

EXPERIENCES WITH BUSINESS PROCESS MODELING NOTATION IN EDUCATIONAL PROCESS

Tomislav Rozman, Institute of Informatics, Faculty of electrical engineering and computer science, University of Maribor, Maribor, Slovenia, tomi.rozman@uni-mb.si

Romana Vajde Horvat, Institute of Informatics, Faculty of electrical engineering and computer science, University of Maribor, Maribor, Slovenia, romana.vajde@uni-mb.si

Ivan Rozman, Institute of Informatics, Faculty of electrical engineering and computer science, University of Maribor, Maribor, Slovenia, i.rozman@uni-mb.si

Abstract

The article presents a new set of standards under development for business process modeling, executing, deploying, querying and maintenance. These standards are being developed by many well-known IT organizations gathered within Business Process Management Initiative. The focus of this paper is the description of Business Process Modeling Notation (BPMN) and our observations of the notation development. The possibilities for extension of notation are discussed. Experiences of its practical usage at the informatics student's lectures are also presented, incorporating a simple example.

1. Introduction

Imagine a person, who is responsible for business process modeling in organization. Due to the availability of a great number of techniques and approaches for business process modeling he (or she) is often confronted with a dilemma of which process modeling technique to use. Those techniques don't share common process representation, (which can be graphical or textual), the level of formality or the level of ambiguity. Some techniques are easier to learn than others, which mainly depends of their primary purpose: is the modeling technique intended for improvement of communication between people which are involved in the process or is it intended for automated execution? Modeling techniques for different businesses are also incompatible with each other. One of the main reasons for poor compatibility is, they often offer only a subset of concepts, which are not enough for detailed process description. The experiences from all areas of standardization are showing, it is almost impossible to create a standard that will satisfy all of the different needs. This fact can be applied also to process modeling technique. It is hard to find an optimal modeling technique for people and machines, because requirements for such technique are very comprehensive and they are frequently conflicting. Only techniques with formal rules are appropriate for automated execution and simulation. The following view should be also considered when constructing process-modeling

technique: people who are documenting and maintaining business processes often don't think like programmers (algorithmic way). Students at the Faculty of Electrical Engineering and Computer Science, University of Maribor are taught how to model a (software) process. In the recent years, many modeling techniques were used for process modeling and many of them were not appropriate for students. The process modeling technique, which is appropriate for students (from teaching assistant point of view), should be easy to learn, it should hide the unnecessary details of process model and broadly accepted. The Business Process Modeling Notation (BPMN) covers almost all of the mentioned requirements, that is the main reason why it was chosen for teaching method. One of the latest results to unify different requirements for process modeling technique is held by Business Process Management Initiative (BPMI).

The rest of the paper is structured as follows. Section 2 provides an overview of the BPMI organization. In section 3, the process modeling concepts are presented. In section 4, the BPMN concepts are explained and differences between specification versions 0.9 and 1.0 are discussed. Section 5 describes our experiences with this notation in educational process and section 6 introduces and discusses some ideas for extension of the notation. Finally, section 7 concludes.

2. BPMI organisation and its work

BPMI is independent [3], non-profit organization of information technology companies, which was founded in the year 2000. Its goal is to develop a set of XML-based standards for process modeling, execution, maintenance and optimization. Their vision is to develop such standards that would enable vendor independent handling with processes on the same way that RDBMS handles data. The following standards are in the development:

- BPML (Business Process Modeling Language). This is markup language for private processes modeling. Its main purpose is for text based, parse-able and executable description of business processes.

- BPMN (Business Process Modeling Notation). This is graphical notation for process modeling and is based on EPC, Petri nets and Activity diagrams (UML). Its main purpose is for human usage. Its specification also defines mapping rules to BPML and BPEL4WS
- BPQL (Business Process Query Language). This language is used for process state querying, deployment and supervision; likewise SQL is used for relational data querying.

Those standards don't replace other similar new standards in the area of business process management, like ebXML [13], RosettaNet¹ [14], Xlang [15], BPEL4WS [7], but they are rather complementary.

Regardless of the chosen modeling technique, it must cover all of the important concepts for process modeling. Let's present those concepts first.

3. Process modeling concepts

The process modeling technique should support modeling of basic process elements [1],[2], which include:

- Concepts: activity², event, state, information object
- Patterns: sequence and parallelism, decision, condition, branching, merging, information hiding, process concepts grouping.

Authors of graphical process modeling notations shouldn't consider only mentioned process elements, but also other aspects, like clarity and readability of the notation. These aspects are not highly important in text-based process descriptions, because their primary intention is for automated execution and their primary requirement is unambiguous definition of the process.

We expect that the graphical notation BPMN covers mentioned concepts very well.

4. Business Process Modeling Notation

BPMN [3] can be treated also as a visual representation of XML-based process description languages, for example Business Process Modeling Language, Business Process Execution Language and others. While it is impossible to represent every detail of the process using graphical symbols, therefore every BPMN element can be

¹ Interesting fact: RosettaNet is named after the Rosetta Stone, a black basalt slab discovered in Egypt in 1799 by a soldier in Napoleon's army

² The process concept is treated as a special kind of complex activity

described using pre-defined attributes. BPMN specification defines graphical symbols for processes [Figure 1], atomic activities, special activities, transactions, events, grouping of elements using pools and lanes and rules for interconnecting these elements.

At the time of writing this article, the BPMN specification is still under the development process. The development of the BPMN specification from its beginning to current version (1.0 draft) was closely followed. Some elements, which were missing (e.g. synchronization) in the early version, were now added. Now the notation is even more appropriate for teaching. It was also noticed, the mapping rules, described in the specification, are shifted towards BPEL language instead of BPML language.

Atomic activity - task 	Looping activity
Ad-hoc process 	Looping ad-hoc process
Process (hidden details) 	Looping process (hidden details)
Process with details 	

Figure 1: Activity types

Sequence of activity execution is represented with full line and message flow with dashed line [Figure 2].

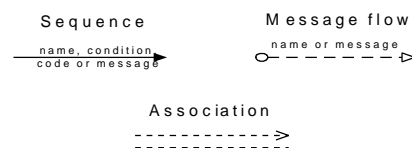
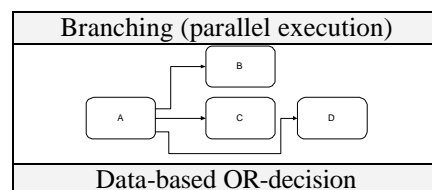


Figure 2: Association types



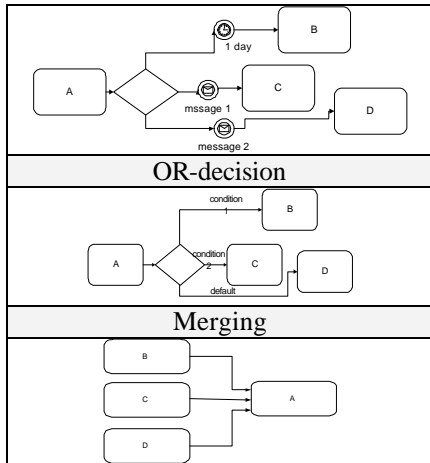


Figure 3 : Branching, decisions and merging

Branching and decisions are modeled as shown in picture [Figure 3]. The difference between mentioned two patterns is, when using branching, all of the successor activities are executed in parallel. Using decision pattern, only one of the successor activities is executed. Decision, which one is executed, can be based on a rule or an event. A special attention should be put on the combination of usage of multiple branching, conditions and merging patterns. In this case, the part of the process can become ambiguous and therefore inappropriate for mapping to execution language.

There are three types of events [Figure 4], which can be modeled using BPMN: start event, stop event and intermediate event. Every type of an event can represent one of the following sub-types of the event: message event, timer event, process error, link event, compensate event, rule-based event or multiple event.

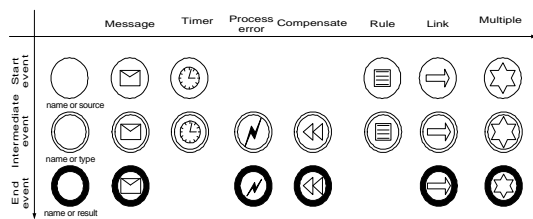


Figure 4 : Event types and sub-types

The notation also offers a symbol, which represents the data object. Primary purpose of modeling data objects is to increase the understanding of the process model and it does not have an influence on the automated process execution.

The pool [Figure 5] symbol usually represents an organization. Sequence flow of the process cannot cross the pool boundaries. The pools can communicate with each other only using message flows. One pool can contain more swim lanes,

which represent divisions or the responsible roles in the organization.

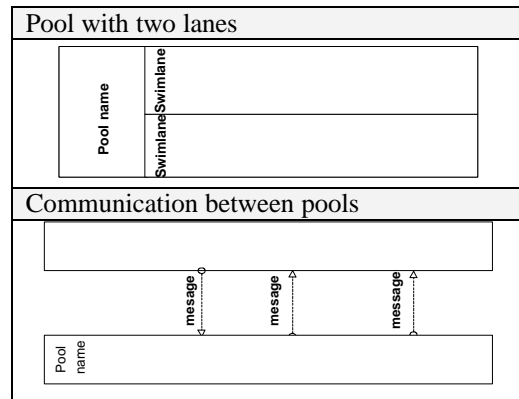


Figure 5 : Pools and swimlanes

Described graphical symbols doesn't represent the whole set of BPMN symbols, because the notation is extensible from the software tool vendors. Presented symbols were available in the early (0.9) version. In the latest version (1.0, August 2003) [5], the notation is extended in the sense of strict specification of branching and merging [Figure 8]. Also new event types [Figure 6], flow types [Figure 8] and transactions were introduced. Transactions are modeled similar as process, but with double line. Special collapsed sub-processes were also added, for modeling of multiple process instances and compensate processes.

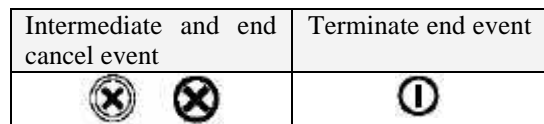


Figure 6 : New cancel events

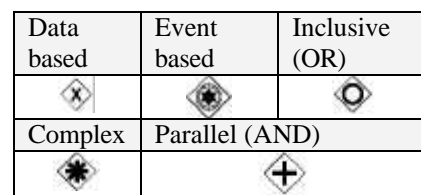


Figure 7 : New branching/merging types

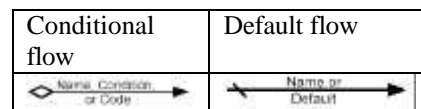


Figure 8 : Additional Flows

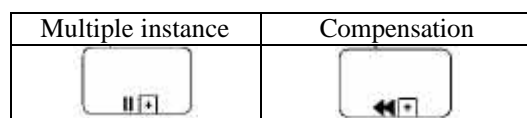


Figure 9 : New sub-process types

With the introduced branching and merging constructs, the process patterns [9], [10] can be now correctly modeled. Although, illegal and non-executable process can still be modeled, so the business process analyst should take care of this.

Until now, there wasn't a standard, commonly accepted and independent way to map graphical process model to executable form. BPMN contains mapping rules, which can be used to map its graphical symbols to XML tags, which are valid for two languages: BPML and BPEL4WS. Mapped process can be executed using a process server. At the time of writing of this article, only a few BPEL4WS compatible process server manufacturers exist, for example Collaxa [11], Intalio [12].

5. Experiences using BPMN in the educational process

The environment

The experiences presented are from teaching assistant point of view. The BPMN notation was used for "Information processing organization and management" practical lectures, as a modeling tool. The aims of these lectures are: to teach Bachelors students (8th semester) how to model (business) processes, to provide the knowledge about software standards and how to organize the development of information systems.

In the previous years, the EPC and Petri net notations were used for teaching process modeling. Although EPC notation is very simple, it was (partly) abandoned because in this notation, some important process concepts can be hardly modeled, for example, different kinds of events and transactions. Petri net notation was also used, but the learning and understanding of it was slower in comparison with other higher-level notations.

Because there is no appropriate educational modeling tool for the BPMN on the market currently, the Visio stencil with appropriate graphical symbols was created to make process modeling more straightforward.

The teaching process

The theory lectures provide the basic theory about SEI CMMI-SW standard to the students. Through the practical lectures, the students are taught the process modeling principles. The practical lectures begin with process modeling basics and concepts (3 hrs). The overview of some most widely used notations is presented, which includes EPC notation, IDEF notation, Petri Nets and UML Activity diagrams with swimlanes (3 hrs). Mapping the basic process modeling concepts into these notations is explained (3 hrs). The introduction to the BPMN notation follows. The similarities with other notations and the origin of BPMN graphical

symbols are stressed (3 hrs). Then, the teaching using practical examples begins. Firstly, some simple process models [Figure 10] are presented, for example, the process of cooking a meal, the process of using ATM, the loan approval process and so on. These examples use only a subset of BPMN symbols (for the clarity's sake): processes, sub-processes, activities and events, which are interconnected using only sequence flows. During that phase, students should model a simple process themselves (3 hrs). Their models are reviewed and discussed.

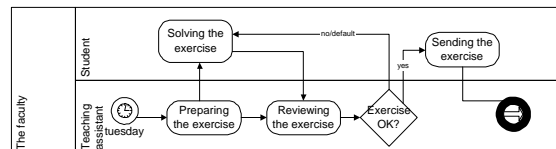


Figure 10 : Introduction to BPMN example

The next phase of the teaching process (3 hrs) consists of model refinement [Figure 11]. Simple process model examples are enhanced as follows: The pools and lanes are created, which are based on the identified process roles (activity owners). The activities are arranged in the appropriate lanes. The message flows between pools and data objects are added.

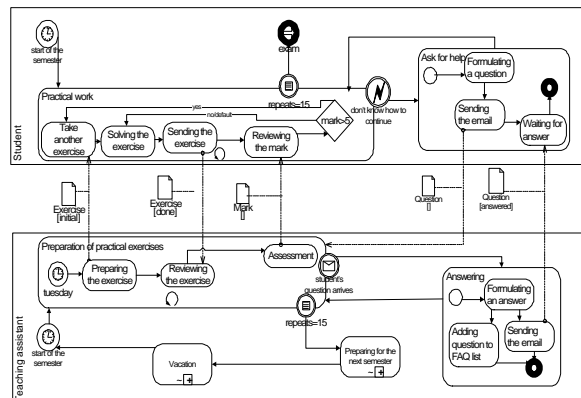


Figure 11: Enhanced BPMN example

Students should also enhance their process models as described. These process models are reviewed again. Their feedback about notation usage is captured using interviews.

In the following practical lectures (9 hrs), the BPMN is taught using examples of software processes. The Configuration Management process for software development is explained in the details, according to CMMI-SW key process area Configuration Management, Maturity Level 2 [16]. We noticed, the students have difficulties understanding software processes, mainly due to the complexity of software processes and the lack of the student's 'real-world' experience with software development. The SEI CMMI-SW

documentation of key process area (the basis for the process model) is used in the following way: Firstly, the key process is thoroughly studied and discussed. Next, the activities are identified (from the Specific and Generic Goals section). The parallelism of identified activities is determined. The activities are linked and grouped into processes and sub-processes. The data objects are extracted from “Examples of work products” section.

The result of this phase is rough process model, which is in accordance with the CMMI standard. Lastly, the pairs of students choose one of the key process areas from the standard, for example: Requirements management, Project Planning, Measurement and Analysis. The aim of their seminar work is to independently model the chosen software process using the standard’s [16] documentation in 18 hours. The teaching assistant is constantly reviewing their work in progress and suggesting the improvements.

When the students finish their software process models, the implementation phase begins. They implement a simple application (using Lotus Notes), which simulate the process model (the flow of data objects).

The research method

As mentioned, information about student’s understanding of the notation was gathered using interactive interviews incorporating reviews of the results of their work during their practical work. The feedback was gathered several times: at the ‘simple process modeling’ phase, at the ‘enhanced process modeling phase’ and especially during their independent software process modeling (seminar work). Improper usages of BPMN concepts are manually extracted from student’s computer files, recorded, grouped and converted to so-called FAQ³ list.

The results

To sum up our experiences: Practical experiences showed that the learning process of the BPMN was fast. The students could model simple processes within 3-6 hours of teaching the notation. Also, more complex examples of processes from every day’s life didn’t cause them any problems.

The students had more problems when modeling software processes, which is mainly due to the complexity of the domain.

The following list contains some of the most frequent questions from the previously mentioned FAQ list:

- When to use BPMN *sub-process* concept instead of *activity*? How to align conceptual levels of the process model?

- If the BPMN *message flow* is already modeled, the *sequence flow* should also be modeled?
- When to use *loop type sub-process*? Can it be replaced with the implicit loop model using decisions and conditions?
- Is it necessary to model the *default* path from *decision* symbol?
- How to model the merging concept/synchronization?
- Can be *pools* and *lanes* used within the sub-process? And so on.

This feedback was very valuable to us, for improving the quality of the practical lectures of the notation for the next years. The FAQ list will be taken into account when the practical lectures for the next semester will be prepared.

[Figure 12] presents a simple example of the student’s practical seminar work, whose goal was modeling the process of Measurement and Analysis key-process area from CMMI-SW standard.

The reader should be noted, at the time of writing this article, version 1.0 of BPMN notation already exists, but it was not used yet in practical lectures. Because of new decision and merging concepts introduced, we assume, the teaching and the practical usage of the notation will be easier (for teacher) less confusing (for students).

Also there is a need for integrated tool which supports BPMN notation, process model checking (syntax, ambiguity and other model properties) and automated mapping to the execution languages. Current solution (self-made Microsoft Visio add-on) is only temporary solution and is intended only for graphical modeling.

6. Discussion

Usage of the early version of BPMN showed itself as a good choice. It proved that notation is very expressive and easy to learn, which is mainly due to previous experience of the authors and students with process modeling in EPC and Petri nets notation. Although it is reasonable to wait with its ‘real-world’ usage until the final version and supportive tools are released.

When (If) this notation is commonly accepted and becomes also a de-facto standard, it can be predicted, the repositories of processes and process patterns will emerge. But, searching for the appropriate process can become burdensome, similar as difficulties when searching World Wide Web for documents. Not only searching, also choosing the right process for organization can be a difficult task, mainly because the area of process relations and their properties (e.g. complexity) is still in the development. One of the possibilities to improve classification of the process models is to describe them before publishing using some

³ Frequently Asked Questions

resource description language, for example RDF or DC (Dublin core classifiers). These descriptions could help to ones who are responsible for discovering, describing, modeling and deploying the process models. If the process models are described in such a way, we could search, for example, all the similar processes to our process, or, previous versions of the process. Also the students could gain better knowledge about process models, if the process models could easily be found on the Internet. Learning through the practical examples always proved itself as the best strategy of learning new things.

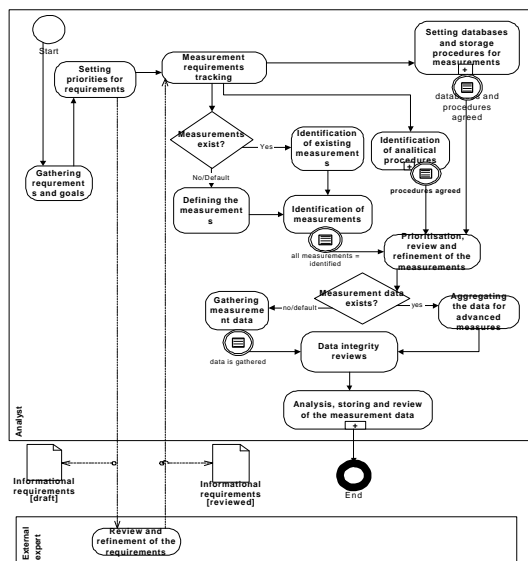


Figure 12 : Example of the process model (student's work-in progress)

7. Conclusion

One question about the BPMN notation still exists: why would this notation succeed to gain broader public attention as other process modeling techniques? The answer could be, BPMN technique is based on well-known and proven concepts, it is user-friendly and allows straightforward mapping to process execution languages. A single corporation does not own the BPMN specification, but it is developed by many organizations (IBM, BEA, IDS Scheer, Fujitsu). Therefore, for wide public adaptation it needs only the critical mass of users. We think, the teaching the students of this new notation is a right way to help the notation to become well known and widely used in the following years.

8. Literature

- [1] ROZMAN, Tomislav. Razvoj in funkcija orodja za ocenjevanje kompleksnosti programskih procesov:diploma work, Maribor, 2001.
- [2] VAJDE HORVAT, Romana, ROZMAN, Ivan, ROZMAN, Tomislav. *How to evaluate the*

complexity of software processes?, MIPRO2000 proceedings, 2000

- [3] BPML.org FAQ, Retrieved January 13, 2003, from <http://www.bpml.org/faq.esp>

- [4] BPML.org, *Business Process Modeling Notation, Working Draft (0.9)*, November 13, 2002, Retrieved December 20, 2002, from <http://www.bpml.org/bpml-spec.esp>

- [5] BPML.org, *Business Process Modeling Notation, (1.0)*, Retrieved August 25, 2003, from <http://www.bpml.org/bpml-spec.esp>

- [6] Assaf Arkin, Intalio, *Business Process Modeling Language*, Retrieved July 22, 2002, from <http://www.bpml.org/bpml-spec.esp>

- [7] BEA Systems, International Business Machines Corporation, Microsoft Corporation, Inc., *Business Process Execution Language for Web Services, Version 1.0*, Retrieved January 13, 2003, from <http://www.ibm.com/developerworks/library/ws-bpel/>

- [8] DCMI Usage Board, *Overview of Documentation for DCMI Metadata Terms*, Retrieved July 12, 2003, from <http://dublincore.org/documents/dcmes-qualifiers/>

- [9] Wil M.P. van der Aalst, Marlon Dumas, Arthur H.M. ter Hofstede, Petia Wohed, *Pattern Based Analysis of BPML (and WSCI)*, Department of Technology Management, Eindhoven University of Technology, The Netherlands, Retrieved January 13, 2003, from http://tmitwww.tm.tue.nl/research/patterns/download/qut_bpml_rep.pdf

- [10] W.M.P. Van der Aalst, *Advanced Workflow patterns*, Eindhoven University of Technology, Retrieved September 23, 2003, from <http://tmitwww.tm.tue.nl/research/patterns/download/coopis.pdf>

- [11] *Collaxa 2 Key Features*, Retrieved October 10, 2003, from <http://www.collaxa.com/product/welcome.html>

- [12] *Intalio/n³ Product tour*, Retrieved October 10, 2003, from <http://www.intalio.com/products/tour/index.html>

- [13] *ebXML FAQ*, Retrieved November 20, 2003, from <http://www.ebxml.org/faq.htm>

- [14] *General FAQ*, Retrieved November 20, 2003, from <http://www.rosettanet.org/>

- [15] *XLANG*, Retrieved November 20, 2003, from <http://www.ebpml.org/xlang.htm>

- [16] CMMI Product team, *Capability Maturity Model® Integration (CMMI-SW)*, Version 1.1, Retrieved October 1, 2003, from <http://www.sei.cmu.edu/pub/documents/02.reports/pdf/02tr029.pdf>