Semantic EPC: Enhancing Process Modeling Using Ontology Languages

Oliver Thomas, Michael Fellmann

Institute for Information Systems (IWi)
at the German Research Center for Artificial Intelligence (DFKI),
Saarbruecken (Germany)
`{oliver.thomas|michael.fellmann}@iwi.dfki.de`
`http://iwi.dfki.de`

Abstract. In this article we describe a semantic extension of event-driven process chains, with which it is possible to specify the semantics of individual model elements as it is indicated by their label in natural language using concepts of a formal ontology. To do so, a multi-level approach was developed, which comprises an ontology level, a metadata level, as well as a model level. With the approach presented here, ambiguity that is introduced by the use of natural language in semi-formal models can be removed. Moreover, new possibilities of reasoning over business process models are introduced which improve the analysis, search and validation of business processes.[1]

Keywords: Process Modeling, Modeling Languages, Event-Driven Process Chain, Semantic Web, Enterprise Ontologies, Ontology Languages

1. Introduction

A multitude of modeling languages for the representation of processes have been developed since the first large data processing applications [4]. Examples are the Petri net [26], the event-driven process chain [23], the UML activity diagram [6] or the Business Process Modeling Notation (BPMN) [7]. The models described by these modeling languages serve the communication between employees in an organization with specialist knowledge and those, with methodical or technical knowledge such as for example, consultants or software engineers [29]. One tries to avoid the problem of fuzziness in natural language and the many problems in the inherent impracticability of mathematical formulations through semi-formal, graphic forms of representation in modeling languages. These are based closely on specialized business terms, exact enough, however that the models can serve as a starting point for the implementation of computer-supported information systems.

---

[1] An extended version of this article will appear in the Special Issue on Information Modeling and Ontologies of the International Journal of Interoperability in Business Information Systems (http://ibis-journal.net/).

In: M. Hepp, K. Hinkelmann, D. Karagiannis, R. Klein, N. Stojanovic (eds.): Semantic Business Process and Product Lifecycle Management. Proceedings of the Workshop SBPM 2007, Innsbruck, April 7, 2007, CEUR Workshop Proceedings, ISSN 1613-0073, online CEUR-WS.org/Vol-251/

Even though this is a fundamental idea for the model-driven development of information systems [14; 17], the said linkage between natural language and graphic representation forms is a main problem of semi-formal modeling languages. The identifiers of the individual elements of a business process model are added in a natural language by the modeler, irrespective of his decision for a certain modeling language. An essential part of the semantics of a process model is thus always bound to the natural language, which, with its ambiguities, allows much room for interpretation. This is not a problem as long as a model is created and read by only one person. Clearly defined semantics for each model element is however necessary, if process models from various modelers are combined, searched and translated [28] or if it is planned that the semantics in the models should be automatically validated and used for the configuration of an information system .

The problem mentioned above can be met through the linkage of the elements of a business process model with concepts from an ontology. In this article we will develop such a semantic extension for a process modeling language, which represents the semantics of the labels of process model elements with concepts of a formal ontology. This semantic extension will be carried out exemplified by the EPC. We selected the process modeling language EPC because of its popularity in modeling practice. However, our approach is principally transferable to other semi-formal modeling languages, such as for example the UML activity diagram or BPMN.

2. Related Work

The idea of using ontologies in the area of business process management is not new. For example, Wand and Weber have used ontologies to describe and evaluate certain aspects of modeling languages [31; 32].

The core area of related work can be found at the intersection of business process management and semantic web, which was currently discussed in the workshop “Semantics for Business Process Management” at the ESWC 2006 [8]. In addition to application possibilities in industry, the usability of ontologies in bridging of semantic differences for administrative processes was exemplified [20]. However, there was no contribution showing a framework for the interplay of process modeling languages and ontologies.

While our approach to the annotation of business process models is, in principle, designed language-independent, there are related projects that are geared exclusively to the semantic annotation of models in a certain language. An approach to semantic annotation for Petri nets [11], a formal framework for process description [15], as well as a tool for the semi-automatic completion of models during model construction on the basis of similarity analyses exist for example [10]. A concept for the automatic synthesis and modification of models after changes to sub-processes also exists for the UML activity diagram [24].

While we focus more on business-level process models, the potential of combining process models with (semantic) web services is described in [18; 19]. This work can be seen as complementary to our approach and might be used in the future in order to provide a framework for the integration of semantic business-level and IT-level process models.

Semtalk is a tool for the linkage of EPC-models with ontologies on the basis of Microsoft VISIO [12]. However, with this tool the semantics of the EPC-model elements is bound to the properties and operations of objects (in the object-oriented meaning), which heavily limits the usability of the modeling language.

3. Research Methodology

With the approach presented here, the semantics of individual model elements will be specified using concepts from a formal ontology. The linkage of model elements with the ontology required for this will be realized using a separate metadata level. Thus, the modeling tools and data formats remain usable while the metadata can be saved in formats accessible to the direct machine processing of the semantics contained in the models.

Altogether, the connections illustrated in the framework for the semantic annotation of business process models exist between models, metadata and ontologies (cp. Fig. 1). Metadata is generated from models (arrow from “Models” to “Metadata”). This metadata contains references to the model elements of the initial model, as well as to the concepts of the ontology. Ontologies and metadata are interdependent (double-headed arrow between “Ontologies” and “Metadata”). Concepts from the ontology are used in the metadata to specify the meaning of model elements. Therefore, the ontologies used must contain the required concepts or they must be added to the ontologies in the course of the creation of the metadata.

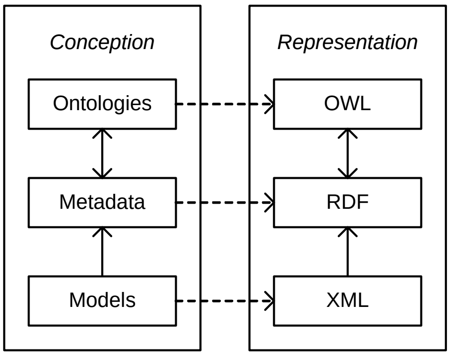


Fig. 1. Framework for the semantic annotation of business process models

The conceptual elements of the approach presented here can be assigned to representation formats for implementation purposes. These can be seen on the right side of Fig. 1 and will be introduced at a later point in time.

In the course of this article, we will first discuss ontologies for business process management. Then, in the main part of the article, we will show how ontologies and event-driven process chains can be combined to form an integrated approach to semantic business process modeling. Finally, the article closes with a discussion and an outlook.

4. Ontologies for Semantic Business Process Management

A standardization of terms for and concepts on ontologies has been the topic of research for years in the field of artificial intelligence and the semantic web. According to Gruber, an ontology is “a formal, explicit specification of a shared conceptualization” [16]. In this article, we transfer the basic idea of the semantic web which is to give information a well-defined meaning in order to make it processable both for humans and machines [9], to the field of business process management. In our approach, ontologies are not only used to clarify the semantics of individual model elements, but also to infer new facts not included in the original process model to enable advanced search and validation capabilities (see also section 5.3).

There are various languages for the explicit and formal representation of an ontology such as, for example CML, Conceptual Representation, CycL, KIF, Loom, OIL and the Web Ontology Language (OWL). OWL [1] is a standard from the World Wide Web Consortium (W3C), which resulted from the merging of DARPA and OIL. OWL will be used here as the language for representing ontologies due to its increased acceptance and, in connection with this, the support of the language through software libraries and tools. OWL is available in three variations: OWL Lite, OWL DL and OWL Full, however, the level “DL” is sufficient for the ontologies discussed in this article.

It is unnecessary to develop completely new ontologies for semantic business process management. First, one should leverage existing ontologies. In the area of enterprise and process modeling, relevant ontologies include the Enterprise Ontology [30], TOVE [TOronto Virtual Enterprise, 13] and BMO [Business Management Ontology, 22]. These ontologies provide a starting point for the coherent description of the enterprise. Second, the definitions for ontology-construction found in established technical standards and vocabularies can be reused as valuable assets. These are, for example, in the business processes field ebXML and RosettaNet, for business transactions EDIFACT and OpenTrans, for business documents UBL and xCBL, for the classification of products and services UNSPSC, eCl@ss, cXML and ISIC – to name but a few. In addition to these enterprise-spanning standards, ontologies can, third, also be obtained from the company-specific conceptualization of a domain. For this, ontologies can also be derived from entity relationship models common in the environment of relational databases and ERP-systems using the Ontology Definition Metamodel (ODM) [3] proposed by the OMG.

In the following, we will show a simple example of an ontology and illustrate it with a graphic representation (cp. Fig. 2). Properties symbolized by arrows signify object properties (ObjectProperties) in OWL, which correlate the instances of classes to one another. Inheritance relations refer to the language construct `rdfs:subClassOf` used in RDF and OWL.

The ontology framework exemplarily contains classes for organizational units, tasks, events, services and rules as relevant elements of an enterprise description. These classes can be specialized arbitrarily. In our example, the classes `Event` and `Service` were further specialized (cp. Fig. 2). In addition to classes, the example ontology contains instances, which symbolize a member of a class. The properties `partOf` and `uses` are defined to be transitive, so that additional facts can be inferred by querying the ontology with query languages. In the course of this article, our example ontology will be used to specify the model element-specific semantics of the elements of an EPC-model.

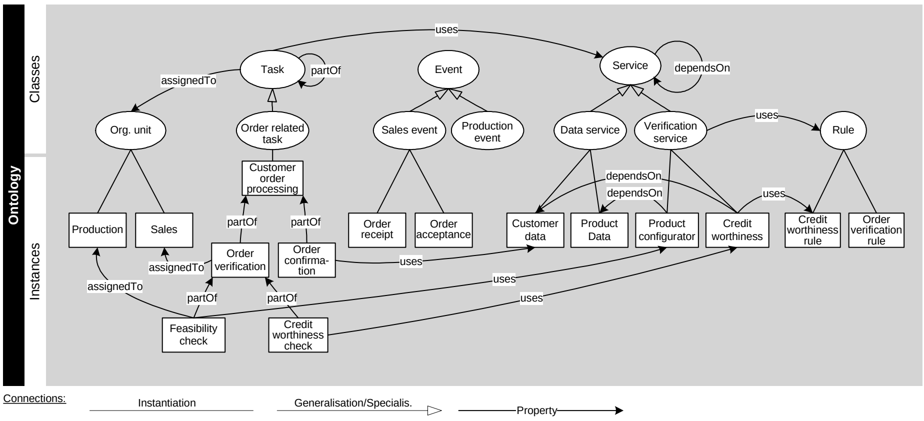


Fig. 2. Framework for an enterprise ontology

5. Semantic Event-Driven Process Chains

5.1. The Modeling Language EPC

The event-driven process chain is a modeling language for the representation of business processes common in research and practice. It was developed at the Institute for Information Systems at the Saarland University in Saarbruecken, in cooperation with the SAP, Inc. [23]. An EPC-model is a directed and connected graph, whose nodes are events, functions and logical connectors. Fig. 3 shows an example EPC-model, which describes the process for customer order processing.

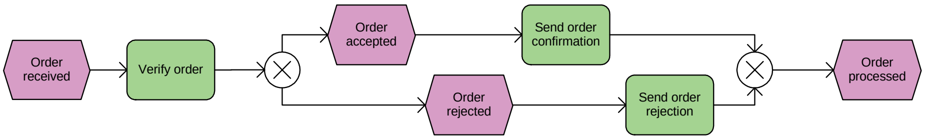


Fig. 3. EPC-model for customer order processing

Events are the passive elements in the EPC and are represented by hexagons. Functions, represented by rounded rectangles, are the active elements in the EPC. The term “function” is equivalent to the term “task” in the EPC [23]. While functions represent time-consuming happenings, events occur at a certain point in time. In literature, the respective object and an infinitive verb are suggested as a naming convention for functions, whereas for events, the object that experiences the change, as well as a verb in perfect tense, which states the type of change are suggested [27]. Events trigger functions and are their result. Control flow edges represent the relationships between functions and events. Conjunctive “$\owedge$“, adjunctive “$\ovee$“ and disjunctive “$\otimes$“ logical connectors are introduced to express that functions are started by one or more events resp. that a function can create one or more events as a result (cp. Fig. 3). They are referred to as AND-, OR- resp. XOR-connectors.

5.2. Ontology-based Representation of the EPC

To specify the semantics of EPC-model elements through relations to ontology concepts, the EPC first must be represented within the ontology. In regard to the representation of the EPC in the ontology, one can differentiate between a representation of EPC-language constructs and a representation of EPC-model elements. EPC-language constructs such as “function” or “event”, as well as the control flow are created in the ontology as classes and properties. Subsequently, the EPC-model elements can be represented through the instantiation of these classes and properties in the ontology. Fig. 4 shows this by means of a simple process fragment.

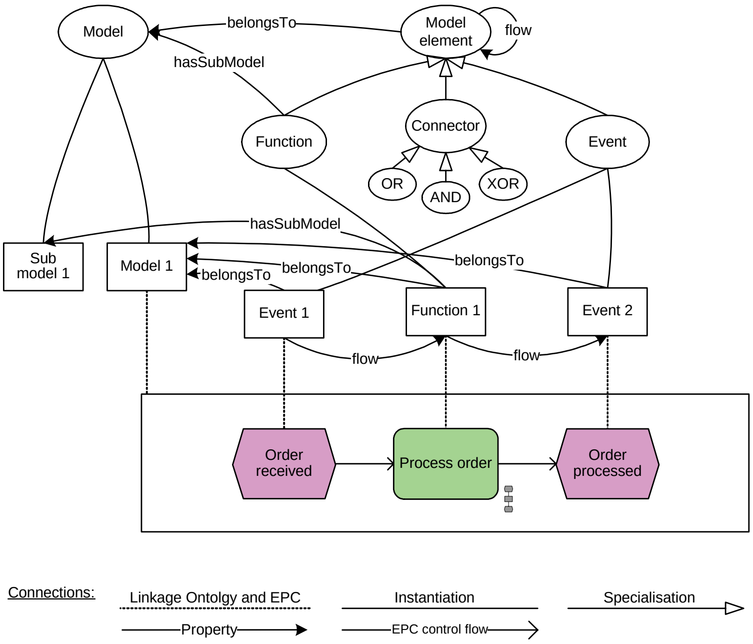


Fig. 4. Representation of the EPC in the ontology

5.3. The Linkage between EPC-Model Elements and Ontology Instances

The linkage of EPC-model elements with ontology instances can also be referred to as a process of semantic annotation. The EPC-model elements already represented in the ontology (cp. preceding section) are thereby put in relation to further instances of the ontology. Fig. 5 shows this linkage based on the example process of Fig. 3 and the example ontology represented in Fig. 2. The linkage of the ontology and EPC-model element instances is accomplished by the usage of properties; these are represented in Fig. 5 as `semType`-properties. Just as the name indicates, these properties specify the semantics of an EPC-model element through a relation to an ontology instance with formal semantics defined by the ontology.

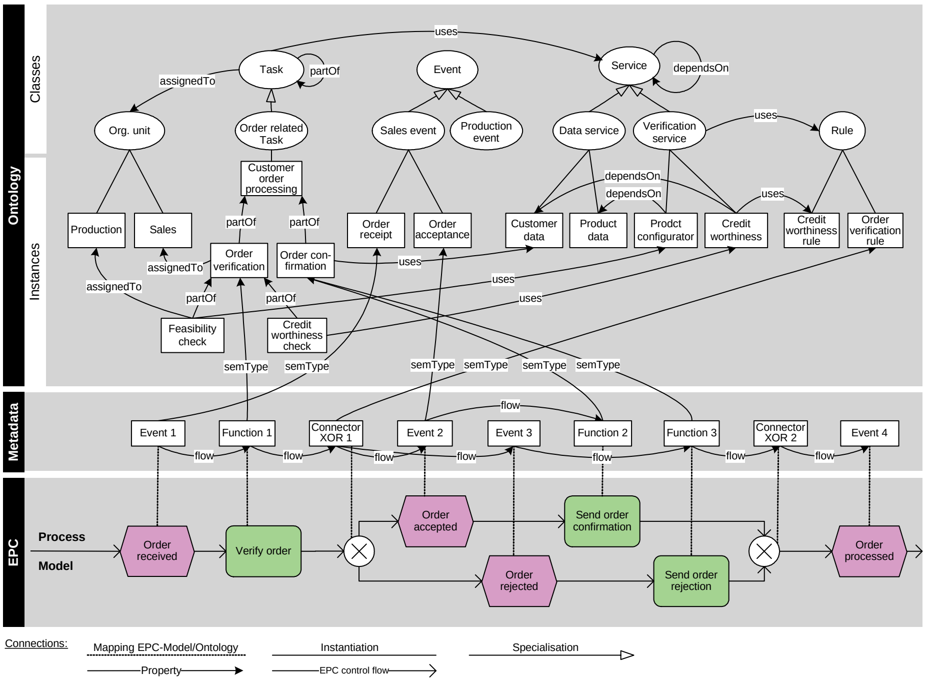


Fig. 5. Semantically annotated process model “customer order processing”

In addition to the decoupling of the semantics of an individual model element from its natural language label, the context of a model element is specified more accurately through the linkage of an ontology instance to the model element. This happens via relations, which exist between the ontology instance representing the EPC-model element and further instances of the ontology. In principle, such a specification of relations to further concepts, such as organizational units or resources, was already suggested with the extended EPC [27] and other approaches to multi-perspective modeling. In contrast to these approaches, the concept presented here uses a flexible, graph-based data model, which allows machine-processable semantics that can be extended by integrating rules.

By means of the graph-based data model provided by the Resource Description Framework (RDF) [2] and OWL, a business process is represented in the semantic metadata as an directed graph with nodes and edges. Consequently, one can traverse the graph jumping from one node to the next via properties using simple patterns, also referred to as graph pattern matching. An example for such a query is the question in the example in Fig. 5, as to whether an EPC-function exists, connected via a property `semType` to a `Task`, whose parts are connected via a property `uses` with instances of the class `Service`, which in turn are connected via a property `uses` with an instance of the class `Rule`. With SPARQL [5], which is recommended by the W3C, we already have a query language for carrying out such queries.

Moreover, new facts that are not explicitly created in the process model by the modeler can be inferred during the execution of the query. In the example in Fig. 5, one can conclude through the transitive definition of the property `partOf`, that the `feasibility check` is a part of `customer order processing`. Rule languages allow a significant extension of the machine-processable semantics. Rules can be embedded in the OWL-ontology using SWRL (Semantic Web Rule Language) [21]. SWRL rules can be expressed using the syntax of OWL, therefore allowing a tight integration of ontologies and rules. An example for a simple rule is the uncle-rule, which implies an uncle-relation through the composition of parent and brother-relations:

$$\text{parent}(?x,?y) \wedge \text{brother}(?y,?z) \;\Rightarrow\; \text{uncle}(?x,?z)$$

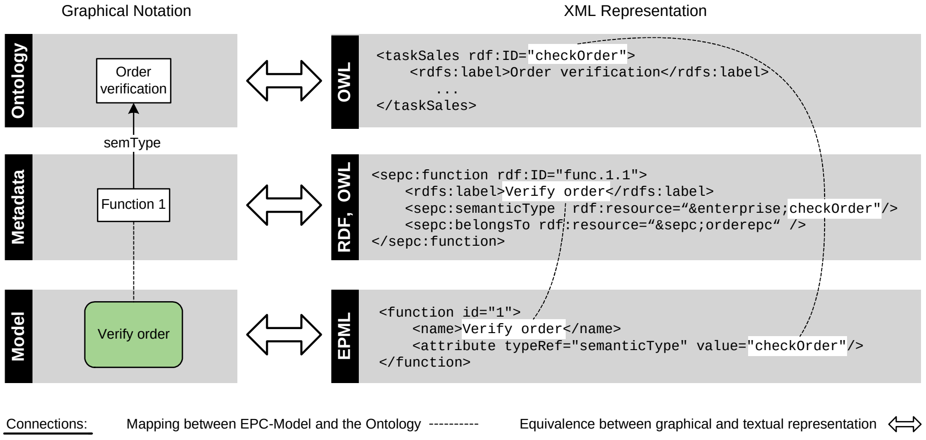
Transferred to business process modeling, such rules allow, as integrity rules, an advanced semantic validation. Thus, for example, the policy can be formulated that all business process related to “order processing” must contain a function “customer confirmation”. In addition, new facts can be won in the form of derivation rules during runtime. Thus, for example, we can conclude that a process, which contains a function that requires semi-finished products, reduces stock.

5.4. RDF-Representation of the Semantic EPC

In technical terms, the linkage of EPC-model elements is realized by adding attributes to the XML-representation of an EPC-model. These attributes identify the ontology instance which semantically specifies the relevant process model element. Fig. 6 illustrates this graphically, as well as with the corresponding XML-vocabularies EPML (Event-Driven Process Markup Language) for the EPC-representation [25], RDF for a semantic representation of the EPC – referred to as sEPC – and OWL for the representation of ontology classes and instances.

As we can see in Fig. 6, a linkage of the EPC-model element and ontology instance occurs over an intermediate step in the form of metadata. This metadata references both the ontology instance and the process model element, which is indicated by the dashed line connecting `checkOrder` in the process model, in the metadata and in the ontology (cp. also Fig. 5). In addition, the natural language labels of the EPC-model elements are used as names in the metadata in the field `rdfs:label` (cp. Fig. 6), indicated by another dashed line going from `name` in the EPML-data to `rdfs:label` in the RDF-data.

Seen from a conceptual point of view, the expressiveness of RDF is sufficient for the metadata, because language constructs from OWL are not used. Seen from a technical view however, then OWL DL is necessary, because the ontology instances used for the annotation must be imported into the metadata for querying and reasoning purposes.



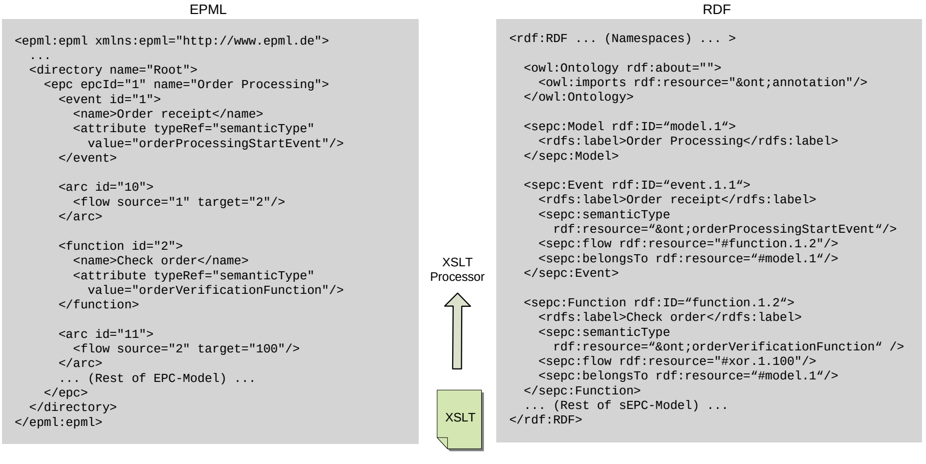
```
<taskSales rdf:ID="checkOrder">
    <rdfs:label>Order verification</rdfs:label>
       ...
</taskSales>
```

```
<sepc:function rdf:ID="func.1.1">
    <rdfs:label>Verify order</rdfs:label>
    <sepc:semanticType  rdf:resource=“&enterprise;checkOrder"/>
    <sepc:belongsTo rdf:resource=“&sepc;orderepc“ />
</sepc:function>
```

```
<function id="1">
    <name>Verify order</name>
    <attribute typeRef="semanticType" value="checkOrder"/>
</function>
```

Fig. 6. Linkage of EPC-models with ontologies (representation)

After the linkage of the EPC-model with the ontology instances, a complete transformation of the EPC into an sEPC can take place on the basis of the representation formats. The sEPC consists of the XML-representation of the metadata shown exemplarily in Fig. 6. The transformation is shown in Fig. 7.



```
<epml:epml xmlns:epml="http://www.epml.de">
  ...
  <directory name="Root">
    <epc epcId="1" name="Order Processing">
      <event id="1">
        <name>Order receipt</name>
        <attribute typeRef="semanticType"
          value="orderProcessingStartEvent"/>
      </event>

      <arc id="10">
        <flow source="1" target="2"/>
      </arc>

      <function id="2">
        <name>Check order</name>
        <attribute typeRef="semanticType"
          value="orderVerificationFunction"/>
      </function>

      <arc id="11">
        <flow source="2" target="100"/>
      </arc>
      ... (Rest of EPC-Model) ...
    </epc>
  </directory>
</epml:epml>
```

```
<rdf:RDF ... (Namespaces) ... >

  <owl:Ontology rdf:about="">
    <owl:imports rdf:resource="&ont;annotation"/>
  </owl:Ontology>

  <sepc:Model rdf:ID=“model.1“>
    <rdfs:label>Order Processing</rdfs:label>
  </sepc:Model>

  <sepc:Event rdf:ID=“event.1.1“>
    <rdfs:label>Order receipt</rdfs:label>
    <sepc:semanticType
      rdf:resource=“&ont;orderProcessingStartEvent“/>
    <sepc:flow rdf:resource="#function.1.2"/>
    <sepc:belongsTo rdf:resource=“#model.1“/>
  </sepc:Event>

  <sepc:Function rdf:ID=“function.1.2“>
    <rdfs:label>Check order</rdfs:label>
    <sepc:semanticType
      rdf:resource=“&ont;orderVerificationFunction“ />
    <sepc:flow rdf:resource="#xor.1.100"/>
    <sepc:belongsTo rdf:resource=“#model.1“/>
  </sepc:Function>
  ... (Rest of sEPC-Model) ...
</rdf:RDF>
```

Fig. 7. Transformation from EPML to RDF

6. Conclusion and Outlook

When selecting a modeling language for the representation of business processes one must balance between formal precision and pragmatic manageability. Modeling languages with formal semantics are suited for machine processing. The interpretation of real-world interrelations can however, become very complex. With our approach, the gap between formal and semi-formal languages can be closed by linking model elements from semi-formal languages with concepts from formal ontologies and thus, receiving a formal semantic. The advantages of this transformation of process models into semantic process models using OWL are:

- *Process knowledge:* On the one hand, the understanding of business processes is increased through the linkage of model elements with the concepts of an ontology, because clearly defined terms are used and on the other, the elements of a business process are thus embedded in a certain context. This context can contain further specialized and technical information, which makes semantically annotated process models suitable as a starting point for process-oriented knowledge management.
- *Process representation:* The effort of “internationalizing” process models is reduced, because identifiers can be stored in the ontology in several languages and are thus, made usable for the automated translation of the labels of the model elements.
- *Process search:* Queries to process models can take place on the semantic level. By using inference mechanisms and rule languages, new facts not explicitly contained in the process models can be inferred at query time.
- *Process validation:* In addition to the syntactic rules defined by the meta-model of a process modeling language such as the EPC, the validation of process models can also occur on a semantic level by the usage of a rule base, which is stored in the ontology. Semantically incorrect business process models can thus be identified before process execution and policies can be enforced on all of the business processes consistently.
- *Process execution:* Process execution is simplified because the ontology acts as the central repository of a hybrid, i. e. a conceptual, as well as technical description of the elements of a business process. Best practices in the transfer of conceptual processes in IT-systems can thus be centrally stored in the ontology, free of redundancies and reusable by means of semantically annotated process models.

The need for further research with reference to the semantic annotation of process models exists regarding IT-support for the approach presented, in particular for the IT-based realization of the annotation. Interesting is also the question as to how to deal with dynamics, i. e. changes in the ontologies used for annotation, as well as the connection of the approach to semantic web services or web services repositories.

7. References

[1] Smith, M. K.; Welty, C.; McGuiness, D. L. (eds.): *OWL Web Ontology Language Guide : W3C Recommendation 10 February 2004.* W3C, 2004

[2] W3C (ed.): *Resource Description Framework (RDF).* URL http://www.w3.org/RDF/ [20.01.2007]

[3] OMG (ed.): *Ontology Definition Metamodel : Third Revised Submission to OMG/ RFP ad/2003–03–40.* Needham : OMG, 2005

[4] Dumas, M.; van der Aalst, W. M. P.; ter Hofstede, A. H. M. (eds.): *Process-aware Information Systems : Bridging People and Software through Process Technology.* Hoboken, New Jersey : Wiley, 2005

[5] Prud‘hommeaux, E.; Seaborne, A. (eds.): *SPARQL Query Language for RDF : W3C Working Draft 23 November 2005.* W3C, 2005

[6] Object Management Group (ed.): *Unified Modeling Language: Superstructure, version 2.0, formal/05–07–04.* Needham : Object Management Group, 2005

[7] Object Management Group (ed.): *Business Process Modeling Notation Specification : Final Adopted Specification dtc/06–02–01.* Needham : Object Management Group, 2006

[8] Hinkelmann, K.; Karagiannis, D.; Stojanovic, N.; Wagner, G. (eds.): *Proceeding of the Workshop on Semantics for Business Process Management at the 3rd European Semantic Web Conference 2006, Budva, Montenegro, June 2006*

[9] Berners-Lee, T.; Hendler, J.; Lassila, O.: *The Semantic Web.* Scientific American, 2001

[10] Betz, S.; Klink, S.; Koschmider, A.; Oberweis, A.: Automatic User Support for Business Process Modeling. In: *Proceeding of the Workshop on Semantics for Business Process Management at the 3rd European Semantic Web Conference 2006, Budva, Montenegro, June 2006*, pp. 1–12

[11] Brockmans, S.; Ehrig, M.; Koschmider, A.; Oberweis, A.; Studer, R.: Semantic Alignment of Business Processes. In: Manolopoulos, Y.; Filipe, J.; Constantopoulos, P.; Cordeiro, J. (eds.): *Proceedings of the Eighth International Conference on Enterprise Information Systems (ICEIS 2006).* Paphos, Cyprus : INSTICC Press, 2006, pp. 191–196

[12] Fillies, C.; Weichhardt, F.: *On Ontology-based Event-driven Process Chains.* URL http://www.semtalk.com//pub/semtalkepk.pdf [25.11.2005]

[13] Fox, M. S.: The TOVE Project: A Common-sense Model of the Enterprise. In: Belli, F.; Radermacher, F. J. (eds.): *Industrial and Engineering Applications of Artificial Intelligence and Expert Systems, 5th International Conference, IEA/AIE – 92, Paderborn, Germany, June 9–12, 1992, Proceedings.* London : Springer, 1992 (LNCS), pp. 25–34

[14] Frankel, D. S.: *Model driven architecture : applying MDA to enterprise computing.* Indianapolis, IN : Wiley, 2003

[15] Greco, G.; Guzzo, A.; Pontieri, L.; Saccá, D.: An Ontology-Driven Process Modeling Framework. In: Galindo, F.; Takizawa, M.; Traunmüller, R. (eds.): *Database and Expert Systems Applications: 15th International Conference, DEXA 2004, Zaragoza, Spain, August 30-September 3, 2004. Proceedings.* Berlin : Springer, 2004 (LNCS), pp. 13–23

[16] Gruber, T. R.: Toward principles for the design of ontologies used for knowledge sharing? In: *International Journal of Human-Computer Studies* 43 (1995), No. 5–6, pp. 907–928

[17] Hailpern, B.; Tarr, P.: Model-driven development: The good, the bad, and the ugly. In: *IBM Systems Journal* 45 (2006), No. 3, pp. 451–461

[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: *Proceedings of the IEEE ICEBE 2005, October 18–20, Beijing, China.* Bejing, China, 2005, pp. 535–540

[19] Hepp, M.; Roman, D.: An Ontology Framework for Semantic Business Process Management. In: Oberweis, A.; Weinhardt, C.; Gimpel, H.; Koschmider, A.; Pankratius, V.;

Schnizler, B. (eds.): *eOrganisation: Service-, Prozess-, Market-Engineering : 8. Internationale Tagung Wirtschaftsinformatik ; Karlsruhe, 28. Februar – 2. März 2007 ; Band 1.* Karlsruhe : Universitätsverlag, 2007, pp. 423–440

[20] Herborn, T.; Wimmer, M. A.: Process Ontologies Facilitating Interoperability in eGovernment A Methodological Framework. In: Hinkelmann, K.; Karagiannis, D.; Stojanovic, N.; Wagner, G. (eds.): *Proceeding of the Workshop on Semantics for Business Process Management at the 3rd European Semantic Web Conference 2006, Budva, Montenegro, June 2006*, pp. 76–89

[21] Horrocks, I.; Patel-Schneider, P. F.; Boley, H.; Tabet, S.; Grosof, B.; Dean, M.: *SWRL: A Semantic Web Rule Language : Combining OWL and RuleML ; W3C Member Submission 21 May 2004.* W3C, 2004

[22] Jenz, D. E.: *Strategic White Paper: Ontology-Based Business Process Management ; The Vision Statement.* Erlensee : Jenz & Partner GmbH, 2003

[23] Keller, G.; Nüttgens, M.; Scheer, A.-W.: Semantische Prozeßmodellierung auf der Grundlage "Ereignisgesteuerter Prozeßketten (EPK)". In: Scheer, A.-W. (ed.): *Veröffentlichungen des Instituts für Wirtschaftsinformatik,* No. 89, Saarbrücken : Universität des Saarlandes, 1992 (in German)

[24] Lautenbacher, F.; Bauer, B.: Semantic Reference- and Business Process Modeling enables an Automatic Synthesis. In: Hinkelmann, K.; Karagiannis, D.; Stojanovic, N.; Wagner, G. (eds.): *Proceeding of the Workshop on Semantics for Business Process Management at the 3rd European Semantic Web Conference 2006, Budva, Montenegro, June 2006*, pp. 89–100

[25] Mendling, J.; Nüttgens, M.: *EPC Markup Language (EPML) – An XML-Based Interchange Format for Event-Driven Process Chains (EPC) : Technical Report JM–2005–03–10.* Vienna University of Economics and Business Administration, 2005

[26] Petri, C. A.: *Kommunikation mit Automaten.* Bonn : Mathematisches Institut der Universität Bonn, 1962 (in German)

[27] Scheer, A.-W.; Thomas, O.; Adam, O.: Process Modeling Using Event-driven Process Chains. In: Dumas, M.; van der Aalst, W. M. P.; ter Hofstede, A. H. M. (eds.): *Process-aware Information Systems : Bridging People and Software through Process Technology.* Hoboken, New Jersey : Wiley, 2005, pp. 119–145

[28] Thomas, O.: Joint Reference Modeling: Collaboration Support through Version Management. In: Sprague, R. H. (ed.): *Proceedings of the 40th Annual Hawaii International Conference on System Sciences : 3–6 January 2007, Big Island, Hawaii ; Abstracts and CD-ROM of Full Papers.* Los Alamitos, CA : IEEE Computer Society Press, 2007

[29] Thomas, O.; Scheer, A.-W.: Tool Support for the Collaborative Design of Reference Models – A Business Engineering Perspective. In: Sprague, R. H. (ed.): *Proceedings of the 39th Annual Hawaii International Conference on System Sciences : 4–7 January 2006, Kauai, Hawaii ; Abstracts and CD-ROM of Full Papers.* Los Alamitos, CA : IEEE Computer Society Press, 2006

[30] Uschold, M.; King, M.; Moralee, S.; Zorgios, Y.: The Enterprise Ontology. In: *The Knowledge Engineering Review* 13 (1998), No. 1, pp. 31–89. – Special Issue on Putting Ontologies to Use

[31] Wand, Y.; Weber, R.: Research Commentary: Information Systems and Conceptual Modeling – A Research Agenda. In: *Information Systems Research* 13 (2002), No. 4, pp. 363–376

[32] Weber, R.: *Ontological foundations of information systems.* Melbourne : Coopers & Lybrand and the Accounting Association of Australia and New Zealand, 1997