Heravizadeh, Mitra and Mendling, Jan and Rosemann, Michael (2008) Root Cause Analysis in Business Processes.

Root Cause Analysis in Business Processes

Mitra Heravizadeh, Jan Mendling, and Michael Rosemann

Queensland University of Technology
126 Margaret Street, Brisbane QLD 4000, Australia
mitra.heravizadeh@gmail.com,j.mendling@qut.edu.au,m.rosemann@qut.edu.au

Abstract. Conceptual modeling is an important tool for understanding and revealing weaknesses of business processes. Yet, the current practice in reengineering projects often considers simply the as-is control flow and uses the respective model barely as a reference for brain-storming about improvement opportunities. This approach heavily relies on the intuition of the participants and misses a clear description of steps to identify root causes of problems. In contrast to that, this paper introduces a systematic methodology to detect and document the quality dimension of a business process. It builds on the definition of softgoals for each process activity, of correlations between softgoals, and metrics to measure the occurrence of quality issues. In this regard our contribution is a foundation of root-cause analysis in business process modeling, and a conceptual integration of goal-based and activity-based approaches to capturing processes.

1 Introduction

Large organizations run complex business processes that hardly any staff member can oversee in their whole complexity. As a consequence of this the fundamental reasons for weak operational decision making often remain undetected until serious losses are caused by it. One such example is German IKB bank that lost about $3.3 billion since the beginning of the 2007 U.S. subprime mortgage crisis due to a wrong assessment of risk. Another case is the new main station in Berlin where the project costs were planned too optimistically in 1997 and almost doubled from 700 million to 1.2 billion euro in the course of construction. While such problems might require a substantial change in the operational processes, i.e. cost estimation and the risk policy in these cases, it is not directly obvious how the actual root causes of these problems can be identified in a systematic way.

Root cause analysis has been conducted in industry using a variety of techniques (see [1]). While flow charts have been identified as a potential tool to facilitate root cause analysis, the current practice mainly builds on brain-storming and semi-formal techniques [2]. The fact that large organizations usually possess extensive documentations of their business processes in terms of models has not be exploited so far. On the other hand, in business process redesign the traditional focus is still very much on the design of models that reflect current practices (so-called as-is models) followed by the design of an improved (to-be) process model [3]. In this context, the white space between as-is and to-be is poorly supported by popular process modeling tools. Indeed, combining business process models and root cause analysis bears the potential of revealing problems in an organization in a more systematic way relying less on the intuition of those being involved in the analysis.

Against this background we present a novel technique to conduct root cause analysis based on business process models. In this regard, our contribution is a mechanism to extend process modeling languages–we use Event-driven Process Chains (EPCs) for illustration purposes–with concepts from requirements engineering. In particular, we reuse established tools from software measurement such as quality models, softgoal elicitation, the goal-question-metric approach, and goal correlation as ingredients of a systematic approach to conduct root cause analysis. We refer to this technique as Process Root Cause Analysis (PRCA).

The remainder of the paper is structured as follows. In Section 2 introduces the case of a software project process that is used by Software House Inc. (SH) for responding to tenders. SH recently won a bid because of an unrealistically low estimation of the project costs. In order to avoid losses in the future SH needs to find out why the estimation went wrong. The starting point is the process model of the process. In Section 3 we introduce our approach to root cause analysis. In particular, we first introduce the metamodel of our technique and then discuss each of the steps of populating it. For illustration we use the example of the SH request for proposal process. After that, Section 4 compares our technique to different areas of related work. Finally, Section 5 concludes the paper.

2 Why did the project estimation fail?

In this section we introduce the *Request for Proposal (RFP)* process of SH. Recently, SH has won a major project using this RFP process. After three months SH management have realized that the deliverables of the project cannot be completed at the estimated cost. As a result the project is likely to become a loss. In order to prevent such losses in the future, SH investigates the root causes for the flawed calculation.

In a first step SH turns to the model of the process. Figure 1 describes the RFP process as an EPC business process model. In this paper we use EPCs for illustration purposes. Other languages like BPMN or Petri nets could be used instead. An EPC basically captures the control flow of a business process. Rounded boxes in an EPC define so-called functions, i.e. the activities of the process like *Review Requirements* or *Estimate*. Functions are executed by organizational entities (ovals on the left-hand side), and they take inputs and create output (right-hand side). So-called events (hexagons) describe the preconditions and postconditions of functions. A process has at least one start event and one end event, in this case *RFP is received* and *Response is sent*. Routing conditions are captured by so-called connectors (circles). In this process there is one split connector and one join connector. Both are XOR-type connectors: the split de-

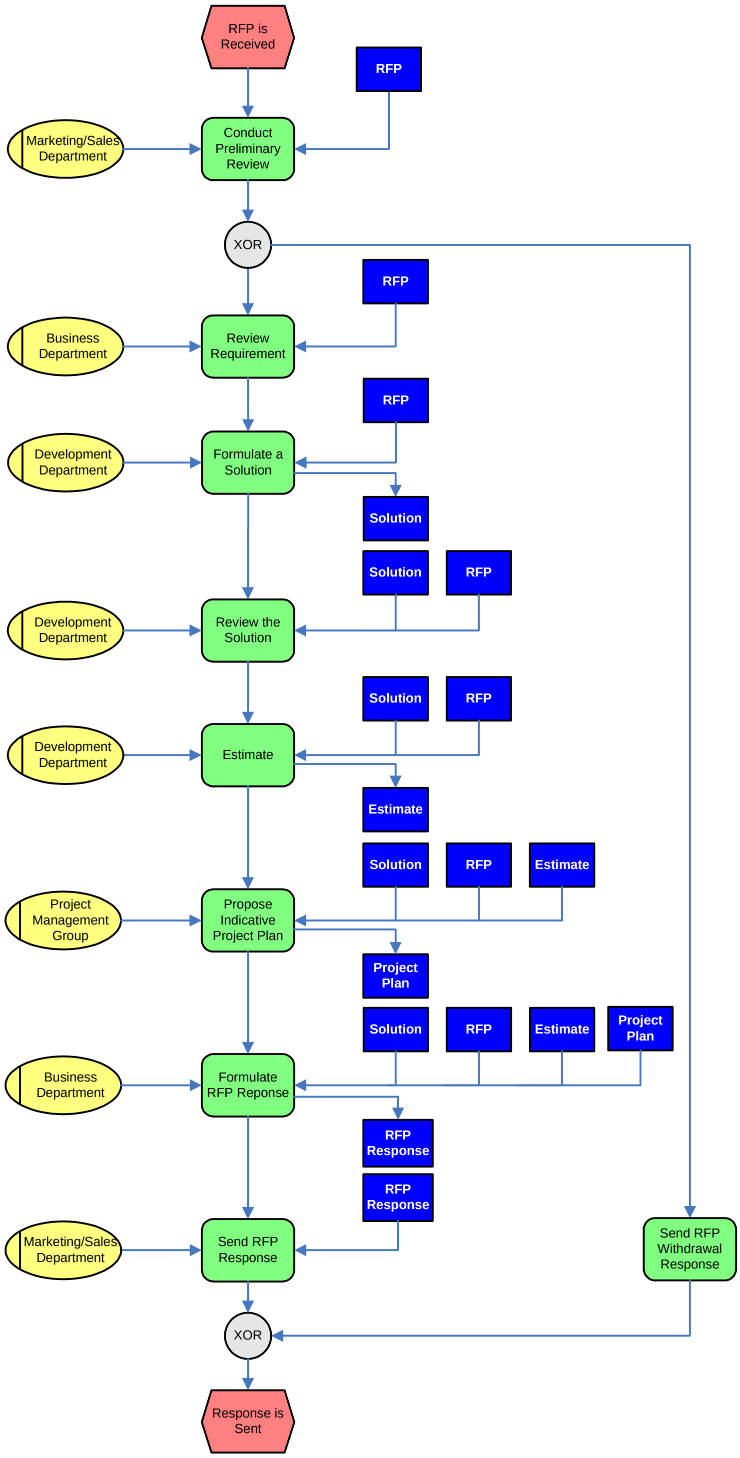

Fig. 1. Request for proposals process of the software house

scribes a decision point and the join merges the two alternative branches. For formal details of EPCs refer to [4].

The process commences with the event that an RFP is sent from a client company and this *RFP is received* by SH. The RFP documents the high level requirements of the system to be developed and typically asks SH to respond by providing a proposed solution, an estimate of the time and cost to complete the work and an indicative project plan. As a first step, the marketing/sales department reviews the RFP to determine the parameters of the response (*Conduct Preliminary Review*). This review provides first insight regarding the potential profit from the project and strategic interest of SH in the client. Depending on this information a decision is made (XOR-split): if the business is likely to be unattractive SH will *Send RFP Withdrawal Response*, otherwise further steps to complete the RFP response are undertaken.

This series of steps starts with a review of the requirements by the business department who forwards the RFP to the development department. Development formulates a technical solution based on the RFP. This solution is reviewed before it is used as a base for estimation of cost and effort. These estimates and the solution are then considered for proposing an indicative project plan worked out by the project management team. In particular, the project schedule, project cost and the project team is outlined. This plan along with the estimation and the solution concept is input for the business department to *formulate RFP response*. The marketing/sales department finally sends this RFP response to the potential client.

The RFP model offers SH management an initial overview of the process. Yet, it does not directly reveal problems with its execution. For tracing back the root causes of the prospective project loss, SH wants to utilize a systematic approach. In particular, they are convinced that techniques from requirements engineering would be helpful.

3 How to find root causes in business processes?

Root cause analysis is a problem solving technique in a variety of quality-centered management approaches such as Six Sigma. The main assumption is that an issue can only be solved by addressing the underlying cause for the problem. Conceptually, root cause analysis is grounded in the principle of double loop learning as part of organizational learning [5]. Double loop learning goes beyond the detection and correction of errors and concentrates on the related policies, systems, norms, procedures, context factors, etc. as the causes of the error. Several approaches to root cause analysis have been proposed, among others so-called Ishikawa diagrams, or fishbone diagrams [6]. These diagrams capture potential causes of a problem and are typically populated in brain-storming sessions. In this section, we introduce our Process Root Cause Analysis approach (PRCA) that combines ideas from Ishikawa diagrams with concepts from process modeling and requirements engineering. Our tailoring of this approach can be regarded as an instance of situational method engineering [7] following an assembly approach [8]. In Section 3.1 we define a metamodel for the problem domain of root-cause analysis in business processes, and in Section 3.2 we describe the process of how to populate the metamodel for one particular analysis case.

3.1 PRCA Metamodel

This section describes the metamodel of the PRCA approach. As we have outlined before, PRCA builds on control flow based business process modeling. Accordingly, the upper left part of Figure 2 captures the essential elements of an EPC, i.e. control flow elements including functions, events, and connectors which are linked by control flow arcs. Furthermore, each function can be described regarding its input and output as well as its resource requirements. This part of the metamodel is classical process modeling and can easily replaced by respective elements of other modeling languages such as BPMN or high-level Petri nets.

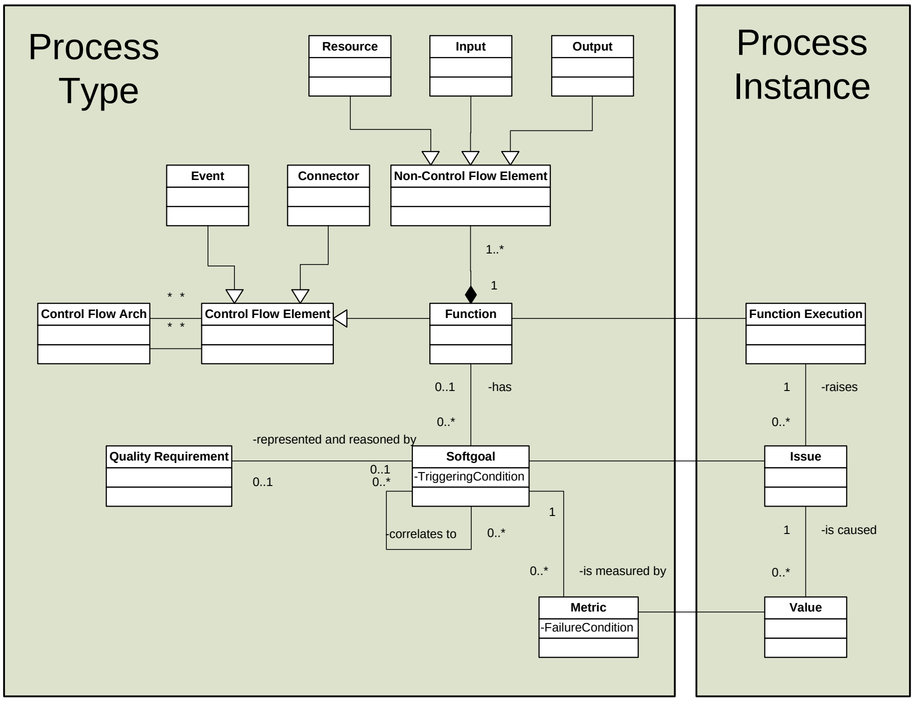

Fig. 2. PRCA Metamodel as a UML class diagram

The PRCA metamodel introduces additional concepts to capture those entities relevant to root cause analysis in the bottom part on the left-hand side and on the right-hand side of Figure 2. First of all, it is important to distinguish process type level and process instance level. *Issues* are raised on the process instance level, i.e. for a particular case like the RFP case of the software house SH that will likely cause a loss. Issues relate to the execution of a function in the process (*Function Execution*) and the way the function is conducted (described by *values* related to a *metric*). In order to classify issues appropriately, the PRCA approach builds on identifying softgoals for a function on the process type level. *Softgoals* in this context refer to non-functional requirements of the function similar to their application in early stages of requirement engineering process [9–11]. An example of a softgoal of the *Estimate* function of the RFP process is *ensure accuracy of estimate.* For the identification of softgoals we use a set of generic *quality requirements* such as e.g. the ISO 9126 [12] that lists accuracy as one particular quality dimension. The relevance of a softgoal is bound to a triggering condition that specified e.g. that accuracy matters only for projects worth more than $1,000. The achievement of a softgoal is made measurable by relating it to a *metric* based on the goal-question-metric approach [13]. The link to the process instance level is provided by the failure condition of the metric: if the *value* of the metric in a particular process instance meets the failure condition, this signals the occurrence of an issue related to the execution of a singular function. In order to trace issues back to root causes, we identify *correlations* between softgoals (cf. [14]). In our example, the accuracy of the solution correlates with the accuracy of the estimate. The following section describes the process to systematically populate the PRCA metamodel.

3.2 PRCA Process

The goal of the PRCA process is to identify the elements of a process and its related softgoals, metrics, and issues for a particular case. The PRCA process essentially includes six major steps.

1. Define a business process model,
2. Define a quality model for a process,
3. Define a softgoal model,
4. Define a correlation model,
5. Define a measurement model for each softgoal, and
6. Identify the issue occurrences.

We discuss each of these steps in detail.

1.) Define a business process model For the presentation in this paper we assume that a business process model has already been defined. If that is not the case, it can be constructed, for instance, following the guidelines of [3, 15].

2.) Define a quality model The objectives of this step is to identify all the potential quality requirements for functions of the process. The notion of a quality requirement basically matches non-functional requirements in the software engineering discipline [12]. According to the ISO standard, quality is defined as "the totality of the characteristics of an entity that bear on its ability to satisfy stated and implied needs". The essential characteristic of a quality requirement is that it can be achieved in different degrees. These quality requirements do not directly relate to functionality in the process, but to conditions and constraints that should prevail [16].

The quality model to be identified in this step can be either defined from scratch or adapted from prior research. The latter choice might seem more appropriate since there exist some useful classifications. Table 1 shows a selection of quality requirements that are listed in the ISO 9126 standard [12] and in the Total Data Quality Management approach [17]. The definition of a suitable quality model for the process then basically becomes a process of selecting the relevant quality requirements from one or more existing models. To achieve this, PRCA follows two steps:

1. Identify process goals: In this step the overall goal of the process is identified. For our SH example process, this is to *submit an accurate and competitive RFP*.
2. Identify quality dimensions: This step yields a list of quality requirements for the process. This critical task heavily relies on the assessment and the expertise of the domain experts. For the RFP process, the main quality dimensions are identified as *accuracy*, *completeness*, *free of errors*, and *reputation*.

Table 1. Two different quality models

ISO 9126 (selection) [12]	TDQM [17]
Accuracy	Accessability
Adaptability	Appropriateness
Analysability	Believability
Attractiveness	Completeness
Changeability	Concise representation
Efficiency	Consistent representation
Fault tolerance	Ease of Manipulation
Interoperability	Free of error
Learnability	Interpretability
Maturity	Objectivity
Recoverability	Relevancy
Security	Reputation
Stability	Security
Suitability	Timeliness
Testability	Understandability
Understandability	Value added

The result of this step is a set of quality requirements. These general requirements have to be linked to functions and their inputs, outputs, and resources in the next step.

3.) Define a softgoal model The objective of this step is to specify softgoals for each function based on the identified quality requirements. For the identification of softgoals, PRCA relies on asking *why*, *what* and *how* questions for each quality requirement (cf. [18]). In order to achieve a higher degree of detail we then decompose these softgoals by asking the same questions with respect to the inputs, outputs, and resources involved in the function.

Consider for example the *accuracy* requirement of the *Estimate* function (see Figure 3). We decompose the softgoal *Ensure accuracy of way estimate is conducted* by asking:

- What input characteristic establishes accuracy? In this case, it is the accuracy of how the solution is defined.
- What output characteristic relates to accuracy? Here, it is the estimate that needs to be accurate.
- What resource characteristic leads to accuracy? For this function, it is the estimation competence of the development department.

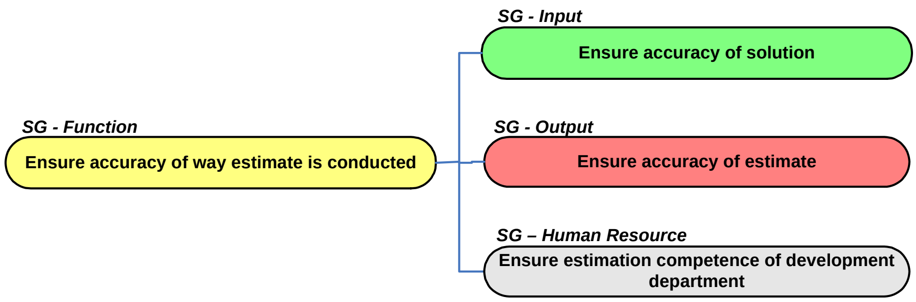

Fig. 3. Softgoals for the function *Estimate*

This analysis is performed for each function yielding a softgoal model for the process.

4.) Define a correlation model In this step the correlations between the different softgoals are identified. In the PRCA approach we only consider positive correlations between goals. In principle, each softgoal can be correlated to any other softgoal. Yet, there are some constraints. First, softgoals related to a function f should only be correlated to softgoals of other functions that are reachable from f. This condition builds on the basic property of causality that a can only cause b if it proceeds b temporally. In the context of a process this means that softgoals of a function f can only have a positive impact on those softgoals that relate to a function executed later. Second, the correlation relation between different softgoals should be irreflexive and acyclic. This guarantees that a softgoal does not influence itself.

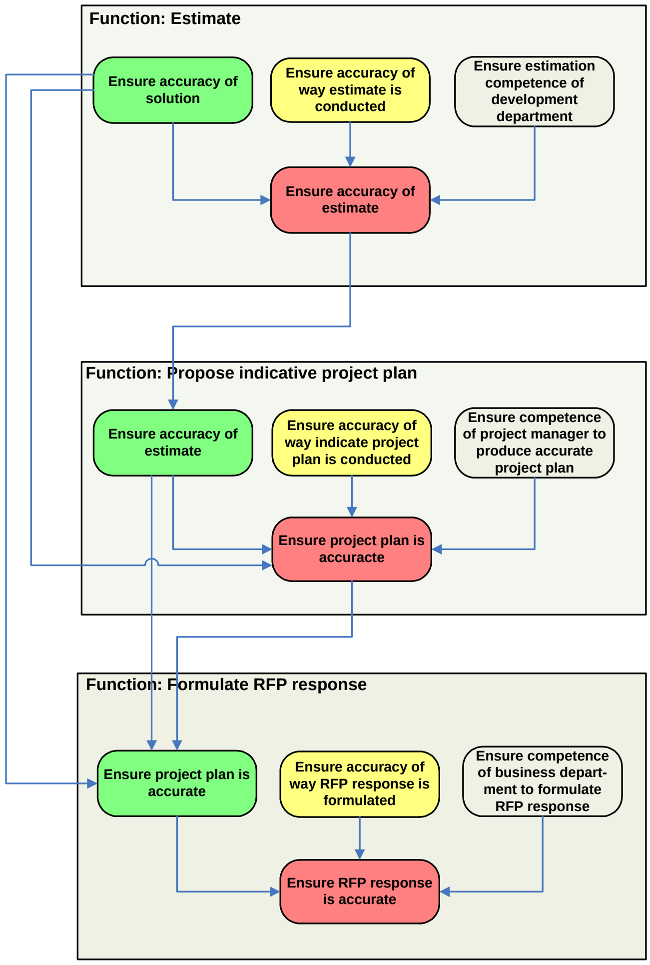

Fig. 4. Correlations of softgoals

Figure 4 shows the correlation model of softgoals from three functions of the RFP process. A direct correlation is established by the fact that softgoals for the output of functions match softgoals where their outputs become inputs of downstream functions. This way the correlation model reflects the data flow and the control flow of the process. From the model is can be seen that the softgoal *Ensure RFP response is accurate* is correlated with various softgoals of previous functions. Furthermore, the *Ensure accuracy of solution* correlates with several softgoals downstream. While the correlation model already highlights some potential root causes for the project loss of SH, it does not reveal which issues caused the problem in our particular case. We have to objectively measure the occurrence of issues before we can trace back the problem.

5.) Define a measurement model The objective of this step is to create a model which contains attributes required to measure a partial softgoal satisfaction. Softgoals as defined in the previous steps can describe quality requirements in abstract terms. Therefore a measurement technique is required to unambiguously identify the degree to which a softgoal is satisfied [10]. To achieve that, we first have to identify suitable metrics that measure the satisfaction level of the softgoal. We adapt the Goal-Question-Metric (GQM) approach [13] for this purpose but cover only the conceptual level (softgoals) and the quantitative level (metrics) in the PRCA metamodel. In particular, we use three groups of questions to generate metrics for softgoals of a function:

1. Questions regarding how to characterize the function, its inputs or outputs, and its resources with respect to its softgoals. For example, the accuracy of the way the indicate project plan is conducted depends on the time that is spent on this function. Accordingly, SH identifies a metric *duration of project plan preparation*.
2. Questions regarding how to characterize the attributes of the function, its inputs or outputs, and its resources that are relevant with respect to its softgoals. For example, the competence of the project manager is measured by *number of planned projects*.
3. Questions regarding how to evaluate the characteristics of the function, its inputs or outputs, and its resources that are relevant with respect to its softgoal. For example, the accuracy of the estimate is measured in how far it deviates from an estimate calculated by a company-specific *cost model* of SH for a given set of solution parameters.

For each of the metrics, we have to identify a failure condition to signal that we have an issue with this quality requirement. The software house SH considers it to be an issue if the *duration of project plan preparation* takes less than one day, or the *number of planned projects* of a manager is less than five, or the *cost model* deviates more than 50% from the estimate.

6.) Identify Issue Occurrence Having identified a suitable set of metrics, the software house SH turns to the way the RFP process was executed for the loss project. It finds out that the project plan was not accurate because it missed to include a major cost block. This comes as a surprise since the project plan was set up by an experienced project manager who has a track record of more than 40 planned projects. Yet, the duration of project plan preparation is less than three hours. This issue appears to be the root cause for the problem with a potential loss from the project.

3.3 Implications of the PRCA Analysis

Given the result of the PRCA analysis there are different ways to tackle the root cause. The board of SH discusses the following options:

- Introduce Quality Assurance Step: In order to guarantee the accuracy of the project plan, an additional step can be introduced in the process to check the correctness, completeness, and accuracy of the project plan.
- Introduce Escalation: Another approach is to introduce an escalation mechanism that keeps track of the metrics while executing the process. The workflow system that supports the RFP process can then automatically send an email to the CIO if an issue occurs.
- Change Resource Allocation: The project manager who designed the project plan obviously did not do a good job for the recent project. The introduction of the four-eyes principle can help to mitigate the risk of bad performance.

An interview with the project manager reveals that he is heavily overloaded with setting up and supervising projects. Indeed, the number of software projects has grown in the last year by 50%. The board of SH agrees to hire a second project manager in order to guarantee that project plans are developed at the required level of accuracy in the future.

4 Related Work

Our PRCA approach essentially relates to two areas of research: on the one hand, work on root cause analysis and process analysis methodologies, and on the other hand goal-oriented approaches to business process modeling.

Root cause analysis is used as a problem solving technique in a variety of quality-centered management approaches such as Six Sigma. Conceptually, it builds on concepts from organizational learning, e.g. [5, 19, 20]. In this context, our PRCA approach is closely related to so-called Ishikawa diagrams [6]. These Ishikawa diagrams visualize the causes behind an issue in form of fishbones. In order to accelerate the identification of such issues, the bones in these diagrams are often pre-populated (e.g. equipment, process, people, materials, environment and management). The PRCA approach builds on identifying softgoals and metrics for objectively measuring the occurrence of an issue related to a softgoal of a particular function. This way, it advances quality management techniques by offering a systematic process to populate the model. Further but typically less popular approaches for root-cause analysis are failure mode and effects analysis, pareto analysis, fault tree analysis or cause mapping. An overview about these and other tools used in Six Sigma can be found among others in [1]. Applications in the area of information systems analysis include [21, 22].

The *business process management community* has been rather slow in exploring the integration of root cause analysis. The traditional focus is still very much on the design of models that reflect current practices (so-called as-is models) followed by the design of an improved (to-be) process model [3]. These models, however, only visualize the symptoms of a process and provide only limited insights into related context factors or even cause-effect-relationships that go beyond the typical semantic relationships in control flows. As a consequence, the white space between as-is and to-be is poorly supported by popular process modeling tools. These tools and related process execution engines rather provide support for the design and instantiation of process models without facilitating a root-cause analysis in the established tradition of the quality management community. Our PRCA approach advances the state-of-the-art in the business process analysis with a systematic process to identify ways to improve weaknesses of a process. This way, it makes concepts from quality management available for to-be process modeling.

There has been substantial work on *goal-oriented approaches* to business process modeling [23, 24]. The importance of identification and inclusion of non-functional aspects of a business process in modelling for the purpose of business process improvement have been discussed, in particular, in [25, 26]. Yet, the integration of control-flow based and goal-oriented modeling of processes has not been explored at its full depth up to now. The research by Soffer and Wand appear to provide the closest alignment by discussing in how far control flow supports the achievement of goals of a process [27, 28]. Our PRCA approach complements this area of research by taking the control flow as a starting point to identifying softgoals and their correlation. To our best knowledge our work in unique in its combination of goal-based and activity-based modeling for root cause analysis in business processes.

5 Conclusions

In this paper we have introduced the PRCA approach to root cause analysis in business processes. In particular, we have defined the respective metamodel and described the process of populating this metamodel for a particular analysis case. We have used the example of the request for proposal (RFP) process of the software house SH to illustrate the approach. While we considered Event-driven process chains (EPCs) as a modeling language in this example, the PRCA approach can easily be adapted for other languages like BPMN or Petri nets. Our contribution is a unique and novel combination of goal-based and activity-based modeling concepts for conducting root cause analysis in business processes.

At this stage our research has some limitations. So far, we have conducted root cause analysis using PRCA for some processes that we had documented in the past. Currently, we are running joint projects with a major Australian bank and with a governmental agency to gain further insights into the scalability of the approach. Based on these projects we aim to acquire a better understanding in which way tools should support PRCA, for instance, by semi-automatically deriving potential questions based on softgoals, or by offering suitable visualizations of the softgoal correlation model. This way, we want to identify how business process modeling tools can be extended to support root cause analysis.

References

1. Antony, J.: Six sigma for service processes. Business Process Management Journal **12** (2006) 234–248

2. Andersen, B., Fagerhaug, T.: Root Cause Analysis: Simplified Tools and Techniques. ASQ Quality Press (2006)
3. Becker, J., Kugeler, M., Rosemann, M.: Process Management: A Guide for the Design of Business Processes. Springer-Verlag (2003)
4. Mendling, J., Aalst, W.: Formalization and Verification of EPCs with OR-Joins Based on State and Context. In Krogstie, J., Opdahl, A., Sindre, G., eds.: Proceedings of the 19th Conference on Advanced Information Systems Engineering (CAiSE 2007). Volume 4495 of Lecture Notes in Computer Science., Trondheim, Norway, Springer-Verlag (2007) 439–453
5. Argyris, C., Schön, D.: Organizational Learning: A Theory of Action Perspective. Addison-Wesley Reading, MA (1978)
6. Ishikawa, K., et al.: What is Total Quality Control?: The Japanese Way. Prentice-Hall Englewood Cliffs, NJ (1985)
7. Brinkkemper, S., Saeki, M., Harmsen, F.: Meta-modelling based assembly techniques for situational method engineering. Information Systems **24** (1999) 209–228
8. Ralyte, J., Rolland, C.: An Assembly Process Model for Method Engineering. Advanced Information Systems Engineering: 13th International Conference, CAiSE 2001, Interlaken, Switzerland, June 4-8, 2001: Proceedings (2001)
9. Chung, L., Nixon, B.A.: Dealing with non-functional requirements: Three experimental studies of a process-oriented approach. In: International Conference on Software Engineering. (1995) 25–37
10. Jureta, I.J., Faulkner, S., Schobbens, P.Y.: A more expressive softgoal conceptualization for quality requirements analysis. In: 25th International Conference on Conceptual Modeling, Tucson, Arizona. (2006) 281–295
11. Cleland-Huangs, J., Settimi, R., BenKhadra, O., Berezhanskaya, E., Christina, S.: A more expressive softgoal conceptualization for quality requirements analysis. In: Proceedings of the 27th international conference on Software engineering, St. Louis, MO, USA. (2005) 362–371
12. ISO/IEC-9126-1: Software engineering – product quality – part 1: Quality model (2001)
13. Basili, V., Rombach, H.D.: The TAME project: Towards improvement-oriented software environments. IEEE Transactions on Software Engineering **14** (1988) 758–773
14. Lamsweerde, A.V.: Goal-oriented requirement engineering: a guided tour. In: Proceedings of the RE'01 Int. Joint Conference on Requirement Engineering. (2001) 249–63
15. Scheer, A.W., Thomas, O., Adam, O.: Process Modeling Using Event-Driven Process Chains. In: Process Aware Information Systems: Bridging People and Software Through Process Technology. Wiley Publishing (2005) 119–146
16. Cysneiros, L.M., do Prado Leite, J.C.S., de Melo Sabat Neto, J.: A framework for integrating non-functional requirements into conceptual models. Requirements Engineering **6** (2001) 97–115
17. Wang, R.: A product perspective on total data quality management. Communications of the ACM **41** (1998) 58–65
18. Lamsweerde, A.V.: Requirements engineering in the year 00: A research perspective. In: Proceedings of the ICSE'2000 - 22nd International Conference on Software Engineering, Limerick. (2000) 5–19
19. Fiol, C., Lyles, M.: Organizational Learning. The Academy of Management Review **10** (1985) 803–813

20. Mason, R.: Strategic information systems: use of information technology in a learning organization. Proceeding of the Twenty-Sixth Hawaii International Conference on System Sciences **4** (1993)
21. Julisch, K.: Clustering intrusion detection alarms to support root cause analysis. ACM Transactions on Information and System Security (TISSEC) **6** (2003) 443–471
22. Pei, D., Azuma, M., Massey, D., Zhang, L.: BGP-RCN: improving BGP convergence through root cause notification. Computer Networks **48** (2005) 175–194
23. Rolland, C., Prakash, N., Benjamen, A.: A multi-model view of process modelling. Requir. Eng. **4** (1999) 169–187
24. Kavakli, E., Loucopoulos, P.: Experiences with goal-oriented modelling of organisational change. IEEE Transactions on Systems, Man and Cybernetics - Part C **36** (2006) 221–235
25. Aburub, F., Odeh, M., Beeson, I.: Modelling non-functional requirements of business processes. Information and Software Technology **49** (2007) 1162–1171
26. Grau, G., Franch, X., Maiden, N.A.M.: Prim: An i*-based process reengineering method for information systems specification. Information and Software Technology **50** (2008) 76–100
27. Soffer, P., Wand, Y.: Goal-driven analysis of process model validity. In Persson, A., Stirna, J., eds.: Advanced Information Systems Engineering, 16th International Conference, CAiSE 2004, Riga, Latvia, June 7-11, 2004, Proceedings. Volume 3084 of Lecture Notes in Computer Science., Springer (2004) 521–535
28. Soffer, P., Wand, Y., Kaner, M.: Semantic analysis of flow patterns in business process modeling. In Alonso, G., Dadam, P., Rosemann, M., eds.: Business Process Management, 5th International Conference, BPM 2007, Brisbane, Australia, September 24-28, 2007, Proceedings. Volume 4714 of Lecture Notes in Computer Science., Springer (2007) 400–407