On the de-facto Standard of Event-driven Process Chains: Reviewing EPC Implementations in Process Modelling Tools

Arne Karhof[1], Sven Jannaber[1], Dennis M. Riehle[2], Oliver Thomas[1], Patrick Delfmann[2], Jörg Becker[3]

Abstract: Nowadays, most process modelling tools implement popular modelling languages such as the Business Process Model and Notation (BPMN) or the Event-driven Process Chain (EPC). However, in contrast to BPMN, no effort has yet been undertaken to standardize the EPC language, thus rendering EPCs as being merely a de-facto standard for business process modelling. Subsequently, this paper addresses this issue by laying ground for a successful EPC standardization. To achieve this task, several process modelling tools have been evaluated regarding their implementation of the EPC language with the objective to derive consensus about important language constructs. The evaluation reveals that there is a high degree of variety in the way tools implement EPCs. Especially syntax, semantic and pragmatic of the EPC language are not perceived homogenously and, in fact, commonly neglected. Hence, our research provides valuable implications for further EPC standardization by highlighting the state-of-the-art of the EPC from a software point of view.

Keywords: Event-driven Process Chain, EPC, Business Process Management, BPM, standardization, modelling tools, tool evaluation

1 The need for an EPC standard - How reviewing BPM tools might help

The modelling of business processes is an integral part of Business Process Management (BPM) [BNT10]. For any successful modelling endeavor, the emphasis has to be put on the choice of modelling software and business process modelling languages (BPML) [WAV04]. Studies reveal that the market volume of BPM software is continuing to grow, reaching 2.7 billion $ in 2015 [Ga15]. In 2013, over 52 vendors for BPM software compete in the broader BPM market [RM13]. Additionally, there exist several niche products of local software companies [DKK14], resulting in a large variety of potential tools to choose from. Beside proprietary notations, most tools support standardized languages such as the

[1] University of Osnabrück, Institute for Information Management and Information Systems, Katharinenstraße 3, 49069, Osnabrück, arne.karhof@uni-osnabrueck.de, sven.jannaber@uni-osnabrueck.de, oliver.thomas@uni-osnabrueck.de
[2] University of Koblenz-Landau, Institute for Information Systems Research, Universitätsstraße 1, 56016, Koblenz, riehle@uni-koblenz.de, delfmann@uni-koblenz.de
[3] University of Münster, European Research Center for Information Systems, Leonardo Campus 3, 48149, Münster, becker@ercis.uni-muenster.de

Business Process Model and Notation (BPMN) or the Unified Modeling Language (UML) for process design [DKK14]. Although the event-driven process chain (EPC) has been one of the most dominant languages for business process modelling in research and practice over the last decades [Fe09, HFL09, KS08], no systematic standardization efforts have taken place. Hence, the EPC is still considered merely a de-facto standard for business process modelling [Fe13, Wa13], though it consists of a variety of different variants [e.g. Ri16a]. The absence of an international accepted standard yields significant drawbacks for the EPC language, since the focus shifts towards standardized languages such as BPMN [DKK14]. This is primarily due to difficulties in terms of interoperability, further development and overall acceptance of modelling languages that are not based on an agreed-upon ground and thus may be implemented differently across modelling tools [Fe13].

In literature, many studies have addressed the amount of BPM software by providing an overview over the (German) BPM software market [BFV07, BS01], evaluating process modelling tools [NS02] or comparing implemented BPML [DGS10]. However, a specific focus on different EPC implementations in common BPM software has not yet been conducted. Furthermore, despite single attempts to provide specifications for parts of the EPC [e.g. MA07, NR02, Ro08], even literature has failed to reach agreement on fundamental EPC constructs. As a consequence of the combination of fragmented EPC research and arbitrary EPC implementations in software there exists no common ground in both theory and practice regarding the EPC language and its integral constructs. This, of course, hampers a systematic standard-making process guided by a standard development organization (SDO), since standardization is fundamentally based on agreement and consensus of a domain community [DG90, FKL03].

In order to prepare for a successful EPC standardization, this paper faces the stated challenges by bringing EPC implementations into perspective. Several BPM tools from both international as well as local vendors are examined in terms of degree and shape of EPC support. Hence, the paper specifically aims at providing an overview of EPC constructs as implemented in common business process modelling tools. Furthermore, the synthesis of the findings is proposed as a basis for further standardization efforts. The evaluation of various EPC implementations yields valuable insight into EPC language constructs that software companies have considered to be relevant. Therefore, the tool evaluation is able to highlight differences as well as congruities of EPC language constructs from a software point of view, which result in a proposition of constructs for standardization purposes.

The structure of the paper is organized as follows. Section 2 covers theoretical background on BPM with special focus on EPC modelling. The research methodology as applied in this paper is presented in Section 3. Subsequently, Section 4 provides an overview of identified EPC implementations. In Section 5, implications for EPC standard-making are carried out based on these findings. The paper concludes with a summary of the gained insights and an outlook on further work.

2 Process Modelling with Event-Driven Process Chains

In all kind of organizations, sets of activities are performed to achieve a superordinate business goal. Such activities, together with the organizational and technical environment they are performed in, are called business processes [We12, p.5]. BPM is about "concepts, methods, and techniques to support the design, administration, configuration, enactment, and analysis of business processes" [We12, p.5]. A key part of BPM is the creation of conceptual models to represent business processes, so called business process models.

Process models may be created using different BPML, which, for instance, can be UML, Petri nets, EPC or BPMN (see e.g. [DAH05]). One dominant language over the last decades is the EPC, which has initially been introduced at the University of Saarbrücken by [KNS92] in 1992. EPC models consist of alternating events and activities, which represent the process flow. Connectors can be used to split or merge the process flow as needed. For this reason, [KNS92] defines the AND connector (all subsequent process flows are performed), the XOR connector (exactly one process flow is performed) and the OR connector (one or more process flows are performed). Later, organizational units, information systems and information objects were added to EPC models to enrich functions with further details [HKS93].

Over the years, the EPC language has been further developed, extended and modified by a large number of different authors. For example, [Pr95] defines a sequence connector (where several process flows are performed in a sequence of any order), [Ro96] presents a decision table connector (ET) and the OR1 connector. New elements have been added to the EPC to model business risks [BO02, RM05, RW08] or to model fuzzy decision-making [TD06, THA02]. New relationships have been added to not only include the process-flow, but also the data-flow in EPC models [ZR96].

Other authors have added different concepts to EPC models, such as [HWS93], who link EPC models to a state-machine for real-time execution, or [TF06] and [FKS09], who annotate EPC models with concepts from an ontology. For different use cases of the EPC, there are several different and incompatible file-based exchange formats [Ri16b]. The idea of EPCs has also been transferred to different application scenarios, for example object-oriented business process modelling [NZ98, SNZ97], modelling of state-based workflow patterns [MNN05] or service-based process modelling [HW07]. Lastly, EPC has also been applied to configurative reference modelling [RA07].

In order to support design and administration of business processes, there are several commercial and a few non-commercial solutions for managing process models. Several studies provide an overview of the market for BPM tools [BH14, DKK14, Ga15, RM13]. As the number of BPM tools on the market is large, different variations and adoptions of the EPC are manifold. As there is no established standard for EPC, it is obvious that the implementation of EPC differs between BPM tools.

3 A methodology for identifying and evaluating BPM tools

In order to provide a comprehensive and holistic overview of the BPM software market, a structured procedure model is applied. For this purpose, we draw upon methodologies coming from marketing science and transform these methods to fit our needs. In [CM99], a systematic procedure model is introduced to support managers in the assessment of their market for potential competitors. A similar approach is presented by [BP02], who use a two-step framework for competitor identification and analysis. In essence, both models share the way in which an unstructured and crowded environment is grasped, and objects of interest are identified and evaluated. We apply this general process to the situation at hand. In this case, the environment is represented by the BPM software market. The detailed procedure model followed throughout this paper is depicted in Figure 1.

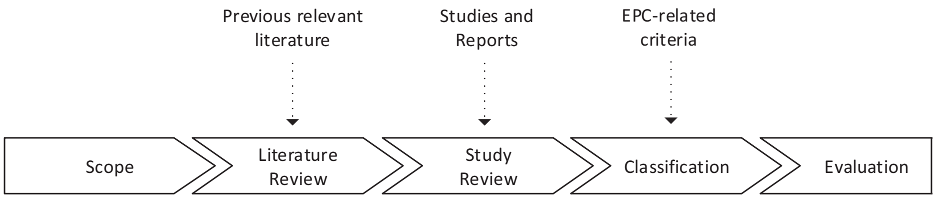

Fig. 1: Applied methodology

Using a five-step process, BPM tools are identified and analyzed. Regarding the scope, we limited our search to international software vendors who explicitly offer BPM software. Hence, workflow modelling suites have not been considered. Additionally, software tools had to be available for testing purposes. Therefore, we excluded vendors that did not provide any form of test or demo access to their software. Similarly, tools that could not have been installed due to technical circumstances have been omitted. Lastly, the final number of BPM tools has been shortened according to their support of the EPC language. Following the setting of the scope, relevant literature has been reviewed. This includes, for example, scientific contributions regarding EPC language constructs, but also related research-driven evaluations of BPM tools. Next, insight into the BPM market has been gained by taking various studies and market reports coming from institutions such as Gartner [SH10], the Fraunhofer Institute [DKK14], Forrester Research [RM13] or Ovum [BH14] into account. The final list has been analyzed using predefined criteria, which directly facilitate the identification of similarities and differences of EPC implementations in BPM tools. The criteria applied in this paper are presented in Table 1. For all EPC implementations, the state of EPC syntax, semantic and pragmatic is investigated. In doing so, we refer to respective sets of rules carried out in [Fe13]. Exemplarily, “1” as a characteristic of the syntax criterion indicates that a particular EPC implementation adheres to syntax rule “1” as presented in [Fe13]. Reviewing the state of syntactical, semantical and pragmatical rules covered by EPC implementations facilitates the understanding of what aspects of the EPC language are considered important in practice, hence supporting the determination of a standardized EPC specification that meets practical demands.

Criterion	Example	Description
Syntax	1,2,5	Refers to the set of syntactical rules as presented in [Fe13]
Semantic	3,4	Refers to the set of semantic rules as presented in [Fe13]
Pragmatic	1,6	Refers to the set of pragmatic rules as presented in [Fe13]
Elements	●	Does the extension introduce new elements beyond [Ro96]?
Connectors	●	Does the extension introduce new connector types beyond [Ro96]?
Checking	M	(M)anual or (D)esign time syntax check
Exchange format	XML	Type of EPC exchange format provided
Guidelines	Tooltips	Guidelines that support the modeler in the creation of sound EPC models

Tab. 1: EPC implementation evaluation criteria

Besides syntax, semantic and pragmatic, we expand the evaluation to modelling guidance and the extent of language constructs covered as well as their graphical representation. For this purpose, we use EPC elements and connector types summarized in [Ro96] as a baseline for further investigation. In particular, it is of interest whether an EPC implementation adheres to these elements resp. connector types. Based on potential similarities or deviations in terms of elements introduced or their graphical layout, a consensus of these elements can be reached from a software perspective. The Checking criterion is used to assess if the tool implements a manual or design-time based checking of the syntax, semantic and pragmatic of an EPC model, which provides insight regarding the importance of model soundness as perceived from BPM software vendors. Furthermore, having a look at the implemented exchange format is used to formulate a consensus regarding a standard exchange format for EPC models. Lastly, it is subject of analysis whether and how EPC implementations support the modeler in creating sound and meaningful EPC models.

Finally, the evaluation of tools using the aforementioned criteria is used to investigate the state of the EPC modelling language from a software point of view, hence to a certain degree reflecting needs that come from practice. The evaluation step not only provides an overview of a tool's capability to handle EPC modelling, but also gives additional input whether to include EPC language constructs in a potential EPC standard that have proven to be applicable and beneficial in practice.

4 An overview of EPC implementations in BPM tools

After setting the scope of our analysis, considering previous relevant literature and integrating similar studies about BPM software (cf. Fig. 1), we initially identified a set of 78 contemplable solutions. At first, no software has been excluded from our investigation. The list included leaders like Pegasystems or Appian, challengers, e.g. Fujitsu, niche players, exemplary Newgen or visionaries such as Intalio or BizAgi [SH10]. While classifying the first set according to our specified criteria in Section 3, we were able to remove 64 software solutions in total that did not fit our requirements. Thus, only 14 software providers have been identified that support EPCs and offer a free trial. We analyzed these solutions according to the predefined criteria. The results are presented in Table 1. Due to lack of space, we relinquish and outline the presentation of some categories. Overall, we relinquish the category miscellaneous, where we recorded additional elements that are not directly related with business process modelling or information about the underlying software like Eclipse or Microsoft Visio. Also, we spare the detailed enumeration of all elements the solutions provide, as we pit the extent against the native EPC components.

First, we recorded the supported business process languages to solely include EPC-supporting BPM solutions. Since we specifically examine EPC functionalities, all of the listed software can be considered as EPC compatible. During the examination, we checked if not only the standard EPC [KNS92] is supported, but also if any EPC extensions can be modeled. Except for *EPC Tools*, every solution supports the extended EPC [HKS93]. The *Bflow** platform additionally provides modelling with object-oriented EPCs [SNZ97]. Multiple tools also offer additional elements, but did not implement a specific EPC extension known from literature [e.g. SDL05]. We fastidious listed every additional element, adjusted the degree of abstraction and used circles as the form of representation. Thereby, a filled circle in the column "Elements" declares that the tool not only covers the extended EPC elements but also additional, partly tool-specific, elements. The software *EPC Tools* only supports the plain EPC and therefore is the only software figured as a bar. Unfilled circles represent just eEPC modelling elements support. Accordingly, unfilled circles in the column "Connectors" signify the AND, OR, XOR operators and the control/information flow. The only software solution that offers more than these basic constructs is the *Bflow** platform that additionally supports relation flows between EPC elements.

Product	Elements	Connectors	Syntax	Semantic	Pragmatic	Checking	Exchange formats	Guidelines
Cockpit Designer	●	○	2 7	—	2	D	—	Recommender
Aris Express	●	○	2 3	—	—	D	—	Ref. Models
Bflow*	●	●	1 2 4 6 7	●	1 2	D	XMI	Tooltips
BIC Platform	●	○	3	—	—	—	XML	Recommender
Cubetto	●	○	—	—	—	—	XML, CSV	—
Edraw Max	●	○	—	—	—	—	XML	—
EPC Tools	—	○	1 6	—	—	M	EPML	—
iGrafx	●	○	—	—	—	M	—	—
Process Modeler for Visio	○	○	—	—	—	—	BPEL, XPDL	—
SemTalk	●	○	1 2 3 5 6 7	—	2 6	D	XML	—
Signavio Process Editor	●	○	1 2 3 5 6 7	—	2 10	M	XML, XPDL	—
Symbio Modeling Client	●	○	2 6	—	1 2	D	—	—
ViFlow	●	○	—	—	—	—	—	—
Visio	○	○	—	—	—	—	XML	—

Tab. 2: Evaluation of EPC supporting BPM Software

The examination of quality criteria revealed an outlier for the second syntactic criterion, which requires a continual transition between events and functions. Many tools do not strictly demand the adherence to this rule, since they permit the omission of (trivial) events. In case this broad interpretation of the second rule has been implemented, the corresponding rule "2" in Table 1 has been underlined. Another striking point is the filled circle in the row of the *Bflow** tool. Because no tool supported any semantically validation, the *Bflow** tool stood out as being the only software that at least demands that the process model entities have to be labeled. Although this is no criterion stated in [Fe13], we decided to document it in the table nevertheless. The last two categories illustrate if there is any exchange format for EPC models supported by the solution and if there is any kind of guidance for the user while modelling in EPC. As the table constitutes, multiple solutions provide the import and export of specific exchange formats. However, the specific type of exchange format heavily differs. Although just three entries could be done in the last category, a heterogeneous distribution can be observed. The three addressed solutions pursue different approaches as there are either tooltips that pop up during design-time, reference models that the user can use as templates or a recommendation system that supports the user in deciding what element to model next.

5 Implications for EPC standardization

As our findings indicatethere is hardly a consensus between software vendors in the field of BPM regarding essential EPC constructs. For our next step, it has to be discussed to what degree the insights gained from reviewing BPM tools influence the proposed standardization process. In order to solve this debate, stakeholders of a standard need to be considered. Possible stakeholders are consultants, academia, research institutes, and governmental agencies [LK06]. From a high-level point of view, these groups can either be considered as (academic) researchers or practical applicants, who do not necessarily have to be disjoint groups. However, for the sake of argument we assume that these groups have different roles regarding the impact factor on standardization. On the one hand, research interest groups are more likely to take part in the (continued) development process of a standard, while on the other hand practical applicants fulfill their role by implementing and actual using this standard. Hence, we argue that the developer point of view of a business process modeling language best reflects the acceptance or denial of specific language constructs in practice. Therefore, it is possible to derive insights about whichcomponents to consider in an EPC standardization process and which not. Furthermore, as Table 1 already demonstates, our examination highlights the negative outcome a missing standard can cause. The practical implementation of EPC modelling is a strikingly heterogonous area, as the software vendors were never able to adhere to an agreed-upon basis for the EPC language.

In favor of a better overview of this diversified field of application, we analyzed our results as shown in Figure 1. Especially the overlaps, where the software vendors achieved agreement among themselves, are of particular interest, as derivations for an EPC standard can directly be deduced. Unfortunately, as depicted in Figure 1, the commonalities are highly underrepresented. Regarding the elements category, only 21.4% still support solely the EPC and eEPC basis elements. However, it is worth mentioning that 78.6% does not imply that all of the vendors actually use *one* set of EPC elements but *their own* notation set that exceeds the eEPC (illustrated by “eEPC+”). Therefore, the majority of EPC tools extends the (e)EPC with additional, self-developed elements. The second category is the only one where a major consensus could be identified. Only 7.1% of all vendors use additional, own developed connector types. Consequently, from a practical point of view, it seems like the standard EPC connector types already cover most of the users’ requirements. The following three categories illustrate the distribution of quality checking in regard to syntax, semantic and pragmatic. While syntax is the only category which is supported by more than half of the tools, the nearly non-existing semantic validation stands out. However, the pragmatic quality seem to be well-covered by BPM tools. Furthermore, as the binary classification only represents the number of tools supporting quality checking, but not *the number of criteria* covered, we added an additional value for the average support. This value is calculated as follows:

$$\text{Ø} - \text{Support} = \frac{\sum_{i=1}^{N} \frac{T_i}{|T|}}{N}$$

The variables are as follows: N = quantity of quality criteria based on [Fe13], i = the specific quality criterion, and T_i = number of tools who support the specific quality criterion i. $|T|$ indicates the cardinality of the set T. This value is equal to the overall number of evaluated tools. In our case, this value always signifies 14. Exemplary, the value for average support of the pragmatic quality can be obtained by calculating:

$$\text{Ø} - Support_{Prag.} = \frac{\frac{2}{14}+\frac{6}{14}+\frac{0}{14}+\frac{0}{14}+\frac{0}{14}+\frac{1}{14}+\frac{0}{14}+\frac{0}{14}+\frac{0}{14}+\frac{1}{14}}{10}$$

$$\text{Ø} - Support_{Prag.} = \frac{0.714286}{10} = 0.07143 \triangleq 7.1\%$$

Considering these values of the three quality categories (illustrated in Figure 1 as hatched areas), it is evident that every category is insufficiently supported by BPM tools. Even the syntactical category, which demands criteria relatively easy to implement, is only associated with an average support of 22.3%. Furthermore, the average semantic support is nearly nonexistent (0.4%). The pragmatic support is weakly represented as well. Finally, the last three categories present the statistically analysis of checking support, provided exchange formats and user guidance. While most of the tools support some form of verification or validation while modelling, it has to be kept in mind that the corresponding Ø-support values show that this support is heavily constrained. Additionally, most of the tools provide some sort of exchange format. However, as Figure 1 depicts, besides using XML as an underlying structure, software vendors have not been able to reach an agreement on an exchange standard. The same problem can be observed with user supporting guidelines. While only three software solutions (21.4%) provide any kind of user guidance, they all confine to different approaches.

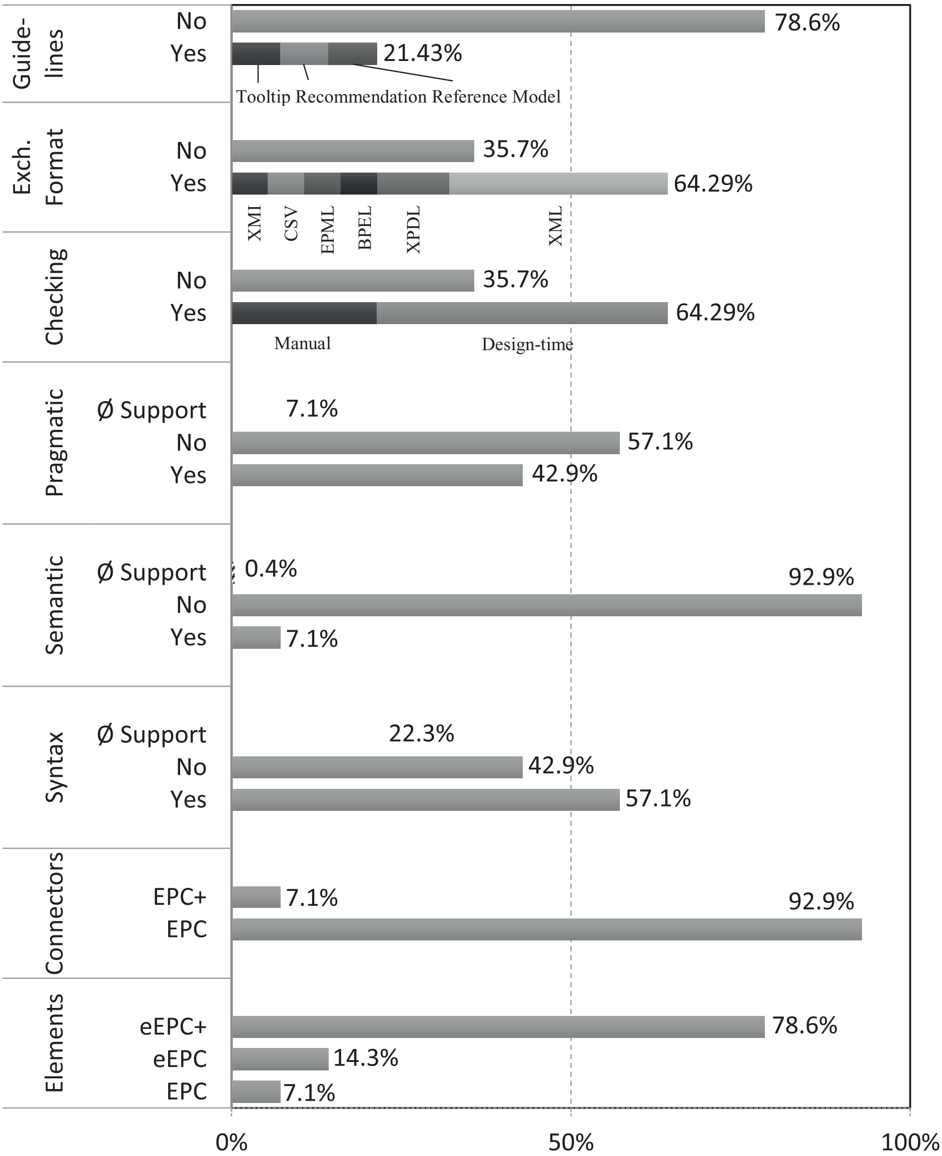

Fig. 2: Evaluation of BPM-Tools

Following the elicitation and evaluation of BPM-Tools, it is of interest to draw - in the best case obvious - implications towards EPC standardization. Our initial intension to interpret commonalities is hampered by the fact that, besides the category Connectors, there are no distinct overlaps among the software solutions. Despite this circumstance, we can ascertain the following points regarding to our evaluation categories:

Elements: Based on the percentage distribution, we assume that the basic (e)EPC elements seem not to completely fit the needs of nowadays users. The majority of software vendors use additional elements, partly self-developed, partly from other BPM languages or proposals from academic research. Nevertheless, it could be observed that most of the tools did use a similar layout for their elements. This includes the hexagon for events, the rectangle for functions (mostly with rounded edges) and circular connectors. However, the labelling of connectors is again diversified in EPC implementation (XOR vs. X). For future work, it will become necessary to value those additional elements in favor of deciding whether to integrate them in an EPC standard.

Connectors: As nearly all solutions use the basis set of EPC connectors, the general applicability can be underlined. Self-evidently, the outlier has to be examined in terms of determining its usability.

Syntax, Semantic, Pragmatic: In this area, there is a heavy mismatch among the tools. It can be concluded that the EPC would heavily benefit from an underlying and formal defined standard as a basis for EPC implementations. Unfortunately, as there are almost no commonalities, explicit rules for EPC standardization cannot be deduced. A single implication that can be made is that the difficulty of verifying EPC quality seems to vary in regard to the respective category. Based on the findings we assume that the semantic quality is the most challenging task for computer-based verification.

Checking: Since the evaluation presents roughly the same distribution between design-time and manual-triggered checking of syntactical, semantical or pragmatically aspects, an evident recommendation from practice cannot be made. Additionally, no coherence between the type of checking and the nature and extent of covered rules could be detected. Generally, a design-time approach seems to be more applicable from a practical perspective, as it prevents users from working with erroneous models. However, it has to be kept in mind that not every error detected by a checking mechanism necessarily has to be an actual error of the underlying process model. Hence, the checking mechanism must not be too restrictive for practical application. In general, the review of checking mechanisms implies that, considering a potential EPC standard, emphasis has to be put on the degree of complexity and restrictions in order to meet practical demands.

Exchange Formats: The area of exchange formats represents the most diversification. Out of nine software vendors, who offer any type of exchange format, six different approaches are pursued. The most common format is XML. Since many other listed formats are also based on XML, namely XMI, EPML, BPEL and XPDL, it can be seen as the major choice for data communication between EPC tools. This heavily implies that a XML-based exchange format might be feasible for an EPC standard.

Guidelines: The provided user guidance is rather underrepresented. Any concrete implications for an EPC standard cannot be deduced. In future, the usability of proposed guidance types has to be valued. Furthermore, it needs to be clarified if the presence of user guidance is actually necessary if there is already a checking mechanism.

6 Conclusion

In order to prepare for EPC standardization, the similarities and differences between EPC implementations in common BPM software has to be revealed. Subsequently, we identified BPM tools that implement the EPC modelling language and evaluated them against our predefined criteria. Altogether, we analyzed 14 tools measured against 8 categories and 34 different quality criteria.. Our findings show that the negative effects caused by the abstinence of a universally accepted EPC standard are prevalent. Regarding the predefined criteria, there has been no consensus in terms of EPC language constructs. The only commonalities that could be identified regard the basic layout of EPC elements and EPC connectors. Most notably, the diversification of quality criteria regarding the syntax, semantic and pragmatic is apparent. Despite 64% of identified tools provide a basic checking mechanism, most of them only consider a small subset of rules. The majority of tools leave the creation of valid process models solely to the user. This may imply a lack of importance for semantic, pragmatic and syntactic issues in practice.

By evaluating, we could further deduce implication for standardization purposes and gain valuable insights about the practical implementation of the EPC. Explicit conclusions could be made regarding the layout, the set of connectors, and the exchange format. In terms of the layout, all EPC implementations share the same graphical representation. Accordingly, also the set of connectors is identical among the tools. Therefore, we conclude that there is an agreement regarding the layout of EPC elements that has to be considered for EPC standard making. Furthermore, we revealed that exchange formats for EPC modelling are dominated by XML. Hence, we suggest a XML-based EPC exchange format to be included in a future EPC standard.

Based on our research, we are able to state and underline the urgent need for an EPC standard. In this paper, we evaluated EPC modelling from a practical point of view and uncovered the consequences of a non-standardized process modelling language. In conclusion, our research contributes to the body of knowledge in two ways. First, we shed light on the BPM software market and provide an overview of core players in EPC modelling which supports both academic and practical disciplines, as we build groundwork for further research and application in the field of EPC modelling. Second, we investigated EPC implementations in order to achieve a consensus regarding essential EPC language constructs from a software point of view.

Acknowledgement: 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

[BFV07] Bartels, G., Frank, P., Völz, M.: Vergleich von BPMN-Modellierungswerkzeugen. Stuttgart 2007.

[BH14] Barnett, A.G., Holt, M.: Ovum Decision Matrix : Selecting a Business Process Management Solution , 2014. 2014.

[BNT10] Becker, J., Niehaves, B., Thome, I.: How Many Methods Do We Need ? – A Multiple Case Study Exploration into the Use of Business Process Modeling Methods in Industry. In: AMCIS 2010 Proceedings, 2010.

[BO02] Brabänder, E., Ochs, H.: Analyse und Gestaltung prozessorientierter Risikomanagementsysteme mit Ereignisgesteuerten Prozessketten. In: EPK 2002 - Geschäftsprozessmanagement mit Ereignisgesteuerten Prozessketten, Proceedings des GI-Workshops und Arbeitskreistreffens, pp. 1–5. 2002.

[BP02] Bergen, M., Peteraf, M. a.: Competitor Identification and Competitor Analysis: A Broad-Based Managerial Approach. Managerial and Decision Economics 23, pp. 157–169. 2002.

[BS01] Bullinger, H.-J., Schreiner, P.: Business Process Management Tools: Eine evaluierende Marktstudie über aktuelle Werkzeuge. Stuttgart 2001.

[CM99] Clark, B.H., Montgomery, D.B.: Managerial Identification of Competitors. Journal of Marketing 63, pp. 67–83. 1999.

[DAH05] Dumas, M., Aalst, W.M.P. van der, Hofstede, A.H.M. ter.: Process-aware Information Systems - Bridging People and Software through Process Technology. John Wiley & Sons, Hoboken, New Jersey 2005.

[DG90] David, P.A., Greenstein, S.: The Economics of Compatibility of Standards: An Introduction to Recent Research. Economics of Innovation and New Technology 1, pp. 3–41. 1990.

[DGS10] Drawehn, J., Gayer, S., Schneider, P.: Business Process Modeling 2010: Modellierung von ausführbaren Geschäftsprozessen mit der Business Process Modeling Notation. Fraunhofer, Stuttgart, Germany 2010.

[DKK14] Drawehn, J., Kochanowski, M., Kötter, F.: Business Process Management Tools 2014. Fraunhofer, Stuttgart, Germany 2014.

[Fe09] Fettke, P.: Ansätze der Informationsmodellierung und ihre betriebswirtschaftliche Bedeutung: Eine Untersuchung der Modellierungspraxis in Deutschland. Schmalenbachs Zeitschrift für betriebswirtschaftliche Forschung (zfbf) 61, pp. 550–580. 2009.

[Fe13] Fellmann, M., Bittmann, S., Karhof, A., Stolze, C., Thomas, O.: Do We Need a Standard for EPC Modelling? The State of Syntactic, Semantic and Pragmatic Quality. In: 5th International Workshop on Enterprise Modelling and Information Systems Architectures (EMISA), pp. 103–116. St. Gallen, Switzerland 2013.

[FKL03] Fomin, V., Keil, T., Lyytinen, K.: Theorizing about standardization: integrating fragments of process theory in light of telecommunication standardization wars. Sprouts: Working Papers on Information Environments, Systems and Organizations 3, pp. 29–60. 2003.

[FKS09] Filipowska, A., Kaczmarek, M., Stein, S.: Semantically annotated EPC within semantic business process management. In: Lecture Notes in Business Information Processing, pp. 486–497. Milano, Italy 2009.

[Ga15] Gartner Inc.: Gartner Says Spending on Business Process Management Suites to Reach $2.7 Billion in 2015 as Organizations Digitalize Processes. 2015. Available at: http://www.gartner.com/newsroom/id/3064717, accessed 15. Oct 2015.

[HFL09] Houy, C., Fettke, P., Loos, P.: Stilisierte Fakten der Ereignisgesteuerten Prozesskette – Anwendung einer Methode zur Theoriebildung in der Wirtschaftsinformatik. In: EPK 2009. 8. Workshop der Gesellschaft für Informatik e.V. (GI) und Treffen ihres Arbeitkreises "Geschäftsprozessmanagement mit Ereignisgesteuerten Prozessketten (WI-EPK). Gesellschaft für Informatik, pp. 22–41. Bonn, Germany 2009.

[HKS93] Hoffmann, W., Kirsch, J., Scheer, A.-W.: Modellierung mit Ereignisgesteuerten Prozeßketten. In: Veröffentlichungen des Instituts für Wirtschaftsinformatik (IWi), No. 101, Universität des Saarlandes 1993.

[HW07] Huth, S., Wieland, T.: Geschäftsprozessmodellierung mittels Software-Services auf Basis der EPK. In: V. Nissen, M. Petsch, H. Schorcht (eds.) Service-orientierte Architekturen. pp. 61–76. Deutscher Universitäts-Verlag, Wiesbaden 2007.

[HWS93] Hoffmann, W., Wein, R., Scheer, A.-W.: Konzeption eines Steuerungsmodells für Informationssysteme – Basis für die Real-Time-Erweiterung der EPK (rEPK). In: Veröffentlichungen des Instituts für Wirtschaftsinformatik (IWi), No. 106, Universität des Saarlandes 1993.

[KNS92] Keller, G., Nüttgens, M., Scheer, A.-W.: Semantische Prozeßmodellierung auf der Grundlage “Ereignisgesteuerter Prozeßketten (EPK).” In: Veröffentlichungen des Instituts für Wirtschaftsinformatik (IWi), No. 89, Universität des Saarlandes 1992.

[KS08] Knuppertz, T., Schnägelberger, S.: Status Quo Prozessmanagement 2007/2008 – Ergebniszusammenfassung. Komptenzzentrum für Prozessmanagement 2008.

[LK06] Lyytinen, K., King, J.L.: Standard Making: A Critical Research Frontier for Information Systems Research. MIS Quarterly 30, pp. 405–411. 2006.

[MA07] Mendling, J., Aalst, W.M.P. van der.: Formalization and Verification of EPCs with OR-Joins Based on State and Context. Proceedings of the 19th International Conference on Advanced Information Systems Engineering (CAiSE’07) 4495, pp. 439–453. 2007.

[MNN05] Mendling, J., Neumann, G., Nüttgens, M.: Yet another event-driven process chain. In: Lecture Notes in Computer Science, pp. 428. 2005.

[NR02] Nüttgens, M., Rump, F.J.: Syntax und Semantik Ereignisgesteuerter Prozessketten (EPK). Prozessorientierte Methoden und Werkzeuge für die Entwicklung von Informationssystemen P-21, pp. 64–77. 2002.

[NS02] Nägele, R., Schreiner, P.: Bewertung von Werkzeugen für das Management von Geschäftsprozessen. Zeitschrift Führung und Organisation 71, pp. 201–210. 2002.

[NZ98] Nüttgens, M., Zimmermann, V.: Geschäftsprozeßmodellierung mit der objektorientierten Ereignisgesteuerten Prozeßkette (oEPK). In: M. Maicher, H.-J. Scheruhn (eds.) Informationsmodellierung - Referenzmodelle und Werkzeuge. pp. 23–35. Gabler, Wiesbaden, Germany 1998.

[Pr95] Priemer, J.: Entscheidungen über die Einsetzbarkeit von Software anhand formaler Modelle. Pro-Universitate-Verlag, Sinzheim 1995.

[RA07] Rosemann, M., Aalst, W.M.P. van der.: A configurable reference modelling language. Information Systems 32, pp. 1–23. 2007.

[Ri16a] Riehle, D.M., Jannaber, S., Karhof, A., Thomas, O., Delfmann, P., Becker, J.: On the de-facto Standard of Event-driven Process Chains: How EPC is defined in Literature. In: Proceedings of the Modellierung 2016, Karlsruhe, Germany 2016.

[Ri16b] Riehle, D.M., Jannaber, S., Karhof, A., Delfmann, P., Thomas, O., Becker, J.: Towards an EPC Standardization – A Literature Review on Exchange Formats for EPC Models. In: Proceedings of the Multikonferenz Wirtschaftsinformatik (MKWI 2016), Ilmenau, Germany 2016.

[RM05] Rosemann, M., Muehlen, M. zur.: Integrating Risks in Business Process Models. Australasian Conference on Information Systems (ACIS) , pp. 62–72. 2005.

[RM13] Richardson, C., Miers, D.: The Forrester Wave™: BPM Suites, Q1 2013. 2013.

[Ro08] Rosa, M. La, Dumas, M., Hofstede, A.H.M. ter, Mendling, J., Gottschalk, F.: Beyond Control-Flow: Extending Business Process Configuration to Roles and Objects. 5231, pp. 199–215. 2008.

[Ro96] Rosemann, M.: Komplexitätsmanagement in Prozeßmodellen. Gabler, Wiesbaden 1996.

[RW08] Rieke, T., Winkelmann, A.: Modellierung und Management von Risiken. Ein prozessorientierter Risikomanagement-Ansatz zur Identifikation und Behandlung von Risiken in Geschäftsprozessen. Wirtschaftsinformatik 50, pp. 346–356. 2008.

[SDL05] Sarshar, K., Dominitzki, P., Loos, P.: Einsatz von Ereignisgesteuerten Prozessketten zur Modellierung von Prozessen in der Krankenhausdomäne – Eine empirische Methodenevaluation. EPK 2005 -Geschäftsprozessmanagement mit Ereignisgesteuerten Prozessketten, pp. 97–116. 2005.

[SH10] Sinur, J., Hill, J.B.: Magic Quadrant for Business Process Management Suites. 2010.

[SNZ97] Scheer, A.-W., Nüttgens, M., Zimmermann, V.: Objektorientierte Ergenisgesteuerte Prozeßkette (oEPK) - Methode und Anwendung. In: Veröffentlichungen des Instituts für Wirtschaftsinformatik (IWi), No. 141, Universität des Saarlandes 1997.

[TD06] Thomas, O., Dollmann, T.: Attributierung und Regelintegration. Proceedings of the 5th GI Workshop on Event-Driven Process Chains (EPK 2006) , pp. 49–68. 2006.

[TF06] Thomas, O., Fellmann, M.: Semantische Ereignisgesteuerte Prozessketten. Data Warehousing , pp. 205–224. 2006.

[THA02] Thomas, O., Hüsselmann, C., Adam, O.: Fuzzy-Ereignisgesteuerte Prozessketten - Geschäftsprozessmodellierung unter Berücksichtigung unscharfer Daten. EPK 2002 - Geschäftsprozessmanagement mit Ereignisgesteuerten Prozessketten, Proceedings des GI-Workshops und Arbeitskreistreffens, pp. 7–16. 2002.

[Wa13] Walterbusch, M., Grove, S., Breitschwerdt, R., Stolze, C., Teuteberg, F., Thomas, O.: Case-based Selection of Business Process Modeling Tools : An Evaluation Criteria Framework. AMCIS 2013 Proceedings, pp. Enduser IS general presentation 21. 2013.

[WAV04] Weske, M., Aalst, W.M.P. van der, Verbeek, H.M.W.: Advances in business process management. Data & Knowledge Engineering 50, pp. 1–8. 2004.

[We12] Weske, M.: Business Process Management. Springer, Heidelberg 2012.

[ZR96] Zukunft, O., Rump, F.: From Business Process Modelling to Workflow Management - An Integrated Approach. In: B. Scholz-Reiter, E. Stickel (eds.) Business Process Modelling. pp. 3–22. Springer, Berlin 1996.