HELP IS ON THE WAY – PROVIDING USER SUPPORT FOR EPC MODELLING VIA A SYSTEMATIC PROCEDURE MODEL

Research in Progress

Jannaber, Sven, Institute for Information Management and Information Systems, University of Osnabrück, Germany, sven.jannaber@uni-osnabrueck.de

Riehle, Dennis M., European Research Center for Information Systems, University of Münster, Germany, dennis.riehle@ercis.uni-muenster.de

Delfmann, Patrick, Institute for Information Systems Research, University of Koblenz-Landau, Koblenz, delfmann@uni-koblenz.de

Thomas, Oliver, Institute for Information Management and Information Systems, University of Osnabrück, Germany, oliver.thomas@uni-osnabrueck.de

Abstract

Process models and consequently business process modelling languages get more and more complex. This is especially true for the event-driven process chain (EPC), since the absence of a clearly defined standard renders EPC modelling difficult. On top, modelling itself is no trivial task. To address this issue, several frameworks and guidelines have emerged to support process modelling. However, most of them remain at a generic level. Currently, there is no user support with respect to the actual modelling process that is specific to the EPC language. To address these needs, the paper applies a design-oriented research approach and proposes a systematic procedure model specifically tailored towards EPC modelling as current outcome of this research in progress. We argue that the procedure model facilitates the modelling process and thus has the potential to increase model quality.

Keywords: business process modelling, EPC, modelling support, procedure model, user support.

1 The Need for User Support in EPC Modelling

The modelling of business processes is a core task of business process management (BPM) (Becker et al., 2010; Indulska et al., 2009) and part of the strategic agenda of many large organisations (PwC, 2011). Key to any successful modelling is the choice of an adequate business process modelling language (BPML) (Weske et al., 2004). One of the most dominant languages is the Event-driven Process Chain (EPC) introduced in 1992. Considered as a de-facto standard for business process modelling (Fellmann et al., 2013; Walterbusch et al., 2013), the EPC is well established in both research and practice and is still subject of ongoing discussion and refinement (Fellmann et al., 2013; Houy et al., 2009; Karhof et al., 2016). The lack of a clearly defined standard, however, renders modelling with the EPC difficult. In contrast to established modelling standards such as the Business Process Model and Notation (BPMN) (OMG, 2011), there exists no single source of guidance modellers can rely on when using the EPC. Instead, EPC research has become more and more fragmented over time, resulting in various propositions regarding EPC modelling, ranging from domain-specific dialects (Riehle et al., 2016b) and exchange formats (Riehle et al., 2016a) to various suggestions regarding syntax, semantics and pragmatics (Fellmann et al., 2013). Naturally, this situation increases the complexity of using the EPC in practice.

Despite the aforementioned challenges, little effort has been made to support and guide users in process modelling (Mendling et al., 2010), though researchers conduct experiments observing modellers (e.g., Pinggera et al., 2011). However, approaches emphasizing successive phases are more appealing to users (Claes et al., 2012) and positively influence modelling accuracy and speed (Claes et al., 2015). As cognitive skills of modellers to memorize reoccurring modelling patterns are limited (Weber et al., 2014), especially unexperienced modellers can benefit from a clear guidance model (Martini et al., 2016). Considering the EPC language, a concrete guidance model for transforming a given operational situation into sound EPC models is missing. Ultimately, the arising consequences are two-fold: On the one hand, practitioners get more and more confused and frustrated when being assigned to EPC modelling tasks due to the unstructured nature of the language. On the other hand, process usability and quality declines, which hamper model reusability and analysis.

User support and process quality have been frequently discussed in BPM literature. Various frameworks and guidelines have emerged that facilitate the modelling process (c.f. section 2 for an overview). However, although frameworks and guidelines provide support on a generic modelling level, there is a lack of immediate guidance on the modelling process specific to the EPC language, which is a) comprehensible and easy to apply and b) yields favourable results in the form of valid EPCs. In our work, we try to overcome challenges that arise with increasingly complex modelling notations and processes with our *research goal*: providing a systematic procedure model that supports users in EPC modelling. In doing so, a design-oriented research methodology has been carried out to reach that goal and, in addition, to answer the *research question*, how such a procedure model can ease modelling process complexity.

The paper is structured as follows: A brief overview over the EPC language and related work on process modelling support is presented in section 2. In section 3, the applied research methodology is described. Section 4 covers the design and development of the procedure model. Furthermore, section 4 outlines implications and further research need. The paper concludes with a summary of results in section 5.

2 The EPC Language and Best Practices in Modelling

The Event-driven Process Chain emerged from a joint research project of the Institute for Information Systems in Saarbrücken, Germany, and SAP. The first publication dates back to 1992 (Keller et al., 1992) and introduces the EPC as a modelling language consisting of alternating events and functions. While functions are business activities performed within an organisation, events represent the conditions before or after executing such a business activity. Using the operators AND (parallel split), OR (inclusive choice) and XOR (exclusive choice), the control flow can be split into multiple paths. The process flow is joined by modelling the corresponding operator of the same type.

While the first version of the EPC only considered the control flow of business activities during the execution of a business process, the EPC has soon been extended to include organisational resources like information objects, organisational units, IT systems or process refinements, which can be assigned to functions (Hoffmann et al., 1993). Many authors have extended and altered the EPC over the years. Riehle et al. (2016b) identify 14 different EPC variants in literature, for instance EPC*, the object-oriented EPC (oEPC) or the configurable EPC (C-EPC). The amount of different EPC variants shows that there is a diversified landscape of EPC modelling and Karhof et al. (2016) demonstrate that this diversified landscape is also reflected in the support for EPC in BPM tools available on the market.

The EPC modelling language is a flexible BPML with rather few elements compared to BPMN, for instance. While BPMN 2.0 comes with seven different activity types and over 50 different event types (OMG, 2011), EPC encompasses much fewer element types and is hence easier and faster to learn for process modellers. While this is certainly beneficial, it bears the risk that models can be less precise or even ambiguous due to the limited explicit expressiveness of EPC. Therefore, for modelling with EPC, it is particularly important to adhere to standards and best practices of process modelling.

In literature, there are several frameworks and standards, of which some are EPC-specific, and some are not. For example, the SIQ-Framework (Reijers et al., 2015) specifically addresses process quality in regard to syntactic, semantic and pragmatic criteria and the SEQUAL framework considers aspects of physical quality, empirical quality, syntactic quality, semantic quality, perceived semantic quality, pragmatic quality, social quality and organisational quality (Krogstie et al., 2006). In terms of the modelling process, the Guidelines of Modelling (GoM) (Becker et al., 2000) provide six general guidelines to improve the quality of models. More EPC-specific and user-oriented guidance can be found in the seven process modelling guidelines (7PMG) (Mendling et al., 2010). In the context of the EPC, efforts have been made to consolidate and ultimately specify the language on rule (Fellmann et al., 2013) or meta-model level (Jannaber et al., 2016). Additionally, Thomas (2009) proposes a descriptive instruction manual to transform a given business situation into an EPC process model.

Besides the mentioned frameworks and guidelines for process modelling, there are also tools and techniques, which provide the user help during the modelling process by suggesting elements the user could place into their model, so called recommender systems. For example, Koschmider et al. (2011) suggests constructs frequently modelled by different users, Smirnov et al. (2012) detects action patterns that structurally appear in different process models, Chan et al. (2012) suggests those business functions with the highest similarity score of the function’s context and Li et al. (2014) provides and efficient algorithm for calculating the distance between process fragments.

3 Applied Methodology

For the development of the proposed EPC instruction model, this paper adopts the Design Science Research Methodology (DSRM) introduced in Peffers et al. (2007). The applied methodology represents a structured approach for constructing IT artefacts tailored towards solving a given problem statement. Exemplary IT artefacts can range from prototypes and instantiations to models and methods (Hevner et al., 2004). This paper proposes a detailed procedure model for the construction of EPC process models as the central IT artefact. The design of the artefact follows the six steps of the DSRM, which consist of problem identification and motivation, objectives of a solution, design and development, demonstration and evaluation. It should be noted that the research presented in the following primarily focuses on the phases problem identification, objectives and design resp. development. Since the contribution is part of a larger scale project, an in-depth demonstration and evaluation will be part of future work.

Regarding the first step of the DSRM procedure, both the introductory section and Section 2 cover the necessity for modelling support, especially in the context of the EPC language. The objectives of a solution and preliminary work for the design and development step have been conducted by performing a multi-source analysis: A literature review has been performed on user support and process quality in business process modelling, as well as EPC modelling. Popular quality frameworks such as 7PMG and GoM have been taken into consideration. Both frameworks propose a set of generic guidelines that facilitate model quality and impose requirements on the modelling process. Specific emphasis has been put on the work of Thomas (2009) and Fellmann et al. (2013). In Thomas (2009), a descriptive procedure of how to transform a given textual description of a business process into an EPC process model is provided. In this description, the modeller is guided step-by-step through the decision making process with respect to elements, operators or resources to be used in order to represent business occurrences on the model level. Fellmann et al. (2013) integrate existing work in the field of EPC modelling with focus on syntax, semantic and pragmatic. For each, a consolidated set of rules is proposed that need to be followed for creating sound EPC models. The descriptive procedure form the basis of the presented artefact. It has been enriched with insights gained from the referred quality framework. To assure syntactical, semantic and pragmatic correctness of the resulting models, the set of rules have been highlighted within the procedure model. The design and development step is covered in Section 4. In preparation for an in-depth evaluation, a small scale pre-test has been conducted to yield initial results about the applicability of the procedure model towards modelling process complexity.

4 Towards a Systematic Procedure Model for EPC Modelling

Based on the consolidation and integration of previous work in the field of EPC modelling support, we propose an iterative procedure model to assist users in EPC modelling tasks. The model is presented in Figure 1 and consists of five phases that resemble the basic approach when constructing an EPC model:

Start event(s), the alternation of function and event modelling, optional resources and end event(s). Although EPC operators are essential constructs, they are not represented by a corresponding phase. Instead, the possible choices for modelling preceding operators are included in the main phases event and function. Every phase is comprised of a detailed set of steps. The procedure model is built on the premise of having a textual model of a given operational situation as an input. An EPC model reflecting the operational situation as a result of the procedure model is considered an output.

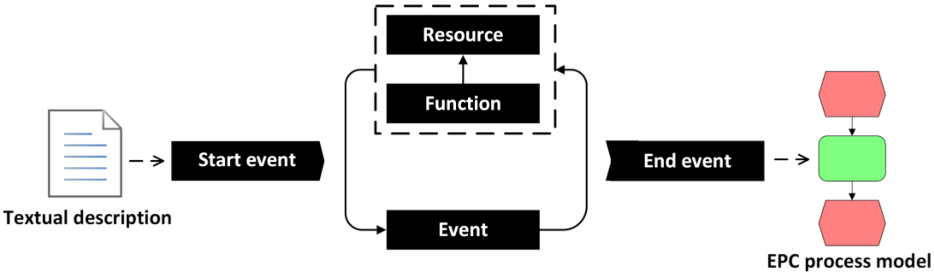

Figure 1. EPC modelling cycle

The model is designed towards the stated objectives of a) providing comprehensible and simple modelling instructions and b) transforming a given operational situation into a valid process model. To fulfil a), a graphical flow chart resp. procedure model has been chosen, since it provides an easy way of following all decisions that need to be made during modelling. In each phase, the user is guided through the decision process. As for objective b), the procedure model integrated insights gained from process quality frameworks and guidelines such as the GoM and 7PMG. Additionally, the model aims at adhering to EPC modelling rules in terms of syntax, semantic and pragmatic. Therefore, the procedure model specifically integrates the consolidated rules proposed in Fellmann et al. (2013).

Figure 2. Notation of the EPC instruction model

The steps of each procedure model are visualised using a generic notation, depicted in Figure 2. The core element is an instruction, which represents a concrete modelling instruction given to the user. Phases separate the procedure model into logical groups. A decision object indicates decisions a user has to make during modelling. Ultimately, marks are included to highlight certain EPC rules that have been taken care of. These rules directly refer to the rules provided by Fellmann et al. (2013). For example, a blue mark represents syntactical rules, where the number refers to the corresponding rule.

4.1 Steps of EPC Modelling

In the following, the phases of the developed procedure model for EPC modelling will be introduced in detail according to the aforementioned notation. Each phase consists of multiple guiding steps.

Start event. The start event phase is the starting point for every EPC modelling endeavour. The modelling of a start event is straightforward, encompassing three steps that are shown in Figure 3.

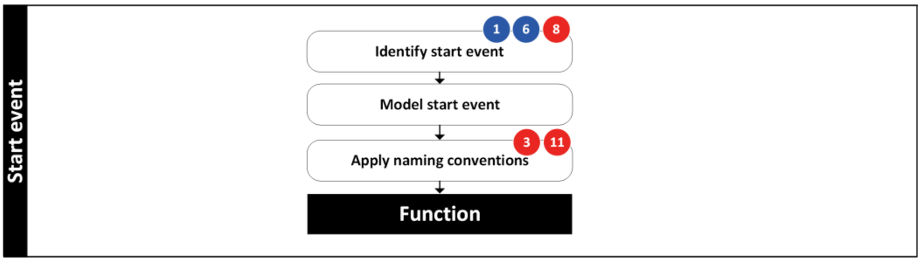

Figure 3. Modelling of the EPC start event(s)

First, the modeller needs to check the textual description of the process to be modelled. Adhering to the definition of the event element given earlier, a possible starting event may be any occurrence or state that triggers the process flow. After the starting event has been identified, the event has to be modelled and subsequently labelled. The modelling of the start event(s) is succeeded by the function phase.

In the case of multiple start events, the demonstrated start event phase needs to be repeated once per event. For each execution path, the EPC modelling cycle is applied separately, until a merge occurs.

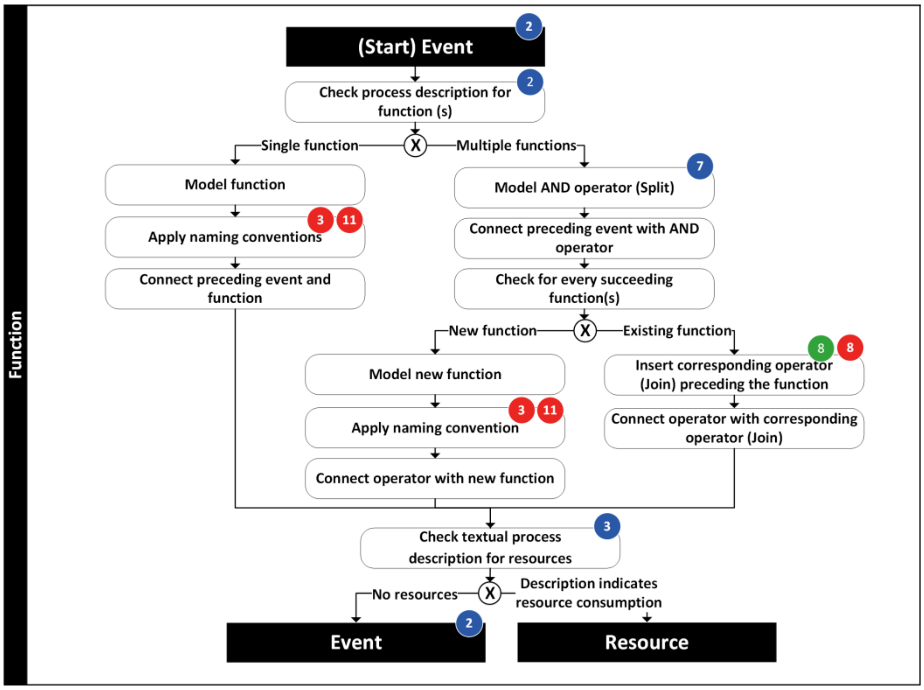

Figure 4. Modelling of the EPC function

Function. The modelling of a function succeeds either a start event or a conventional event. Figure 4 visualises the steps that are necessary to model an EPC function. Since the modelling of a function succeeds an event, there are only two possible options: The user can either model a single succeeding function or multiple functions that are split via an AND operator, since EPC rules only allow parallel splits here (Keller et al., 1992). In the first case, the single function has to be modelled and labelled accordingly. In the second case, the user needs to model the operator first.

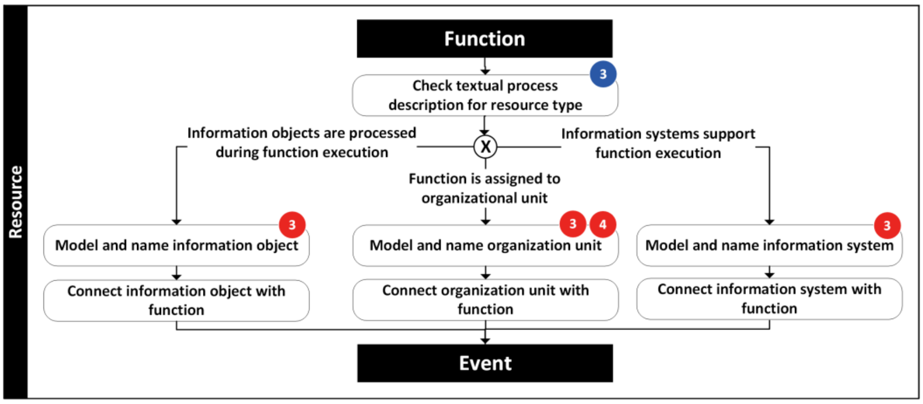

Figure 5. Modelling of EPC resources

Afterwards, two scenarios might occur for every function the modeller has identified. In the first case, the textual description of the process refers to a new function, which then has to be modelled and labelled. In the second case, the description refers to a function already included in the model. Here, the modeller needs to connect the process flow by inserting a new XOR operator (Join) preceding the existing function to fulfil EPC syntax restrictions. At the end of the function modelling phase, there is a check for resources. If the description indicates potential resource consumption, the resource phase is triggered. Otherwise, the function phase follows.

Resource. For EPC resources, we adhere to common ground in the EPC literature and therefore only account for three resource types. The resource phase as depicted in Figure 5 is straightforward and splits the decision process into three paths, each for a specific resource type.

Hence, the modeller identifies the required resource in the process description and assigns the resource object to the function. Afterwards, the user is guided towards the event phase.

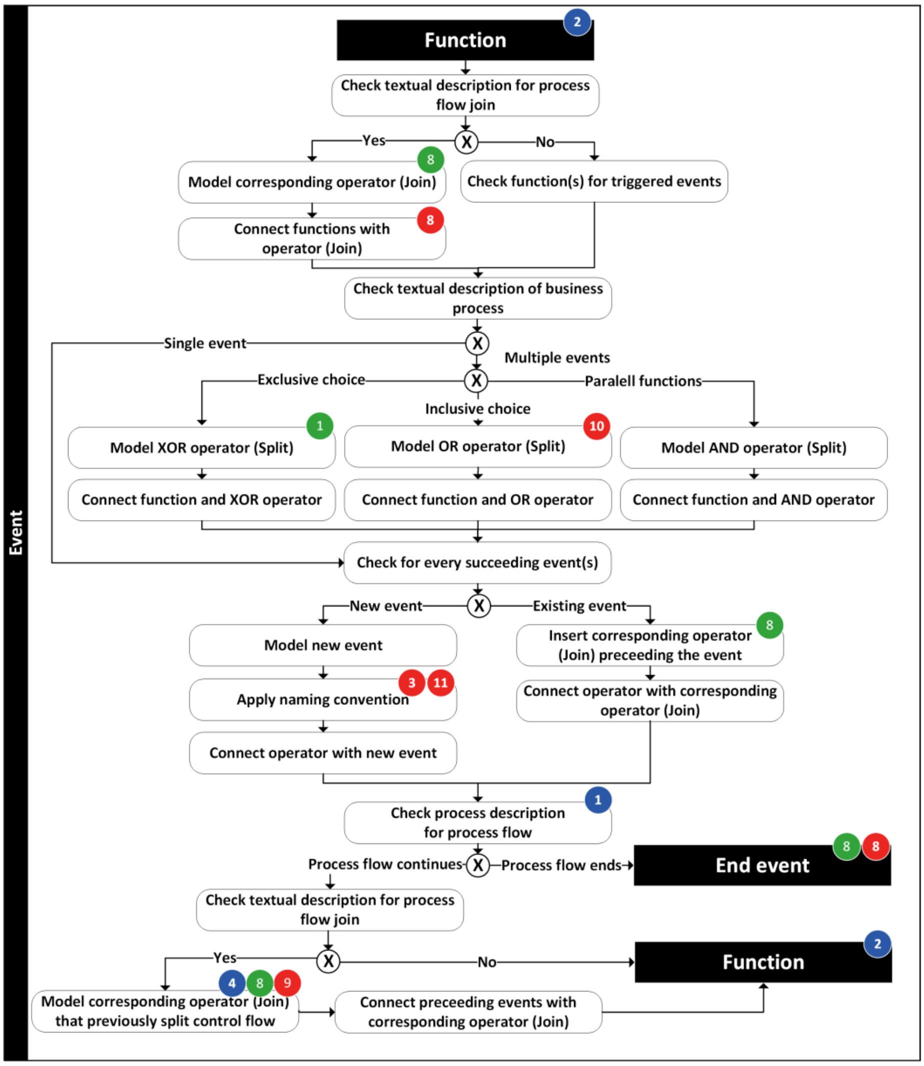

Figure 6: Modelling of the EPC event

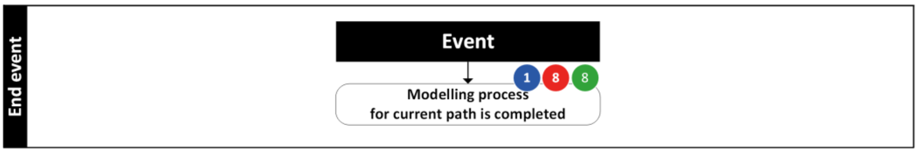

Figure 7. Modelling of the EPC end event

Event. The modelling of the event is a crucial phase of the proposed procedure model, since it includes most decisions regarding the modelling of operators. Additionally, the event phase is highly influenced by EPC rules, as highlighted in Figure 6. The modelling of an event succeeds, in any case, the function phase. Initially, it needs to be determined whether there is a previously modelled split of the process flow operator or, in the case of multiple start events, two process paths that need to be merged. In that case, a corresponding join operator is inserted. The main task of the phase is the check and subsequent decision-making in terms of events to be modelled. In the simplest case, there is no split and the process flow continues via a single event. If the process description indicates the need for multiple events, a subsequent check of the textual description reveals in which way the multiple functions need to be modelled. The user decides whether an exclusive choice (XOR operator), and inclusive choice (OR operator) or a parallel occurrence is required to fit the description. The corresponding operator is added and connected to the preceding function. Again, there are two scenarios the modeller might face.

In the first case, the event to be modelled is an entirely new one, hence not already included in the model. In the second case, the event already exists in the model. Subsequently, the previously chosen split connector needs to be connected to a corresponding join operator preceding the identified existing event. Towards the end of the event phase, the user needs to assess the process description whether the process flow ends. If so, the end event phase is triggered. If the process flow continues, the user then needs to check for a potential join. When there is no need for a join, the modeller enters the function phase again.

End event. If no further events indicating a continuing process flow can be identified in the textual description, the modelling process comes to an end. The corresponding steps are shown in Figure 7.

By reaching the end event phase, the modeller has finished the modelling process. As a result, the former textual process description has been transformed into a semi-formal EPC model.

4.2 Discussion and Further Research Need

Processes and BPMLs grow more and more complex. Subsequently, modelling becomes an increasingly difficult task. Especially in the EPC case, users are overwhelmed by a highly unstructured language specification. The procedure model as the main IT artefact of this research in progress is designed to address these challenges by providing a systematic procedure for the creation of EPC models. Henceforth, the artefact can be applied to guide users through the modelling process. A crucial prerequisite for concrete modelling support in practice is the integration of an instruction-based modelling procedure in common process modelling tools, in which the modeller may be guided step-by-step through the EPC modelling process. In the context of enterprises, the procedure model opens up new possibilities as well: Being integrated into a structured training program, the developed model may facilitate the introduction of the EPC language or enhance the employees’ modelling capabilities in regular workshops. Additionally, a systematic procedure for process modelling can be customised to specifically adjust to the organisation’s needs. Exemplarily, an organisation can propose own, organisation-specific modelling rules that are integrated into the modelling process. Regarding research, the procedure model serves as a basis for further work in terms of user support and modelling guidance. By applying and subsequently refining the instruction model in practice, detailed insight with respect to the thought process of EPC users can be gained, which facilitates the understanding of how process modelling is actually conducted. Additional research potential can be seen in the outlined workflow support for modelling tools. Additional work needs to investigate requirements and design principles for process modelling workflow systems.

For the design of the procedure model, some assumptions had to be made limiting its applicability: First, the model is tailored towards rather “simple” EPC models. While the user is able to model splits and iterations following the suggested steps, there is a certain level of complexity that has been left out on purpose. However, any modeller should be able to model most of processes occurring in practice. Up to now, only a small scale evaluation has been conducted. A sophisticated evaluation needs to be carried out that on the one hand focuses on the complexity reduction provided by the designed procedure model, while on the other hand relate this reduction on complexity to model quality metrics, especially in terms of semantics and pragmatics, in order to investigate potential links between modelling support and increased model quality. At the same time, an eye has to be kept on the complexity of the procedure model itself. Although the model follows the objective of only supporting the most common model occurrences while working under several assumptions and restricting the EPC language, the instructions given in the model have been perceived as rather complex during the pre-test. Although initial results indicate that the model has been valued as helpful and beneficial, extending the model towards additional elements, non-trivial iterations and further quality metrics may increase its complexity to a point where applying the model is too time-consuming for usage in practice. Furthermore, the procedure model is built under the assumption that a textual description of a real-world already exists. However, in practice this may not be always the case, at least not in a structured form that can be directly translated into an EPC model using the proposed procedure.

It should be noted, however, that the presented research in this paper is still considered research in progress. Adhering to the methodology presented in section 3, the procedure model is the outcome of the design and development step. Further research on this matter will have to put special emphasis on an in-depth evaluation and demonstration in order to conclude a first iteration of the DSRM. As a preparation for this evaluation, a small-scale evaluation scenario has already been carried out with mixed group of five participants (undergraduate students and graduate students). Each participant was able to follow the instructions and to create an EPC model representing a given textual description. Initial results indicate the procedure model's viability to facilitate the modelling process. This preparatory results need to be verified within a dedicated evaluation including a larger target group. Ultimately, this evaluation may shed light on the feasibility of reducing modelling complexity via a procedure model.

5 Summary of Results and Conclusion

Given the complexity of process modelling we have discussed in our introduction and our literature review on modelling guidance and frameworks stated in the methodology section, we argue that there is a crucial need for systematic modelling support concerning the EPC language in research and practice. With the five steps of our developed artefact in Section 4, we realised our research goal: a procedural model supporting EPC modelling. Being addressed directly to modellers, our model supports the transformation of textual process descriptions into EPC models. By considering well-known quality frameworks like GoM, SEQUAL or 7PMG and especially by integrating consolidated work in terms of EPC modelling, particularly the syntactic, semantic and pragmatic rules provided by Fellmann et al. (2013), the procedure model is anchored in literature and builds upon the existing knowledge base. Related to our research question, an initial pre-test signals that the presented procedure model has the potential to facilitate the EPC modelling process and thus tackles the increasing complexity which modellers have to face. A further validation to link the reduction of modelling process complexity to increased model quality is a superordinate goal of our research project and will be covered in ensuing work.

Acknowledgments

The research presented in this paper is part of the SPEAK project and is funded by the Federal Ministry for Economic Affairs and Energy (BMWi) under grant number 01FS14030.

References

Becker, J., Niehaves, B. and Thome, I. (2010), "How Many Methods Do We Need?-A Multiple Case Study Exploration into the Use of Business Process Modeling Methods in Industry.", *AMCIS 2010 Proceedings*, Citeseer, p. Paper 534.

Becker, J., Rosemann, M. and Von Uthmann, C. (2000), "Guidelines of business process modeling", *Business Process Management*, Springer, pp. 30–49.

Chan, N.N., Gaaloul, W. and Tata, S. (2012), "Assisting business process design by activity neighborhood context matching", *International Conference on Service-Oriented Computing*, Springer, pp. 541–549.

Claes, J., Vanderfeesten, I., Gailly, F., Grefen, P. and Poels, G. (2015), "The Structured Process Modeling Theory (SPMT) a cognitive view on why and how modelers benefit from structuring the process of process modeling", *Information Systems Frontiers*, Vol. 17 No. 6, pp. 1401–1425.

Claes, J., Vanderfeesten, I., Reijers, H.A., Pinggera, J., Weidlich, M., Zugal, S., Fahland, D., et al. (2012), "Tying Process Model Quality to the Modeling Process: The Impact of Structuring, Movement, and Speed", in Barros, A., Gal, A. and Kindler, E. (Eds.), *Business Process Management*, Vol. 7481, Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 33–48.

Fellmann, M., Bittmann, S., Karhof, A., Stolze, C. and Thomas, O. (2013), "Do We Need a Standard for EPC Modelling? The State of Syntactic, Semantic and Pragmatic Quality", *5th International Workshop on Enterprise Modelling and Information Systems Architectures (EMISA)*, St. Gallen, Switzerland, pp. 103–116.

Hevner, A.R., March, S.T., Park, J. and Ram, Sudha. (2004), "Design Science in Information Systems Research", *MIS Quarterly*, Vol. 28 No. 1, pp. 75–105.

Hoffmann, W., Kirsch, J. and Scheer, A.-W. (1993), "Modellierung mit Ereignisgesteuerten Prozeßketten", Universität des Saarlandes.

Houy, C., Fettke, P. and Loos, P. (2009), "Stilisierte Fakten der Ereignisgesteuerten Prozesskette – Anwendung einer Methode zur Theoriebildung in der Wirtschaftsinformatik", in Nüttgens, M., Rump, F.J. and Mendling, J. (Eds.), *EPK 2009. 8. Workshop Der Gesellschaft Für Informatik e.V. (GI) Und Treffen Ihres Arbeitkreises "Geschäftsprozessmanagement Mit Ereignisgesteuerten Prozessketten (WI-EPK). Gesellschaft Für Informatik*, Bonn, Germany, pp. 22–41.

Indulska, M., Recker, J., Rosemann, M. and Green, P. (2009), "Business Process Modeling: Current Issues and Future Challenges", *Advanced Information Systems Engineering*, pp. 501–514.

Jannaber, S., Karhof, A., Riehle, D.M., Thomas, O., Delfmann, P. and Becker, J. (2016), "Invigorating Event-driven Process Chains - Towards an integrated meta model for EPC standardization", *Modellierung 2016 Workshopband, 2.–4. März 2016, Karlsruhe*, Köllen Druck+Verlag, Bonn, Germany, pp. 13–22.

Karhof, A., Jannaber, S., Riehle, D.M., Thomas, O., Delfmann, P. and Becker, J. (2016), "On the de-facto Standard of Event-driven Process Chains: Reviewing EPC Implementations in Process Modelling Tools", *Modellierung 2016, 2.-4. März 2016, Karlsruhe*, Köllen Druck+Verlag, Bonn, Germany, pp. 77–92.

Keller, G., Nüttgens, M. and Scheer, A.-W. (1992), "Semantische Prozeßmodellierung auf der Grundlage 'Ereignisgesteuerter Prozeßketten (EPK)'", Universität des Saarlandes.

Koschmider, A., Hornung, T. and Oberweis, A. (2011), "Recommendation-based editor for business process modeling", *Data & Knowledge Engineering*, Vol. 70 No. 6, pp. 483–503.

Krogstie, J., Sindre, G. and Jørgensen, H. (2006), "Process models representing knowledge for action: a revised quality framework", *European Journal of Information Systems*, Vol. 15 No. 1, pp. 91–102.

Li, Y., Cao, B., Xu, L., Yin, J., Deng, S., Yin, Y. and Wu, Z. (2014), "An Efficient Recommendation Method for Improving Business Process Modeling", *IEEE Transactions on Industrial Informatics*, Vol. 10 No. 1, pp. 502–513.

Martini, M., Pinggera, J., Neurauter, M., Sachse, P., Furtner, M.R. and Weber, B. (2016), "The impact of working memory and the 'process of process modelling' on model quality: Investigating experienced versus inexperienced modellers", *Scientific Reports*, Vol. 6 No. 1, available at:https://doi.org/10.1038/srep25561.

Mendling, J., Reijers, H.A. and van der Aalst, W.M. (2010), "Seven process modeling guidelines (7PMG)", *Information and Software Technology*, Vol. 52 No. 2, pp. 127–136.

OMG. (2011), *Business Process Model and Notation (BPMN) Version 2.0*, available at: http://www.omg.org/spec/BPMN/2.0/ (accessed 27 September 2015).

Peffers, K., Tuunanen, T., Rothenberger, M.A. and Chatterjee, S. (2007), "A Design Science Research Methodology for Information Systems Research", *Journal of Management Information Systems*, Vol. 24 No. 3, pp. 45–77.

Pinggera, J., Zugal, S., Weidlich, M., Fahland, D., Weber, B., Mendling, J. and Reijers, H.A. (2011), "Tracing the process of process modeling with modeling phase diagrams", *International Conference on Business Process Management*, Springer, pp. 370–382.

PwC. (2011), *Zukunftsthema Geschäftsprozessmanagement*, available at: http://www.pwc.de/de_DE/de/prozessoptimierung/assets/PwC-GPM-Studie.pdf.

Reijers, H.A., Mendling, J. and Recker, J. (2015), "Business Process Quality Management", edited by vom Brocke, J. and Rosemann, M.*Handbook on Business Process Management 1*, No. 2. edition, pp. 167–185.

Riehle, D.M., Jannaber, S., Karhof, A., Delfmann, P., Thomas, O. and Becker, J. (2016a), "Towards EPC standardization - A literature review on EPC exchange formats", presented at the Proceedings of the Multikonferenz Wirtschaftsinformatik (MKWI 2016), Ilmenau, Germany.

Riehle, D.M., Jannaber, S., Karhof, A., Thomas, O., Delfmann, P. and Becker, J. (2016b), "On the de-facto Standard of Event-driven Process Chains: How EPC is defined in Literature", *Modellierung 2016, 2.-4. März 2016, Karlsruhe*, Köllen Druck+Verlag, Bonn, Germany, pp. 61–76.

Smirnov, S., Weidlich, M., Mendling, J. and Weske, M. (2012), "Action patterns in business process model repositories", *Computers in Industry*, Vol. 63 No. 2, pp. 98–111.

Thomas, O. (2009), *Fuzzy Process Engineering*, 1st editio., Gabler Verlag, Wiesbaden, available at: http://books.google.com/books?id=UTyBHAFF2zwC&pgis=1.

Walterbusch, M., Grove, S., Breitschwerdt, R., Stolze, C., Teuteberg, F. and Thomas, O. (2013), "Case-based Selection of Business Process Modeling Tools : An Evaluation Criteria Framework", *AMCIS 2013 Proceedings*, No. October 2015, p. Enduser IS general presentation 21.

Weber, B., Zeitelhofer, S., Pinggera, J., Torres, V. and Reichert, M. (2014), "How advanced change patterns impact the process of process modeling", *Enterprise, Business-Process and Information Systems Modeling*, Springer, pp. 17–32.

Weske, M., van der Aalst, W.M.P. and Verbeek, H.M.W. (2004), "Advances in business process management", *Data & Knowledge Engineering*, Vol. 50 No. 1, pp. 1–8.