

Discovering, Analyzing and Enhancing BPMN Models Using ProM^{*}

Anna A. Kalenkova¹, Massimiliano de Leoni², and Wil M.P. van der Aalst^{2,1}

¹ National Research University Higher School of Economics, Moscow, 101000, Russia
akalenkova@hse.ru

² Eindhoven University of Technology, Eindhoven, The Netherlands
{w.m.p.v.d.aalst,m.d.leoni}@tue.nl

Abstract. Process mining techniques relate observed behavior to modeled behavior, e.g., the automatic discovery of a process model based on an event log. Process mining is not limited to process discovery and also includes conformance checking and model enhancement. Conformance checking techniques are used to diagnose the deviations of the observed behavior as recorded in the event log from some process model. Model enhancement allows to extend process models using additional perspectives, conformance and performance information. In recent years, BPMN (Business Process Model and Notation) 2.0 has become a de facto standard for modeling business processes in industry. This paper presents the BPMN support current in ProM. ProM is the most known and used open-source process mining framework. ProM's functionalities of discovering, analyzing and enhancing BPMN models are discussed. Support of the BPMN 2.0 standard will help ProM users to bridge the gap between formal models (such as Petri nets, causal nets and others) and process models used by practitioners.

1 Overview

Process Aware Information Systems (PAIS) are increasingly used by organizations to support their businesses. All these systems record the execution of process instances in so-called event logs. These logs thus capture information about activities performed. Each event records the execution of an activity instance by a given resource at a certain point in time along with the output produced. Analyzing event logs, understanding and improving processes based on facts are the primary objectives of *process mining* [9]. In this relatively short timespan, this discipline has proven to be capable of extracting from event logs in-depth insights into process-related problems that contemporary enterprises face. Through the application of process mining, organizations can discover models of the processes as they were conducted in reality, check whether the actual executions comply with a prescribed process model, which encode certain practices and regulations. Furthermore, process mining allows analysts to gain insight into bottlenecks, resource utilization, and other performance-related aspects of processes.

* This work is supported by the Basic Research Program of the National Research University Higher School of Economics. When conducting this work, Dr. de Leoni was also affiliated with University of Padua, Italy and financially supported by the Eurostars-Eureka project PROMPT (E!6696)

Most of the process-mining techniques usually need a process model as an input or it is produced as an output. The academic world has proposed many process-model notations in the last years, such as Petri nets, causal nets [4] and process trees [5]. Nonetheless, the evidence is showing that, during the last years, BPMN (Business Process Model and Notation) 2.0 [2] is becoming the de-facto standard for modeling business processes in the industry. Therefore, it seems crucial that every process-mining technique is able to deal with BPMN models. Unfortunately, these techniques have been devised using different notations, which are often equivalent. To overcome this problem, two solutions are possible: either these techniques are adapted to use a BPMN model as input or produce one as output, or robust conversion mechanisms are provided to convert process models from these alternative notations to the BPMN notation, and vice versa.

This paper starts from the belief that the second solution is the most feasible. In this paper, we discuss the operationalization of various techniques to convert models from certain notations to BPMN. These conversions techniques have been implemented as plugins for ProM [3], a generic open-source framework for implementing process mining tools in a standard environment. Several companies and universities around the globe have contributed to the ProM development, designing and implementing plugins to test their own process mining algorithms. The ProM framework is based on the concept of packages each of which is an aggregation of several plugins that are conceptually related. In the latest version, 6.3, there are already more than 120 packages containing more than 500 plugins available, operationalizing techniques in the entire spectrum of process mining. By implementing our conversion techniques in ProM, many of these plugins can now produce BPMN models. Furthermore, we extended ProM to be able to load and store BPMN models using standard formats (see later). In this way, the discovered models can be loaded into an external BPMN modeling tool or into a workflow engine that supports the execution of BPMN models. Similarly, the BPMN models drawn in external tools can be loaded in ProM and used as an input for the diverse process mining analysis.

Last but not least, we have developed a technique to enhance a BPMN model using performance and conformance information. To analyze a process represented as a BPMN model first the model has to be converted to a corresponding Petri net or other formal model. After that this model is verified against the log, retrieving performance characteristics (activities working times, frequencies and probabilities of occurrence in a trace) and conformance information (deviations between the process model and the log).

We support a large subset of the entire BPMN notation; in the addition of supporting the control-flow constructs (activities, connecting arcs and gateways), we also allow for the different types of data objects, swimlanes, subprocesses and events. The structure of the BPMN-related packages in ProM is depicted in Fig. 1. The core BPMN package operates BPMN models and gives an ability to import and export BPMN diagrams in BPMN XML 2.0 [2] and XPDL 2.2 [1] formats. Other BPMN packages depend on this package. The BPMN Conversions package allows to construct BPMN-process models from well-known control flow modeling formalisms such as Petri nets, causal nets [4] and process trees [5].

Thus, using BPMN Conversions package, BPMN processes can be discovered. The BPMN to Petri net conversion was implemented as well [6]¹, this conversion can be used to analyze BPMN diagrams. Moreover, the BPMN Conversions package provides the ability to enhance BPMN diagrams with data and resource perspectives: data Petri nets discovered using the data-aware process mining algorithm [8] can be converted to BPMN diagrams capturing both the control and data perspectives (including data objects and gateway guards), process trees can be converted to BPMN along with the resource nodes, translating them to BPMN lanes. The BPMN Analysis package in its turn enhances BPMN diagram by adding performance and conformance (log and model discrepancies) information. The core BPMN package supports all main BPMN elements and has been continuously extended. The BPMN Conversion package offers fully implemented plugins for conversion between BPMN and formal process models. The functionality of the BPMN Analysis package is still being improved and tested. All the plugins tailored towards working with BPMN in ProM are presented in Table 1.

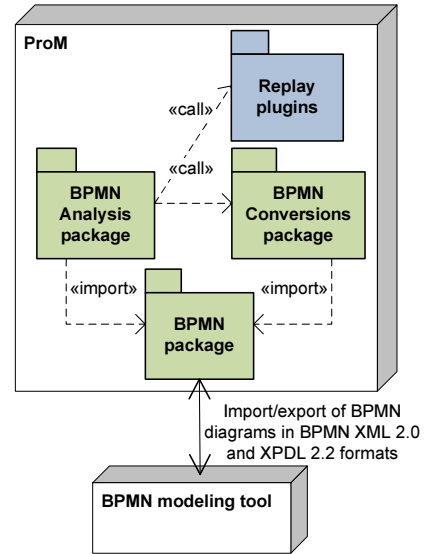


Fig. 1: BPMN packages architecture

Package name	Plugin name	Functionality
Core BPMN package	BPMN Import/Export plugins	Implements import and export of BPMN diagrams in BPMN XML 2.0 and XPDL 2.2 formats
BPMN Conversions	Convert Petri net	Converts a given Petri net to a BPMN model
BPMN Conversions	Convert Data Petri net	Converts a data Petri net to a BPMN model with data perspective
BPMN Conversions	Convert causal net	Converts a causal net to a BPMN model
BPMN Conversions	Convert process tree	Converts a process tree along with resource nodes to a BPMN model with a resource perspective
BPMN Conversions	Convert BPMN model to Petri net ¹	Converts a BPMN model to a corresponding Petri net
BPMN Analysis	Analyze BPMN model	Enhances a BPMN model using performance and conformance information

Table 1: The list of BPMN plugins in ProM

2 Use cases

In this section we discuss usage scenarios of BPMN-related functionality of ProM. The scheme of usage of BPMN plugins in ProM is presented in Fig. 2. The user can

¹ A special thanks to Dirk Fahland, who has implemented the BPMN to Petri net conversion algorithm in ProM.

discover a BPMN model applying discovery and BPMN conversions plugins, after that this model can be annotated with conformance and performance information.

Let us consider an example of constructing a BPMN process model from an event log. Suppose that we have discovered a data Petri net using data-aware process mining algorithm [8] (Fig. 3 a.). A BPMN process model constructed from the data Petri net is presented in Fig. 3 b. This BPMN model can be exported to an external BPMN modeling tool such as Signavio [7] (Fig. 3 c.). This example illustrates that the process discovered from an event log can be finally represented as a BPMN diagram with data and gateway guards and loaded to an external BPMN tool for further analysis or even execution.

The other possible usage is that the user imports a BPMN model from an external BPMN modeling tool (or discovers a BPMN model using discovery and conversion plugins), applies replay technique to retrieve performance and conformance information and annotate the BPMN diagram using this information. Figure 4 a. shows a BPMN process model created in the Signavio tool, this model is loaded to ProM (Figure 4 b.), and then analysis techniques are applied: performance and conformance information for the entire process model and each activity in particular are added to the diagram (Figure 4 c.).

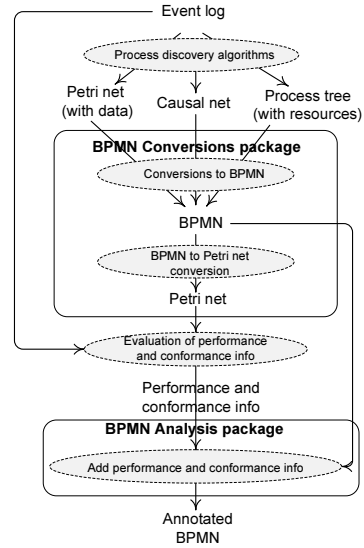


Fig. 2: Functionality of BPMN Conversions and BPMN Analysis packages

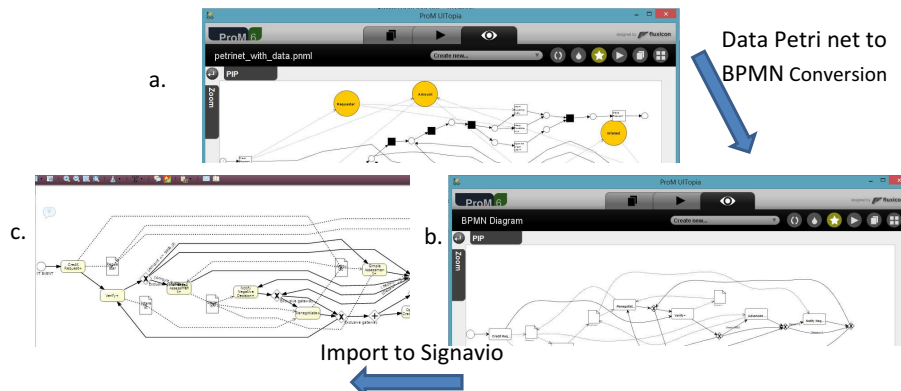


Fig. 3: Discovering a BPMN model with data

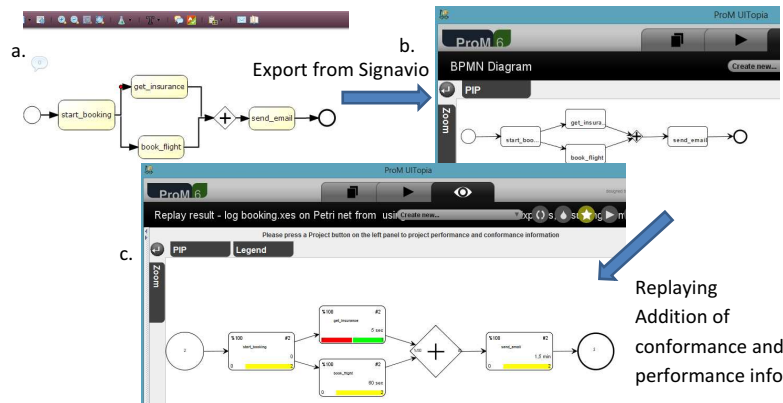


Fig. 4: Analysis of a BPMN model

The entire support for BPMN discussed in this paper is available in the nightly build of ProM and is mature enough to be applied to real business cases. Readers can learn how to install and try out the BPMN support for ProM at <http://pais.hse.ru/en/research/projects/HLM>. At the same link, a screencast video and a presentation are available, showing the application of the two use cases discussed in Section 2.

References

1. Process Definition Interface – XML Process Definition Language (XPDL) 2.2. <http://www.xpdl.org/>.
2. Business Process Model and Notation (BPMN). <http://www.omg.org/spec/BPMN/2.0/>.
3. ProM tool 6.3. <http://www.promtools.org/prom6/>.
4. W.M.P. van der Aalst, A. Adriansyah, and B.F. van Dongen. Causal Nets: A Modeling Language Tailored Towards Process Discovery. In J.P. Katoen and B. Koenig, editors, *22nd International Conference on Concurrency Theory (CONCUR 2011)*, pages 28–42, 2011.
5. W.M.P. van der Aalst, J. Buijs, and B.F. van Dongen. Towards Improving the Representational Bias of Process Mining. In K. Aberer, E. Damiani, and T. Dillon, editors, *IFIP International Symposium on Data-Driven Process Discovery and Analysis (SIMPDA 2011)*, volume 116, pages 39–54, 2012.
6. C. Favre, D. Fahland, and H. Volzer. The relationship between workflow graphs and free-choice workflow nets. *Information Systems*, Elsevier (2014) In Press.
7. M. Kunze and M. Weske. Signavio-Oryx Academic Initiative. In *Proceedings of the Demo Track of BPM 2010*, volume 615 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2010.
8. M. De Leoni and W.M.P. van der Aalst. Data-Aware Process Mining: Discovering Decisions in Processes Using Alignments. In S.Y. Shin and J.C. Maldonado, editors, *ACM Symposium on Applied Computing (SAC 2013)*, pages 1454–1461. ACM Press, 2013.
9. Wil M. P. van der Aalst. *Process Mining - Discovery, Conformance and Enhancement of Business Processes*. Springer, 2011.