Improving Business Process Model Quality Using Domain Ontologies

Samira Si-Said Cherfi · Sarah Ayad ·
Isabelle Comyn-Wattiau

Received: 1 September 2012 / Revised: 27 November 2012 / Accepted: 20 December 2012 / Published online: 4 June 2013
© Springer-Verlag Berlin Heidelberg 2013

Abstract This paper addresses the issue of improving quality of business process (BP) models by exploiting domain knowledge. Indeed, business process models reflect the business processes of companies. The success of these processes has a direct and undeniable impact on business operations success. Managing them through their underlying models helps improving their effectiveness, consistency, and transparency. BP modeling aims at a better understanding of processes, allowing deciders to achieve strategic goals of the company. However, several studies from the literature showed that in experienced system analysts often produce low-level quality. This situation is partly due to lack of domain knowledge. In this paper, we propose to support this modeling effort with an approach that uses domain knowledge to improve the semantic quality of BP models. We suggest to use ontologies as a mean to capture domain knowledge and meta-modeling techniques to deal with BP models independently of languages in which they are expressed. Our contribution is threefold: (1) the meta-models describing both a domain ontology and a BP model are described, (2) the alignment between the concepts of both meta-models is defined and illustrated, (3) a set of Object Constraint Language mapping rules is provided. A simple case study illustrates the process.

Keywords Domain knowledge · Domain ontology · Semantic quality · Business process modeling · Quality improvement

S. Si-Said Cherfi (✉) · S. Ayad
CEDRIC-CNAM, 292 Rue Saint Martin,
75141 Paris Cedex 03, France
e-mail: samira.cherfi@cnam.fr

I. Comyn-Wattiau
CEDRIC-CNAM and ESSEC Business School, Cergy-Pontoise, France

1 Introduction

Nowadays, it is widely recognized that conceptual models play a crucial role in information system's development. As they are a high level abstraction of the represented reality, they constitute a vehicle for communication, provide a comprehensive documentation, and are the basis for the implementation and the evolution of the developed system as well [14,31]. However, modeling is the intellectual activity implying stakeholders with different skills, knowledge, and experiences. Several studies [19,26] show that quality of models produced by novice modelers are less complete and lack flexibility and innovation. This is essentially due to their incomplete domain knowledge and the lacks of expertise in methods used.

Business process (BP) models are conceptual models supposed to provide a complete description of the underlying business processes (BPs). Consequently, companies are today aware of the undeniable impact of a better tuning of business processes on the effectiveness, consistency, and transparency of their business operations. This tuning requires a better understanding and an effective management of BP. However, to achieve the expected benefits it is necessary to rethink the approach of designing these processes. BP modeling is a prerequisite. It is now considered as an engineering activity aiming at providing the actors with a better understanding of the processes in which they are involved. But BP modeling is a difficult task. It needs to be performed by trained experts.

And, what about quality? Quality can be defined as the total of properties and characteristics of a product or service that are relevant for satisfying specific and obvious requirements [11]. The business process modeling approaches share many similarities with conceptual modeling activities, but they are much more complex [30]. Indeed, a business process

model captures a dynamic vision of the system through activities descriptions, generally done at a low level of abstraction. The modeler has the difficult task to describe these processes at a more general level of abstraction requiring a good understanding of domain knowledge. This is the reason why the activity of modeling BP requires a high degree of pragmatic expertise generally based on empirical rules and heuristics. The latter are difficult to formalize and to share. Commercial tools for business process modeling activities mainly focus on the accuracy of models based on a set of syntactic criteria imposed by the notation and provide little or no guide to guarantee the quality of produced models.

We propose to assist the modeling activity with a quality centered approach that aims to exploit the domain knowledge. The domain knowledge in Information Systems discipline refers to knowledge provided by both methods and application domain [17]. In our approach we propose to refer to domain ontologies knowledge with alignment rules in order to identify similarities between BP models and domain ontologies elements. The aim is to improve the semantic completeness and expressiveness of BP models according to domain knowledge contained in the ontologies.

This paper is organized as follows. A state of the art is described briefly in Sect. 2. Section 3 introduces a motivating example that will serve to illustrate our solution. The overall approach is broadly described in Sect. 4. The meta-models structuring both BP models and domain ontologies are described in detail in Sect. 5. Section 6 is dedicated to alignment rules. The proposed approach is applied on the motivating example in Sect. 7. Finally Sect. 8 concludes and describes future research.

2 State of the Art

A business process is a set of related activities that transform an input to create an output with added values [15]. Experts in information systems and professionals agree that the success of a company depends particularly on a good understanding of its business processes [5]. Modeling BP is a useful mean to facilitate their comprehensibility. To make a business process model understandable, reliable, and reusable it is important to ensure its quality. Several approaches that work in this direction are described in the literature. We have classified them into three categories: (1) approaches focused on improving BP methods of analysis and design, (2) process quality measurement, and (3) process model quality measurement. In the first category the approaches are intended to provide advice and best practices to improve the quality of models. The hypothesis is that improving the process development impacts positively the quality of available products. As an illustration, we can mention [7] where the authors propose a set of guides to improve various characteristics of a process

model such as clarity, comprehensibility, or accuracy (correctness). Other authors focus on improving the comprehensibility of models by providing naming rules, documentation, and use of icons or symbols graphs [21]. Other approaches, such as [3], propose a set of best practices encapsulated in reusable and applicable patterns depending on the context.

The second category is concerned with the quality level of business processes and their execution. In this family, we categorize simulation and control of process such as in [13] where the authors present a set of simulation tools for business process evaluation. Others focus on the verification of certain characteristics, when executing the process. In [2] for example, the authors present and discuss several techniques for the analysis of processes during execution such as verification, or for the discovery of a process (process mining), etc.

The approach we propose belongs to the third category that addresses the quality from the point of view of its evaluation and improvement. Process model quality has been investigated in different disciplines. Consequently, a variety of standards have been introduced to define, manage, monitor, and improve that quality. In [30], the authors present a typology and an overall view of the business process model metrics. They mention the five most important measures: coupling, cohesion, complexity, modularity, and finally the size. An approach based on goal-question-metric method (GQM) is proposed in [11] to help finding, among a set of quality characteristics, those that are relevant in a given framework and to deduce how to measure them. One of the characteristics that has been the subject of several proposals is the complexity [6, 12, 9]. However, these studies are based primarily on structural characteristics of processes and their models. Complexity, as an example of structural characteristics, has a direct impact on comprehensibility and maintainability. Aguilar et al. [4] adapted the complexity metrics from software engineering to evaluate the complexity of BP models. They conducted a controlled experiment to study the impact of complex process models on their maintainability. The work presented in [22] aimed at finding out the factors that impact the understandability of process models.

In conclusion, our analysis of the state of the art led us to argue that the quality of BP models is mainly addressed in terms of structural and syntactic aspects and rarely in terms of semantics. In the remainder of this paper, we present our approach, which aims to go a step forward into a semantic quality-based approach of BP models.

3 A Motivating Example

Before detailing our solution we first introduce an example of business process model that will serve to illustrate the approach. Figure 1 shows an example of model of a ‘Mission order’ business process case study. It represents the business

Fig. 1 An example of a BP model for processing missions' orders

process followed by a researcher/employee in our university who plans to attend a conference or to participate to an exchange program. In each case he/she should fill an official 'Mission Order'. This means in particular that he/she is covered by insurance, that the university pays for the plane/train ticket, that the employee gets some money in advance for his/her fees and that he/she is reimbursed after the mission. The employee has first to get an authorization from his/her service/laboratory director/administrator. The administrator analyzes the request. Based on his/her decision, the employee fills a form called mission order (MO), sends it to the financial service and at the same time carries out mission formalities (book a ticket, a hotel, etc.). When he/she comes back after his/her travel, he/she provides the financial service with expense accounts. The financial service calculates the reimbursement costs only if the employee provides mission costs proofs (tickets, bills etc.). Finally, the BP is terminated by the bank transfer activity. The BP model expressed using BPMN [1] is presented at Fig. 1.

4 An Overview of the Approach for Semantic Quality Improvement

Modeling activity in general and BP modeling in particular are creative activities conducted by modelers using a given notation or modeling language. The result is of course highly dependent on the modeler experience in the notation practice, on his/her interpretation of the reality, and on the decisions

he/she makes regarding the choice of concepts and details to be modeled. This explains the fact that several correct but different models can usually be generated from the same reality. However, these models are supposed to be faithful representations of the reality. Thus, the definition of quality requirements for these models is, in fact, a mean to evaluate this modeling activity and ensure a better result. Many factors may be defined to characterize this quality. The semantic quality measures the degree of correspondence between the model and the domain. The semantic quality is related to both completeness and validity of the models; here the BP models [18]. To improve the quality of the models produced, several approaches are possible: assistance in the development process phase by generic methodological guides from experience, measurement and improvement of the specifications quality, reusing approved specifications fragments, etc. In this paper, we propose to exploit domain knowledge, which is supposed to reflect the knowledge shared by a community of actors, in order to improve the quality of process models.

The approach is based on the process described at Fig. 2. The entry point is composed of a business process model under construction and the associated domain knowledge provided by the process modeler. The process involves three steps described in the following sections.

4.1 Identifying Model-Ontology Similarities

In the first step, the approach consists in discovering the mappings between business process model and domain ontology

Fig. 2 The overall approach for quality evaluation and improvement

elements. This is supported by the alignment rules that should be generic and independent of both the BP modeling notation and the ontology implementation language. To ensure this characteristics, two meta-models, namely a BP meta-model and an ontology meta-model, have been defined. They are presented in detail in Sect. 5.

Indeed, there exist several notations for modeling BP models, each having its own vocabulary. Writing the rules at the notation level implies specifying the same rules several times to take into account the variety of vocabulary used to point out the same concept. The alignment rules aim to identify similarities between the process model elements and the domain ontology concepts. Once these similarities are identified, they serve as input for both semantic quality evaluation and improvement activities. In this paper, we mainly focus on this alignment activity.

4.2 Evaluating Semantic Quality

Semantic quality expresses the degree of correspondence between the information expressed within a model and the domain that is modeled. It helps evaluating both the validity and the completeness of the model [18]. The validity checks that all the statements of the model are correct and related to the domain. The completeness, as for it, verifies that all the pertinent statements from the domain are captured by the model. In order to evaluate the semantic quality we have identified a set of what we call quality deficiencies such as incompleteness and ambiguity. These deficiencies result from modeling choices producing mod-

els that do not cover the intended requirements or with low expressiveness. Such models lead to inadequate systems because of their incompleteness or their misunderstanding by developers during their implementation. Once a similarity has been identified between a BP model element, let it be *bpmi* and an element from the domain ontology *doj*, our approach exploits the knowledge from the domain ontology related to *doj* to detect and measure semantic quality deficiencies according to quality metrics we have defined.

4.3 Quality Improvement

The quality improvement activity consists in suggesting to the analyst or the quality expert a set of improvement guidelines to improve the quality of their models. Again, this step uses the domain knowledge to generate improvement actions. This means that the completeness and even the relevance of these guides rely partly on the quality of the domain ontology. For example, if the approach identified a similarity between *bpmi* a BP model element—and *doj*—an element from the domain ontology—and the domain ontology describes a relationship between *doj* and *dok* (an other element from the ontology), then our approach will propose an enrichment action on the BP model based on the relationship between *doj* and *dok*.

The analyst, with the help of the suggestions made by the approach, adds/removes/modifies elements from the initial model that evolves. However, quality improvement is never done in one shot and this is why the approach is iterative and

Fig. 3 Business process meta-model

incremental. The further sections detail the different steps of the proposed approach.

5 Ontology and Process Model Meta-Models

Our approach relies on the assumption that a domain ontology and a BP model share a common knowledge. It relies on alignment techniques to identify the pieces of shared knowledge. To ensure the generality of these rules, we have chosen to define them at a meta-modeling level. Hence, the first contribution is the definition of meta-models representing ontologies and BP models.

5.1 Business Process Meta-Model

There are several advantages of defining such a meta-model. First, the meta-model provides a synthetic representation of concepts used independently of specific notations helping in the understandability of models. Second, as the BP models are not produced by our approach and are the result of an independent modeling process they are not supposed to be expressed in the same notation. The usage of a meta model will allow us to express the several models as instances of the same meta-model enabling the application of our approach on models from several notations. Moreover, this characteristic allows us to define mapping rules for each couple (BP modeling notation, ontology language) in a generic manner (on the meta-model level) and be able to apply them at a specific level for each model. Finally, since we consider that domain

knowledge contains also knowledge embedded in methods and consequently in notations, we will use meta-models to integrate completeness, validation, and correctness rules defined by BP notations to enrich our domain knowledge.

This last part of the work is under investigation. The meta-model defined in this section (Fig. 3) is a synthesis of a selection of concepts proposed by several authors according to several notations and more specifically the contribution presented in [20, 23]. We have restricted our meta-model to concepts required by the approach and it is consequently not as detailed as models constructed for method engineering needs for example.

The meta-model expresses a business process model as composed of flows of objects and connectors. A flow object can be an event, an activity, or a gateway. An event that occurs impacts the progress of a process. The events could be of three types: initial, intermediate, and final. An activity can be an atomic task if it is not decomposable or a process if it is complex and has a visible structure. A gateway is a mechanism able to manage the convergence or divergence of activities flows. A connecting element can be an association, a sequence, or a message flow. An association is a simple link between two concepts. The sequence flow defines an execution order of activities. A message flow is used to represent exchange of information between two participants in the process. Activities refer to resources. A resource encompasses abstract concepts such as the human agent responsible for execution of the activity and also information produced or consumed by it. The exact role of the resource in the process is explained by the concept of role.

Fig. 4 Ontology meta-model
(an extract)

Examples of the meta-model instances are ‘Employee’ and ‘Responsible’ as People Resource concepts. A sequence of the processes ‘Request authorization’ and ‘Analyze request’ led to a process divergence (Gateway). The two processes ‘Establish MO’ and ‘Carry out mission formalities’ may be executed in parallel.

5.2 Ontology Meta-model

The ontology meta-model ensures the independence of domain knowledge representation from the language used for its implementation. There exists in the literature several contributions on ontology meta-modeling. The authors in [29] introduced simple concepts and constructors (negation, conjunction, disjunction) to define complex concepts. They also defined several relationships including inheritance links, instantiation, and constraints. In [16] five types of concepts have been proposed to represent the functional requirements (function, object, and environment) and non-functional requirements (constraints, quality).

In our approach, we consider an ontology as a set of classes and relationships. This vision is largely adopted and is suitable for our approach as it gathers concepts and relation details that help us to align them with the BP model concepts. We distinguish between three types of concepts of type class: actor, action, and artifact (see Fig. 4). An actor is an independent entity, able to perform actions. An action represents the execution of an action. An artifact is an inanimate object incapable of performing an action. An artifact may represent an information or an abstract concept. However, most meta-models take into account two kinds of relationships, namely inheritance and structural relationships. Figure 4 illustrates some concepts of the ontology meta-model we have adopted. For the needs of our approach we adapted the classification of relationships proposed in [25,27], which has been initially defined to analyze semantics of relationships within a relational database. This classification offers several types of relationships allowing us to characterize precisely the nature of links between concepts.

Relations are first decomposed into three categories: *Status* represents relationships that may be structural (inheritance, composition, instantiation, etc.), influence (own, control, create, destroy, etc.), or temporal (follow, require, etc.). *Change of status* reveals the occurrence of remarkable events. This type of relationship is primarily used to express the interdependence of status in the life cycle of an entity. *Interaction* represents short-term relationships between entities. Several semantic relations are defined for interactions such as communication, observation, execution, etc. Figure 5 shows an example of a domain ontology. This is an extract from the ontology ‘mission plan’. Actors ‘Internal staff’ and ‘External staff’ are linked to the actor ‘Staff’ by an Is-A relationship of type structural (status). Actors ‘secretary’ and ‘missionary’ are related to the actor ‘internal staff’ by an is-a relation. In addition the actor ‘missionary’ is related to the action ‘Formalities management’ by a control relation of type influence (status), similarly to the actor ‘external’ and the action ‘authorization request’. Moreover, ‘Formalities management’ action is related to ‘return mission’ action with a temporal relationship indicating that managing formalities precedes a return mission. The example indicates also that ‘Hosting costs’ and ‘Travel costs’ abstract concepts are linked to ‘Return mission’ action by observation relationships, meaning that the action performs no changes on the abstract concept.

6 Mapping Process Model and Ontology Meta-Models

Thanks to the precise categorization of PM concepts in both ontology and process model meta-models we are able to pre-define some concepts correspondences enabling the mapping of the domain ontology concepts with the PM concepts. We have defined two kinds of mapping, namely type-based mapping and semantics-based mapping.

6.1 Type-Based Mapping Rules

This mapping involves the types of concepts in order to establish correspondences between the concepts at the meta-

Fig. 5 Instantiation of the ontology meta-model **Table 1** Concepts alignment

BP model meta-model	Domain ontology
People resource	Actor
Abstract resource	Abstract
Information resource	Knowledge
Process/activity	Action

Table 2 Relationships alignment

BP model connectors	Domain ontology relations
Sequence flow	Temporal
Message flow	Communication
Role	Transfer Execution Manipulation Observation Influence

level. These correspondences allow reconciliation based on the types of concepts independently of their meaning. These rules are essential to avoid typing errors. An extract of pre-defined meta-model concept mappings is given in Table 1

Similarly, we have established mappings between meta-model relations of the BP model and of the ontology. The result is synthesized in Table 2.

The second type of mapping, presented in the following section, is richer, being based on the semantics of concepts.

6.2 Semantics-Based Mapping Rules

Based on meta-models presented above, we developed a set of matching rules. These rules allow mapping the ontology concepts with elements from process models. They compute similarity distances and are written in Object Constraint Language (OCL) [8]. The similarity computation is based on the names, the synonyms, and keywords associated to ontology concepts. It exploits WordNet and distance computation algorithms from the literature such as Resnik information content [28], Wu and Palmer path length [32], and Pathwardhan and Pedersen context vectors [24].

There are four classes of matching rules. The rules are all defined as functions having as input one or several BP model concepts and returning one or several concepts from the domain ontology:

- *Syntactic equivalence* returns the ontology concepts which are syntactically equivalent (have the same names) to a BP model element.
- *Synonymy* returns a set of ontology concepts that are synonyms of the BP model concept. The synonymy value is calculated by comparing the existence of common names or synonyms based on Wordnet [10].
- *More general* returns the ontology concepts having a superiority relationship (also called hyperonymy or IS-A relationship) with a concept from the ontology already

detected as synonym or syntactically equivalent to a BP model element.

- *More Specific* returns the ontology concepts having an inferiority relationship with a concept from the ontology detected as synonym or syntactically equivalent to the BP model element. These classes of rules are instantiated for each of the concepts of the BP meta-model. For each class, an example of instantiated rule for the PeopleResource BP concept is given below.

PR_syntactic_equivalence This function returns all the actors from the ontology that are equivalent to a people resource (from the process model). This rule works when the names in the BP model and in the ontology are identical.

```
PR_syntactic_equivalence(p : PeopleResource)
: Set(Actor)
post : Actor.allInstances() → forAll(a : Actor |
a.name = p.name) implies
PR_syntactic_equivalence → includes(a)
and result = PR_syntactic_equivalence
end
```

PR_synonymy This function returns all the actors from the ontology that are synonyms of a PeopleResource (from the process model). The synonymy closeness is computed by comparing the existence of common names or common synonyms for both terms. In the ontology, all the terms are enriched by sets of synonyms extracted from Wordnet.

```
PR_synonymy(p : PeopleResource) : Set(Actor)
post : Actor.allInstances() → forAll(a : Actor |
(a.synonym → for exists(b : Actor | p.name =
b.name)) implies PR_synonymy → for includes(a)
and result = PR_synonymy)
end
```

PR_more_general This function returns the list of the more general actors of the ontology, i.e., there is a relationship of superiority (also called hyperonymy or IS-A relationship) between the people resource in parameter and the ontology actor in return.

```
PR_more_general(p : PeopleResource) : Set(Actor)
post :
if (Relation.allInstances() → exists(r : Relation |
r.type = #is - a and r.domain = a and
PR_synonymy(p) → include(a) then
PR_more_general → includes(r.range) and
return = PR_more_general
endif
end
```

PR_more_specific This function returns the list of the most specific actors from the ontology, i.e., there is a relation of inferiority between the people resource and the ontology actor in return

```
PR_more_specific(p : PeopleResource) : Set(Actor)
post :
if (Relation.allInstances() → exists(r : Relation | r.type =
#is - a and r.range = a and PR_synonymy(p) →
include(a) then PR_more_specific → includes(r.domain)
and return = PR_more_specific
endif
end
```

6.3 Quality Defects Detection

First, similarities have to be identified between a BP model element, let it be *bpm* and an element from the domain ontology *o*. Our approach exploits the knowledge from the domain ontology related to *o* to detect and measure semantic quality deficiencies. In order to exploit the knowledge related to *o* we use the mappings identified in Sect. 6.2. Second, we have identified a set of what we call quality deficiencies. These deficiencies result from modeling choices producing models that do not cover the intended requirements or with low expressiveness.

- *Ambiguity* results from using different names and constructs to express the same reality. This makes models unclear and generates confusion when trying to understand them. For example, consider a model providing a property ‘telephone number’. If the reality corresponding to this model allows several categories of telephone number such as personal, office, home, and cellular telephone numbers having each its own usage then the property ‘telephone number’ from the model is ambiguous as it represents several realities. For quality evaluation we use the word clarity (as desired characteristics) instead of ambiguity which is a non-quality property. To compute the clarity of *p* a PeopleResource element from the BP

model one of the metrics is

$$PR_clarity(p) = 1/|PR_syntactic_equivalence(p)|$$

if $|PR_syntactic_equivalence(p)| \neq 0$
meaningless otherwise

Where $|PR_syntactic_equivalence(p)|$ is the number of ontology concepts returned by syntactic equivalence rule

The more the concept has synonyms, the more ambiguous it is. When the value of clarity equals 1 for p, a given BP model element, this means that there is only one corresponding concept from the domain ontology and thus p is not ambiguous. The value of clarity decreases when the number of corresponding elements in the ontology increases. When the value of clarity is meaningless, this means that there are no corresponding elements found for p. In this case, p can be a meaningless state. A validation from the modeler has to be performed since the domain ontology is not the exclusive source of knowledge.

- *Completeness* is related to an incomplete representation of the real world. This incompleteness can result from the complexity of concepts for which only a sub-set of the description is captured within the process model or to a lack of domain knowledge of the analysts or both of them. For example, if a BP model considers only missions with train tickets' cost reimbursement whereas the reality includes also missions with hotel costs then the BP model is incomplete. It is an example of metrics that helps detecting activities (tasks and processes) that could have incomplete input definition.

$$Input_completeness(a) = \frac{|required(oc) \cap input(a)|}{|required(oc)|}$$

Where a is an activity element from the BP and $oc \in A_synonymy(c)$.

$|input(a)|$ computes the number of resources required by activity a . oc designates an ontology concept validated by the synonymy function $A_synonymy$ defined for activities. $|required(oc)|$ computes the number of concepts from the ontology related to oc by a "requires" relationship.

For example, in mission costs reimbursement case study the ontology indicates that the reimbursement of a mission 'requires' tickets and hotel bill. Consequently, the *Input_incompleteness* metric will compute a 0.5 values since only one input is provided for the activity from the BP model whereas two concepts are expected by the ontology.

- *Abstraction level* is related to the use of the suitable level of generality. Indeed, in some cases, using concepts that are general decreases the size of models and thus helps in their understanding. For example, 'analyse the request' activity is a general activity that could hide more precise and alternative request treatments. On the contrary, using very specialized terms may decrease the understandability of the models. Indeed, splitting excessively steps such as 'check the hotel reservation invoice', 'check the plan ticket', 'check the registration invoice' activities could be replaced by 'check the documents'. The relevant choice of an abstraction level depends on several factors among which we can mention the nature of audience (developers or users), the objective of the model (explanation or implementation), etc.
- *Meaningless states* correspond to states and constructs from the models for which no correspondence is found in the ontology. This decreases the relevance of models and has an impact on its intelligibility. In fact, the analyst may add an activity which does not represent a domain-related role, in other words it represents meaningless data.

6.4 Quality Defects Correction

Third, the quality improvement activity consists in suggesting to the analyst or to the quality expert a set of improvement guidelines to improve the quality of their models.

- *Correcting ambiguity defects* consists in replacing a BP model element by a more adequate one. This one is chosen by the analyst from a list of domain ontology synonyms computed by the approach. As we detect ambiguity when we have a concept in the BP model that has more than one corresponding in the ontology, based on the above example, the analyst may replace the telephone number by one of the ontology concepts, such as office or home telephone.
- *Correcting incompleteness defects* In case of incompleteness, the analyst can rely on the knowledge provided by the ontology to complete the missing parts of the model. For example, an activity in the ontology requires abstract/knowledge concepts as a result. The analyst may complete his model by these concepts, for example each activity should have a human resource responsible of its execution. If the approach detects that an activity has no assigned responsible, it is able to provide the user with suggestions based on the domain ontology.
- *Correcting the abstraction level* Likewise, the user can rely on the knowledge provided by the ontology (hypernyms/hyponyms) to choose the adequate abstraction level of the concept improving the model's comprehensibility.

- *Correcting the meaningless states* occurs when a concept does not match any of the ontology ones. This could mean that the BP model element is out of the scope or that it is not named in a comprehensible way for the domain vocabulary.

7 Application to the Example

To illustrate the alignment activity of our approach, we consider the example of ‘mission order’ process depicted in Fig. 1. An excerpt from the ‘mission plan’ domain ontology is given in Fig. 5. As mentioned above, all the semantic rules are based on type mapping rules, i.e., equivalence takes place between concepts that are not only syntactically equivalent but also type mapped.

7.1 Similarities and Quality Defect Detection

The first step applies mapping rules between the BP model and the domain ontology of Fig. 4. Based on the type mappings, our prototype computes a list of action/actors for each activity a human resource present in the BP model. These mappings are refined based on the semantic mapping rules. The result is a list of synonyms/hypernyms/hyponyms given in Table 3. Notice that the mapping is currently a word-based mapping. We plan to extend the similarity distance definition to take into account sentences. This will increase the relevancy of results obtained in Table 3.

Regarding the quality evaluation, the more an element from BP model has synonyms the more it is ambiguous. Thus, based on results listed in Table 3, we conclude that ‘Do the transfer’ activity is less ambiguous than ‘Mission order request’ activity since the number of synonyms from the domain found for the first activity is smaller than the number of synonyms found for the second one. Furthermore, based on the knowledge provided by the ontology, some elements from the BP model are more general or less

Table 3 Extract of synonyms’ results

BP activity	Action synonyms
Mission order request	Ask for assignment Call for a mission An operation command Postulate to a work mission Fill for a mission Fill for a foreign mission Fill for a local mission Supply for reimbursement
Provide mission costs details for reimbursement	Provide the total spent Provide the compensation paid
Do the transfer	Reassign Offer a deal

general than the concepts defined in the ontology. For example, the activity ‘Analyse request’ has many hyponyms as shown in Table 4. The user has to decide to maintain the generality level defined in the BP model or to change it for a more precise description by choosing one of the hyponyms. This could generate other impacts. Changing an activity may imply updating the actor responsible of it and/or change the information resources required or produced by the activity, etc. These changes are deduced from the ontology. An extract of computed hyponyms are shown in Tables 4 and 5 respectively.

In addition, concepts that are more general are suggested to each activity/human resource, for example ‘Administrator’ can be replaced by more generic concepts such as ‘Decision maker’ or ‘Executive’. The approach proposes the several alternatives and the analyst has to decide about the most appropriate choice.

Finally, the context provides knowledge allowing the BP model enrichment and/or completion. This context is defined through the keywords and relationships related to domain ontology concepts. The related concepts returned by the function and proposed to the user may help enriching the model.

Table 4 Extract of hyponyms’ results

BP activity	Hyponyms
Analyse request	Analyze the demand Examine the demand Break down the demand Correct the demand Review the demand
Carry out mission formalities	Fetch the mission order Open mission recorder Approve order
Provide mission costs details for reimbursement	Provide in France mission costs details Provide in Europe mission costs details Support with business expense Provide prices Provide costs
Estimate costs	Calculate compensation Estimate compensation
Do the transfer	Deliver Lend Convey

Table 5 Extract of hypernyms’ results

BP resources	human	Hypernyms
Employee		Worker Administrator Decision maker Executive
Financial Service		Financial department Supplier Contractor

These concepts are ontology concepts related to the BP model concept.

Based on the results shown in Table 6, time delay concept can be related to ‘establish MO’ activity. Time delay is an important condition in any administrative file request. As a result the user can add a timer and information about mission order time delay in his/her process. Also the activity ‘Analyse request’ is performed by manipulating the mission record so that it has to be an output sent to the administrator. Thus, the example above illustrates how the meta-models, the mappings, and the alignment rules allow the BP model designer to incrementally enrich his/her model.

7.2 Improving Detected Quality Defects

The quality improvement activity consists in suggesting to the analyst a set of improvement guidelines to increase the quality of their models.

- *Correcting ambiguity defects* The ambiguity hampers the possibility to decide whether the statement from the model is meaningful according to the domain. In fact, ‘Mission order request’ activity does not match a domain-specific contract. It is an application form to fill. Thus, the verb ‘fill for’ is more adequate and the mission order is replaced by more precise activities ‘Fill for a foreign mission’ and ‘Fill for a local mission’ activities.
- *Correcting abstraction level defects* provides advices on the generality discourse level to use in order to increase models understandability.

For example, ‘analyse request’ activity is a general activity that may skip a lot of detailed activities. Thus, based on the hyponyms provided by the ontology, the analyst can replace it by ‘Examine the demand’, ‘Review the demand’ and ‘Correct the demand’ activities provided by the ontology. In other cases, using very specialized terms may decrease the understandability of the models. For example, the actor ‘administrator’ can be replaced by

‘decision maker’.

- *Correcting incompleteness defects* could be done by enriching the description of existing concepts or by adding new concepts using their context (related concepts).

For example, we can see in Table 6, the activity ‘establish MO’ can be related to mission order time delay. So a timer is added as output for this activity. And ‘carry out mission formalities’ is enriched by adding abstract resources’ mission program. In addition, ‘Do transfer’ is related semantically to the concept ‘BAD:Bank Account Detail’. This led the analyst to add an activity ‘ask for BAD’ by exploiting this knowledge.

The improved process model is sketched in Fig. 6. The changes are highlighted by squares.

8 Conclusion

The quality of business process models is a hot topic for both researchers and practitioners. Many studies showed that the quality of produced models depends highly on the degree of expertise of the modelers. Moreover, modeling activities are practiced by a significant number of non-experts including IT professionals. One of the reasons impacting the quality of produced models is the lack of domain knowledge covering both knowledge about the methods and notations used as well as application domain knowledge.

In this paper, we tried to propose a solution aiming to exploit application domain knowledge for the improvement of BP models. Our approach considers domain ontologies that are produced in several disciplines (web services, health care, administrative processes, etc.) to improve the semantic quality of BP models. We have defined BP model and ontology meta-models in order to build a uniform description of both process models and domain ontologies. We have then defined an alignment process using both type- and semantics-based mappings to detect similarities between concepts from the BP models and domain ontologies. The results serve as an

Table 6 Extract of related concepts

BP model activity	Ontology action	Related concept	
		Concept	Relationship
Establish MO	Establish Mission Order	Documents required (abstract) Time delay (knowledge)	Assigned to
Reserve flight and hotel	Carry out mission formalities	Flight reservation (abstract) Ticket reservation (abstract)	Requires
Calculate reimburse costs	Estimate costs	Commission (knowledge) invoice	Requires
Analyse Request	Analyse The request	Mission (abstract) Record (abstract)	Manipulates
Do the transfer	Transfer	Bank Account Detail (BAD)	Requires

Fig. 6 The BP model resulting from our quality improvement process

input for the semantic quality evaluation and improvement processes.

The validation of the approach is ongoing. It is being conducted with 20 undergraduate students taking a course on UML activity diagrams. To ensure equal domain knowledge of the participants, the case study deals with diabetes care medical practices unknown by the participants. The validation process aims to verify the impact of domain knowledge on the quality of produced process models. In order to facilitate the experiment, domain knowledge is provided using natural language and structured texts, which is easier to understand by the students than ontologies. However, we need ontologies to manage knowledge easier than unstructured or semi-structured texts for a tool-supported approach.

The experiment is being conducted according to the following four phases:

- (1) the participants have to establish, based on a problem statement, an activity diagram aiming at representing a business process model for a type 2 diabetes care.
- (2) the models produced by the students are evaluated by a professor, expert in business process modelling and aware of the related domain knowledge, according to quality criteria addressed by the approach (completeness, clarity, etc.).
- (3) the participants are provided with domain knowledge described by a clinical pathway related to type 2 diabetes care. They are invited to detect and to correct qual-

ity deficiencies of their own models using this domain knowledge.

- (4) Finally, the revised models are, once again, evaluated by a professor.

Steps (2) and (3) are conducted in parallel. At the current stage, the first step of the experiment is being completed as we could not start before ending the training on UML activity modelling.

Regarding the alignment rules they need to be completed to cover all the kinds of concepts and relationships semantics.

The approach suffers from some limits. The main one is the fact that the domain knowledge in hand is not necessarily complete. This is the reason why we propose to enrich the sources of domain knowledge with requirements and domain models. Further research will include the enrichment of quality defects detection rules and the development of a metrics suite for semantic quality evaluation. We are also developing a prototype implementing the proposed approach for BP model quality evaluation and improvement. There is also need for more validation effort by conducting real-life case studies with practitioners.

References

1. Business Process Modeling Notation (BPMN) Version 1.2. Technical report (2009). <http://www.omg.org/spec/BPMN/1.2/PDF>
2. van der Aalst WMP (2007) Challenges in business process analysis. In: ICEIS (Selected Papers). pp 27–42

3. van der Aalst WMP, ter Hofstede AHM, Kiepuszewski B, Barros AP (2003) Workflow patterns. *Distrib Parallel Databases* 14(1):5–51
4. Aguilar ER, Ruiz F, García F, Piattini M (2006) Applying software metrics to evaluate business process models. *CLEI Electron J* 9(1)
5. Aguilar-Saven RS (2004) Business process modelling: review and framework. *Int J Prod Econ* 90(2):129–149. <http://www.sciencedirect.com/science/article/B6VF8-48WPV50-1/2/46adae8b9ceafa72bb47d4abadc2b2ad>
6. Basili VR, Caldiera G, Rombach HD (1994) The goal question metric approach. In: *Encyclopedia of software engineering*. Wiley, New York, vol 2, pp 528–532
7. Becker J, Rosemann M, von Uthmann C (2000) Guidelines of business process modeling. In: *Business process management: models, techniques and empirical studies*. Springer, Berlin, pp 30–49
8. Brucker AD, Krieger MP, Longuet D, Wolff B (2011) A specification-based test case generation method for uml/ocl. In: *Proceedings of the 2010 international conference on Models in software engineering, MODELS'10*. Springer, Berlin, pp 334–348. <http://dl.acm.org/citation.cfm?id=2008503.2008548>
9. Cardoso J (2007) Business process quality metrics: log-based complexity of workflow patterns. In: *Proceedings of the 2007 OTM Confederated international conference on On the move to meaningful internet systems: CoopIS, DOA, ODBASE, GADA, and IS—volume Part I, OTM'07*. Springer, Berlin, pp 427–434. <http://dl.acm.org/citation.cfm?id=1784607.1784645>
10. Fellbaum C (1998) WordNet: an electronic lexical database. Language, speech, and communication. Mit Press, Cambridge. <http://books.google.fr/books?id=Rehu8OOzMIMC>
11. Ghani A, Muketha K, Wen W (2008) Complexity metrics for measuring the understandability and maintainability of business process models using goal–question–metric (GQM). *Int J Comput Sci Netw Secur* 8:219–225
12. Gruhn V, Laue R (2006) Complexity metrics for business process models. In: *9th international conference on business information systems (BIS 2006)*. Lecture notes in informatics, vol 85. pp 1–12
13. Jansen-vullers MH, Netjes M (2006) Business process simulation a tool survey. In: *Workshop and tutorial on practical use of coloured petri nets and the CPN*
14. Jarke M, Informatik THAF (1992) Theories underlying requirements engineering: an overview of NATURE at genesis. In: *Aachener Informatik-Berichte. Fachgruppe Informatik der RWTH*. pp 19–31. <http://books.google.fr/books?id=0kKHAAACAAJ>
15. Johansson H (1993) Business process reengineering: breakpoint strategies for market dominance. Wiley, New York. <http://books.google.fr/books?id=wA7tAAAAAAAJ>
16. Kaiya H, Saeki M (2005) Ontology based requirements analysis: Lightweight semantic processing approach. In: *Proceedings of the fifth international conference on quality software, QSIC '05*. IEEE computer society, Washington, DC, pp 223–230. doi:10.1109/QSIC.2005.46
17. Khatri V, Vessey I, Ramesh V, Clay P, Park SJ (2006) Understanding conceptual schemas: exploring the role of application and is domain knowledge. *Inf Syst Res* 17(1):81–99
18. Krogstie J, Lindland OI, Sindre G (1995) Defining quality aspects for conceptual models. In: *Proceedings of the IFIP international working conference on information system concepts: towards a consolidation of views*. Chapman & Hall, Ltd., London, pp 216–231. <http://dl.acm.org/citation.cfm?id=645618.660960>
19. Leung F, Bolloju N (2005) Analyzing the quality of domain models developed by novice systems analysts. In: *Proceedings of the 38th annual Hawaii international conference on system sciences, vol 07. HICSS '05*. IEEE Computer Society, Washington, DC, p 188.2. doi:10.1109/HICSS.2005.98. <http://dx.doi.org/10.1109/HICSS.2005.98>
20. List B, Korherr B (2006) An evaluation of conceptual business process modelling languages. In: *Proceedings of the 2006 ACM symposium on applied computing, SAC '06*. ACM, New York, pp 1532–1539. doi:10.1145/1141277.1141633
21. Mendling J, Recker JC, Reijers HA (2010) On the usage of labels and icons in business process modeling. *Int J Inf Syst Model Design* 1(2):40–58. <http://eprints.qut.edu.au/32288/>
22. Mendling J, Reijers HA, Cardoso J (2007) What makes process models understandable? In: *Proceedings of the 5th international conference on business process management, BPM'07*. Springer, Berlin, pp 48–63. <http://dl.acm.org/citation.cfm?id=1793114.1793120>
23. Lopes de Oliveira J, Graciano Neto VV, Larissa da Costa S (2010) A business process metamodel for enterprise information systems automatic generation. In: *Brazilian workshop on model-driven development, Brazil. UFBA, Salvador, BA, Brasil*, p 4552
24. Purandare A, Pedersen T (2004) Word sense discrimination by clustering contexts in vector and similarity spaces. In: *Proceedings of the conference on computational natural language learning*. pp 41–48.
25. Purao S, Storey VC (2005) A multi-layered ontology for comparing relationship semantics in conceptual models of databases. *Appl Ontol* 1(1):117–139. <http://dl.acm.org/citation.cfm?id=1412350.1412360>
26. Shanks G (2007) Conceptual data modelling: an empirical study of expert and novice data modellers. *Australas J Inf Syst* 4(2): 63–73. <http://dl.acs.org.au/index.php/ajis/article/view/360>
27. Sugumaran V, C Storey V (2006) The role of domain ontologies in database design: an ontology management and conceptual modeling environment. *ACM Trans Database Syst (TODS)* 31:1064–1094
28. Sun PR (1995) Using information content to evaluate semantic similarity in a taxonomy. In: *Proceedings of the 14th international joint conference on artificial intelligence*. pp 448–453
29. Tzaviskou C, Keet CM (2007) A meta-model for ontologies with orm2. In: *Proceedings of the 2007 OTM confederated international conference on On the move to meaningful internet systems—volume Part I. OTM'07*. Springer, Berlin, pp 624–633. <http://dl.acm.org/citation.cfm?id=1780909.1781012>
30. Vanderfeesten I, Cardoso J, Reijers HA, Van Der Aalst W (2007) Quality metrics for business process models. *BPM and Workflow handbook*, pp 179–190
31. Wieringa R (1989) Three roles of conceptual models in information system design and use. In: *Falkenberg ED, Lindgreen P (eds) Information system concepts: an in-dept analysis*. North-Holland, Amsterdam, pp 31–51. <http://doc.utwente.nl/67619/>
32. Wu Z, Palmer M (1994) Verbs semantics and lexical selection. In: *Proceedings of the 32nd annual meeting on association for computational linguistics, ACL '94*. Association for computational linguistics, Stroudsburg, pp 133–138. doi:10.3115/981732.981751