Model-Based Domain Ontology Engineering

Janina Fengel
Darmstadt University of Applied Sciences
Hohe Strasse
D-64807 Dieburg
+49 6151 16 9458
janina.fengel@h-da.de

Michael Rebstock
Darmstadt University of Applied Sciences
Haardtring 100
D-64295 Darmstadt
+49 6151 16 8392
michael.rebstock@h-da.de

ABSTRACT
Business Process Models describe sequences of activities, expressed in a certain modeling language, with the model elements being labeled following the business terminology in use in the applicable domain. In case no predefined vocabulary or rules for assigning those labels are in place, terms are chosen individually on a case-by-case-basis. As a result, models are often semantically heterogeneous concerning the domain language. For resolving these ambiguities when comparing or integrating models, support through a consensus terminology is required. We show in this discussion paper how to reuse models to capture the domain knowledge contained and relate it for usage and further collaborative evolution, so that over time an authentic domain ontology can emerge.

Categories and Subject Descriptors
H.4.0 [**Information Systems**]: Information Systems Applications – *General.*

General Terms
Management, Standardization, Languages.

Keywords
Business process modeling, semantic heterogeneity, domain ontology, collaborative ontology engineering, ontology emergence.

1. INTRODUCTION
Over the past decades, the modeling of business processes has become an indispensable means of conceptualizing and designing business organizations and the subsequent engineering of appropriate IT-support. Since over time various different modeling languages have been developed, as a result differing legacy models exist. However, conceptual models such as business process models are an important source of information and automated support for retrieving the knowledge contained increases the models' usefulness [3]. Unfortunately, in practice, such processing is frequently hindered, as models originate from different sources and application areas and, very often, differ considerably not only syntactically, but mostly semantically [19]. Semantic ambiguity arises through the natural language chosen for labeling the elements within models, but also through the semantic differences between the constructs of the various modeling languages chosen for building those models [23]. Hence, a lot of models are semantically incompatible, especially when several modelers or decentralized teams are involved [13]. Variations occur in the terms selected, the natural language chosen and encoding or in form of synonymy or homonymy, next to differences in granularity, scope or modeling style and the perspective and intended usage of a model.

As a solution for preventing such semantic differences, the use of agreed-upon sets of terms for creating labels for model elements has been suggested [23]. Such a set of elements may be a business or domain model on the conceptual, computation independent level [17], a domain specific language [19], or more generally a domain ontology [21; 14] or even a specific enterprise ontology [7; 12]. These approaches assume the prior existence or a separate top-down development of such domain models. However, the creation of a centrally defined universal umbrella, or more specific, domain model, cannot solve the problem of semantic interoperability completely. The development and maintenance of an ontology demands time- and costwise huge efforts. Furthermore, models are already often effectively in use and can therefore not be easily recreated or amended according to new guidelines. Consequently, more pragmatic and thus easier applicable approaches are needed for resolving semantic heterogeneity in the area of the domain language present in business process modeling. We here propose a bottom-up approach at establishing a knowledge model of the business terminology by using the wealth of domain knowledge hidden in existing business process (and other) models. The resulting comprehensive conceptual knowledge model can be collaboratively improved and further refined into a domain ontology. In Section 2 the technological foundations are presented. In Section 3 their application for realizing our approach is described, followed in Section 4 by showing how this relates to other works. We conclude with a brief discussion of our proposition and an outlook onto the next steps.

2. SEMANTIC HETEROGENEITY IN BUSINESS PROCESS MODELS
Frequently, business process models have to be matched for the purpose of integration or reuse [14], e.g., in business-IT-alignment scenarios, when new systems are implemented on existing processes or when reference models are introduced. Also, process and application integration following reorganizations or mergers and acquisitions calls for model matching.

The goal is to semantically synchronize the intended meaning of existing models. Often, semantic mediation is needed for answering questions regarding how often a certain business object is being part of which models (e.g. "How many times is an incoming invoice being checked?") or if a term in use is similar to another one in use (e.g. "Is a proposal an offer or a tender and how does it relate to an RfQ?").

2.1 Business Process Modeling

For creating a business process model, awareness of the *modeling language* specifics and of the applicable business terminology is required. The terminology to be used can be contained in a *domain language*. Such a domain language describes the universe of discourse to be represented by the model [3]. Thus, process models may be viewed as a composition of a set of *model elements* and *model element labels*. Figure 1 depicts the composition procedure.

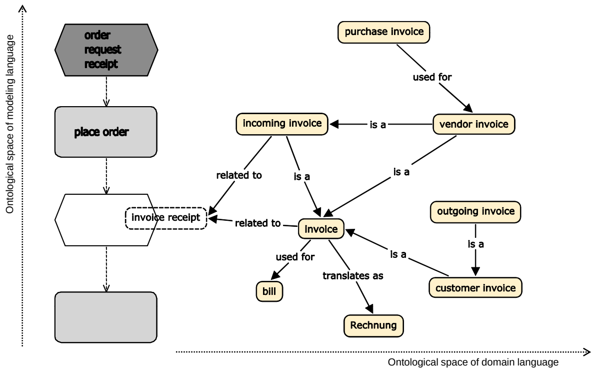

Figure 1. Business Process Model Composition

A process model possesses two ontological spaces: the domain language expresses specific conceptual knowledge of the problem domain, while the modeling language provides a means for sorting it. Even though business process modeling languages differ considerably in detail, the basic concepts of some of the prominent ones resemble each other, even though these similar notions are designated differently. All these languages offer a graphic visualization and enable describing sequences of activities. In doing so, process models are rather work-flow-oriented, as they focus more on the technical realization [14]. As a result, business process models can be migrated from one language to another [16; 9; 2] or transformed for realization and implementation purposes [17]. Such operations are performed without the need for a separate handling of the semantics of the domain language, it is taken as is and carried forward. This allows for looking at it separately, thus providing the means for expressing domain content in different modeling languages without losing its meaning. Hence, the knowledge explicated within can be drawn on for further use. We suggest employing semantic technologies, in particular ontology matching and mapping, for capturing and improving the contained domain knowledge.

2.2 Ontologies for Knowledge Representation

For reaching mutual understanding about models, the creation of a collection of relevant domain terms as a domain language in form of a *domain model* has been often suggested in the literature, e.g., in [10; 24; 17; 21; 14; 5]. However, the design and development of such a model is a non-trivial task, as substantial work has to be done for determining the meaning of terms [1]. Also, these tasks are repetitive, as changes over time need to be considered. As an approach for capturing semantics, the use of ontologies is becoming popular, as they express declarative knowledge in machine-processable form for sharing and reusing it, e.g., in corporate settings [24]. A domain model represented as a *domain ontology* allows for further semantic processing. Ontologies can have different degrees of formality. With increasing amount of meaning specified and thus higher degree of formality, ambiguity decreases, and ontologies become more expressive [15; 24]. From a pragmatic point of view, for the purpose of being usable as an ontology, the domain model needs to be a formal ontology, so that it can be reasoned upon. For this purpose, the description logic based OWL DL is often chosen. By means of inference engines, new knowledge can be derived through logical deduction [10]. Furthermore, expressing collections of concepts in ontological form allows for applying ontology engineering technologies, in particular matching and mapping, for establishing logical relations between the elements of one or more ontologies for the purpose of comparing and reconciling them. Matching systems try to find pairs of entities from different ontologies with the same intended meaning [8]. In doing so, semantic correspondences are looked for, which express equivalence or similarity between elements and can serve as mappings between the elements of different ontologies. A comprehensive overview can be found in [8]. The thus aligned ontologies can remain in their original form, as the mappings can further on serve as a means to a virtual integration or be used for a new ontology to be merged.

3. DOMAIN ONTOLOGY EMERGENCE

Usually, the creation and ongoing care of a domain ontology is a major task. In the past, the design and collection of knowledge has often proved to be a huge challenge [1]. To ease this workload, we suggest to automate this step by reusing the conceptual domain knowledge contained in existing business process models and relating it semantically. However, the results produced by matching tools are not perfect. Even though the process of finding such relations can be automated by means of ontology matching tools, the resulting ontology mappings found are usually not ideal as they may be ambiguous or incorrect. Therefore, for optimization and disambiguation, human involvement is required [25]. Especially in a group of experts concerned with modeling, capturing the implicit part of domain knowledge provides important subject matter expertise needed for resolving uncertainties and ambiguities, as in general exploiting such collective intelligence facilitates knowledge growth [11]. Through active user participation enablement by way of adding, editing information and providing feedback on relations found, this otherwise hidden knowledge is extracted, shared and made usable for quality enhancement and subsequent ongoing formalization of the ontology built.

3.1 Integrative Ontology Development

The core idea of our approach is to extract the model element labels from existing models for using them as ontology building blocks. They can be matched and mapped and subsequently improved. Thereby, the basis for a domain ontology is developed without initial ramp-up effort by combining the computing power of automated matching with the correctness of users manually enhancing the initially derived information at the time of using the ontology. This iterative evolution process is comprised of several steps, which are shown in the following. Overall, the creation of an initial basis enhanced with further input helps to evoke the emergence of a shared understanding and conceptualization of the actual applicable domain language. We call this "*ontology emergence*", since over time the tendency towards a mutually accepted language use becomes evident.

3.1.1 Concept Elicitation

Modeling tools are mostly capable of exporting models in XMI format. Therefore, transformations into OWL DL can be performed through applying XSLT. In a pragmatic attempt the domain knowledge residing in models is extracted in that the model element labels are transformed into ontology classes. This abstracts from the statement a model intends to do and leaves the model as is. Labels are presently taken without further processing, so that not only terms, but also common expressions are transferred. As a result, very lightweight individual ontologies are created. Each of them is a knowledge model of the concepts in use in a particular business process model.

3.1.2 Matching and Mapping

In a next step, these single pieces are related by means of ontology matching. Instead of having to compare labels manually and establish mappings for explaining or exchanging them, an initial set of mappings can be established automatedly. A mapping describes the semantic relation between two classes of two different ontologies and shows its confidence for qualifycation. Furthermore, by means of inference, new mappings may be deduced. Thus, a first lightweight coherent ontology evolves, serving as a bridge between models and providing semantic relations.

3.1.3 Ontology Application

For analyzing and comparing models, the domain ontology created provides support for semantically disambiguating model element labels and detecting relations. Basically, the search for semantic correspondences is as easy as searching a dictionary. The query for a term or an expression composed of several terms shows how they are used and which terms are semantically related. Users may rate the automatedly computed information, so that a dynamically computed acceptance value attached to a mapping reflects users' assessment of its appropriateness. The result provides user guidance in selecting labels upon creating new business process models and the facility to disambiguate and synchronize labels from existing models.

3.1.4 Collaborative Evolution

In case mappings present in the ontology are not satisfactory or sufficient, users may add and edit mappings themselves. Exploiting the skills of modelers and model users alike allows for evaluation of the automatically derived mappings, thus improving their quality. This is accomplished through active input as well as, presently in a simple way, through the feedback provided in form of ratings regarding the appropriateness of mappings found. Thus, over time their reliability becomes obvious. The dynamics of real life terminology use can be captured when new concepts are introduced through user input. Future research includes the addition of context information to ratings given, so that ambiguous mappings' applicability can be determined according to the domain perspective.

3.1.5 Ontology Progression

The initial ontology can grow over time through further adding of new models. Additionally, knowledge engineers can take this basis as a starting point for developing a business specific, fully formalized ontology. More elaborate relations can be added, e.g., similar to the possibilities available for creating thesauri, or any other further formalization. Considering that corporations are mostly hierarchically organized, changes and amendments from top-down can be incorporated, such as changes in the terminology due to marketing or policy requirements.

3.2 Realization

Over the past years, extensive work has been done in developing ontology engineering tools and frameworks. Building on these foundations, we are implementing our solution for facilitating the reconciliation of differing terminology as described above. The basis is a dynamically emerging and growing domain ontology that does not require initial manual creation efforts. Upon conceptualizing our system, we could draw on prior experience gained from reconciling e-business standards in the field of B2B e-commerce. Thereby, we found a modular architecture to be highly flexible and adaptable to specific needs [20]. The goal is to provide a framework by way of reusing existing technologies in combination with our own developments for overall collaborative usage and evolution. Therefore, our model integration framework allows connecting different mapping, inference and storage tools by an adapter mechanism as individually needed. The system is accessed through a web service interface, thus allowing integration into arbitrary modeling tools. The framework is implemented in Java on top of the Jena2 framework. It offers a standard interface for daily operation and an expert interface for administration, maintenance and knowledge work purposes.

4. RELATED WORK

Often, efforts in matching models have concentrated on the aspect of model language semantics, e.g., in [16; 9; 18]. Other works propose the use of a pre-defined business terminology [5; 2; 23]. In the field of ontology engineering, the inclusion of ontology learning [6] and user participation [4; 22] has been suggested. To the best of our knowledge, so far no combination has been proposed.

5. CONCLUSION AND OUTLOOK

In our approach shown here we have combined collaborative and semantic technologies onto the challenges involved in achieving a semantically unambiguous use of domain terms for modeling business processes. Through developing a method and supporting system for reusing existing models and relating the conceptual knowledge contained, a possibility for obtaining a domain ontology without huge prior efforts is shown. By way of actively including its users, the shortcomings of automated knowledge computing can be overcome. Our next steps are the finalizing of the implementation works by adding user support in form of semantic net visualization and the extension to include knowledge processing of further types of models.

6. ACKNOWLEDGMENTS

Our work is supported by the German Federal Ministry of Education and Research (BMBF) under grant no. 1728X07 MODI.

REFERENCES

[1] D'Aquin, M., Motta, E., Sabou, M., Angeletou, S., Gridinoc, L., Lopez, and V., Guidi, D. 2008. Toward a New Generation of Semantic Web Applications. IEEE Intell Syst (May/June 2008), 20–28.

[2] Becker, J., Bergener, P., Lis, Ł., Pfeiffer, D., and Räckers, M. 2008. Migrating Process Models between PICTURE and BPMN/EPC. In Niedzwiedzinski, M., Lange-Sadzinska, K. (eds.) Wybrane Problemy Elektronicznej Gospodarki. Lódz, 9–20.

[3] Becker, J. and Pfeiffer, D. 2007. Automatic Knowledge Retrieval from Conceptual Models. In Proper, E., Halpin, T., and Krogstie, J. (eds.) 12th Intern. Workshop on Exploring Modeling Methods for Systems Analysis and Design (EMMSAD'07) in conjunctiun with 19th Conference on Advanced Information Systems (CAiSE'07). CEUR-WS 365, 153–162.

[4] Braun, S., Schmidt, A., and Walter, A. 2007. Ontology Maturing: a Collaborative Web 2.0 Approach to Ontology Engineering. In Proceedings of the Sixteenth Intern. World Wide Web Conference (WWW2007) Workshop Social and Collaborative Construction of Structured Knowledge.

[5] Brockmans, S., Ehrig, M., Koschmider, A., Oberweis, A., and Studer, R. 2006. Semantic Alignment of Business Processes. In Manolopoulos, Y. (ed.) ICEIS 2006 - Proceedings of the 8th Intern. Conference on Enterprise Information Systems: Databases and information systems integration, 191–196.

[6] Cimiano, P., Hotho, A., Stumme, G., and Tane, J. 2004. Conceptual Knowledge Processing with Formal Concept Analysis and Ontologies. In Eklund, P.W. (ed.) Proceedings of The Second Intern. Conference on Formal Concept Analysis (ICFCA 04) "Concept lattices". Springer, Berlin, 189–207.

[7] Dietz, J. L. G. and Hoogervorst, J. A. P. 2008. Enterprise Ontology in Enterprise Engineering. In Wainwright, R.L., Haddad, H.M. (eds.) Proceedings of the 2008 ACM symposium on Applied computing. ACM Press, New York, 572–579. http://doi.acm.org/10.1145/1363686.1363824

[8] Euzenat, J. and Shvaiko, P. 2007. Ontology Matching. Springer, Berlin.

[9] Gehlert, A. 2007. Migration fachkonzeptueller Modelle. Logos-Verl., Berlin.

[10] Goméz-Pérez, A., Fernandéz-López, M., Corcho O. 2004. Ontological Engineering. Springer, London.

[11] Gruber, T. (2007). Collective knowledge systems: Where the Social Web meets the Semantic Web. Web Semantics: Science, Services and Agents on the World Wide Web 6(1), 4–13.

[12] Grüninger, M., Atefi, K., and Fox, M. A. 2000. Ontologies to Support Process Integration in Enterprise Engineering. Computational & Mathematical Organization Theory 6, 381–394.

[13] Hadar, I. and Soffer, P. 2006. Variations in Conceptual Modeling: Classification and Ontological Analysis. Journal of the AIS 7(8), 568–592.

[14] Hepp; M. and Roman, D. 2007. An Ontology Framework for Semantic Business Management. In Oberweis, A., Weinhardt, C.; Gimpel, H. (eds.) eOrganisation. Service-, Prozess-, Market-Engineering. 8. Intern. Tagung Wirtschaftsinformatik (WI 2007). Univ.-Verl., Karlsruhe, vol. 1, 423–440.

[15] McGuiness, D. L. 2003. Ontologies Come of Age. In Fensel, D.; Hendler, J.; Lieberman, H.; Wahlster, W. (eds.) Spinning the Semantic Web: Bringing the World Wide Web to Its Full Potential. MIT Press, Cambridge, 171–193.

[16] Murzek, M. and Kramler, G. 2007. The Model Morphing Approach – Horizontal Transformations between Business Process Models. In Nummenmaa, J., Söderström, E. (eds.) Proceedings of the 6th Intern.Conference on Perspectives in Business Information Research (BIR'2007). Report D-2007-13. University of Tampere, Finland, 88–103.

[17] OMG 2003. MDA Guide Version 1.0.1 12th June 2003. http://www.omg.org/docs/omg/03-06-01.pdf.

[18] OMG 2006. MOF QVT Final Adopted Specification. http://www.omg.org/docs/ptc/05-11-01.pdf.

[19] Pfeiffer, D. 2007. Constructing Comparable Conceptual Models with Domain Specific Languages. In Österle, H., Schelp, J., Winter, R. (eds.) 15th European Conference on Information Systems (ECIS2007) Relevant rigour – Rigorous relevance.

[20] Rebstock, M., Fengel, J., and Paulheim, H. 2008. Ontologies-Based Business Integration. Springer, Berlin.

[21] Saeki, M. and Kaiya, H. 2006. On Relationships among Models, Meta Models and Ontologies. In Gray, J., Tolvanen, J. -P; Sprinkle, J. (eds.) Proceedings of the 6th OOPSLA Workshop on Domain-Specific Modeling. http://www.dsmforum.org/events/DSM06/Papers/14-saeki.pdf.

[22] Siorpaes, K. and Hepp, M. 2007. myOntology: The Marriage of Ontology Engineering and Collective Intelligence. In: Proceedings of the ESWC 2007 Workshop "Bridging the Gap between Semantic Web and Web 2.0".

[23] Thomas, O. and Fellmann, M. 2007. Semantic Business Process Management: Ontology-Based Process Modeling Using Event-Driven Process Chains. IBIS Interoperability in Business Information Systems 2(1), 29–44.

[24] Uschold, M. and Grüninger, M. 1996. Ontologies: principles, methods, and applications. Knowl Eng Rev 11(2), 93–155.

[25] Zhdanova, A., de Bruijn, J., Zimmermann, K., and Scharffe, F. 2004. Ontology Alignment Solution v2.0. Esperonto project deliverable D1.4 V2.0. http://www.deri.at/fileadmin/documents/deliverables/Esperonto/Del1.4-V2.0-final.pdf.