

Supporting and Assisting the Execution of Knowledge-intensive Processes

Steven Mertens

Department of Business Informatics and Operations Management
Faculty of Economics and Business Administration
Ghent University, Tweekerkenstraat 2, 9000 Ghent, Belgium
steven.mertens@ugent.be

Supervisors: Frederik Gailly and Geert Poels

Abstract. Business processes management (BPM) offers the tools to model business processes. In order to implement and support the created process models, business process management systems (BPMS) have been introduced. Such a system handles the coordination between the process actors and makes sure that the real process is executed in conformance to the specified model. The existing BPMSs focus on tightly framed processes and offer little to no support for less tightly framed processes. Knowledge-intensive processes (KiP) are an important class of such loosely framed processes. In this research project we want to develop a BPMS architecture that offers both support and assistance during the execution of KiPs. Some components of traditional BPMSs can be reused, but other components will have to be adapted or even created to fit the new purpose. In this paper we present the problem description, research goals, methodology and current status of the research project.

Keywords: Knowledge-intensive Processes, Business Process Management Systems, Decision Support

1 Introduction

A business process is a set of one or more connected activities which collectively realize a particular business goal. Business process management (BPM) includes concepts, methods, and techniques to support the design, administration, configuration, enactment, and analysis of business processes [1]. Process models are used to facilitate communication between business stakeholders, to analyze and redesign the as-is business process and finally are put into execution by the Business Process Management Systems (BPMS) of the organization. The goals of applying BPM are a better understanding of the process and to continually upon it.

Business processes can be classified according to level of utilization of process models [2]. First, unframed processes do not have an explicit process model (e.g., collaborative processes supported by groupware systems). Next, a process is ad-hoc framed if it has a predefined process model, but this model is only used a small num-

ber of times before it is discarded or changed (e.g., adaptive case management). When the process is a bit more structured it is called loosely framed. This entails that a process model allowing ‘the normal way of doing things’ and some deviations is a priori defined by way of a set of constraints. Lastly, a tightly framed process consistently follows a predefined and unambiguous process model (e.g., traditional process engines and workflow management systems).

The design and administration of tightly framed processes is supported by well-known imperative process modeling languages like UML, BPMN, and EPC’s. The design and administration of loosely framed business processes, however, is only supported by a limited set of declarative process modeling languages (e.g. Declare). While imperative approaches focus on explicitly defining the exact path of activities to reach the process goals, declarative approaches determine only the activities that may be performed as well as constraints prohibiting undesired behavior [3].

The configuration and enactment of tightly framed business processes is supported by BPMS, which enable the execution of the business process by means of imperative process model. The configuration and enactment of loosely framed person-2-person (P2P) processes is harder to realize for several reasons [4]. For one, the development of declarative process modeling languages is still in its infancy. Different declarative process modeling approaches have been proposed which support the specification of different business concerns, the specification of different constraint types and use different reasoning paradigms [5]. Moreover, in contrast to imperative process modeling languages, it is not clear how declarative process models in general can be transformed into executable models that can be used directly by the BPMS. Additionally, the participants of loosely framed P2P processes are primarily knowledge workers (e.g., doctors) who decide in which order activities need to be performed based on business data and past experience. As a consequence the configuration and implementation of these processes requires a tight integration of processes data, business data and users. Currently available BPMSs have their origin in workflow management systems, which are primarily used to support routine and structured processes and, as a consequence, do not support this kind of integration.

The general objective of this research project is the specification of a BPMS architecture for KiPs. This requires the development of concepts, methods and techniques for the configuration, enactments and run-time analysis of knowledge-intensive business processes (KiPs). Different BPM researchers have recently recognized the need to extend existing techniques to support KiPs [3, 5, 6]. KiPs correspond to loosely framed P2P processes and are becoming more and more relevant. A typical example of an environment where a lot of KiPs are executed is a healthcare organization. A lot of healthcare processes are loosely framed P2P processes in which the doctor and nurse are the knowledge workers who will decide which path the patient will follow, taking into account certain preferences, conditions and norms.

The decision making process of knowledge workers is considered to be the essence of KiPs. This leads to an important secondary goal of this research project: offering decision support during the execution of these processes. This decision support will be based on a combination of existing techniques from operations research and knowledge management. The former has been used in the past by the BPM discipline

for model-based process analysis (e.g. simulation, queuing theory). The latter has mainly been used in the context of process mining, which aims to discover, monitor and improve real processes by extracting knowledge from event logs [7]. Process mining techniques have been successfully used to analyze the logs of business processes which already have been completed (i.e., offline). In this project we focus on knowledge extraction techniques that can be used for online decision support. Some techniques for online decision support using process mining techniques have been proposed [8–10], but it still remains a challenge in the context of KiPs. Note, the ultimate goal is not to automate decision making processes (i.e., expert systems), but rather to offer support to the knowledge workers during a decision making process.

2 Research goals

The general objective of this project is translated into four research goals: developing an architecture for KiP management systems, making declarative process models executable, making tacit decision knowledge explicit by analyzing the decisions of knowledge workers and assisting knowledge workers when making path decisions.

2.1 Developing an architecture for KiP management systems

A standard architecture for workflow management systems has been published in 1995 [11]. It is our intent to create a similar architecture for KiPs. The three research goals discussed below are components that we identified as missing in the original architecture. Additionally, we need to evaluate which components of the original architecture will be useful in this new context. The resulting architecture will thus integrate the outcomes of the other three research goals, as well as some existing components and possibly other to be determined components inherent to KiPs.

2.2 Making declarative process models executable

Business process models typically follow one or more modeling perspectives [12]. For example, analytical BPMN models focus on modeling the activities (i.e. functional perspective) and control flow (i.e. behavioral perspective) of business processes, and less on the data (information perspective) and resources (organizational perspective) needed by these processes. Making a process model executable corresponds to focusing or extending a perspective which was previously not taken into account by the model. For instance, for BPMN this could be specifying how the different services can be implemented by application services (operation perspective) or how user tasks can be assigned to organizational resources (organizational perspective).

KiPs are typically modeled by means of declarative business process models which take a rule-based perspective on process models. The second research goal of this research project investigates what it means to transform a declarative process model into an executable process model. This transformation is different for declarative process models because the control flow of a KiP cannot be specified at build-time,

but instead is only determined at run-time by the knowledge worker. Additionally, in the context of KiPs the different perspectives are also more integrated and, as a consequence, run-time coordination between the perspectives is needed. The execution of a specific path should take into account the rules and constraints of the KiP. On the other hand it is also influenced by the decisions made by the knowledge worker and the relevant data that is available. A possible solution is presented by Barba et al. [13], in which an enactment plan is generated from a given declarative process model by means of constraint programming. These enactment plans can be transformed directly into BPMN models [14]. An enactment plan is essentially a simple and imperative sequence of activities that complies with the declarative model. However, since only enactment plan is calculated at a time, which is insufficient to handle the dynamic and flexible nature KiP, we would have to enumerating all possible enactment plans. This is not feasible for realistic cases as this would result in an enormous amount of enactment plans. Therefore, we would like to find a way to generate a specific subset of all possible imperative models that comply with the declarative model.

2.3 Making tacit decision knowledge explicit by analyzing the decisions of knowledge workers

KiPs are inherently people-centric [4]. Each knowledge worker has a specific background, expertise and experience and will leverage this to make the decision on which activity to do next during the execution of the process. These decisions are driven by the status and availability of data and knowledge objects. Traditionally in BPMS a distinction is made between application data, process-relevant data and process control data [15]. For knowledge-intensive BPMS the distinction between application-data and process-relevant data is less clear. For instance, the data of the patient (application data) in combination with the availability of resources (process-relevant data) will be used by the doctor to decide which execution path shall next be taken (process-control data). The third research goal of this project focuses on identifying knowledge management techniques that support the creation of new knowledge objects by extracting and integrating information from application, process-relevant and process control data. For example, the created knowledge object can correspond to a set of decision rules, extracted from historical process control data. This externalizes the tacit knowledge of experienced knowledge workers into guidelines for less experienced knowledge workers. These less experienced knowledge workers could in turn contribute knowledge about more state-of-the-art research or just provide an out-of-the box vision.

2.4 Assisting knowledge workers when making path decisions

An apparent paradox exists between providing guidance and run-time flexibility [16]. Guidance is often thought of as forcing the user in a certain direction. In contrast, run-time flexibility can be only realized if the knowledge worker is not forced to execute a certain activity next. There is however a suitable middle ground: a BPMS offering the knowledge workers specific recommendations on what he could do next. This does

not introduce any new restrictions, as the user can still decide not to follow the recommendations. The idea of using recommendation systems during the operational support for KiPs is not new [17, 18].

A key aspect to consider when making a recommendation system is what to convey to the users. A single recommendation might be enough for smaller decisions, but in the context of KiPs this will be insufficient. A list of recommendations will be more appropriate as decisions tend to be complex. This list can be provided as-is or in a specific order. The latter is preferable as it gives the users more information. Additionally, we can make the sorting criteria and predicted or known consequences explicit in a clear manner to give the user even more insight into the decision. The operations research side and the knowledge management side will both contribute to the sorting criteria. The former provides advanced analytical methods that focus on concepts like efficiency, while the latter uses the accumulated knowledge base to account for preferences and hidden norms or rules. The use of operations research techniques in recommendation systems is not completely new and has been investigated to a smaller extent by Barba et al. [13]. Similarly, Schonenberg et al. [18] have also touched the surface of using knowledge management in this context. This research project aims to find deeper roots in both areas to create a combined technique.

3 Methodology

In the Information Systems domain the design science research is considered as a generally accepted research methodology [19, 20]. Typically, design science research consists of the following phases: 1) motivation of the problem, 2) definition objectives of the solution, 3) design and development, 4) demonstration 5) evaluation and communication [20]. The research project must result in the identification of the requirements for the four design science artifacts which will be developed:

1. An BPMS architecture for KiPs
2. A method for transforming declarative process models into executable models
3. A business process recommendation generator that combines operations research and knowledge management techniques
4. A method for creating a decision knowledge base using application data, process-relevant data and process control data

The research structure is visualized using design cycles [21] in **Fig. 1**. An engineering cycle (EC) provides the necessary steps to design and evaluate an artefact. A research cycle (RC) is responsible for resolving research related issue (e.g., establishing the state of the art for a problem, finding and adapting related techniques...) [22].

The main engineering cycle (EC1) will result in the specification of a BPMS architecture for KiPs and a prototype. This cycle has three smaller engineering cycles (EC2-EC4) and one research cycle (RC6).

In the second engineering cycle (EC2) a method for making declarative process models executable will be developed. Before we can do this, we will first need to perform the first small research cycle (RC1): identify and assess the available declara-

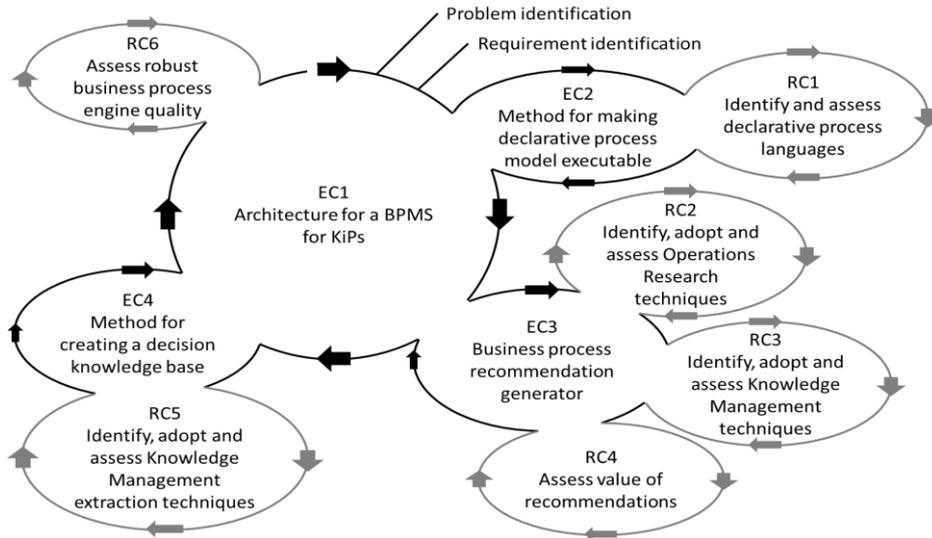


Fig. 1. Research structure in terms of regulative design cycles

tive process languages in the context of KiPs. If no languages are found to be sufficient, a new language will be developed during the project.

The third engineering cycle (EC3) will create the blueprints for a business process recommendation generator. This artifact should be able to generate a ranked list of recommended next activities. Two research cycles, RC2 and RC3, will identify and adopt techniques from, respectively, operations research and knowledge management. The criteria (e.g., estimated duration/cost, built-time flexibility, run-time flexibility, historic compliance...) to rank the recommendations will be produced in these cycles. In the last research cycle (RC4) of EC3, we will evaluate the value of these recommendations. The effect of having ranked recommendations will be compared to having unranked recommendations) and having no recommendations at all.

In the last engineering cycle, EC4, a method for creating a decision knowledge base will be developed. This starts by logging all relevant data (i.e., which activities are performed, resource availabilities, data generated during activities and general information about the people involved in the process). In the next phase, this logged data will be analyzed using decision mining techniques. These techniques are the outcome of RC5, which will identify, adopt and assess knowledge management techniques for extracting knowledge from raw data. The decision knowledge base will be used as input for the knowledge management techniques from RC3 for the ranking of recommendations and, if a certain previously hidden rule is confirmed by users or domain experts, will be used to improve the general process model.

Finally, the BPMS for KiPs will be evaluated in the last research cycle (RC6). The prototype of the system will be assessed in a real context (e.g., emergency department of a hospital). This assessment will primarily be performed by the actual users of the system and measure their thoughts on the potential and possible shortcomings of the system. The latter can then be examined in order to improve the system.

4 Current status

We started with the problem identification as was briefly discussed in the introduction of this paper. From there on we created a list of requirements for the system, which translated into EC1, EC2, EC3 and EC4. In order to evaluate the system, a target environment and evaluation setup was specified (RC6).

The first obstacle, making a declarative model executable, was tackled in combination with the recommendations. The idea arose to equate a declarative process model to a set of executable imperative process models. It is even possible to create one imperative model that is equivalent [23], but this would be a very complex and unclear model for realistic KiPs. So we decided to use a set of what we call ‘simple’ imperative process models. These models have one direct path from start to end with possibly one or more loops, even nested loops, on the way. Each imperative model in essence represents one trace with all of its variations based on repeated sequences within the trace. This is a trade-off between simplicity and the amount of possible models. These models are a bit more complex than, for example, the enactment plans proposed by Barba et al. [13], but they remain clear and understandable. At the same time, each of these models can represent a whole set of enactment plans. This dramatically reduces the number of models in the set of imperative models equivalent to a given declarative model. The chosen representation also corresponds very well to what are called sequential and iterative care processes in healthcare [24].

Theoretically, this allows us to keep track of all paths that are still available, represented by a set of imperative models, at each point during the execution of the declarative model. In practice however, we expect there to be numerous imperative models in set that is equivalent to a practical declarative model. This means that it would be very time consuming to generate the complete set, so we will possibly have to make due with knowing only a subset of the complete set of imperative models in order to keep the system usable. But since each imperative model represents many activity traces, this subset will still cover a comprehensive set of possible cases.

In the next step, we identified and compared the available declarative process modeling languages in order to find a suitable language for this purpose. All candidate languages had their shortcomings, but we eventually chose Declare [16, 25] (formerly known as ConDec), and more specifically the Declare-R extension [26], as this was at the time by far the most popular of them all (of course mainly in the research community). However, we demonstrated by way of a realistic case that Declare is insufficiently expressive to capture some the important knowledge concerning the decision making process required to model and offer guidance to users of KiPs. This resulted in a paper that is accepted to the BPMDS’15 working conference [27]. The paper also proposes the basics of a new extension, Declare-R-DMN, which bridges the Declare-R language with DMN [28]. DMN is a standard for decision logic that was just recently adopted by OMG. This will be further elaborated to create a formal metamodel and corresponding language.

In conjunction with the previous step, a blueprint for a recommendation generator was developed. This uses a variant of a genetic algorithm, called a population based meta-heuristic, to generate a subset of executable imperative process models that

conform to a given declarative process model. This subset is optimized according to fitness criteria from both the operations research and knowledge management domains. A prototype was created which currently only supports Declare, but support for Declare-R-DMN will be added as that language is formalized. A paper describing this algorithm was accepted and recently published [29].

The further elaboration of EC2 and EC3 is planned as the next phase of this research project. Meanwhile, in order to get a deeper insight in the healthcare industry, we also plan to meet with people involved in this service branch and discuss their vision on this project. As a starting point, a paper describing the intent of this project has been accepted and will be discussed at the ProCare workshop of the International Conference on Pervasive Computing Technologies for Healthcare 2015.

References

1. Weske, M.: *Business Process Management: Concepts, Languages, Architectures*. Springer (2012).
2. Van der Aalst, W.M.P.: *Business Process Management: A Comprehensive Survey*. *ISRN Softw. Eng.* 2013, 1–37 (2013).
3. Haisjackl, C., Barba, I., Zugal, S., Soffer, P., Hadar, I., Reichert, M., Pinggera, J., Weber, B.: *Understanding Declare models: strategies, pitfalls, empirical results*. *Softw. Syst. Model.* (2014).
4. Di Ciccio, C., Marrella, A., Russo, A.: *Knowledge-Intensive Processes: Characteristics, Requirements and Analysis of Contemporary Approaches*. *J. Data Semant.* 29–57 (2014).
5. Goedertier, S., Vanthienen, J., Caron, F.: *Declarative business process modelling: principles and modelling languages*. *Enterp. Inf. Syst.* 1–25 (2013).
6. Mulyar, N., Pesic, M., van der Aalst, W.M.P., Peleg, M.: *Declarative and procedural approaches for modelling clinical guidelines: addressing flexibility issues*. *BPM Workshops. LNCS*, vol. 4928. pp. 335–346. Springer (2007).
7. Van der Aalst, W.M.P.: *Process Mining*. Springer (2011).
8. Ghodspour, S.H., O'Brien, C.: *A decision support system for supplier selection using an integrated analytic hierarchy process and linear programming*. *Int. J. Prod. Econ.* 56-57, 199–212 (1998).
9. Shim, J.P., Warkentin, M., Courtney, J.F., Power, D.J., Sharda, R., Carlsson, C.: *Past, present, and future of decision support technology*. *Decis. Support Syst.* 33, 111–126 (2002).
10. Rozinat, A., Wynn, M., van der Aalst, W.M.P., ter Hofstede, A.H., Fidge, C.J.: *Workflow simulation for operational decision support*. *Data Knowl. Eng.* 68, 834–850 (2009).
11. Hollingsworth, D.: *The Workflow Reference Model*. (1995).
12. Krogstie, J.: *Perspectives to process modeling - A historical overview*. *Enterprise, Business-Process and Information Systems Modeling. LNBIP*, vol. 113. pp. 315–330. Springer (2012).
13. Barba, I., Weber, B., Del Valle, C., Jiménez-Ramírez, A.: *User recommendations for the optimized execution of business processes*. *Data Knowl. Eng.* 86, 61–84 (2013).

14. Barba, I., Del Valle, C.: Automatic generation of optimized business process models from constraint-based specification. *Int. J. Coop. Inf. Syst.* 22, (2013).
15. Hollingsworth, D.: The Workflow Reference Model: 10 Years On. *The Workflow Handbook*. pp. 295–312. Workflow Management Coalition (2004).
16. Van der Aalst, W.M.P., Pesic, M., Schonenberg, H.: Declarative workflows: Balancing between flexibility and support. *Comput. Sci. - Res. Dev.* 23, 99–113 (2009).
17. Jiménez-Ramírez, A., Barba, I.: Generating multi-objective optimized business process enactment plans. *Advanced Information Systems Engineering. LNCS*, vol. 7908. pp. 99–115. Springer (2013).
18. Schonenberg, H., Weber, B.: Supporting flexible processes through recommendations based on history. *BPM. LNCS*, vol. 5240. pp. 51–66. Springer (2008).
19. Hevner, A.R., March, S.T., Park, J., Ram, S.: Design science in Information Systems research. *MIS Q.* 28, 75–105 (2004).
20. Peffers, K., Tuunanen, T., Rothenberger, M.A., Chatterjee, S.: A Design Science Research Methodology for Information Systems Research. *J. Manag. Inf. Syst.* 24, 45–77 (2007).
21. Wieringa, R.J.: Design science as nested problem solving. *Proceedings of the 4th International Conference on Design Science Research in Information Systems and Technology - DESRIST '09*. pp. 1–12. ACM Press, New York (2009).
22. Wieringa, R.J., Heerkens, J.M.G.: The methodological soundness of requirements engineering papers: a conceptual framework and two case studies. *Requir. Eng.* 11, 295–307 (2006).
23. Prescher, J., Ciccio, C. Di, Mendling, J.: From Declarative Processes to Imperative Models. *Proceedings of the 4th International Symposium on Data-driven Process Discovery and Analysis, Milan, Italy*. pp. 162–173 (2014).
24. Bohmer, R.M.J.: *Designing care - Aligning the nature and management of health care*. Boston: Harvard Business Press (2009).
25. Pesic, M.: *Constraint-based workflow management systems: shifting control to users*, (2008).
26. Barba, I., Del Valle, C.: Filtering rules for ConDec templates - Pseudocode and complexity, <http://www.lsi.us.es/quivir/irene/FilteringRulesforConDecTemplates.pdf>, Accessed on 9 October 2014.
27. Mertens, S., Gailly, F., Poels, G.: Enhancing declarative process models with DMN decision logic. *Enterprise, Business-Process, and Information Systems Modelling. LNBIP*, vol. 214. Springer (2015).
28. OMG: Decision Model and Notation (DMN), www.omg.org/spec/DMN/Current/, Accessed on 10 November 2014.
29. Mertens, S., Gailly, F., Poels, G.: Generating business process recommendations with a population based meta-heuristic. *BPM Workshops. LNBIP*, vol. 202. pp. 516–528. Springer (2014).