

Quality Evaluation of a Business Process Semantic Annotations Approach

Yun Lin
Agresso
Oslo, Norway
yun.lin@agresso.no

John Krogstie
IDI, NTNU
Trondheim, Norway
krogstie@idi.ntnu.no

Abstract: Semantic annotation is an approach to achieve semantic interoperability of heterogeneous resources. An ontology-based semantic annotation framework has been proposed, that can enrich and reconcile semantic representations of process models. The approach has been implemented in a prototype annotation tool, Pro-SEAT, to facilitate the annotation process. Evaluation methods for ontology-based semantic annotation is a new research topic. We view the annotation framework as a modeling approach, and have chosen to adapt SEQUAL for the evaluation of the approach. SEQUAL has previously been used for the evaluation of modelling languages and approaches, including the evaluation of ontologies, process models, and requirements models and modeling languages. Using the semiotic quality categories in SEQUAL, an evaluation has been made at both the meta-model level (GPO and the PSAM specifications) and model level (the PSAM model instances and the Pro-SEAT tool).

Introduction

Business process models built as solutions for different enterprises are heterogeneous, using different process modeling languages, process context and intentions. It is difficult to share and reuse the knowledge within those business process models across organizations, if there is no common understanding of the meaning of the process modeling languages, the terms used in the model and the intentions behind the models. Such a problem is usually called semantic interoperability problem.

An ontology is a catalog of the types of things that are assumed to exist in a domain of interest D using a language L for the purpose of talking about D [SN00]. When it is in the form of an explicit representation of the conceptualization, it can be used as a reference semantics to provide common understanding of heterogeneous representations. The process of building the references from the heterogeneous representations to the ontology is ontology-based semantic annotation.

Ontology-based semantic annotation is usually considered as a technique to achieve semantic interoperability by introducing common understanding and standardization. Research and application of ontology have been promoted by Semantic Web technology. Most semantic annotation approaches are developed and evaluated on both unstructured and structured artifacts to improve semantic

interoperability (e.g. textual resources [KH01] [HSS03], and Web services [SMSV04] [SAWSDL07]). A semantic annotation framework for a type of semi-structure artifact - enterprise/business process models, was initiated and reported in previous work of the authors [LSHKS06][LS07]. The proposed approach includes a process annotation ontology (GPO), an annotation model (PSAM) and a tool (Pro-SEAT). The goal of the framework is to achieve interoperability in process knowledge management, i.e. for the involved stakeholders to easily understand the business processes, sharing process knowledge, analyzing business opportunity, reuse and consolidating business process, across organizations. In the framework, ontologies are modeled in OWL DL [OWL09] - a description logic based Web ontology language for Semantic Web. Intentionally, the model semantics annotated with OWL DL ontologies are machine-interpretable with OWL DL engine.

In this paper, we present a comprehensive evaluation of the approach. We view the annotation framework as a modeling approach, and apply a generic quality framework for models and modeling languages - SEQUAL to provide a systematic analysis on the quality of GPO (General Process Ontology), PSAM (Process Semantic Annotation Model) and the annotation tool Pro-SEAT (Process SEMantic Annotation Tool). The quality analysis is based on the usage experiences from an exemplar study, in which the GPO is taken as a modeling language, the PSAM model of the exemplar as a model instance of GPO, and Pro-SEAT as modeling editor tool.

The rest of the paper is organized as follows. First, the semantic annotation framework for process models is described in more detail. Then, the exemplar study is described. Third, SEQUAL is presented. Applying SEQUAL, the quality analysis of the ontology-based semantic annotation framework is then elaborated based on the experience of use. Finally, conclusions are drawn and future work is outlined based on the evaluation results.

Framework for Semantic Annotations of Business Process

Models

The framework aims to support the semantic reconciliation of business process models. The approach taken is (1) to express the process properties of each business process model in a common annotation system, (2) to express model context in ontologies that may be compared to each other, and (3) to map the intentions of the systems' owners to goal structures that may be compared to each other. Therefore, four main annotation sets constitute the framework: namely, profile annotation, metamodel annotation, model annotation and goal annotation. In the profile annotation, a set of metadata specify the significant characteristics of the process models. In the meta-model annotation and the model annotation, we use ontologies to relate constructs across different modeling languages, as well as to align domain specific terminology used in models. Furthermore, the goal annotation [LS07] is to specify the capacities of process models using a goal ontology.

A GPO (General Process Ontology) [LSHKS06] [L08] has earlier been described. GPO is used for annotating the process modeling languages in the meta model annotation. For meta-model annotations, a process ontology is an explicit

and formal specification of concepts which are used to model processes in general. The main concepts in GPO are **Activity(AV)**, **Artifact(AF)**, **Actor-role(AR)**, **Input(I)**, **Output(O)**, **Precondition(Θ^{pre})**, **Postcondition(Θ^{pos})**, **Exception(E)** and **WorkflowPattern(WP)**. GPO consisting of those concepts and their relationships is represented using a UML class diagram in Figure 1.

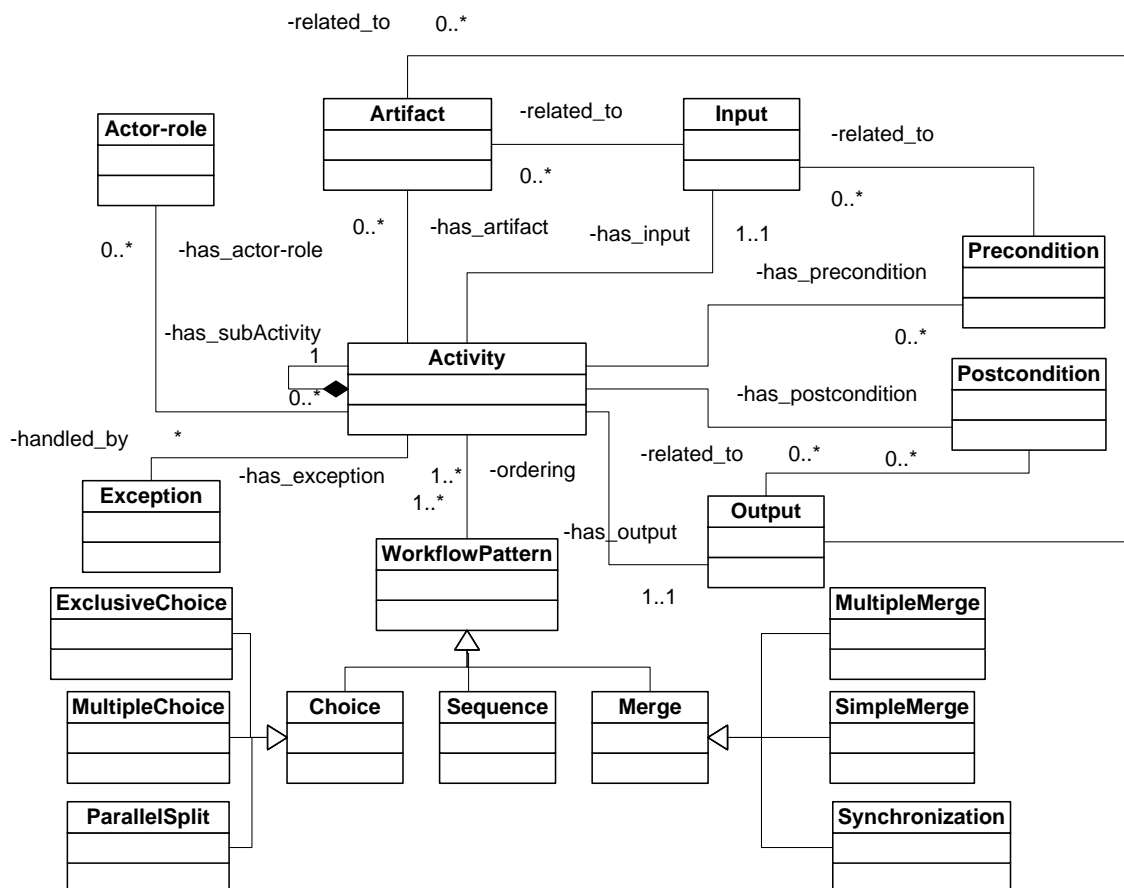


Figure 1: General Process Ontology [L08]

GPO builds the basis for the annotation model - PSAM (Process Semantic Annotation Model), which is formalized with a set of mapping strategies and rules between model objects and ontologies [LSHKS06]. PSAM provides a common semantic annotation schema for annotating semi-structured process models.

OWL DL is chosen for modeling ontology in this framework. Building upon RDF and RDFS, OWL DL provides machine-interpretable semantics. As a meta-model ontology, GPO categorizes modeling element types in the domain of process modeling (*D*) using OWL DL (*L*). A process model annotated by GPO is transformed in a PSAM model, which is in terms of the GPO ontology. PSAM models therefore comply with the syntax rules for OWL DL ontology instances, which makes the annotated process semantic representations machine-interpretable.

The PSAM contains the concepts of GPO and domain specific ontology including domain ontology (*PD*) and goal ontology (*PG*):

$$PSAM = (AV, AR, AF, WP, I, O, \Theta^{pre}, \Theta^{pos}, E, PD, PG)$$

Each element (AV, AR, AF, WP, I, O, Θ^{pre} and Θ^{pos}) in the GPO will be annotated with profile information. Specifically, an annotated activity (AV_i) in the PSAM is modelled as:

$AV_i = (id, model_fragment, name, alternative_name, has_Actor_role, has_Artifact, has_Input, has_Output, is_in_WorkflowPattern_of, has_Precondition, has_Postcondition, has_Exception, has_subActivity, same_as, different_from, kind_of, superConcept_of, phase_of, compositionConcept_of, instance_of, achieves|positively_satisfies|negatively_satisfies)$

Each element in the annotation model has an *id* and *name* to uniquely identify the element. *Model_fragment* is the identifier of model fragment in the original process model for keeping the link between the annotated model fragment and its annotation information. *Alternative_name* provides a synonym of the name at the terminology level. Elements *has_Actor-role*, *has_Artifact*, *has_Input*, *has_Output*, *is_in_WorkflowPattern_of*, *has_Precondition*, *has_Postcondition*, *has_Exception*, *has_subActivity* denote the relationships between the activity and other related elements according to the GPO definition. The *ids* of the related elements are used in those relationships. We use *same_as*, *different_from*, *kind_of*, *superConcept_of*, *phase_of*, *compositionConcept_of* to annotate the activities with the domain ontology concepts, i.e. using semantic relationship to map an activity to a concept in the domain ontology. *instance_of* is to specify that the modelled activity is an instance of the domain ontology class. The goal ontology is referenced through the annotation keywords *achieves*, *positively_satisfies* and *negatively_satisfies*.

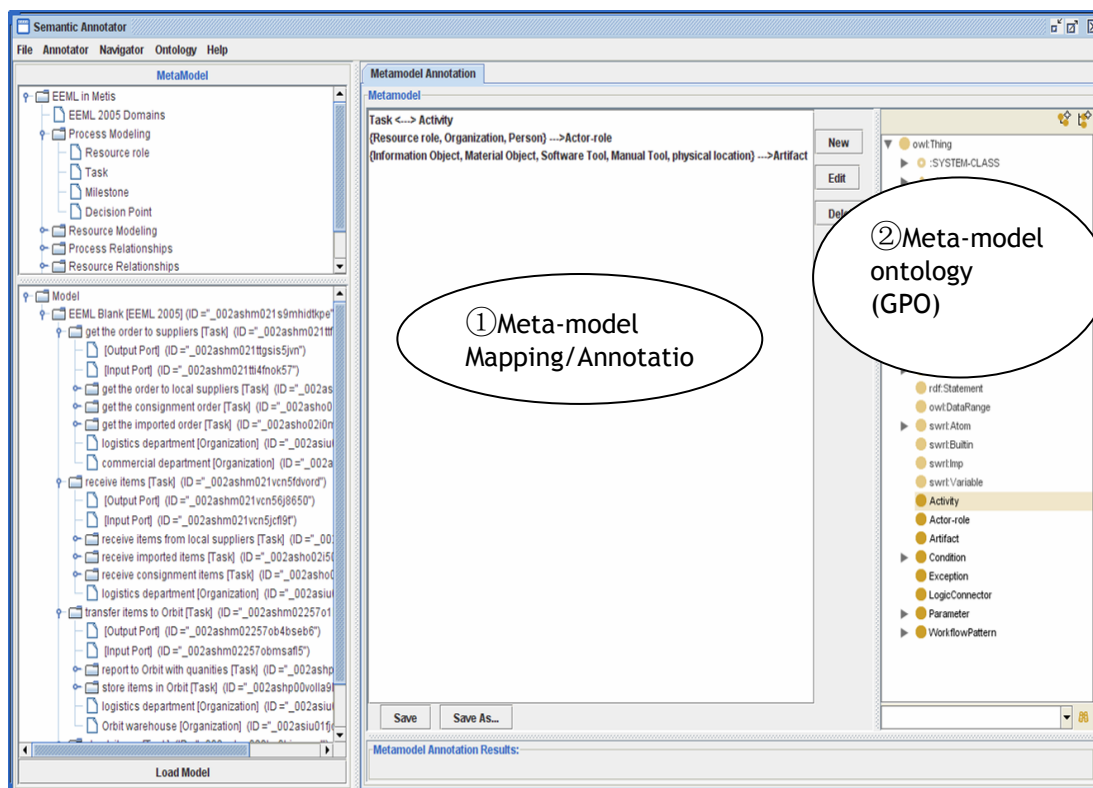


Figure2: Meta-model Annotation in Pro-SEAT

Based on the framework, Pro-SEAT is developed as an annotation tool independent of modelling tools. The annotation tool reads models created by modelling tools. In this work, we take Metis¹ as our modelling environment. Metis supports different modelling languages, such as UML, EEML [K08] and BPMN [BPMN04] through its powerful meta-model Developer. Meta-models and models created by Metis are stored in XML. The annotation tool therefore contains functions to parse and read the XML-representation of Metis-models.

Pro-SEAT can read process models and OWL ontologies. The main task of the tool is to apply the annotation framework to build relationships between models and ontologies. As an output of the system, the annotation result is stored in an OWL instance file, separately from the original process model. Figure 2 and 3 display the meta-model and model annotation UIs of Pro-SEAT.

The annotation tool reads the original Metis meta-model and model files, and displays the Metis meta-model and model tree. The loading of a meta-model ontology, e.g. GPO is illustrated in 2 in Figure 2. The meta-model annotation is achieved through building the mappings between meta-model elements and meta-model ontology concepts (1 in Figure 2). The mappings are saved in a meta-model annotation file in the annotation repository.

Based on the meta-model annotation, the model construct can be represented using the meta-model ontology concepts (1 in Figure 3). The content in each meta-model annotated model construct can be annotated with parts of the domain ontology. In the case presented below, the domain ontology is from SCOR represented in OWL (3 in Figure 3). 2 in Figure 3 shows the model annotation schema.

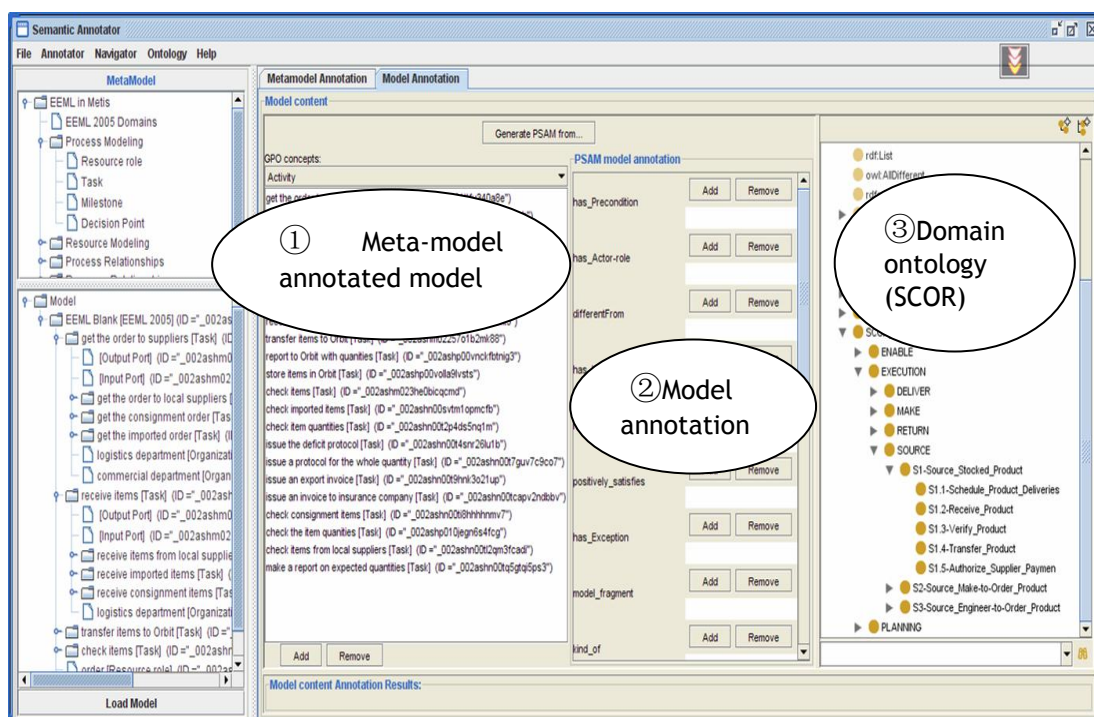


Figure3: Model Annotation in Pro-SEAT

¹ Metis is a commercial enterprise modeling tool, which is now named as Troux Architect™.

Exemplar case study

In order to evaluate the framework, an exemplar study has been the basis for examining the user experience on this approach and tool. In the study, we have process models that describe a same business domain(logistics process), but are modeled in different modeling languages and by different enterprises. We need domain and goal ontologies about the business domain for the annotation. Since there are no formal logistics ontology available, we formalized the SCOR (Supply Chain Operations Reference-model) [SCOR07] specifications into logistics domain and goal ontologies using OWL DL. I.e. the SCOR ontology provides a catalog of the types of processes in the logistics domain (D) using OWL DL (L).

In the case PM_A is from enterprise A and it is a purchase order process made in BPMN, whilst two other models PM_{B1} (an item receiving process) and PM_{B2} (an item delivery process) from the logistics department in enterprise B are built in EEML. Due to different business focus and modeling languages, the models depict different details and perspectives of the logistic process. For example, the delivery process models for enterprise B (Figure 4) are relatively simpler compared with the delivery processing in enterprise A (Figure 5 and Figure 6). However, both of them are expected to achieve the same goals.

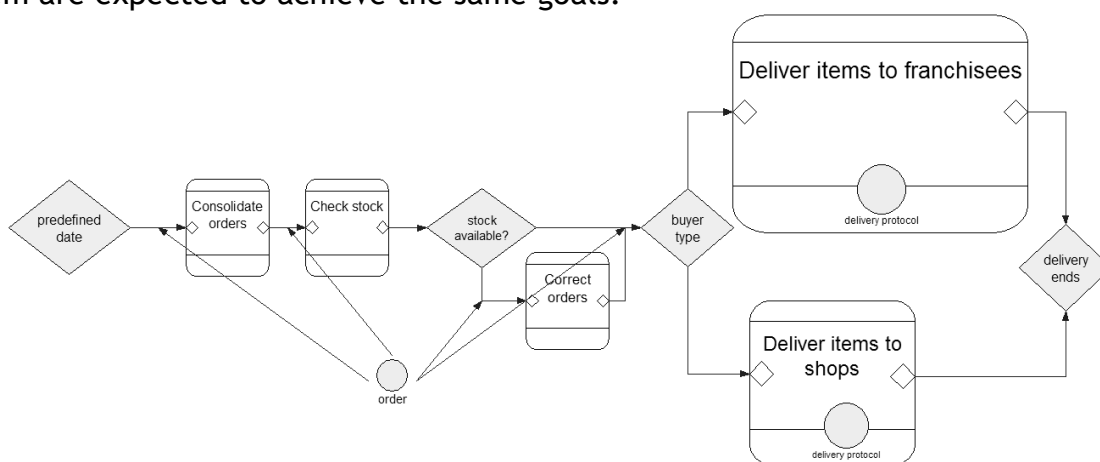


Figure 4: The item delivery process of enterprise B in EEML

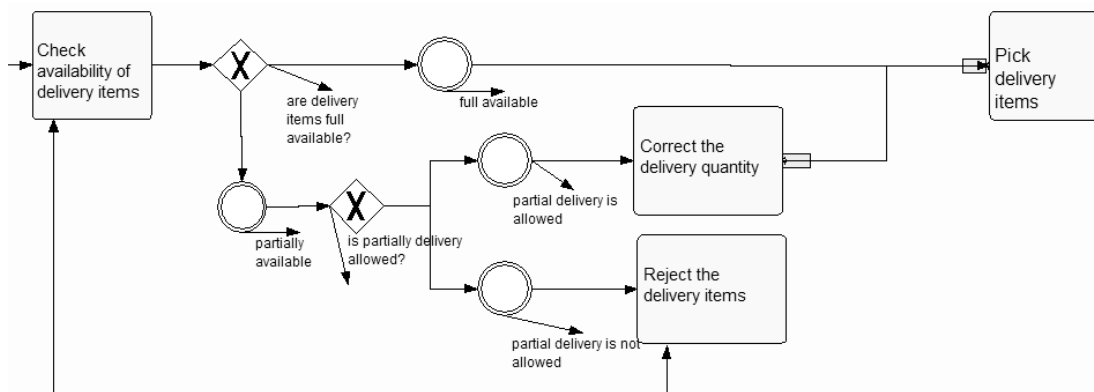


Figure 5: Checking availability of the delivery of enterprise A in BPMN

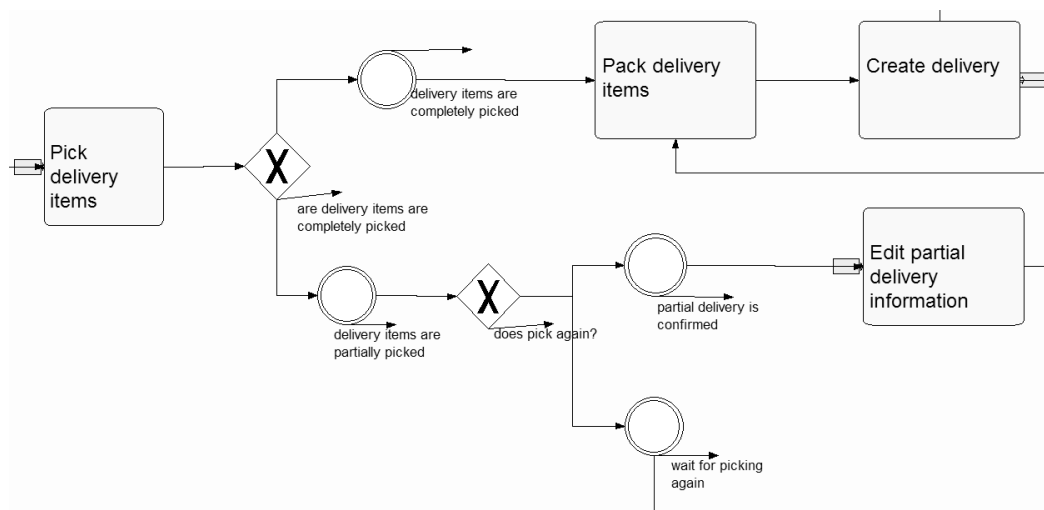


Figure 6: Picking, packing and creating delivery of enterprise A in BPMN

The three process models were modelled in Metis and imported into Pro-SEAT. The SCOR ontology and GPO-model are first edited in an OWL ontology editor - Protégé², and then also loaded into Pro-SEAT. By using the mapping functions provided by Pro-SEAT, the process modelling language elements and model fragments are annotated with the concepts in the ontologies.

In the case study, the meta-model annotation of EEML and BPMN elements with GPO concepts are listed in Table 1.

Table 1. Meta-annotation of the EEML and BPMN with the GPO concepts

GPO concepts	EEML elements	BPMN elements
Activity	Task	Logical Process
Artifact	Organization, Person	Swimlane including Horizontal Swimlane and Vertical Swimlane
Actor-role	Resource, Role, Information Object, Software Tool, Manual Tool, Material Tool, Location	Data Object, property 'Data' in Event
Input	Input Port	Input
Output	OutputPort	Output
WorkflowPattern	Combination of Flow, Milestone and Decision Point	Combination of Event (including Start Event, Intermediate Event, End Event), Sequence Flow, Gateway
Precondition	Milestone, Decision Point	Event (including Start Event, Intermediate Event), Gateway
Postcondition	Milestone, Decision Point	Event (including End Event, Intermediate Event), Gateway
Exception	Decision Point	Event (when event type is error) together with Sequence Flow

The model and goal annotation results of the three models are exemplified in Table 2, 3 and 4.

² <http://protege.stanford.edu/>

Table 2. Part of semantic annotation results of PM_A

PSAM Activity Instance	Model Annotation with SCOR Domain Ontology	Goal Annotation Relations	SCOR Goal Ontology
Check delivery items	<i>phase_of</i> D1.3 Reserve Inventory and Determine Delivery Date	<i>positively_satisfies</i>	Improve Deliver Performance; Raise Order Quantity Fillrate
Check availability of delivery	<i>phase_of</i> ED.3 Manage Deliver Information	<i>positively_satisfies</i>	Ensure Full Delivery; Improve Deliver to Customer On Time Delivery Performance; Improve Deliver to Customer Delivery to Date Performance
Create delivery	<i>kind_of</i> ED.3 Manage Deliver Information	<i>Achieves</i>	Delivery is Scheduled; Delivery Terms are Generated
		<i>positively_satisfies</i>	Ensure Full Shipment; Ensure Full Delivery

Table 3. Part of semantic annotation results of PM_{B1}

PSAM Activity Instance	Model Annotation with SCOR Domain Ontology	Goal Annotation Relations	SCOR Goal Ontology
Get the order to suppliers	<i>phase_of</i> S1.1 Schedule Product Deliveries	<i>Achieves</i>	Sourced Product are On Order; Order is Placed
		<i>positively_satisfies</i>	Reduce Order Processing Time; Reduce Order Receipt Time
Check items from local suppliers	<i>kind_of</i> S1.3 Verify Product	<i>Achieves</i>	Sourced Product are Verified
		<i>positively_satisfies</i>	Reduce Order Receipt Time
Issue the deficit protocol	<i>phase_of</i> ES.9 Manage Supplier Agreements	<i>Achieves</i>	Procurement Notification to Supplier
		<i>negatively_satisfies</i>	Reduce Order Processing Time; Improve Supplier On Time Delivery Performance; Reduce Order Processing Costs; Reduce Order Receipt Time

Table 4. Part of semantic annotation results of PM_{B2}

PSAM Activity Instance	Model Annotation with SCOR Domain Ontology	Goal Annotation Relations	SCOR Goal Ontology
Consolidate orders	<i>same_as</i> D1.4 Consolidate Orders	Achieves	Order is Processed; Order is Consolidated
		<i>positively_satisfies</i>	Reduce Order Processing Costs; Reduce Order Processing Time
Check stock	<i>phase_of</i> D1.3 Reserve Inventory and Determine Delivery Date	<i>Achieves</i>	Order is Validated
		<i>positively_satisfies</i>	Raise Order Quantity Fillrate
Generate delivery protocol	<i>kind_of</i> ED.3 Manage Deliver Information	<i>Achieves</i>	Delivery Terms are Generated
		<i>positively_satisfies</i>	Improve Deliver to Customer Delivery to Date Performance; Improve Deliver to Customer On Time Delivery Performance

Evaluation

Our evaluation of the method and prototype implementation consisted of two parts: 1) analytical quality analysis of semantic annotation framework and tool, and 2) empirical applicability validation based on annotation results. The analytical quality analysis uses a quality framework --- SEQUAL [K08] to provide a systematic analysis on the quality of GPO (General Process Ontology), PSAM (Process Semantic Annotation Model) and the annotation tool Pro-SEAT. We will in this paper present the analytical evaluation. The details of the empirical evaluation can be found in [L08].

SEQUAL

SEQUAL has been used for the evaluation of models, modelling tools and modelling languages in a number of areas, such as ontologies [LSHD04], business process and workflow models [CKSL97,NK06,RRK07], enterprise models [KA04,KDJ05], object-oriented models [K03], goal-models [K08], requirements models [K01] and interactive models [KSJ06]. The main concepts (sets) of the SEQUAL-framework and their relationships are shown in Figure 7. Quality has been defined referring to the correspondence between statements belonging to the following sets:

- G, the goals of the modelling task.
- L, the language extension, i.e., the set of all statements that are possible to make according to the graphemes, vocabulary, and syntax of the modelling languages used.

- D, the domain, i.e., the set of all statements that can be stated about the situation at hand.
- M, the model itself.
- Ks, the relevant explicit knowledge of those being involved in modelling.
- I, the social actor interpretation, i.e., the set of all statements that the audience thinks that the model consists of.
- T, the technical actor interpretation, i.e., the statements in the model as 'interpreted' by modelling tools.

The main model quality types are indicated by solid lines between the sets, and are described briefly below:

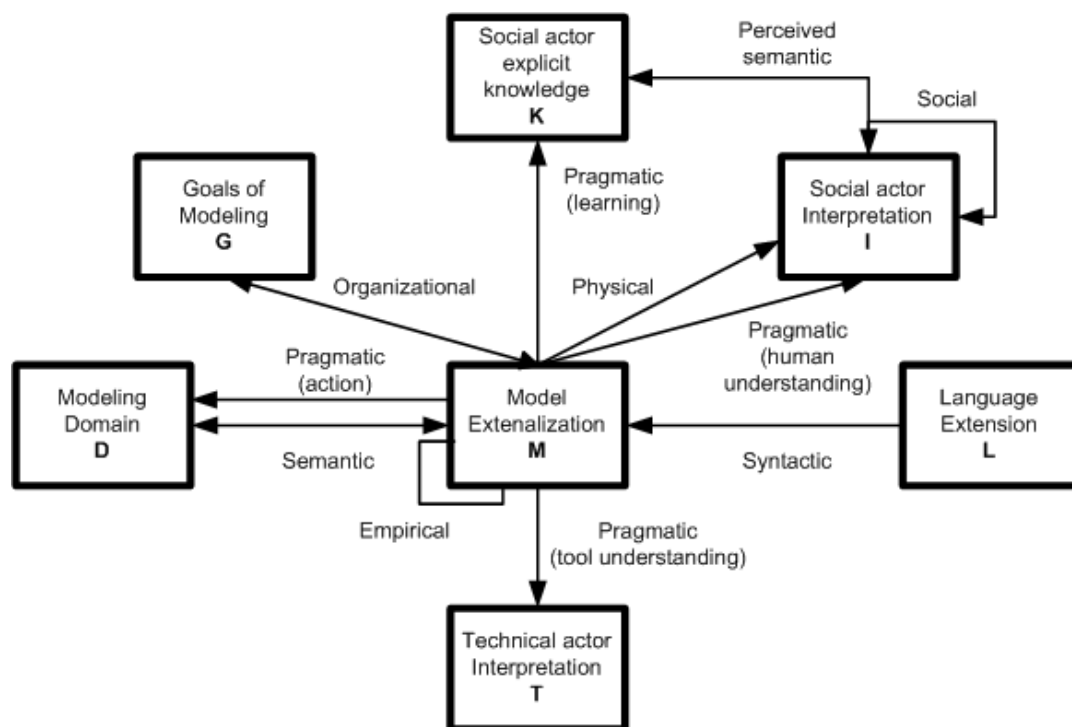


Figure 7: SEQUAL: Framework for discussing the quality of models

- Physical quality: The basic quality goal is that the model M is available for the audience.
- Empirical quality deals with predictable error frequencies when a model is read or written by different users, coding (e.g. shapes of boxes) and HCI-ergonomics for documentation and modelling-tools. For instance, graph layout to avoid crossing lines in a visual model is a mean to address the empirical quality of a model.
- Syntactic quality is the correspondence between the model M and the language extension L.
- Semantic quality is the correspondence between the model M and the domain D. This includes validity and completeness.
- Perceived semantic quality is the correspondence between the audience interpretation I of a model M and his or hers current knowledge K of the domain D.
- Pragmatic quality is the correspondence between the model M and the audience's interpretation and application of it (I). We differentiate be-

tween social pragmatic quality (to what extent people understand and are able to use the models) and technical pragmatic quality (to what extent tools can be made that interpret the models).

- The goal defined for social quality is agreement among audience members' interpretations I.
- The organizational quality of the model relates to that all statements in the model contribute to fulfilling the goals of modelling (organizational goal validity), and that all the goals of modelling are addressed through the model (organizational goal completeness).

Language quality relates the modelling language used to the other sets. Six areas for language quality are identified

- Domain appropriateness. This relates the language and the domain. Ideally, the language should be able to express anything in the domain, not having what is termed construct deficit. On the other hand, you should not be able to express things that are not in the domain, i.e. what is termed construct excess. Domain appropriateness is primarily a mean to achieve semantic quality.
- Participant appropriateness relates the social actors' explicit knowledge to the language. Participant appropriateness is primarily a mean to achieve pragmatic quality both for comprehension, learning and action.
- Modeler appropriateness: This area relates the language to the participant knowledge. The goal is that there are no statements in the explicit knowledge of the modeler that cannot be expressed in the language. Modeler appropriateness is primarily a mean to achieve semantic quality.
- Comprehensibility appropriateness relates the language to the social actor interpretation. The goal is that the participants in the modelling effort using the language understand all the possible statements of the language. Comprehensibility appropriateness is primarily a mean to achieve empirical and pragmatic quality.
- Tool appropriateness relates the language to the technical audience interpretations. For tool interpretation, it is especially important that the language lend itself to automatic reasoning. This requires formality (i.e. both formal syntax and semantics being operational and/or logical), but formality is not necessarily enough, since the reasoning must also be efficient to be of practical use. Different aspects of tool appropriateness are means to achieve syntactic, semantic and pragmatic quality (through formal syntax, mathematical semantics, and operational semantics respectively).
- Organizational appropriateness relates the language to standards and other organizational needs within the organizational context of modelling. These are means to support organizational quality.

SEQUAL based evaluation of the semantic annotation framework

In order to apply SEQUAL, we first specify what corresponds to the sets in the quality framework.

Model (M)

The annotation process and the annotation result are evaluation targets. In our approach, the annotation result is an instance of the annotation model. Therefore, the PSAM model is the model that is evaluated using the quality framework.

Goals (G)

The annotation is to represent the knowledge stored in the existing process models through a set of agreed semantically-defined concepts and formats. Therefore, the goals of annotation depend on the original modelling goals in each case and also depend on the goals of knowledge management. A number of goals are identified from the cases as follows.

- G1 - The annotation should improve the readability of the existing process models.
- G2 - The annotation should help sharing process knowledge among different organizations within a domain.
- G3 - The annotation should help to analyze and validate the existing process models.
- G4 - The annotation should be helpful in the semantic reconciliation of models and to facilitate reuse and integration of models.

Domain (D)

In general, the modelling domain is related to processes in the SCO (Supply-Chain-Operation) domain.

Language (L)

GPO is the meta-model of PSAM and determines the definitions of PSAM. Thus, GPO defines the syntax of the annotation model. Since GPO is created in OWL DL, a PSAM model is the instance of the OWL DL model and it has the syntactical features and constraints of OWL DL.

Modeler (K)

Here, the model annotators are the modellers. They create the annotation by applying their modelling and domain knowledge. In the exemplar studies, we assume that the annotators are familiar with the modelling languages and models.

Participant (I)

Annotation users are the consumers of the annotation results. In the case, they make use of the annotation information in the process knowledge management

activities, such as querying information, analyzing models, and eliciting/infering interested knowledge.

Tool (T)

The annotation tool -- Pro-SEAT -- provides the functions for profile annotation, meta-model annotation, model annotation and goal annotation.

Quality analysis

The overall quality evaluation includes the quality of GPO, the PSAM definitions, the annotated PSAM model instances and the annotation tool for the cases. GPO is defined as the meta-model of the annotation model, and the PSAM definitions are applied as the notation of the modelling. Thus, the language quality is analyzed on GPO and the PSAM definitions. The model quality is discussed based on the models in the exemplar studies.

Language Quality

Domain appropriateness. GPO has been defined by looking at central process modelling languages in enterprise modelling, and GPO includes the most vital and frequently used concepts in describing a business process. Compared with those languages, the GPO concepts are more general. Accordingly, specific semantics in the domain can only be abstracted or encapsulated into a set in GPO. Such a quality evaluation can be made through the analysis of the meta-model annotation in the exemplars. The relations linking the GPO concepts and the ontology concepts are specified in a PSAM model as well. All in all, GPO and PSAM have proper domain appropriateness.

Participant appropriateness.

GPO is relatively simple compared with most enterprise process modelling languages. Since we assume that the annotators are process modelling specialists, learning GPO should not be a problem. In the profile, model and the goal annotation, we also assume the annotators are domain experts and understand the original models quite well. The only additional requirement is to have some knowledge about ontology and semantic relationships when using the ontology-based annotation.

Modeler appropriateness.

Since the GPO concepts are more general than a particular enterprise process modelling language, some specific semantics in the original models could not be conveyed in the PSAM model. However, GPO is not initially created as a comprehensive process modelling language. The intention of the proposal is to elicit the most important process knowledge and to represent it in annotation models for knowledge management. In the meta-model annotation phase, a

possible mapping from GPO to a particular modelling language has only three types --- one-to-one, one-to-many, and one-to-combination, which means that the modelling language elements can be mapped to only one GPO concept. In the current prototype implementation many-to-one mappings are not supported. After the transformation based on the meta-model annotation, all the PSAM models have the same structure. Therefore the modeler can not make any other creative structure.

Comprehensibility appropriateness.

29 concepts are found in the current version of GPO and the PSAM specification. The concepts and the relationships between them are relatively straightforward for annotation users to interpret just by reading the names. With respect to the relations linking the GPO concepts and the ontology concepts, only three semantic relationship categories (synonym, hypernym, and meronym) from PSAM are used for in the exemplar studies. Also the name of those semantic relationships are specified according to the GPO concepts. For example, the meronym is represented by "phase of" for *Activity*, "part of" for *Artifact* and "member of" for *Actor-role*.

Tool appropriateness.

PSAM is defined using a formal syntax and modeled in OWL. OWL is XML-syntactic compilable and it can be parsed by available parsers. The semantics of GPO and PSAM are also formally defined according to the OWL DL semantic definitions. OWL DL was designed to support the existing Description Logic business segment and has desirable computational properties for reasoning systems. The Protégé-OWL API is integrated in the annotation tool so that the GPO and PSAM models can be interpreted by the tool.

Organizational appropriateness.

The approach is based on standards such as OWL, and is reasonably easy to integrate with existing approaches and tools

Model Quality

A specific annotation model is an instance of the PSAM model that is transformed from the original process models and annotated with ontological concepts. The quality of GPO and PSAM will impact the quality of the PSAM models. The evaluation of the PSAM models concludes how the model quality relates to the usability of the annotation results.

Physical Quality

We first look at how the knowledge of the domain has been externalized by the annotation models. As we have discussed in the quality analysis of domain appropriateness above, the PSAM models can present most information about the process and functional perspective. The EEML models have presented the logistics processing comprehensively. Based on the meta-model annotation results, the original models are transformed into the PSAM models. Ideally, the transformation should keep exactly the same information as the original models. It is obvious the more one-to-one mappings in the meta-model annotation, more knowledge represented in the modelling elements can be preserved after the transformation.

In the exemplar study, three one-to-one mappings are in the annotation of EEML. In EEML, different resources are specified and they all could be mapped to the GPO concepts Artifact and Actor-role respectively. Since the specific resources are not much applied in the original EEML models, not much domain knowledge is lost because of the meta-model annotation in this case. However, annotating the relationships defined in the modelling languages is not supported by the approach. The relationships between the GPO concepts are defined in GPO so that knowledge in the PSAM models are interpreted according to the GPO definitions. Compared with the original models, the PSAM models have additional knowledge --- the ontological knowledge, which is introduced during the model annotation and the goal annotation. In the exemplar study, the domain ontology is related to the SCOR standard, while the goal ontology provides the knowledge about the objectives of the process and also the links to the SCOR process metrics.

The PSAM models are then checked to determine if they are easily available and maintainable. The meta-model annotation result for each modelling language is stored in an XML file. Such results can be reused in generating the PSAM models by different models in the same modelling language. The generated PSAM models are OWL models, so that the PSAM models can be read and edited by any OWL editing tools. Model annotation and goal annotation is then made to the PSAM models. However, if there is any change of the meta-model annotation, the PSAM models based on the meta-model annotation result have to be re-generated. That means all the model and the goal annotation work made on the PSAM models will be lost.

Empirical Quality

There is no graphic notation for the annotation model. All the generated PSAM models are textual OWL files so that the readability of the model is poor without tool support. The PSAM models are categories according to the GPO concepts. Since no graphic notation is used for the GPO concepts, the concepts such as "Activity", "Artifact", "Actor-role" are not shown graphically differently to the user.

Syntactic Quality

As an OWL model with classes and instances, the PSAM models should comply with the OWL syntax. Since the PSAM models are generated from the meta-model annotation result, the premise is that the syntax of GPO and the mappings have been validated. GPO is created by Protégé and the correctness of the syntax is checked by Protégé. The meta-model mapping rules have also been set to comply with OWL syntax.

Semantic Quality

The semantic quality of the annotation models depends on the semantic quality of both the original models and the annotation. We have assumed that the original process models are semantically correct and complete. The semantics of the generated PSAM models are consequently determined by the transformation from the original models to the PSAM definitions.

More semantics are introduced during the model and the goal annotation. The quality of such semantics is categorized into the perceived semantic quality in this approach because it corresponds to annotators' and annotation users' interpretations and their knowledge of the domain. Most annotation operations are

manual, but Pro-SEAT supports the semi-automatic goal annotation which might help to achieve semantic validity and completeness. In Pro-SEAT, the ontology-based query interface provides the service to perceive the semantics of the annotation. The perceived semantic quality of the annotation model is further validated through the applications and analyzed in the empirical analysis [L08].

Pragmatic Quality

There is a limited number of categories since only eight GPO concepts are used in the exemplar studies. However, when the original model is large, the list of the instances for each GPO concept is long. From the experience of using Pro-SEAT, it turns out that it is more difficult to find a desired instance in the PSAM model in larger models. The instance is named with the model textual title and its model id from the original process models. When two instances have the same model title but different model id, mistakes can be made by picking the wrong instance because of a confusion with the title.

Since the annotated process models are OWL model instances, it is not difficult for an annotation user to understand the annotation schema and structure. Moreover, the ontology is designed in a human understandable way and the annotation user is supposed to know about the domain. There is no problem for the annotation user to read the models, but it is hard for the user to see the whole picture of the models without the support of a visualization tool. However, the OWL models can be interpreted by any tools supporting OWL DL. Since the SCOR ontology provides explicit representation of conceptualization of the supply chain domain, the references in the annotation help the annotation users to learn the domain and adapt processes, which was evident in the applicability validation. Besides, the pragmatic quality of models in Pro-SEAT also depends on the techniques to support the development of pragmatic quality in Pro-SEAT, which is discussed below.

Social Quality

One of the goals for the proposed approach is to help process knowledge sharing among different organizations within a domain. The ontologies are assumed to be the domain standards which are agreed by the different audiences. GPO is the meta-model ontology. The PSAM models are generated from this standard and the model content is annotated with domain standard terms.

Organizational Quality

For the organizational quality of the PSAM models and the tool, we can check if the modelling goals can be fulfilled and addressed by the approach.

G1 - The annotation should improve the readability and comprehensibility of the existing process models.

The semantic annotation enriches the model semantics by referencing the ontology concepts using semantic relationships. Thus, with the referenced ontology, the semantics of the model elements can be interpreted more correctly and completely by both human and machine. With respect to the pragmatic quality of Pro-SEAT we find that the annotation functions do not really address this goal. However the ontology-based query in Pro-SEAT provides the functions to navigate the process

models. Since the annotation results are in OWL, a tool can read and reason on the semantics in the annotation model.

G2 - The annotation should help process knowledge sharing among different organizations within a domain.

From the discussion on social quality, we find that this goal has the potential to be addressed by the annotation models, but further investigations are needed to evaluate this further. The applicability evaluation of the approach supports that the annotation models can fulfill this goal.

G3 - The annotation should help to analyze and validate the existing process models.

The empirical evaluation executed on the exemplar study exemplified that the annotation results can be used for semantic validation for process models. In annotation of workflow pattern for the EEML model, the logic connection "join" or "choice" is modelled by an EEML Milestone or Decision Point, and the annotated workflow branches from this connection share a same GPO Condition ("eeml:milestone" is mapped to "gpo:condition" in the meta-model annotation). Two evaluation results are therefore brought out: a) The EEML model should be improved by changing the way of modelling to specify the semantics of different conditions (this possibility is present in the EEML language). b) In the annotation model, conditions for different branches should be separate.

G4 - The annotation should be helpful in model reuse and model integration.

This goal is related to G2. The applicability of the annotation results has been described in the application scenario under the exemplar study [L08].

Tool Quality

The annotation tool is also the means to achieve quality annotation results. Thus the tool evaluation is to see to what extent Pro-SEAT includes functionality to support the achievement of model quality.

Physical Quality - Pro-SEAT provides the functions of importing the original process models, loading the ontology, editing and saving the meta-model annotation, the model and the goal annotation results. The current version of the tool is mainly designed for interpreting the models in XML created by the Metis tool. The tool can load the ontology in OWL format and the integrated ontology API is the Protégé 3.2.1 OWL API. There is no existing database repository for the annotated process knowledge, but the XML or OWL files and folders are used to manage the knowledge. Therefore, the benefit is that the knowledge is easy to distribute and could be published on the Web, while the disadvantage is the lack of a systematic way to manage the files and their inherent links.

Empirical Quality - The details of model annotation and goal annotation are displayed in the property fields for each instance in Pro-SEAT. The layout of those properties are not well organized in groups and the sequence of the properties is displayed differently each time when running the tool. Thus, it is hard to navigate the properties when manipulating the annotation. Nevertheless, the operation of the annotation is simple --- just select the reference concept from the ontology tree by clicking or entering the reference value.

Syntactic Quality - The prototype of the annotation tool does not provide the functions for performing a syntax check from the user interface, but invalid OWL models can not be parsed by the OWL API in Pro-SEAT.

Semantic Quality and Perceived Semantic Quality - The current annotation tool does not support any semantic consistency or completeness checking during the annotation.

Pragmatic Quality - Pro-SEAT does not provide a visualization of process models, annotation models and ontologies. The imported original process model from Metis is listed in the tree-view. The tree-view of the model keeps the same structure as in the Metis tool. It is not difficult for the Metis user to interpret the model in Pro-SEAT. The Protégé's tree-view of ontology is also integrated in Pro-SEAT so that the subsumption relationship between ontology concepts can be viewed in Pro-SEAT. The other relationships represented in OWL properties and constraints can not be displayed in Pro-SEAT, which would hide the complexity and help the annotator to identify the concepts easily. The annotation models are listed and navigated according to the GPO concepts. In Pro-SEAT, it is difficult to establish an overall view of the annotation model or check the inter-relationships between annotations. A model search interface is integrated in Pro-SEAT. Based on the annotation results, it facilitates the navigation of the process models and the annotation information by focusing on the GPO categories.

Conclusion and Further Work

Semantic annotation is an approach to achieve semantic interoperability of heterogeneous resources. However, such an approach has usually been applied to enhance the semantics of unstructured and structured artifacts (e.g. textual resources, and Web services). The semantic annotation framework for business process models introduce an ontology-based semantic annotation approach to enrich and reconcile semantics of process models --- a kind of semi-structured artifact, for managing process knowledge. The proposed approach has been implemented in a prototype annotation tool --- Pro-SEAT to facilitate the annotation process. The annotation approach and the prototype tool are evaluated using a comprehensive quality framework, SEQUAL.

Evaluation method for ontology-based semantic annotation is a new research topic. According to the author's knowledge, there is no systematic evaluation methodology for semantic annotation approaches and tools. Maynard in [M05] identified some requirements for ontology-based annotation tools such as expected functionality, interoperability, usability, accessibility, scalability and reusability.

Usually, criteria and metrics for performance evaluation, such as precision, recall and F-measure [R79], are defined for the evaluation of semi-automatic or automatic semantic annotations by using information extraction techniques. However, the evaluation is mainly for the semantic annotation of textual contents. The model features are certainly not covered in this approach, but they are very important in our case of the semantic annotation of business process models. Moreover, those metrics are not sufficient for ontology-based information extraction, because the distinction between right and wrong is less obvious [M05]. We do not apply any information extraction techniques in our system and the current prototype of the annotation tool mainly supports manual annotation.

We have chosen SEQUAL that has been widely and successfully applied in the information modelling area, to evaluate the proposed approach. According to the semiotic quality categories, a quality analysis has been made at both the meta-model level (GPO and the PSAM specifications) and model level (the PSAM model instances and Pro-SEAT).

According to the quality analysis results, we found out some limitations of the work and possible extensions:

- Further automatic enhancement is needed to facilitate the annotation procedure.
- The approach should be evaluated in various domains and applications
- Social quality issues should be evaluated, in particular relative to semantic reconciliation of models
- Evolution of ontologies and model changes have not been taken into account in the annotation.
- Semantics of relationships in the original models have not been taken into account in this approach
- Relationship between the semantic annotations of process models at the conceptual level and at the execution level could be developed.

References

- [BPMN04] BPMI.org: Business process modelling notation version 1.0 May 3, 2004. <http://www.bpmn.org/Documents/BPMN%20V1-0%20May%203%202004.pdf> Last visited 2007.9.1
- [CKSL97] Carlsen, S., J. Krogstie, A. Sølvyberg, and O.I. Lindland. Evaluating Flexible Workflow Systems. in *Proceedings of Hawaii International Conference on System Sciences (HICSS-30)*. 1997. Maui, Hawaii.
- [HSS03] Handschuh, S.; Staab, S.; Studer, R.: Leveraging metadata creation for the semantic web with CREAM. In *Proc. of the Annual German Conference on Advances in Artificial Intelligence (KI 2003)*, page 19-33, LNCS 2821, Springer, 2003
- [K01] Krogstie, J. A Semiotic Approach to Quality in Requirements Specifications. in *Proceedings of IFIP 8.1. Working Conference on Organizational Semiotics*. 2001. Montreal, Canada.
- [K03] Krogstie, J., Evaluating UML Using a Generic Quality Framework, in *UML and the Unified Process*, L. Favre, Editor. 2003, IRM Press. p. 1-22.
- [K08] Krogstie, J.: Integrated goal, data and process modelling: from TEMPORA to model-generated work-places. *Information systems engineering: from data analysis to process networks*, IGI publishing, 2008.

- [KA04] Krogstie, J. and S. Arnesen, Assessing Enterprise Modelling Languages using a Generic Quality Framework, in *Information Modelling Methods and Methodologies*, J. Krogstie, K. Siau, and T. Halpin, Editors. 2004, Idea Group Publishing.
- [KDJ05] Krogstie, J., V. Dalberg, and S. Jensen, Using a Model Quality Framework for Requirements Specification of an Enterprise Modelling Languages, in *Advanced Topics in Database Research - Vol. 4*, K. Siau, Editor. 2005, Idea Group Publishing.
- [KSJ06] Krogstie, J., G. Sindre, and H. Jørgensen, Process Models as Knowledge for Action: A Revised Quality Framework. *European Journal of Information Systems* 2006. 15(1): p. 91-102.
- [KH01] Kogut, P.; Holmes, W.: AeroDAML: Applying information extraction to generate DAML annotations from web pages. In *Proc. Of the 1st International Conference on Knowledge and Capture (K-CAP 2001)*, 2001
- [L08] Lin, Y.: Semantic annotation for process models: facilitating process knowledge management via semantic interoperability. *PhD thesis*, Norwegian University of Science and Technology, 2008
- [LS07] Lin, Y.; Sølvberg, A.: Goal annotation of process model for semantic enrichment of process knowledge. In *Proc. Of the 19th international Conference on Advanced Information Systems Engineering (CAiSE 2007)*, page 355-369, LNCS 4495, Springer-Verlag, 2007
- [LSHD04] Lin, Y.; Sampson, J.; Hakkarainen, S.; Ding, H.: An evaluation of UML and OWL using a semiotic quality framework. *Advanced Topics in Database Research Volume 4*, pages 178--200, Idea Group Publishing. Hershey, PA, USA, 2004.
- [LSHKS06] Lin, Y.; Stransunskas, D.; Hakkarainen, S.; Krogstie, J.; Sølvberg, A.: Semantic annotation framework to manage semantic heterogeneity of process models. In *Proc. of the 18th international Conference on Advanced Information Systems Engineering (CAiSE 2006)*, page 433-446, LNCS 4001, Springer-Verlag, 2006
- [M05] Maynard, D.: Benchmarking ontology-based annotation tools for the semantic web. In *Workshop of Text Mining, e-Research and Grid-enabled Language Technology at UK e-Science Programme All Hands Meeting (AHM 2005)* Nottingham, UK, 2005.
- [NK06] Nysetvold, A.G. and J. Krogstie, Assessing Business Process Modelling Languages Using a Generic Quality Framework, in *Advanced Topics in Database Research Series* K. Siau, Editor. 2006, Idea Group. p. 79-93.
- [OWL09] W3C. OWL web ontology language overview. <http://www.w3.org/TR/2004/REC-owl-features-20040210>, Last Visited: 2009.1.26
- [RRK07] Recker, J., M. Rosemann, and J. Krogstie, Ontology- versus pattern-based evaluation of process modelling language: A comparison. *Communications of the Association for Information Systems* 2007. 20: p. 774-799.
- [R79] Rijsbergen, C.J.: *Information Retrieval* (2nd edition). Butterworths, 1979.
- [SAWSDL07] W3C. Semantic annotations for WSDL and XML schema. <http://www.w3.org/TR/sawSDL>, Last Visited: 2007.9.20
- [SCOR07] SCOR model. <http://www.supply-chain.org/page.wv?section=SCOR+Model&name=SCOR+Model>, Last Visited: 2007.5.10
- [SMSV04] Sivashanmugam, K.; Miller, J.; Sheth, A.; Verma, K.: Framework for semantic web process composition. Special Issue of the *International Journal of Electronic Commerce (IJEC)*, 9 (2), 2004
- [SN00] Scheer, A; Nuttgens, M.: ARIS architecture and reference models for business process management. *Business process management, models, techniques, and empirical studies*, page 376-389, ISBN 3-540-67454-3, Springer-Verlag, 2000