental Design

4 Performing the Experiment

By now, the setup of the experiment has been explained. Section 4.1 describes the preparation and execution of the experiment. Then, the analysis and interpretation of the gathered data are presented in Section 4.2. Finally, a discussion of the results is provided in Section 4.3.

4.1 Experimental Preparation and Execution

Preparation: As part of the experimental preparation, we created the model for the "Transport of Equipment" process and two different change task descriptions, one rather sequential change task and one rather circumstantial change task. In order to ensure that each description is understandable and can be modeled in the available amount of time, we conducted a pre-test with 14 graduate students at the University of Innsbruck. Based on their feedback, the change task descriptions were refined in several iterations; the resulting tasks are shown in Appendix A.
Execution The experiment was conducted in January 2010 in Potsdam. A session started with a familiarization phase, in which students had 10 minutes to investigate the given model for the "Transport of Equipment" process. At the end of the familiarization phase, students had to answer comprehension questions on

the “Transport of Equipment” process before they were able to proceed with the experiment. The familiarization phase was followed by a modeling tool tutorial in which the basic functionalities of the BPMN Modeler were explained to our subjects. The students were then randomly divided into two groups. As pointed out in Section 3, the experiment was executed in two subsequent runs. After completing the two change tasks, a questionnaire on cognitive load was presented to the students.
Data Validation: Once the exploratory study was carried out, the logged data was analyzed. Data provided by 15 students was used in our data analysis.

4.2 Data Analysis

In this section, we describe the analysis of gathered data and interpret the obtained results.
Testing for Differences in Modeling Speed: To test for differences in terms of modeling speed, a t-test for homogeneous variances was applied [13]. The test was applicable to analyze time differences because the samples of both factor levels follow normal distributions and the variances of the samples are homogeneous. With an obtained p-value of 0.818 (> 0.05), hypothesis $H_{0,1}$ cannot be rejected at a confidence level of 95%. In other words, there is no statistically significant difference with respect to the speed of answering between the two factor levels. This outcome is re-enforced by the overlapping boxplots in Fig. 4.

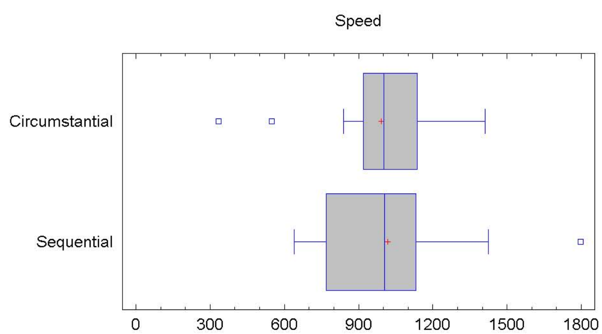


Fig. 4. Speed of Executing the Modeling Task

Testing for Differences in Accuracy: Fig. 5 shows the boxplots displaying the distribution of the accuracy values as obtained for the two factor levels, i.e., the circumstantial and the sequential change task. For the circumstantial task compared to the sequential task the median value is lower, as well as the overall distribution is being situated at the lower side of the accuracy axis. To test whether differences in terms of accuracy are statistically significant, we deployed the t-test. The test is applicable again because both samples are normally distributed and the variances of the samples are homogeneous. With an obtained p-value of 0.042 (< 0.05) hypothesis $H_{0,2}$ is rejected at a confidence level of 95%. In other words, the lower accuracy values obtained for the circumstantial task are statistically significant.

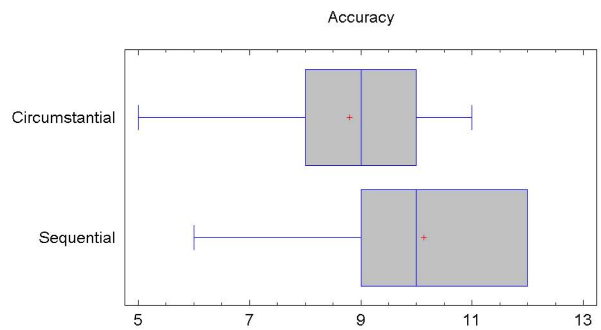


Fig. 5. Accuracy of Process Models

Testing for Differences in Correctness: To test for differences in correctness between the two factor levels, we inspected all models against the BPMN standard and scored whether these models were syntactically correct or not. Since the binomial data that was obtained in this way was not normally distributed, we applied the non-parametric Mann-Whitney test [19]. This resulted in a p-value of 0.053 (> 0.05). As an alternative way to compare the correctness of the models, we considered the *soundness* of the produced models, which is a well-established correctness notion for process models [20]. We applied the same statistical test to compare the two factor levels, which led to a p-value of 0.275 (> 0.05). Since both p-values exceed the threshold of 0.05, either narrowly or widely, the hypothesis $H_{0,3}$ cannot be rejected at a confidence level of 95%: No statistically significant differences with respect to correctness can be observed.
Testing for Differences in Cognitive Load: As stated before, we asked all respondents to rate the cognitive load of the two modeling tasks after they had been performed. We rated this complexity on a 7-point Likert scale, ranging from 'very low' to 'very high'. The values that were obtained in this way were in conformance with the requirements for a standard t-test. The application of this test resulted in a p-value of 0.735 (> 0.05). Consequently, hypothesis $H_{0,4}$ cannot be rejected at a confidence level of 95% or, phrased differently, no statically significant difference can be established between the cognitive load between the groups.

4.3 Discussion of Results

With respect to the four different performance indicators that were examined for differences, only accuracy indicates a significantly better performance for the group performing the sequential change task. In this case the obtained p-value of 0.042 is slightly below the cut-off value of 0.05. For all other indicators, i.e., correctness, speed, and cognitive load, no significant differences could be detected.

These outcomes point at the type of change not being an overly strong factor with respect to the maintainability of a process model. A significant difference is expected from a theoretical point of view, as the respondents were asked to carry out a change task on a process model that is captured with a technique that emphasizes a sequential view on the process. Therefore, we expected a change task that is captured in the same, sequential style to be performed easier or better than a circumstantial change task.

For the interpretation of these results we have to consider two major factors that we tried to neutralize. First, there are characteristics of the process modeling language that influence the ease of change. Arguably, BPMN process models can be rather easily changed in comparison to Petri nets, which require the alternation of places and transitions to be preserved. Accordingly, the size of our models in the experiment could have been too small for the effect of change type to materialize. Second, experiments like ours are strongly influenced by the process modeling expertise of the participants [15]. It might have been the case that our pre-test population was less proficient in process modeling, such that the selected models again could have been too simple for the experimental group.

There are alternative explanations. We purposefully chose change tasks of a different type, while ensuring that the graph-edit distance for solutions to the sequential and circumstantial tasks are the same. This might also be a hint that the graph-edit distance could be a much stronger factor for approximating the difficulty of a change requirement[1]. On the other hand, the number of respondents that has been involved in this experiment (15) is rather low, which makes statistical inferences hazardous due to the high impact of individual observations. Given such a small sample size we are only able to detect strong effects in the data. The impact of change type on accuracy seems to be such a strong effect. Finally, the familiarization phase during which all respondents could inspect the base model has been considerable. It could be argued that the remaining sense-making task (e.g., the interpretation of the change task) is a minor effort in the overall task. All these issues can only be settled satisfactorily by replicating this experiment with a larger respondent base, a shorter familiarization phase, and another set of change tasks.

5 Related Work

In this section we first discuss factors that influence the usability of process models and which we strived to keep constant. Then, we relate works to our experiment that investigate the impact of representational characteristics of a model on comprehension and maintainability.

There are several factors influencing process model usability including domain knowledge, tool support, and selection of tasks. Prior *domain knowledge* can be an advantage for participants of an experiment. People may find it easier to read a model about the domain they are familiar. It is known from software engineering that domain knowledge affects the understanding of particular code [11]. Its impact is neutralized in experiments by choosing a domain that is usually only known by experts. *Tool support* plays a fundamental role in fostering process

[1] Note that in practical settings the graph-edit distance of a change often cannot be assessed beforehand, so this insight is mostly of a theoretical value.

changes and hiding the complexity behind high-level change operations [18, 23]. We tried to neutralize the impact of tool support by offering only the most atomic change operations. The *selection of experimental tasks* can also have an impact on the validity of an experiment. It has been shown that understanding tasks can vary in their degree of difficulty even if they relate to the same model [14]. We tried to neutralize the impact of the tasks by choosing tasks of equal graph-edit distance.

Our experiment can be related to various experiments that investigate how characteristics of a particular problem representation influences problem-solving performance. We have already referred to work on software program comprehension [6, 7, 8]. It showed that declarative programs are better at explicating circumstantial information while imperative programs more handily show sequential information. This work is particularly interesting as it contributed to settling a long debate on whether declarative or imperative computer programs should be considered to be superior. Confirming results are reported among others in [1, 10, 16] where the impact of a particular information representation is tested as a factor of comprehension performance. This exactly matches the more general argument of cognitive fit theory, which states that a problem representation should match the problem solving task [22].

6 Summary and Conclusion

In this paper we investigated the relationship between the type of change requirement and the performance of modifying a process model. We designed and conducted an experiment in which graduate students received sequential and circumstantial change requirements and changed a BPMN model accordingly. The results show that there is partial support for the type of change being a factor for process model maintainability. Our findings are of significant importance to future experiments on business process maintainability. Apparently, the type of change requirement has an impact on the ease of changing the model. Experiments that do not investigate this effect must neutralize its impact either by using only one type of change requirement or by making a balanced selection of change tasks from both types.

In future research we aim to replicate this experiment with more students in a similar classroom setting. It will be interesting to check whether a larger sample size will reveal effects that have been too weak to be detected with our small sample. Furthermore, we plan to conduct experiments that vary the set of change operations that are offered to the modeler. While we currently provided only basic change operations in this experiment, it has to be investigated whether complex changes can be easily made once high-level change operations are available. This argument points also to the need for further research into change operations. We consider it to be an important question how circumstantial change requirements can be directly translated into corresponding change macros.

Acknowledgement

This research is supported by the Technology Foundation STW, applied science division of NWO, and the technology program of the Dutch Ministry of Economic Affairs.

References

1. R. Agarwal, P. De, and A.P. Sinha. Comprehending object and process models: An empirical study. *IEEE TSE*, 25(4):541–556, 1999.
2. W. Bandara, G. Gable, and M. Rosemann. Factors and measures of business process modelling: model building through a multiple case study. *EJIS*, 14(4):347–360, 2005.
3. G. Canfora, F. García, M. Piattini, F. Ruiz, and C.A. Visaggio. A family of experiments to validate metrics for software process models. *JSS*, 77(2):113–129, 2005.
4. R. Dijkman, M. Dumas, and L. García Bañuelos. Graph Matching Algorithms for Business Process Model Similarity Search. In *Proc. BPM '09*, pages 48–63, 2009.
5. D. Fahland and H. Woith. Towards process models for disaster response. In *Proc. BPM '08*, pages 254–265, 2008.
6. D.J. Gilmore and T.R.G. Green. Comprehension and recall of miniature programs. *IJMMS*, 21(1):31–48, 1984.
7. T.R.G. Green. Conditional program statements and their comprehensibility to professional programmers. *Journal of Occupational Psychology*, 50:93–109, 1977.
8. T.R.G. Green and M. Petre. Usability analysis of visual programmingenvironments: A 'cognitive dimensions' framework. *J. of Vis. Lang. and Computing*, 7(2):131–174, 1996.
9. A.S. Guceglioglu and O. Demirors. Using Software Quality Characteristics to Measure Business Process Quality. In *Proc. BPM'05*, pages 374–379, 2005.
10. J. Hahn and J. Kim. Why are some diagrams easier to work with? effects of diagrammatic representation on the cognitive integration process of systems analysis and design. *ACMTCHI*, 6, 1999.
11. A. Lakhotia. Understanding someone elses code: Analysis of experiences. *JSS*, 23(3):269–275, 1993.
12. C. Li, M. Reichert, and A. Wombacher. On measuring process model similarity based on high-level change operations. In *Proc. ER'08*, pages 248–264, 2008.
13. L.A. Marascuilo and R.C. Serlin. *Statistical Methods for the Social and Behavioral Sciences.* Freeman New York, NY, 1988.
14. J. Melcher, J. Mendling, H. A. Reijers, and D. Seese. On measuring the understandability of process models. In *Proc. BPM '09*, 2009.
15. J. Mendling, H. A. Reijers, and J. Cardoso. What makes process models understandable? In *Proc. BPM '07*, pages 48–63, 2007.
16. T.G. Moher, D.C. Mak, B. Blumenthal, and L.M. Leventhal. Comparing the Comprehensibility of Textual and Graphical Programs: The Case of Petri Nets. In *Empirical Studies of Programmers: Fifth Workshop*, pages 137–161, 1993.
17. J. Pinggera, S. Zugal, and B. Weber. Investigating the process of process modeling with cheetah experimental platform. In *Proc. ER-POIS 2010 (to appear)*, 2010.
18. M. Reichert and P. Dadam. ADEPT$_{flex}$ – Supporting Dynamic Changes of Workflows Without Losing Control. *JIIS*, 10(2):93–129, 1998.

19. S. Siegel and N.J. Castellan. Nonparametric Statistics for the Social Sciences. *McGraw-Hill*, 1988.
20. W.M.P van der Aalst. The Application of Petri Nets to Workflow Management. *The Journal of Circuits, Systems and Computers*, 1998.
21. I. Vanderfeesten, H.A. Reijers, and W. M. P. van der Aalst. Evaluating workflow process designs using cohesion and coupling metrics. *Computers in Industry*, 59(5):420–437, 2008.
22. I. Vessey. Cognitive Fit: A Theory-Based Analysis of the Graphs Versus Tables Literature*. *Decision Sciences*, 22(2):219–240, 1991.
23. B. Weber, M. Reichert, and S. Rinderle-Ma. Change patterns and change support features - enhancing flexibility in process-aware information systems. *DKE*, 66(3):438–466, 2008.
24. C. Wohlin, R. Runeson, M. Halst, M.C. Ohlsson, B. Regnell, and A. Wesslen. *Experimentation in Software Engineering: an Introduction.* Kluwer, 2000.

A. Change Descriptions Used in the Experiment

Sequential description. Customs of the host country may deny clearance of equipment after presenting equipment at customs or after demonstration of devices. If equipment is not cleared by customs of the host country, the task force members try to convince customs officials to clear the equipment with incomplete documents. In the meantime, task force members contact their partners to trigger support from higher-ranked authorities of the host country. If the customs officials finally clear the whole equipment by negotiation and support, the equipment is transported to a storage location. In the other case, equipment is usually not cleared because of incomplete documents for some parts of the equipment. Those parts that have been cleared are transported to the storage location, whereas the missing documents for the remaining parts are retrieved from the office in Germany. Once these documents are available, the remaining parts of the equipment are transported to the storage location as well.

Circumstantial description. Usually, equipment transport capacity is not available immediately. Therefore, the process is adapted to ensure efficient handling of the equipment. The task force team members travel in split groups to the destination. A first group flies to the host country ahead of the equipment right away. After having presented itself at the immigration it takes care of road maps, renting of vehicles, and organizing accommodation. In the meantime, a second group handles all equipment logistics in Germany and then flies to the disaster area independently of the equipment. Eventually, the second group passes immigration and contacts the other task force team members. In the meantime, the second group also contacts local geologists, if there is a local institution with geologic know-how. The equipment is cleared in the host country as soon as it arrives. The whole equipment handling in the host country including customs is done by the second group of task force members. The first and the second team synchronize after their respective processes and transport the cleared equipment to the storage location.