On the Social Network based Semantic Annotation of Conceptual Models

Hans-Georg Fill

University of Vienna - RG Knowledge Engineering, 1090 Vienna, Austria
hans-georg.fill@univie.ac.at,
WWW home page: http://homepage.dke.univie.ac.at/fill

Abstract. The semantic annotation of conceptual visual models permits to leverage the contained semantic information to a machine processable level. At the same time the intuitive and non-technical nature of graphical models can be maintained. Although this leads to direct benefits in terms of potential analysis functionalities, the addition of annotations for large model corpora requires quite some effort by the modelers. In order to contribute to the efficiency when adding semantic annotations, we propose an approach that links information on semantic annotations to information available from social network applications. In this way the addition of semantic annotations can be facilitated in terms of communication support, economic benefits, and technical opportunities. For the design of the approach we use a specifically developed modeling method, which allows to reason about the requirements at a high abstraction level and permits to add more formal specifications, e.g. to define behavior. A first prototype of the modeling method has been implemented using the freely available ADOxx meta modeling platform based on the SeMFIS toolkit.

Keywords: Semantic annotation, Conceptual Model, Social Network Application, Meta Modeling, ADOxx

1 Introduction

In the past, several areas in information systems research and practice have made use of conceptual visual models. Prominent examples can be found in the field of requirements analysis and engineering [30], business process management and business-IT alignment [25] as well as in software engineering and IT architecture management [26]. Among their many applications they are used to support the communication between developers and users [30], to help analysts understand and evaluate a domain by simulating certain behavior [19], for documenting requirements and for the configuration and the engineering of IT systems, e.g. to realize model-driven-architectures. A core aspect of these types of models is that they are directed towards supporting human communication and understanding and do not aim for an entirely machine processable representation of a domain [27]. They are thus based on a formal syntax that is complemented by formal semantic defintions where needed. In this way the models can be used by domain experts in an intuitive way and then gradually enriched with formal semantics depending on the required machine processing capabilities.

One particular type of such enrichments are *semantic annotations* that offer a way to make unstructured natural language information contained in models processable [15]. The main idea hereby is to add mappings to formal semantic schemata without changing the original structure and purpose of conceptual models but to provide semantic processing functionalities as needed. Especially in business process management where conceptual models are used extensively in many companies, several approaches have been described for applying and using semantic annotations. Besides tasks such as semantic similarity matching of models [6], the user-centric visualization of process views [15], or the automation of processes using web services [18], semantic annotations have recently also been used to conduct complex process analysis, e.g. for risk management [10].

Despite the potential advantages the effort for adding this specific semantic information is still very large. It is therefore of particular interest to enhance this process and thus increase its efficiency. Although several techniques have been developed for suggesting annotations based on natural language processing or context-based similarity matching, the user still needs to select the correct annotation based on his or her knowledge. Another direction to enhance the process is by supporting *human annotators* in a way that they can perform the annotations more efficiently. Besides organizational measures, also the use of crowd-sourcing mechanisms that distribute the annotation tasks to a multitude of people may offer potential benefits for this purpose. However, to realize such approaches and ideally combine them with the automated techniques for bringing up suggestions for possible annotations, the appropriate technical foundations have to be established and made available to the community.

Following ideas presented in [16], we describe an approach that integrates the information provided by social network applications with semantic annotations. In particular we build upon the SeMFIS approach for conducting loosely-coupled semantic annotations of arbitrary conceptual models using separate annotation models [11]. By integrating information on actors, resources, and groups from social network applications, additional functionalities for supporting the definition of semantic annotations can be provided. This concerns in particular functionalities for supporting the collaboration on annotation tasks and for the personalization of the user experience. In addition, the integration of social network applications provides technical opportunities through re-using services such as the workflows for user authentication. In order to analyze the requirements and dependencies of such an approach, we extend the SeMFIS modeling method and present a first implementation on the ADOxx meta modeling platform.

The remainder of the paper is structured as follows: In section 2 we will briefly discuss some foundations for our approach. Subsequently, in section 3 the approach itself is presented which is then discussed in section 4. The paper ends with a conclusion and an outlook in section 5.

2 Foundations

In order to clarify some of the terms we are going to use within the scope of this paper in regard to conceptual models, their semantic annotations, and social network functionalities, we will briefly outline these three areas.

2.1 Conceptual Visual Models

For characterizing the core constituents of conceptual models we refer to a framework that has been been described by Karagiannis and Kuehn [23]. In this framework the top level concept are modeling methods which consist of a modeling technique and mechanisms and algorithms. The modeling technique is further specialized by a modeling language and an according modeling procedure. The modeling language is then defined by its syntax, semantics, and visual notation and the modeling procedure by its steps and results. In contrast to other approaches - as e.g. described in [17] - the notation of the modeling language defines the graphical representation of the syntax and may thus also be modified independently of the syntax [9].

The semantics of the modeling language is assigned to the syntax by mapping the elements of the syntax to a semantic schema. This schema may be natural language or may be some formal semantics, e.g. when using a formal schema such as an ontology. The mechanisms and algorithms of the modeling method can be specialized by generic, specific, and hybrid mechanisms and algorithms. Thereby, generic mechanisms and algorithms can be applied to any modeling language, specific mechanisms and algorithms only to a particular subset of modeling languages and hybrid mechanisms and algorithms can be parameterized to be applicable for several modeling languages.

The syntax of a modeling language can be further specialized by an abstract syntax, which may be represented by a meta model, and the concrete syntax, which is represented by the resulting model instance. The meta model is thus a model that defines the language for expressing a model. The model is then the concrete realization using this language.

2.2 Semantic Annotation of Conceptual Models

Based on these definitions we can now detail the notion of semantic annotations. The grammars of conceptual models traditionally give their users a large degree of freedom when it comes to describing their content. Therefore, the labels and attached comments of the elements and relations are often expressed in natural language. It may however be beneficial to be able to process the information contained in these descriptions at a later stage. To make the information contained in these descriptions machine processable one approach is to add mappings to formal semantic schemata [8]. Thereby, the semantic information can be raised to a machine processable level without requiring a modification of the original modeling language. This has two particular advantages. The first is that the consistency to other models that are based on the original specification of the modeling language is ensured. It also means that any algorithms that were based on the original state of the language do not need to be adapted. The second advantage is that these annotations and according processing capabilities can also be added ex-post, i.e. after the conception of the modeling language and after the creation of according model instances. It is thus not necessary for the user to deal with formal semantic definitions at first hand and only add them as needed.

For the representation of the formal semantic schema it is common to use ontologies which define the vocabulary used to describe and represent an area of knowledge. They are usually expressed in a logic-based language that permits to make fine, accurate, consistent, sound, and meaningful distinctions among the elements of the ontology [29]. Furthermore, ontology languages may permit to conduct reasoning tasks to ensure the consistency of the ontology and automate the classification of new terms [20]. Today it can be chosen from a wide variety of ontology languages including W3C standards such as OWL or RDF.

Several benefits have been described by using semantic annotations. During the stage of model creation the ontologies can be used to check whether the attributes of the model elements are valid [1] or to provide auto-completion functionalities for increasing the efficiency of modelling [2]. For the analysis of models semantic annotations have been described to measure the similarity between process models [6] and to validate models against formal specifications [22]. The benefits for the execution of processes using semantic annotations have been described for the case of semantic web services by [5].

2.3 Social Network Based Applications

In the last years several *social network sites* have emerged. They are commonly defined as "web-based services that allow individuals to (1) construct a public or semi-public profile within a bounded system, (2) articulate a list of other users with whom they share a connection, and (3) view and traverse their list of connections and those made by others within the system" [3][p. 211]. Due to their enormous participation rates they have achieved so far, they have been an attractive subject for researchers as well as the advertising and media industry.

More recently, the opportunities of *social network based applications* have been investigated [28]. These are software applications that access the data of social network sites either by using proprietary protocols such as the Facebook Graph API [7] or open source third-party APIs. The data provided by the social network sites can then be used to retrieve information about a user's personal profile as well as information about the profile of his or her fellows and other resources.

Depending on the social network site and also which information the user is willing to share, this includes for example information about the user's age, spoken languages, home town, current workplace or his or her interests in literature, philosophy or music. In addition, several of the social network sites allow users to form special interest groups or to group together by expressing that they like a particular resource on the social network such as a fan page. This data can then be analyzed and used for scientific analyses [24], e.g. for determining subgroups with similar properties [21].

3 Social Network based Semantic Annotation of Conceptual Models

With the foundations presented in the previous chapter, we can present our approach as a synthesis of a. conceptual models, b. their semantic annotations with ontologies, and c. social network applications. The core idea as shown in figure 1 is to use the information provided through social network applications in conducting semantic annotations. In order to add semantic annotations to conceptual models, additional knowledge needs to be provided by human actors. These human actors in turn - exemplarily numbered from 1 to 7 in the figure - have connections to other actors and other social network ressources, e.g. in the way of subscriptions to information feeds in the network. These connections are made explicit by the data structures provided by the social network applications. This information therefore be used to support the tasks in conducting the annotations.

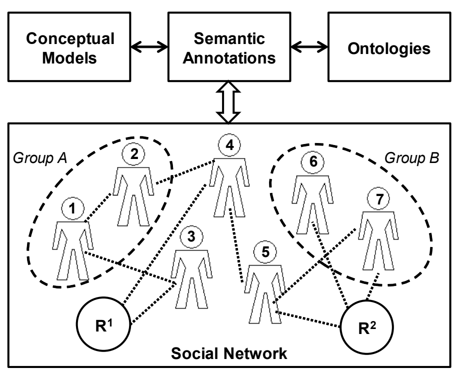

Fig. 1. Major Components of the Proposed Approach

For detailing the relationships between conceptual models, semantic annotations, ontologies and social network data we will use in the following a semi-formal, model-based representation [4, 23]. This will not only permit us to discuss how the three parts interact in detail and which information is accessed by each part. It will provide a foundation for subsequently specifying how to process the contained information in a formal way.

3.1 Model-based Representation

To represent the information about the linkage of conceptual models, semantic annotations, ontologies, and social network applications we designed a specific modeling language as an extension of the SeMFIS modeling language [11] - an excerpt of its meta model is shown in figure 2. To illustrate the application to particular conceptual models, this modeling language contains a simplified version of the BPMS modeling language for business processes [13] – shown in the upper left of figure 2. It provides elements for representing the *information flow* and the *control flow* of a business process as well as the according *sequence flow* relations.

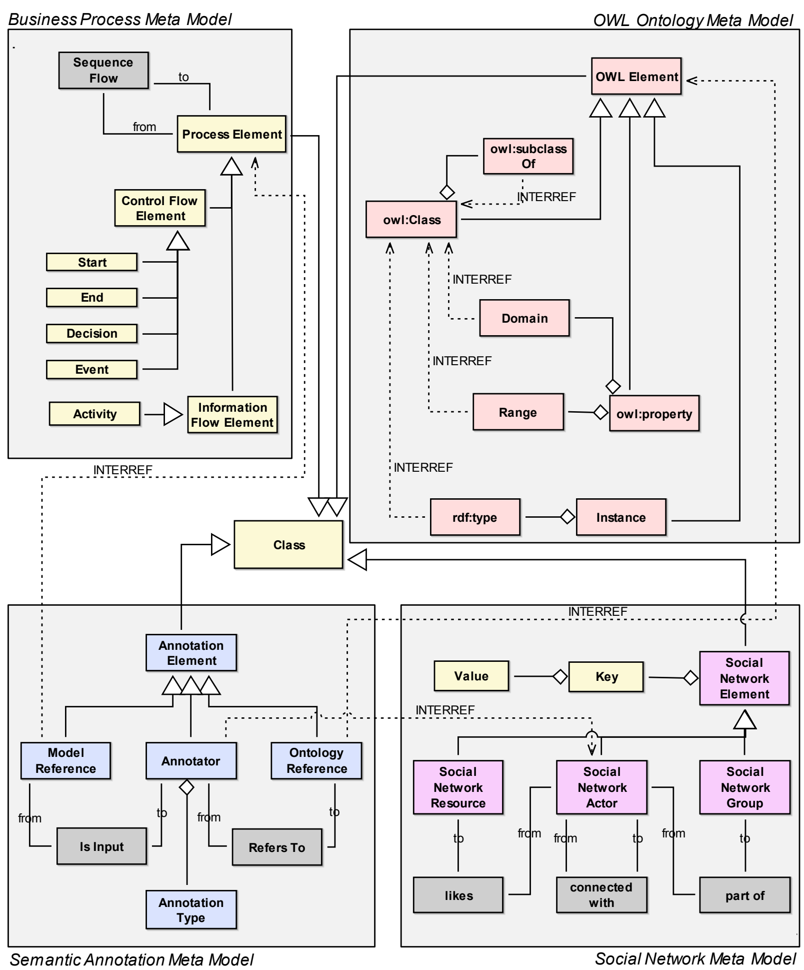

Fig. 2. Excerpt of the Alignment between a Business Process, an OWL Ontology, a Semantic Annotation, and a Social Network Meta Model

Furthermore, to represent ontologies that can be later used for the semantic annotations of the conceptual models, an OWL ontology meta model is added for representing ontologies in the web ontology format – see the upper right corner in figure 2. The meta model does not provide any formal semantic definition for the OWL ontologies. It is assumed that a consistent and sound representation of an OWL ontology is available, e.g. by using an import from an ontology management toolkit such as Protégé [12]. For the representation of the annotations, the meta model provides an *annotation element*. This is specialized in the form of a *model reference*, an *ontology reference*, and an *annotator* element. Thereby, the annotation can be specified independently both of the used conceptual modeling language and the used ontology language. The annotation can be further detailed by specifying an *annotation type*, e.g. to express is-broader-than or is-narrower-than relationships.

For integrating the information from social network applications, the meta model is complemented by a *social network element* and its specializations. These are used to represent the core concepts of social network websites. Thereby the following relations can be represented: Between *social network actors* by using the *connected* relation, between *social network actors* and *social network resources* by using the *likes* relation, and between *social network actors* and *social network groups* by using the *part of* relation. Additionally, each *social network element* can have *key/value* pairs assigned to it. These can be used to integrate the range of attributes that are available for detailing the entities in the social network, e.g. the personal data of the actors or the description of a group. To integrate the semantic annotations and the social network entities, a reference relation (*INTERREF*) is shown between the *social network element* and the *annotator* element. This permits to express that an annotation is conducted by an actor in the social network.

The meta model has been implemented as an extension of the SeMFIS toolkit on ADOxx - see figure 3 [11, 13]. Based on this implementation concrete model instances of the meta model elements can be created and used as input for the development of algorithms for processing this data. Thus, semantic model annotations respectively the information provided by social network applications, can be represented and linked using the references as defined by the meta model. In addition to the early analysis of these relationships, the meta model serves as a starting point for further implementations.

3.2 Addition of Behavior

With the information structures provided by the semi-formal meta models, more formal specifications can be realized. For example, the behavior can be specified using a rule-based approach as shown in the following. Based on the linkage of semantic annotations of model contents with the actors in the social network, it could be of benefit to identify other actors in the network who engage in similar annotation tasks. Thus, several additional scenarios can be realized. E.g., it could be searched for actors who possess similar knowledge based on their annotations to incite communication between actors with similar interests; or, existing semantic annotations could be proposed for review and evaluation to other actors for quality or correctness assessments. Another scenario would be to

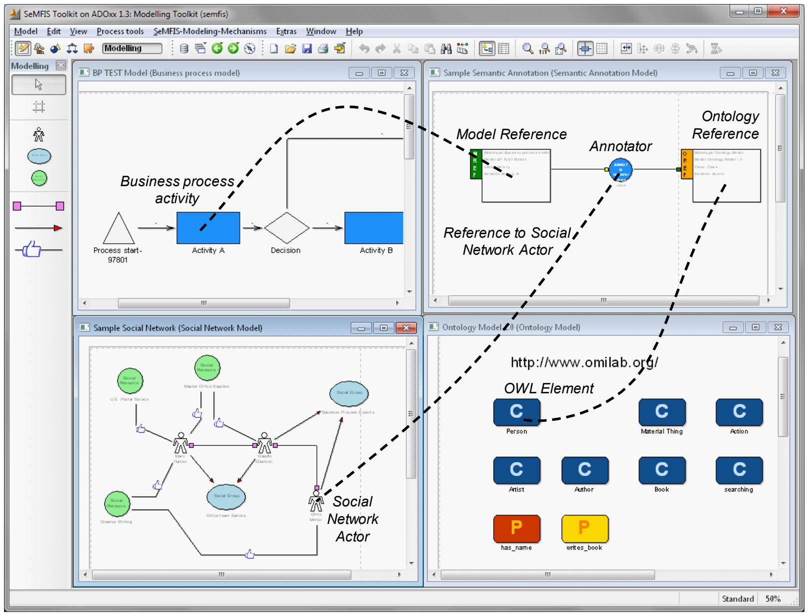

Fig. 3. Implementation of the Meta Models on the ADOxx Meta Modeling Platform

propose the membership in a group of the social network for actors with similar interests.

In the following we outline a formal definition of rules based on the structures in the meta model - a more thorough formal foundation may be achieved with the FDMM formalism but is omitted here due to limited space [14]. At first, we define the antecedent that has to be met. The rules thus requires the existence of two social network actors a, b who engage in semantic annotation tasks on process elements p_1, p_2. The annotations are specified by model references m_1, m_2, ontology references r_1, r_2 and annotator elements t_1, t_2. It is assumed by the rule that one common ontology element o from an OWL ontology is used, as shown in equation 1.

$$\begin{aligned} \exists a, b, p_1, p_2, o, m_1, m_2, r_1, r_2, t_1, t_2 \\ \mathit{SocialNetworkActor}\,(a) \wedge \mathit{SocialNetworkActor}\,(b) \wedge \\ \mathit{ProcessElement}\,(p_1) \wedge \mathit{ProcessElement}\,(p_2) \wedge \\ \mathit{OWLElement}\,(o) \wedge \mathit{ModelReference}\,(m_1) \wedge \mathit{ModelReference}\,(m_2) \wedge \\ \mathit{OntologyReference}\,(r_1) \wedge \mathit{OntologyReference}\,(r_2) \wedge \\ \mathit{Annotator}\,(t_1) \wedge \mathit{Annotator}\,(t_2) \end{aligned} \tag{1}$$

The linkages between the elements can then be defined by *interref*, *isInput*, and *refersTo* predicates as shown in 2.

$$
\begin{aligned}
&\mathit{interref}\,(m_1,p_1) \wedge \mathit{interref}\,(m_2,p_2) \wedge \mathit{interref}\,(r_1,o) \wedge \\
&\mathit{interref}\,(r_2,o) \wedge \mathit{isInput}\,(m_1,t_1) \wedge \mathit{isInput}\,(m_2,t_2) \wedge \\
&\mathit{refersTo}\,(t_1,r_1) \wedge \mathit{refersTo}\,(t_2,r_2) \wedge \mathit{interref}\,(t_1,a) \wedge \mathit{interref}\,(t_2,b)
\end{aligned} \tag{2}
$$

Finally, we can refer to the connections between the two social actors via the *connectedWith* predicate and define different variants for consequents of the rule based on the antecedents defined in 1 and 2. In the first variant in equation 3 it is assumed that the actors were not connected before and are now being connected based on their annotation activity. In equation 4 it is shown how an existing connection between the two actors leads to their inclusion in the socical network group g for bringing together actors with similar interests based on their common usage of the ontology concept o for their annotations.

$$
(1) \wedge (2) \rightarrow \mathit{connectedWith}\,(a,b) \tag{3}
$$

$$
\begin{aligned}
(1) \wedge (2) \wedge \mathit{connectedWith}\,(a,b) \rightarrow \exists g, \mathit{SocialNetworkGroup}\,(g)\,, \\
\mathit{partOf}\,(a,g)\,, \mathit{partOf}\,(b,g)
\end{aligned} \tag{4}
$$

However, from a more practical perspective it would be desirable not to force actors in the social network into groups automatically or connect them with other actors. Rather, a less deterministic approach may be more appropriate that gradually increases the likelihood of the consequents to become effective. We show this formally in the following via the function *recommendConnection* that maps to a real number. The consequent of the rule is thus modified as shown in 6, where a modification factor β is added each time the rule fires. In this concrete case this means that whenever two actors refer to the same ontology concept in their annotations, the likelihood of connecting them is increased. Based on the attainment of a threshold value θ, as in equation 7, the actual connection is established in 8.

$$
\mathit{recommendConnection} : (a \times b) \rightarrow \; [0 \ldots 1] \tag{5}
$$

$$
(1) \wedge (2) \rightarrow (\mathit{recommendConnection}(a,b) + \beta) \tag{6}
$$

$$
\mathit{relevantConnection}\,(a,b) = \begin{cases} \mathit{true} \text{ if } \mathit{recommendConnection}(a,b) \geq \theta \\ \mathit{false} \text{ if } \mathit{recommendConnection}(a,b) < \theta \end{cases} \tag{7}
$$

$$
(1) \wedge (2) \wedge \mathit{relevantConnection}\,(a,b) \rightarrow \mathit{connectedWith}\,(a,b) \tag{8}
$$

4 Discussion

With the above descriptions we can now discuss the benefits and limitations of the chosen approach. In particular we will refer to three aspects that have been proposed in [16] for using social network information in the collaborative formalization of semantics. These are: *communication support*, *economic benefits*, and *technical opportunities*.

Regarding communication support, the major advantage of the approach is that it enables the *collaboration on annotation tasks* while at the same time preserving existing conceptual model and ontology handling interfaces. A user who does not want to engage in the annotations but is just interested in modeling tasks is not affected by the new annotation functionalities. By integrating the functionalities offered by the social networking websites, new possibilities for interaction emerge: Not only can it be shared with connections in the social network, which annotations an actor has conducted by making this information public on the social network. This opens the potential for integrating a multitude of users in the sense of crowd-sourcing for the annotation tasks, which leads to economic benefits in terms of *reduced effort* for annotations by a single user. In contrast to other annotation solutions it can be easily communicated and made visible to other users which annotation tasks have already been conducted and where additional information is required. Besides the information about the semantic annotation also the content of the models and the ontologies can be shared. At the same time of course privacy aspects may constitute a limitation that needs to be specifically considered.

The second characteristic of the proposed approach is that it allows to *personalize the user experience* in regard to semantic annotations. By accessing the social network information together with the information about the annotations, new ways for suggesting annotations to users can be designed. This has already been shown by the descriptions in section 3.2. It could be further extended by specifying a number of additional rules, e.g. to suggest connections based on commonly used social network resources, commonly used model references, ontology references or more detailed subsets of these elements such as elements of the information and control flow of business processes. Furthermore, additional information available in the social network that can be stored using the key/value pairs can be accessed. This concerns for example information about the work experience of actors or common professional interests.

From a technical side, many social network applications provide single-sign-on functionalities that can be easily re-used for implementing semantic annotation applications. Due to the central handling of the *user authentication*, all implementation effort that usually has to be devoted to creating separate authentication services can be omitted. This concerns also the workflows for password changes, retrieving forgotten passwords or changes in the user profile. Therefore, additional economic benefits may be reaped through less implementation effort. Nevertheless it may be additionally necessary to restrict the access to the information to certain groups of the social network. However, these group policies may be handled using the social network's functionalities, e.g. based on the administrative rights for the groups.

5 Conclusion and Outlook

With the presented approach we could show how information from social network applications can be integrated with the semantic annotation of conceptual models. Furthermore, based on the semi-formal definition of the according meta models, more formal specifications for the behavior could be added using a rule-based approach. The next steps will include the further implementation of the approach by adapting and extending existing tools for conceptual modeling and ontology handling. Based on such an implementation it can then be investigated which of the many options for combining social network information, conceptual models, ontologies and the annotations best meets the user requirements for conducting efficient annotations. For the future it is planned to integrate more information from social network applications in the domain of conceptual modeling, e.g. by analyzing statements from social network actors made in natural language and transferring them to the domain of conceptual models.

References

1. Becker, J., Delfmann, P., Herwig, S., Lis, L., Stein, A.: Towards Increased Comparability of Conceptual Models - Enforcing Naming Conventions through Domain Thesauri and Linguistic Grammars. In: ECIS 2009. Verona, Italy (2009)
2. Betz, S., Klink, S., Koschmider, A., Oberweis, A.: Automatic user support for business process modeling. In: Workshop Semantics for Business Process Management. Budva, Montenegro (2006)
3. Boyd, D., Ellison, N.: Social Network Sites: Definition, History, and Scholarship. Journal of Computer-Mediated Communication 13, 210–230 (2008)
4. Demirkan, H., Kauffman, R.J., Vayghan, J.A., Fill, H.G., Karagiannis, D., Maglio, P.: Service-oriented technology and management: Perspectives on research and practice for the coming decade. Electronic Commerce Research and Applications 7(4), 356–376 (2008)
5. Drumm, C., Lemcke, J., Namiri, K.: Integrating semantic web services and business process management: A real use case. In: Semantics for Business Process Management 2006. Budva, Montenegro (2006)
6. Ehrig, M., Koschmider, A., Oberweis, A.: Measuring Similarity between Semantic Business Process Models. In: Roddick, J., Hinze, A. (eds.) APCCM 2007, vol. 67, pp. 71–80. ACM (2007)
7. Facebook: Facebook Developers - Graph API (2011), http://developers.facebook.com/docs/reference/api/
8. Fill, H.G.: Design of Semantic Information Systems using a Model-based Approach. In: AAAI Spring Symposium. AAAI, Stanford University, CA (2009)
9. Fill, H.G.: Visualisation for Semantic Information Systems. Gabler (2009)
10. Fill, H.G.: An Approach for Analyzing the Effects of Risks on Business Processes Using Semantic Annotations. In: Proceedings of ECIS 2012. AIS (2012)
11. Fill, H.G.: SeMFIS: A Tool for Managing Semantic Conceptual Models. In: Kern, H., Tolvanen, J.P., Bottoni, P. (eds.) GMLD Workshop (2012)
12. Fill, H.G., Burzynski, P.: Integrating Ontology Models and Conceptual Models using a Meta Modeling Approach. In: 11th International Protégé Conference. Amsterdam (2009)

13. Fill, H.G., Karagiannis, D.: On the Conceptualisation of Modelling Methods Using the ADOxx Meta Modelling Platform. Enterprise Modelling and Information Systems Architectures 8(1), 4–25 (2013)
14. Fill, H.G., Redmond, T., Karagiannis, D.: FDMM: A Formalism for Describing ADOxx Meta Models and Models. In: Proceedings of ICEIS 2012, Wroclaw, Poland. vol. 3, pp. 133–144 (2012)
15. Fill, H.G., Reischl, I.: An Approach for Managing Clinical Trial Applications using Semantic Information Models. In: Rinderle-Ma, S., Sadiq, S., Leymann, F. (eds.) BPM Workshops 2009, pp. 581–592. Springer, Ulm, Germany (2009)
16. Fill, H.G., Tudorache, T.: On the Collaborative Formalization of Agile Semantics Using Social Network Applications. In: AAAI Spring Symposium 2011 - AI for Business Agility. AAAI (2011)
17. Harel, D., Rumpe, B.: Modeling Languages: Syntax, Semantics and All That Stuff - Part I: The Basic Stuff. Tech. Rep. MCS00-16, The Weizmann Institute of Science (August 22, 2000 2000)
18. Hepp, M., Leymann, F., Domingue, J., Wahler, A., Fensel, D.: Semantic business process management: a vision towards using semantic Web services for business process management. In: ICEBE Conference 2005. pp. 535–540 (2005)
19. Herbst, J., Junginger, S., Kuehn, H.: Simulation in Financial Services with the Business Process Management System ADONIS. In: 9th European Simulation Symposium (ESS97). Society for Computer Simulation (1997)
20. Horrocks, I., Patel-Schneider, P., Van Harmelen, F.: From SHIQ and RDF to OWL: The Making of a Web Ontology Language. Web Semantics: Science, Services and Agents on the World Wide Web 1(1), 7–26 (2003)
21. Hsieh, M.H., Magee, C.L.: A new method for finding hierarchical subgroups from networks. Social Networks 32, 234–244 (2010)
22. Ibanez, M., Alvarez, P., Bhiri, S., Ezpeleta, J.: Unary RDF-Annotated Petri Nets: A Formalism for the Modeling and Validation of Business Processes with Semantic Information. In: SBPM 2009 Workshop. ACM (2009)
23. Karagiannis, D., Kuehn, H.: Metamodeling Platforms. In: Bauknecht, K., Min Tjoa, A., Quirchmayer, G. (eds.) Third International Conference EC-Web 2002 Dexa 2002, p. 182. LNCS2455, Springer, Aix-en-Provence, France (2002)
24. Lewis, K., Kaufman, J., Gonzalez, M., Wimmer, A., Christakis, N.: Tastes, ties, and time: A new social network dataset using facebook.com. Social Networks 30, 330–342 (2008)
25. List, B., Korherr, B.: An evaluation of conceptual business process modelling languages. In: SAC'06. ACM, Dijon, France (2006)
26. Moser, C., Bayer, F.: IT Architecture Management: A Framework for IT Services. In: Desel, J., Frank, U. (eds.) EMISA Workshop. LNI Series, Klagenfurt, Austria (2005)
27. Mylopoulos, J.: Conceptual Modeling and Telos. In: Loucopoulos, P., Zicari, R. (eds.) Conceptual Modelling, Databases and CASE: An Integrated View of Information Systems Development, pp. 49–68. Wiley (1992)
28. Nazir, A., Raza, S., Chuah, C.N.: Unveiling facebook: A measurement study of social network based applications. In: IMC'08. ACM, Vouliagmeni, Greece (2008)
29. Obrst, L.: Ontologies for semantically interoperable systems. In: Proceedings of the 12th International Conference on Information and Knowledge Management. ACM Press, New Orleans (2003)
30. Wand, Y., Weber, R.: Research Commentary: Information Systems and Conceptual Modeling - A Research Agenda. Information Systems Research 13(4), 363–376 (2002)