A Dynamic Approach to Process Design: A Pattern for Extending the Flexibility of Process Models

Jiri Kolar, Lubomir Dockal, Tomas Pitner

HAL Id: hal-01474745
https://hal.inria.fr/hal-01474745

Submitted on 23 Feb 2017

A Dynamic Approach to Process Design: A Pattern for Extending the Flexibility of Process Models

Jiri Kolar, Lubomir Dockal, and Tomas Pitner

Masaryk University, Faculty of Informatics,
Botanicka 68a, 602 00 Brno, Czech Republic
kolar@fi.muni.cz http://fi.muni.cz/~xkolar2/

Abstract. This paper presents a specific approach to Business Process design by combining selected principles of Adaptive Case Management, traditional modeling of processes executable in Business Process Management Systems, and a constraint-based approach to process design. This combined approach is intended for business situations, where traditional process models with rigid structures can lead to limitations of business flexibility. We propose a process design pattern that is suitable for the modeling of ad-hoc processes within common BPMS-based systems. The pattern can be used to define a process structure in a declarative constraint-based manner. Further, we present an application of the approach in an actual project, which is an end-to-end BPM project from an insurance business. The project uncovered needs for an extended flexibility of process structures. This along with requirements based on ad-hoc processes led to advancement in the presented approach. This paper presents a versatile, generally applicable solution, which was later tailored for the purpose of the aforementioned project and led to the successful satisfaction of the requirements. The approach is part of a more comprehensive research effort – complex BPM adoption methodology BPM4SME designed primarily for Small and Medium Enterprises, which put emphasis on the agility of the BPM adoption process and consequent flexible implementations of BPMS-based systems.

Key words: BPM, process design pattern, Agile process design, process flexibility, ad-hoc processes, ad-hoc process pattern process discovery

1 Introduction

In the scope of traditional business process models, we usually define a set of work tasks, their performers, and an explicit order in which those tasks should be performed [1]. Nowadays, an increasing amount of process models are defined in modeling languages such as Business Process Modeling Notation (BPMN), based on Petri-nets formalism [2], [3]. BPMN 2.0 was recently recognized as an industry standard for process modeling and is widely accepted in practice [4]. By modeling a process in such a language, we define the allowed sequences

in which process tasks can be performed by defining possible paths through the process graph. Such rigidly defined process models are well applicable to situations where the explicit definition of task order is known in design-time. This has several positive outcomes, such as the establishment of a uniformed work-process, efficient Business Activity Monitoring [5] and other general outcomes of Business Process Management (BPM) [6], [7]. Such BPMN process models can also be consequently executed on a process execution engine - a core component of every Business Process Management System (BPMS) [1].

In business, situations other than the ones outlined above may also arise. For certain type of work, it is optimal to decide about task ordering in run time, according to decisions carried out by task performers. For this kind of work, we will use the well-established term, knowledge work [8],[9].

For knowledge work, the desired order of tasks can differ from case to case. In these situations, process models would have to cover all possible scenarios. This can result in very chaotic process models and model clarity - the important benefit of process modeling [7] is lost. Therefore, there are different needs in the context of supportive Information Systems for knowledge intensive work. Systems built on approaches such as Adaptive Case Management (ACM) provide *palettes* of tasks [9] which can be performed in any order. Such systems do not put hard restrictions on ordering. Instead, they help knowledge workers find similarities among cases and provide soft recommendations based on the orders of tasks previously performed in similar or related cases. In this way, another very important outcome of the BPM-based approach, the codification of business know-how, is preserved and the flexibility of knowledge work is not limited [10], [11], [9].

However, in business practice, it is very common to mix these two kinds of work [12]. Management processes are often performed on the top level [13] where traditional rigid process definitions lead to better monitoring and process unifica-tion. Such management processes consequently instantiate sub-processes dealing with certain business activities and only these sub-processes are often knowledge-intensive. The schism of selection among these approaches during consequent Information System development often leads to a paralysis in decision-making, as it is not clear which paradigm to follow. This was recognized in practice as a strong showstopper of many BPMS-based system implementations.

One could clearly argue that BPMS systems should not be used in the busi-ness context where such ad-hoc processes appear. However, in practice, we often find situations where traditional BPM seems to be a perfect solution for major amounts of work and some minor cases involving ad-hoc processes are identi-fied much later when BPM adoption is already in progress. In such cases, it often does not make sense to combine BPM with different paradigms such as ACM. This is because it would significantly raise costs, increase complexity, and confuse business users, which already have problems understanding the process-based BPM paradigm. This is also the case of the BPM project presented later in this paper.

Before we clearly define this problem and propose a solution, we should clarify the terminology of Dynamic characteristics of Process Models. We will follow the terminology of [12] and define three terms related to business process dynamics:

– **Dynamism** is a characteristic related to the evolution of process model initiated either by changes in business environment or process re-engineering efforts. These changes are made in design-time and they involve a non-trivial problem - the migration of previously executed instances [12]. Although this characteristic is not directly in scope with this paper, the proposed solution should decrease the number of situations where a change in the process model is needed.
– **Adaptability** is the process ability to cope with exceptional circumstances and non-standard behavior. This can be partially solved in design-time by adjusting the process model structure. However, there are still many situations when we have to rely on BPMS-specific technological solutions and run-time work-arounds. We will partially touch on this problem later in the paper.
– **Flexibility** is a characteristic of the process model related to loose or partially defined model structures specified in design-time. The full specification of the process is completed at run-time and it may differ for each process instance. Flexibility is the main focus of this paper, and its improvement is the main objective of our pattern-based approach.

2 Problem Description

Let us sum up three important facts for a clearer definition of the problem.

1. Traditional process models with an explicit ordering of tasks can limit work-process flexibility in certain cases of knowledge-intensive work in which we are not able to determine the exact ordering of tasks before executing a particular process instance [14]. On the other hand, traditional process models significantly help to codify the know-how in their structure [15], [16].
2. To codify know-how of knowledge work, the previous individual decisions of knowledge workers have to be recorded and related to new cases with similar characteristics [9].
3. In practice, major parts of the process models often correspond to traditional rigid structure and only relatively small parts demand a high level of flexibility [12].

Our focus is restricted to situations where we generally want to have a traditional BPMS-based solution. This is due to the fact that nowadays, BPMSes are generally available technologies and we can find relatively mature BPM products [17]. In opposition, ACM-based solutions are usually developed as a custom solution [9]. Therefore, our desire is to achieve the extended flexibility of process models to make them suitable for knowledge work and avoid mixing of ACM and BPM technologies. Obviously, limited flexibility is a common obstacle of

BPM adoptions [18], [19]. This was also confirmed during the practice project in which we participated.

According to the previously described circumstances, we are trying to find a process design pattern, which is applicable either during the initial process design, or even to redesign the existing process. The pattern should meet the following requirements, which were defined during the aforementioned project and later refined according to the related research [12], [13]:

1. The pattern will be applicable to traditional rigid BPMN process models.
2. The pattern will isolate the ad-hoc process parts into sub-process without interfering with the rigid structure of the parent process.
3. The pattern will provide a mechanism to influence such isolated sub-processes from their parent process and provide a mechanism for defining declarative constraints on an ad-hoc sequence.
4. The pattern will record sequences of tasks claimed in run-time for each instance of an ad-hoc sub-process and provide valuable historical data for the discovery of soft structures. This gives certain guidance to the worker, recommending but not directing him on how to proceed in the work process. At the same time, it preserves the know-how codification feature of BPM.
5. The pattern should be usable within conventional modeling methods and implementable in various BPMSes. Therefore, to keep the approach as versatile as possible, it should only use standardized BPMN constructs that are available in most BPMSes.

3 Related Work

Our approach is a specific application of several autonomous principles, which were the subject of several research efforts in the past.

Research related to process design patterns has been well known since the very beginning of BPM era [20], [21]. Most of these efforts focus primarily on describing the best practices for modeling certain logical structures in processes. Other efforts [20] describe patterns for exception handling aimed towards the extension of process adaptability [22]. We can also find later updates of pattern approaches [23], [24] covering more recent advancements in BPM. Such patterns are generally applicable in any process and business context. In opposition, we propose a more specific pattern for handling ad-hocness which is suitable for situations described later in this paper.

We can find several sources related to the characteristics of process flexibility in [12] and [13]. A very relevant topic is research related to the flexibility of processes, particularly a construct called *pockets of flexibility* [12], and the later effort to solve this problem with constraints [25], [26]. The constraint approach is partially used in our work as well. We build our pattern on top of these approaches and use the terminology established mainly in [12]. However, these approaches are rather general and discuss the ad-hoc principles in a general workflow-modeling perspective. Since this research was published, BPM has

made big steps forward and we are therefore able to be more specific and propose solutions, which are directly applicable in the context of currently available BPM technologies. Another interesting approach, which could solve our problem, is based on *Worklets* [27], [28]. However, this is something purely implementation-specific and therefore does not meet our requirement in versatility.

Furthermore, we can also find attempts to achieve flexibility by the adaptation of process definitions, such as [29]. Certain principles used here are also highly relevant in our context. Surveys assessing the current advancements in research on process flexibility such as [18] can be found as well.

Probably the most complete work about declarative approach to process definitions can be found in [13] and consequently in more recent publications [30], [31] related to complementary Declare tooling. Some case studies from practical applications of this approach exist such as [32].

Relevant recommendation-based approach is described in [11]. Research focused on process discovery and mining which is discussed at the end of this paper can be found in [33] and [34].

3.1 Our research context: BPM4SME methodology for agile BPM adoption

The presented approach is a part of a more comprehensive research effort, BPM4SME methodology for small-scale BPM adoptions. As confirmed by related research sources [35], [7], the adoption of the BPM paradigm in the Small and Medium Enterprise (SME) sector has several specific obstacles. We try to overcome these obstacles and develop a methodology, which provides helpful guidelines for successful and flexible BPM end-to-end adoptions in organizations of SME sizing. Our methodology is built on the following key principles. This paper is mainly contribution to the second one.

1. The application of agile collaboration while defining the business motivation model, process architecture, and process design
2. The creation of patterns for designing non-restrictive processes, which do not interfere with the turbulent character of SME business
3. A design for lightweight BPMS-based systems resulting in easy customizable solutions with low initial and consequent maintenance costs
4. A simplified documentation structure where documented processes can be easily transformed into their executable form and vice versa

An agile practice research approach is applied; thusly we verify each component of the methodology in a practical project in an actual business environment at the end of each milestone. Results of those projects are regularly published and discussed with practitioners from different business environments. Thus far, the methodology has been applied in practice in the following projects:

1. A BPM adoption in a web-app oriented SME - the software company IT Logica s.r.o. [36].

2. Process design and optimization in the context of human resources in the ICT department of Masaryk University in Brno [37].
3. Analysis at the headquarters of Masaryk University, several results discussed in the paper related to collaboration in process design [38] and [39].
4. An end-to-end implementation of a BPMS-based solution in a global insurance company, presented at the end of this paper.

4 Proposed Solution

It is now appropriate to clarify certain terminology: *task* vs. *activity*. According to the BPMN standard, *activity* is a process element, which can contain either one task or the nested sub-process consisting of several tasks and other modeling elements. Our pattern is intended to handle not just ad-hoc sequences of tasks but also sequences of activities. Therefore, starting for now, to stay consistent with BPMN, we will use the term of *activity* without differentiating between task and nested sub-process.

We now present our pattern with a step-by-step application in process design. To model and execute ad-hoc sub-process according to our approach, the following steps must be performed:

4.1 The Separation of Ad-hoc Process Parts

The first step is to identify parts of existing processes where an ad-hoc order of activities is desired. We should include all process activities which can be ordered ad-hoc and separate them into one or more sub-process. All activities in one ad-hoc sub-process should be logically related and ideally belonging to one unit of work. Such a separation assures that the rest of the process will be isolated from the on-demand order of activity execution. The separated sub-process will be managed from the parent process. This means that it can be multi-instantiated, repeated several times, or terminated on-demand by events triggered from the parent process.

4.2 The Application of an Ad-hoc Pattern

As depicted in (Fig. 1), the pattern consists of three basic sections. The two on the left serve for assigning tasks. This can be done either by performers, which can freely choose any allowed activity they want, or the assignment can be directed from the *Managing process* with the use of inter-process communication. These sections are synchronized and re-executed in loops. In these sections we also record the sequence of performed activities, evaluate Constraint Business Rules on the recorded sequence and determine which activities are allowed to be assigned in each process state. We will explain the details of the constraint concept in following paragraph. In the *activity section*, assigned activities are being performed and they are either completed or re-assigned. This mechanism
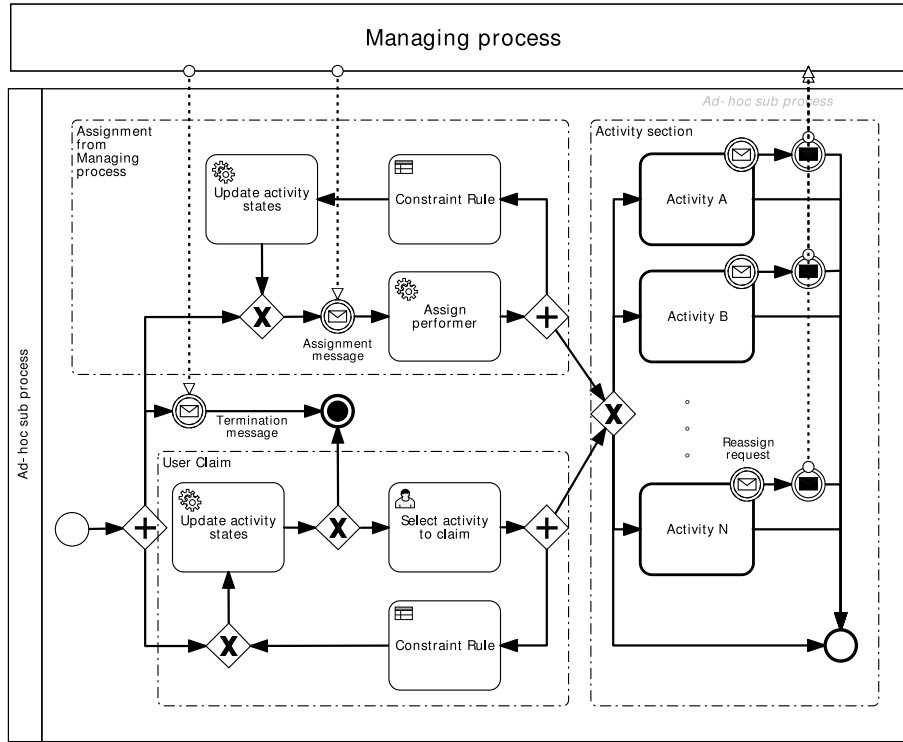
Fig. 1. Process pattern

is used for delegation, which is common in the context of ad-hoc processes. The process can be terminated either by the intervention of the Managing process or it can terminate itself after conditions defined by the business rules are met.

4.3 The Definition of Constraints

The pattern contains business rule tasks, which are responsible for deciding which activities can be assigned to a performer in a certain moment. As an input, this business rule takes a list of assigned activities ordered by time, evaluates constraints that can prohibit the assigning of certain activities in the next assignment step, and produces a list of activities, which can be assigned as an output. Those lists are saved into dedicated objects in process data in the case of data-model-based BPMS, or into an external business object in the case of pure service-based BPMS.

For particular constraint scenarios, we can use the declarative structural patterns defined in a constraint-based approach [13]. In this way we can use the rule language to define complex Linear Time Logic (LTL) based constraints. In practice this concept can be eventually replaced by simple if-then-else rules. Nevertheless, the sophisticated usage of the rule language for defining adequate LTL

[40] provides mechanisms for an implementation of the declarative constraint-based approach [13] within a conventional BPMN 2.0 based BPMS.

4.4 The Identification of Common Patterns in Activity Order

Suppose we have an ad-hoc process modeled according to our pattern with constraint rules. To achieve the last goal and satisfy the requirement for detecting the relationship between activity order and case characteristics, we record each path through ad-hoc process and group similar sequences into a set of common patterns. Consequently we compare these patterns to the active process instance and we are able to provide soft recommendations based on similarities. For that, we have to use two mechanisms not described by the pattern. First, we have to use some mechanism to find similarities. This has to be performed by a service outside of the process. Secondly, we use such a similarity detection mechanism to provide just-in-time intentions to the user in a user interface. For example, we can highlight certain activities, which are usually successors of the last claimed activity. The implementation of this construct can be different for each BPMS and therefore we do not propose any general solution.

4.5 Approach Limitations

Let us discuss the limitations of our approach. Despite the fact that we try to propose a generally applicable solution, we presume that the used BPMS will provide several core functionalities. First, it must implement BPMN 2.0 specifications to the extent in which used modeling elements are covered. Second, for a full-blown constraint driven solution, an adequate Business Rule Management System (BRMS) must be used. Third, there must be a mechanism for in-process reassigning of activity executors. All of these features are generally available in many BPMS products, however, not necessarily in all of them. Last, we expect a certain business setting in which we are able to identify Managing processes in the process architecture; moreover, the *ad-hocness* is mostly present in nested processes, not processes on the top level. Our pattern is also generally applicable for cases where the last condition is not met. However, some modifications of terminology and eventually of the pattern itself may be needed.

5 Application of the Approach in Insurance Business

We are going to present a real BPM project from an insurance business. The problem we present in this paper was introduced in this project. We searched for an optimal solution, which led to the creation of the presented approach. The light-weighted version of the approach was later applied in this project and led to the process redesign, which satisfied given requirements on extended process flexibility. Certain customizations of the approach were made in accordance to specifics of used Bizagi BPMS.

The project was elaborated in a global insurance company. The company recently acquired several smaller insurance companies across Eastern Europe. These acquired companies had similar business processes customized to local regulations and business environments in particular countries. Acquisitions were transformed into Business Units (BU) of the new owner. The transformation typically brought about the need for process unification and the development of a centralized process-driven Information System, which could be used across these newly created BUs. For the implementation of such a system, Bizagi BPMS was chosen as a main BPM platform.

5.1 Project Goals

Three main goals were set in the project:

1. The unification of processes across all business units with small customizations per country respecting differences in country-specific legal restrictions and locally used systems
2. The consequent unification of process monitoring and reporting processes, which could enable the mother company to compare business results achieved across BUs
3. The integration of locally used systems to collect critical business data in one system.

5.2 A Change in Requirements and Process Redesign

According to the initial settings and consequent prototype developed by the BPMS supplier Bizagi, all of the processes were modeled as hard-ordered and the groups of related ad-hoc activities were typically concentrated into one form-based *blob task* executed in loops. Therefore, performers of *blob tasks* had to wait for other performers to complete their all activites before they could claim the *blob task*. From the outer process, there was no control over activities performed inside these *blob tasks*.

The problem was identified after initial testing in one BU, where related ad-hoc parts of the process were performed by various performers. Local project coordinators in the BU complained about inefficient collaboration and they expressed several change requests to solve this issue. Such change requests were analyzed on the side of the BPM team and led to the advancement of the presented approach and consequent application of the pattern to the process redesign. An important influence on the redesign had also the RIVA method [41] used in the initial process design.

About 15 core processes were implemented in the first iteration. However, some of those were nested in each other. A need for *ad-hocness* was identified in three cases. We mention two of the most important top-level processes to understand the context and then we take a closer look at the *Quote evaluation* sub-process, one of three processes, where our pattern was applied.

1. The Opportunity management process
   This is the most important top-level process used for managing potential business opportunities, the state of each opportunity, and communication with the potential customer. The process is started by the first contact with a potential customer and ends ideally with the acceptance of a proposed offer from the customer's side, which leads to consequent signatures of the policy document. This process usually instantiates the important underwriting process described below. The underwriting process is executed in parallel with Opportunity management and the results of underwriting are sent back to the Opportunity Management process.
2. The Underwriting process
   This top-level process is selecting risks for insurance in respect of a plan and classifying members according to their degrees of insatiability so that the appropriate premium may be charged and the terms offered may be reviewed. This is the process responsible for handling particular offers consisting of the assessment of the customer's requests regarding insurance products. Underwriting itself, and carrying out the quoting process and agreements on particular business contracts is the ideal case.
3. The Quote evaluation sub-process
   In this process parameters for each product included in the resulting quotes are considered. There are one or more Underwriters assigned to the preparation of each particular product and one Chief Underwriter responsible for the whole quote. He must confirm every completed product preparation. He also has the rights to re-initiate any of the Prepare product activities, assign particular Underwriter to a particular *product preparation* activity, and terminate the whole process by quote completion. This sub-process was the subject of redesign in accordance with our pattern and the result is depicted in (Fig. 2).

We applied several specific customizations of the general pattern presented in this paper due to the specifics of Bizagi BPMS:

1. Business rules were hard-coded into a complex event gateway
2. Re-assignation messaging was not applied as Bizagi BPMS has built-in features for delegation
3. The activity of *Manage Quote* allows the Quote supervisor to perform all management tasks, such as activity re-assignation, delegation, termination of activities, parametric changes in constraint conditions, etc. and has the same role as the Managing process in our pattern

5.3 Results

Redesigned processes were deployed for testing in one BU and offered to others. Within two months, other local project coordinators of three other countries also requested the redesign of their version of the process according to this approach. Finally, four out of five BUs are now working according to the new processes. In
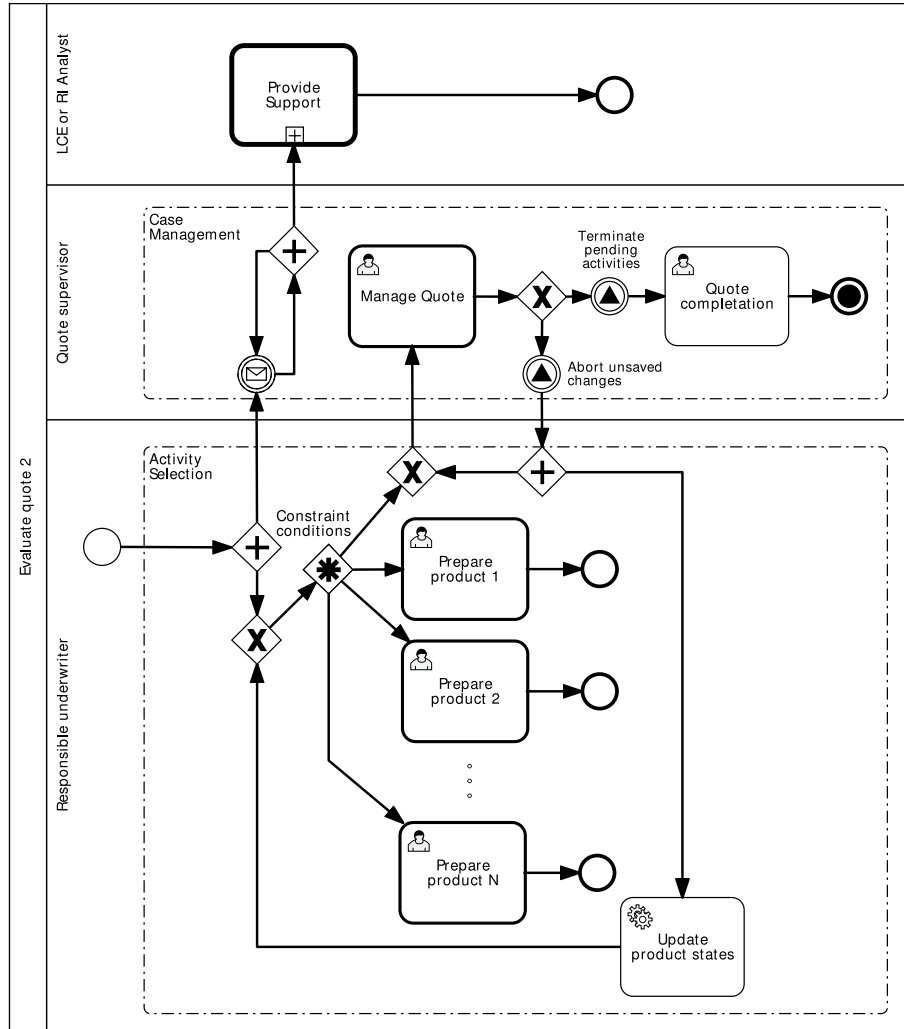
Fig. 2. Figure 2. (Evaluate quote process)

two BUs the process change was deployed into production environment, in the other two the change was made during the pilot system testing period. For evaluation of results, three employees from each BU were asked to provide feedback. In each BU, each of the following roles was represented by one individual

– Project Coordinator (PC) - the person responsible for the definition of requirements on the system, customization of the processes in respective BU and moderation of the communication between the users and the system developers.
– Chief Underwriter (CU) - the person responsible for managing the underlying sub-processes, thus the performer of the Managing Process

– Underwriter (U) - the employee performing particular ad-hoc ordered product preparation tasks.

Each of them was asked to compare the former and the redesigned process and rate the impact of the process redesign against the aspects listed in (Table 5.3). For each aspect, there were the following possible ratings: "-1" for worse than before the change, "0" for no improvement, "1" for improvement and "-" for no answer.

.

	BU1 (production)			BU2 (production)			BU3 (pilot)			BU4 (pilot)			Sum per aspect <-12,12>
Role	PC	CU	U	PC	CU	U	PC	CU	U	PC	CU	U	
Process model clarity	-1	0	0	0	-1	0	0	-1	0	0	0	0	-3
Work-flow flexibility	1	1	1	0	1	1	-	1	1	-	1	1	9
Time efficiency	-	0	1	-	0	1	-	1	1	1	0	1	6
Process manageability	-	1	1	1	1	1	1	0	1	-	1	-1	7
Sum per role <-4,4>	0	2	3	1	1	3	1	1	3	1	2	1	
Sum per BU <-12,12>	5			5			5			4			

According to acquired figures in (Table 5.3), there is a slight decrease of process model clarity observed by the Project Coordinators and some of the Chief Underwriters, the main users of the process model. On the other hand, there was a significant improvement of work-flow flexibility observed by all the Underwriters and more than half of the Chief Underwriters, as they could operate more flexibly without dependence on each other's tasks. As we can see in (Table 5.3), the overall perception of the process redesign was rather positive, especially in BUs where the system was deployed into the production environment with more intensive system usage. The best rating was generally given by Underwriters, thus we claim that the highest improvement was perceived by ad-hoc task performers.

6 Discussion and Conclusions

The presented approach evolved from the request for process redesign during the elaboration of the presented project. We developed the approach as a general solution of the change request, as generally applicable pattern for the modeling of ad-hoc processes with standard BPMN. We consider the approach versatile

enough to be used within most modern BPMS platforms. We further customized this approach for purposes of the presented project and applied it during the process redesign. The redesign impact was evaluated by several process performers and well accepted across four out of five BUs of the company, as we describe in Results section. Our future aim is to verify this approach in different business environments and to perform further extensions to establish a versatile best practice, which can be used in the context of our BPM4SME methodology.

During the advancement of the approach, we noticed interesting fact that will be the subject of further research. The recorded sequences of activities in ad-hoc parts of the process produce data, which can be used for several purposes. First, we are able to use it to find common ordering patterns related to process data. Second, we can use them to improve our constraints on fly to make them more restrictive to achieve higher determinism. Third, these data can be used for a semi-automatic process discovery. Therefore, in the future, we plan to perform a process discovery experiment in an organization interested in BPM adoption.

Concerning limitations, we have to admit that it can be used to extend process Flexibility and partially process Adaptability. However, it does not provide any improvement to process Dynamism as it was defined at the beginning of this paper. Therefore, once we want to add completely new activities into an ad-hoc process, we still have to solve the same problems related to process model changes in a traditional approach to process modeling. This problem remains unsolved and we are not able find any satisfactory solution applicable on a BPMN level. There were some discussions about generic activities, but they are not defined in BPMN standards and such a solution could be done by hacking particular BPMS. All other proposed solutions ended up with this result as well. Something similar is described by [27], but this is also a proprietary solution and far from being generalized the way our pattern is. Therefore, this problem, which could significantly extend our approach, remains as another challenge for further research.

References

1. Jeston, J., Nelis, J.: Business Process Management: Practical Guidelines to Successful Implementations. Taylor & Francis Group (2006)
2. van der Aalst, W.M.P.: The application of Petri nets to workflow management. The Journal of Circuits, Systems and Computers **8**(1) (1998) 21–66
3. Ramadan, M., Elmongui, H.G., Hassan, R.: Bpmn formalisation using coloured petri nets. Proceedings of the 2nd GSTF Annual International Conference on Software Engineering & Applications (SEA 2011) (1997)
4. Ko, R.K.L., Lee, S.S.G., Lee, E.W.: Business process management (BPM) standards: a survey. Business Process Management Journal **15**(5) (2009) 744–791
5. Kolar, J.: Business activity monitoring. Master's thesis, Masaryk University (2012)
6. Rudden, J.: Making the case for bpm–a benefits checklist. BPTrends (2007)
7. Mertens, W., Van den Bergh, J., Viaene, S., Schroder-Pander, F.: How bpm impacts jobs: An exploratory field study. In: System Sciences (HICSS), 2011 44th Hawaii International Conference on. (2011) 1–10

8. Magal, S.R., Word, J.: Essentials of Business Processes and Information Systems. 1st edn. Wiley Publishing (2009)

9. Swenson, K.D.: Mastering the Unpredictable: How Adaptive Case Management Will Revolutionize the Way That Knowledge Workers Get Things Done. 1st edn. Meghan-Kiffer Press, Tampa, FL (April 2010)

10. van der Aalst, W.M., Weske, M., Grnbauer, D.: Case handling: A new paradigm for business process support. Data and Knowledge Engineering **53** (2005) 2005

11. Schonenberg, H., Weber, B., Dongen, B., Aalst, W.: Supporting flexible processes through recommendations based on history. In: Proceedings of the 6th International Conference on Business Process Management. BPM '08, Berlin, Heidelberg, Springer-Verlag (2008) 51–66

12. Sadiq, S., Sadiq, W., Orlowska, M.: Pockets of flexibility in workflow specifications. In: In Proc. ER01 Conf. (2001) 513–526

13. Pesic, M.: Constraint-Based Workflow Management Systems: Shifting Control to Users. PhD thesis, Eindhoven University of Technology (2008)

14. Deng, S., Yu, Z., Wu, Z., Huang, L.: Enhancement of workflow flexibility by composing activities at run-time. In: Proceedings of the 2004 ACM symposium on Applied computing. SAC '04, New York, NY, USA, ACM (2004) 667–673

15. Wyssusek, B., Schwartz, M., Kremberg, B., Baier, F., Krallmann, H.: Business process modelling as an element of knowledge management - a model theory approach (2001)

16. Kalpic, Brane; Bernus, P.: Business process modeling through the knowledge management perspective. Journal of Knowledge Management, Vol. 10(3) **53** (2006) 40–56

17. Jim Sinur, J.B.H.: Magic quadrant for business process management. Gartner RAS Core Research Note G00205212, Gartner (10 2010)

18. Schonenberg, H., Mans, R., Russell, N., Mulyar, N., Aalst, W.: Process flexibility: A survey of contemporary approaches. In Dietz, J., Albani, A., Barjis, J., eds.: Advances in Enterprise Engineering I. Volume 10 of Lecture Notes in Business Information Processing. Springer Berlin Heidelberg (2008) 16–30

19. Imanipour, Narges, T.K., Rezazadeh, S.: Obstacles in business process management (bpm) implementation and adoption in smes. working papers series (2012)

20. van der Aalst, W., ter Hofstede, A., Kiepuszewski, B., Barros, A.: Workflow patterns. Distributed and Parallel Databases **14**(1) (2003) 5–51

21. Aalst, W.M.P.v.d., Barros, A.P., Hofstede, A.H.M.t., Kiepuszewski, B.: Advanced workflow patterns. In: Proceedings of the 7th International Conference on Cooperative Information Systems. CooplS '02, London, UK, UK, Springer-Verlag (2000) 18–29

22. Russell, N., van der Aalst, W.M.P., Arthur: Exception Handling Patterns in Process-Aware Information Systems. Technical report, BPMcenter.org (2006)

23. Russell, N., ter Hofstede, A.H.M., van der Aalst, W.M.P., Mulyar, N.: Workflow Control-Flow Patterns: A Revised View. Technical report, BPMcenter.org (2006)

24. Russell, N., ter Hofstede, A.H.M., Edmond, D., van der Aalst, W.M.P.: Workflow data patterns: identification, representation and tool support. In: Proceedings of the 24th international conference on Conceptual Modeling. ER'05, Berlin, Heidelberg, Springer-Verlag (2005) 353–368

25. Sadiq, S.W., Orlowska, M.E., Sadiq, W.: Specification and validation of process constraints for flexible workflows. Inf. Syst. **30**(5) (July 2005) 349–378

26. Mangan, P., Sadiq, S.: On building workflow models for flexible processes. Aust. Comput. Sci. Commun. **24**(2) (January 2002) 103–109

27. Adams, M., Hofstede, A.H.M.T., Edmond, D.: Worklets: A service-oriented implementation of dynamic flexibility. In: in Workfows, 2005
28. Aalst, W., Adams, M., Hofstede, A., Pesic, M., Schonenberg, H.: Flexibility as a service. In: Database Systems for Advanced Applications. Volume 5667 of Lecture Notes in Computer Science. Springer Berlin Heidelberg (2009) 319–333
29. Weber, B., Rinderle, S., Reichert, M.: Change patterns and change support features in process-aware information systems. In: Proceedings of the 19th international conference on Advanced information systems engineering. CAiSE'07, Berlin, Heidelberg, Springer-Verlag (2007) 574–588
30. van der Aalst, W., Pesic, M., Schonenberg, H.: Declarative workflows: Balancing between flexibility and support. Computer Science - Research and Development **23**(2) (2009) 99–113
31. Pesic, M., Schonenberg, H., Aalst, W.: Declarative workflow. In Hofstede, A.H.M., Aalst, W.M.P., Adams, M., Russell, N., eds.: Modern Business Process Automation. Springer Berlin Heidelberg (2010) 175–201
32. Mulyar, N., Pesic, M., Van Der Aalst, W.M.P., Peleg, M.: Declarative and procedural approaches for modelling clinical guidelines: addressing flexibility issues. In: Proceedings of the 2007 international conference on Business process management. BPM'07, Berlin, Heidelberg, Springer-Verlag (2008) 335–346
33. Van der Aalst, W.M.P., Giinther, C.: Finding structure in unstructured processes: The case for process mining. In: Application of Concurrency to System Design, 2007. ACSD 2007. Seventh International Conference on. (2007) 3–12
34. Maggi, F., Mooij, A., Van der Aalst, W.M.P.: User-guided discovery of declarative process models. In: Computational Intelligence and Data Mining (CIDM), 2011 IEEE Symposium on. (2011) 192–199
35. Singer, R., Zinser, E.: Business process management - s-bpm a new paradigm for competitive advantage? In Buchwald, H., Fleischmann, A., Seese, D., Stary, C., eds.: S-BPM ONE. Volume 85 of Communications in Computer and Information Science., Springer (2009) 48–70
36. Kolar, J.: Process analysis at it logica s.r.o. Business analytical document, Jiri Kolar, BPM analyst (2011)
37. Kolar, J.: Process analysis at ict department faculty of arts masaryk unversity. Business analytical document, Jiri Kolar, BPM analyst (2011)
38. Kolar, J., Pitner, T.: Collaborative process design in cloud environment. In Haller, A., Huang, G., Huang, Z., Paik, H.y., Sheng, Q., eds.: Web Information Systems Engineering - WISE 2011 and 2012 Workshops. Volume 7652 of Lecture Notes in Computer Science. Springer Berlin Heidelberg (2013) 55–69
39. Kolar, J., Pitner, T.: Agile bpm in the age of cloud technologies. Scalable Computing: Practice and Experience **13**(4) (2012)
40. Rozier, K.Y.: Survey: Linear temporal logic symbolic model checking. Comput. Sci. Rev. **5**(2) (May 2011) 163–203
41. Ould, M., Ltd, V.C., Society, B.C.: Business Process Management: A Rigorous Approach. Meghan-Kiffer Press (2005)