TOWARDS A DATA INTEGRATION APPROACH BASED ON BUSINESS PROCESS MODELS AND DOMAIN ONTOLOGIES

Fernanda Baião, Flavia Santoro, Hadeliane Iendrike, Claudia Cappelli
Mauro Lopes, Vanessa T. Nunes
NP2Tec – Research and Practice Group in Information Technology, Department of Applied Informatics
Federal University of the State of Rio de Janeiro (UNIRIO), Rio de Janeiro, Brazil

Ana Paula Dumont
Department of Business Solutions for Petroleum Exploration and Production
Petrobras – Brazilian Petroleum Organization, Rio de Janeiro, Brazil

Keywords: Ontology, Business Process Modeling, Data Integration.

Abstract: Information integration is still a challenge in the Information Systems research area. Domain ontologies are intensively studied to solve this problem, since they allow people and software agents to share common agreement about information and semantics on a specific domain of knowledge. However, for this integration to be carried out effectively, the ontology should be kept up-to-date according to concept definitions and current business rules. This may be very difficult to achieve in dynamic organizations. In this paper we present an approach for developing domain ontologies from business process models, thus helping in building integrated data models.

1 INTRODUCTION

Building integrated technological solutions in an organization without non-desired information redundancy within several databases is still a challenge in the Information Systems research area. The maintenance of knowledge and consistent databases is a difficulty faced, since in general, systems are developed based on particular requirements to support daily user activities, without considering the way such activities are integrated to the business as a whole. It is very difficult to find real scenarios in which the system development cycle includes activities for analyzing common information manipulated and the possible integration to existing databases, in order to prevent data redundancy that can impact its consistency.

Noy and McGuiness (2001) recognize the use of Ontology to share common agreement about information among people and software agents. Domain ontologies make the understanding of a domain explicit, allowing reuse, separation from the operational knowledge and analysis of the domain knowledge. The concepts represented in an ontology are a starting point for the logical and physical data models, serving as a reference for data integration.

On the other hand, business processes models include elements that express domain concepts and therefore are able to facilitate the analysis of the information from a conceptual point of view. The modeling of business processes allows establishing semantic relationships among concepts used in the processes definition, and thus makes the creation of common sense possible.

The goal of this paper is to present a method for construction and maintenance of domain ontology derived from business process models in which information manipulated throughout its activities is identified. We show that the association of information and activities produces a resource for conceptualization, minimizes the risk of bad interpretation of concepts and allows keeping a safe reference to carry out integrated data models.

The paper is organized as: Section 2 discusses related works about business process modeling and ontology; Section 3 presents the method proposed, and Section 4 concludes the paper and points out future work.

Baião F., Santoro F., Iendrike H., Cappelli C., Lopes M., T. Nunes V. and Paula Dumont A. (2008).
TOWARDS A DATA INTEGRATION APPROACH BASED ON BUSINESS PROCESS MODELS AND DOMAIN ONTOLOGIES.
In *Proceedings of the Tenth International Conference on Enterprise Information Systems - ISAS*, pages 338-342
DOI: 10.5220/0001714303380342

2 FROM BUSINESS PROCESSES TO INTEGRATED DATA MODELS

According to Gruber (1995), ontology is an explicit representation of a conceptualization, and can be seen as a formal specification of concepts and terms of a domain. Ontologies define the rules that regulate the combination among the terms. However, the conceptual modeling of a domain through ontologies is a complex task (Guizzardi, 2005).

A domain can be represented through diverse perspectives: What? How? Where? Who? When? Why? (Smith & Welty, 2001). Thus, the representation of a domain makes use of diverse models for each perspective. These models may be built complementarily, and contribute to the agreement of the domain as a whole (Sowa & Zachman, 1992).

The process perspective (How?) focuses on the flow control representation, that is, the sequence of activities. It may be expressed using, for example, modeling languages such as Petri Nets (Keller & Teufel, 1998;) or event-driven process chains (EPC) (Scheer, 1997). By using the EPC language, the representation of a business process model encompasses several constructs. First, a business process is represented as a set of activities that are linked to one another according to certain execution logic (that is, a syntactically correct combination of and/or/xor connectors between activities). Second, activities in a process may be triggered by events. Third, it is possible to represent the set of resources that are produced and/or consumed by an activity, as well as input/output pieces of information that comes to/from the execution of an activity, and products that are delivered by each activity. Fourth, one may explicitly relate activities to business rules that constrain its execution. Finally, business process models often represent relationships between actors and activities (who are responsible for executing it, who must be informed about it, who is involved in its execution).

All the elements represented in a business process model contribute to increase the understanding of the domain of interest, and the business itself. Accordingly, the business process model can be defined as a set of combined views that allow a proper agreement on the business.

There are some works in the literature dealing with data models associated to business models. Bringel et al. (2004) state that the understanding of processes behaviour provides the means to reuse it, and adapt its organizational concepts.

Koschmider and Oberweis (2005) discuss process interoperability within organizations. In this context, the authors present an ontology for business process based on Petri nets. They explain that the extraction of ontological descriptions from business processes and mapping to the Petri net ontology should be done during the modeling process and is made automatically, not visible to the modeller.

Zhao et al. (2004) propose AKEM (Application Knowledge Engineering Methodology), which is a method to specify an ontology based on textual descriptions of a business process (using a structured natural language). The instrument for describing a semantic space of business in AKEM is a story in which the main elements are settings, characteristics, episodes and scenarios.

Those works deal with data models associated to business process models, highlighting the importance of establishing an explicit source of quality and representation of concepts. In our proposal we try to reach to a conceptual domain model by analyzing specific elements from the business process model. While AKEM is based only in textual descriptions of the process; our proposal explores the graphical characteristic of the EPC to obtain important information and create real links among those elements: activities, business concepts and supporting systems.

3 AN ONTOLOGY-DRIVEN APPROACH FOR DERIVING DATA MODELS FROM BUSINESS DATA

This work proposes an approach for deriving integrated logical data models from business process models. This derivation encompasses the elaboration of a domain ontology. The knowledge from the domain that should be represented by the ontology (concepts, relationships, axioms) is captured from the several elements existing in the process model. The proposed approach consists of 5 phases, described as follows.

Analysis of Glossary Terms. Terms and relationships are extracted from the definition of the glossary terms presented in the business process model. The analysis of each glossary term must consider its application context, that is, the set of all process activities that are associated to that glossary term.

Terms and relationships extraction is text-based and a linguistic activity. The ontology engineer should identify key words and sentences of semantic importance within the definition of a glossary term (e.g., "oil", "well"). The selected sentences are translated into a structured form of a binary relationship between terms (term1 relation term2) (e.g., "oil isExtractedFrom well").

For example, the "Oil Reservoir" concept is defined as an accumulation of "Fluids" located in a "Permoporose Stone". This definition derives the relationships "OilReservoir accumulates Fluid" and "Fluid isLocatedAt PermoporoseStone".

The set of extracted terms is then analyzed in order to eliminate possible redundancies.

Analysis of Sets of Information. This phase is responsible for identifying ontology terms, attributes and relationships from the sets of information that are consumed and/or produced by activities in a process model.

The extraction of ontological constructs based on the sets of information is simpler than in the previous phase, since sets of information are already structured elements. Each set of information is mapped into one of the constructs of the ontology, which is a matter of design rationale. Some works in the literature (Guizzardi, 2005), (Medeiros & Schwabe, 2007) discuss modeling guidelines that may help the ontology engineering in choosing the most adequate language construct to represent the domain of knowledge without loss of semantics.

Analysis of Product-like and Document-like Elements. This phase is responsible for identifying terms in the ontology from additional elements of the process model, such as products and documents generated by activities during its execution. Products and documents also define key concepts of the domain. Therefore, each product or document is analyzed to define terms of the ontology that may not be defined yet.

Analysis of Business Rules. Business Rules *guide* Business Processes, and may influence the behavior of people (in the case of an operative business rule) or their understanding of concepts (in the case of a structural rule). The different categories of business rules are (Wagner, 2005):

- Integrity rules, denoting constraints (e.g., Rule I1: "Each project must have one and only one project manager");
- Derivation rules, denoting conditions resulting in conclusions (e.g., Rule D1: "the production manager of the most productive well of the year receives a bonus of 0.01% of the production profit");
- Reaction rules, in the form <Event, Condition, Action, Alternative action, Post-condition> (e.g., Rule R1: "an invoice is received. If the invoice amount is more than $1,000 then a supervisor must approve it");
- Production rules, in the form <condition, action> (e.g., Rule P1: "if there are no defects in the valve then the valve is approved"); and
- Transformation rules, denoting change of state (e.g., Rule T1: "an employee's age can change from 30 to 31, but not from 31 to 30").

For the scope of this work, business rules are expressed informally, in natural language. Business rules definitions are parsed in order to define constructs in the ontology, according to the proposed guidelines below. The three first guidelines follow the ideas presented in (OMG, 2007) to relate structural business rules to concepts of the domain:

Guideline 1: If a structural rule uses universal quantification (e.g., "each" or "all") to propose a necessary characteristic of a concept, then the structural rule proposes that something is always true about all instances of the concept.

In this case, the referred concepts are translated into terms in the ontology (if they do not exist yet), and each characteristic of the concept is translated into an attribute of the concept, or a relationship between two concepts. For example, the integrity rule I1 generates two terms "project" and "projectManager", and a relationship (project "isManagedBy" projectManager).

Guideline 2: For each individual concept mentioned in the business rule definition, the instance of the individual concept exists.

In this case, each individual is translated into an ontology instance or property value. For example, take the derivation rule D1 and the following individual concepts (or ground facts):

- "John Doe was the production manager of the P1 well on 2006";
- "The P1 well was the most productive well during 2006", and
- "the production of the P1 well in 2006 resulted in a $1,000,000 profit".

The following constructs are defined in the ontology:

- "John Doe", as an instance of the "ProductionManager" term;
- "P1", as an instance of the "well" term; and
- "$1.000,000", as the value of the"wellYearProfit" property of the "well" term

Future queries may conclude that “John Doe received a bonus of $1.000”, due to the inference capabilities of ontology query languages.

Guideline 3: If a structural rule proposes something to be necessarily true, then the rule may generate either an instance or a property value in the ontology. For example, suppose the two business rules that follow:

- “the oil production estimative of a well is always verifiable”, and
- “a verification procedure for oil production estimative always exists”

The second rule follows logically from the first rule, and generates an instance “verificationProcedure” in the ontology, which is an individual concept.

Guideline 4: Structural rules may also derive axioms in the ontology. In the given examples, the following axioms could be defined in the ontology:

– From Rule I1:

```
{ forAll p, exists (m1,m2) | project(p),
   projectManager(m1), projectManager(m2),
   equalTo(m1,m2), manages (m1,p)}.
```

– From Rule D1:

```
{ forAll (m,w,a) |
    productionManager(m), well(w),
    mostProductiveOftheYear(w,y),
    wellProductionProfitOfTheYear(p,w,y),
    b = p *0.0001, receivesBonus(m,b)}.
```

– From Rule D1:

```
{ forAll i, exists s |
    invoice(i), invoiceReceived(i, TRUE),
    invoiceAmount(i,a), a > 1000,
    supervisor(s), approvedBy(i,s)}.
```

– From Rule P1:

```
{ forAll v | valve(v), numberOfDefects(v, 0),
    approved(v) }.
```

Generation of the Logical Data Model. The ontology is a representation of a semantically rich conceptual data model, and as so can be used for the derivation of logical data models. The benefits of deriving logical elements from ontological constructs, instead of from conventional conceptual models, are that some inconsistencies could be avoided. For instance, in the domain of Education, the N:M relationship between “UniversityStudents” and “Advisors” denotes that each student can be advised by more than one advisor and each advisor can advise more than one student during his career. However, it is not clear which of the following real scenarios occurs in reality: (a) an advisor can advise more than one student simultaneously; (b) two students can work together on the same project, being advised by the same advisor; or (c) more than one teacher can advise the project conducted by a student. These situations may not be distinctly represented using a conventional conceptual modeling language, although each of them would ideally generate a distinct logical data structure in the relational model. There is a need to represent specific properties of the relationship between Student and Advisor, which may be done in the domain ontology, so as to derive distinct logical database models for each scenario, thus avoiding integration problems.

4 CONCLUSIONS

This paper addresses data integration common problems: inconsistency and redundancy within organization’s databases where business concepts are not always clear and shared among professionals. We propose a method in which the domain ontology is extracted systematically from a detailed representation of business processes, and provides a basis for generating logical data models.

By using our approach, the generated logical data model will avoid data integration problems, since it will be derived from a rich and shared representation of the domain. We evaluated the proposal through a case study, which was carried out in a real and very complex domain of a Petroleum company, in which data integration was defined as a goal. Our results shown that business process models helps to understand and to reach to a consensus regarding the semantics of the concepts of the domain.

As a future work we intend to accomplish case studies in out other domains in order to validate our results. Besides we are studying the possibility of automate the method proposed using text analysis and applying techniques to explore formal relationships in the process model.

REFERENCES

Bringel, H., Caetano, A., Tribolet, J., 2004. Business Process Modeling Towards Data Quality Assurance. Proc 6th Intl Conf Enterprise Information Systems, ICEIS 2004, Portugal.

Gruber, T. R., 1995. Toward Principles for the Design of Ontologies Used for Knowledge Sharing, International Journal of Human and Computer Studies, 43 (5/6), 907-928.

Guizzardi, G., 2005. Ontological Foundations for Structural Conceptual Models, PhD Thesis, U Twente, The Netherlands

Keller, G., Teufel. T., 1998. SAP R/3 Process Oriented Implementation, Addison-Wesley, Reading MA.

Koschmider, A., Oberweis, A., 2005. Ontology based business process description, in J. Castro & E.

Teniente, eds, Proceedings of the CAiSE-05 Workshops, Lecture Notes in Computer Science, Springer, Porto, Portugal, pp. 321—333.

Medeiros, A., Schwabe, D., 2007. Using Metamodels Formal Semantics for representing Rationale during the Software Design , 2nd Workshop on Ontologies and Metamodels in Software and Data Engineering, João Pessoa, Brazil.

Noy, N.F., MacGuinness, D.L., 2001. Ontology Development 101: A guide to Creating Your First Ontology, Semantic Web Working Symposium.

OMG, 2007. Semantics of Business Vocabulary and Business Rules (SBVR) Specification, available at http://www.omg.org/cgi-bin/doc?dtc/2007-06-06.

Scheer, A., 1999. ARIS – Business Process Frameworks, Springer

Smith, B., Welty, C., 2001. Ontology: Towards a new synthesis, In: Welty, C. e Smith, B., eds., Formal Ontology in Information Systems. iii-x, ACM Press.

Sowa, F., Zachman, J.A., 1992. Extending and formalizing the framework for information systems architecture, IBM Systems Journal 31(3), 590—616.

Wagner, G., 2005. Rule Modeling and Markup in Reasoning Web, 3564 ed, N. Eisinger and J. Maluszynski, Eds. Msida, Malta: Springer, 2005, pp. 251-274

Zhao, G., Gao, Y., Meersman, R., 2004. Ontology-Based Approach to Business Modeling. Proceedings of the International Conference of Knowledge Engineering and Decision Support (ICKEDS2004).