Requirements Engineering Based on Business Process Models: A Case Study

Evellin Cristine Souza Cardoso, João Paulo A. Almeida and Giancarlo Guizzardi
Ontology and Conceptual Modelling Research Group (NEMO)
Computer Science Department, Federal University of Espírito Santo (UFES), Brazil
ecardoso@inf.ufes.br, jpalmeida@ieee.org, gguizzardi@inf.ufes.br

Abstract

This paper reports a study in which business process modelling was regarded as a useful tool for requirements engineering. We have used business process models as a starting point to derive alternative sets of requirements for a process-oriented system. These alternative sets capture different decisions regarding the intended "level of automation" for the various activities in a business process. The approach is demonstrated in a case study which consists of the development of a real system to manage processes in a Human Resources (HR) Department of a large organization. We use a requirements specification that was previously obtained through what we characterize as a "conventional" requirements engineering technique and we compare it with the requirements specifications derived from the business process model. We discuss the impact of the use of business process models in the completeness, correctness, consistency and traceability of requirements in the case study.

1. Introduction

Requirements engineering is "the systematic process of developing requirements through an iterative process of analyzing a problem, documenting the resulting observations, and checking the accuracy of the understanding gained" [1].

The main objective of requirements engineering is to understand a customer's needs, problems to be solved before system development, the delimitation of system boundaries, as well as other types of constraints imposed to the solution, such as economic, technical, systemic, environmental, time and resource constraints.

Since requirements engineering involves the attempt to conciliate the stakeholders' viewpoints involved in the design process [2], its success is directly related to the efficient communication between stakeholders. Without this, the development process can lead to systems whose functionalities are unnecessary, incorrect or which do not reflect organizational objectives and activities. Consequently, there is an increasing necessity for methodologies or techniques which bridge the gap caused by deficient communication. In this paper, we consider the role of *business process modelling* as a means to address the gap between those defining and performing organizational activities and those developing the systems which support the execution of organizational activities.

Business process modelling consists of an activity whose main goal is to formalize business processes in an organization, capturing the context in which these processes are executed [8]. It allows the documentation and communication of organizational activities which compose business processes and also allows the identification of the computational support provided by systems to these activities.

We use business process models as a starting point to obtain system requirements, and argue that business process modelling can facilitate requirements engineering. In particular, the availability of business process models allows the analyst to consider alternative sets of requirements based on varying the "level of automation" for the various activities in a business process. The approach is demonstrated in a case study which consists of the development of a system to manage processes in a Human Resources (HR) Department of a large organization in the energy industry. We use a requirements specification that was previously obtained through a conventional requirements engineering technique by an independent team and we contrast it with a requirements specification derived from the business process model as proposed in this paper.

This paper is structured as follows: section 2 characterizes what we refer to as the "conventional" requirements engineering approach in the scope of this work; section 3 presents the "process-oriented" requirements engineering approach to be contrasted with the "conventional" approach presented in section 2; section 4 compares the results obtained by applying both techniques in a particular process of the Human Resources Department involved in the study and section 5 discusses related work. Finally, our general conclusions are presented in section 6, as well as recommendations for future work.

2. Conventional requirements engineering approach

The first technique we consider for requirements elicitation is denominated "conventional approach" in this paper[1]. This technique is well-established in the organization in which this study has been conducted (the Human Resources Department of a large organization in the energy industry), and is well-documented in private documents. It consists of a set of interviews performed by system analysts with the customer or client. In the case study reported here, the customers are employees of the HR department that are prospective users of a system to manage certain activities of the department.

The approach includes a preliminary problem identification phase whose main product is an informal description of the problem to be solved in the business context of the customer. Subsequently, in the absence of existing solutions for the problem identified, business macro-requirements must be elicited, determining clearly the system boundaries and scope. The next step involves approval of the scope negotiation of the terms of project execution.

In the subsequent phase of detailed problem description, the main product is a business requirements specification in a textual format. Requirements are classified in this phase as functional requirements, non-functional requirements and inverse requirements. After the completion of this phase, a document with a list of business rules and a glossary which clarifies the domain terminology area also generated.

The subsequent phase has the main purpose of constraining the solution space, and therefore, this phase is called solution phase (which focuses on the system to be conceived). The final product of this phase is a system requirements specification. System requirements in this phase are divided into functional requirements and non-functional requirements; the former are specified through use cases and the latter are defined in terms of measurable metrics.

After detailing the use cases, the requirements engineering phase is complete and the design phase starts. During design, business requirements as well as system requirements can undergo changes in an iterative approach. Table 1 summarizes the phases in the "conventional approach" and the products obtained in each phase.

[1] What we refer to as "conventional approach" to requirements engineering should not be interpreted as a generalization of the state-of-the-practice in requirements engineering. We use the term here only to denote the established practice in the organization in which the study has been conducted.

Table 1. Requirements Engineering phases and products obtained in each of these phases

<table>
<tr><th>Phase</th><th>Product</th><th>Requirement Type</th><th>Domain</th></tr>
<tr><td>Preliminary problem identification phase</td><td>Macro-Requirements Specification</td><td>Business (macro-requirements)</td><td rowspan="4">Problem</td></tr>
<tr><td rowspan="3">Detailed problem description phase</td><td>Business Solution Document</td><td>Business (functional, non-functional and reverse)</td></tr>
<tr><td>Business Rules</td><td rowspan="2">Business</td></tr>
<tr><td>Glossary</td></tr>
<tr><td rowspan="2">Solution phase (system)</td><td>Use Cases Specification</td><td>System (functional)</td><td rowspan="2">Solution (System)</td></tr>
<tr><td>Complementary Specification</td><td>System (non-functional)</td></tr>
</table>

3. Requirements engineering based on business process models

Differently from the conventional requirements approach, the requirements engineering approach based on business process modelling includes a phase of explicit formalization of the business processes that will be supported or managed by the system to be developed. This phase aims at understanding the organizational environment in which the system will be used and providing rationale to justify the alignment of system requirements and organizational goals.

We have applied the "ARchitecture of Integrated Information Systems" (ARIS) [3] framework for the description of business processes and organizational structures. This framework is widely accepted within the business community [10], establishing as *a de facto standard* in several large organizations and was already adopted by the organization in which this study was conducted.

3.1. Business Process Modelling

The business process modelling technique employed starts with capturing a Value-Added Chain (VAC) which represents all macro-processes which are executed in order to achieve organizational strategies. They represent a business processes set whose level of aggregation is the highest possible. Figure 1 shows the macro-processes for Human Resources Management as elicited in the study reported here. The macro-process "Manage human resources" can be decomposed into the macro-process "Manage social security requests" which, in turn, is decomposed into: "Manage medical assessment", "Manage retirement" and "Manage pension". Only a small part of this VAC is shown here.

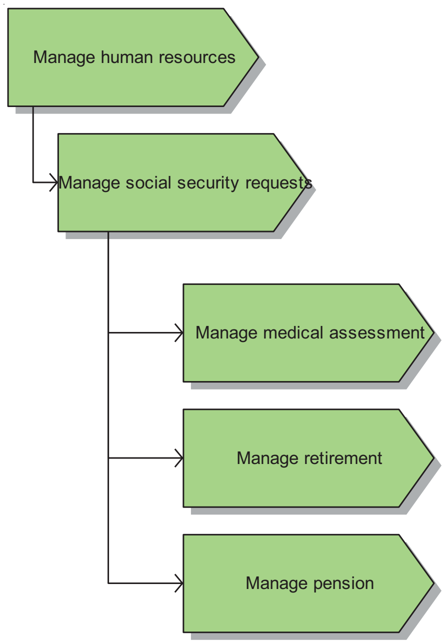

Figure 1. Value-Added Chain of Human Resources Management

Since macro-processes represent a bundle of business processes, the refinement of macro-processes produces chains of processes that operationalize organizational procedures. A business process is a systematic sequence of related actions which produce measureable results by consuming inputs of varied nature [8].

Figure 2 represents a fragment of the refinement of the business process "Manage medical assessment" (in an EPC (Event-driven Process Chain) diagram [3]. In these diagrams, actors that perform activities are symbolized by ellipses and may be organized in swimlanes. The activities executed by each actor are placed in his/her respective swimlane and are symbolized by rectangles while events are represented by hexagons. There are also logical operators which determine the flow of execution. For instance, in Figure 2, after the activity "Notify HR about schedule for medical assessment" is executed, the flow of control must follow in both branches to execute the activities "Notify employer about date of medical assessment" and "Confirm schedule for medical assessment". After the execution of the activity "Confirm schedule for medical assessment", the flow control can follow only one of the branches associated with either the event "Rescheduling is required" or the event "Schedule is confirmed". Please note that only a small fragment of this particular EPC is shown here due to space restrictions (the EPC elicited in the study has over 50 activities and involves several other actors).

When an activity in a business processes can be described as a single action without further decomposition, Function Allocation Diagrams (FADs) are created for each activity. In an FAD, one can assign resources to the execution of activities, revealing the organizational units where the activities occur, their executors, the systems which support them, the incoming and outgoing documents and information, business rules, business requirements and risks associated with them.

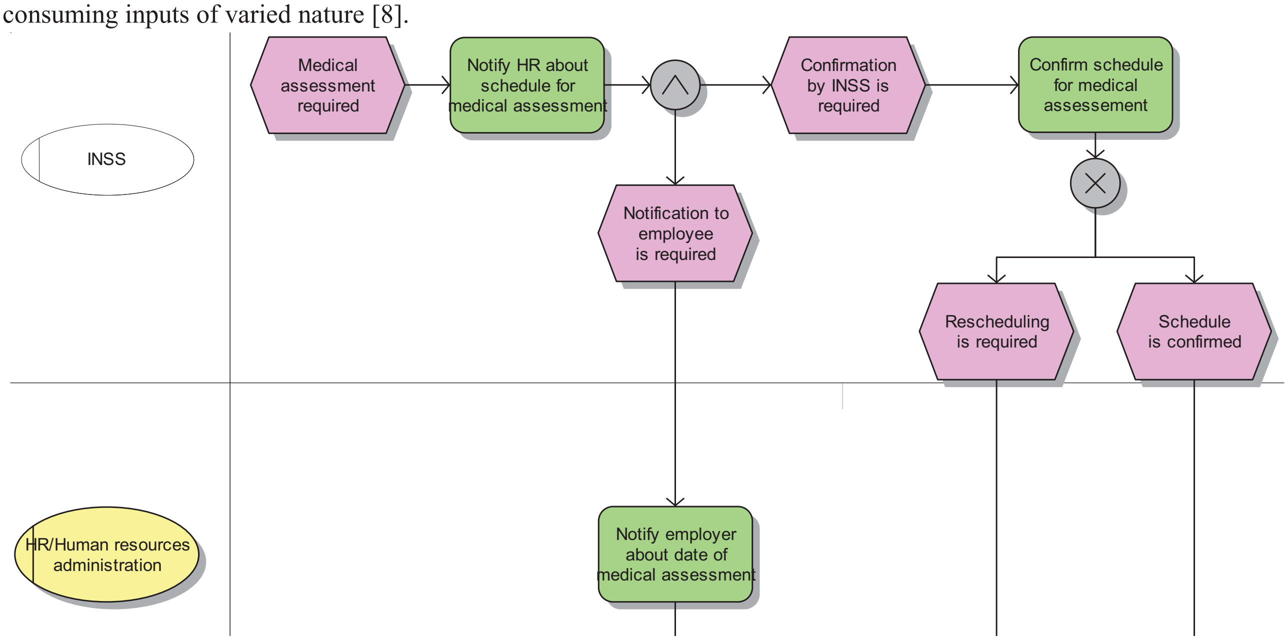

Figure 2 A fragment of the "Manage medical assessment" process model

Figure 3 shows the FAD for the activity "Schedule medical assessment" revealing the system used for registering medical assessment schedules ("Social Security requests management system"). It also reveals that the activity produces information related to the confirmation of scheduled medical assessment ("Confirmation of scheduled medical assessment"). Further, the diagram reveals a business rule that states that all medical assessment schedules can only take place after the fifth day of each month ("Period for scheduling medical assessment"). The information in an FAD will be used later during business requirements elicitation.

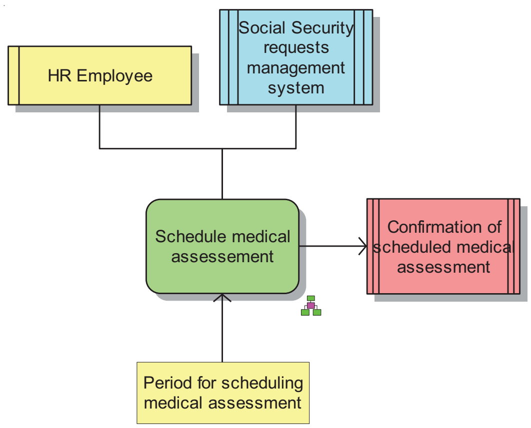

Figure 3. The "Schedule medical assessment" Function-Allocation Diagram

3.2. Deriving system requirements from business process models

After the business processes modelling phase, business process models are used as a starting point to derive system requirements.

We distinguish three sets of activities in a business process model: The first set consists of the activities which are not passive to automation, such as those operational activities executed by humans. The second set consists of those activities which can be supported by systems, while the third set refers to those activities fully automated, i.e., activities which can be executed by systems without any human intervention.

The distribution of activities into these three sets is performed jointly by the analyst and the customer [4] and must take into account a set of factors such as organization policies, technological constraints, safety constraints, among others. From this point of view, the business process models provide a basis for discovering services to be provided by the systems to be. Further, the models describe how the various pieces of information and resources must be manipulated by the systems supporting the business process. Since business processes models are abstractions of actions executed by some actors in an organization independently of which systems are used, given a business process model, it is possible to conceive a multiplicity of systems which can be used for supporting these processes (depending on the role that systems play during the execution of these activities).

Figure 4 represents schematically the relation between a business process model and a spectrum of possible system requirements which can support the business process (each requirements set corresponds to a different level of automation of the activities of this process). In this figure, the business process model is considered as a starting point in requirements engineering which involves the system stakeholders. From the same business processes model it is possible to obtain a requirements set R_1 to a system S_1 which is going to support it, or a requirements set R_2 to a system S_2 which exhibits a higher level of process automation, and so on. The choice of a particular requirements set is an explicit design step in which a great number of issues must be taken into account. These issues include safety (the automation of some critical activities which involves risks to those ones who execute it can be advantageous by avoiding occasional economical damages and human life loss), development costs, process execution costs, etc.

We use the term "level of automation" considering both the activities that are fully executed by the system and the activities whose execution is only supported by the system (i.e., activities in which the system is used as a tool by human actors).

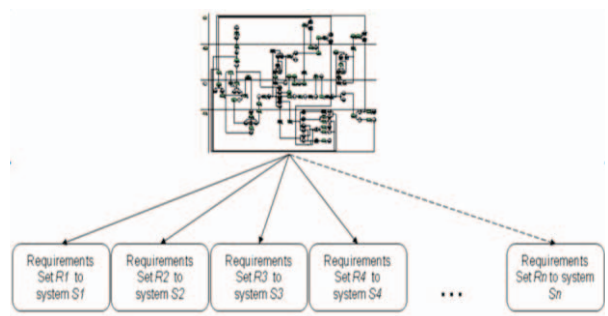

Figure 4. Requirements sets obtained from a business processes model

As an example of what has been mentioned above, Figure 5 shows the same business processes model subject to different levels of automation (the activities subject to automation are surrounded with ellipses). Figure 5(a) illustrates low level of automation: in this case, only the occurrence of a few events is recorded in the system. Figure 5(b) illustrates a higher level of automation: in this case, a greater number of activities and events are supported by a system. Figure 5(c) illustrates the case in which other functionality is inserted into the system, providing the capacity to automate an even greater number of activities and events. Eventually, in Figure 5

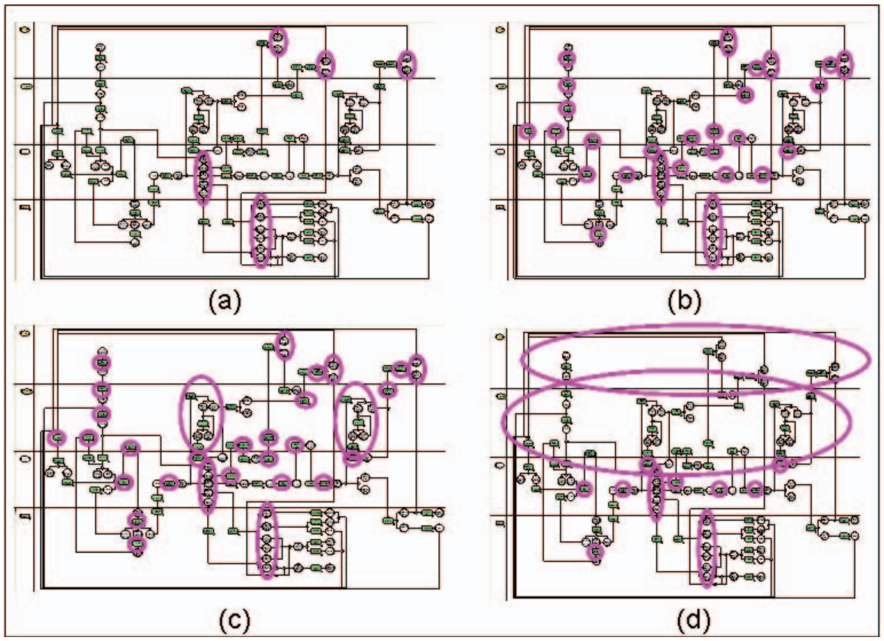

Figure 5. Systems generated from business processes model with different levels of automation

(d), automation has increased to such a level that the system may substitute some of the actors.

After distributing the activities in business process in the three automation categories, the system design phase can start. The next section aims at briefly illustrating how to derive different system designs from different requirements sets.

3.3. Design of a process-oriented system

Following requirements engineering, the design phase starts with the conception of the process-oriented system.

At this point, there is a transition between conceptual models (related with the process at a high abstraction level) to logical models (related with system design)[2]. While business process models are concerned with capturing causal and temporal relations, process-oriented models address aspects related with implementation (such as activities in which the actors insert information in a system, or activities in which there is interaction between the system at hand and other transactional systems).

As a starting point for designing the system, the first requirement we have identified is the need to control the business process from the point of view of a particular actor (an employee of the Human Resources Department). So, the system only interacts with this actor or with another transactional system.

The first system design may be derived from the requirements set shown in Figure 5(a). This design includes activities for inserting the occurrence of events and decisions by the employee of the HR department (this employee tracks all the events which represent the possible decisions of each actor after the execution of a medical assessment). Since the insertion of data by the human actor determines the flow of control of the system, the first alternative design does not provide control and notification mechanisms. This first alternative is the one that is most comparable with the requirements as obtained in the "conventional" approach. A second system design corresponds to the requirements set shown in Figure 5(b). The requirement of controlling the business process from the perspective of one human actor is also applied here. Thus, it has been necessary to insert artificial activities executed by the workflow system to verify whether the human actors who interact with HR have performed their respective activities. Besides, it has been added activities for notifying these actors about the occurrence of events.

4. Discussion

The two approaches were applied in the requirements elicitation phase for a system responsible for managing a business process in the Human Resources (HR) department of a large company in the energy business. We consider here: (i) a requirements specification previously derived (by an independent team) through the "conventional" approach described in section 2, and (ii) a requirements specification based on the business-process modelling approach presented in section 3. The requirements specification is omitted for space restrictions

[2] In our approach, business process models are the conceptual models, whereas system designs are process-oriented systems designs (workflow models).

and we report here the conclusions obtained when comparing the two approaches.

We must emphasize we do not have the ambition of generalizing the results based on this discussion, although we believe that the study here reported can justify this qualitative discussion. This comparison is descriptive and, certainly, should not be generalized without further validation. Moreover, we are aware that requirements engineering is highly dependent on the particular previous experiences of analysts [1]. In our study, the conventional approach and the BPM RE approach has been conducted by two independent teams.

Completeness. In the conventional approach, the requirements set did not cover all actors' actions, e.g., there were requirements missing with respect to functionalities for inserting and manipulating the decisions of a whole organizational unit called Medical Assistance (MA). During the business process modelling phase, we have detected the absence of these requirements and the need of inserting the MA's decisions in system. This can be regarded as indispensable for the business once the MA's decisions support the employer's decision with respect to a particular employee request. In the conventional approach, the interviews have focused on capturing the sequence of system actions, the data that actors must feed into the system, and the screens presented in the system's interface. This was done purely from the perspective of actors that directly interact with the system. Therefore, the requirements elicitation concentrated in particular actor-system interactions, showing certain deficiencies in identifying situations which involve other actors that indirectly influence the system behaviour.

Correctness. When the INSS performs a medical assessment, it must issue a result to serve as a basis for further decisions of the employer and the employee with respect to a particular social security request. In the conventional approach, when identifying the decisions that may follow medical assessment, the analyst has incorrectly identified a decision regarding an extension request by an employee. However, through business process modelling, we have identified that this decision was only possible for pensions that had been granted previously. This is an example which reveals the importance of business process models in clarifying the context of activity execution.

Consistency. During business process modelling, the meeting between all business process executors allowed a broad view of the system context, in which mutual divergences were identified. While the same business process executors had also participated in the meetings organized for the application of the conventional requirements engineering approach, the focus of those meetings was the description of the isolated interaction between types of actors and the system, thus not requiring reflection on the organizational *modus operandus* as a whole. The emphasis on the description of the flow of activities has enabled us to identify inconsistencies which would not be easily noticed otherwise.

Contextualization to customer and analyst. In the application of the conventional approach in the study, problem contextualization was not properly formalized. In the business process modelling approach, process contextualization was part of the method, i.e., the purpose of system construction and the activities which would be automated were properly formalized. We believe this is the main benefit of this approach in the scope of the study conducted.

Business process modelling has enabled customers to provide a high amount of details which had not been identified in the conventional approach. As customers provided more coherent and detailed descriptions, analysts could better formulate requirements for the system. In fact, both the customers and analysts could concentrate in activities which must be executed ("what" – in problem domain) and not in the way how they have to be executed ("how" – in solution domain).

Diversity in requirements set. The analysis of the business process model led us to the conclusion that it is possible to find a diverse number of systems which support the considered business process. The conventional approach adopts a requirements elicitation process whose requirements are obtained in an implicit way, delegating to the analyst the responsibility of choosing a system which in his/her point of view fits the problem at hand. However, the customer is unaware of the possible spectrum of systems that can be defined. In the business process modelling approach, deciding which system fits customer's needs better is an explicit step in the design process. Therefore, given a business process model, process executors can choose which system is better aligned to their aspirations, needs and costs. In the conventional requirements engineering technique, the analysts have proposed a system based on a low level of automation because they believed the business process at hand was simple. However, business process modelling showed us it was not the case, enabling us to conceive systems potentially better aligned to the customer's reality.

Traceability. In the conventional approach, after the requirements elicitation phase, system behaviour is often visualized by constructing system prototypes. In the business process modelling approach, this can be applied after diagrams have been presented to the user. This allows the identification of errors and points of improvement in documentation of the problem domain before prototype construction. More importantly, it is possible for an analyst to relate system requirements and business rationale. This way, it is possible to achieve traceability [5][6] in the requirements identified with business process models. Further, when there is a change in the business, it is possible to locate the functionality set related with this change, through the mapping previously established.

5. Related work

There are several works which explore business process modelling as a useful tool for system design. For example, the work presented in [7] establishes a direct connection between BPMN models (Business Process diagrams) and BPEL definitions. In order to achieve this, Business Process Diagrams (BPD) components (BPD are made up by a core subset of the BPMN elements) are mapped incrementally to suitable "BPEL blocks" by an integrated set of techniques. This mapping is done based on three different categories of components (i.e. patterns in BPMN models) and for each of these categories, the corresponding translation approach is applied. The final product is a readable BPEL code. The method assumes that BPMN models are entirely classified according to patterns and subsequently transformed into BPEL blocks, not considering the different roles a system may play in the execution activities. The system derived through this approach satisfies the set of requirements at the maximum level of automation in our approach. Since the approach focus on system design, requirements engineering is not addressed explicitly.

A proposal which takes business process modelling into account for requirements specification is presented in [8]. In this work, the construction of a business processes model has the purpose of providing maturity to customers in describing the system, and the business process model is not directly used for the system requirements derivation.

The work presented in [9] describes an approach to derive requirements from organizational models. The organizational models are captured as goal models in order to express business strategy. These goal models are extracted from business processes modelled in BPMN. The leafs of the goal tree are labelled according to the role that the system will have on the process (at this moment, the activities are going to be classified according to human intervention, i.e., if the system just supports the human action, if system fully automate the activity or if human perform the activity without no system). Finally, the use cases are derived from the goal tree by selecting the leaves that represent activities which are supported by systems. These leafs are the use cases. Although there are the three types of activities in the goal tree, just those ones that are supported by systems will be used for deriving use cases.

In [11], the paper proposes a comprehensive approach for identification and analysis of business services and its correspondent supporting software services. The approach is subdivided into two main parts (namely, derivation of business services and derivation of software services). Similarly to our approach, the authors have proposed the classification of process steps into those that can be executed automatically, semi-automatically, and those that must be executed manually. In their approach, this is part of what they call process-to-application mapping, in which existing processes and applications are analyzed to identify opportunities for service enablement. While they focus on identifying candidate services for existing applications in process-to-application mapping, we focus on the derivation of requirements for a process-oriented application that is yet to be developed. We believe that automated process activities identified using our approach could serve as candidate task services. In this sense, our approach could be considered a "top-down approach" for "task service" identification (adopting the terminology used in [11]).

Differently from previous approaches, our work reports a case study in which the requirements set that is identified depends on the level of automation chosen for the business process. Given a certain business process, it is possible to derive multiple system requirements sets, depending on the objectives that justify the construction of the system.

6. Conclusions

The study reported in this paper provides some evidence for the claim that requirements engineering can profit from the availability of business process models. Business process modelling complements conventional practices in requirements engineering, by facilitating problem comprehension, helping customers to acquire maturity about their own business processes and revealing how the system relates with overall organizational activities. When contrasted with a "conventional" requirements engineering technique (as characterized in section 2), the approach based on business process modelling has led to a more complete, correct and traceable requirements specification.

While the conventional approach leads to a specific requirements set for a system which supports an arbitrary set of activities of a process, the requirements engineering approach proposed in this paper supports the generation of multiple requirements sets with different levels of automation. The choice of which requirements set will be used is an explicit step in the process and depends on the purposes of the systems which automate the business processes at hand, taking into account factors such as organizational policies, legal constraints and technological constraints. Thus, this choice can be guided or constrained by non-functional requirements or system quality attributes which are not captured in the business process model. As a future work, we aim at providing methodological heuristics to justify the choice for a particular set of requirements, using non-functional requirements as guidelines.

In order to enable a model-driven approach for the construction of process-oriented systems, we intend to investigate the systematic derivation of system designs from business process models by allowing a designer to parameterize a source business process model to specify the desired level of automation for the various activities.

We should stress out that business process modelling does not depend on a system development lifecycle and can be conducted independently to develop a reference enterprise architecture model regardless of a particular system need. When it becomes necessary to develop a system to support part of a business process modelled in this architecture, one can return to the customer to elicit only information that is related to the detailed design of the system. Information on the business process model can be directly obtained from the reference enterprise architecture model.

We have assumed that business process elicitation leads to a stable business process model that is not altered by the introduction of the process-oriented system. Since this is not the general case, further investigation is necessary to accommodate business process changes in our approach.

The explicit step of choosing the level of automation also allows one to consider the trade-off between the level of automation of a business process and the volatility of system requirements. If a system supports a large number of activities of a volatile business process, system requirements will be subjected to frequent change. We intend to investigate the relation between the level of automation of a business process and the volatility of system requirements. We believe that abstraction through aggregation of activities in a given business process that is considered unstable can produce business processes that are more stable for automation, leading to more stable process-oriented systems.

The availability of business process models allows the dynamic behaviour of the organization to be understood by people who have not participated in the requirements elicitation activities. We believe that this may reduce the effort to introduce new members in a development team, suppressing or significantly shortening the problem contextualization phase for a specific system context. In this case, business process models are used as knowledge management tools, facilitating organizational knowledge sharing. This is, at this moment, a subjective claim that is still object for further investigation.

Acknowledgements

This work has been supported by CNPq (Brazilian National Council for Scientific and Technological Development), FAPES (Fundação de Apoio à Ciência e Tecnologia do Espírito Santo) in the context of the INFRA-MODELA project, and FACITEC (Fundo de Apoio à Ciência e Tecnologia do Município de Vitória) in the scope of the MODELA project.

References

[1] M. G. Christel, K.C. Kang, *Issues in requirements elicitation,* Technical Report CMU/SEI-92-TR-012 ESC-TR-92-012, Carnegie Mellon University, Pittsburgh, EUA. 1992.

[2] A. Pouloudi, “Aspects of the Stakeholder Concept and their Implications for Information Systems Development”, *Proceedings of the 32nd Annual Hawaii International Conference on Systems Sciences (HICSS-32)*, Hawaii, IEEE Computer Society Press, 1999.

[3] A. W. Scheer, *Aris – Business Process Modelling*, Springer-Verlag, New York, 2000.

[4] D. M. Knight, *Elicitação de Requisitos de Software a partir do Modelo de Negócio*, Master Thesis UFRJ/NCE, Rio de Janeiro, 2004.

[5] O. Gotel, A. Finkelstein, “An Analysis of the Requirements Traceability Problem”, *Proc. First International Conference of Requirements Engineering,* 1994, pp. 94-101.

[6] O. Gotel, *Contribution Structures for Requirements Traceability*. PhD Thesis, Imperial College, Department of Computing, United Kingdom, 1995.

[7] C. Ouyang, M. Dumas, A.H. ter Hofstede, W.M. van der Aalst, “From BPMN Process Models to BPEL Web Services”, *Proceedings of the IEEE international Conference on Web Services (ICWS 2006)*. IEEE Computer Society, Washington, DC, 2006, pp. 285-292.

[8] A. Sharp, P. McDermott, “Workflow Modelling: Tools for Process Improvement and Application Development”, Artech House Computing, 2001.

[9] J. González, J. Díaz, “Business process-driven requirements engineering: a goal-based approach”. In Proc. of 8th Workshop on Business Process Modelling, Development and Support (BPMDS'07), 2007.

[10] D. Neiger, L. Churilov, “Goal-Oriented Business Process Modelling with EPCs and Value-Focused Thinking”, Business Process Management, LNCS, vol. 3080, Springer, Berlin/Heidelberg, 2004, pp. 98-115.

[11] T. Kohlborn, A. Korthaus, T.Chan, M. Rosemann, “Identification and Analysis of Business and Software Services—A Consolidated Approach,” *IEEE Transactions on Services Computing*, vol. 2, no. 1, 2009, pp. 50-64.