

Towards Rule-based Pattern Perspective for BPMN 2.0 Business Process Models

Krzysztof Kluza, Grzegorz J. Nalepa
AGH University of Science and Technology
al. A. Mickiewicza 30, 30-059 Krakow, Poland
E-mail: {kluza,gjn}@agh.edu.pl

Abstract—Rules and processes constitute powerful representation forms. Although the same notions can be expressed in both of these representations, there is a significant difference in abstraction levels between processes and rules. In practice, rules are mostly used for the specification of rule task logic in processes. In this paper, we present various options where and how rules can be perceived in business processes. We introduce rule-based pattern perspective for process models, focusing on BPMN 2.0 Business Process Models.

Index Terms—BPMN, Business Processes, Business Process Hierarchization, Business Process Configuration

I. INTRODUCTION

BUSINESS Process (BP) [1] models constitute visual representations of processes of an organization. BPMN is a dominant visual modeling language used for describing processes. A BPMN model should be easy to understand for non-business users. However, sometimes such a model becomes illegible or unclear due to its complexity. One of the ways of reducing complexity is to introduce rules in order to specify low level business logic of the process. Such Business Rules (BR) [2] can describe business policies, goals, strategies or guidelines, in a form of declarative statements, constraints or predicated actions. Another issue of improving comprehensibility of the model is to use the notation in a proper way.

As BPMN specifies only a notation, thus there can be several ways of using it. There are style directions how to model BPs [3], or guidelines for analysts based on BPs understandability (e.g. [4]). However, a proper business process modeling is still a challenging task, especially for inexperienced users.

Design patterns in software engineering propose reusable solutions independent from the implementation technology. Similarly, workflow patterns address business requirements independently from specific workflow languages, and describe the problem, conditions that should hold for the pattern in order to be applicable, examples of business situation, as well as its realization or implementation [5]. As the existing workflow patterns perspectives does not take into account rules, we propose rule-based pattern perspective for process models.

The paper is organized as follows. In Section II we present the motivation for our research. Section III provides a short overview of related approaches. In Section IV we present our proposal of Rule-based Pattern Perspective for BPMN Process Models. Section V summarizes the paper.

II. MOTIVATION

Although it is possible to express the same notions (concepts, objects, system behaviour description etc.) using processes as well as rules. These two approaches are not suitable for this purpose, as they use different languages, depict different aspects of a system, and thus there is a significant difference in abstraction levels between processes and rules. However, this is neither standardized nor obvious which method is better for expressing a particular semantics.

During modeling a system using processes and rules, several questions can arise, from the simple ones like which representation is better for a specific purpose, through the integration of processes and rules, to detail aspects of representation pragmatics. Thus, in order to organize the knowledge about process models with rules, rule-based pattern perspective for processes is introduced. Based on workflow patterns, we selected several attributes which are applied to rule-based patterns in process models. Presenting the patterns, we focused on their BPMN representation. Such patterns can help in designing models with rules and increase the comprehension level of the model.

Similarly to other languages, in the BPMN modeling language, three aspects can be distinguished: syntax, semantics and pragmatics. In the case of syntax, the current BPMN 2.0 specification [6] describes it in detail, and our research is consistent with the specification describing the syntax of rule related elements (event, tasks). BPMN semantics have also been considered in several papers [7], however, not with an emphasis on rules. As in practice, the pragmatics of language use is essential, a short overview of the BPMN pragmatics is presented in the following section.

III. RELATED WORKS

In the case of the BPMN pragmatics the following particular issues can be considered: modeling techniques and styles, workflow patterns, as well as domain patterns. These issues are elaborated in the next subsections.

A. Workflow Patterns and Domain Process Patterns

As design patterns provide reusable solutions [9] independent from the implementation technology, workflow patterns address business requirements independently from specific workflow languages. Such patterns describe conditions that should hold for them in order to be applicable, examples of business situation, description of the problem, as well as realization or implementation in current languages [5].

Table I
FRAGMENT OF IMPLICIT TERMINATION PATTERN SPECIFICATION¹

Implicit Termination Pattern	
Description:	<i>A given process (or sub-process) instance should terminate when there are no remaining work items that are able to be done either now or at any time in the future and the process instance is not in deadlock. There is an objective means of determining that the process instance has successfully completed.</i>
Motivation:	<i>The rationale for this pattern is that it represents the most realistic approach to determining when a process instance can be designated as complete. This is when there is no remaining work to be completed as part of it and it is not possible that work items will arise at some future time.</i>
Solutions:	<i>For simple process models, it may be possible to indirectly achieve the same effect by replacing all of the end nodes for a process with links to an OR-join which then links to a single final node. However, it is less clear for more complex process models involving multiple instance tasks whether they are always able to be converted to a model with a single terminating node. Potential solutions to this are discussed at length by Kiepuszewski et al. [8].</i>

Originally, van der Aalst et al. identified a set of 26 patterns that describe the control-flow perspective of business processes [5], [10]. Table I presents an example of a typical structural control-flow pattern for implicit termination of a process¹. In the case of the BPMN notation, this pattern is directly supported from the very beginning (version BPMN 1.0) by ending every thread of a process with an end event. When the last token in the process generated by the start event is consumed, the process instance terminates.

Over the years, the existing workflow patterns have been evaluated [11], [12], [13], revised [14], [15], and extended to cover new perspectives like the data and resource perspectives [16], [17], or time perspective [18].

Until now, much more workflow patterns have been identified. The Workflow Patterns Initiative² distinguishes:

- 42 control-flow patterns,
- 43 workflow resource patterns,
- 40 workflow data patterns,
- 12 abstract and 8 concrete syntax modification patterns,
- over 100 exceptions patterns of various exception types.

Such patterns serves not only as good practices but also for a pattern-based evaluation of tools or standards [19], [20], [21].

What is more, one can also distinguish patterns concerning the process of modeling itself [22], [23], like fixation patterns during process model creation [24], or abstract and concrete syntax modifications patterns [25], [26].

Another kind of patterns – Domain Process Patterns (DPP) [27] – represent functions of process model fragments that are applicable to some modeling domain. They were introduced as a result of the investigation of business processes from the order management and the manufacturing production domains. Such patterns describes some domain related business operations representing a small fragment of the process. Although it is possible to consider involving rules in DPP, e.g. in *Inventory Pattern* one can consider rules involved in inventory management, it is not a purpose of DPP.

¹An example can be accessed at <http://www.workflowpatterns.com/patterns/control/structural/wcp11.php>.

²See: <http://www.workflowpatterns.com>.

Neither workflow patterns nor domain process patterns do not consider the integration of processes with rules.

B. Business Rule Patterns

A business rule taxonomy that serves as a rich source of business rules can be found in [28], [29]. The authors specified more than 60 business rule structures that for a specific process instance: restrict the number of allowed instances of a specific process elements, restrict the coexistence of process elements of different types, specify the influence of specific data elements on the occurrence of process elements, a time restriction on process elements, or a property for a process element at a predefined process state. These rules were used for a comprehensive rule-based compliance checking approach. However, they were not analyzed from the business process representation perspective.

In the following section, we present a short overview of 10 selected rule-based patterns, analyze the rules from the business process representation perspective, and show how they can be observed in the BPMN process models.

IV. RULE-BASED PATTERNS IN BPMN PROCESS MODELS

In order to specify the rule patterns in BPMN process models, we use the following attributes:

- 1) **Pattern Name:** Descriptive name of the pattern.
- 2) **Description:** Description of the pattern.
- 3) **Motivation:** Description of the pattern purpose or a problem that is addressed by the pattern.
- 4) **BPMN Elements:** The list of the BPMN elements used in the pattern.
- 5) **Process Place:** Where in the process model the pattern can be applied: Start – at the beginning of the process, Intermediate – during the course of the process, or End – at the end of the process.
- 6) **An example:** An illustrative example presenting the pattern.

In the following subsections we present 10 selected rule patterns that can be observed in BPMN process models: Conditional Flow, Conditional Trigger, Conditional Task/Subprocess Interruption, Conditional Process Interruption with Initiation, Rule-based Task, Simple Conditional Choice, Rule-based

Choice, Deferred Conditional Choice, Conditional Task Multiplicity and Task Performer Assignment.

A. Conditional Flow

Description: Conditional Flow provides the ability to control the flow of a token based on the evaluation of the *condition* expression in the process instance.

Motivation: Controlling the flow using conditional expressions serves as an additional building block for process model that allows for detailed controlling of the flow of token through the branch of sequence flow. This pattern provides a condition for a sequence flow and is a variant of Data-based routing pattern³ in the Data-based perspective.

BPMN Elements: Conditional Sequence Flow

Process Place: Intermediate

An example: Conditional flow with the condition: *An amount has to be higher than 100* is presented in Fig. 1.

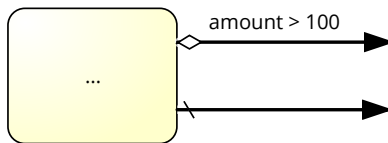


Figure 1. Conditional flow example

B. Conditional Trigger

Description: Conditional Trigger provides the ability to trigger the flow of a token based on the evaluation of the *condition* expression in the process instance.

Motivation: This pattern provides a means of triggering the initiation or resumption of the token flow when a *condition* in the process instance is satisfied. The pattern provides a condition for an event and is a variant of Data-Based Task Trigger pattern⁴ in the Data-based perspective.

BPMN Elements: Conditional Event

Process Place: Start / Intermediate

An example: Conditional Trigger which triggers when *a customer credit rating will be higher than 4* is presented in Fig. 2.

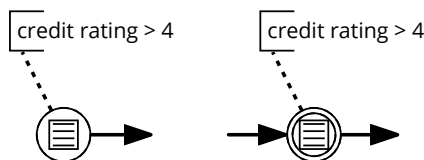


Figure 2. Conditional trigger examples

³See: <http://www.workflowpatterns.com/patterns/data/routing/wdp40.php>

⁴See: <http://www.workflowpatterns.com/patterns/data/routing/wdp39.php>

C. Conditional Task/Subprocess Interruption

Description: When a specific condition is satisfied, a task/subprocess is interrupted and its further execution is abandoned.

Motivation: Conditional Task Interruption pattern provides the ability to abandon the execution of a task/subprocess based on fulfilling a *condition* of the Conditional Trigger. The pattern is related to Cancel Task pattern⁵ in the Control Flow perspective.

BPMN Elements: Interruptive Boundary Conditional Event attached to a Task or a Subprocess

Process Place: Intermediate

An example: A task that will be interrupted when *Credit rating is below minimum* is presented in Fig. 3.

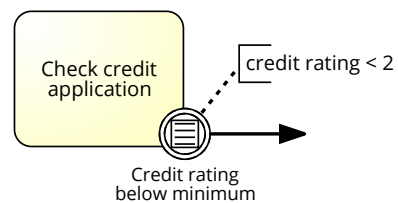


Figure 3. Conditional task interruption example

D. Conditional Process Interruption with Initiation

Description: When a specific condition is satisfied, the current process is interrupted and its further execution is abandoned, and a fragment of the process is started from the Conditional Trigger. A started fragment is not part of the regular control flow.

Motivation: Conditional Process Interruption pattern provides the ability to abandon the execution of a process based on fulfilling a *condition* of the Conditional Trigger and initiates a new subprocess not instantiated by normal control flow. The pattern is related to Cancel Case⁶ and Cancel Region⁷ patterns in the Control Flow perspective.

BPMN Elements: Event Subprocess with Conditional (Interruptive) Start Event

Process Place: Intermediate

An example: A Process will be interrupted when *a customer credit rating is invalid*. Then a procedure of handling the invalid credit rating will be initiated (see: Fig. 4).

E. Rule-based Task

Description: Rule-based Task allows for specification of the task logic using rules and delegating work to a Business Rules Engine in order to receive calculated or inferred data.

⁵See: <http://www.workflowpatterns.com/patterns/control/cancellation/wcp19.php>

⁶See: <http://www.workflowpatterns.com/patterns/control/cancellation/wcp20.php>

⁷See: <http://www.workflowpatterns.com/patterns/control/cancellation/wcp25.php>

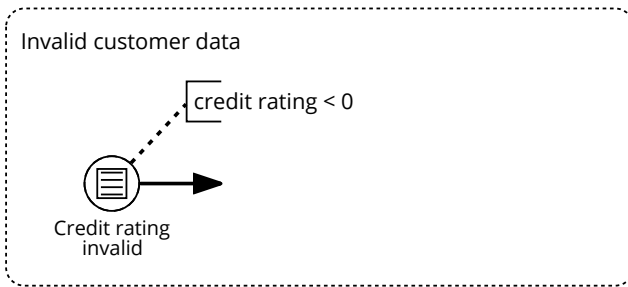


Figure 4. Conditional Process Interruption with Initiation example

Motivation: This pattern provides a means of using Business Rules Engine for execution of rules related to the task.

BPMN Elements: Business Rule Task, Business Rule Task (Call Activity)

Process Place: Start, Intermediate

An example: An example of Rule-based Task which determines a credit card type according to the defined rules is presented in Fig. 5.

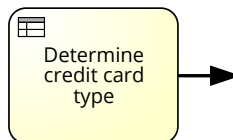


Figure 5. Rule-based Task example

F. Simple Conditional (Exclusive/Multi-Choice/Complex) Choice

Description: Simple Conditional Choice diverges a branch into two or more branches, and according to specified type of choice (Exclusive/Multi-Choice/Complex) the token from the incoming branch is passed to one or more outgoing branches based on the evaluation of the *condition* expressions of the branches in the process instance with support of a mechanism that can limit the number of the outgoing branches.

Motivation: This pattern provides a means of detailed controlling of the flow of token through the branches of sequence flow. It is a variant of Exclusive Choice⁸ and Multi-Choice⁹ patterns in the Control Flow perspective.

BPMN Elements: Exclusive Gateway, Inclusive Gateway, Complex Gateway

Process Place: Intermediate

An example: In Fig. 6 an example of Conditional Exclusive Choice is presented. In this example the choice is made according to the account limit (*either higher or equal 1000, or lower than 1000*).

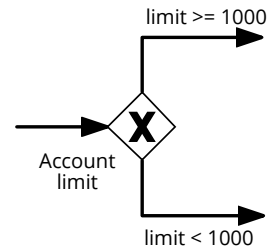


Figure 6. Simple Conditional Exclusive Choice example

G. Rule-based Choice

Description: This pattern is essentially an extension (combination) of the Rule-based Task and the Simple Conditional Choice pattern described above. It provides the Simple Conditional Choice behaviour, but ensures that a decision is made based on the output of the Rule-based Task (the result of inference).

Motivation: This pattern provides a means of detailed controlling of the flow of token through the branches of sequence flow based on the on the output from the Business Rules Engine.

BPMN Elements: Gateway preceded by a Business Rule task

Process Place: Start, Intermediate

An example: In the example in Fig. 7, the choice is made according to the verification result (positive/negative) which is a value obtained from customer verification. The customer verification is performed automatically by a Business Rule Engine using the predefined rules.

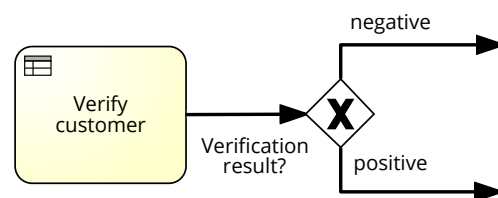


Figure 7. Rule-based Choice example

H. Deferred Conditional Choice

Description: Deferred Conditional Choice determines a point in a process where one or more branches are chosen according to the Conditional Triggers (see Sect. IV-B). The difference from the Simple Conditional Choice pattern is that in this pattern the decision is deferred and depends on the Conditional Triggers.

Motivation: This pattern provides a means of deferring the moment of choice in a process to the last possible time and is based on the conditional triggers. The pattern is a variant of the Deferred Choice pattern¹⁰ in the Control

⁸See: <http://www.workflowpatterns.com/patterns/control/basic/wcp4.php>.

⁹See: http://www.workflowpatterns.com/patterns/control/advanced_branching/wcp6.php.

¹⁰See: <http://www.workflowpatterns.com/patterns/control/state/wcp16.php>.

Flow perspective.

BPMN Elements: Event-based Gateway followed by Conditional Events.

Process Place: Start, Intermediate

An example: The choice in Fig. 8 is deferred to a point in which either a customer credit rating will be below minimum or a customer has more than 2 unpaid loan installments.

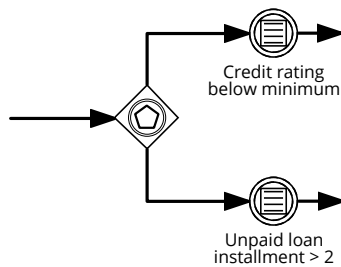


Figure 8. Deferred Conditional Choice example

I. Conditional Task Multiplicity

Description: Within a given process instance, multiple instances of a task can be created according to a condition specified in the rule.

Motivation: This pattern provides a means of rule-based specification of the task multiplicity. The pattern is related to the Multiple Instance Patterns¹¹ in the Control Flow perspective.

BPMN Elements: Multiinstance task with the specified attributes

Process Place: Start, Intermediate, End

An example: Credit card application has to be approved by 2 bank employees (see Fig. 9).

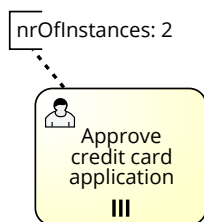


Figure 9. Conditional Task Multiplicity example

J. Task Performer Assignment

Description: This pattern specifies who performs which tasks in a process.

Motivation: This pattern provides a means of a kind of deontic rule that specify the performer role assigned to

particular tasks. The pattern is related to the Role-Based Distribution pattern¹² in the Resource perspective.

BPMN Elements: Lanes

Process Place: Start/Intermediate/End

An example: See Fig. 10: An approval task has to be performed by a supervisor.

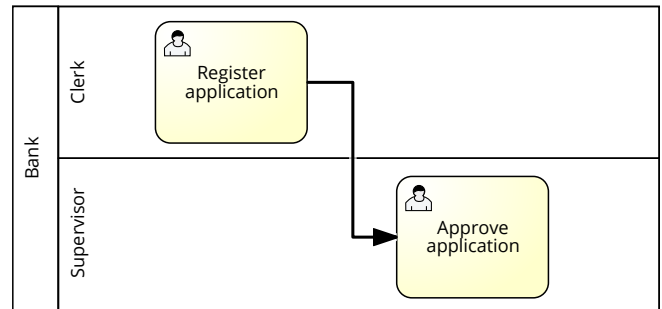


Figure 10. Task Performer Assignment example

V. CONCLUDING REMARKS

In the paper, various types of workflow patterns and domain process patterns were presented. However, these patterns does not take into account rule-based aspects. Thus, we considered various options where and how rules can be perceived in business processes.

The original contribution of this paper is presentation of rule pattern perspective for process models. We described 10 selected rule-based patterns and showed the examples of such patterns in BPMN 2.0 Business Process Models. We focused on the BPMN 2.0 notation, as it is standardized process representation use both by researchers and enterprises.

As future work, we plan to supplement the patterns with other decision aspects in BP models and extend them by investigating other process languages as well as extensions of the BPMN notation [30]. Furthermore, we plan to evaluate these patterns empirically using real-life process models as well as to check the possible anomalies that these patterns can induce [31].

We also want to include design issues using the proposed patterns [32]. as well integrate the pattern library with a process editor [33]. especially for model recommendations based on patterns during the design [34].

ACKNOWLEDGMENT

The paper is supported by the AGH UST Grant.

REFERENCES

[1] M. Dumas, M. La Rosa, J. Mendling, and H. A. Reijers, *Fundamentals of Business Process Management*. Springer Berlin Heidelberg, 2013.
 [2] D. Hay, A. Kolber, and K. A. Healy, "Defining Business Rules – what they really are. final report." Business Rules Group, Tech. Rep., July 2000.
 [3] B. Silver, *BPMN Method and Style*. Cody-Cassidy Press, 2009.

¹²See: <http://www.workflowpatterns.com/patterns/resource/creation/wrp2.php>.

¹¹See: <http://www.workflowpatterns.com/patterns/control/>.

- [4] J. Mendling, H. A. Reijers, and W. M. P. van der Aalst, "Seven process modeling guidelines (7pmsg)," *Information & Software Technology*, vol. 52, no. 2, pp. 127–136, Feb 2010.
- [5] W. Aalst, A. Barros, A. Hofstede, and B. Kiepuszewski, "Advanced workflow patterns," in *Cooperative Information Systems*, ser. Lecture Notes in Computer Science, P. Scheuermann and O. Etzion, Eds. Springer Berlin Heidelberg, 2000, vol. 1901, pp. 18–29.
- [6] OMG, "Business Process Model and Notation (BPMN): Version 2.0 specification," Object Management Group, Tech. Rep. formal/2011-01-03, January 2011.
- [7] R. M. Dijkman and P. V. Gorp, "Bpmn 2.0 execution semantics formalized as graph rewrite rules," in *Proceedings from the Business Process Modeling Notation – Second International Workshop, BPMN 2010, Potsdam, Germany, October 13-14, 2010*, ser. Lecture Notes in Business Information Processing, J. Mendling, M. Weidlich, and M. Weske, Eds., vol. 67. Springer, 2011, pp. 16–30.
- [8] B. Kiepuszewski, A. ter Hofstede, and W. van der Aalst, "Fundamentals of control flow in workflows," *Acta Informatica*, vol. 39, no. 3, pp. 143–209, 2003.
- [9] A. Nolte, E. Bernhard, J. Recker, F. Pittke, and J. Mendling, "Repeated use of process models: The impact of artifact, technological and individual factors," *Decision Support Systems*, 2016.
- [10] W. M. P. van der Aalst and A. H. M. ter Hofstede, "Workflow patterns: On the expressive power of (petri-net-based) workflow languages," in *Proceedings of the Fourth International Workshop on Practical Use of Coloured Petri Nets and the CPN Tools, Aarhus, Denmark, August 28-30, 2002*, K. Jensen, Ed., University of Aarhus. DAIMI PB-560, Aug 2002, pp. 1–20.
- [11] W. van der Aalst, A. ter Hofstede, B. Kiepuszewski, and A. Barros, "Workflow patterns," *Distributed and Parallel Databases*, vol. 14, no. 1, pp. 5–51, 2003.
- [12] W. Aalst and A. ter Hofstede, "Workflow patterns put into context," *Software & Systems Modeling*, vol. 11, no. 3, pp. 319–323, 2012.
- [13] S. A. White, "Process modeling notations and workflow patterns," in *Workflow Handbook 2004*, L. Fischer, Ed. Future Strategies Inc., 2004, pp. 265–294.
- [14] N. Russell, A. ter Hofstede, W. van der Aalst, and N. Mulyar, "Workflow control-flow patterns: a revised view," BPM Center Report, Tech. Rep. BPM-06-22, 2006, bpmcenter.org.
- [15] M. Zapletal, W. M. van der Aalst, N. Russell, P. Liegl, and H. Werthner, "An analysis of windows workflow's control-flow expressiveness," in *Seventh IEEE European Conference on Web Services, 2009. ECOWS'09*. IEEE, 2009, pp. 200–209.
- [16] N. Russell, W. M. Aalst, A. H. Hofstede, and D. Edmond, "Workflow resource patterns: Identification, representation and tool support," in *Advanced Information Systems Engineering*, ser. Lecture Notes in Computer Science, O. Pastor and J. Falcao e Cunha, Eds. Springer Berlin Heidelberg, 2005, vol. 3520, pp. 216–232.
- [17] N. Russell, A. H. Hofstede, D. Edmond, and W. M. der Aalst, "Workflow data patterns: Identification, representation and tool support," in *Conceptual Modeling – ER 2005*, ser. Lecture Notes in Computer Science, L. Delcambre, C. Kop, H. Mayr, J. Mylopoulos, and O. Pastor, Eds. Springer Berlin Heidelberg, 2005, vol. 3716, pp. 353–368.
- [18] A. Lanz, B. Weber, and M. Reichert, "Time patterns for process-aware information systems," *Requirements Engineering*, vol. 19, no. 2, pp. 113–141, 2014.
- [19] M. Skouradaki, V. Ferme, C. Pautasso, F. Leymann, and A. van Hoom, "Micro-benchmarking bpmn 2.0 workflow management systems with workflow patterns," in *International Conference on Advanced Information Systems Engineering*. Springer, 2016, pp. 67–82.
- [20] K. Kaiser and M. Marcos, "Leveraging workflow control patterns in the domain of clinical practice guidelines," *BMC medical informatics and decision making*, vol. 16, no. 1, p. 1, 2016.
- [21] A. Delgado, D. Calegari, P. Milanese, R. Falcon, and E. García, "A systematic approach for evaluating bpm systems: case studies on open source and proprietary tools," in *IFIP International Conference on Open Source Systems*. Springer, 2015, pp. 81–90.
- [22] J. Claes, I. Vanderfeesten, F. Gailly, P. Grefen, and G. Poels, "The structured process modeling theory (spmt) a cognitive view on why and how modelers benefit from structuring the process of process modeling," *Information Systems Frontiers*, vol. 17, no. 6, pp. 1401–1425, 2015.
- [23] J. Claes, I. Vanderfeesten, J. Pinggera, H. A. Reijers, B. Weber, and G. Poels, "A visual analysis of the process of process modeling," *Information Systems and e-Business Management*, vol. 13, no. 1, pp. 147–190, 2015.
- [24] B. Weber, J. Pinggera, M. Neurauter, S. Zugal, M. Martini, M. Furtner, P. Sachse, and D. Schnitzer, "Fixation patterns during process model creation: Initial steps toward neuro-adaptive process modeling environments," in *2016 49th Hawaii International Conference on System Sciences (HICSS)*, Jan 2016, pp. 600–609.
- [25] M. L. Rosa, A. H. M. ter Hofstede, P. Wohed, H. A. Reijers, and W. M. P. Van der Aalst, "Managing process model complexity via abstract syntax modifications," *Industrial Informatics, IEEE Transactions on*, vol. 7, no. 4, pp. 614–629, 2011.
- [26] M. L. Rosa, A. H. M. ter Hofstede, P. Wohed, H. A. Reijers, J. Mendling, and W. M. P. van der Aalst, "Managing process model complexity via concrete syntax modifications," *IEEE Transactions on Industrial Informatics*, vol. 7, no. 2, pp. 255–265, 2011.
- [27] A. Koschmider and H. A. Reijers, "Improving the process of process modelling by the use of domain process patterns," *Enterprise Information Systems*, 2013.
- [28] F. Caron, J. Vanthienen, and B. Baesens, "Comprehensive rule-based compliance checking and risk management with process mining," *Decision Support Systems*, vol. 54, no. 3, pp. 1357–1369, 2013.
- [29] F. Caron, J. Vanthienen, and B. Baesens, "Business rule patterns and their application to process analytics," in *17th IEEE International Enterprise Distributed Object Computing Conference Workshops (EDOCW 2013)*, Sept 2013, pp. 13–20.
- [30] A. Ligeza and T. Potempa, "Artificial intelligence for knowledge management with bpmn and rules," in *IFIP International Workshop on Artificial Intelligence for Knowledge Management*. Springer, 2012, pp. 19–37.
- [31] A. Mroczek and A. Ligeza, "A note on bpmn analysis. towards a taxonomy of selected potential anomalies," in *Computer Science and Information Systems (FedCSIS), 2014 Federated Conference on*. IEEE, 2014, pp. 1097–1102.
- [32] G. J. Nalepa and A. Ligeza, *Software engineering: evolution and emerging technologies*, ser. Frontiers in Artificial Intelligence and Applications. Amsterdam: IOS Press, 2005, vol. 130, ch. Conceptual modelling and automated implementation of rule-based systems, pp. 330–340.
- [33] K. Kluza, K. Kaczor, and G. 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*, ser. Lecture Notes in Artificial Intelligence, L. Rutkowski and [et al.], Eds., vol. 7268. Springer, 2012, pp. 573–581. [Online]. Available: <http://www.springerlink.com/content/u654r0m56882np77/>
- [34] S. Bobek, G. J. Nalepa, and O. Grodzki, "Integration of activity modeller with bayesian network based recommender for business processes," *Knowledge Engineering and Software Engineering (KESE10)*, p. 42, 2014.