

# Automatically Annotating Business Process Models with Ontology Concepts at Design-Time

Dennis M. Riehle<sup>1</sup>, Sven Jannaber<sup>2</sup>, Patrick Delfmann<sup>3</sup>, Oliver Thomas<sup>2</sup>, Jörg Becker<sup>1</sup>

<sup>1</sup>European Research Center for Information Systems, University of Münster, Münster, Germany, <sup>2</sup>Institute for Information Management and Information Systems, University of Osnabrück, Osnabrück, Germany, <sup>3</sup>Institut for Information Systems Research, University of Koblenz-Landau, Koblenz, Germany  
{dennis.riehle|joerg.becker}@ercis.uni-muenster.de  
{sven.jannaber|oliver.thomas}@uni-osnabrueck.de  
delfmann@uni-koblenz.de

**Abstract.** In business process modelling, it is known that using a consistent labelling style and vocabulary improves process model quality. In this regard, several existing approaches aim at the linguistic support for labelling model elements. At the same time, domain-specific ontologies have been proposed and used to capture important process-related knowledge. However, these two areas are largely disconnected up to now. Although some research suggests annotating ontology concepts to process models, for instance, to interpret and reason about a process model, annotation has not yet gained traction in practice since it still has to be done in a highly manual effort. We thus provide an automated, language-independent methodology for using labelling assistance functionalities to identify and annotate relevant ontology concepts to process model elements using a four-step procedural model.

**Keywords:** process models, ontology, automatic annotation, analysis

## 1 Motivation

In information systems, conceptual models are frequently used within research and practice. They enable a graphical and structural representation of the main concepts of an application domain and their relations [1]. One type of conceptual models is business process models, which represent the set of activities performed within a business process together with their logical execution order and environment. Business process models support the analysis of business processes, which is a common task of business process management [2]. Over the last decades, business process management (BPM) has been of growing interest. Researchers have developed new approaches and algorithms to further improve BPM techniques and practitioners have implemented BPM techniques in their organisations [3]. Not surprisingly, organisations nowadays maintain large repositories of process models, consisting of hundreds if not thousands of models. A three-study at Suncorp-Metway Ltd (an Australian insurer) dealt with over 6000 process models, after the organisation had gone through several mergers and

acquisitions [4]. Such process models are usually created by modellers, that is, human beings. Therefore, the creation of process models is influenced by the individual perception of the modellers and, hence, different modellers may create models of different quality. Modelling expertise of modellers has been shown to influence the comprehensibility of models [5]. However, different models may still name the same things differently, leading to naming conflicts [6]. The more people there are involved in the modelling process, the more it is likely that resulting process models differ largely in respect to the used terminology [7]. Still modellers appreciate a consistent naming of model elements [8].

While there are different approaches to prevent naming conflicts (see section ‘Related Work’), these approaches usually only improve the labelling of process models (e.g., by defining a syntax for natural language), but two different modellers do still not necessarily refer to the same semantic concept. Consequently, the domain specific concepts need to be shared among all modellers, using an ontology to represent the body of domain knowledge [9]. An ontology is an engineering artefact, which includes not only a vocabulary to describe reality, but also includes a set of intended meanings for concepts and vocabulary included in the ontology [10].

To connect process models with domain ontologies, elements from the process models need a relation to concepts from the ontology. Given that the ontology contains organisational knowledge, this allows an advanced business process analysis. For example, organisational compliance regulations regarding process execution could be stored in the ontology and then, during analysis, be used to check business process models for compliance. Currently, the relation between an ontology concept and a process element needs to be established manually, which is a challenging and resource consuming task given the size of current process model repositories. Therefore, this paper presents a methodology to automatically create the relations between elements in a business process model and concepts from an ontology during design-time. Reducing the administrative overhead to link process models and ontologies may foster the use ontology-based process analysis in the future.

This paper is structured as follows: First, we shortly discuss related work. Then, we discuss the research gap that our methodology aims to close. After this, we introduce our methodology automatic annotations. Finally, we consider limitations of our work and provide a conclusion and outlook.

## 2 Related Work

Early research towards terminological standardisation proposes glossaries and structural rules to achieve unified naming. A rule for activities in business process might be, for example, “<Verb, Imperative> <Noun>”, which would make “process order” a valid label, but not “order is processed”. More advanced approaches like [11] suggest to use a domain ontology instead of the glossary. A partially automatic methodology for copying concept labels from ontologies into models modelled with Business Model and Notation (BPMN) has been introduced by [12]. However, their approach requires states that business objects can undergo and has been applied only to BPMN so far.

Another approach to match elements from conceptual models with concept from formal semantic schemata is provided by [13], but their approach does not provide the modeller with suggestions at design-time. Several studies (e.g., [14, 15]) use online dictionaries like WordNet to suggest labels for model elements.

Regarding terminological standardization, we can distinguish between approaches which provide terminological standardisation at design time and those, which check for terminological standardisation as part of a process analysis procedure. For process models which are not terminologically standardized, there are several approaches in the literature, which judge the quality of process element labels and give hints on possible naming violations (e.g., [16]) or resolve them (e.g., [17]). Compared thereto, approaches which support process modellers at design time are rather young. A technique which provides suggestions to the modeller has been suggested by [18]. Their approach uses a dictionary with a vocabulary that contains all necessary words to label process elements, the so-called domain thesaurus. By using structural rules they provide the modeller with terminologically standardised labels at design time. This approach has the advantage of supporting the creation of correct process models right from scratch, while still being applicable to existing process models as well.

A recent study presents an automatic annotation of process models with concepts from a taxonomy [19]. While this approach tracks a similar target, it is different in the manner that it computes distributional similarities between process model elements and taxonomy concepts to detect inconsistent terminology. This can lead to wrong results, especially if labels are not terminologically standardised beforehand. Additionally, this approach does not provide an automatic annotation at design-time.

Summing up, there are several techniques to achieve terminological standardisation, some at design time, some as part of process analysis. However, up to now, no approach is capable of automatically annotating ontology concepts to process models at design time. Therefore, annotation of ontologies requires a prohibitively high manual effort and annotation is rarely used in practice. Consequently, existing approaches utilizing domain ontologies during the process analysis and improvement cannot gain their full potential. To close this research gap, we thus propose a methodology to automatically annotate conceptual models with ontology concepts at design-time.

### **3 Automatic Annotation of (Process) Models with Ontologies**

Our methodology aims to achieve essentially two things: First, terminological standardisation shall be assured, for which we follow the methodology provided by [18]. Second, elements from process models shall be automatically related to concepts from an ontology to foster a common understanding of process elements and to enable further ontology-based process analyses and process modelling assistance in the future.

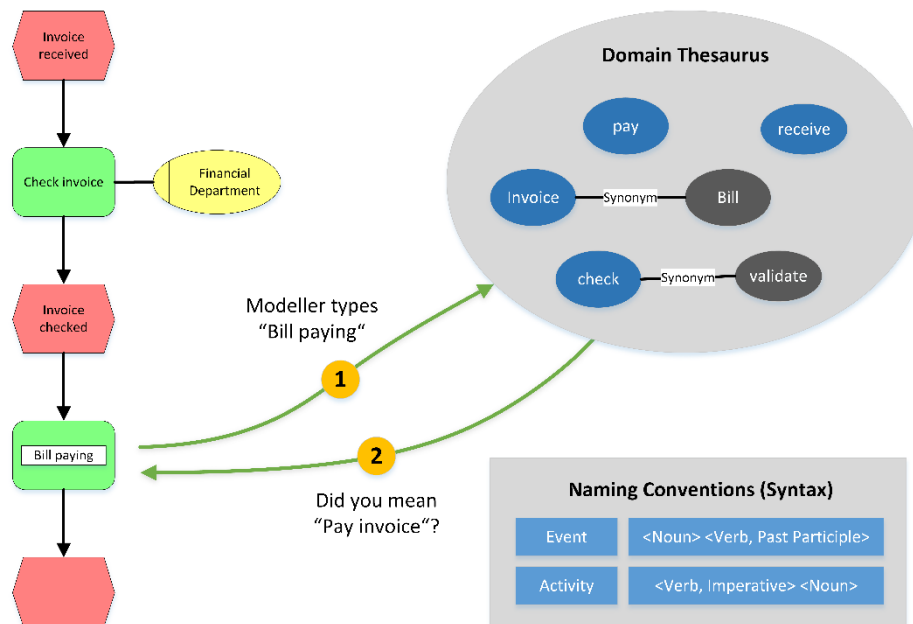
#### **3.1 Terminological Standardization**

A key requirement for automatic annotation are standardised and unambiguous identifiers for all elements within a business process model. This is achieved by ensuring

terminological standardisation of all elements' labels at design-time. For this, essentially two things are required (cf. Figure 1):

1. A domain thesaurus with valid words (ideally corresponding to the labels of the ontology's concepts to easily establish the annotation)
2. Syntactical naming conventions

The domain thesaurus contains a vocabulary of all words of a natural language considered valid in the respective application domain. Words in the domain thesaurus are declared as being a noun or a verb and need to be in their uninflected form, which is the singular form for nouns and the infinitive for verbs. Furthermore, the domain thesaurus includes relations between words, for example synonyms, homonyms or antonyms. An exemplary thesaurus could include the information of "bill" being a synonym for "invoice", "invoicing" being a word formation of "invoice" and "correct" being an antonym of "incorrect".



**Figure 1.** Terminological Standardization at Design-Time [18]

While lexical databases like WordNet can easily be adapted rather than building a domain thesaurus from scratch, the domain thesaurus needs to contain additional, domain-specific information. In case of synonyms, one of the synonyms has to be marked as dominant, in order to specify which of the synonyms is to be used preferably within the application domain. Dominant words are shown with a blue background in the domain thesaurus of Figure 1, while non-dominant words are shown with a grey background.

Next, syntactical naming conventions are required. These naming conventions follow the suggestions by [20]. Such naming conventions differ depending on the type of element which is to be named. In the area of process modelling, typical element types

are activities, events or organisational units. For example, activities could be named by the rule “<Verb, Imperative> <Noun>”, which would make “Write paper” a syntactically correct label, while “Writing paper” or “Paper is written” would be syntactically incorrect labels. Note that there can be more than one syntactical naming convention per element type. For instance, activities in process models may require more complex phrases than the one mentioned above.

Using the domain thesaurus and the syntactical naming conventions, an automatic suggestion of terminologically standardised labels can be supported within any modelling tool. In concordance with [18], this works as follows: First, the phrase entered by the modeller (cf. Step 1 in Figure 1) needs to be parsed into uninflected words. Single words have to be recognised and turned into their corresponding lexeme, which is the singular form for nouns or the infinitive for verbs. Software for this is already known in the literature (see [18] for an overview). These lexemes are then looked-up in the domain thesaurus in order to resolve synonyms to their dominant terms. In case a dominant term cannot be found because, for example, the term entered by the modeller is not known to the thesaurus, a further automatic search can be performed in general lexicons like wordnet, which may return synsets that contain a term known to the thesaurus. In the example shown in Figure 1, the modeller entered “Bill paying”, which would result in the lexemes “bill” and “pay”, of which the premier would resolve into the dominant term “invoice” after consulting the domain thesaurus.

With the standardised words and the syntactical naming conventions, the phrase can be reformulated based on the element type the original phrase was typed into. Again, in the example shown in Figure 1, the modeller typed a label for an activity, therefore the rule “<Verb, Imperative> <Noun>” is to be used. With the words “invoice” and “pay” this results in the phrase “pay invoice”. This phrase is suggested to the modeller as a correct and valid labelling (cf. Step 2 in Figure 1).

### 3.2 Domain Ontology

The second key requirement for automatic annotation of process models is the presence of a domain ontology, which represents all concepts of the organisation. Such an ontology also includes elements and mutual relations (so far similar to the domain thesaurus). In addition, ontology elements are semantic concepts rather than simple words, and relations can be semantically typed (e.g., “requires”, “is a”, “targets at”). These semantic relationships are used to express interrelations between concepts that do not necessarily occur in process models, organisational charts, data models or the like. Therefore, they can be used to depict advanced domain semantics. Furthermore, by using relations defining inheritance, it is possible to store abstract instances (often called classes, stored in the so-called TBox of the ontology) and concrete instances (stored in the so-called ABox of the ontology) in one and the same ontology [21].

The following example consists of a small organisation, which has two departments and two processes (see Figure 2). Two abstract classes are defined: “departments” and “processes”, which is achieved with the two prefixes defined in the second and third line of the listing below. Furthermore, there are two instances of the process class: “pro:invo\_check”, which represents the task of checking an invoice and

“pro:invo\_pay”, which represents the task of paying an invoice. In Addition, there are two instances of the department class: “org:dep\_fin” is the financial department and “org:dep\_hr” is the human resources department. The financial department (“org:dep\_fin”) is responsible for both tasks, which is defined by the relation “pro:responsible”. Further relationships can be added as needed.

For all concrete concepts in the ontology, we define a label, under which the concept can be displayed to a user. One very important requirement here is that these labels use the same domain thesaurus and syntactical naming conventions as described above to allow proper annotation of process model elements. To ensure that this requirement is met, one could, for example, derive the domain thesaurus automatically from an existing ontology on the one hand, or terminologically standardise an ontology based on an existing domain thesaurus using the above-mentioned methodology.

### 3.3 Methodology to Realise Automatic Annotations

Relying on the two requirements discussed above, we define the following steps as our methodology to enable automatic annotations of process models at design-time:

1. A phrase entered by the modeller is parsed into lexemes, which are then looked-up in the domain thesaurus.
2. From the domain thesaurus, a terminologically standardised phrase is generated.
3. The standardised phrase is used for naming process elements and to search for related concepts in the domain ontology.
4. Matching concepts are proposed to the modeller, who can decide whether or not the proposed concepts are appropriate for annotation. If so, the annotation link is established automatically.

Step 1 and 2 are described in Section 3.1 (see also Figure 1). These two steps ensure terminological standardisation and are in harmony with the methodologies already described in literature. Step 3 and 4 realise the automatic annotation itself.

If the modeller accepts one of the suggested phrases, this phrase is used to query the domain ontology for a matching concept (cf. Step 3 in Figure 2). Since labels in the ontology are already terminologically standardised, searching for related concepts is left to a simple string comparison. Considering the sample process shown in Section 3.1 and the sample ontology presented in Section 3.2, the follow-up steps 3 and 4 are shown in Figure 2. With the example phrase “pay invoice”, the ontology concept “pro:invo\_pay” is found, which describes the task of paying an invoice and hence matches the activity the modeller wants to label. In the last step, a link between the matching ontology concept “pro:invo\_pay” and the process element is created automatically if the modeller accepts the suggested annotation (cf. Step 4 in Figure 2). Finally, the function “Check invoice” is linked to the concept “pro:invo\_check”, the organisational unit “Financial department” is linked to the concept “org:dep\_fin” and the function “Pay invoice” is linked to the concept “pro:invo\_pay” from the domain ontology.

While searching for concepts with the exact same title is rather straight-forward, more sophisticated searches can be applied as well. Due to the use of terminological standardisation, we can also search for individual lexemes within the ontology. This

allows to also allows to suggest the user with concepts that do not fit their process element exactly, but to provide them with additional concepts known in the organisation that could fit the process. In consequence, it is also possible to annotate multiple concepts of different types, establishing relations to all process-related knowledge present in the organisation.

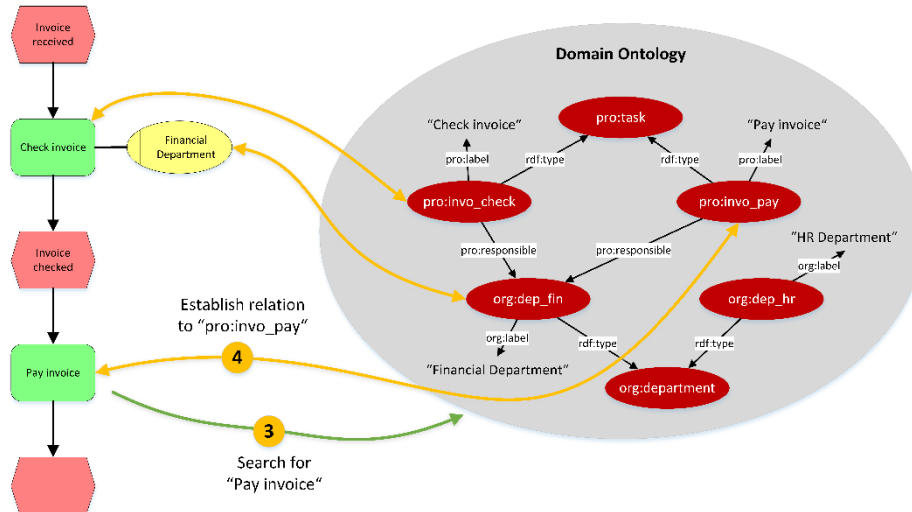


Figure 2. Automatic Annotation of Ontology Concepts

## 4 Limitations and Outlook

While our methodology provides automatic annotation of business processes with concepts from an ontology, it is limited to the presence of a terminologically standardised ontology. However, since ontologies are usually modelled as well (though probably not by process modellers), the same technique (without the automatic linking) can be applied to the process of modelling the ontology, ensuring a terminologically standardised ontology during creation already. Similarly, the technique could also be applied to already existing ontologies, which, however, involves human interaction to fix invalid, non-terminologically-standardized labels in the ontology.

A further limitation, which needs to be regarded more critically, is the fact that ontologies actually need to be “useful” to foster a common understanding of domain concepts. Such usefulness highly depends on the actual ontology and its contents. It is easy to state that a useful ontology should contain all necessary information, while not specifying which information are actually necessary. For an organisation, this might be all tasks performed within the organisation, all actors and stakeholders, all inter-process dependencies and all responsibilities. It is important that the information present in the ontology is correct, accurate and most importantly complete.

Until now, we have not yet discussed an ontology-based analysis of process models. Through the introduction of our automated annotation, links between process models

and ontologies no longer need to be created manually by means of extensive human work. The simplicity with which such links can now be created enables a whole new spectrum of business process analyses. Let us start with an example. Besides what is contained in the sample ontology presented in Section 3.2, tasks, departments and responsibilities, a useful ontology might include the information that an invoice needs to be checked before it can be paid. This literally means that the business process “Check invoice” is required to be executed before the business process “Pay invoice” can be executed. Other useful information an ontology could include, are goals which a process reaches when executed. This has already shortly been mentioned previously, when the process of submitting an invoice at the financial department leads to reaching the goal of submitting an invoice.

Keeping such ontology in mind, we can regard new aspects in the process of process modelling as well as in the process of analysing process models. Sticking to the example that invoices should only be paid if they have been checked previously, a process modeller could be given constructional assistance, i.e. by automatically suggesting them the task “pay invoice” after they have placed the task “check invoice” in their process model. While there are already a lot of papers on so-called recommender systems (e.g. [22–25]), these approaches currently learn from existing process models. Consequently, the quality of such suggestions falls with the quality of the existing models. Introducing a domain ontology to generation modelling suggestions from, bears great potential for better modelling suggestions.

Furthermore, process models could also be checked for compliance with business regulations or for inefficiencies or general flaws automatically. For this, compliant task sequences could be added to the ontology and query languages could be used to validate compliance. Besides inter-process dependencies, responsibilities could be validated as well. In the example shown in Figure 1, a suitable algorithm could notify the modeller that the activity “Pay invoice” (“Bill paying” before terminological standardization) is missing the organizational unit of the financial department, since the ontology has the information that the financial department is necessary for the task of paying an invoice.

While both the recommender system as well as the compliance analysis require a well-modelled ontology, we argue that due to the reusability of ontology in many different aspects, companies will be more willing to spend time and effort in creating a domain ontology of high quality. With our prototypical implementation, we have shown that our methodology actually works and can be used to work with at the design-time of process modelling. We are aware that this is only a limited evaluation, as we have applied it to artificial data only. In future research, we plan to apply our methodology to real-world processes and ontologies to see if it also works in practice. Since an in-depth evaluation requires more space, we plan to publish them as a separate paper including more details about experiences applying our methodology in practice.

Lastly, further research might also further investigate the domain thesaurus. Currently, we consider the domain thesaurus as an organization-dependant artefact, but it might be possible to reuse the domain thesaurus across different organisations. Researchers should analyse to which extent the thesaurus can be reused, across different organisations in the same industry or probably also across organisations in different industries. A high reusability would be beneficial under economic aspects.



## 5 Conclusion

We have presented a methodology to automatically annotate business process models with ontology concepts at design-time, by creating links between the process models and the ontology through terminological standardisation. Significantly reducing the effort which currently needs to be put into this by establishing these links manually, our methodology has the potential to improve future process modelling. Relations between ontology concepts and process models do not only help modellers of multi-national organisations to share a common understanding of business process models, but also enable further modelling support techniques and process model analyses, which – in this form – have not been possible with the techniques known in literature before.

While our paper has a conceptual perspective on automatic annotation, we have already implemented a prototypical artefact and used our methodology on artificially created models. In addition, we have shown areas for further research, namely to examine linguistic modelling assistance from a technical perspective, with the goal of providing and evaluating different algorithms for a possible implementation in the future. With our outlook to ontology-based model analysis, we have shown that our work sets the base for complex and advanced analyses in the area of compliance checking, which is an area that becomes more and more important for many organisations.

## References

1. Schreiber, G., Akkermans, H., Anjewierden, A., Hoog, R.D., Shadbolt, N.R., Wielinga, B.: Knowledge Engineering and Management: The CommonKADS Methodology. (2000).
2. Weske, M.: Business Process Management: Concepts, Methods, Technology. Springer, Berlin (2007).
3. van der Aalst, W.M.P.: Business Process Management: A Comprehensive Survey. *ISRN Softw. Eng.* 2013, 1–37 (2013).
4. La Rosa, M., Dumas, M., Uba, R., Dijkman, R.M.: Business Process Model Merging: An Approach to Business Process Consolidation. *ACM Trans. Softw. Eng. Methodol.* TOSEM. (2012).
5. Mendling, J., Strembeck, M., Recker, J.: Factors of process model comprehension-Findings from a series of experiments. *Decis. Support Syst.* 53, 195–206 (2012).
6. Batini, C., Lenzerini, M.: A Methodology for Data Schema Integration in the Entity Relationship Model. *IEEE Trans. Softw. Eng.* SE-10, 650–664 (1984).
7. Hadar, I., Soffer, P.: Variations in Conceptual Modeling: Classification and Ontological Analysis. *J. Assoc. Inf. Syst.* 7, 20 (2006).
8. Fellmann, M., Zarvic, N., Metzger, D., Koschmider, A.: Requirements Catalog for Business Process Modeling Recommender Systems. In: 12th International Conference on Wirtschaftsinformatik (WI 2015). pp. 393–407 (2015).
9. Studer, R., Benjamins, V.R., Fensel, D.: Knowledge Engineering: Principles and methods. *Data Knowl. Eng.* 25, 161–197 (1998).

10. Guarino, N.: Formal Ontology and Information Systems. In: Proceedings of the FOIS '98. pp. 3–15. , Trento, Italy (1998).
11. Greco, G., Guzzo, A., Pontieri, L., Saccà, D.: An ontology-driven process modeling framework. *Database Expert Syst. Appl.* 13–23 (2004).
12. Born, M., Dörr, F., Weber, I.: User-friendly semantic annotation in business process modeling. *Proc. Int. Workshop HumanFriendly Serv. Descr. Discov. Matchmaking HfSDDM 2007 8th Int. Conf. Web Inf. Syst. Eng. WISE 2007.* 260–271 (2007).
13. Fill, H.-G., Schremser, D., Karagiannis, D.: A Generic Approach for the Semantic Annotation of Conceptual Models Using a Service-Oriented Architecture: *Int. J. Knowl. Manag.* 9, 76–88 (2013).
14. Rizopoulos, N., McBrien, P.: A general approach to the generation of conceptual model transformations. *Adv. Inf. Syst. Eng.* 326–341 (2005).
15. Bögl, A., Kobler, M., Schrefl, M.: Knowledge acquisition from EPC models for extraction of process patterns in engineering domains. In: *iProceedings of the 4th Multikonferenz Wirtschaftsinformatik (MKWI 2008).* pp. 1601–1612 (2008).
16. Leopold, H., Eid-Sabbagh, R.H., Mendling, J., Azevedo, L.G., Baião, F.A.: Detection of naming convention violations in process models for different languages. *Decis. Support Syst.* 56, 310–325 (2013).
17. Pittke, F., Leopold, H., Mendling, J.: Automatic Detection and Resolution of Lexical Ambiguity in Process Models. *IEEE Trans. Softw. Eng.* 41, 526–544 (2015).
18. Delfmann, P., Herwig, S., Lis, L.: Unified Enterprise Knowledge Representation with Conceptual Models - Capturing Corporate Language in Naming Conventions. In: *Proceedings of the 30th International Conference on Information Systems (ICIS 2009)* (2009).
19. Leopold, H., Meilicke, C., Fellmann, M., Pittke, F., Stuckenschmidt, H., Mendling, J.: Towards the Automated Annotation of Process Models. In: *Advanced Information Systems Engineering: 27th International Conference, CAiSE 2015.* pp. 401–416. , Stockholm, Sweden (2015).
20. Kugeler, M., Rosemann, M.: Fachbegriffsmodellierung für betriebliche Informationssysteme und zur Unterstützung der Unternehmenskommunikation. *Informationssystem Archit. Fachaussch.* 5, 8–15 (1998).
21. Giacomo, G.D., Lenzerini, M.: TBox and ABox Reasoning in Expressive Description Logics. In: *Proceedings of the 5th International Conference on Principles of Knowledge Representation and Reasoning (KR 96)* (1996).
22. Chan, N.N., Gaaloul, W., Tata, S.: Assisting business process design by activity neighborhood context matching. In: *International Conference on Service-Oriented Computing.* pp. 541–549. Springer (2012).
23. Koschmider, A., Hornung, T., Oberweis, A.: Recommendation-based editor for business process modeling. *Data Knowl. Eng.* 70, 483–503 (2011).
24. Li, Y., Cao, B., Xu, L., Yin, J., Deng, S., Yin, Y., Wu, Z.: An Efficient Recommendation Method for Improving Business Process Modeling. *IEEE Trans. Ind. Inform.* 10, 502–513 (2014).
25. Smirnov, S., Weidlich, M., Mendling, J., Weske, M.: Action patterns in business process model repositories. *Comput. Ind.* 63, 98–111 (2012).