Overview of BPMN Model Equivalences. Towards normalization of BPMN diagrams[1]

Krzysztof Kluza and **Krzysztof Kaczor**[2]

Abstract. In various application domains, there is a desire to standardize modeling techniques. Business Process Model and Notation (BPMN) is currently the most widespread language used for modeling Business Processes (BP). Although there are some guidelines how to use this notation, the issue of modeling technique is not standardized. The same semantics can be represented in BPMN using various but behaviorally equivalent model structures. In this paper, we present an overview of the BPMN models equivalences topic. We point out various possibilities of equivalence patterns. This can help to structure diagrams and decrease their semantic complexity. Such research can be further useful for such tasks as analyzing similarities or measuring compliance of processes.

1 Introduction

Business Process (BP) models constitute a graphical representation of processes in an organization. Business Process Model and Notation (BPMN)[3] [1, 23] is a notation for modeling Business Processes, which contributed significantly in Software Engineering when it comes to collaboration between developers, software architects and business analysts. Although there are many new tools and methodologies which support the BPMN notation, they neither support some recommended modeling techniques nor make BPMN models easily comprehensible.

Two models with different structure, but behaviorally equivalent, can be both correct and unambiguous. This stems from the BPMN specification allowing for expressing the same semantics using various syntactic structures. However, this can cause difficulties in modeling or understanding of the model – the *modeling challenge*.

Although behaviorally equivalent structures can be replaceable, some of them may be not translatable to other languages in order to be analyzed or verified [29, 33]. This makes practical problems with model analysis – the *analysis challenge*. Thus, to avoid such problems, a set of best practices for modelers is needed, and it would be useful to *normalize* the preffered model structures.

The first step towards such a structure normalization process is to identify behaviorally (or semantically) equivalent structures. One model can be transformed to the equivalent model to make it consistent in a way which it might not have been before [14]. While this may be done manually, and usually is in the case of ad hoc modeling, it is possible to support a normalization task with tools. The goals of such a normalization can be to maintain compatibility, interoperability, safety, repeatability, or quality of models.

[1] The paper is supported by the *BIMLOQ* Project funded from 2010–2012 resources for science as a research project.

[2] AGH University of Science and Technology, Poland, Email: `{kluza,kk}@agh.edu.pl`

[3] See: `http://www.bpmn.org/`.

Although there are several research papers concerning equivalences of Business Process models, most authors do not consider using of the BPMN notation, but analyze equivalences of models for Petri nets [5, 32] or web services [12, 25]. The thorough research in the area of BPMN models equivalences was carried by Vitus Lam and can be found in his papers [14, 15, 16]. Although Lam's equivalences of models are formalized, he analyzes only several equivalence patterns. Thus, it is advisable to address the issue of BPMN models equivalences in a wider range.

In this paper, we present an overview of the BPMN models equivalences and show various possibilities of equivalent structures. This research can be useful in different areas of BPMN application, such as: process matching [36], identifying the differences between process models [13], analyzing similarities [3, 6, 19] or measuring compliance of processes [2, 8].

The rest of this paper is organized as follows. In Section 2, BPMN models and elements are introduced. Section 3 provides a review of various equivalence patterns in BPMN models. The conclusion with suggested course of action is presented in Section 4.

2 BPMN models and elements

A Business Process [34] can be defined as a collection of related tasks that produce a specific service or product (serve a particular goal) for a particular customer. BPMN constitute the most widespread language for modeling BPs. It uses a set of predefined graphical elements to depict a process and how it is performed. The current BPMN 2.0 specification defines three models to cover various aspects of processes:

1. *Process Model* — describes the ways in which operations are carried out to accomplish the intended objectives of an organization. The process can be modeled on different abstraction levels: *public* (collaborative Business 2 Business Processes) or *private* (internal Business Processes).
2. *Choreography Model* — defines expected behavior between two or more interacting business participants in the process.
3. *Collaboration Model* — can include Processes and/or Choreographies, and provides a Conversation view (which specifies the logical relation of message exchanges).

In most cases, using only the Process Model is sufficient. In our research, the internal Business Process Model is considered. Four basic categories of elements used to model such processes, presented in Fig. 1, are: flow objects (*activities*, *gateways*, and *events*), connecting objects (*sequence flows*, *message flows*, and *associations*), swimlanes, and artifacts [23].

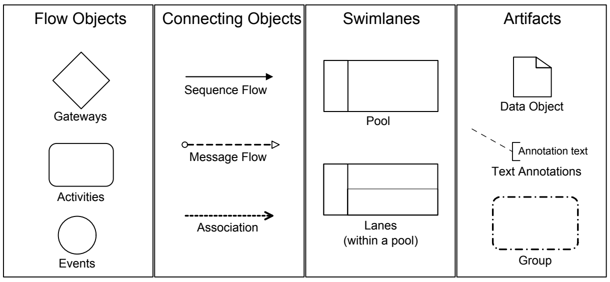


Figure 1. BPMN core objects

Activities constitute the main BPMN elements. They denote tasks that have to be performed and are represented by rectangles with rounded corners. The sequence flow between activities, the flow of control, is depicted by arcs. The directions of arcs depict the order in which the activities have to be performed.

Events, represented by circles, denote something that happens during the lifetime of the process. The icon within the circle denotes the event type, e.g. envelope for *message event*, clock for *time event*.

Gateways, represented by diamond shapes, determine forking and merging of the sequence flow between tasks in a process, depending on some conditions.

3 Equivalences of BPMN Models

In various application domains there is a need to compare process models [32]. One of the possible results of such a comparison can be that two structurally different graphical representations of a business process are behaviorally (and semantically) equivalent. Thus, BPMN processes can be regarded as equivalent if both of them can be transformed into a common graphical representation [14].

There is ongoing research in the area of process models equivalences [5, 32, 35]. However most of the researchers do not consider BPMN notation, but e.g. Petri nets [5, 32]. There are tools which can prove selected equivalences of BPMN processes [14]. However, this topic still remains an open research problem [16].

3.1 Basic equivalent structures

Some basic equivalences that follow directly from the semantics of model elements described in the BPMN specification [23] are presented in Table 1.

Other basic equivalences have been presented by Wohed et al. [35] when defining the five simple control-flow patterns for process control based on the concepts defined by Workflow Management Coalition [4], such as:

1. *sequence* — the ability to depict a sequence of activities,
2. *parallel split* — the ability to capture a split in a single thread of control into multiple threads which can execute in parallel,
3. *synchronization* — the ability to capture a synchronization of multiple parallel subprocesses/activities into a single thread,
4. *exclusive choice* — the ability to represent a decision point in a workflow process where one of several branches is chosen,
5. *simple merge* — the ability to depict a point in the workflow process where two or more alternative branches come together without synchronization.

Apart from the sequence, the other patterns can be modeled in several ways. The models in each column of the Table 2 are equivalent.

One can also observe that in many cases multiple gateway structure can be replaced by a single gateway, as shown in Table 3. Moreover, Gruhn and Laue described patterns in BPMN models that deal with OR-gateways which can be replaced by AND- or XOR-gateways [9], as presented in Table 4 (each row contains an equivalent pair of structures). They claimed that the equivalent model is easier to understand, as it is cognitively less complex. Such transformation is also consistent with a study on the comprehensibility of BPM carried out by Sarshar and Loos [28], which shows that OR-gateways are significantly less comprehended than AND or XOR gateways. Thus, Mendling et al. recommended to avoid OR-gateways [20].

Several researchers noticed that in several situations it is possible to reduce number of repeated activities [14, 17]. The first example in Table 5 shows a situation where the same activity is located at the last position of all incoming sequence flow paths before a join gateway. It is possible to reduce the number of nodes by moving this activity behind the join gateway. The second one is similar but concerns a situation in which the repeated activity is located at the first position after a split gateway.

In [10], Jung et al. proposed a transformation from the BPMN-formed business process to its semantically equivalent XPDL process. Although both BPMN and XPDL are conceived of as a directed graph structure and the mapping should be straightforward, there are some differences between BPMN and XPDL. Thus, in the paper [10] several BPMN transformations are considered.

One of them concerns a loop mechanism. A loop in a process can be depicted as in Fig. 2a. The BPMN 2.0 specification defines the "testBefore" standard loop attribute, which constitutes a flag that controls whether the loop condition is evaluated at the beginning (testBefore = true) or at the end (testBefore = false) of the loop iteration. Instead of using this attribute, a loop can be depicted explicitly as in Fig. 2b (test time: before) and Fig. 2c (test time: after).

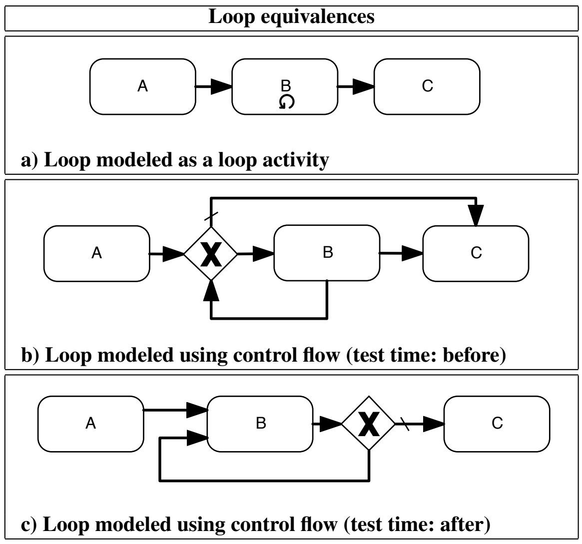


Figure 2. Variants of a loop structure [10]

Another transformation of loops in graphs was proposed by Zhongjun Du and Zhengjun Dang in [7]. Based on the graph reduction technique [27], they proposed an algorithm which transforms the loop in the workflow to an acyclic sub-graph. Although their solution does not use BPMN, it is rather general and should be applicable to BPMN models as well.

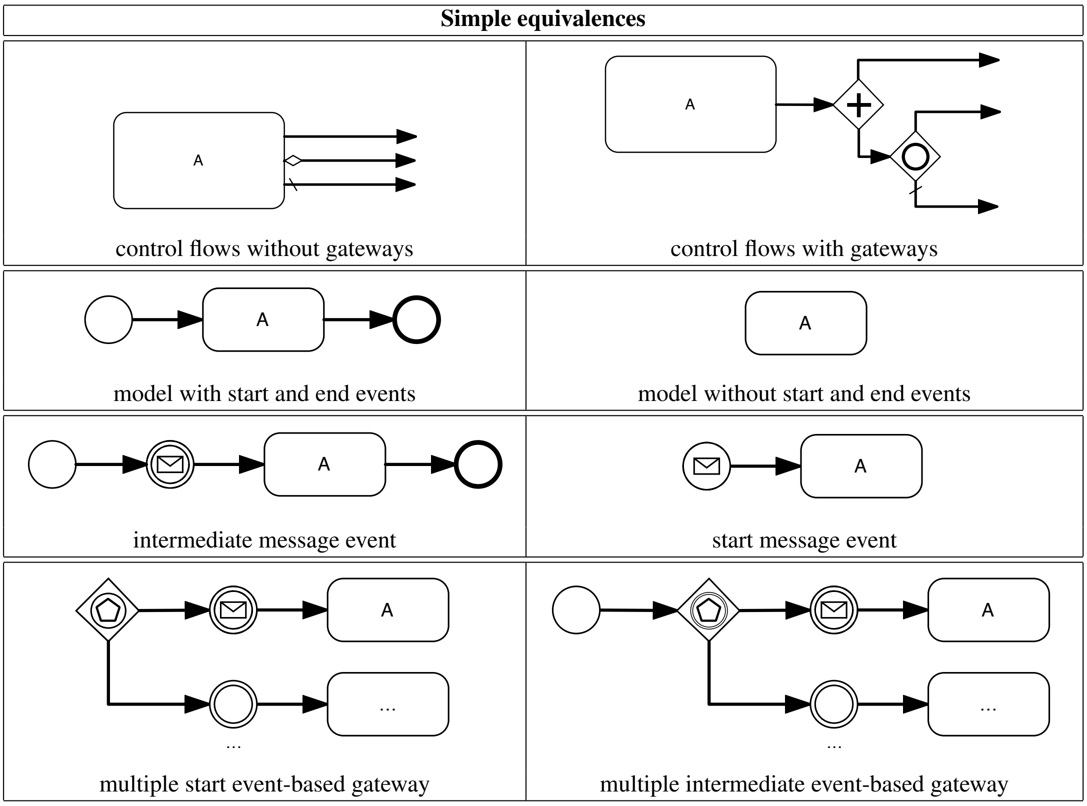


Table 1. Equivalences of BPMN structures based on the semantics of elements (based on the BPMN specification [23])

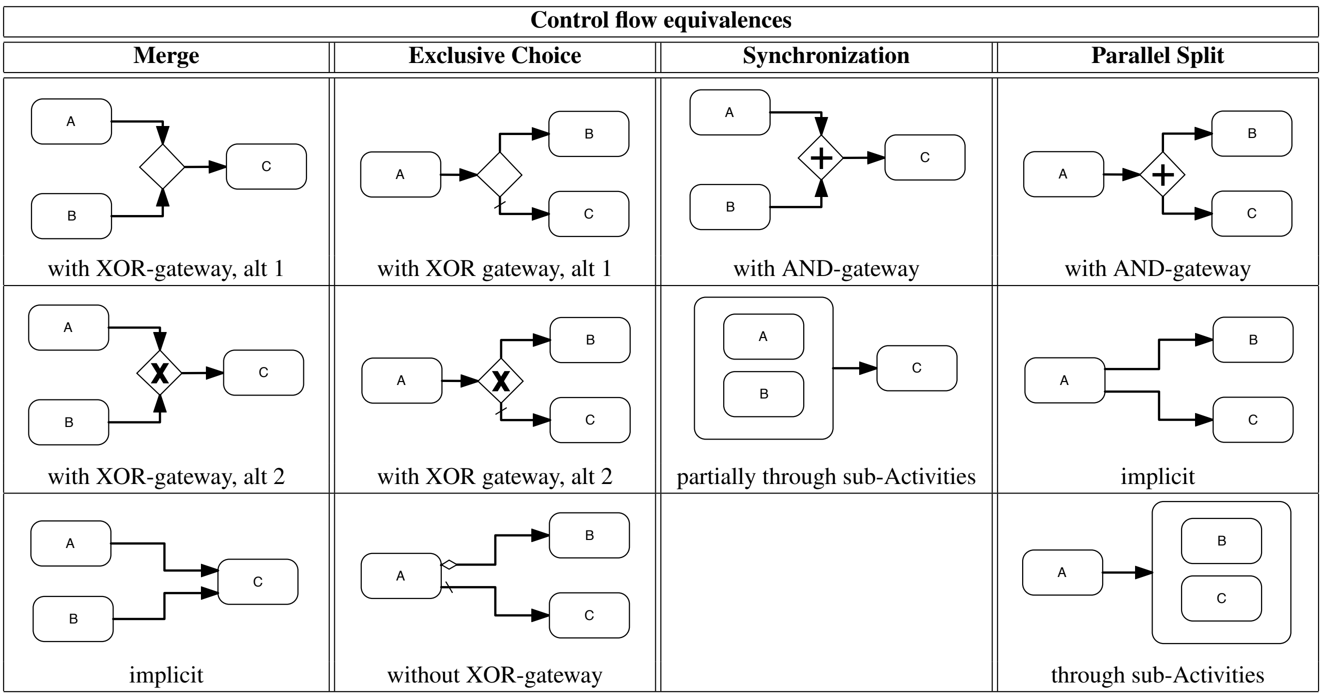


Table 2. Basic control-flow patterns in BPMN [35]

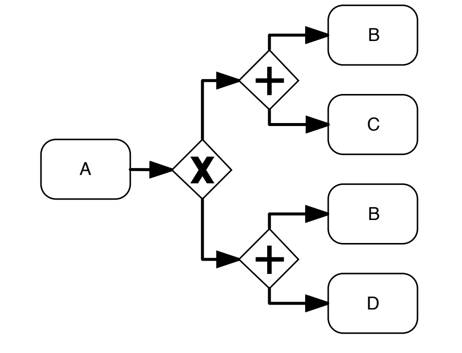
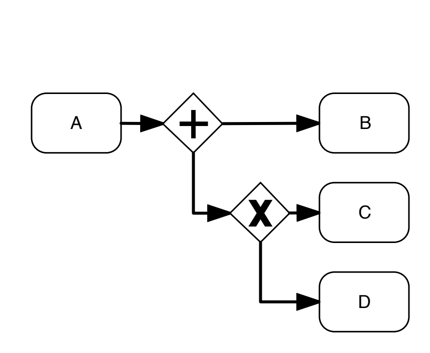
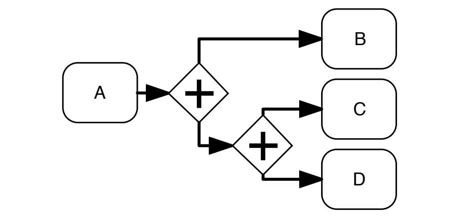
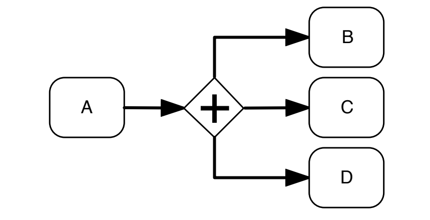
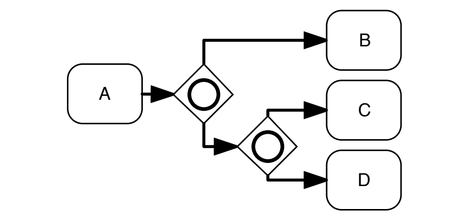
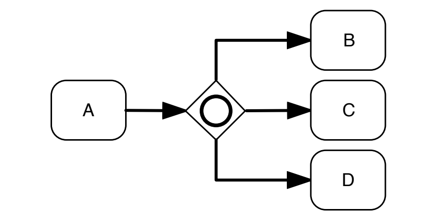
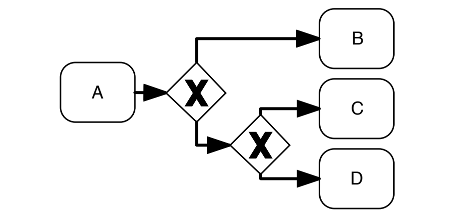
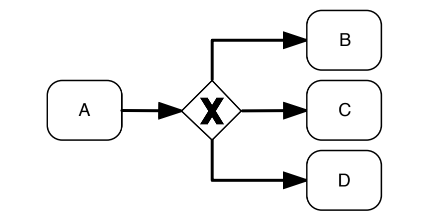
Multiple gateways equivalences	
 multiple parallel gateway with a common task	 single parallel gateway
 multiple parallel gateway	 single parallel gateway
 multiple inclusive gateway	 single inclusive gateway
 multiple exclusive gateway	 single exclusive gateway

Table 3. Equivalences of BPMN structures based on multiple gateway elements

3.2 Complex Equivalences of BPMN structures

Other transformations considered in [10] concern discrimination and serialization mechanisms. In Table 6 several examples of the application of the discriminator transformation to selected BPMN elements are presented. The serialization examples, which transform something serialized implicitly to another thing serialized explicitly, are shown in Table 7.

Qing-xiu et al. [24], in order to verify a workflow model based on Petri net, proposed several reduction actions, such as reduction of sequential, iterative, or adjacent structure. However, the proposed reductions are not directly applicable to BPMN models.

Tantitharanukul and Jumpamule [31] defined Generalized Business Process Modeling Notation (GBPMN) as a notation for diagrams which nodes are labeled with the process expression. They presented an algorithm which converts any BPMN into GBPMN form. It is important to mention that the GBPMN is not a standardized solution, thus it is not very useful in practice. However, one of the steps of their algorithm is taken if the existing diagram has more than one start event or end event. In such a case, they stipulate adding a new single start event and/or a new single end event, and connecting these events to the existing diagrams by using inclusive gateway which is capable of capturing whether they simultaneously start or not. Using single start and end events should be taken into account when modeling, and such a procedure should be considered as a part of a normalization algorithm for business processes as well.

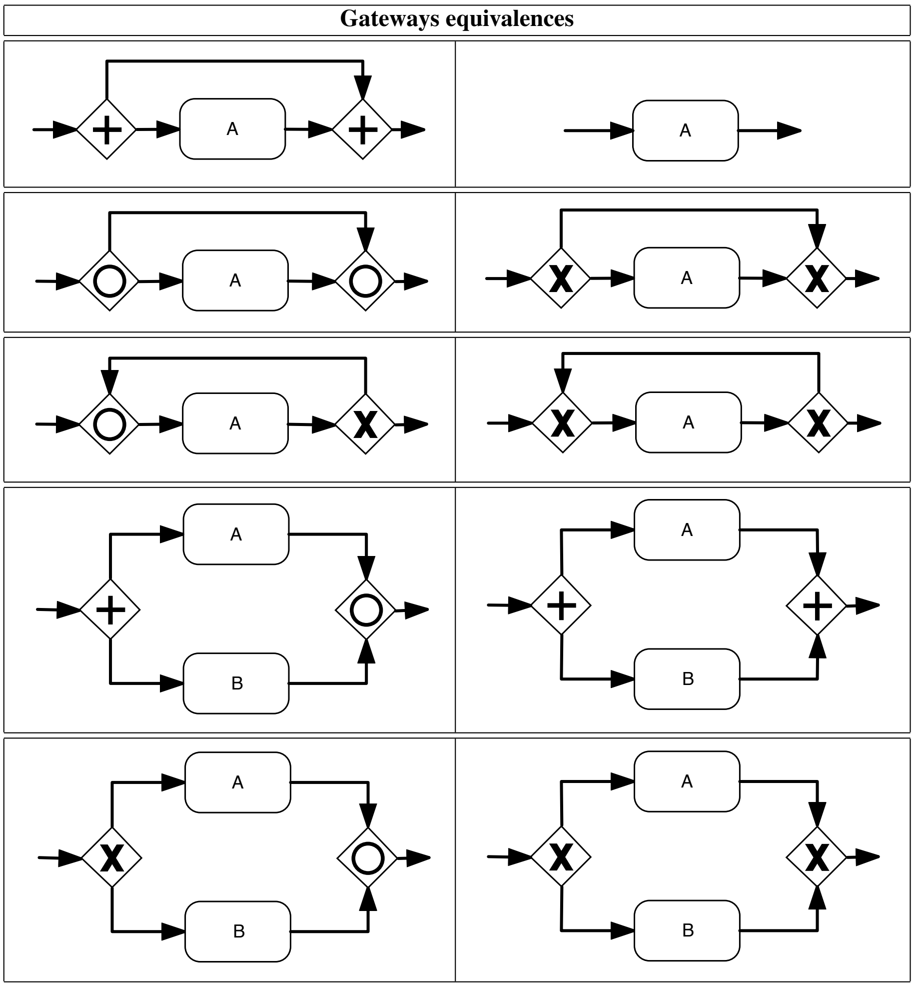


Table 4. Gateways equivalences of BPMN structures (based on [9])

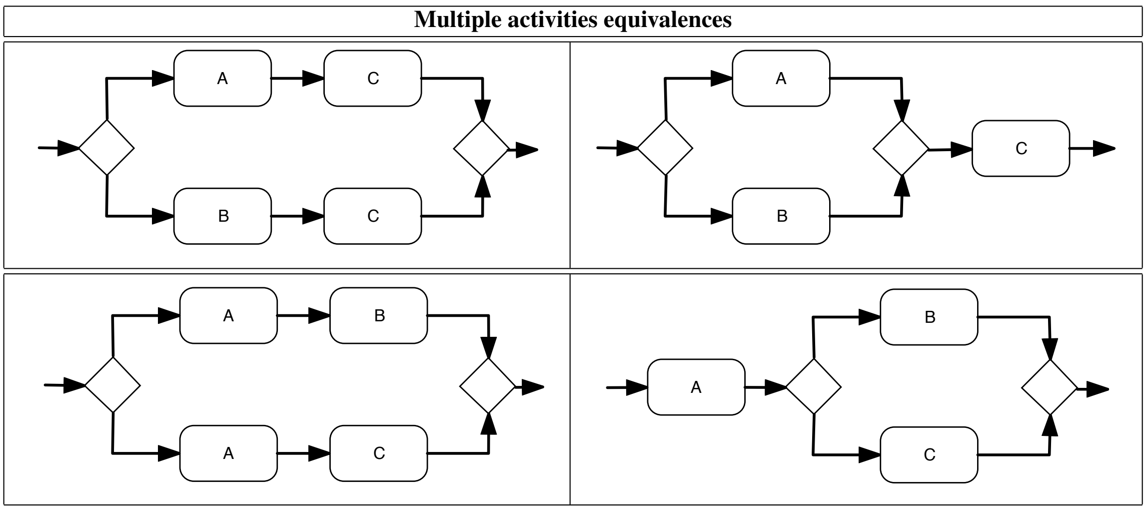


Table 5. Multiple activities equivalences of BPMN structures (based on [14, 17])

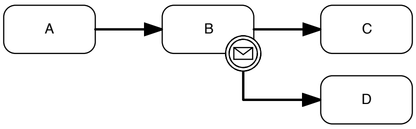
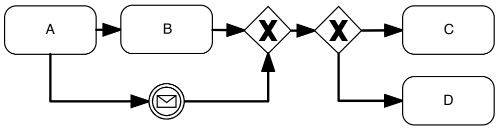
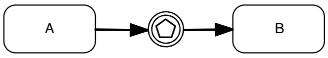
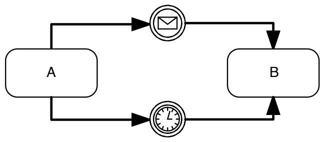
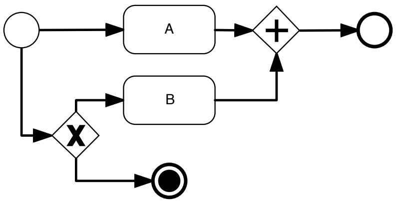
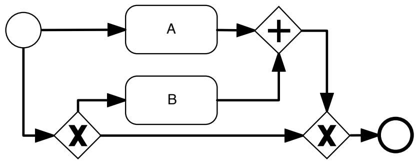
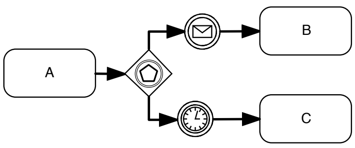
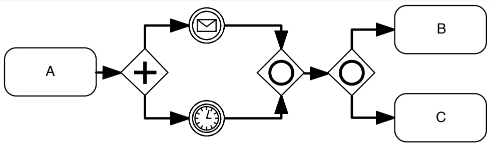
Discriminator equivalences of events	
 boundary intermediate event	 intermediate event in normal flow
 multiple event	 a number of single events
Discriminator equivalences of gateways	
 termination of a process using terminate event	 normal process termination
 multiple event gateway	 a combination of gateways and single events

Table 6. Discriminator equivalences of BPMN structures (based on [10])

3.3 Guidelines for modelers

The normalization process should also take into account the existing guidelines for business modelers. Most of the existing tools do not require to comply with any guidelines or modeling requirements, so a user has to adhere to them itself.

One of the papers with most impact in the business process modeling field by Mendling et al. [20] concerns guidelines for business process modelers, which should be taken into account when modeling business processes. They formulated seven guidelines and prioritized them with the help of industry experts [20]:

1. Model as structured as possible.
2. Decompose a model with more than 50 elements.
3. Use as few elements in the model as possible.
4. Use verb-object activity labels.
5. Minimize the routing paths per element.
6. Use one start and one end event.
7. Avoid OR routing elements.

La Rosa et al. [26] performed a systematic analysis and proposed a number of concrete syntax modifications for business process models to manage their complexity. They presented a collection of patterns that generalize and conceptualize various existing mechanisms to change the visual representation of a process model. Their goal was to simplify the representation of processes. Thus, they identified eight patterns which reduce the perceived model complexity without changing the abstract syntax of the model and classified them according to the following hierarchy [26]:

1. Layout Guidance — describes features to modify the process model layout.
2. Outline visual mechanisms to emphasize certain aspects:
   (a) Enclosure Highlight — for visually enclosing close a set of logically related model elements,
   (b) Graphical Highlight — to change the visual appearance of model elements, such as shape, line thickness and type, etc.
   (c) Pictorial and Textual Annotation — to assign pictorial elements, such as icons or images, to modeling elements, or to visually represent free-form text in the canvas, which can be attached to modeling elements without changing semantics.
3. Two representation patterns:
   (a) Explicit Representation — to capture process modeling concepts via a dedicated graphical notation,
   (b) Alternative Representation — to capture process modeling concepts without the use of their primary graphical notation.
4. Naming Guidance — naming conventions or advice for model elements' labels, which can be syntactic (e.g. using a verb-object style) or semantic (e.g. using a domain-specific vocabulary).

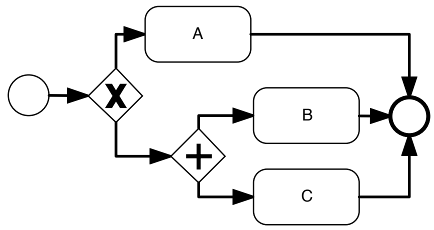
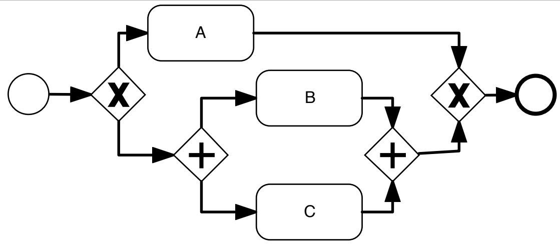
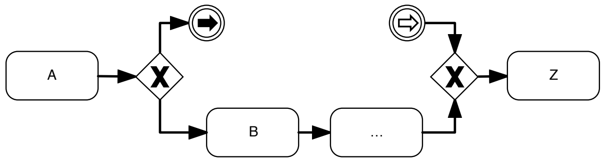
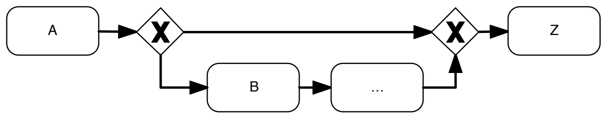
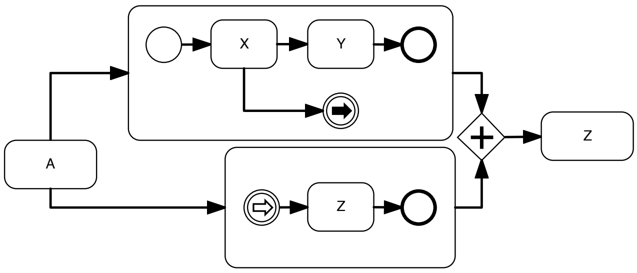
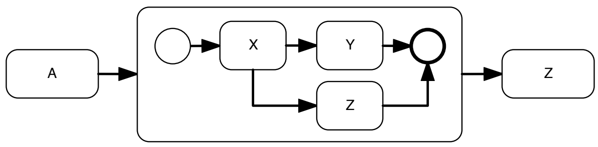
Serialization equivalences for gateways	
 implicit join	 explicit join
Serialization equivalences for links	
 internal links in a model	 a model without links
 external links in a model	 a model without links

Table 7. Serialization equivalences of BPMN structures (based on [10])

4 Conclusion

Although BPMN is the most widespread notation used by software architects and business analysts for modeling Business Processes, it is not clear which structures should be preferred and which avoided. The BPMN specification does not clarify how the notation should be used for modeling various processes. Thus, the standardization of such modeling technique in BPMN is desired.

As BPMN allows for expressing the same semantics using various syntactic structures, this can cause the modeling and analysis challenges. Cognitive understanding of model semantics can vary in case of complex syntactic differences. Furthermore, a behaviorally equivalent but syntactically different structures can be analyzed in different ways or even can be untranslatable to other languages in order to be verified. To address these issues, a set of best practices for modelers as well as normalization of BPMN models are needed.

In this paper, we prepared the first step towards such a normalization process – based on a literature review, we presented an overview of the topic of BPMN models equivalences, identified various behaviorally (or semantically) equivalent structures, and pointed out possibilities of equivalent patterns.

Moreover, we presented several guidelines for modelers, which should be taken into account when modeling, and considered as a part of a normalization algorithm for business processes.

While normalization can be performed manually, and usually is in the case of ad hoc modeling, it is possible to support such a process with tools. However, most of the existing tools do not require to comply with any guidelines or modeling requirements, so a user has to adhere to them itself.

Furthermore, normalization can help in the future research on structuring diagrams in order to decrease their semantic complexity. Our research can be further useful for many purposes, such as process matching, identifying the differences between process models, analyzing similarities or measuring compliance of processes.

In our future research, we will formalize the presented equivalences. This will allow for implementing a tool for proving that two models are equivalent or using some of the existing tools for analyzing BPMN patterns for this purpose [15, 17, 18, 30]. Our goal is to define the preferable structures of the model, which will constitute a normalization process and a part of a modeling methodology for modeling business processes integrated with rules [22, 21]. Such process can be further supported by a proper tool framework [11].

References

[1] Thomas Allweyer, *BPMN 2.0. Introduction to the Standard for Business Process Modeling*, BoD, Norderstedt, 2010.

[2] Ahmed Awad, Matthias Weidlich, and Mathias Weske, 'Visually specifying compliance rules and explaining their violations for business processes', *Journal of Visual Languages & Computing*, **22**(1), 30–55, (2011). Special Issue on Visual Languages and Logic.

[3] Michael Becker and Ralf Laue, 'A comparative survey of business process similarity measures', *Computers in Industry*, **63**(2), 148–167, (Feb 2012).

[4] Workflow Management Coalition, 'Workflow management coalition terminology & glossary', Technical Report WFMC-TC-1011, WfMC, United Kingdom, (Feb 1999).

[5] Ana Karla Alves de Medeiros, Wil M. P. van der Aalst, and A. J. M. M. Weijters, 'Quantifying process equivalence based on observed behavior', *Data Knowl. Eng.*, **64**(1), 55–74, (2008).

[6] Remco Dijkman, Marlon Dumas, Boudewijn van Dongen, Reina Käärik, and Jan Mendling, 'Similarity of business process models: Metrics and evaluation', *Information Systems*, **36**(2), 498–516, (Apr 2011).

[7] Zhongjun Du and Zhengjun Dang, 'A new algorithm based graph-search for workflow verification', in *Proceedings of the 2nd International Conference on Information Engineering and Computer Science (ICIECS), 25-26 Dec. 2010*, pp. 1–3. IEEE, (2010).

[8] Kerstin Gerke, Jorge Cardoso, and Alexander Claus, 'Measuring the compliance of processes with reference models', in *On the Move to Meaningful Internet Systems: OTM 2009*, eds., Robert Meersman, Tharam Dillon, and Pilar Herrero, volume 5870 of *Lecture Notes in Computer Science*, 76–93, Springer Berlin / Heidelberg, (2009).

[9] V. Gruhn and R. Laue, 'Reducing the cognitive complexity of business process models', in *Proceedings from the 8th IEEE International Conference on Cognitive Informatics, 15-17 June 2009. ICCI'09.*, pp. 339–345, (2009).

[10] Moonyoung Jung, Hak Soo Kim, Myung Hyun Jo, Kyung Hyun Tak, Hyun Suk Cha, and Jin Hyun Son, 'Mapping from BPMN-formed business processes to XPDL business processes', in *Proceedings of the Fourth International Conference on Electronic Business – Shaping Business Strategy in a Networked World ICEB*, pp. 422–427. Academic Publishers/World Publishing Corporation, (2004).

[11] Krzysztof Kluza, Krzysztof Kaczor, and Grzegorz J. Nalepa, 'Enriching business processes with rules using the Oryx BPMN editor', in *Artificial Intelligence and Soft Computing: 11th International Conference, ICAISC 2012: Zakopane, Poland, April 29–May 3, 2012*, eds., Leszek Rutkowski and [et al.], volume 7268 of *Lecture Notes in Artificial Intelligence*, pp. 573–581. Springer, (2012).

[12] Li Kuang, 'A formal analysis of behavioral equivalence for web services', in *Proceedings from the IEEE Congress on Services - Part I, 2008*, pp. 265–268, (2008).

[13] Min-Hsun Kuo and Yun-Shiow Chen, 'A method to identify the difference between two process models', *Journal of Computers*, **7**(4), 998–1005, (2012).

[14] Vitus S. W. Lam, 'Equivalences of BPMN processes', *Service Oriented Computing and Applications*, **3**(3), 189–204, (2009).

[15] Vitus S. W. Lam, 'Formal analysis of BPMN models: a NuSMV-based approach', *International Journal of Software Engineering and Knowledge Engineering*, **20**(7), 987–1023, (2010).

[16] Vitus S. W. Lam, 'Foundation for equivalences of BPMN models', *Theoretical and Applied Informatics*, **24**(1), 33–66, (2012).

[17] Ralf Laue and Ahmed Awad, 'Visual suggestions for improvements in business process diagrams', *Journal of Visual Languages & Computing*, **22**(5), 385–399, (2011).

[18] Antoni Ligęza, 'BPMN – a logical model and property analysis', *Decision Making in Manufacturing and Services*, **5**(1-2), 57–67, (2011).

[19] N.M.b. Mahmod and S.b.A. Radzi, 'An approach to analyse similarity of business process variants', in *Proceedings from the IEEE International Conference on Progress in Informatics and Computing (PIC), 2010*, pp. 640–644, (2010).

[20] J. Mendling, H. A. Reijers, and W. M. P. van der Aalst, 'Seven process modeling guidelines (7pmg)', *Information & Software Technology*, **52**(2), 127–136, (Feb 2010).

[21] Grzegorz J. Nalepa, 'Proposal of business process and rules modeling with the XTT method', in *Symbolic and numeric algorithms for scientific computing, 2007. SYNASC Ninth international symposium. September 26–29*, eds., Viorel Negru and et al., pp. 500–506, Los Alamitos, California ; Washington ; Tokyo, (september 2007). IEEE Computer Society, IEEE, CPS Conference Publishing Service.

[22] Grzegorz J. Nalepa, Krzysztof Kluza, and Sebastian Ernst, 'Modeling and analysis of business processes with business rules', in *Business Process Modeling: Software Engineering, Analysis and Applications*, ed., J.A. Beckmann, Business Issues, Competition and Entrepreneurship, 135–156, Nova Science Publishers, (2011).

[23] OMG, 'Business Process Model and Notation (BPMN): Version 2.0 specification', Technical Report formal/2011-01-03, Object Management Group, (January 2011).

[24] Liu Qing-xiu, Cao Bao-xiang, and Zhao Yi-wei, 'An improved verification method for workflow model based on petri net reduction', in *Proceedings of the 2nd IEEE International Conference on Information Management and Engineering (ICIME), 2010*, pp. 252–256. IEEE, (2010).

[25] S. Rinderle-Ma, M. Reichert, and M. Jurisch, 'Equivalence of web services in process-aware service compositions', in *Proceedings from the IEEE International Conference on Web Services, 2009. ICWS 2009*, pp. 501–508, (2009).

[26] Marcello La Rosa, Arthur H. M. ter Hofstede, Petia Wohed, Hajo A. Reijers, Jan Mendling, and Wil M. P. van der Aalst, 'Managing process model complexity via concrete syntax modifications', *IEEE Transactions on Industrial Informatics*, **7**(2), 255–265, (2011).

[27] Wasim Sadiq and Maria E. Orlowska, 'Analyzing process models using graph reduction techniques', *Information Systems*, **25**(2), 117–134, (2000).

[28] Kamyar Sarshar and Peter Loos, 'Comparing the control-flow of epc and petri net from the end-user perspective', in *Business Process Management*, eds., Wil van der Aalst, Boualem Benatallah, Fabio Casati, and Francisco Curbera, volume 3649 of *Lecture Notes in Computer Science*, 434–439, Springer Berlin / Heidelberg, (2005).

[29] Marcin Szpyrka, Grzegorz J. Nalepa, Antoni Ligęza, and Krzysztof Kluza, 'Proposal of formal verification of selected BPMN models with Alvis modeling language', in *Intelligent Distributed Computing V. Proceedings of the 5th International Symposium on Intelligent Distributed Computing – IDC 2011, Delft, the Netherlands – October 2011*, ed., Frances M.T. Brazier et al., volume 382 of *Studies in Computational Intelligence*, 249–255, Springer-Verlag, (2011).

[30] N. Tantitharanukul, P. Sugunnasil, and W. Jumpamule, 'Detecting deadlock and multiple termination in bpmn model using process automata', in *Electrical Engineering/Electronics Computer Telecommunications and Information Technology (ECTI-CON), 2010 International Conference on*, pp. 478–482, (May 2010).

[31] Nasi Tantitharanukul and Watcharee Jumpamule, 'Detection of livelock in BPMN using process expression', in *Advances in Information Technology*, eds., Borworn Papasratorn, Kittichai Lavangnananda, Wichian Chutimaskul, and Vajirasak Vanijja, volume 114 of *Communications in Computer and Information Science*, 164–174, Springer Berlin Heidelberg, (2010).

[32] Wil M. P. van der Aalst, Ana Karla A. de Medeiros, and A. J. M. M. Weijters, 'Process equivalence: Comparing two process models based on observed behavior', in *Business Process Management, 4th International Conference, BPM 2006, Vienna, Austria, September 5-7, 2006, Proceedings*, volume 4102 of *Lecture Notes in Computer Science*, pp. 129–144, (2006).

[33] Matthias Weidlich, Gero Decker, Alexander Grosskopf, and Mathias Weske, 'Bpel to bpmn: The myth of a straight-forward mapping', in *Proceedings of the OTM 2008 Confederated International Conferences, CoopIS, DOA, GADA, IS, and ODBASE 2008. Part I on On the Move to Meaningful Internet Systems*, OTM '08, pp. 265–282, Berlin, Heidelberg, (2008). Springer-Verlag.

[34] Stephen A. White and Derek Miers, *BPMN Modeling and Reference Guide: Understanding and Using BPMN*, Future Strategies Inc., Lighthouse Point, Florida, USA, 2008.

[35] Petia Wohed, Wil M. P. van der Aalst, Marlon Dumas, Arthur H. M. ter Hofstede, and Nick Russell, 'On the suitability of bpmn for business process modelling', in *Business Process Management, 4th International Conference, BPM 2006, Vienna, Austria, September 5-7, 2006, Proceedings*, volume 4102 of *Lecture Notes in Computer Science*, pp. 161–176, (2006).

[36] Jian Zhu and Hung Keng Pung, 'Process matching: A structural approach for business process search', in *Future Computing, Service Computation, Cognitive, Adaptive, Content, Patterns, 2009. COMPUTATIONWORLD '09. Computation World*, pp. 227–232, (2009).