



HAL
open science

Integrating Process Modeling Methodology, Language and Tool – A Design Science Approach

Jörg Becker, Nico Clever, Justus Holler, Johannes Püster, Maria Shitkova

► **To cite this version:**

Jörg Becker, Nico Clever, Justus Holler, Johannes Püster, Maria Shitkova. Integrating Process Modeling Methodology, Language and Tool – A Design Science Approach. 6th The Practice of Enterprise Modeling (PoEM), Nov 2013, Riga, Latvia. pp.221-235, 10.1007/978-3-642-41641-5_16 . hal-01474749

HAL Id: hal-01474749

<https://inria.hal.science/hal-01474749v1>

Submitted on 23 Feb 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

Integrating Process Modeling Methodology, Language and Tool – A Design Science Approach

Jörg Becker, Nico Clever, Justus Holler, Johannes Püster, Maria Shitkova

University of Muenster – ERCIS
Leonardo-Campus 3, 48149 Münster, Germany

{firstname.lastname}@ercis.uni-muenster.de

Abstract. Providing high quality process models in a timely manner can be of major impact on almost all process management projects. Modeling methodologies in the form of normative procedure models and process modeling guidelines are available to facilitate this cause. Modeling languages and according tools, however, do neglect the available methodologies. Our work searches to close this research gap by proposing a modeling environment that integrates insights from modeling methodologies with a modeling language and a tool. Main features are a simple modeling language that generalizes most existing languages, four layers of abstraction and semantic standardization through a glossary and use of attributes. Our approach allows for rapid preparation of modeling activities and ensures high model quality during all modeling phases, thus minimizing rework of the models. The prototype was evaluated and improved during two practical projects.

Keywords: Business Process Modeling, Business Process Modeling Tool, Process Modeling Methodology, Business Process Modeling Language

1 Introduction

Business process modeling has received considerable attention in practice and theory during the last decades. Following Becker and Kahn [1], “a process is a completely closed, time-logical sequence of activities that are required for working on a process-oriented relevant business object”. Process modeling as a form of conceptual modeling is guided by modeling methodologies, uses modeling languages and is supported by modeling tools [2]. Following software development terminology, the combination of methodology, language and tool will further be addressed as the modeling environment. Modeling methodologies are guidelines for the modeling process and available for different purposes, for example business process reengineering [2] or process-oriented reorganization [3]. Modeling languages are available in abundance, ranging from early languages such as EPC [4] and the more formal Petri nets [5], up to BPMN [6] and UML Activity Diagrams [7]. Tool support for all modeling languages has become a necessity, and the tools have evolved up to integrated environments, so called business process management suites [8]. To ensure high model quality and, thus, to reduce costs, numerous methodologies and guidelines

are available [3, 9–11]. We, however, argue that methodology insights are neglected in modeling languages and tools because they contradict the high degrees of freedom in existing modeling languages. Therefore, the research question is, how can methodological guidelines and restrictions to process modeling be integrated in the modeling language and modeling tool. The purpose of this paper is to close this research gap by proposing a modeling environment that incorporates features of modeling methodologies into the modeling language and tool for efficient and effective model creation. We have to stress that we propose a complete modeling environment and not “yet another modeling language” [12]. Furthermore, the modeling language used integrates with existing languages, such as EPC and BPMN, as it generalizes the languages by using a common subset of elements.

The development process follows the design science research methodology proposed by Peffers et al. [13], as both language and tool are design science artifacts. The remainder of the paper is structured as follows: The design science approach is described in the second section. An overview on related work is carried out in the third section to derive the research gap. Section four presents the proposed modeling environment and the results of its practical evaluation are described in section five. The paper is concluded with a discussion of the results and an outlook in section six.

2 Research Methodology

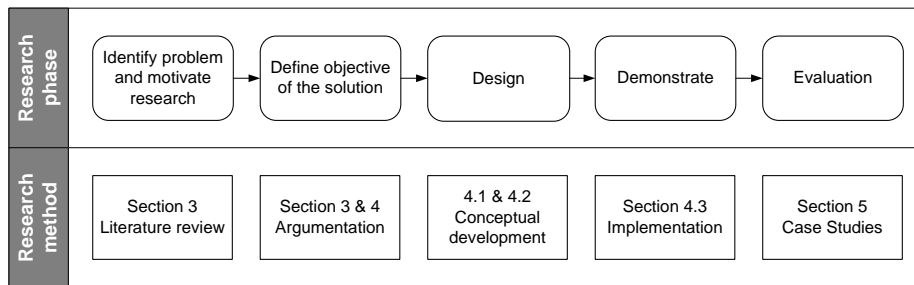


Fig. 1. Research methods applied in this paper mapped to research phases

The modeling environment proposed in this paper was developed according to the design science research methodology (DSRM) introduced by Peffers et al. [13]. The DSRM consists of six consecutive phases of which the last phase, communicating the research results, is achieved with this article. The remaining five phases, along with the respective research method, are depicted in Fig. 1. To identify the problem and motivate our research, we conducted a keyword based database search [14, 15]. Based on the literature review we provide a line of argumentation to define the objectives of the proposed solution. In order to enable the subsequent implementation of our solution, a conceptual model of the modeling environment is developed in the design phase. Afterwards, the artifact was implemented as a web-based tool to demonstrate its practicability. To demonstrate the applicability of the modeling environment, we present the findings of two case studies projects with archetypical small and medium sized enterprise in which our prototype was used.

3 Related Work

The literature review revealed several noticeable procedure models for process modeling differentiating themselves in number and naming of the phases, the purpose and intention of the procedures are nevertheless comparable. Kettinger et al. [2], performed an analysis of 25 business process reengineering (BPR) methodologies used in practice by different organizations and constructed a generalized procedure model with six phases. Due to its consolidating nature, modeling related steps are mostly concealed within the general steps of initiation, diagnosis and redesign. Allweyer [16] describes a BPR procedure model of five phases that could also be a special case of the framework by Kettinger et al. [2]. A more abstract model is proposed by Schmelzer and Sesselmann [17] that hides modeling related activities within the phases of process identification, implementation, control and optimization. In the work of Becker et al. [3], a procedure for process-oriented reorganization projects is described that consists of seven phases, including phases for the preparation of modeling, process framework design, as-is modeling and to-be modeling.

As depicted in Fig. 2, the procedures can be grouped into four steps and differentiated in the proportion of modeling related activities. All procedures start with a preparation step that is not directly related to model construction. It is followed by a step of process modeling, analysis and optimization that includes a high proportion of modeling related activities. The subsequent implementation and evaluation of the processes is less focused on the models. As modeling activities are the initial steps before the actual implementation of changes in the organization, it is important to assure their quality at earlier phases, because error correction at later stages is expensive [18, 19]. We will, therefore, focus on the phases of preparation and modeling, analysis and optimization, because they determine the quality of generated models and, thus, influence the success and costs of the whole project.

Normative approaches to ensure high model quality during preparation of modeling and the actual modeling activities are available in the form of guidelines. Six guidelines of modeling (GoM) are proposed by Becker et al. [20]. They are divided into mandatory and optional guidelines that give general advice on how modeling should be conducted. Because of their general nature, the GoM have been criticized of being too theoretical and abstract, meaning they cannot directly be operationalized [9]. Mendling et al. [9], therefore, present seven process modeling guidelines (7PMG) that provide concrete actions for modelers. Complementing the guidelines on what to do, pitfalls to avoid are listed by Rosemann [10, 11]. A detailed overview of the available guidelines and pitfalls is given in Fig. 3. Because the GoM are more abstract, all guidelines and pitfalls presented in Mendling et al. [9–11] are related to at least one or more GoM.

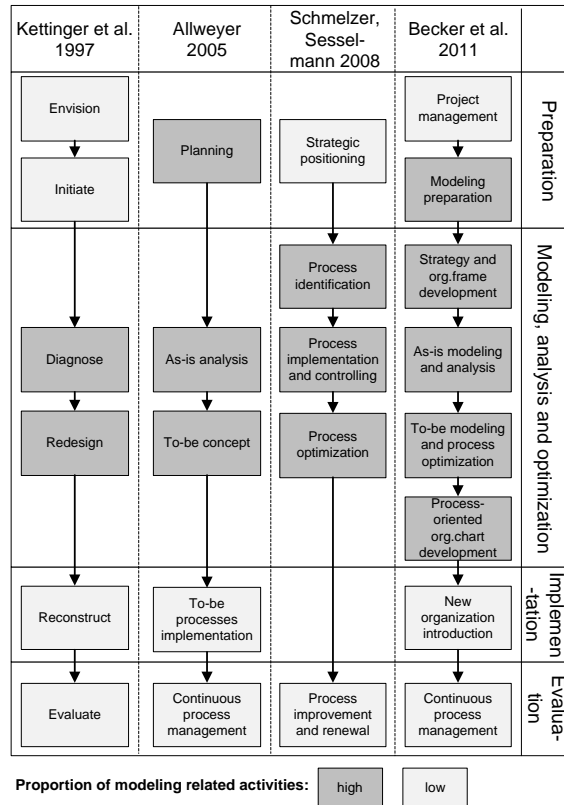


Fig. 2. Business process management procedures (Source: [2, 3, 16, 17])

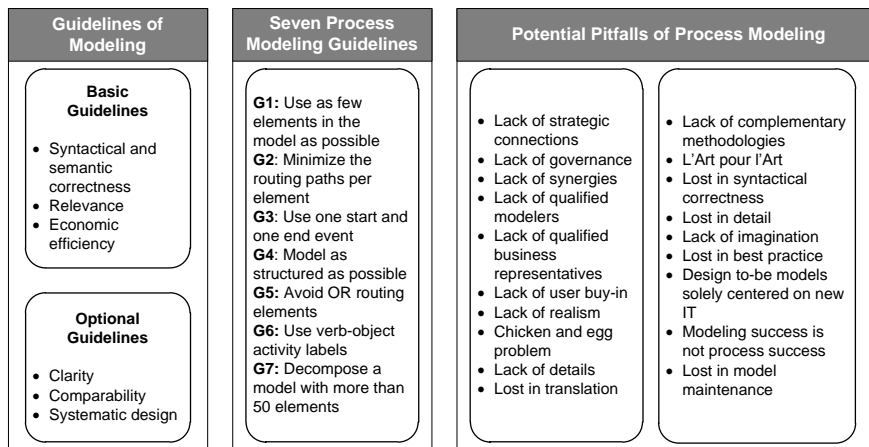


Fig. 3. Guidelines and pitfalls for process modeling

While adherence to the described guidelines can increase model quality and reduce costs, empirical results also indicate that it can cause an increase in perceived usefulness, perceived ease of use and satisfaction with the modeling language [21]. Existing process modeling languages, such as Petri nets, EPC, BPMN and UML Activity Diagrams, however, offer a high degree of freedom by the design of their meta-models. Restrictions, e. g. on how language elements like activities, connectors and flow should be used, should therefore be enforced by the modeling tool.

The process modeling tool market is vast and ranges from simple tools like Microsoft Visio up to business process management suites with many functionalities that surplus modeling functionality (see [8] and [21] for an overview on existing tools). Most tools support user in creating, standardizing, storing and sharing process models, i. e. they offer a replacement for pen and paper. Additional functionality for the analysis of process models has received academic attention lately and is already available in some tools [22]. As outlined by Mendling et al. [9], the modelers are, however, hardly supported in creating high quality and analyzable models because the available guidelines are not enforced or too abstract. We argue that this is caused by the degree of freedom in the modeling languages, which the tool vendors do not restrict [21]. An exception is an approach for the standardization of model element labels by means of naming conventions for both allowed words and phrase structures that are enforced during the modeling process [23].

We conclude that most of the available guidelines are not enforced in available tools, because tool vendors do not to limit the degree of freedom which existing modeling languages allow for. We aim to close this research gap by proposing an integrated modeling environment that tries to enforce the guidelines summarized in Fig. 3.

4 The Modeling Environment

The proposed approach consists of a modeling language and a tool and was built with the goal to integrate principles from existing methodologies and guidelines. The first design principle of the environment was simplicity to keep business process modeling projects as simple as possible and as complex as necessary. Simplicity of the modeling language reduces the degrees of freedom and therefore fosters model quality by taking the available modeling methodologies and guidelines into account. Simplicity of the modeling tool enables a wider group of users to utilize the tool and, thus, facilitates distributed modeling, which reduces modeling costs [21]. The second design principle is transparency, because the guidelines should be enforced already during modeling without the knowledge and additional interaction of the modeler.

The targeted audience of the modeling environment encompasses all the enterprises that choose or were chosen to model, discuss and analyze their processes. Our environment, however, was not built to model workflow processes that can directly be transformed into executable application code, because workflow modeling and organization-oriented process modeling differ substantially in the required level of detail [3]. Opening the environment for modeling in such detail contradicts the goal to reduce the degree of freedom and to take the modeling guidelines into account. The

environment thus does not support process modeling as a preparation for the implementation of a workflow engine. It aims to support process modelling in projects focussed on the organizational issues, for example BPR, organizational documentation, knowledge management or software selection.

To address the 7PMG and the GoM as summarized in Table 1, the following rationales of the environment are presented in the upcoming sections:

- Simple syntax of the modeling language
- Layer structure to control the leveling of modeling detail
- Variants and process element references to reduce model complexity
- Glossary and semantic standardization to eliminate naming conflicts
- Attributes to adapt the environment to a concrete modeling project

While certain guidelines are enforced by the modeling tool, or imply impossible by language design, other aspects, such a the amount of elements used per model, can only be facilitated but not completely restricted.

Table 1. Guidelines and pitfalls for process modeling

7PMG		Guidelines of Modeling	Enforced (✓) or Facilitated (o) by:	
1	Use as few elements in the model as possible	Clarity, Comparability, Economic efficiency	o	Syntax, Structure
2	Minimize the routing paths per element	Clarity, Correctness	✓	Syntax
3	Use one start and one end event	Clarity, Comparability Systematic design	o	Syntax
4	Model as structured as possible	Clarity, Comparability, Economic efficiency	✓	Structure, Variants
5	Avoid OR routing elements	Clarity	✓	Syntax
6	Use verb-object activity labels	Clarity, Systematic design	✓	Glossary
7	Decompose a model with more than 50 elements	Clarity, Systematic design	o	Syntax, Structure

4.1 Syntax and Structure of the Modeling Language

While Petri-nets use 3 model elements (places, transitions, flow), EPC features over 20 elements and BPMN offers more than 90 elements, our modeling language only allows for two constructs: activities and flow. This decision was supported by empirical evidence which suggests that modelers use fractions of the available elements in other languages [24, 25]. Activities and flow are constructs that are available in virtually all process modeling languages. Therefore, we do not propose a new language, but use a subset of existing modeling languages. The meta-model of the proposed language is depicted in Fig. 4.

Our concept of flow does not include routing logic in form of connectors in order to increase model quality [25]. Activities are nonetheless allowed to have more than one predecesing or succeeding process element but cyclic edges are prohibited. The flow direction is top to bottom by convention within a model and only one start and end activity is allowed. Due to the secondary role of flow, all modeling detail is included in the activities and their attributes. Events are not available, because they are always connected to activities and their additional value can be included in a detailed description of the process elements, which we also call process bricks.

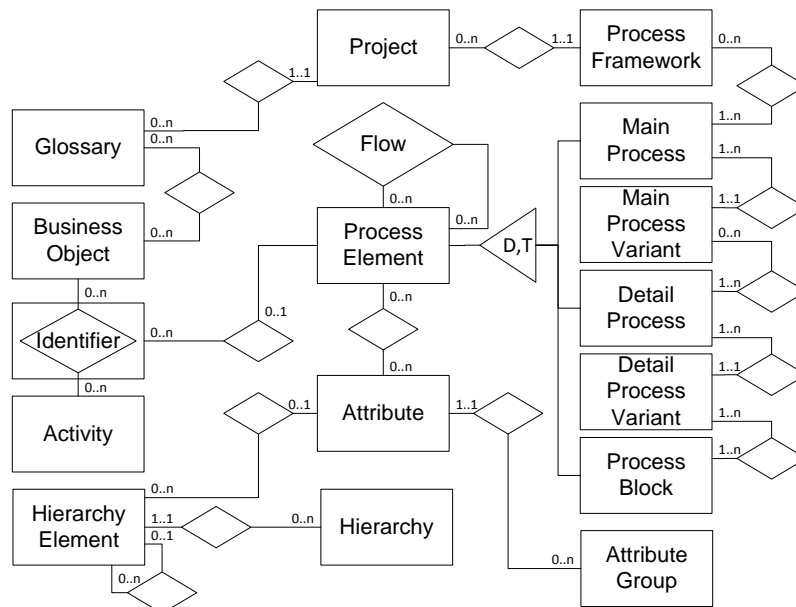


Fig. 4. Meta-model of the modeling language

Modeling within the environment is structured into four distinct layers, which enforce a comparable degree of modeling detail [9, 20]. An example of an instantiation of the meta-model is depicted in Fig. 5. The first layer is the process framework that depicts the process landscape and consists of process elements, which can be freely arranged and shaped to represent arbitrary process frameworks and thus do not require to be connected by control flow. Each process brick in the framework represents a main process. Main processes again contain process bricks, and explicitly require control flow. Each process brick in a main process in turn is a detail process. The difference between main and detail process therefore is only on a level of detail. Each detail processes consist of several process bricks.

To further reduce branching and the number of elements per process model, our modeling environment allows referencing existing elements of the same level and variants. The process brick “Print invoice”, for example, can be defined in in the detail process “Handle customer order” and reused by reference in the detail process “Revise complaint”. Variants can exist on the main process and detail process level

and allow the modeler to depict substantially different variations of the same process. The main process “Send invoice” could for example be constituted by the two variants “Send electronic invoice” and “Send manual invoice”, which do not share any process bricks. For a more detailed description, please see [26].

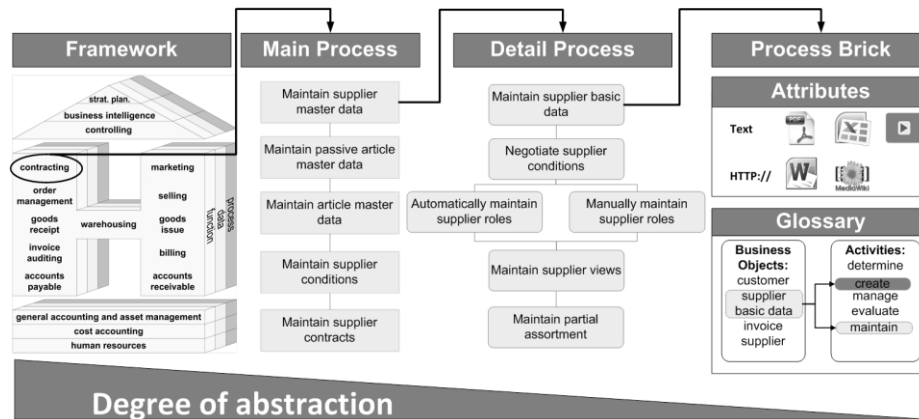


Fig. 5. Examples for framework, main process, detail process and process brick

4.2 Semantic Standardization through Glossary and Attributes

To semantically standardize the constructed models for analysis purposes and ensure high model quality, the proposed environment enforces naming conventions on the process brick labels. It builds on the approach by Delfmann et al. [23] and further extends it by eliminating the freedom of choice for phrase structures. Analogue to the syntax of our modeling technique, the semantic standardization uses the simplest structure available – the verb-object phrase structure. Phrase structures of this composition have been proven to be better understandable than other phrase structures [9]. In the context of process modeling, verb and object can furthermore be interpreted as activity and business object.

The modeling environment contains a glossary that consists of business objects, activities, and relations between both. The business object “supplier basic data”, for example, allows the activities “create” and “maintain”, but does not allow “determine”, which in turn could be used for “customer”. The naming conventions are enforced whenever a process brick is created or edited by the fact that the modeler has to use a combination of existing business object and activity to name the process brick, as shown in Fig. 5.. Similar to Delfmann et al. [23] or standardization approaches based on ontologies, the glossary requires a high initial effort of creation and subsequent maintenance. However, it allows to reuse existing content, such as domain specific business objects or generally applicable activities.

Besides their label, process bricks can own an arbitrary amount of attributes. There are a number of predefined attribute types available, which include text, number, selection, file, URL, reference and hierarchy. In order to constrain the amount of

attributes the attribute groups with the corresponding attributes are defined by an administrator and the normal user is only able to select from the offered set. Attributes of type hierarchy refer to hierarchies, which can also be modeled in the environment to document, for example, the organizational structures or hierarchies of the IT-architecture and use this structured attribute to annotate process elements. Similar to the glossary, attributes can be used for process bricks on all layers, from framework to process building block. Attribution is used to increase the expressiveness of our modeling environment, because it allows for the simple annotation of detailed information without requiring new language elements.

4.3 Implementation

The prototypical implementation of our modeling environment is a web-based business process modeling tool, as presented in [27]. It is implemented as a Ruby on Rails application and integrates a role and rights management concept to support distributed modeling in large scale modeling projects with many stakeholders. See Fig. 6 for a screenshot. Process bricks of all layers are labeled using the business objects and activities that can be maintained within the glossary. Analogously, attributes can be defined within the prototype and used on all layers they are assigned to. Hierarchies for organizational structure and IT-architecture can be created and maintained directly in the prototype.

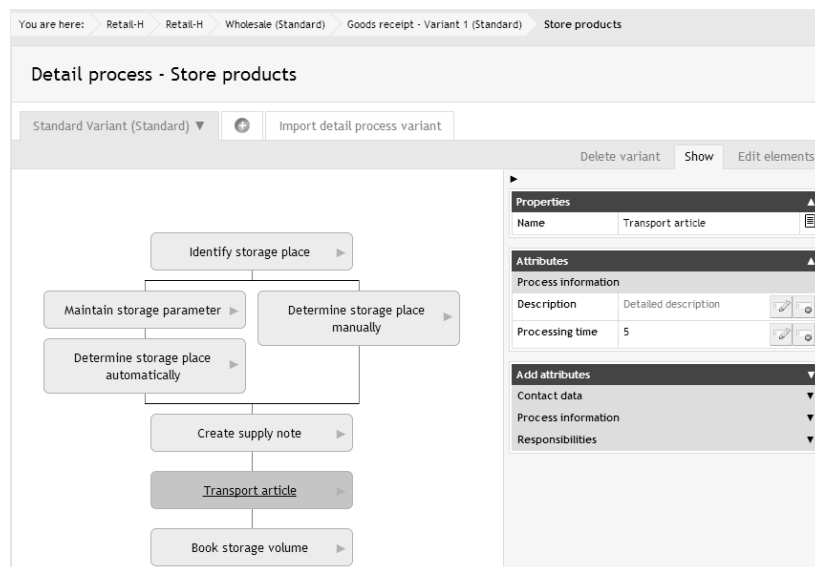


Fig. 6. Screenshot of detail process editing in the prototype

5 Case Studies

In order to evaluate the modeling environment we applied a prototypical implementation in two consulting projects that required business process modeling. Characteristics of the case study projects are summarized in Table 2. We conducted semi-structured interviews and used focus groups with the respective project leaders and key users to analyze the prototype. In both case studies, we focused on answering the question if the prototype fulfilled its purpose as efficient and effective modeling environment. Findings from the first case study were used to improve the prototype before the start of the second case study [28]. In accordance with our research methodology, we applied the enhanced prototype to the second case study and intend to keep up the iterative improvement procedure through further application.

Table 2. Case study project characteristics

	Case study 1	Case study 2
Modeling purpose	Business process reorganization, ERP selection	Organizational documentation, knowledge management
Project duration	Winter 2010 – Summer 2011	Winter 2011 – Summer 2012
Domain	Retail	Warehousing
Articles	~20.000, sports and fashion	~2.500, promotion material
Employees	>1.600	>250 worldwide
Customers	B2C customers	~6000 B2B customers
Processes documented	2 frameworks (store and headquarters), 13 main processes + 6 variants, 55 detail processes, 168 process building blocks	1 framework, 12 main processes + 10 variants, 49 detail processes, 133 process building blocks

5.1 Project 1: Business Process Reorganization and ERP Selection

The first case study was conducted within a business process reorganization and ERP selection project at a German sports and fashion retailer. The company sells sports and fashion goods in over 60 stores. The purpose of the project was the reorganization of processes in both, headquarters and stores, and the mutual consistent selection of a new ERP system to align processes and IT systems.

The project consisted of three phases. During the first phase, the as-is process models were documented. To-be processes were designed in the second phase that overlapped with the interrelated third phase of an ERP software selection process. All the phases were conducted by the company with the help of external consultants. Concerning the modeling methodologies presented in section three, the project encompassed modeling preparation and modeling conduction steps.

The modeling team consisted of external consultants and employees of the company headquarter, which were trained in the modeling environment during an initial

workshop. The as-is models were created on the basis of interviews in all company departments. As the company had not undertaken any modeling activities before the project, the processes were designed on the basis of a reference model for retail processes, already available in the prototype [29]. Due to the restrictive nature of the prototype, the team did not have to discuss modeling guidelines on syntax, object types to be used, naming conventions, degree of detail or layout conventions. All adaption of the prototype to the project goals was achieved through the definition of attributes. During the first as-is modeling project phase, these company-specific attributes were defined and later on enhanced during the second to-be modeling phase. To support the final ERP choice, attributes to measure the IT-related requirements and their fulfillment were added to the modeling environment and filled out during workshop meetings with all stakeholders and process-based vendor presentations.

The case study revealed that the reduction in preparation activities was perceived very positive by the whole modeling team and allowed for a fast start into the project. Company employees stated the simplicity of the modeling language as a driver to facilitate the initial acceptance of the project in all departments. The external consultants especially complimented the four-layer architecture, because it helped to standardize the degree of detail within the models and improved inter-model comparability. The effort to prepare the models for the semi-automated creation of an ERP requirements definition could, therefore, be kept low. The main argument to use the proposed modeling environment over other modeling languages or tools in new projects, stated by almost all team members interviewed, turned out to be the ability to adapt the environment to the modeling purpose through attributes, because all models could be re-used throughout the project phases by small additions to the attributes in use. Negative comments were received during the initial preparation of the glossary. The reservations were, however, partly dissolved for the most part during later stages of the project, when renaming tasks could be executed centrally. Major criticism was furthermore voiced towards the general usability of the prototype, in particular to glossary management and the element naming. All members of the modeling team stated export (Word, Excel export and XML import/export) and analysis features, such as statistics on the used attributes, as most important missing functionalities.

We concluded that our proposed modeling environment facilitated cost-efficient modeling by accelerating modeling preparation activities and achieved high acceptance in all departments because of the simple modeling language. Re-work by the external consultants could be minimized, because model quality was ensured during all phases of the project. Adaption of the prototype to the project could be achieved through extensive use of attributes that facilitated model re-use. Nonetheless, the case study revealed improvement points, in particular usability aspects and the need for model export and analysis features.

5.2 Project 2: Organizational Documentation and Knowledge Management

After the prototype had been improved, we conducted a second case study with a leading German wholesaler of promotional material. The company has been founded two decades ago and grown ever since. In order to keep pace with the changing processes and the growing employee base, they decided to document their processes and use the documentation for knowledge management purposes. Till now, the project only included an as-is modeling phase, but as the ERP system of the company will undergo a major update by the time of this work, the models will be used to derive software test cases.

Similar to the first project, the company had not documented any processes before the project. The project team was therefore supported by external consultants. The processes were created on the basis of retail reference processes with our prototype during interviews with employees from all departments.

The second case yielded results similar to the first project. the modeling language and the enforced modeling structure were perceived very positive by all project stakeholders. All further information could be expressed using the attributes, which again allowed for extensive reuse of the models for both organizational documentation and knowledge management. Problems for the preparation of the model-based software tests are not expected.

As the prototype was improved according to the findings from the first case study, discussions of the usability of the prototype gave much more positive responses but revealed additional issues that will be addressed in the next version of the prototype. The added export functionality could be fully tested during the project and was perceived very helpful by the modeling team. In opposite, functionality for analysis could not be properly evaluated, as the project did not encompass as-is analysis or to-be modelling steps, which could have required an tool support in model analysis.

We conclude that all requirements of both projects could be met and the discussions led to a substantial improvement of the prototype. The positive comments about productivity and model quality furthermore indicate that the design goals of the modeling environment could be met. As both companies represent typical small- or medium-sized enterprises, the application results are expected to be reproducible in other companies.

6 Summary, Limitations and Outlook

This paper presented a modeling environment, consisting of a language and a tool that integrate ideas from existing methodologies. It combines a simple modeling syntax that generalizes many existing languages with a four-layer structure to control the degree of modeling detail. The use of a glossary and attributes allows for the creation of standardized process models in a cost-efficient manner. The development process of the environment is based on the design science research methodology by [13] and the resulting prototype advances the current state of the art, because it transparently operationalizes process modeling guidelines that have been proposed in literature. The prototype was successfully evaluated in two modeling projects as an easy and fast to use modeling environment for beginners but also for advanced modelers who furthermore especially appreciated the standardization regarding naming and level of detail.

The environment is limited to business modeling and analysis purposes, such as business process reengineering or knowledge documentation, and not applicable for workflow modeling. It can, however, serve as a starting point for more detailed modeling by integrating workflow models as attributes of the process bricks. The case studies on the prototype moreover only cover German companies and, thus, did not consider social factors, such as values and beliefs. Moreover, both cases were conducted within companies that had no modeling experience beforehand. An application within a company with an existing modeling landscape would be needed to reveal potential issues in model transformation. Similarly, we did not compare our approach to existing modeling languages and tools. Such a comparison could be useful to further improve the proposed environment.

During the application, we also identified potential for further research. One aspect that requires further investigation is the four-layer architecture and we intend to evaluate how it performs against structures with a different number of layers. The cases also revealed challenging usability issues, which we already address in research [30] and subsequently intend to implement in the prototype. The proposed glossary also raised need for improvements, because it was the main source of reservations in our case studies. We intend to identify methods to improve the glossary management, for example by including existing verb-hierarchies. The prototype furthermore serves as a starting point for current research on model version control [31] and modeling support features such as auto-suggestion. As it is the underlying idea behind our work, future research will also focus on the use of attributes in process modeling and the relation of attribute-based complexity with modeling language-based complexity.

References

1. Becker, J., Kahn, D.: The Process in Focus. In: Becker, J., Kugeler, M., and Rosemann, M. (eds.) *Process Management: A Guide for the Design of Business Processes*. pp. 3–13. Springer, Berlin (2011).
2. Kettinger, W.J., Teng, J.T.C., Guha, S.: *Business Process Change: A Study of Methodologies, Techniques, and Tools*. *MIS Quarterly*. 21, 55–80 (1997).
3. Becker, J., Kugeler, M., Rosemann, M.: *Process Management: A Guide for the Design of Business Processes*. Springer, Berlin (2011).
4. Scheer, A.-W.: *Aris - Business Process Modeling*. Springer, Berlin (2000).
5. Murata, T.: Petri nets: Properties, analysis and applications. *Proceedings of the IEEE*. 77, 541–580 (1989).
6. OMG: Business Process Model and Notation (BPMN), <http://www.omg.org/spec/BPMN/2.0/>.
7. OMG: Unified Modeling Language (UML), <http://www.omg.org/spec/UML/2.4.1/>.
8. Gartner: *Magic Quadrant for Business Process Management Suites*, (2010).
9. Mendling, J., Reijers, H., van der Aalst, W.M.P.: *Seven Process Modeling Guidelines (7PMG)*. *Information and Software Technology*. 52, 127–136 (2008).
10. Rosemann, M.: Potential pitfalls of process modeling: part A. *Business Process Management Journal*. 12, 249–254 (2006).
11. Rosemann, M.: Potential pitfalls of process modeling: part B. *Business Process Management Journal*. 12, 377–384 (2006).
12. Wand, Y., Weber, R.: *Research Commentary: Information Systems and Conceptual Modeling - A Research Agenda*. *Information Systems Research*. 13, 363–376 (2002).
13. Peffers, K., Tuunanen, T., Rothenberger, M.A., Chatterjee, S.: *A Design Science Research Methodology for Information Systems Research*. *Journal of Management Information Systems*. 24, 45–77 (2007).
14. Webster, J., Watson, R.T.: *Analyzing the Past to Prepare for the Future: Writing a Literature Review*. *MIS Quarterly*. 26, xiii–xxiii (2002).
15. Vom Brocke, J., Simons, A., Niehaves, B., Riemer, K., Plattfaut, R., Clevn, A.: *Reconstructing the Giant: On the Importance of Rigour in Documenting the Literature Search Process*. *Proceedings of the ECIS 2009*. pp. 2206–2217. , Verona (2009).
16. Allweyer, T.: *Geschäftsprozessmanagement*. W3L-Verlag, Herdecke, Bochum (2005).
17. Schmelzer, H.J., Sesselmann, W.: *Geschäftsprozessmanagement in der Praxis*. Carl Hanser Verlag, München (2008).
18. Moody, D.: Theoretical and practical issues in evaluating the quality of conceptual models: current state and future directions. *Data & Knowledge Engineering*. 55, 243–276 (2005).
19. Vergidis, K., Tiwari, A., Majeed, B.: *Business process analysis and optimization: beyond reengineering*. *IEEE Transactions on Systems, Man, and Cybernetics*. 38, 69–82 (2008).
20. Becker, J., Rosemann, M., von Uthmann, C.: *Guidelines of Business Process Modeling*. In: van der Aalst, W., Desel, J., and Overweis, A. (eds.) *Business Process Management: Models, Techniques and Empirical Studies*. pp. 30–49. Springer, Berlin et al. (2000).
21. Recker, J.: “Modeling with tools is easier, believe me”—The effects of tool functionality on modeling grammar usage beliefs. *Information Systems*. 37, 213–226 (2012).
22. Delfmann, P., Herwig, S., Lis, L., Stein, A., Tent, K., Becker, J.: *Pattern Specification and Matching in Conceptual Models. A Generic Approach Based on Set Operations*. *Enterprise Modelling and Information Systems Architectures*. 5, 24–43 (2010).
23. Delfmann, P., Herwig, S., Lis, L.: *Unified Enterprise Knowledge Representation with Conceptual Models - Capturing Corporate Language in Naming Conventions*. *30th International Conference on Information Systems (ICIS 2009)*. , Phoenix, Arizona, USA (2009).

24. Siau, K., Erickson, J., Lee, L.: Theoretical vs. Practical Complexity. *Journal of Database Management*. 16, 40–57 (2005).
25. Zur Muehlen, M., Recker, J.: How Much Language is Enough? Theoretical and Practical Use of the Business Process Modeling Notation. *Proceedings 20th International Conference on Advanced Information Systems Engineering*. Springer, Montpellier, France (2008).
26. Becker, J., Clever, N., Holler, J., Püster, J., Shitkova, M.: Semantically Standardized and Transparent Process Model Collections via Process Building Blocks. *Proceedings of the The Fifth International Conference on Information, Process, and Knowledge Management - eKNOW 2013*. pp. 172–177. , Nice (2013).
27. Becker, J., Clever, N., Holler, J., Shitkova, M.: icebricks - Business Process Modeling on the Basis of Semantic Standardization. *Proceedings of the Design Science Research in Information Systems and Technologies (DESRIST) 2013*. , Helsinki (2013).
28. Tremblay, M., Hevner, A., Berndt, D.: The Use of Focus Groups in Design Science Research. *Design Research in Information Systems*. pp. 121–143. Springer US (2010).
29. Becker, J., Schütte, R.: A Reference Model for Retail Enterprises. In: Fettke, P. and Loos, P. (eds.) *Reference Modeling for Business Systems Analysis*. pp. 182–205. Idea Group Publishing (2006).
30. Becker, J., Clever, N., Holler, J., Shitkova, M.: Towards A Usability Measurement Framework for Process Modelling Tools. *Proceedings of the Pacific Asia Conference on Information Systems (PACIS) 2013*. , Jeju Island, Korea (2013).
31. Clever, N., Holler, J., Püster, J., Shitkova, M.: Growing Trees – A Versioning Approach for Business Process Models based on Graph Theory. *Proceedings of the European Conference on Information Systems (ECIS) 2013*. , Utrecht (2013).