Caring of Intentionality in Business Process Models Using Business Process Patterns.

Václav Řepa[1]

[1] Department of Information Technologies, University of Economics, Prague,
W.Churchill sqr. 4, 130 67 Prague, Czech Republic

Abstract. The paper introduces the problem of process intentionality as a basic condition for fulfilling the contents of the idea of process-orientation in process models. Consequently it discusses the role of the business process modeling methodology in achieving the substantial effects of the idea of process-oriented management. In this context it introduces the concept of business process patterns as a part of the Methodology for Business Process Analysis and Management (MMABP) and explains their important role in the methodology. Different conceptions and meanings of the popular concept of BP Patterns are discussed before the specific meaning of this concept in the MMABP is explained. The basic BP Pattern is described and explained in detail together with connected principles and features of the MMABP with the help of the Business Process Metamodel. Specific respect is paid to the role of BP Patterns as an expression and explanation of the most important principles and rules for modeling business process details.
In the final section some wider consequences as well as future development of the concept of the Business Process Patterns in the context of the MMABP are discussed.

Keywords: business process management, business process modeling, intentionality, Enterprise Architecture, design and analysis patterns, process based management, MMABP methodology.

1 Introduction

Business process management undoubtedly represents the most significant movement in the management theory in last several decades. Nevertheless, it represents the crucial movement also in the field of information systems development. It is mainly because of the natural complexity of these ideas which typically cover many different dimensions of enterprise management including the technology on the first place. Technology plays the central role there; it is a trigger as well as an enabler of possible important changes in the company behavior in terms of exploiting the new possibilities created by the technology development [7]. To achieve such effects it is necessary to well understand the technology and its possibilities on one hand together with understanding the business area and connected content aspects (like managerial,

social, psychological, political, and other) on the other hand. The complexity of the approach is absolutely crucial there. Neglecting just one of these important aspects reliably leads to the final failure.

Consequently, the primary critical success factor of the implementation of these ideas is an ability to mentally cover all necessary dimensions of a problem. This factor typically causes many misunderstandings like overestimating the technical aspects of the business process together with omitting the non-technical ones which is typical for IT-oriented people, or, in turn, ignoring the technical aspects which is typical for managerial approaches.

2 The problem

Importance and popularity of the Business process management phenomenon historically led to the creation of specialized process modeling languages and standards. During the last two decades many different standards for business process modeling have been created from different purposes and for different reasons. Most of them are not in use already. Two of them can be regarded as dominant: Business Process Modeling Notation (BPMN) [11] and Architecture of Integrated Systems Notation (ARIS) [17]. Business Process Modeling Notation (BPMN) as a language for modeling business processes [1] is a most important standard fulfilling the above stated requirements for the standardization. Among other popular standards ([18], [9], [17]) only the BPMN became widely accepted by users as well as by CASE tools producers, and is developed concerning other related significant technology standards [19], [20], [16] which is the basic condition for fulfilling the full meaning of standardization. This fact qualifies the BPMN for being a leading professional standard in the field of business process modeling and it is also the reason for choosing it as a basic notation for business process models in the MMABP methodology (see the following chapter).

As process modeling languages has been created by the IT community they naturally suffer from the overestimating of technical aspects of business process and omitting the non-technical ones. Moreover, in terms of the approach of the leading modeling standards UML [16] and BPMN [11] these languages are not aimed on expressing the methodical principles. They rather support the modeler with the general frame for modeling following basic, general restrictions which is methodologically independent in principle. The main task for a methodology is than completing the language with other, still uncovered, aspects which are often connected to special aspects of the subject and purpose of modeling. Thus these *standards for modeling business processes, as they are, are naturally insufficient in terms of the principles of business process management*. They have to be always completed by the methodology the main task of which is to bring additional rules, constraints and tools for eliminating the mentioned insufficiencies.

One of the most important problems often occurring in business process models which do not sufficiently respect the non-technical aspects of business processes is a

lack of intentionality (purposefulness). As we regard this problem as crucial in terms of process management ideas expressed in [7] the following text focuses on it.

In the legendary article [21] Norbert Wiener expressed the idea which fatally influenced the later development of cybernetics: "*all purposeful behavior may be considered to require negative feed-back*". The concept of *negative feed-back* is explained there as follows: "*...the behavior of an object is controlled by the margin of error at which the object stands at a given time with reference to a relatively specific goal. The feed-back is then negative, that is, the signals from the goal are used to restrict outputs which would otherwise go beyond the goal.*".

According to the basic work in the field of process-driven management ([7]) business process always follows some goal. The goal is a fundamental attribute of a business process as it is regularly used in matured methodologies like in [9] for instance. That means that **business process is always an intentional process**. By the term *intentional process* we mean the process of **purposeful behavior of interested object** following some goal.

Concluding from previous two paragraphs we can find that the **business process**, as it is an intentional kind of a process, **have to have some negative feed-back** which ensures restriction of its outputs in order to keep them in the margins of its goal. This characteristics strongly distinguishes the business process from the process in general (ie. in just technical /physical sense) as well as from processes which do not need any feed-back like machine-managed or automated processes running without a contact with their environment.

In case of business process feed-back means some **input to the process from its environment which is causally connected with some process output**. The value of the input should influence the following behavior of the process in terms of keeping it in the margins of its goal. This means that "intermediate" inputs to the process (i.e. none-starting inputs to the process coming between its starting and end points) are critically important parts of the business process distinguishing it from other, non-intentional (i.e. non-business), processes. Working with processes we have to take into the account even the time dimension; every input to the process from its environment has to be synchronized with the process run. Thus in each part of the process where some input which will influence the following process run is expected the **process state** has to be placed. Process state means such point in the process structure where nothing can be done before the input to the process occurs, i.e. point of **waiting for the input**. The concept of process state is present just in some process modeling standards (like IDEF, see [18] or Petri Nets based languages[1]), partially present in some others (like ARIS[2], see [17]), many standards do not support it. Widely accepted process modeling standard BPMN ([11]) does not recognize this concept at all.

---

[1] Although Petri Nets, where the phenomena of states and synchronization are essential, are not originally intended for modeling business processes there are some interesting attempts to use them for this purpose in the community, like in [3].

[2] ARIS methodology mixes the concept of *process state* with the concept of *event* which is confusing and may contradict with the idea of negative feed-back.

Regarding the importance of the above outlined problem together with the insufficient support in most of process modeling standards it can be said that the primary task for every process modeling methodology is to **allow the modeling of process states ensuring the critically important presence of the negative feed-back** no matter which notation and/or modeling standard is used.

3   The concept of Process State in the MMABP Methodology

The ideas described in this paper are the part of the MMABP methodology (Methodology for Business Processes Analysis and Design). MMABP distinguishes between two main types of models: business process versus business object models. In both types of model the methodology recognizes the global model (system view) and the detailed model. The modeling tools used by the methodology are based on common standards BPMN [10], UML [15], and Eriksson/Penker Notation [8]. The essence of the methodology is defined in the formal meta-model [12] as a part of the development project OpenSoul [11]. The metamodel consists of three main packages which correspond to the main dimensions of the business system model mentioned above: Business Substance Metamodel, Business Process Metamodel, and Business Models Consistency Model. The patterns which are the main subject of this paper are based on the principles defined in the Business Process Metamodel and express them (see Fig. 1).
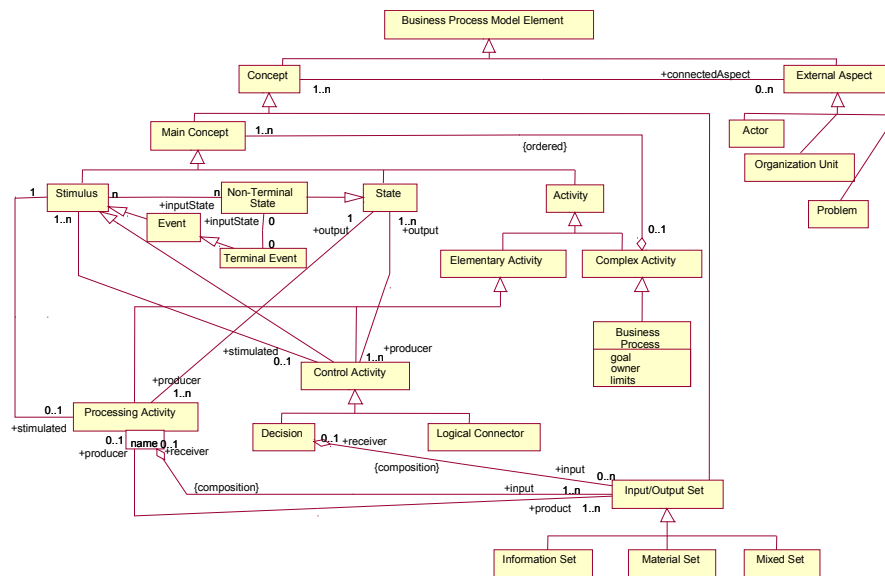

Fig. 1. Business Process Metamodel [12]

As the metamodel shows the concept of **state** is one of the crucial concepts in the methodology. Besides the fact that it is one of three basic process elements (together with *stimulus* and *activity*) it plays the specific role of "connector" of two main kinds of process activities: *processing activity* and *control activity*. This fact is closely connected to the concept of negative feed-back according to the definition in [21]: processed outputs are connected to inputs which influence the following process behavior. Further explanation of the metamodel can be also found in [13].

The information contained in the metamodel is complete but hardly understandable as many facts are expressed there as a consequence of other facts. Therefore, in order to express the methodological principles in better understandable form, the metamodel is completed in the MMABP with the **business process patterns**. Business Process Patterns in the MMABP methodology express the general solutions for typical situations occurring in the process of modeling the business processes. Business Process Patterns complete the methodology with the further information about how to create models which are fully consistent with its principles and rules. They can be also used as the general examples of typical segments which the business process should consist of. These segments can be instantiated for the particular situations and used as basic building blocks of the business process model.

In terms of the main idea of this paper – ensuring the intentionality of the business process - the key pattern from the MMABP called *Basic Process Flow Pattern* is especially important. Therefore the following text focuses on this pattern in detail.

4   Business Process Patterns

The concept of pattern is very popular in the field of IS development methodologies and connected areas for a couple of decades already. This term has been introduced in this field inspired by the concept of "architectural patterns" introduced by the architect Christopher Alexander in [1]. Alexander defined the basic general aspects of so-called "pattern language" like syntax, grammar, and dictionary and put the basics for its systemic use in all fields where the concept of design is naturally relevant.

In the field of object-oriented analysis and design this concept has been introduced already in 1992 by Peter Coad [4] in the form of several simple structures of analysis objects generally usable (under given conditions and with respect to the given meaning of objects) in any OO system. The most significant and complete elaboration of this phenomenon can be found in the book from Martin Fowler [5]. Fowler describes there a dozens of analysis and design patterns divided into several categories according to different domains of their use. A comprehensive detailed view on the use of the concept of patterns in the field of IS development can be found in [23], and its specialization in the field of OO analysis and design together with a precise literature overview in [3].

In the field of business processes the concept of patterns is most frequently used in the context of workflow management. The most visible initiative in this area is the Workflow Patterns project [9], a joint effort of Eindhoven University of Technology

and Queensland University of Technology which is aimed on "providing a conceptual basis for process technology". In the project the various perspectives of needed support by a workflow or a business process modeling language are examined. In accordance with the main focus of the workflow management mainly the technical characteristics of the process are emphasized, i.e. those characteristics which follow from the general theory of algorithms. Nevertheless the Workflow Patterns are typically oriented on the dynamic (process) aspects of the business process which is their main difference from the analysis patterns in the field of OOA.

Hans-Erik Eriksson and Magnus Penker in [8] transferred the concept of pattern to the field of the business system contents. They recognize several categories of so-called. "Business Patterns" which correspond to the essential concepts of their specific methodology aimed on the use of the UML [15] for modeling the business systems. One of these essential concepts is also the concept of business process. The authors introduce different patterns in three categories describing basic structural aspects of business processes and their mutual relationships. The novelty of their approach lies in the strong orientation on the business contents which gives their patterns the methodology dimension. Nevertheless in accordance with the focus of their object-oriented methodology only the structural characteristics of the process are described. Erikson/Penker patterns thus cannot be fully regarded as process (dynamics) oriented.

There are several definitions of the concept of pattern in the field of analysis, for instance: "an idea that has been useful in one practical context and will be probably useful in others" [5], "abstraction from a concrete form which keeps recurring in specific non-arbitrary contexts" [7], "formalized triple of problem, solution, and context which solve domain-specific problem by encapsulating specialist knowledge" [2] (translated by [3]), "self-contained model template with well-defined connectors to application environments capturing knowledge about best practices for a clearly defined task" [22] (*task pattern*). All these definitions nevertheless contain the same popular common ideas of "reusability" and "knowledge sharing". At the same time these ideas characterize in fact the essence of the concept of "methodology" as well.

Process Patterns in the MMABP

There are two main kinds of Business Process Patterns in the MMABP methodology:
- **Basic Business Process Flow Pattern** which defines the basic procedure and decision points of the process of model creation. This pattern is absolutely essential in the MMABP, it expresses the main principles, rules, and other aspects of the MMABP approach to the business process modeling and therefore it is explained in detail in this paper.
- **BP patterns for particular situations** which cover typical situations frequently occurring in business process models where it is possible to find

some generally valid structures, principles and constructions which should be fulfilled by the process description undoubtedly.

There are three most general patterns: Complementary Events Pattern, Repeating State Pattern, and <<inclusive OR>> Pattern. As the complementary patterns for particular situations are not so closely connected with the problem of process intentionality as the basic pattern we do not pay more attendance to them in this paper.

Basic Process Flow Pattern

The Basic Process Flow Pattern expresses the basic structure of the process model which respects the essential rules of the MMABP methodology. These rules express the "technical" necessities which mainly follow from the general theory of algorithms as well as the specific aspects of the business process which distinguish the business process from a process in general (i.e. just technical) sense. The lately mentioned rules follow from the theory of BP management and reengineering which is anchored already in the basic work in this field: [6].
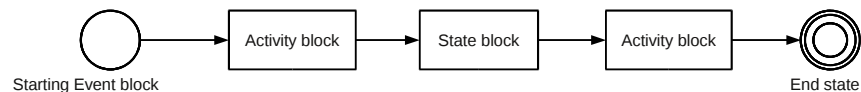

Fig. 2. Basic Business Process Flow Pattern (BPF Pattern) [12]

Basic Process Flow Pattern (see Fig. 2) expresses the essence of the process flow using three methodically essential types of the process elements: events, activities, and states.

According to the Basic Process Flow Pattern the business process should be described as a sequence of *Activity blocks* interrupted by *State blocks* starting with just one *Starting Event block* and resulting in one or more *End states*. Fig. 3 illustrates an exact definition of these basic blocks. The definition is written in the semi-formal metalanguage based on the simplification of the standard Extended Backus-Naur Form (for details and explanation of the EBNF see [14]). Used meta-symbols have following meanings:

- A = [element1 | element2 | element3] means that the item *A* can be either *element1* or *element2* or *element3* exclusively.
- A = { elementX } means that the item *A* consists of one or more *elementsX*.
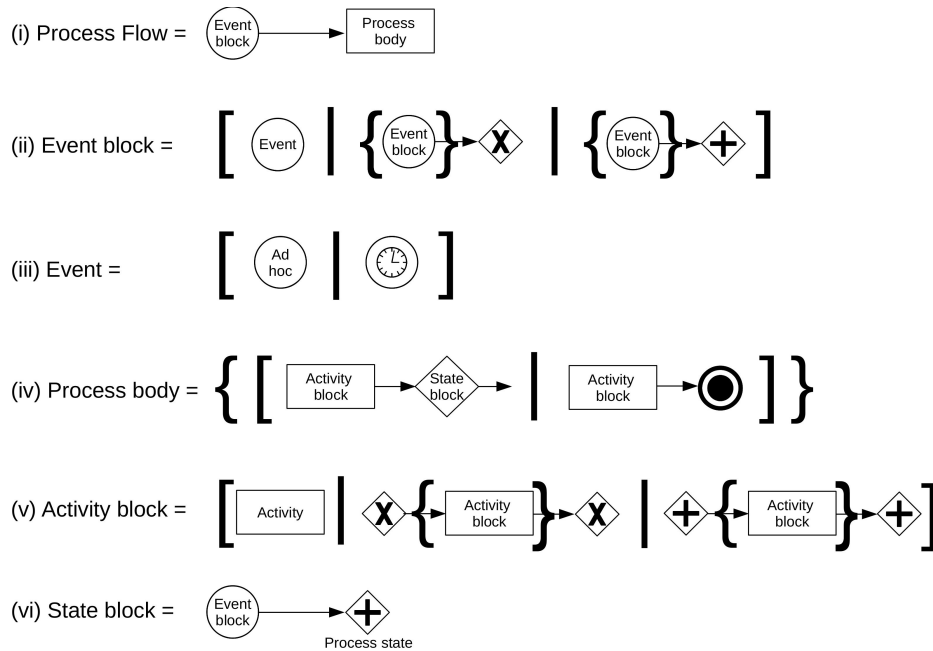
Fig. 3. Definition of basic blocks and concepts of the BPF Pattern

Particular definition sentences can be read as follows::

Def (i):   Process flow begins with starting Event block followed by the Process Body.
Def (ii):   Event block is either a single event, or structure of mutually exclusive Event blocks, or structure of mutually synchronized Event blocks.
Def (iii):   Event can be either an ad-hoc event or a timer.
Def (iv):   Process body consists of one or more pairs where each pair consists of an Activity block followed by either State block or End State. If the pair ends with State block the description should continue with another pair (see the arrow after the State block). End state always means the end of the process.
Def (v):   Activity block is either a single Activity, or structure of mutually exclusive Activity blocks, or structure of parallel Activity blocks.
Def (vi):   State block is a synchronization of internal process flow with expected event(s) expressed as an Event block (in other words: waiting for the event(s)).

Event block represents the external influence which the process always has to respect. It works either as a trigger or a limiter of the process. In both cases it has to be unambiguous which means, among others, that is has to represent a single point of time. Therefore it can be either a single event or a time-elementary structure of

events. If it is a structure it can express either the synchronization of parallel events (event blocks) or the set of possible mutually exclusive alternative events (event blocks) in order to be time-elementary.

Activity block represents an action element of the process. It can be either a single activity or a structure of activities (activity blocks). Similarly as in the case of event also an activity should be unambiguous. Therefore, if it is a structure, it can express either the synchronization of parallel activities (blocks) or the set of possible mutually exclusive activities (blocks). It cannot express a sequence of activities as it would be a violation of the elementariness rule. The methodical reasons and meaning of the need for the elementariness of activities in the process description is discussed in more detail below.

State block represents the essential need to synchronize the process run with expected events. This need follows from the fact that the event is always an objective external influence and thus it must be respected. From the physical point of view such respect means synchronization – waiting for the event (event block). As the BPMN notation do not recognize the concept of process state there is no other way than to express the process state with the general symbol for synchronization – the "AND gate". In order to distinguish between the general synchronization and its specific meaning as a process state we complete the BPMN with the stereotype *<<process state>>*.

One of the most important ideas expressed in this pattern is that *there can not be a sequence of process activities uninterrupted by the process state*. This rule reflects the essence of the **definition of an elementary process activity**:

 (a) the process activity is regarded as an elementary one if there is no objective reason for its interruption,
 (b) the reason for the interruption of the activity is objective if it comes from outside of the process.

Rule (b) of this definition means that each objective reason for the process interruption is represented by an event (external influence) in fact. Thus any activity of the process, no matter how technically complex it is, must be regarded as elementary if there does not exist an external influence (event) which the process has to respect (i.e. wait for). This consequence well illustrates the fact that the elementariness of a business process activity is not only its physical but much more a functional attribute as the business process itself is always more than a physical process (algorithm) only. This way the methodology prevents the analyzer from the pointless unlimited dividing of the process activities which is a frequent mistake in the field of BP modeling. The necessity of such safety fuse in the methodology against the unlimited division of activities is given by the fact that in the field of process-oriented modeling the aggregation is a dominating type of abstraction (unlike

in the field of object-oriented modeling where the generalization is a dominating type of abstraction). This fact manifests itself in the principally unlimited possibilities of division of activities known as a rule: *any single process activity can be decomposed into the structure of sub-activities – a process* (as it is also defined by the process metamodel at Fig. 1). As the division of activities is physically unlimited the methodology has to define some logical – functional definition of the very low level: the level of the process elementariness.
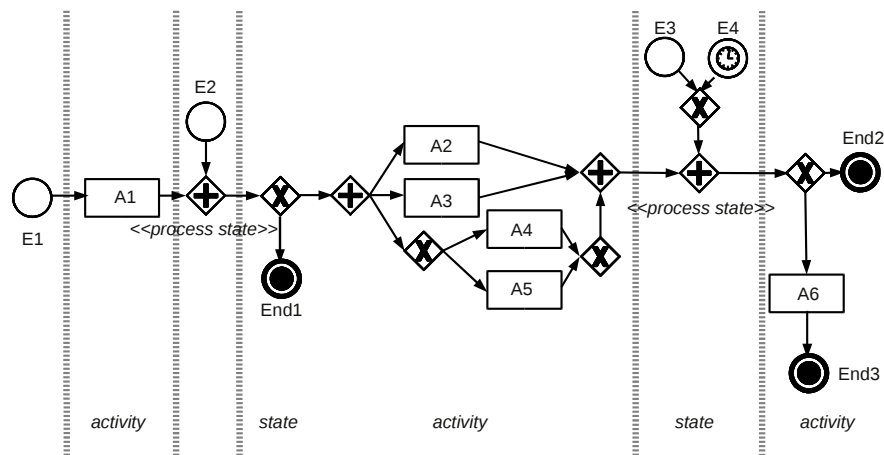
Fig. 4. Correct Business Process Flow Example (source: author)

Fig. 4 shows the symbolic example of the process which can be regarded as correct according to the Basic Business Process Flow Pattern. The process can be seen as a sequence of several parts each representing one block of a particular basic type (see the division of the whole process by vertical dashed lines). It is beginning by the starting event block which consists of just single event E1 in this case – the starting event of the process. The staring event is followed by the activity block consisting of just single processing activity A1. According to the pattern the activity block is followed by the state block in the form of synchronization of the process run with just a single event E2. Following activity block represents more complex structure of activities: it consists of two main alternatives: either the process end End1 or the structure of three parallel activities where the first two are single processing activities A2 and A3, and the last one is a structure of two alternative processing activities A4 or A5. Following state block represents the waiting for two alternative events E3 or E4. The last activity block is a structure of two alternatives: the processing activity A6 followed by the process end End3 or the immediate process end End2.

The example at Fig. 4 illustrates that and how any algorithmic structure of the process can be checked whether or not it fulfills the basic definition of the business process expressed by the Basic BP Flow Pattern: *the business process is a sequence of Activity blocks interrupted by State blocks starting with just one Starting Event block and resulting in one or more End states*.

5 Conclusions

This paper presents the role of the business process modeling methodology in achieving the substantial effects of the idea of process-oriented management. It explains the importance of process intentionality as a basic condition for fulfilling the contents of the idea of process-orientation in process implementation. In this context the paper introduces the concept of business process patterns as an important part of the business process modeling and management methodology. The popular concept of *pattern* is widely used in many different meanings. The common value of this style of presenting in all areas which makes it so popular is the possibility to express important rules a very understandable way. The MMABP business process patterns are specific by their focus on the rules for modeling business processes which include the technical aspects of the process modeling together with the content aspects which follow from the theory of process based management. The essence of these methodology rules is expressed in the Basic Process Flow Pattern which explains how the way of modeling the process should generally respect the main ideas of the process oriented management. The system of patterns has been added to the MMABP based on the experience with its use in dozens of business process modeling projects. Further experience confirmed the importance and suitability of this form of methodical support especially for its integration effect: it naturally integrates different methodical principles in different, mutually connected, views on the business process system. Respect to states between activities in the form of waiting for the external event leads the modeler to the respect to associations with other processes which are giving the process the meaning of the part of the process system. Prohibition of the sequence of activities without a state between them underlines the methodical idea that the prime purpose of the process description is to map the human influences in the process (decision mechanism) instead of the unimportant sequence of activities which is ensured by the technology. The process management means primarily the way of management instead of the technology.

Other mentioned patterns support the methodology ideas in typical specific situations. These partial purpose patterns complete the basic process flow pattern and should be understood strictly under given methodology assumptions. Moreover, their development is principally unlimited; as the evolution of the methodology goes on together with the penetration of specific areas of its application the number of these specific purpose patterns will increase as well.

References

1 Alexander, C., Ishikawa, S., Silverstein, M.: A Pattern Language. Towns, Buildings, Construction, Oxford University Press, 1977, ISBN 978-0-19-501919-3.

2 Bender, K.: Analysemuster in der Architektur kommerzieller Informationssysteme. Josef Eul Verlag, Lohmar 1999.

3 Billington, J., Christensen, S., van Hee, K., Kindler, E, Kummer, O, Petrucci, L., Post, R., Stehno, C., Weber, M.: The Petri Net Markup Language: Concepts, Technology, and Tools. In W. van der Aalst and E. Best, editors, Application and Theory of Petri Nets, 2003, 24th International Conference, LNCS 2679, pages 483–505.

4 Blaimer, N., Bortfeldt, A., Pankratz, G.: Patterns in Object-Oriented Analysis, Diskussionsbeitrag Nr. 451, Fakultät für Wirtschaftswissenschaft, FernUniversität in Hagen, 2010.

5 Coad, P.: Object-Oriented Patterns. Communications of the ACM, Vol. 35, No. 9, ACM-Press, USA, New York 1992.

6 Fowler, M.: Analysis Patterns – Reusable Object Models. Addison-Wesley Publishing, 1997.

7 Hammer, M., Champy, J., 1993. Reengineering the Corporation: A Manifesto for Business Revolution. London: Nicholas Brealey Publishing

8 Riehle, D., Züllighoven, H.: Understanding and Using Patterns in Software Development. Theory and Practice of Object Systems 2, 1, 1996.

9 Eriksson , H.E., Penker, M.: Business Modeling with UML: Business Patterns at Work, John Wiley & Sons, 2000, ISBN: 0471295515.

10 Workflow Patterns Initiative: http://www.workflowpatterns.com/.

11 Business Process Model and Notation (BPMN), 2011. OMG Document Number: formal/2011-01-03, Standard document URL: http://www.omg.org/spec/BPMN/2.0

12 OpenSoul Project: http://opensoul.panrepa.org

13 Business System Modeling Specification: http://opensoul.panrepa.org/metamodel.html

14 Repa V.: "Business System Modeling Specification", Proceedings of the CCCT2003 International Conference, IIIS, Orlando, FL, 2003.

15 Syntactic Metalanguages ISO/IEC International Standard 14977, ISO/IEC, 1996.

16 UML Superstructure Specification, v2.0 document 05-07-04, Object Management Group, 2004.

17 Scheer, A. W: Architecture of Integrated Information Systems: Principles of Enterprise Modeling. Berlin et al.. (1992).

18 Mayer, R.J., Menzel, C.P., Painter, M.K., deWitte, P.S., Blinn, T., Perakath, B.: IDEF3 Process Description Capture Method Report, Knowledge Based Systems, Inc., 1997.

19 Reference Model for Service Oriented Architecture 1.0, OASIS Standard, 12 October 2006 (on: http://docs.oasis-open.org/soa-rm/v1.0/)

20 Service Science Management and Engineering: http://www.ibm.com/developerworks/spaces/ssme

21 Rosenblueth, A., Wiener, N., Bigelow, J.: Behavior, Purpose and Teleology. In: Philosophy of Science, 10(1943), pp. 18–24.

22 Sandkuhl, K.: Capturing Product Development Knowledge with Task Patterns: Evaluation of Economic Effects. In: Quarterly Journal of Control & Cybernetics, Issue 1, 2010. Systems Research Institute, Polish Academy of Sciences.

23 Schümmer, T. and Lukosch, S. Patterns for Computer-Mediated Interaction, Wiley & Sons, ISBN: 978-0-470-02561-1, 2007.