An Ontological Model to Support the Application of Quality Practical Guidelines to Assess Business Process Descriptions

Olga Y. Rojas, Nemury Silega, Ashutosh Sharma, Yuri I. Rogozov, and Vyacheslav S. Lapshin

Abstract Business process models are a powerful means to show the business view of an organization. These models enhance the common understanding about the process of the organization and represent a key artifact to design information systems. Therefore, detecting and correcting errors in business process models are crucial to prevent errors in these models from spreading to other stages of the software development process. However, several studies have demonstrated that business process models usually have errors. The use of semiformal notations is one of the reasons for this fact, since these notations make the semantic validation of models difficult. Ontologies has become in a suitable solution to represent business process models. Since ontologies are a formal language based on description logics, its adoption enables semantic validation of modes. This paper aims to describe an ontology-based approach to detect errors in business process models. This approach implements a set of practical guidelines to assess the quality of process models. To develop the ontology, a solid methodology was followed. Likewise, the ontology was validated through a recognized method. Some examples that illustrate the applicability and impact of this approach are provided.

Keywords Business process models · Ontology · Quality practical guidelines

O. Y. Rojas
Universidad de Las Ciencias Informáticas, Habana, Cuba
e-mail: yarisbel@uci.cu

N. Silega (✉) · A. Sharma · Y. I. Rogozov · V. S. Lapshin
Department of System Analysis and Telecommunications, Southern Federal University, 347900 Taganrog, Russia
e-mail: silega@sfedu.ru

A. Sharma
e-mail: ashutosh@sfedu.ru

Y. I. Rogozov
e-mail: yrogozov@sfedu.ru

V. S. Lapshin
e-mail: lapshin@sfedu.ru

P. K. Singh et al. (eds.), *Futuristic Trends in Networks and Computing Technologies*, Lecture Notes in Electrical Engineering 936,
https://doi.org/10.1007/978-981-19-5037-7_36

1 Introduction

Business process modeling is a complex task where the knowledge of different people is combined, for example business specialists, stakeholders, and analysts [1]. The tacit knowledge of the business experts is an usual challenge during the business process modeling [2].

For the software development, business process models are a useful artifact for the software developers in order to understand the business of the companies [3–5]. Hence, to obtain free of errors user requirement specifications in the first stages of the information systems development process, the modeling of business processes is essential [6–8]. However, several studies provide evidences that demonstrate the existence of errors in business process model usually [7, 9]. The wrong modeling of business processes and its lack of correspondence with the system design are usual factors for the systems failure [10].

In spite of the relevance of process modeling, often is carried out by inexpert modelers [11]. Therefore, the quality of the models is affected in terms of flexibility and completeness [12, 13]. Even, sometimes modelers are not aware of the importance of their work to prevent that errors in these models from spreading to other stages of the software development process [14, 15].

Since the importance of the business process modeling for the software development, its evaluation is critical for the success of the process. In that sense, the understandability is a key quality indicator of business process models [16]. The understandability of models can be related to structural properties of its graphical elements [6]. Some practical guidelines to evaluate the understandability of business process models have been proposed [15]. However, it is not easy to find proposals that support the automatic application of these practical guidelines.

On the other hand, ontologies have become in a suitable technology to represent, validate, and analyze business processes [17]. The description of business processes in an ontology helps to detect model inconsistencies automatically and avoids the propagation of errors to system models [18].

This paper aims to describe an ontological model to assess business process descriptions. The assessment is based on practical guidelines to check the structural properties and the understandability of models [15]. Some specifications to automatically apply these guidelines in the ontology were implemented. To ensure the quality of the ontology, a solid methodology was adopted. Likewise, a procedure to evaluate the ontology was applied.

The rest of the paper is organized as follows. In Sect. 2, the practical guidelines and the basic technologies to develop the ontology are introduced. In Sect. 3, an ontological model to assess business process descriptions is described. In Sect. 4, some examples to illustrate the applicability of the approach are presented. Section 5 presents the conclusions and future work.

2 Background

Several works that deal with the quality of business process descriptions were analyzed [4, 6, 19–24]. Some authors have proposed practical guidelines to enhance the quality of the business process descriptions [6, 25, 26]. The application of these practical guidelines may reduce errors and improve the understandability of the business process models. However, some practical guidelines are not support by the modeling tools. Therefore, an alternative to support the application of these practical guidelines may be a useful contribution.

Several authors foster the adoption of ontologies to represent and validate business process descriptions. An ontological approach may support the identification of errors and improve activity labels of process models [2, 17, 18]. Therefore, we have adopted an ontological approach to support the application of practical guidelines to assess business process descriptions.

2.1 *Quality Practical Guidelines*

Table 1 gives ten practical guides related to the size of business process models. These guidelines allow to classify business process models according to the number of modeling elements they have [21, 27].

Table 1 Practical guidelines regarding size

Size problem	Guideline
P1. *High number of elements*	More than 31 elements in a model must with be avoided
P2. *High number of events*	More than seven events in a model must be avoided
P3. *High number of start event*	More than two events in a model must be avoided
P4. *Lack of start events*	The start event cannot be missed
P5. *High number of end events*	More than two end events in a model must be avoided
P6. *Lack of* end *events*	The end event cannot be missed
P7. *High number of intermediate events*	More than five intermediate events in a model must be avoided
P8. *High number of sequence flows*	More than 34 sequence flows in a model must be avoided
P9. *High number of* gateways	More than 12 gateways in a model must be avoided
P10. *High number of activities*	More than 31 activities in a model must be avoided

2.2 Methodology and Tools to Support the Development of the Ontology

The selection of a proper methodology is a key step to develop an ontology. To guide the development of the ontology presented in this paper, the methodology of Noy and McGuinness was adopted [28]. This is a solid methodology that has been widely adopted and includes the following steps:

- Determine the domain and scope of the ontology.
- Consider reusing existing ontologies.
- Enumerate important terms in the ontology.
- Define the classes and the class hierarchy.
- Define the properties (called relationships or slots) of the classes.
- Define facets and/or restrictions on slots or relationships.
- Define instances.

Web Ontology Language (OWL) [Ref] was adopted to represent the ontology. OWL is based on description logics and includes the operator's intersection, union, and negation which are very useful to represent knowledge. Furthermore, the models represented in OWL can be analyzed by reasoners which automatically check the consistency and infer new knowledge. The reasoner Pellet was adopted to analyze our ontology. To implement the ontology, the tool Protégé [Ref] was adopted. Protégé is multiplatform and open-source, and it has a flexible and extensible architecture. The language OWL and the reasoner Pellet are supported by Protégé.

3 An Ontological Model to Support the Application of Practical Guidelines to Assess Business Process Description

To develop the ontology, the steps defined in the methodology of Noy and McGuinsess were carried out. Below the main results are described.

Step 1. Determine the domain and scope of the ontology

The ontology has the purpose of assessing the quality of the business process descriptions applying practical guidelines. To achieve this objective, the ontology must be able of answering the following competence questions (CQ):

1. Does the process meet basic workflow patterns?
2. What activities are included in the process?
3. What processes have problem of size?
4. What processes have problem of morphology?
5. What processes are efficient?
6. What processes are inefficient?

7. What processes are very efficient?
8. What processes are very inefficient?
9. What processes are low efficient?

Step 2. Consider reusing existing ontologies

Concepts of the ontology described by Silega and Noguera were reused [18]. This ontology supports the description of business processes and includes specifications for its validation.

Step 3. Enumerate important terms in the ontology

The main terms are related with the components of a process such as activity, event, gateway, input, output, and others terms. Furthermore, terms related with the application of practical guidelines (see Table 1) to assess the models are considered.

Step 4. Define the classes and the class hierarchy

To define the classes of the ontology, three main elements were considered. First of all, we considered the concepts related to the representation of business process, for this regard, the concepts of the ontology developed by Silega [18], such as **Processs**, **Activity**, **Event,** and **Gateway** were reused. Likewise, this ontology includes classes such as **Step** and **FlowElement** to represent the flow of activities. Furthermore, we included classes to assess the processes based on the practical guide. For example, to classify the processes that do not fulfill some practical guideline related to the size, the class **ProcessWithSizeProblem** was included. The classes ProcessWith-MorfologyProblem, **ProcessEfficient**, **ProcessInefficient**, **ProcessVeryInefficient**, and **GatewayWithProblem** are related to the application of the practical guidelines too. These last classes were declared as defined classes. Defined classes in OWL include a set of necessary and sufficient conditions, thus a reasoner can automatically infer their instances. Hence, the process with problems will be automatically identified. Figure 1 shows an excerpt of the class hierarchy.

Step 5. Define the properties

Properties are the other core component of ontologies. Object properties and data properties are the two types of properties in an ontology. The object properties allow to represent a binary relationship between two individuals. Each object property has an inverse object property, for example, an **Activity** *belongsTo* a **Process** and a **Process** *HasActivity* an **Activity**. A total of 64 object properties in the ontology were defined. Table 2 gives some object properties related to the classes Process and Step.

After creating the properties, it is possible to declare some necessary and sufficient conditions to automatically classify the instances of the defined classes. For example, Fig. 2 shows the necessary and sufficient conditions to identify the processes that belong to the class **ProcessWithSizeProblem**.

Other rules to assess the processes also have been implemented. In spite of the expressivity richness of OWL, some complex relations cannot be expressed.

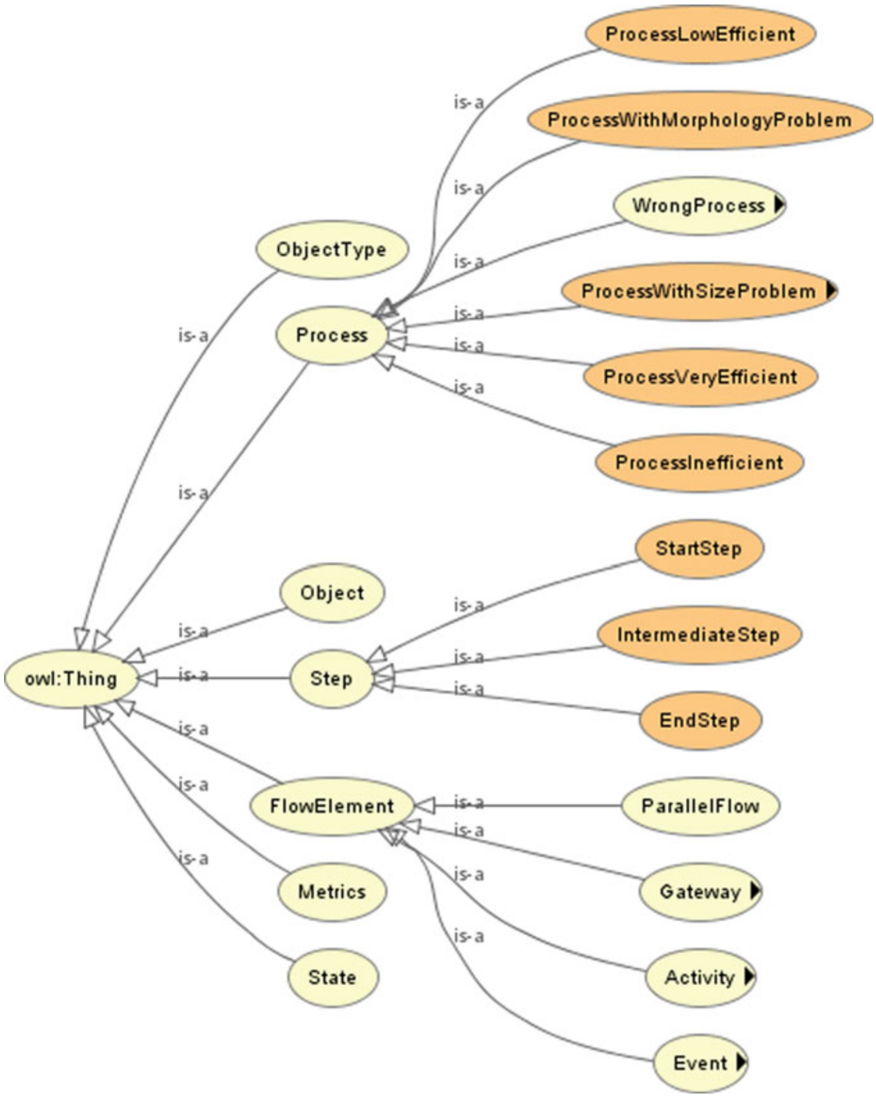

Fig. 1 Classes hierarchy

Therefore, the semantic web rule language (WSRL) was adopted to complement OWL.

Step 6 Define instances

An example to illustrate the creation of instances in the next section is presented.

Table 2 Examples of object properties

Domain	Object	Range
Process	***HasActivity***	Activity
Process	***HasStartEvent***	StartEvent
Process	***EndEvent***	StartEvent
Process	***HasStep***	Step
Process	***HasProblemWithMetrics***	Step
Step	***ExecutesTo***	Metrics
Step	***FollowsTo***	FlowElement

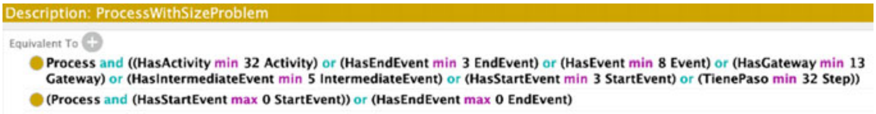

Fig. 2 Example of a set of necessary and sufficient conditions

4 Evaluation of the Ontology

Checking that the ontology fulfills its conditions as a logical-formal system is the first step to evaluate its quality. The reasoner Pellet confirmed that our ontology fulfills its conditions as a logical-formal system.

On the other hand, to demonstrate the applicability of our approach, some business processes in the ontology were described and assessed. The processes *Process_Make-A_Deposit* and *Process-Example2* were modeled in the ontology and classified as a **ProcessVeryEfficient** while the process *Process-Example1* was classified as **ProcessWithSizeProblem** because meets the necessary and sufficient conditions of this class. Figure 3 depicts a view of Protégé where the classifications carried out by the reasoner are displayed. This view shows the classifications for *Process_Make-A_Deposit*, *Process-Example1,* and *Process-Example2*. This examples answer the competence questions 4 and 7.

By means of this ontology, other practical guidelines can be verified. For example, it is possible to identify the gateways with multiple input and output flows. A process with this type of gateways should be classified as a process with problem of morphology. Figure 4 depicts an example of a process description with this problem.

To identify the gateways with multiple input and output flows, some rules were specified. The gateways with this problem will be classified as **GatewayWith-Problem**. Furthermore, we defined that if a processes has some **GatewayWith-Problem** then it is classified as **ProcessWithMorphologyProblem**. After modeling in the ontology, the process of Fig. 4, the reasoner classified *Gateway2* as a **Gate-wayWithProblem** (Fig. 5a). Since that Process-Example3 has to *Gateway2,* it is classified as a **ProcessWithMorfologyProblem** (Fig. 5b).

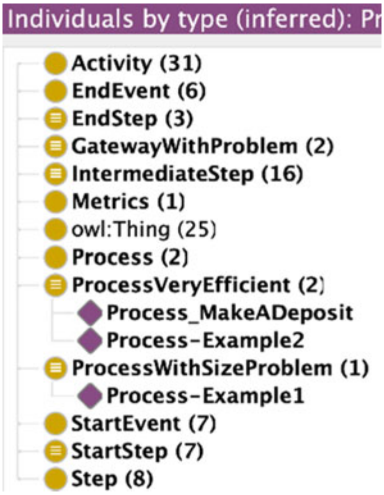

Fig. 3 Classification of business processes

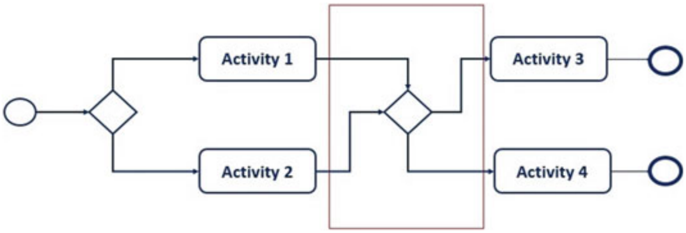

Fig. 4 Model of a process with problem of morphology

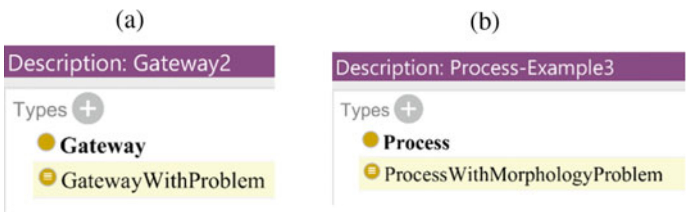

Fig. 5 Classification of a process with problem of morphology

5 Conclusions

The business process models are a useful instrument to understand the business of organizations. Likewise, it is a key artifact to design information systems. Therefore, assuring the quality of process descriptions is crucial to prevent errors in other stages of the software development process. Some quality practical guidelines to evaluate business process descriptions have been proposed. In this article, an ontological model to represent and assess business processes was introduced. The formalization of the process models through the OWL language allows verifying the problems related to non-compliance with the quality practical guidelines related to general complexity. The compliance of these practical guidelines improves the understanding between business experts, analysts, and the development team. The conditions of the ontology as a logical-formal by means of a reasoner were verified. Some examples to illustrate how the expressivity richness of OWL was exploited to represent and assess business process were presented.

Acknowledgements This research is supported by a postdoc fellowship granted by the Institute of Computer Technologies and Information Security, Southern Federal University, project № PD/20-02-KT.

References

1. Marlon Dumas MLR, Mendling J, Reijers HA (2013) Fundamentals of business process management. Springer, Austria
2. Júnior VHG (2016) Utilização de Ontologias para Certificação de Boas Práticas em Modelagem de Processos de Negócio. Universidad de Federal Do Rio Grande Do Sul, Brasil, pp 13–14
3. Jean Carlos Guzmán FL, Matteo A (2013) Del Modelo de Negocio a la Arquitectura del Sistema considerando Metas, Aspectos y Estándares de Calidad, in Revista Antioqueña de las Ciencias Computacionales y la Ingeniería de Software, RACCIS, Venezuela, pp 19–37
4. Isel Moreno Montes de Oca MS, Reijersc HA, Morffi AR (2015) A systematic literature review of studies on business process modeling quality. Inform Softw Technol 58:187–205
5. Méndez R, Urrutia A (2016) Complejidad en modelos conceptuales de procesos de negocios. Propuesta de métricas de calidad de modelos conceptuales de procesos. Revista GTI 15(43):47–62
6. Mendling JHAR, van der Aalst Barjis WMPJ (2010) Seven process modeling guidelines (7PMG). Inform Softw Technol 52:127–136
7. Silega N (2014) Método para la transformación automatizada de modelos de procesos de negocio a modelos de componentes para Sistemas de Gestión Empresarial, in Centro de Informatización de Entidades. Universidad de las Ciencias Informáticas, La Habana, Cuba, p 5
8. Ahmet Dikici O, Demirors O (2018) Factors influencing the understandability of process models: a systematic literature review. Inform Softw Technol 93:112–129
9. Mendling J (2009) Empirical studies in process model verification, vol 5460. Springer, Berlin, Heidelberg, pp 208–224
10. Barjis J (2008) The importance of business process modeling in software systems design. In: Science of computer programming, 2008. Science of Computer Programming, pp 73–87
11. Recker J et al (2009) Business process modeling-a comparative analysis. J Assoc Inf Syst 10(4):1
12. Leung FBN (2005) Analyzing the quality of domain models developed by novice systems analysts. In: Proceedings of the 38th annual Hawaii international conference on system sciences. 2005, IEEE Computer Society, pp 188.2
13. Samira Si-Said SA, Comyn I (2013) Improving business process model quality using domain ontologies. J Data Semant 2:75–87

14. Mendling J, et al (2006) A quantitative analysis of faulty EPCs in the SAP reference model. BPM Center Report BPM-06-08, BPMCenter.org
15. Oca IMMD (2015) Patrón y clasificación taxonómica para directrices prácticas en modelos de procesos de negocio. In: *Departamento de Computación*, Universidad Central "Marta Abreu" De Las Villas, Santa Clara, Cuba
16. James Nelson GP, Genero M, Piattini M (2012) A conceptual modeling quality framework. Software Qual J 201–228
17. Gassen JB et al (2017) An experiment on an ontology-based support approach for process modeling. Inf Softw Technol 83:94–115
18. Silega N, Noguera M (2021) Applying an MDA-based approach for enhancing the validation of business process models. Procedia Comp Sci 184:761–766
19. Aguilar-Saven RS (2004) Business process modelling: review and framework. Int J Prod Econ 90(2):129–149
20. Becker J, Rosemann M, Von Uthmann C (2000) Guidelines of business process modeling. Business process management. Springer, pp 30–49
21. Flavio Corradini AF, Fornari F, Gnesi S, Polini A, Re B, Spagnolo GO (2017) Quality assessment strategy: applying business process modelling understandability guidelines. Italia
22. Figl K (2017) Comprehension of procedural visual business process models. Bus Inf Syst Eng 59(1):41–67
23. Genero M et al (2011) Research review: a systematic literature review on the quality of UML models. J Database Manage (JDM) 22(3):46–70
24. O'Neill P, Sohal AS (1999) Business process reengineering a review of recent literature. Technovation 19(9):571–581
25. Moreno-Montes de Oca I, Snoeck M (2014) Pragmatic guidelines for business process modeling. T.R. 2592983, Editor. 2014, Faculty of Economics and Business, KU Leuven
26. Corradini F et al (2017) A Guidelines framework for understandable BPMN models. Data Knowl Eng 113:129–154
27. Sánchez-González L, et al (2010) Quality assessment of business process models based on thresholds. In: OTM confederated international conferences on the move to meaningful internet systems. Springer
28. Noy NF, McGuinness DL (2001) Ontology development 101: a guide to creating your first ontology. Stanford Medical Informatics, Stanford