

ITC 3/46Journal of Information Technology
and Control

Vol. 46 / No. 3 / 2017

pp. 360-371

DOI 10.5755/j01.itc.46.2.18520

© Kaunas University of Technology

**Transforming BPMN 2.0 Business Process Model into SBVR
Business Vocabulary and Rules**

Received 2017/07/13

Accepted after revision 2017/09/11

<http://dx.doi.org/10.5755/j01.itc.46.2.18520>

Transforming BPMN 2.0 Business Process Model into SBVR Business Vocabulary and Rules

Egle MickeviciuteKaunas University of Technology, Department of Information Systems
Studentų Str. 50, LT-51368, Kaunas, Lithuania**Rimantas Butleris**Kaunas University of Technology, Centre of Information Systems Design Technologies
K. Barsausko Str. 59, LT-51423, Kaunas, Lithuania**Saulius Gudas**

Vilnius University, Institute of Mathematics and Informatics, Akademijos Str. 4, LT-08663, Vilnius, Lithuania

Eimutis Karciauskas

Kaunas University of Technology, Software Engineering Department, Studentu Str. 50, Kaunas, Lithuania

Corresponding author: egle.mickeviciute@ktu.lt

As the success of information system projects and their development also relies on the knowledge of domain experts, the modelled processes should be presented to them for validation purposes before the implementation stage. Domain experts do not have the knowledge of business process modelling notations. Therefore, the validation may not be correct. However, they better understand structured natural language, such as SBVR that defines the meaning of business concepts and make them unambiguously understandable by human experts and also by software systems. The solution presented in this paper allows transforming BPMN 2.0 business process models into SBVR business vocabularies and business rules. This solution is implemented as a plug-in and is available in the MagicDraw CASE tool. An experimental evaluation of the proposed solution with three domain areas proved that SBVR business vocabulary and business rules could be fully obtained from BPMN 2.0 business process model when certain requirements for BPMN 2.0 business process model are met. The advantages of the solution are an automatic transformation for various BPMN modelling situations, tracing links between two models in one modelling environment and avoidance of necessity to use linguistic techniques.

KEYWORDS: Business process, Business vocabulary, Business rules, Transformation, BPMN 2.0, SBVR, QVT.

1. Introduction

Information system (IS) projects usually start with forming business vocabularies, creating business processes and business rules models that enable the possibility to avoid misunderstandings and inconsistencies between business stakeholders and IS developers that can end with implementation errors, with a poor performance of implemented IS or even with the unsuitable IS. The goal of the paper is to introduce the solution of transforming BPMN 2.0 business process model into SBVR business vocabulary and business rules to create comprehensive BPMN business process and business rules models that domain experts are able to understand for validation purposes. This ability will gain the importance when the application of business vocabularies will become available and used for a wide range of business users. As Leopold et al. [6] stated, the validation of integrated business process and business rules models still has to rely on a natural language. We argue that for assuring such validation, business process and business rules models cannot be freely created but must be based on business vocabularies shared among IS stakeholders that support business processes and are interested to enforce their compliance with business vocabularies and business rules, which represent conceptual data models. Usually, business processes tend to be expressed in an imperative way using graphical models like BPMN 2.0, while vocabulary and rules are expressed in a declarative way using some formal or informal (natural) language. Though these aspects are complementary to each other [27, 4], until recently it was impossible to model them in the same Computation Independent Modelling (CIM) environment.

BPMN (Business Process Model and Notation) 2.0 [18], as well as other process modelling languages, lack standard, unambiguous conceptual means for representing business rules. Without business rules, business process models are incomplete, ambiguous and even misleading. A systematic literature review [32] indicated that the integrated modelling of business processes and business rules has the greatest value among other solutions; methods and tools for such modelling are not sufficiently developed, and the largest contributions are required at the conceptual level. Studies [22, 23] show that graphical business process models are better understandable by the analysts who have experience with modelling languages. Literature

review on business process and business rules modelling languages showed that 10-20 percent of errors occur in requirement and modelling stages [13]. In addition, the visualization has a huge impact – understanding depends on knowledge. This gives the conclusion that domain experts cannot fully understand graphical notations, as they do not have the knowledge [23], so they always tend to choose natural language representation.

Object Management Group (OMG) has created the Semantics of Business Vocabulary and Business Rules (SBVR) meta-model [19] that provides an opportunity to describe business concepts and business rules using Controlled Natural Language (CNL), which is understandable for business users. When SBVR is used, domain experts can construct business vocabularies and business rules, or at least understand them for validation purposes. Furthermore, creating SBVR model before software implementation stage could provide an opportunity to facilitate its maintenance and evolution [17] as business knowledge extraction from existing software systems is widely investigated to make software changes with minimum resources. SBVR is based on formal logics and can be applied for computer processing, but it cannot be directly used in semantic technologies because these have their languages, for instance, OWL2 or RDFS.

The paper presents a comparison of our solution with the most advanced, to our knowledge, BPMN into SBVR transformation solutions and an experimental evaluation of the implemented transformation plug-in.

The rest of the paper is structured as follows. In section 2, the related work is analyzed and compared. Section 3 presents BPMN 2.0 into SBVR transformation rules and their implementation using QVT transformation language. Section 4 introduces with the requirements for BPMN 2.0 business process model elements to get semantically correct transformation result (SBVR business vocabulary and rules) and points out the cases that are not covered by this transformation. In section 5, an implemented plugin in CASE tool is shortly presented. Section 6 provides experimental evaluation with selected BPMN 2.0 business process models. Section 7 draws conclusions. Acknowledgement finalizes the paper.

2. Comparison of BPMN into SBVR transformation solutions

As IS are becoming more complicated, they require applying several models [7] to describe business processes. Business process modelling and business rules modelling approaches are considered to complement each other [3] in order to have the comprehensive representation of a problem domain. There are many ways of how business processes and business rules could be used together. Each of them has their advantages and disadvantages; some are just more theoretical than practical. Some proposals would be hard to implement [1, 2], others do not provide enough infor-

mation about the implementation [4, 5]. This section presents the analysis of related work of BPMN into SBVR transformation and their comparison with our solution. The comparison was made on the base of analyzed publications (to our knowledge, the implementations of related works are not publicly available or reported). Our solution is based on transformations analysis between BPMN and SBVR modelling languages to obtain a full SBVR model (business vocabulary and business rules) that gives the same semantic meaning of BPMN business process model after the transformation.

Table 1 presents a comparison of the most advanced BPMN 2.0 into SBVR transformations proposed in

Table 1

Comparison of BPMN into SBVR transformation solutions presented in publications of different authors (the symbols “(+)” and “(-)” indicate whether the criteria are fulfilled or not; (+/-) mean there is no full information provided. Some of the symbols are explained below the table with corresponding references, e.g., (*i))

Criteria/Solution	Malik and Bajwa [8,9]	Skersys et al. [28,29,31]	Leopold et al. [6]	Our solution
Vocabulary representation	(-)*1	(+)	(-)*2	(+)
Rules representation	(+)	(-)*3	(+)	(+)
Textual representation	(+)	(+)	(+)	(+)
Visual representation (Model)	(-)	(-)	(-)	(+)*4
Usage of DSL	(-)	(-)	(-)	(+)*5
Without supplementary data structure	(+)	(-)	(+)	(+)
Integrated solution	(-)*6	(+)*8	(-)*7	(+)*9
Full set of BPMN elements	(+/-)*10	(+)	(+/-)*11	(+)*12
Element tracing	(-)	(-)	(-)	(+)
Without linguistic techniques	(-)*13	(-)	(-)*14	(+)
Automatic transformation	(+)	(-)*15	(+)	(+)

*1, *2 - Vocabulary is presented within the rules, but not separately

*3 - Rules can be formed from the vocabulary by a user using given templates

*4 - Business vocabulary can be presented as a diagram with the use of adapted UML class model

*5 - SBVR profile [15] was implemented and used

*6 - Separate tool called BR - Generator

*7 - Separate tool called Natural language generator (NLG)

*8 - VETIS plug-in

*9 - VEPSEM plug-in

*10 - Limited BPMN 2.0 element types (main elements but not all their types)

*11 - There is no information about exact BPMN element types used in this solution

*12 - All elements and their types with additional timer events properties (Time Duration, Time Cycle, Time Date)

*13 - Simple linguistic methods to resolve some phrases

*14 - Combines different techniques from linguistics and graph decomposition

*15 - Semi-automatic transformation

publications of different authors. The automatic method for transforming BPMN business process model into SBVR and a tool under implementation was presented in [8, 9]; however, the method does not cover a full set of BPMN 2.0 elements. As changes to SBVR meta-model could cause problems in the solution maintenance, Skersys et al. [28, 29, 31] proposed to use supplementary mapping data between two meta-models and semi-automated extraction of business vocabularies from BPMN business process model; the latest solution improvements have not considered business rules. Finally, the method for generation of natural language texts from BPMN business process model [6] in order to validate business processes requires sophisticated linguistic processing techniques and does not guarantee completeness and reliability of results.

In contrast to the aforementioned solutions, our proposed solution obtains not only SBVR business vocabulary but also forms SBVR business rules from BPMN 2.0 business process model. It does not apply any sophisticated linguistic techniques, instead, the specific requirements must be followed while creating BPMN 2.0 business process model. These requirements are based on an existing good practice for BPMN modelling, therefore, they should not be considered as a disadvantage.

The usage of DSL (for SBVR Profile) in a CASE tool allows having two different models in one modelling environment and creating trace links during the transformation between BPMN 2.0 business process model elements and SBVR business vocabulary elements. This capability provides the information about the source of every SBVR business vocabulary concept in BPMN 2.0 business process model.

The novelty of our proposed solution can be described by the following results:

- 1 The transformation rules from BPMN 2.0 into SBVR are formally defined, therefore, they can be implemented in other CASE tools. The defined rules cover all elements in BPMN 2.0 using their names. Other properties were not analyzed (except for timer events).
- 2 The defined set of requirements for BPMN 2.0 business process model allows transforming

them into the consistent and full SBVR business vocabulary and business rules.

3. Transformation rules and their implementation using QVT

The transformation from one model into another requires mappings between their elements of meta-models. Mapping pairs of BPMN and SBVR meta-models' elements were found in [31]. However, these element mappings specify compliance between two models without the information on what kind of compliance it is (full or partial) and leaving with no information of inner logic on how to transform certain BPMN element into a certain SBVR concept. In contrast to this solution, we defined detailed mappings of the elements between BPMN 2.0 and SBVR (Table 2), which helped us to define the transformation rules that also include SBVR business rules extraction from BPMN 2.0 business process model.

Our solution uses an implemented SBVR profile in the UML CASE tool MagicDraw that is based on its DSL engine, which was created in compliance with UML profiling [16, 24, 25].

Based on element mappings, BPMN 2.0 into SBVR transformation rules were defined and divided into three main groups: BPMN elements to SBVR general concepts, BPMN elements or their combinations to SBVR verb concepts and BPMN elements or their combinations to SBVR business rules. A small fragment of all 46 transformation rules is presented in Table 3.

All defined transformation rules were used to create the transformation algorithm, which was implemented in QVT (Query/View/Transformation) transformation language and executed using QVT transformation engine, which takes as input BPMN process model, exported from MagicDraw into Eclipse XMI. The transformation results are exported back into MagicDraw in the form of UML models customized in SBVR profile, where noun concepts extend UML classes, attributes, and instance specifications; verb concepts extend relationships, and business rules – expressions [15]. An example of the QVT transformation code is presented in Table 4.

Table 2

A fragment of mapping pairs of elements between BPMN and SBVR meta-models (the letters “F” and “P” mean full or partial compliance, the letters “F/P” mean full and partial compliance (both))

BPMN2 Element	SBVR General Concept	SBVR Verb Concept	SBVR Rule	Example
Event	P	F/P	P	SBVR Verb Concept (unary): car booking request <i>is_received</i> 2 SBVR General Concepts: car booking request received Concept type: state SBVR Verb Concept (binary): car booking request <i>has_state</i> received
Activity	P	P	P	Part of SBVR Verb Concept branch <i>create</i> rental_contract
Message	F	P	P	SBVR General Concept: rental_contract_proposal
Sequence Flow	P	P	F	SBVR Rule: It is obligatory that car booking request <i>is_canceled</i> after rental contract <i>is_rejected</i> .

Table 3

Examples of the transformation rules (one from each step)

Step	Transformation Rule
1	<p>BPMN 2.0 Pool → SBVR General Concept $T_5: transform(BPM, pool: Pool) \rightarrow SBVR\ General\ Concept$ e.g.: $transform(BPM, 'operating\ company') \rightarrow operating_company$</p>
2	<p>BPMN 2.0 Activity → SBVR Verb Concept $T_{11}: transform(BPM, pool/lane: Pool/Lane, activity: Activity) \rightarrow SBVR\ Verb\ Concept$ e.g.: $transform(BPM, 'branch', 'approve\ car\ booking\ request') \rightarrow branch\ approve\ car\ booking\ request$</p>
3	<p>Activity initiates ExclusiveGateway, which has outgoing SequenceFlows (with Conditions) and Activities (3 rules) (Figure 1) $T_{33}: transform(BPM, pool/lane: Pool/Lane, activity_1: Activity_1, sequence_flow(activity_1, exclusiveGateway): SequenceFlow, exclusive_gateway: ExclusiveGateway, ((sequenceFlow(exclusiveGateway, condition_1: Condition_1, activity_1): SequenceFlow, activity_1: Activity_1) (sequenceFlow(exclusiveGateway, condition_2: Condition_2, activity_2): SequenceFlow, activity_2: Activity_2))) \xrightarrow{a} SBVR\ Business\ Rule$ e.g.: $transform(BPM, 'branch', 'check\ renter', sequence_flow('check\ renter', ExclusiveGateway), ((ExclusiveGateway, (sequence_flow(ExclusiveGateway, 'car\ booking\ request\ is_valid', 'accept\ car\ booking\ request'), 'accept\ car\ booking\ request') (sequence_flow(ExclusiveGateway, 'car\ booking\ request\ is_invalid', 'reject\ car\ booking\ request')))) \xrightarrow{a}$ It is obligatory that branch <i>accept</i> car_booking_request or branch <i>reject</i> car_booking_request after branch <i>check</i> renter It is obligatory that branch <i>accept</i> car_booking_request if car_booking_request <i>is_valid</i> It is obligatory that branch <i>reject</i> car_booking_request if car_booking_request <i>is_invalid</i></p>

Figure 1

Example of BPMN 2.0 ExclusiveGateway

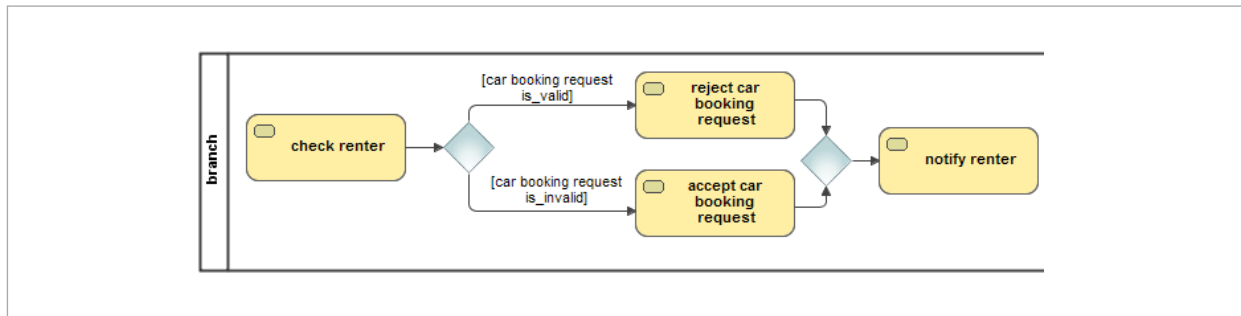


Table 4

QVT code from transformation rules (no. corresponds to step number in Table 2; 1 no. transformation transforms BPMN 2.0 Pool and Lane elements)

No.	QVT code
1	<pre> mapping ActivityPartition::containerToGeneralConcept() when {self.isStereotypedBy('Process', 'Lane')} -- <i>mapping guard</i> {var oneClass:Class:= self.returnGeneralConcept(getNameWithNoSpace(getLaneIfRepresents(self))); if(oneClass = null) then {oneClass:= self.map toGeneralConcept(getNameWithNoSpace(getLaneIfRepresents(self))); genConArray+=List{oneClass};} endif;} </pre>
2	<pre> mapping ActivityNode::allActivityToVerbConcept() when { self.stereotypedAsTask() or self.stereotypedAsSubProcess() } {var verbConcept:String; verbConcept:=getNameWithNoSpace(self.getActivityNounPart().toLower()) + "+getNameWithNoSpace(self.getActivityVerbPart().toLower())+" + getLaneIfRepresents(getActualContainer(self)).toLower(); if(self.returnVerbConceptString(verbConcept)=null) then {self.map activityToVerbConcept(); stateAssocArray+=List{verbConcept};} endif;} </pre>
3	<pre> mapping ActivityNode::inclusiveGatewayOutcomeToBusinessRule() when { self.isStereotypedBy('Gateways', 'InclusiveGateway') } {var rule : String; rule:= 'It is obligatory that ' + self.outgoing[ControlFlow].target[ActivityNode]->asSequence()->first(). returnCatchLink() + ' after' + getInclusiveRulePart(self.incoming[ControlFlow].source[ActivityNode]->asSequence()); log("inclusiveRuleout: " + rule); if(rule <> 'It is obligatory that ') then rule->toConstraint() endif;} </pre>

4. Requirements and excluded cases

In order to get full and semantically correct business vocabulary and business rules based on SBVR from BPMN 2.0 business process model without using any linguistic analysis, two main requirements were formulated for BPMN processes. These requirements were presented in detail in [15].

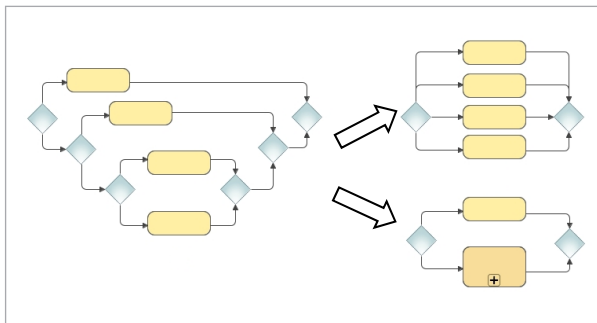
It is important to mention that there are various studies about how business process models should be created (model size, element count in one diagram, element naming structures, etc.), as well as what causes misunderstandings between analysts and domain experts [10-12, 20, 21]. Silver [26] named ten tips for effective process modelling, and one of them is still used by many modelers. It is an activity name structure (verb-noun), which was also used in our requirements for our solution.

Excluded cases, that is, the cases that are not covered by the developed solution, were defined due to the large-scale of this project or trying to avoid complicated process model cases where the logic of BPMN 2.0 process model can be misunderstood. These cases are listed below (C1-C5).

C1. The transformation does not cover successive decisions points (BPMN 2.0 gateways) when there are more than two of them. The usage of successive decision points is not a good solution in order to create a clear BPMN 2.0 business process model. Such parts of the BPMN2 process model can be replaced with equivalent cases using fewer decision points, as showed in Figure 2.

Figure 2

An example of how to decrease the amount of successive decision points (BPMN 2.0 Gateways)



C2. Transformation does not cover the expanded sub-processes that are used to view detailed sub-processes in the same diagram. Sub-processes should be created in separate activity diagrams. Business process logic does not depend on whether one uses expanded sub-processes or sub-processes in separate diagrams, it is just an option of a business process graphical display.

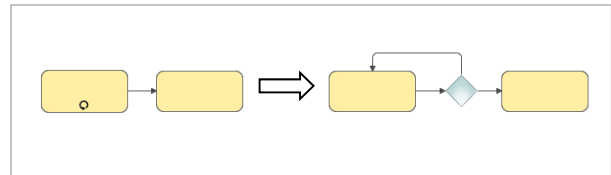
C3. It is not sufficient to solely have an assigned process for BPMN 2.0 element Call Activity, the element's name must be declared. Otherwise, it will be ignored.

C4. There is no checking whether all parts of rules exist to get a full rule and consequently this (when requirements are not used) results in semantically incorrect and incomplete rules.

C5. Transformation does not cover all the parameters of BPMN 2.0 business process model elements except BPMN 2.0 element names, and more analyzed BPMN2 timer events parameters [14], leaving it for further researches. Now, cases, where BPMN element properties are used, can be changed into equivalent cases using additional elements, as showed in Figure 3.

Figure 3

An example of how to represent an activity in a loop without using loop property of activity in BPMN 2.0

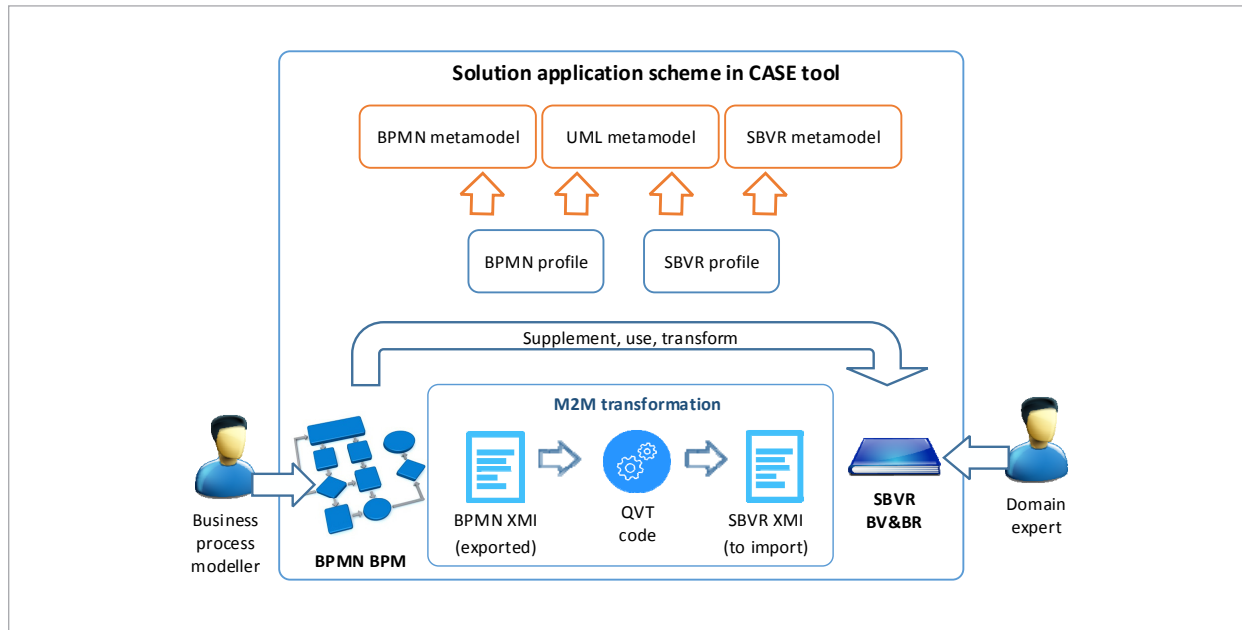


5. A plug-in for BPMN 2.0 into SBVR transformation

The principal scheme for the transformation of BPMN 2.0 into SBVR is presented in Figure 4. Creation of business concept vocabulary is the first and requisite step before creating a business process model. Such a vocabulary can exist in an enterprise and can be created in advance or developed during the creation of business process model. Therefore, business concepts and business model can be created simultaneously.

Figure 4

The principal scheme for BPMN 2.0 into SBVR transformation



The business process modeller can create a BPMN business process model using concepts from a SBVR business vocabulary and business rules defined by domain modeler or without using a SBVR vocabulary. Created business concepts can be used to create BPMN 2.0 business process model by using D&D (drag and drop) capability that was developed in [30]. When the business model is complete, the transformation from BPMN 2.0 business process model into SBVR business vocabulary and business rules can be performed. The plug-in exports BPMN XMI and gives it to the QVT code for creating SBVR XMI, which is imported into CASE tool after the transformation.

To perform transformation in the MagicDraw CASE tool, the project must be created using a template called “VEPSEM Template Project”. The template provides a specific structure of packages whose names indicate what kind of models must be placed in. The plug-in also allows merging or overwriting transformation results and provides the capability of trace links creation that allows linking two models elements in one modelling environment. The transformation result (SBVR business vocabulary and business rules) is placed in a package “SBVR Model”. SBVR vocabulary can be presented graphically,

and the whole SBVR concept list can be seen in a tree structure. SBVR business rules are placed in a table structure. The transformation from BPMN process model into SBVR business process vocabulary and business rules can be performed in CASE tools that support BPMN and SBVR meta-models and are implemented using BPMN and SBVR profiles [15], based on UML meta-model.

6. The experimental approval

The goal of the experiment was to evaluate the transformation plug-in (transformation from BPMN 2.0 business process model into SBVR business vocabulary and business rules) concerning the quality criteria for this research. The evaluation was done from the perspective of the users who will use the prototype, and from the perspective of the researchers who will analyze the solution, further develop the tool, or propose new ideas.

The experimental evaluation was conducted by transforming BPMN 2.0 business process models into SBVR business vocabularies concepts and business rules using four domains. A representative example

of the EU-Rent processes was created according to our requirements on the base of the EU-Rent example (OMG 2008). It contains two processes of the 1st layer, three sub-processes of the 2nd layer, and seven sub-processes of the 3rd layer. For Magic Library, Training Centre and Order Handling processes, created by NoMagic Company for its customers, transformations were accomplished with their original models and adapted ones (according to requirements and excluded cases that were presented earlier). The following quality criteria are analyzed in our experiment:

$$P = \frac{|RE \cap AE|}{|AE|} \quad (1)$$

$$R = \frac{|RE \cap AE|}{|RE|} \quad (2)$$

$$F = 2 \times \frac{|P \times R|}{|P + R|} \quad (3)$$

Here P means precision, R means recall; RE is a set of transformable BPMN 2.0 elements; AE is a set of actually transformed elements; $RE \cap AE$ means a set of correctly transformed elements; F is F -measure (a balanced F -score).

The results of the experimental evaluation for transforming original process models are presented in Table 5 and the results for transforming adapted business process models are presented in Table 6.

Results in Tables 5 and 6 show that transformations performed using requirements for BPMN 2.0 business process models allow to ensure desired transformation outcome while transformation results with original models are significantly worse. It is worth to mention that results from the original models depend on how close a business process model is to the requirements of this solution. Furthermore, the requirements for BPMN 2.0 business process models are based on a good modelling practice – they do not complicate the creation of business processes, on the contrary, they make models solid and easy to read. Of course, modelers have to put more effort into modelling and to avoid complicated, non-compliant process models, which often means that processes are un-

Table 5

The experimental results of transforming original process models

Domain	$ RE \cap AE $	$ AE $	$ RE $	P	R	F
Magic Library	55	131	127	0,42	0,43	0,42
Order Handling	45	88	127	0,51	0,35	0,41
Training Centre	22	37	92	0,59	0,24	0,34
Mean				0,51	0,34	0,39

Table 6

The experimental results of transforming process models adapted to our requirements

Domain	$ RE \cap AE $	$ AE $	$ RE $	P	R	F
Magic Library	127	127	127	1	1	1
Order Handling	127	127	127	1	1	1
Training Centre	92	92	92	1	1	1
EU-Rent	430	430	430	1	1	1

der-investigated and their logics are misunderstood. However, this allows obtaining such a SBVR model that can be used for validation purposes.

7. Conclusions

The analysis of related works showed that the problem of modelling business processes and business rules is widely investigated. Other proposed transformation solutions do not cover all BPMN elements and their types, or do not give that kind of information. In addition, they do not cover the full SBVR model (business vocabulary and business rules), work separately from BPMN 2.0 model, so there is no possibility to create trace links between elements of models.

The paper presents an automatic transformation from BPMN 2.0 business process model into SBVR business vocabulary and business rules along with the evaluation of the proposed transformation solution using developed plug-in in the MagicDraw CASE tool. We defined transformation rules that allow representing BPMN 2.0 concepts by SBVR concepts and obtaining SBVR business rules. These rules can be applied to other CASE tools. The solution does not require sophisticated linguistic processing due to requirements for BPMN 2.0 elements. It allows linking two models (BPMN 2.0 and SBVR) with trace links in the same modelling environment.

The experimental evaluation of the proposed solu-

tion proved that the transformation is effective when modelers use the defined requirements and avoid excluded cases (C1-C5). It can, therefore, be concluded that the obtained SBVR business vocabulary and business rules can be used for validation purposes by domain experts.

Acknowledgement

The work is supported by the project VP1-3.1-ŠMM-10-V-02-008 "Integration of Business Processes and Business Rules on the Basis of Business Semantics" (2013-2015), which is funded by the European Social Fund (ESF).

References

1. Agrawal, A. Semantics of Business Process Vocabulary and Process Rules. In ISEC'11 proceedings of the 4th India Software Engineering Conference, 2011, 61-68. <https://doi.org/10.1145/1953355.1953363>
2. Goedertier, S., Vanthienen, J. Declarative Process Modeling with Business Vocabulary and Business Rules. OTM 2007 Ws, Part I, Springer-Verlag Berlin Heidelberg, 2007, 4805, 603-612.
3. Hohwiller, J., Schlegel, D., Grieser, G., Hoekstra, Y. Integration of BPM and BRM. Dijkman R, Hofstetter, J., Koehler, J. (Eds.) LNBIP, Springer, Heidelberg, 2011, 95, 136-141. https://doi.org/10.1007/978-3-642-25160-3_12
4. Koehler, J. The Process-Rule Continuum – Can BPMN & SBVR Cope with the Challenge? Commerce and Enterprise Computing (CEC), 2011 IEEE 13th Conference, Luxembourg, 2011, 302-309.
5. Koehler, J. The Process-Rule Continuum – How Can the BPMN and SBVR Standards Interplay? Lucerne University of Applied Sciences and Arts, Switzerland, 2011, 1-8.
6. Leopold, H., Mendling, J., Polyvyanyy, A. Generating Natural Language Texts from Business Process Models. Advanced Information Systems Engineering, Lecture Notes in Computer Science, Springer, 2012, 7328, 64-79. https://doi.org/10.1007/978-3-642-31095-9_5
7. Lopata, A., Veitaitė, I., Gudas, S., Butleris, R. Case Tool Component – Knowledge-Based Subsystem UML Diagrams Generation Process. Transformations in Business & Economics, 2014, 13(2B), 676-696.
8. Malik, S., Bajwa, Sarwan, I. A Rule Based Approach for Business Rule Generation from Business Process Model. Business Process Management Workshops, Rules on the Web: Research and Applications, LNCS, Springer-Verlag Berlin Heidelberg, 2012, 7438, 92-99.
9. Malik, S., Bajwa, Sarwan, I. Back to Origin: Transformation of Business Process Models to Business Rules. Business Process Management Workshops, LNBIP, 132, Springer-Verlag Berlin Heidelberg, 2012, 132, 611-622.
10. Mendling, J., Reijers, H. A., Recker, J. C. Activity Labeling in Process Modeling: Empirical Insights and Recommendations. Information Systems, 2010, 35(4), 467-482. <https://doi.org/10.1016/j.is.2009.03.009>
11. Mendling, J., Reijers, H. A., van der Aalst, W. M. P. Seven Process Modeling Guidelines (7PMG). Information and Software Technology, 2010, 52(2), 127-136. <https://doi.org/10.1016/j.infsof.2009.08.004>
12. Mendling, J., Reijers, H. A., Cardoso, J. What Makes Process Models Understandable? Business Process Management, 5th International Conference, BPM 2007, Brisbane, Australia, September 24-28, Proceedings, Lecture Notes in Computer Science, Springer, Brisbane, Australia, 2007, 4714, 48-63. https://doi.org/10.1007/978-3-540-75183-0_4

13. Mendling, J., Verbeek, H. M. W., van Dongen, B. F., van der Aalst, W. M., Neumann, G. Detection and Prediction of Errors in EPCs of the SAP Reference Model. *Data & Knowledge Engineering*, 2008, 64(1), 312-329. <https://doi.org/10.1016/j.datak.2007.06.019>
14. Mickeviciute, E., Nemuraite, L., Butleris, R. Improving BPMN2 Business Process Model to SBVR Business Vocabulary and Business Rules Transformation with BPMN2 Event Naming Patterns. *Information Technology and Control*, 2016, 45(4), 443-451.
15. Mickeviciute, E., Pavalkis, S., Nemuraite, L., Butleris, R. Using SBVR Profile for Integrating Business Vocabulary with BPMN Process Models. *International Journal of Advances in Computer Science & Its Applications*. New York: IRED. ISSN 2250-3765. 2014, 4(3), 108-113.
16. No Magic, Inc., UML Profiling and DSL, 2011. Available at: <http://www.nomagic.com/files/manuals/MagicDraw%20UMLProfiling&DSL%20UserGuide.pdf>. Accessed on October 17, 2013.
17. Normantas, K., Vasilecas, O. A Systematic Review of Methods for Business Knowledge Extraction from Existing Software Systems. *Baltic Journal of Modern Computing (BJMC)*, 2013, 1(1-2), 29-51.
18. OMG, Business Process Model and Notation (BPMN), v.2.0.1, OMG Document Number: formal/2013-09-02, Object Management Group (OMG), <http://www.omg.org/spec/BPMN/2.0.1>. Accessed on September 10, 2013
19. OMG, Semantics of Business Vocabulary and Business Rules (SBVR) Specification.v1.3, OMG Document Number: formal/2015-05-07, Object Management Group (OMG). <http://www.omg.org/spec/SBVR/1.3>. Accessed on June 29, 2015
20. Pinggera, J., Soffer, P., Fahland, D., Weidlich, M., Zugal, S., Weber, B., Reijers, H., Mendling, J. Styles in Business Process Modeling: An Exploration and a Model. *Software & Systems Modeling*, 2015, 14(3), 1055-1080. <https://doi.org/10.1007/s10270-013-0349-1>
21. Pinggera, J., Soffer, P., Zugal, S., Weber, B., Weidlich, M., Fahland, D., Mendling, J. Modeling Styles in Business Process Modeling. *Enterprise, Business-Process and Information Systems Modeling*, Springer Berlin Heidelberg, 2012, 113, 151-166.
22. Reijers, H. A., Mendling, J. A Study into the Factors that Influence the Understandability of Business Process Models. *IEEE Transactions on Systems, Man, and Cybernetics, Part A*, 2011, 41(3), 449-462.
23. Rodrigues, R. de A., Barros, M. de O., Revoredo, K., Azevedo, L. G., Leopold, H. An Experiment on Process Model Understandability Using Textual Work Instructions and BPMN Models, *Software Engineering (SBES)*, 2015 29th Brazilian Symposium, 2015, 41-50.
24. Silingas, D., Butleris, R. Towards Implementing a Framework for Modeling Software Requirements in MagicDraw UML. *Information Technology and Control*, 2009, 38(2), 153-164.
25. Silingas, D., Vitiutinas, R., Armonas, A., Nemuraite, L. Domain-Specific Modeling Environment Based on UML Profiles. *Information Technologies, Proceedings of the 15th International Conference on Information and Software Technologies, IT 2009*, Kaunas, Lithuania, April 23-24, 2009, 167-177.
26. Silver, B. Ten Tips for Effective Process Modeling, *Bpms Watch*, BPMInstitute.org, <http://www.bpminstitute.org/articles/articleasharticle/bpms-watch-ten-tips-foreffective-process-modeling.html>. Accessed on December 11, 2014.
27. Sinur, J. The Art and Science of Rules vs. Process Flows. *Research Report G00166408*, Gartner, 2009.
28. Skersys, T., Tutkutė, L., Butleris, R. The Enrichment of BPMN Business Process Model with SBVR. *Proceedings of the 34th International Conference, Information Technology Interfaces (ITI)*, 2012, 65-72.
29. Skersys, T., Butleris R., Kapocius, K., Vileiniskis, T. An Approach for Extracting Business Vocabularies from Business Process Models. *Information Technology and Control*, 2013, 42(2), 178-190. <https://doi.org/10.5755/j01.itc.42.2.2310>
30. Skersys, T., Danėnas, P., Butleris, R. Model-Based M2M Transformations Based on Drag-and-Drop Actions: Approach and Implementation. *Journal of Systems and Software*, 2016, 122, 327-341. <https://doi.org/10.1016/j.jss.2016.09.046>
31. Skersys, T., Tutkutė L., Butleris, R., Butkienė, R. Extending BPMN Business Process Model with SBVR Business Vocabulary and Rules. *Information Technology and Control*, 2012, 41(4), 356-367. <https://doi.org/10.5755/j01.itc.41.4.2013>
32. Wang, W., Indulska, M., Sadiq, S. Integrated Modeling of Business Process Models and Business Rules: A Research Agenda. *25th Australasian Conference on Information Systems*, 8th-10th Dec. 2014, Auckland, New Zealand, 2014, 1-10.

Summary / Santrauka

As the success of information system projects and their development also relies on the knowledge of domain experts, the modelled processes should be presented to them for validation purposes before the implementation stage. Domain experts do not have the knowledge of business process modelling notations. Therefore, the validation may not be correct. However, they better understand structured natural language, such as SBVR that defines the meaning of business concepts and make them unambiguously understandable by human experts and also by software systems. The solution presented in this paper allows transforming BPMN 2.0 business process models into SBVR business vocabularies and business rules. This solution is implemented as a plug-in and is available in MagicDraw CASE tool. An experimental evaluation of proposed solution with three domain areas proved that SBVR business vocabulary and business rules could be fully obtained from BPMN 2.0 business process model when certain requirements for BPMN 2.0 business process model are met. The advantages of the solution are an automatic transformation for various BPMN modelling situations, tracing links between two models in one modelling environment and avoidance of necessity to use linguistic techniques.

Informacinių sistemų projektai bei jų kūrimo sėkmė taip pat priklauso ir nuo dalykinės srities ekspertų žinių, todėl informacinių sistemų modeliai turėtų būti pateikiami jiems validuoti prieš realizavimo etapą. Dalykinės srities atstovai dažnai neturi žinių apie įvairias veiklos procesų modelių notacijas, todėl toks validavimas gali būti neteisingas. Be to, jie geriau supranta natūralią struktūrizuotą kalbą, kaip pavyzdžiui SBVR, kuri apibrėžia veiklos konceptų prasmę ir padaro juos vienareikšmiškai suprantamus visiems veiklos ekspertams, taip pat ir sistemoms. Šiame straipsnyje pateiktas sprendimas, kuris leidžia transformuoti BPMN 2.0 veiklos procesų modelius į SBVR veiklos žodynus bei veiklos taisykles. Šis sprendimas buvo realizuotas kaip įskiepis ir yra pasiekiamas MagicDraw CASE įrankyje. Eksperimentinis siūlomo sprendimo įvertinimas su trimis dalykinėmis sritimis parodė, kad SBVR veiklos žodynas ir veiklos taisyklės gali būti pilnai išgaunamos iš BPMN 2.0 veiklos procesų modelio, kai laikomasi tam tikrų veiklos procesams apibrėžtų reikalavimų. Šio sprendimo pranašumai yra tokie: automatinis transformavimas taikomas įvairioms BPMN modeliavimo situacijoms, atsekamumo ryšių užmezgimo galimybė vienoje modeliavimo aplinkoje ir lingvistinių metodų nenaudojimas.