Perceived Feasibility of Using Root Cause Analysis in Post Project Reviews: an Empirical Investigation

Timo O.A. Lehtinen
Department of Computer Science and Engineering, School of Science, Aalto University
P.O. BOX 15400
FI-0076 Aalto Finland
+358 40 775 2781
timo.o.lehtinen@aalto.fi

ABSTRACT

Root cause analysis (RCA) is a structured investigation of the problem to identify which underlying causes need to be fixed. In software engineering (SE), the scientific work on RCA is rather scarce. Feasibility of RCA to identify SE problem causes is not widely studied, the RCA methods are not compared with one another, and there are only a few studies on the required effort to conduct RCA. Additionally, studying how the participating people experience RCA is totally set aside.

This dissertation focuses on analyzing the perceived feasibility of using RCA, as a part of post project reviews to reveal the root causes of software engineering problems aiming to increase productivity through software process improvement. The overall approach in this thesis is a mixed-methods approach that combines three main research approaches: design science with observation-based industrial field studies, case studies, and experiments.

So far, this work has made four contributions. First, based on prior studies, definitions for RCA and a root cause have been made. Second, a comparison of the known RCA methods is done. Third, an evaluation of a lightweight RCA method, the ARCA method, has been conducted in four medium sized software companies. Fourth, analyses on the problem causes detected in those software companies have been done.

Currently, this research aims at showing whether the ARCA method can be utilized to analyze the causal relationships of problem causes over process areas. An article related to this study will be submitted during the next few months.

Keywords

Software Engineering (SE), Software Process Improvement (SPI), Root Cause Analysis (RCA), Post Project Review

1. INTRODUCTION

Software projects usually encounter problems and challenges [8, 11, 12, 14, 22, 23, 31, 33, 34, 43]. Analyzing the causal relationships of the problem causes has been considered in various software process improvement models, e.g., CMMI, ISO/IEC 12207, and Six Sigma [21]. It is argued that the key for effective problem prevention is to know why the problem occurs as the reoccurrence of the problem can be prevented only through the elimination of its causes [36]. Root cause analysis (RCA) is a structured investigation of the problem to identify which underlying causes need to be fixed [24]. RCA takes a problem as an input and provides a set of related causes with cause and effect structure as an output [26]. It aims to state what the causes of the problem are and where in the development process they occur [25].

Methodologies of RCA [26] are fairly little studied area in the context of software engineering (SE). Additionally, the terminology of causal analysis is vague as there seems to be no commonly accepted definition for RCA [3, 24] or for a root cause. In general, RCA methods contain three phases: target problem detection, root cause detection, and corrective action innovation [26]. Many authors define a root cause as a cause that management has the power to fix [2, 3, 29, 36]. RCA is an approach that can help with the software process improvement and problem prevention in various contexts [1, 4, 5, 15, 18, 21, 32, 39, 40, 42] and all across software organizations, including product development, hardware design, product engineering, and manufacturing [32].

This dissertation focuses on analyzing the perceived feasibility of using RCA as a part of post project reviews to reveal the root causes of SE problems aiming to increase productivity. The overall research approach in this thesis is a mixed-methods approach [38] that combines three main research approaches: design science with observation-based industrial field studies, case studies, and experiments.

The rest of the paper is structured in the following way. Section 2 introduces the background of the dissertation and summarizes the gaps in the prior studies. Section 3 presents the research objectives and questions. Thereafter, the research methods are introduced in Section 4 and Section 5 shows the publication plan. Finally, Section 6 discusses the validity threats and Section 7 describes the open issues which the author would like to get the most advice on. Section 8 summarizes this research plan and presents the main conclusions from the first publications of the dissertation.

2. BACKGROUND

2.1 Definitions of Root Cause Analysis

The goal of RCA is to decrease the likelihood of a problem's reoccurrence by controlling its root causes [9, 10, 27, 36]. There is a slight disagreement while considering the definition of RCA [3, 24] and a root cause. RCA has been introduced as a cause detection method only [2, 5, 24, 27, 36], but also as a problem prevention method that includes the detection of problem causes and development of corrective actions [3, 9, 10, 40].

A root cause has been defined as the deepest cause at the end of the causal structure [2, 3], but also as any underlying cause of a target problem [36]. Most of the authors recognize a root cause as a problem cause that management has the power to fix [2, 3, 29, 36]. Conceptually, a target problem may have numerous root causes.

In the terminology of this dissertation, *RCA is a process of detecting a target problem, detecting and organizing its causes, and recognizing its root causes.* We define a root cause as *an underlying cause of the target problem that explains the occurrence of the target problem unambiguously*.

2.2 Phases of Root Cause Analysis

There are three phases that are common between the RCA methods introduced in the literature: 1. target problem detection, which defines the target problem to which the RCA method is applied, 2. root cause detection, which identifies and organizes the root causes of the target problem, and 3. corrective action innovation in which corrective actions for the most important root causes are innovated [26]. Alternative methodologies have been presented for each of the above phases [26], which are introduced below.

A target problem for RCA is detected through problem sampling [3, 6, 9, 15, 18, 21, 24, 27], interviewing [24, 35, 36], brainstorming [3, 24], or flowcharting [2, 3, 24]. Usually there is a meeting where the target problem is finally decided upon [6, 9].

Identifying and organizing the target problem root causes can be done with different means [35]. Usually the root cause detection relies on the assumptions of various stakeholders [6, 9, 24, 35]. Techniques to identify the problem root causes from the stakeholders include interviewing [2], questionnaires [3, 7], brainstorming, and brainwriting [3, 7, 24].

The target problem root causes are organized into a cause and effect diagram based on their causal relationships. Various cause and effect diagramming techniques has been introduced and compared [26]. Tree diagrams include a fishbone diagram [3, 5, 6, 40, 41], a fault tree diagram [3], a logic tree [24], and a causal factor chart [36]. Network diagrams include a directed graph [5] and a matrix diagram [3]. Additionally, lists, worksheets, and charts can be used to organize the target problem root causes [2].

Cause Enumeration [3] is a strategy where the problem causes are first brainstormed individually and then grouped under various categories. In the SE context, the tree structured fishbone diagram has been combined with the cause enumeration strategy [9, 15, 16, 18, 27].

Dispersion Analysis [3] is a visualization technique for tree and network structures. There the target problem is presented as a root of the cause and effect structure which is made by looking up the causes for the target problem and then collecting their sub causes by constantly asking "why" for every cause detected.

The root causes are detected by focusing on the target problem causes that will be prevented [9, 36]. It has been indicated that the root cause detection should emphasize the level of controllability while considering the root causes [26]. Thereafter, corrective actions are developed for the selected root causes.

2.3 Root Cause Analysis in software process improvement

RCA has been introduced as a feasible approach to software process improvement at various levels of the company. Additionally, it has been considered in various software process improvement models, e.g., CMMI, ISO/IEC 12207, and Six Sigma [21]. In the models, the target problem for RCA is detected from one of the main targets for software process improvement, i.e., the costs of development work, the time to market, and the product quality [7].

Massive defect prevention programs aiming to improve company processes have been utilized [32]. On the other hand, individual project reflections [4, 13, 15] aiming at continuous learning and improvements are usual. RCA has been successfully utilized in both of these extremes. In software defect prevention programs [32], RCA has been successfully utilized to detect problem causes from various company processes, whereas, RCA has been successfully utilized in post project reviews [5, 13] to detect causes internal to the project team helping the team to improve their work practices.

2.4 Gaps in the Priors Studies

The scientific work on the RCA methodologies is rather scarce. RCA has been introduced as a feasible approach for post project reviews [5], where the goal is to improve the development processes through learning from the past failures. However, the practices [26] and output [25] of RCA are fairly little studied area in the SE context.

There are only a few previous studies [5, 21, 26] on how to collect, organize, and select causes of a target problem and how to develop corrective actions for them. The RCA methods presented by many authors [2, 3, 5, 6, 9, 10, 15, 18, 19, 21, 24, 27, 29, 36, 40] are too generally introduced to be adopted as such, e.g. the RCA method presented by Card [9] introduces the mandatory phases of RCA but does not go beyond that, e.g. he does not explain details about how these phases should be conducted. Thus, there is lack of knowledge on what RCA methodologies are feasible in post project reviews and how to apply them.

Additionally, most of the industrial cases of RCA [9, 15, 16, 18, 27] have focused on the problem causes only while disregarding the analyses of the related causal relationships, possibly one of the main benefits of RCA. Furthermore, the cases are based on defect causal analysis [9, 15, 16, 18, 27], which aims solely at lower defect rates by revealing the root causes of the most typical types of the defects. The high number of particular types of software defects is not the only target problem that should be analyzed while evaluating the feasibility of using RCA to reveal the root causes of SE problems, e.g., negative project experiences [5], software project overruns and challenging product installations [25] are all industrially relevant and severe problems but have only been exiguously explored using RCA [26].

Furthermore, the feasibility of RCA in general and in contrast to other process improvement approaches is not widely studied. Similarly, the RCA methods are not compared with one another. Finally, there are only a few studies on the required effort to conduct RCA [9, 15, 26, 32], and studying how the participating people experience RCA is totally set aside, e.g. do the practitioners experience RCA as a useful approach for software process improvement?

Even though the results of the prior RCA studies are promising covering 50 percent decrease in defect rates [9], 53 percent savings in costs and 24 percent increase in productivity [27], the studies do not indicate the feasibility of using RCA in post project reviews where also other problems than technical quality deviations are analyzed. Thus, many questions remain unanswered including "How do practitioners experience using RCA in post project reviews," "Can RCA reveal the causal relationships of problem causes detected in post project reviews," and "Is the output of post project reviews utilizing RCA cost efficient?" The research problem follows:

Is RCA feasible in post project reviews?

3. RESEARCH OBJECTIVES

This dissertation focuses on studying the perceived feasibility of using RCA in post project reviews in the SE context. In the terminology of this work, the feasibility refers to the ease of use and cost efficiency of using RCA to detect process improvement

targets. Furthermore, the term post project review refers to general project reflections conducted at the end of the development work aiming to learn from the past failures, e.g., agile retrospectives.

This work is limited to evaluations of the work practices of the ARCA method [26], which is a lightweight RCA method developed at the beginning of this PhD research. As already presented in Section 2.4, the prior RCA methods are too generally introduced to be adopted as such. Additionally, in SE context, the prior RCA methods are mainly used to analyze the quality deviations of the product, which requires specific work practices being infeasible for many SE problems other than software defects, e.g., defect sampling cannot be used to the problems which are not reported. Thus, in order to have measurable RCA construction and be able to experiment RCA with various types of SE problems, developing the ARCA method was important.

The ARCA method is based on a structured literature review on prior RCA methods and it follows their common phases and work practices (see Section 2.2). In the ARCA method, the target problem detection is conducted at a focus group meeting with the key stakeholders of the project. Thereafter, the root cause detection is conducted through confidential email inquiry and a public RCA workshop. Finally, after the RCA workshop, the corrective actions for the selected root causes are developed. More details of the ARCA method and its evaluations can be found in [26].

The following research questions are addressed in the dissertation:

1. *Is the ARCA method perceived feasible in post project reviews to detect the root causes of SE problems and develop corrective actions for them?* This question includes the experimentations of RCA work practices and analyses of their perceived feasibility.
2. *Is the output of the ARCA method perceived as beneficial for solving SE problems?* This question includes analyses of the effect of RCA, i.e. the perceived correctness of detected causes, and quality of corrective actions developed.
3. *Is the effort the ARCA method requires perceived as feasible?* This question includes analyzes on the man-hours required to conduct the ARCA method.

4. METHODS

The overall approach in this dissertation is a mixed-methods approach [38] that combines three main research approaches: design science [17] with observation-based industrial field studies [28], case studies [44], and experiments [20]. The empirical data is based on qualitative and quantitative sources: interviews, questionnaires, observations, video analyses, measurements (effort used, the number of participants, etc.), and the output of the ARCA method including the detected target problems, related root causes and their corrective actions. Table 1 summarizes the studies of the dissertation in chronological order and shows the research data collected to answer the related research questions.

The observation-based industrial field studies [28] with software product companies are important sources of research data when evaluating the practical value of the observed aspects. In this dissertation, these include the work practices of the ARCA method developed by using a framework similar to that of design science [17, 30]. Field studies at four medium-sized software companies have already been conducted [26]. In the field studies, the case companies applied the ARCA method to their software project failures. The field studies were video recorded and observed. Additionally, the output of the ARCA method was evaluated by the company people, i.e. the people were interviewed before and after each field study, and the people were asked to provide feedback though inquiry forms, prepared by the researchers.

Case studies [44] are used to analyze the output of the ARCA method in industrial settings. This includes analyzing the problem root causes and their corrective actions in the case companies to understand whether the output of the ARCA method is useful and valid. The research focuses on analyzing whether the causal relationships detected resulted in corrective actions, which are feasible, accurate, and have a high impact while trying to prevent the reoccurrence of the target problem.

The controlled experiments [20] extend the field studies and case studies by focusing on the cause and effect diagramming techniques to organize the detected root causes and their causal relationships. The experiment with 11 student software project teams has already been conducted. The experiment followed the factorial experiment design with repeated measures in blocks of size 2 [20], i.e., two varying treatments were experimented with random order with each student team. Additionally, 50% of the teams started with the first treatment and 50% as vice versa. In the first treatment, a directed graph of causes introduced in the ARCA method was used to organize the detected problem root causes whereas the second treatment applied a hierarchical list of causes. Dada collection was carried out by using the same procedures as in the field studies. In addition to the student experiment, an industrial experiment is planned to be conducted. There, the focus will be on comparison of whether or not utilizing cause enumeration during the root cause detection is feasible.

5. PUBLICATION PLAN

This work was started on June 2010 and currently 2 / 6 research articles are published [25, 26]. Additionally, the research data is collected for three of the remaining articles and currently the writing work and related analyses are in progress for the next two articles. The industrial experiment on the feasibility of utilizing a cause enumeration during RCA is still in the planning phase. The dissertation summary will be finalized by the end of 2013. Next, we look at the articles and their status in chronological order based on the time the research takes place. All the articles can be seen in Table 2.

Table 1: Studies of the Dissertation

Publication	Article 1	Articles 2 and 3	Article 4	Articles 5 and 6
Research Method	Industrial Field Study	Case Study	Case Study	Experiment
Research Focus	*Effectiveness and Feasibility*	*Output Quality*	*Output Quality*	*Effectiveness and Feasibility*
Research Data	Observations, Interviews, Questionnaires, Effort measurements	Analysis of problem causes and related causal relationships	Analysis of corrective actions developed	Observations, questionnaires, interviews, Cause measurements
Research Q1	X			X
Research Q2	X	X	X	X
Research Q3	X			X

Table 2: Articles of the Dissertation

1.	T. O. A. Lehtinen, M. V. Mäntylä and J. Vanhanen, Development and evaluation of a lightweight root cause analysis method (ARCA method) – field studies at four software companies, Information and Software Technology 53 (10) (2011) 1045-1061.
2.	T. O. A. Lehtinen and M. V. Mäntylä, "What are Problem Causes of Software Projects? Data of Root Cause Analysis at Four Software Companies," ESEM, pp.388-391, 2011 International Symposium on Empirical Software Engineering and Measurement, 2011.
3.	Analyzing Causal Relationships Networks of Software Project Failures
4.	Does Root Cause Analysis Result to Accurate and Effective Process Improvement Ideas?
5.	Is a Cause and Effect Diagram Really Needed in Post Project Reviews?
6.	Feasibility of Using a Cause Enumeration in Post Project Reviews to Detect Problem Causes

The first article of the dissertation [26] provides a terminology of root cause analysis. It is based on a structured literature review and introduces the most recognized RCA methods [2, 9, 24, 36] while using them as a starting point. It includes an analytical argumentation of those methods and introduces a new RCA method, the ARCA method, combining the best practices of the prior RCA studies. The ARCA method is evaluated in industrial field studies at four software product companies where it is compared with the current software process improvement practices. Based on the results from feedback forms, interviews and observations, our results indicate that the ARCA method is perceived as a highly feasible method for software process improvement. This article provides information that will be used to answer the research questions 1-3.

The second article of the dissertation [25] uses the cause data, collected in the field studies as a starting point. The article introduces a classification system that can be used to analyze the causes related to software project failures. Additionally, the paper includes the preliminary results of the distributions of the problem root causes in our field studies. The classification system was developed iteratively by using a literature review followed by a grounded theory approach. The finalized classification system includes two dimensions: process areas and cause types. The cause types states what the failure cause is, whereas the process areas states where it occurs.

The third article of the dissertation continues the work of the second article [25]. It aims to show whether RCA can be utilized to analyze causal relationships over the detected problem root causes. The article reports the results from analyses on what kinds of causal relationships are related to software project failures. This increases our understanding about the causal structures and case sensitivity of software project failures. As far as we know, there are no prior studies indicating how the failure causes are interconnected to one another. To validate the conclusions of this study, the causal relationships of the detected root causes are compared with the failure causes detected in prior studies. We hypothesize that the failure causes detected in prior studies need to be interconnected to one another in our data set. This article provides information that will be used to answer the research question 2.

The fourth article of the dissertation uses the results from the second and third articles, and the data from the field studies as a starting point. It aims to show whether RCA helps to develop accurate corrective actions in contrast to the detected problem causes. It continues the prior studies of the dissertation by focusing on the corrective actions developed in the field study companies. This article provides information that will be used to answer the research question 2.

The fifth article of the dissertation challenges the unproven value of using a cause and effect diagram in RCA when detecting and analyzing the problem root causes. Although the results of the first four articles were acquired from industrial settings, they contained limitations. Usage of a cause and effect diagram has become a de facto in the RCA methods [3, 5, 6, 24, 36, 40, 41] and thus our field studies were also based on using that technique [26]. However, as far as we know, the added value of using the cause and effect diagram has not been questioned before. Indeed, in contrast to creating the cause and effect diagram, creating a hierarchical list of causes [2] could be more efficient and user friendly. To assess this limitation, a student experiment was arranged. The experiment provided a higher number of individuals and a better control of the research settings. The experiment was conducted in the post project reviews of 11 software teams. Each team consisted of five to seven team members who conducted RCA in two separated reviews. The first review was conducted either with or without a cause and effect diagram whereas the second review was conducted as vice versa. Each review included an analysis of two occurred problems of the team's project. The experiment measures the causes and causal relationships detected and how the students experienced the usefulness of RCA in each post project review. This article provides information that will be used to answer the research questions 1-3.

The sixth article reports our findings from the industrial experiment where RCA will be utilized in post project reviews. The goal is to study the feasibility of using the cause enumeration technique (see Section 2.2) during RCA. This continues the work of the fifth article by questioning the feasibility of using the fishbone diagram in RCA. The post project reviews are conducted with two agile software teams that both develop software components into the same product and in the same company. Both teams conduct two separate reviews. The experiment measures the causes and causal relationships detected and how the practitioners experience the usefulness of RCA in each post project review. This article provides information that will be used to answer the research questions 1-3.

6. VALIDITY THREATS

The validity threats of the dissertation are divided into four aspects. Construct validity reflects the extent to which the studied operational measures really represent what is investigated according to the research questions [37]. External validity is concerned with whether it is possible to generalize the findings of the study and to what extent they can be generalized [37]. Internal validity is of concern when the causal relations of the measured factors are examined [37]. Reliability is concerned with the extent to which data and analysis are dependent on a specific researcher [37].

6.1 Construct Validity

The construct validity relates to the RCA settings in the studied domain and the measurements, query forms, and interviews that were carried out to answer the research questions. There is a validity threat related to the ARCA method. A high number of different RCA settings exist, but only a few of them are experimented in the dissertation. The experiments and field studies are mostly being relied on the ARCA method that was a synthesis of the known RCA methods including their best practices introduced in the first article [26]. While trying to understand general characteristics of the RCA methods and even though the ARCA method likely includes the best practices of known RCA methods, it does not cover them all, e.g., a problem sampling with Pareto Analysis [9] is excluded from this dissertation. Thus, while referring to RCA, the dissertation mostly measurers the practices of the ARCA method and covers the other RCA methods only through their similarities with the ARCA method.

Additionally, there is a threat to the construct validity regarding the evaluations on the output quality and feasibility of the ARCA method. The analyses are based on the experiential evaluation of the case attendees only, not on monitoring the problem domain systematically afterward. Generally, it should be noted that this sort of validity problem is common, as it is practically impossible to separate the effects of the RCA method from the company specific context factors.

6.2 Internal Validity

This dissertation includes two experiments (see Table 1). There is a threat to the internal validity while considering the causal relationships between the measured factors. In the first experiment, the impact of using a cause and effect diagram in post project reviews was measured through the amount of new problem causes detected and the causal structure over the detected problem causes. Additionally, feedback from participating people was measured by using feedback forms. These results were thereafter compared with the results made by using the hierarchical list with otherwise similar RCA settings. Such a comparison is also planned to be used in the industrial experiment where using the cause enumeration during the post project review is compared with the results where the cause enumeration is made after the review. The threat is that the measured impact might not be an effect of the techniques used in the review. Instead, the impact might be an effect of the problem domain itself including the participating people and the target problem. This threat is controlled by using both treatments with the same teams and by randomizing the starting technique of the experiments.

The number of experimentations was relatively high in the first experiment. This increases the internal validity as the effect of the problem domain in contrast to the measured factors decreases. Instead, the number of experimentations in the second experiment is low as the experimentation covers only two teams. While considering the internal validity, the effect of the problem domain remains relatively low as both teams are developing the same product in the same company and thus it is expected that the problem causes are likely similar between the teams.

6.3 External Validity

All of the industrial cases vary and, thus, consider the ARCA method from different perspectives. Though the cases are conducted in different companies, all with different case attendees and target problems, and though the interviews slightly differ between the cases, the results collectively increase the external validity of the results of this dissertation.

Lack of comparison between the RCA methods creates a severe threat to the external validity. It cannot be concluded whether or not the ARCA method is truly efficient and easy to use while there are no extensive comparisons between all possible RCA methods. The main cause for this is the lack of prior studies (Section 2.4). Additionally, the experimentations of the dissertation do not cover a case where the case attendees are highly experienced with RCA methods. Thus, the case attendees' evaluations are not based on prior RCA experiences.

6.4 Reliability

It is possible that the researchers' contribution bias the results of this dissertation. The fact that the researchers involved steering RCA during the experimentations is both a strength and a weakness. The strength is that it makes the results more comparable, as almost everything is done similarly in the experiments and field studies. On the other hand, the weakness is that the collected research data is partially bounded by the researchers' contributions. Thus, there is a threat that the evaluations on the effectiveness and feasibility of RCA made by the participating people are biased.

Furthermore, analyses on the output of RCA are dependent on the researchers' interpretations. The detected root causes and developed corrective actions are studied to evaluate whether the corrective actions are accurate in terms of the related root causes. There is a threat that the accuracy evaluations are biased. This risk is controlled by systematizing the evaluation process. A classification system, which was initially introduced in [25], is used to characterize the detected root causes and related corrective actions. Furthermore, the validity of the classification system is analyzed to conclude whether the system is valid in terms of the reliability threats. Kappa values are used to evaluate the inter-rater agreement of the classification system. However, regardless of the hard effort while trying to make the classification system as comprehensive as possible, classifying problem causes dissipates the dissimilarities and simultaneously highlights the similarities of the problem causes. This means that there is a risk that using the classification system results in systematic errors not dependent on the researcher. This validity threat is controlled by utilizing qualitative research methods. The results of the classification system are compared and combined to interpretations made by looking up the detected root causes and related corrective actions without using the classification system.

7. ISSUES OF THE DISSERTATION

There are several issues of which the author of this dissertation would like to get the most advice on. These are described below. While trying to show evidence on the feasibility of using RCA in post project reviews the research results to analyzing the feedback of the participating people only. Unfortunately, such research data is mostly anecdotal as it is relying on assumptions, not on real evidence. Thus, there is an open issue of how to collect and analyze "real evidence" while trying to analyze whether the detected target problem causes and causal relationships are correct? Furthermore, in general, what evidence can be considered as "real evidence" in the SE context?

Similarly, while trying to analyze the impact of post project reviews, there is a problem of being difficult to measure the change in the upcoming projects as the problem domain varies while being highly dependent on dynamic context variables, e.g.,

project members and scope. Thus, the question is how to measure the change in the vague settings of software engineering?

8. SUMMARY

This dissertation focuses on studying the perceived feasibility of using RCA in post project reviews in the SE context. This contributes to the research problem: *is RCA feasible in post project reviews?* The overall approach in this thesis is a mixed-methods approach [38] that combines three main research approaches: design science with observation-based industrial field studies, case studies, and experiments. The empirical data is based on qualitative and quantitative sources: interviews, questionnaires, observations, video analyses, measurements (effort used, the number of participants, etc.), and the output of the applied RCA methods (target problems, detected causes, and corrective actions).

So far, this work has made four scientific contributions. First, based on prior studies, definitions for root cause analysis and a root cause have been made [26]. This systematizes the otherwise vague terminology of RCA. In our terminology, RCA is a process of detecting a target problem, collecting and organizing its causes, and recognizing its root causes. Furthermore, a root cause refers to any underlying cause of the target problem that the management has the ability to fix. However, this definition has been sharpened to "*an underlying cause of the target problem that explains the occurrence of the target problem unambiguously".* We found three steps that are common to the RCA methods introduced in the literature: 1. target problem detection, which defines the target problem of the RCA method, 2. root cause detection, which detects and organizes the causes of the target problem, and 3. corrective action innovation, which develops corrective actions for the most important root causes.

Second, a comparison of the known RCA methods has been made [26]. This increases our knowledge on what different RCA methods have been introduced. The known RCA methods vary in terms of the target problem and root cause detection. The target problem detection is relying on quantitative methods which include problem sampling combined with Pareto Analysis and qualitative methods such as interviewing software development managers to name the main problems of the development work. Furthermore, the root cause detection varies in terms of the methodologies used to detect and organize the problem root causes. The root causes are usually detected from various stakeholders by using interviewing, questionnaires, brainstorming, and brainwriting methods. The target problem root causes are usually organized into a cause and effect diagram by using a fishbone diagram, a fault tree diagram, a causal map, a matrix diagram, a scatter chart, a logic tree, or a causal factor chart. Additionally, lists, worksheets, and charts can also be used to organize the causes.

Third, the evaluation of a lightweight RCA method, the ARCA method, has been conducted in four medium sized software companies [26]. The companies utilized the ARCA method in their post project reviews. The evaluation of the ARCA method increases our knowledge on the experienced feasibility, cost efficiency and output quality of post project reviews using RCA. In prior works, such data is often missed. For example, in [9], the costs of the RCA method are reported only as a percentage of the yearly development budget instead of more concrete man-hours. Furthermore, the general satisfaction of the participating people is not reported by any of the prior studies. We did that by using interviews and query forms. Our results [26] show that the participating people perceived that the ARCA method required an acceptable level of effort and resulted in numerous feasible corrective actions that will have a high impact on the target problem. Additionally, in contrast to the current company practices, the ARCA method was experienced as an efficient method to detect new process improvement opportunities and develop new process improvement ideas. Furthermore, the ARCA method was perceived as easy to use.

Fourth, analyses on the problem causes detected in the field studies are started [25]. This increases our knowledge on the case sensitivity of software engineering problems and their causes. The results indicate that the causes of software project failures evolve in the steps of process lifecycle through the causes of people, methods, tasks, and environment. This indicates that a cross functional team is likely required while conducting RCA. Additionally, our results show that the problem causes of software project failures are varying, which indicates that a case specific RCA should be conducted rather than by only trying to prevent the common causes of software engineering problems listed in the prior studies.

Currently, the dissertation aims at showing whether RCA can be utilized to analyze the problem causalities over the process areas. An article related to this study will be submitted during the next few months. The results increase our knowledge on how to analyze the causal relationships of SE problems. The preliminary results indicate that by using RCA with a network structured cause and effect diagram, the causal relationships over process areas are detected and can be further analyzed in post project reviews.

Additionally, analyzing the results of the student experiment is in progress. Interestingly, in post project reviews, it seems that using a cause and effect diagram do not increase the amount of new problem causes detected when compared with the treatment where a hierarchical list of problem causes is used. However, the participating people experience using the cause and effect diagram as easier than using the hierarchical list. This indicates that the perceived value of using the cause and effect diagram is higher than it is with the hierarchical list. Next, the comparison of the detected causal relationships will be conducted. This study will increase our knowledge on the impact of using the cause and effect diagramming techniques while analyzing problem causalities.

9. REFERENCES

[1] Al-Mamory, S. O. and Zhang, H. Intrusion detection alarms reduction using root cause analysis and clustering. Computer Communications, 32, 2 (February 2009), 419 - 430. DOI=10.1016/j.comcom.2008.11.012.

[2] Ammerman, M. The Root Cause Analysis Handbook: A Simplified Approach to Identifying, Correcting, and Reporting Workplace Errors. Productivity Press, 444 Park Avenue South, Suite 604, New York, NY 1016, USA, 1998.

[3] Andersen, B. and Fagerhaug, T. eds. *Root Cause Analysis: Simplified Tools and Techniques.* Tony A. William American Society for Quality, Quality Press, United States, Milwaukee 53203, 2006.

[4] Bhandari, I., Halliday, M., Tarver, E., Brown, D., Chaar, J. and Chillarege, R. A Case Study of Software Process Improvement during Development. IEEE Transactions on Software Engineering, 19, 12 (December 1993), 1157-1170.

[5] Bjørnson, F. O., Wang, A. I. and Arisholm, E. Improving the effectiveness of root cause analysis in post mortem analysis:

A controlled experiment. Information and Software Technology, 51, 1 (January 2009), 150 - 161. DOI=10.1016/j.infsof.2008.02.003.

[6] Burnstein, I. *Practical Software Testing.* Springer Science+Business Media, New York, 2003.

[7] Burr, A. and Owen, M. eds. *Statistical Methods for Software Quality: Using Metrics for Process Improvement.* ITP A division of International Thomson Publishing Inc., 1996.

[8] Capers, J. Social and Technical Reasons for Software Project Failures. Crosstalk The Journal of Defense Software Engineering, 6 (2006), 4-9.

[9] Card, D. N. Learning from our mistakes with defect causal analysis. IEEE Software, 15, 1 (1998), 56-63.

[10] Card, D. N. Defect-Causal Analysis Drives Down Error Rates. Quality Time, 10, 4 (July 1993), 98 - 99.

[11] Cerpa, N. and Verner, J. M. Why Did Your Project Fail? Communications of the ACM, 52, 12 (2009), 130-134.

[12] Demir, K. A. A Survey on Challenges of Software Project Management. In Anonymous *Proceedings of the 2009 International Conference on Software Engineering Research Practice,* 2009, 13-16.

[13] Dingsøyr, T. Postmortem reviews: purpose and approaches in software engineering. Information and Software Technology, 47, 5 (2005), 293-303.

[14] Dye, J. and van der Schaaf, T. PRISMA as a quality tool for promoting customer satisfaction in the telecommunications industry. Reliability Engineering & System Safety, 75, 3 (2002), 303-311.

[15] Grady, R. B. Software Failure Analysis for High-Return Process Improvement Decisions. Hewlett-Packard Journal, 47, 4 (August 1996), 15 - 25.

[16] Gupta, A., Li, J., Conradi, R., Rönneberg, H. and Landre, E. A case study comparing defect profiles of a reused framework and of applications reusing it. Empirical Software Engineering, 14, 2 (20 August 2008), 227 - 255. DOI=10.1007/s10664-008-9081-9.

[17] Hevner, A. R., March, S. T., Park, J. and Ram, S. Design Science in Information Systems Research. MIS Quarterly, 28, 1 (2004), 75-105.

[18] Jalote, P. and Agrawal, N. Using defect analysis feedback for improving quality and productivity in iterative software development. In Anonymous *Proceedings of the Information Science and Communications Technology (ICICT 2005).* Infosys Technologies Limited Electronics City, Hosur Road, Bangalore, India, 2005, 701 - 714.

[19] Jin, Z. X., Hajdukiewicz, J., Ho, G., Chan, D. and Kow, Y. Using Root Cause Data Analysis for Requirements and Knowledge Elicitation. *International Conference on Engineering Psychology and Cognitive Ergonomics (HCII 2007).* (Berlin, Germany). Springer Verlag, 2007, 79 - 88.

[20] Juristo, N. and Moreno, A. M. *Basics of Software Engineering Experimentation.* IBT Global, London, 2003.

[21] Kalinowski, M., Travassos, G. H. and Card, D. N. Towards defect prevention based process improvement approach. *Proceedings of the 34th EUROMICRO Conference on Software Engineering and Advanced Applications.* (Parma, Italy, September 3-5). IEEE Computer Society, 2008, 199 - 206.

[22] Kappelman, L. A., McKeeman, R. and Zhang, L. Early Warning Signs of IT Project Failure: The Dominant Dozen. Information System Management, 23, 4 (2006), 31-36.

[23] Keil, M., Cule, P. E., Lyytinen, K. and Schmidt, R. C. A Framework for Identifying Software Project Risks. Communications of the ACM, 41, 11 (1998), 76-83.

[24] Latino, R. J. and Latino, K. C. eds. *Root Cause Analysis: Improving Performance for Bottom-Line Results.* CRC Press, 6000 Broken Sound Parkway NW, Suite 300 Boca Raton, FL 33487-2742, 2006.

[25] Lehtinen, T. O. A. and Mäntylä, M. V. What Are Problem Causes of Software Projects? Data of Root Cause Analysis at Four Software Companies. *Empirical Software Engineering (ESEM 2011).* (Canada, Banff, 22. - 23.9.2011). IEEE, 2011.

[26] Lehtinen, T. O. A., Mäntylä, M. V. and Vanhanen, J. Development and evaluation of a lightweight root cause analysis method (ARCA method) – Field studies at four software companies. Information and Software Technology, 53, 10 (2011), 1045-1061. DOI=10.1016/j.infsof.2011.05.005.

[27] Leszak, M., Perry, D. E. and Stoll, D. A Case Study in Root Cause Defect Analysis. *Proceedings of the 2000 International Conference on Software Engineering.* Institute of Electrical and Electronics Engineers, Los Alamitos, CA, United States, 2000, 428 - 437.

[28] Lethbridge, T. C., Elliott Sim, S. and Singer, J. Studying Software Engineers: Data Collection Techniques for Software Field Studies. Empirical Software Engineering, 10, 3 (2005), 311-341. DOI=10.1007/s10664-005-1290-x.

[29] Livingstone, A. D., Jackson, G. and Priestley, K. Root causes analysis: Literature review. Health & Safety Executive, Contract Research Report, 325(2001), 1-53.

[30] March, S. T. and Smith, G. F. Design and natural science research on information technology. Decision Support Systems, 15 (1995), 251-266.

[31] May, L. J. Major Causes of Software Project Failures. Crosstalk The Journal of Defense Software Engineering, 11, 7 (1998), 9-12.

[32] Mays, R. G. Applications of Defect Prevention in Software Development. IEEE Journal on Selected Areas in Communications, 8, 2 (February 1990), 164-168. DOI=10.1109/49.46867.

[33] Moløkken-Østvold, K. and Jørgensen, M. A Comparison of Software Project Overruns - Flexible versus Sequential Development Models. IEEE Transactions on Software Engineering, 31, 9 (2005), 754-766.

[34] Moløkken-Østvold, K. and Jørgensen, M. A Review of Survey on Software Effort Estimation. In Anonymous *Proc. 2003 ACM-IEEE Int'l Symp. Empirical Software Eng. (ISESE 2003).* 2003, 220-230.

[35] Rooney, J. J. and Vanden Heuvel, L. N. Collecting Data for Root Cause Analysis. Quality Progress, 36, 11 (November 2003), 104.

[36] Rooney, J. J. and Vanden Heuvel, L. N. Root cause analysis for beginners. Quality Progress, 37, 7 (August 2004), 45 - 53.

[37] Runeson, P. and Höst, M. Guidelines for conducting and reporting case study research in software engineering.

Empirical Software Engineering, 14 (19 December 2008), 131-164. DOI=10.1007/s10664-008-9102-8.

[38] Shull, F., Sjøberg, D. I. K. and Singer, J. *Guide to Advanced Empirical Software Engineering.* Springer-Verlag London Limited, 2008.

[39] Siekkinen, M., Urvoy-Keller, G., Biersack, E. W. and Collange, D. A root cause analysis toolkit for TCP. Computer Networks, 52 (2008), 1846-1858. DOI=10.1016/j.comnet.2008.03.005.

[40] Stålhane, T. Root Cause Analysis and Gap Analysis - A Tale of Two Methods. *EuroSPI 2004.* (Trondheim, Norway). Springer Verlag, Berlin Heidelberg, 2004, 150 - 160.

[41] Stevenson, W. J. ed. *Operations Management.* McGraw-Hill/Irwin, New York, 2005.

[42] Traeger, A., Deras, I. and Zadok, E. DARC: Dynamic Analysis of Root Causes of Latency Distributions. *SIGMETRICS '08.* (Annapolis, Maryland, USA, 2. June). ACM, 2008, 277-288.

[43] Verner, J., Sampson, J. and Cerpa, N. What factors lead to software project failure? *Proceedings of Research Challenges in Information Science (RCIS 2008).* 2008, 71-80.

[44] Yin, R. K. ed. *Case Study Research: Design and Methods.* SAGE Publications, United States of America, 1994.