KEEPING THE RATIONALE OF IS REQUIREMENTS USING ORGANIZATIONAL BUSINESS MODELS

Juliana Jansen Ferreira, Victor Manaia Chaves, Renata Mendes de Araujo
and Fernanda Araujo Baião
NP2Tec – Research and Practice Group in Information Technology
Federal University of the State of Rio de Janeiro (UNIRIO)
Av. Pasteur, 458, Urca, 22290-240, Rio de Janeiro, RJ, Brazil

Keywords: Information system requirements, Requirements elicitation, Business processes modelling, IT organizational architecture.

Abstract: This paper proposes an approach for identifying, documenting and keeping the rationale of information systems requirements starting from the organizational business model. The approach comprises a method and the implementation of a supporting tool. The paper also discusses the results of preliminary case studies with this approach.

1 INTRODUCTION

One of the great challenges of requirements elicitation is ensuring that system requirements are aligned with organizational business needs. Requirements that represent just a single user perspective, which are partially analyzed and do not consider the environment where the system will be used, may lead to systems which will not address its users and organizational expectations.

Aligning system requirements with the business needs of an organization requires understanding the organizational context. In order to understand and represent how organizations work, the Business Process Modelling (BPM) area has defined a set of concepts, models and techniques with the purpose of drawing up organizational business models (PROFORMA, 2000) (PROFORMA, 1998) (Sharp and Mcdermott, 2001).

Previous work of our research group (MacKnight *et al.*, 2005) proposed a systematic approach for using business models, especially business process models, for the discovery or elicitation of system requirements. In that work, a method was proposed to guide the software development team, in collaboration with the organization employees, to understand the business model, identify business needs and define system requirements aligned to the business needs. Case studies, along with the continuous use of the method in real projects in industry pointed to the need for a tool to support the method, thus providing automatic traceability and documentation of the information gathered during the method execution.

The present work discusses how the combination of the method and the tool helps organizations to keep the rationale on the context in which systems requirements have been identified.

The paper is structured as follows: Section 2 presents BPM objectives and concepts and how it has been used in IS requirements elicitation. Section 3 describes the method focusing on the information and rationale generated in each step. Section 4 presents the tool main functionalities. Section 5 concludes the paper, suggesting the opportunities for improving the proposed approach.

2 WHY BUSINESS PROCESS MODELING

Business Process Modelling (BPM) aims at building a set of combined views that allow a proper agreement on the business (Eriksson and Penker, 2000) (PROFORMA, 2000) (PROFORMA, 1998) (Sharp and Mcdermott, 2001) (BABOK, 2008). Through BPM, an organization domain can be represented through diverse perspectives (Figure 1), and the representation of a domain makes use of diverse models for each perspective. Business process models reflect the way an organization is structured in order to guarantee business rules and

Jansen Ferreira J., Manaia Chaves V., Mendes de Araujo R. and Araujo Baião F. (2009).
KEEPING THE RATIONALE OF IS REQUIREMENTS USING ORGANIZATIONAL BUSINESS MODELS.
In *Proceedings of the 11th International Conference on Enterprise Information Systems - Information Systems Analysis and Specification*, pages 292-297
DOI: 10.5220/0002153602920297

strategic goals achievement.

Figure 1: Business process models and perspectives.

There are many approaches for business modelling, and each of them defines different notations and languages for organizational business design. Independently of the approach used to build the business model, the following models can be considered by the business analyst to start requirements identification:

1) **Organizational Model.** stands for the organizational units, its roles and the relationships between them. For example, an IT department may be considered an organizational unit, and a programmer, a role. Besides that, an organizational model represents which roles belong to each unit, depicting an organizational hierarchy;

2) **Goal Model.** shows the business goals and the hierarchical relationships between them. Represents how a goal is divided into subgoals which can be achieved by business processes;

3) **Process Model.** represents all business processes and their hierarchical structure. It also represents the activities that comprise each business process;

4) **Activity Model.** represents the relationships between business processes activities and those responsible for each of them. It also shows which business objects (that may comprise any information storage) are handled in each activity and which events trigger or are triggered by the execution of each activity.

3 IDENTIFYING IS REQUIREMENTS FROM BUSINESS MODELS

There are several works in the literature considering business modelling as a useful starting point for system requirements elicitation (Eriksson and Penker, 2000) (PROFORMA, 2000) (Christel and Kang, 1992) (Rolland et al, 1998) (Marshall, 2008) (De la Vara et al., 2007) (Röhrig, 2003) (Demidörs et al., 2003)(Pavlovski and Zou, 2008). The common agreement among them is that the business modeling task is part of the system development process, which means one of the first activities for system development should be the representation of the business portion to be supported by the system. What is different in our approach is that it aims at using business models as an instrument for requirements discussion among stakeholders. It also registers rationale information for future use.

Some business analysis approaches, especially the ones concerning software development, consider the use of intentional business modelling such as i*(Yu, 1995). For example, the Tropos Framework proposes a software development methodology based on i* concepts to model early requirements (Mylopoulos, Castro and Tropos, 2000). Although the i* concepts embraces business concepts – such as actors, goals and activities – the use of intentional modelling notations such as i* has not been widely used in organizations as an alternative for business process modelling. It also lacks for supporting tools which could be enhanced in order to cope with rationale traceability.

The approach presented in this paper differs by considering graphical models as an instrument for brainstorming system requirements, and for keeping and generating different views/reports on requirements elicitation information. A tool was implemented on top of an integrated repository of process models. This provides for system analysts an integrated view of all information systems that support business activities, and of other activities that are indirectly related to the ones within the scope of the system request.

4 THE METHOD

The method is divided in two main phases: **understanding the problem** and **building the solution view.** The objective of the first phase is to discover the actual organizational needs in business processes. Those needs should be addressed by the system to guarantee that the right problem is being solved. In the solution view, the identified needs are analyzed to assess how they can be addressed.

The examples shown in this section illustrate a scenario of a radiation laboratory (RML) in a real organization in Brazil. RML is responsible for providing radiation measurement services and personnel protection for enterprises whose

employees work under exposure to different kinds of radiations, and therefore need to be constantly monitored. The monitoring process consists of *in vivo* bioanalysis, which directly analyzes human bodies to measure the levels of radiation. This process is supported by a set of systems and a request was made to develop an integrated system to support the monitoring process.

This section highlights the information generated by the method at each step execution, and why it is important for requirements elicitation memory and traceability.

4.1. Understanding the Problem

The first phase of the method comprises 3 main steps or activities:

Step 1: Identify the Context of the Request. The aim of this step is to establish the boundaries of the business area involved with the existing system development request. This means analyzing the business model, specifically the process models identifying which processes must be considered for further detail, concerning the request. For each identified process, it is also important to identify which activities will be supported by the new requested system/functionalities. The activity model guides the discussions, being useful not only as information basis but also as an instrument for discussion and knowledge build.

Step 2: Identify Business Needs. By analyzing the organizational model it is possible to notice which process participants can be considered as future system actors. These are candidates for interviews to understand their needs and possible difficulties encountered during process execution.

To discover the business needs it is necessary to find the difficulties in the process and look for their reasons. These reasons are the business needs which must be addressed so that the problems encountered are solved. It is important to note the relationship among each difficulty, its related business need, and the activity containing the difficulty (Table 1). Through this relation, we can trace the reasons why a business need is being considered by the system because there was a certain difficulty in a specific activity of the process.

Step 3: Identify Impacts of Business Needs. The aim of this step is to have an overview of the impact the identified business needs cause to the organization. These impacts will help define the priorities of business needs and also help to decide which of them will be addressed. The consequences of each business need are identified with the help of the activity model. Each business need can be assessed to discover its impacts to the activities and to the whole process. With the help of the goal model, the organizational goals and people affected were identified (Table 2).

Table 1: Relationship activities, difficulties and needs.

Activities	Difficulties	Business needs
Input personal information	Write personal information in a form and enter the same information in the patient catalogue system	System interface and performance should be efficient and let the technician enter needed information directly into the system
Input measures	Write the measures in a form and then enter again the measures into the measurement catalogue system	
	To associate the personal information of a worker to his/her measurement, in the two different systems, it is necessary to copy manually the workers ID in the patient catalogue into the measurement catalogue	Functionalities of each existing system should be supported by the new system

Table 2: Business needs impacts on the processes.

Business Needs	- System interface and performance should be efficient and let the technician enter necessary information directly into the system - Functionalities of each existing system should be absorbed by the new system - Create an enterprise catalogue and relate each worker to an enterprise
Consequences	- Delay in activities - Redundant work - Need for more employees - Chances of making mistakes
Goals	- Be approved by clients - Have competitive costs
People	- Technicians - Client
Priority	- High

4.2 Building the Solution View

The second phase of the method comprises 4 main steps or activities:

Step 4: Identify System Functionality and Restrictions. Each activity in the identified set should be analyzed concerning how it can be supported by the system to find out the system main functionalities and restrictions. Table 3 shows the main functionalities identified from the activities.

Step 5: Identify System Boundaries. In this step, the objective is to evaluate roles, other systems, storages and external actors which may have to exchange information with the system. The activity model is analyzed to find out if there are any

activities that the new system should support, which are actually supported by other systems. This can also be done in a simplified manner, as a glossary. Some BPM approaches use to build domain glossaries which definitions could be here reused.

It is interesting to analyze the process business needs to identify other functionalities which solve the problems found in the process. At this moment, non-functional requirements could also arise. Table 4 shows the result, relating each functionality and restriction found to the business need that originated it. This relationship is important as it documents the reason for the existence of the requirements.

Step 6: Identify System Boundaries. In this step, the objective is to evaluate roles, other systems, storages and external actors which may have to exchange information with the system. The activity model is analyzed to find out if there are any activities that the new system should support, which are actually supported by other systems. This can also be done in a simplified manner, as a glossary. Some BPM approaches use to build domain glossaries which definitions could be here reused.

Table 3: System main functionalities.

Activities	Functionalities
Input personal information	Manage personal information
Query personal information	
Input measurements	Manage measurements
Analyze readings	Manage readings
Enter readings	
Calculate level of radiation	Calculate level of radiation
	Manage radiation information
Generate report	Generate report

Table 4: System aspects identified from business needs.

Activities	Business needs	Functionalities and restrictions
Input personal information		Fast processing
		Simple interface
Generate report	Create an enterprise catalogue and relate each worker to an enterprise	Manage information about RML clients

Step 7: Identify the Impacts of the New System. It is important to assess the way the new system interferes in the organizational business so that those impacts are known and updated in the business model. This can be done, with the help of the activity model, by interactions with future users based on the main aspects defined for the system.

Step 8: Generate Software Requirements Document. Finally, software engineers should register in the software requirement document all relevant information about the system that resulted from previous steps.

The method was applied into three case studies in real organizations (MacKnight *et al.*, 2005). The case studies showed that the method led to a valuable system requirements list. It was also observed that the more detailed the business model is, the richer the requirements identification can be. However, these case studies suggested that the effort required to follow the method was great considering the level of documentation necessary to cope with each step. This issue pointed to the fact that the systematic use of the method in any organization should be supported by automated tools. The customization of BPM tools to cope with the requirements rationale documentation was one step forward.

5 THE TOOL

The tool support for the Mac Knight method was developed on top of the ARIS Business Architect (IDS Scheer, 2008) platform, due to its support for developing customized reports and adding user-defined objects and diagrams properties, as well as because of its popularity. The experience we had with the tool and the possibilities for further experimentation in real scenarios were also motivations to use it for the method support.

The ARIS tool supports several models or diagrams, each of them composed by a rich set of objects with specific semantics. The development of the necessary functionalities for supporting the method comprised: creating new symbols in ARIS symbols set, overloading existing objects, overloading model types, creating new relationships among objects and models, and creating filters, templates and report generation scripts.

For the first method phase (understanding the problem), extensions were created to help the identification of the request context witihin the process models. A new attribute – *system support* – was created and associated to each process model. The attribute indicates if the process activity is requested to be system supported.

After filling this attribute, users may execute a script that was developed for generating the "*Processes and goals related to client request" report*. The report lists and describes all objects with the *system support* attribute checked, and the goals related to them. This helps stakeholders in analyzing the context of the system support request within the process and business strategic goals.

Another report is provided, relating process with their executors, thus helping analysts in identifing possible participants in requirements discussion.

Four new objects - *difficulty*, *cause*, *effect* and *need* – were created, providing ways to register the rationale around a business requirement. New attributes were created for some of these objetcs: *causes*, for difficulties; *impact* and *priority*, for needs. The *difficulties report* and *needs report* were also developed.

Business needs that will not be addressed by the system may have the *system support* attribute unchecked, while other may be organized in order of priority by setting the *priority* attribute. A report was also created to list the needs that will be addressed and their priorities.

For the second phase (building the solution view), business needs can be associated to the *functional* and *non-functional* requirements. A report listing all requirements related to activities and needs is also provided.

To identify system boundaries (points where the system will change information with others business resources - users, others systems, and databases), a new diagram (*System Use Diagram)* was defined, as well as new objects, such as the object *user* to represent the users of the system and the attribute *permission* for each user. For each database, a conceptual data model could optionally be created. Different reports are defined: users and related privileges; data dictionary; and the relationship among business needs and the new system.

Impacts of the new system can be identified by checking each activity that will be automated trhough the functionality Change Management, avaiable in ARIS Business Architect. A report generated at the end of this step lists the changes that should be done in the process.

Finally, the tool allows the generation of the *Software Requirements Document.* It merges all previous reports, configuring a first draft of a system vision document. The *Requirements diagram* collects information about all system requirements elicited along time for an information system (Figure 2). *Needs* are represented in blue, *functional requirements* in yellow and *non functional requirements* in red, and *systems* in light blue. Through this map of requirements, it is possible to keep the traceability between needs and requirements (functional and non functional); and which systems implement each requirement.

The information provided by the method is organized into groups to separate different instances of models, according to current (as-is) and future (to-be) situations, concerning business needs and system requirements.

We used the support functionalities presented in this paper in a real project of a large Brazilian company, where business processes were modelled to discuss requirements for the acquisition of a new system for people control access, for corporative use. The main observations of this experience were: 1) the tool was a good facilitator for information gathering and association; 2) considering that the tool was designed to help IT experts, the developed scripts were of easy and intuitive use. They were helpful to consolidate the information on reports that were used on the analysis steps of the method; 3) the *requirements diagram* was a useful map of elicited information; 4) the association of the actor of the process activity and the system user, with his needs and functionalities, is useful information for the validation phase of the new system development; 5) the s*oftware requirements document* report that can be automatically generated presents all the information related to the elicitation, and only requires format adjustments after its generation.

All extensions that were developed on top of ARIS are structured according to the conceptual model illustrated in Figure 3. This model shows how the elements added by the tool implementation are related to existing elements in ARIS. This model shows how the elements added by the tool implementation are related to existing elements in ARIS.

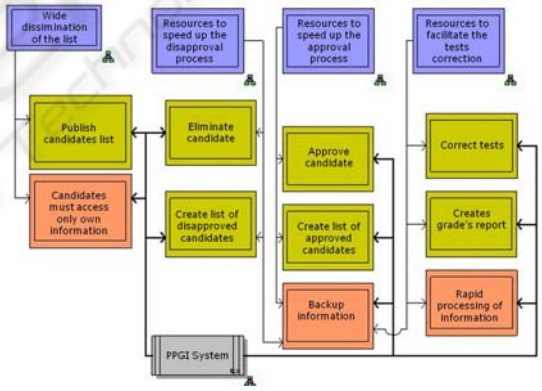

Figure 2: Requirements diagram.

6 CONCLUSIONS

Our research group - NP2Tec - conducted several consulting experiences in BPM in Brazilian organizations, which resulted in more than 200 modelled business processes. We observed that 100% of the business process modelling initiatives we conducted led organizations to think on system requirements even if it was not the initial intention.

Our proposed tool helps organizations to address the alignment of system requirements identification to business models, by providing automatic support

for keeping the rationale of these decisions. Additionally, the ability to keep alignment information from business to IT is a crucial step for organizations willing to build corporate IT architectures, as suggested by (Zachman, 1987) (Nikolaidou and Alexopoulou, 2008) (Dietz, 2004). The organizations learn how business process management can be helpful and continuously maintain a business process model, this model can be used as a high level infrastructure for system development and for maintaining the organizational information architecture.

Future work includes evolving the proposed solution to cope with non-functional requirements identification (Bittencourt and Araujo, 2008) (Röhrig, S., 2003) (Pavlovski and Zou, 2008) and to keep information about requirements change and management also aligned to organizational business processes evolution.

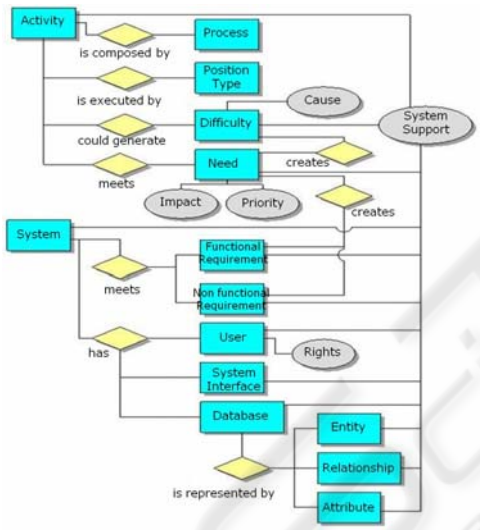

Figure 3: Conceptual model.

REFERENCES

Babok, 2008. *International Institute of Business Analysis – Business Analysis Body of Knowledge* – http://www.theiiba.org, accessed February, 2009.

Bittencourt, R. V., Araujo, R. M., 2008. Identifying IS quality expectations with the support of business models. In: *II Workshop on Business Process Modeling*, in conjunction with Webmedia 2008, Vitória (*in portuguese*).

Christel, M. G., Kang, K. C., 1992. Issues in Requirements Elicitation, Technical Report, Software Engineering Institute, CMU. CMU/SEI-92-TR-12 ESC-TR-92-012.

De la Vara, J., Alcolea, D., Diaz, J., 2007. Descomposición de árboles de metas a partir de modelos de procesos. *In: Workshop em Engenharia de Requisitos*, Toronto, 35-46. (*in spanish*)

Demidörs, O., Gencel, Ç., Tarhan, A., 2003. Utilizing Business Process Models for Requirements Elicitation. In: *Proc. 29th EUROMICRO*, IEEE CS Press, 409-412.

Dietz, 2004. Basic Notions regarding Business Processes and Supporting Information systems, CAISE Workshops (2).

Eriksson, H.-E., Penker, M., 2000. *Business Modeling with UML: Business Patterns at Work*. New York: Wiley Publishers.

IDS Scheer, 2008. ARIS Business Architect: The Web-Based Benchmark for Business Process Management. http://www.ids-scheer.com/en/Software/ARIS_Software/ARIS_Business_Architect/3731.html. accessed in February, 2009.

MacKnight, D., Araujo, R., Borges, M., 2005. A Systematic Approach for Identifying System Requirements from the Organization's Business Model. In: *II Simpósio Brasileiro de Sistemas de Informação*, Florianópolis, Brasil.

Marshall, C., 2008. *Enterprise Modeling with UML: designing successful software through business analysis*. Addison-Wesley.

Mylopoulos, J., Castro, J. 2000. Tropos: A Framework for Requirements-Driven Software Development. In: S01vberg, Brinkkemper, and Lindencrona (eds.) *Information Systems Engineering: State of the Art and Research Themes*, Springer Verlag.

Nikolaidou, M. Alexopoulou, N., 2008. Enterprise Information System Engineering: A Model-Based Approach based on the Zachmann Framework, HICSS.

Pavlovski, C.J., Zou, J., 2008. Non-Functional Requirements in Business Process Modeling. In: *Proceedings 5th Ásia-Pacific Conference on Conceptual Modeling*, Wollongong, Austrália.

PROFORMA, 2000. Enterprise Application Modeling - Vision and strategy for the ongoing development of ProVision Workbench. Proforma Technical White Paper by Proforma Corporation.

PROFORMA, 1998. Integrating Business Processes, Workflows, and Object Models via Use Cases. Technical White Paper, Proforma Corporation.

Rolland, C., Achour, C., Cauvet, C., Ralyt, J., Sutcliffe, A., Maiden, N. A. M., Jarke, M., Haumer, P., Pohl, K., Dubois, P. Heymans, P., 1998. A Proposal for a Scenario Classification Framework, *Requirements Engineering Journal*, Vol 3, No. 1.

Röhrig, S., 2003. *Using Process Models to Analyze IT Security Requirements*. DSc Thesis, Zurich.

Sharp, A., Mcdermott, P., 2001. *Workflow Modeling: Tools for Process Improvement and Application Development*. Norwood: Artech House.

Yu, E., 1995. Models for supporting the redesign of Organizational Work. In: *Proc. of Conf. On Organizational Computing Systems*, California, 225-236.

Zachman, John A., 1987. *A framework for information systems architecture*, IBM Systems Journal, v.26 n.3, 276-292.