A Tool Supporting Root Cause Analysis for Synchronous Retrospectives in Distributed Software Teams

Timo O.A. Lehtinen, Risto Virtanen, Juha O. Viljanen, Mika V. Mäntylä, Casper Lassenius
Department of Computer Science and Engineering
Aalto University School of Science
Helsinki, Finland
P.O. BOX 19210, FI-00076, Aalto, Finland
[timo.o.lehtinen, risto.virtanen, juha.o.viljanen, mika.mantyla, casper.lassenius]@aalto.fi

Pre-print accepted for publication in Information and Software Technology
http://dx.doi.org/10.1016/j.infsof.2014.01.004

Abstract

Context: Root cause analysis (RCA) is a useful practice for software project retrospectives, and is typically carried out in synchronous collocated face-to-face meetings. Conducting RCA with distributed teams is challenging, as face-to-face meetings are infeasible. Lack of adequate real-time tool support exacerbates this problem. Furthermore, there are no empirical studies on using RCA in synchronous retrospectives of geographically distributed teams.

Objective: This paper presents a real-time cloud-based software tool (ARCA-tool) we developed to support RCA in distributed teams and its initial empirical evaluation. The feasibility of using RCA with distributed teams is also evaluated.

Method: We compared our tool with 35 existing RCA software tools. We conducted field studies of four distributed agile software teams at two international software product companies. The teams conducted RCA collaboratively in synchronous retrospective meetings by using the tool we developed. We collected the data using observations, interviews and questionnaires.

Results: Comparison revealed that none of the existing 35 tools matched all the features of our ARCA-tool. The team members found ARCA-tool to be an essential part of their distributed retrospectives. They considered the software as efficient and very easy to learn and use. Additionally, the team members perceived RCA to be a vital part of the retrospectives. In contrast to the prior retrospective practices of the teams, the introduced RCA method was evaluated as efficient and easy to use.

Conclusion: RCA is a useful practice in synchronous distributed retrospectives. However, it requires software tool support for enabling real-time view and co-creation of a cause-effect diagram. ARCA-tool supports synchronous RCA, and includes support for logging problems and causes, problem prioritization, cause-effect diagramming, and logging of process improvement proposals. It enables conducting RCA in distributed retrospectives.

Key words: ARCA-tool, Root Cause Analysis, Distributed Retrospective, Global Software Engineering

1. Introduction

Retrospectives, also known as post-mortems, are activities where the team members share experiences about problems and their causes [1], analyzing a recently ended project and/or iteration. Root Cause Analysis (RCA) is a structured investigation of a problem to detect which underlying causes need to be solved [2], and a useful practice for retrospectives [3-5]. Retrospectives are typically conducted in face-to-face meetings, in which the team members first identify problems that occurred. Subsequently, they conduct lightweight RCA by collaboratively creating a cause-effect diagram visualizing the causes of problems [5].

Global software engineering, employing geographically distributed teams, has become a standard way of operating in today's business [6]. This way of working creates new challenges related to geographical, temporal, cultural and organizational distance [7]. The use of distributed teams also creates a major challenge for conducting team retrospectives [8]. In previous work, we developed a lightweight focus group based RCA method, ARCA, and evaluated it in four industrial field studies using collocated teams [9]. Even though the method was well liked, the companies pointed out the need to conduct RCA with their distributed teams. Literature on distributed retrospectives identifies a similar need and discusses the use of a combination of email, spreadsheets and an online audio bridge to help facilitate the retrospectives [8]. However, relying on such tools in focus group based synchronous RCA is not feasible, as organizing and interpreting a high number of causes using emails and spreadsheets would be highly difficult. Instead, cause-effect diagrams [9] supporting real-time online environment should be used in distributed retrospectives.

There are many proprietary software tools for RCA.[1] However, we have not succeeded in finding a web-based tool that fulfills the needs of conducting lightweight RCA in synchronous distributed software project retrospectives. First, the tool should make it possible for RCA participants to co-create a cause-effect diagram [5, 9], which stays in-sync between the sites. Second, the tool should allow the development of process improvement ideas for the causes and maintain links between the improvement ideas and the detected causes [10-14]. Third, the tool should make it possible to vote on the most severe causes and best improvement ideas [9]. Fourth, the tool should also make it possible to capture and refine the findings of several retrospectives, in order to support organizational learning and knowledge management [3]. To the authors' best knowledge[2] the most frequently lacking feature of current software tools for RCA is the syncing mechanisms needed for simultaneous co-creation of cause-effect diagrams, see Table 1. There are tools for simultaneous graph drawing, e.g., Google Docs drawings [15], but these tools lack features to support RCA, e.g. automatically capturing and refining the findings of retrospectives.

Furthermore, to our knowledge, there are no empirical studies on the feasibility of using RCA in synchronous distributed retrospectives. While there is ample evidence for the benefits of RCA to detect the causes of problems and make improvements in various contexts [9-13, 16-21], the existing studies have been conducted in a face-to-face context. Thus, in order to contribute to the existing studies, we developed an online tool for supporting synchronous RCA in distributed software project retrospectives called ARCA-tool[3]. It provides features for distributed RCA, idea development, and capturing the lessons learned in many retrospectives.

The goals of this paper are *to present ARCA-tool including its technology and main features, and to provide an empirical evaluation of the tool and synchronous RCA in the context of industrial software development with agile teams*. In order to evaluate the usefulness of RCA and ARCA-tool, we used interviews, questionnaires, and observations in the retrospectives of geographically distributed industrial software teams, that followed the Scrum methodology [22]. Our research questions were:

RQ1: Is ARCA-tool perceived as useful in the distributed retrospectives of agile software teams?

RQ2: Is ARCA-tool perceived as easy to use in the distributed retrospectives of agile software teams?

RQ3: Is RCA perceived as a good approach to use in the distributed retrospectives of agile software teams?

While the first two questions are related directly to ARCA-tool, we evaluate the RCA method, since the evaluators might have difficulty separating the effect of the tool and the context in which it was applied, i.e. the synchronous retrospective method used and the company context. Naturally, ARCA-tool can be used without the retrospective with the RCA method and vice versa.

The rest of the paper is structured in the following way. Section 2 covers the related work and identifies a gap in research, which is then filled by introducing ARCA-tool in Section 3. Section 4 explains the field study method used to evaluate the tool in real industrial contexts and the results of this evaluation are given in Section 5. Finally, Section 6 contains the discussion and Section 7 provides conclusions and directions for further work.

---

[1] http://open-tube.com/10-best-software-tools-to-conduct-root-cause-analysis-and-solve-complex-problems/

[2] Investigation of proprietary RCA tools is difficult as freely available information of the tools is limited

[3] http:// wirca.soberit.hut.fi/prod/?language=en

2. Related Work

In this section, we introduce the concept of software project retrospectives and present problems related to conducting RCA with distributed software teams. We also compare RCA software tools that we have found.

2.1 Software project retrospectives

The key for effective problem prevention is controlling the causes of problems [23]. It is claimed that problems cannot be solved without solving their causes [9]. Retrospectives are one means to help identify and prevent the reoccurrence of problems that have occurred in prior projects [8, 24-26].

In retrospectives, the team members share their experiences about problems and their causes [4, 5, 24]. Retrospectives enable learning at the individual, team, and organizational level. At the individual level, learning is based on shared experiences [27]. Thus, at the team level, learning is related to the shared experiences among the team members [27]. Furthermore, learning at the organizational level requires knowledge management, i.e. the shared experiences are captured and refined, and thereafter distributed to the teams [3]. Therefore, the output of retrospectives must be captured and refined.

A software project retrospective can be viewed as a step-by-step process [5, 28]. In the first step, problems related to the past project, iteration, or milestone are identified. Thereafter, the participants collaboratively identify the causes of the problems by using RCA. In RCA, the causes of problems are identified by constantly asking "why" for every cause [9]. The causes are visualized by using a cause-effect diagram, e.g., a fishbone diagram [5, 14, 19], or a directed graph [5, 9]. The diagram represents the cause-and-effect relationships between the causes of problems. It aims to assist the participants to detect underlying causes for the problems. After the cause-effect diagram is finalized, the participants detect the root causes, defined as the underlying and controllable causes of the problem [9]. Process improvement ideas are then developed for the selected root causes.

While the traditional use of retrospectives has been fraught with problems [25], modern agile development processes, such as Scrum [22], have made the practice common in modern organizations. As such, Scrum or other agile development processes do not require the use of RCA as part of their retrospectives – however RCA can well be used in Scrum retrospectives as a practice that helps add both structure and provides additional value to the teams.

2.2 Root cause analysis and distributed retrospectives

The issue of distributed team members has been considered as the greatest challenge that organizations face while conducting retrospectives [8]. Retrospectives should be lightweight [28] but under the influence of budget constraints and time pressure, they are rarely conducted [25]. While the project members are geographically dispersed, arranging face-to-face retrospectives requires too much effort. Conducting face-to-face retrospectives in such settings is often cumbersome. Distributed retrospectives are introduced as substitutes for face-to-face retrospectives [8]. Such retrospectives are typically conducted with the aid of an audio or video bridge [8]. Logically, in distributed software projects, conducting distributed retrospectives require less effort than conducting them face-to-face due to decreased traveling time.

Conducting RCA in distributed retrospectives is difficult as it requires tools that are not yet mature enough. It has been claimed that a combination of emails, spreadsheets, and an audio bridge are enough to support distributed retrospectives [8]. However, in software projects, conducting RCA with spreadsheets is difficult [9]. This is because of the high number of detected causes [9, 11-13]. For example, in our previous work, four software product companies conducted two hour RCA workshops (similar to retrospectives) each and 80 to 135 causes of software project problems were found in each workshop [9]. The causes were spread over various process areas [29] and had complex cause-and-effect relationships to one another.

Several tools for distributed software development exist [30-32]. The tool types that are the most similar to ARCA-tool are collaborative modeling tools [30] that allow collaborative and distributed software modeling. However, the main goal of those tools is software design modeling, while our tool is focused on RCA cause-effect diagram modeling. Additionally, knowledge management tools [30, 31] include knowledge sharing features, which ARCA-tool also provides. Furthermore, our tool reduces – but does not replace – the need for the use of other communication tools, e.g., a chat, as the cause-effect diagram is constantly updated to all participants, which helps group awareness. Our tool also has similarity with virtual whiteboards, such as Google Docs drawings [15], but our tool has more specific features for cause-effect diagramming and the development of process improvement ideas. Additionally, none of the virtual whiteboards offers support for capturing and refining the shared experiences from the retrospectives of many teams. Based on the literature it seems that it would be possible to combine the existing collaborative tools for performing the same tasks as with our ARCA-tool. However, this would require switching between tools and require cumbersome copy-pasting (from the original cause-effect diagrams to some separate list of process improvement targets and ideas) between different tools.

2.3 Comparison of root cause analysis software tools

Software tools that support RCA in synchronous distributed retrospectives are rare. We searched RCA software tools from Google, Sourceforge, Google Scholar, and Scopus. We found a total of 35 tools and compared their features with ARCA-tool (see Section 3).

We searched for existing root cause analysis software in Google using two search strings: <“root cause analysis software”> and <“root cause analysis software” free>. The first search string resulted in 404,000 estimated hits. Thus, it appears the topic is of high interest. For both search strings, we included all software tools that we found from search result pages until there was a search result page which did not extend the found tools any further (10 hits + adds of the search result page). The number of search result pages was eight for the first and two for the second search string. With this limitation, our search resulted in 24 unique tools for RCA. We applied this limitation in order to complete our search within reasonable time.

Searching for the tools from Google also revealed two additional websites that summarize software tools for RCA[4]. We also included these tools in the evaluation. The websites revealed 17 different software tools for RCA. However, 8 tools were already found in Google. Thus, we were left with 33 unique tools for RCA.

We also searched the sourceforge.com database with search string “root cause analysis” and found one open source alternative that claimed to support root cause analysis (DecisionTreeExpert). Unfortunately, there was no guidance on how to use the tool and we were unable to see how the tool could be used for RCA, and thus excluded it from the comparison.

Furthermore, we searched academic works from Google Scholar and Scopus with the search string “root cause analysis software”. Google Scholar resulted in 58 articles and Scopus resulted in 37 articles. For each article, we read its heading, abstract, key words, and skimmed the content of paper. If the article indicated that a tool for RCA is introduced, we selected the article for further evaluation. Six articles were selected from Google Scholar and three articles were selected from Scopus. The selected articles were thereafter read. One article from Google Scholar [33] and two articles from Scopus [34, 35] introduced a software tool for RCA. Two of these articles introduced a tool that we had already found (Lassale and REASON) from non-academic databases. Finally, we decided to make a comparison to Google Docs drawings, an online collaborative graph drawing tool. Thus, we had 35 existing software tools that we compared with ARCA-tool.

We made our comparison based on the material freely available to us. The sources of information included demonstration videos, free trial versions, marketing material and other available documentation as the majority of the tools were proprietary.

The features that we compared cover seven aspects important for conducting synchronous distributed software project retrospectives. We introduce these aspects below and present analytical arguments for them based on our experience in conducting industrial RCA sessions [9] and prior literature on software project retrospectives [4, 5], and organizational learning systems [36]. The comparison is summarized in Table 1 while further details of the comparison are in Appendix 1.

First, we argue that web browser based software outperforms native client software in the ease of adoption. The software teams rarely have time to conduct retrospectives [25] and therefore the ease of adoption is an important aspect. Native client software requires installation whereas web browser based software can be immediately used. Furthermore, people can use web browser based software with computers having a different operating system and hardware including tablets and smart phones. This is the case unless the web browser based software requires plugins that only work on certain systems, e.g., the flash plugin. Web browser based software can also be used from home computers that might not have the native client software pre-installed or might lack the required licenses. Thus, web browser based clients make organizing retrospectives more lightweight and hassle free. Four of the existing tools are used with a web browser, see Table 1.

Second, in order to conduct distributed synchronous retrospectives similarly to collocated retrospectives [4, 5] the RCA software tool needs to support real-time collaboration among all participants. This means that the RCA software outcome stays in sync between the different sites. Additionally, all team members should be able to contribute to the analysis as it takes place. Therefore, all clients need to have synchronous editing access to the

---

[4] http://open-tube.com/10-best-software-tools-to-conduct-root-cause-analysis-and-solve-complex-problems
http://www.rootcauselive.com/library/Software.htm

analysis results. We see that push-pull technology is needed to implement such requirements as it removes the need for clients to constantly reload their view. Only six of the existing tools fully support real-time collaboration.

Third, co-creation of a cause-effect diagram is at the core of RCA in retrospectives, as introduced in [4, 5, 9]. Using the cause-effect diagram helps the team members to understand and explain a complex problem in terms of its causes, sub-causes, and causal relationships. The majority of the RCA software tools enable creating the cause-effect diagram. Considering the structure of the cause-effect diagram, only three of the existing tools support drawing a graph, while the majority of the tools support tree based cause-effect diagrams. A graph structure has been claimed as more efficient for software project retrospectives than the tree structure [5].

Fourth, RCA aims to develop process improvement ideas for the causes of problems [9]. Thus, the RCA software tool should make it possible to develop and link improvement ideas to the identified causes of problems. Such features are supported by the majority of the tools.

Fifth, it is important that the team members can vote on the most severe causes and best improvement ideas [9]. This is important if a high number of causes and improvement ideas are detected [9]. The team members can focus on the causes perceived as the most severe. Similarly, they can decide collaboratively which improvement ideas should be implemented. Voting is supported only in one of the existing tools.

Sixth, the RCA software tool should support knowledge management, which is about creating "learning organization" [4]. Dingsøyr presents that retrospectives are "a method for leveraging knowledge from the individual level to the organizational level" [4]. Lee et al. [36] present that organizational learning system should include "global knowledge base" that combines "cognitive maps" (cause-effect diagrams of experiences) created by individuals. Thus, the software tool should include the knowledge base which enables combining the lessons learned from many retrospectives and teams over the years. The majority of the tools support knowledge management and allow accessing past RCA session results.

Seventh, we analyzed the costs of existing tools. One of the tools is under an open source license, two are otherwise free to use, whereas the majority of the tools are subject to a fee.

To summarize, in contrast to the existing RCA software tools, only ARCA-tool covers all of the seven aspects discussed above. However, our analysis was limited as described at the beginning of this section and the evaluation of many tools was challenging due to proprietary licenses and limited access to many commercial tools. Thus, it is possible that software tools with similar features as ARCA-tool exist. In any case, the results of our field study can be used as evidence for the usefulness of any tool that implements these features. Furthermore, the comparison of these 35 prior RCA software tools is the largest according to our knowledge.

Table 1. Comparison of RCA software (for more details see Appendix 1)

	Technical features[a]		**RCA features[a]**				**Costs**
Software	**Client: Browser/Native**	**Real-time collaboration**	**Cause-effect diagram**	**Idea development**	**Voting**	**Knowledge management**	**Costs**
ARCA-tool	Browser	Yes	Graph	Yes	Yes	Yes	Free (MIT)
Google Docs drawings	Browser	Yes	Graph	Yes	-	-	Free to use
TapRooT Enterprice ed.	Both	(Yes)	Tree	Yes	-	Yes	Fee
REASON	Both	-	-	Yes	-	Yes	Fee
XFRACAS	Browser	(Yes)	-	Yes	-	Yes	Fee
RCAT Software	?	?	Tree	?	?	Yes	?
PathMaker	Native	Yes	Tree	Yes	-	Yes	Fee
Cause link	Native	-	Tree	Yes	-	Yes	Fee
Solve	?	?	Tree	?	?	?	?
SIM®	Native	(Yes)	Tree	Yes	?	Yes	Fee
PROACT	Native	Yes	Tree	Yes	?	Yes	Fee
Catalyst	Native	-	-	-	-	-	Free (GPL)
Blackbox	Native	?	Tree	(Yes)	?	Yes	Fee
Investigator 3	Native	?	Tree	Yes	?	Yes	Fee
Track	Native	-	-	-	-	Yes	Fee
Corrective Action	Browser	Yes	-	Yes	-	Yes	Fee
RealityCharting	Both	Yes	Tree	Yes	Yes	Yes	Fee
ABS Cons. Root Cause Map	?	Yes	?	Yes	?	Yes	Fee
RCA Software 5.1	Native	-	Tree	-	-	-	Fee
ThinkReliability Excel Template	Native	-	Tree	Yes	-	-	Free
Enablon IMS	?	?	Tree	Yes	?	Yes	Fee
Smartdraw	Native	-	Tree	-	-	-	Fee
Set-Based Thinking	Native	-	Graph	Yes	?	Yes	Fee
PHRED	(Browser)	(Yes)	Tree	Yes	?	Yes	Fee
BowTieXP	Native	-	Tree	(-)	?	Yes	Fee
FMEA Software	Native	(-)	Tree	Yes	-	Yes	Fee
Systems2win	Native	-	Graph	-	-	-	Fee
iReliability	Browser	-	Tree	Yes	-	(-)	Fee
FMECA Software	Native	-	Tree	(-)	(-)	(Yes)	Fee

Rapid Problem Isolation	Native	-	Tree	-	-	(Yes)	Fee
Lassale [33]	Native	-	Tree	-	-	(-)	?
CA Spectrum	?	?	?	?	?	?	Fee
RootCause	?	-	(-)	Yes	?	(Yes)	Fee
Speechminer	?	?	?	?	?	?	Fee
Root Cause Analyst	(Native)	?	(Tree)	?	?	(-)	Fee
RCA GUI [35]	Native	-	Tree	Yes	-	Yes	?

[a] -=this feature is not available in the software tool, Yes=this feature is available in the software tool, (-)=it is likely that this feature is not available in the software tool, but we were not able to verify that, (Yes)= it is likely that this feature is available in the software tool, but we were not able to verify that, ?=we were not able to find any evidence on the occurrence of this feature, Fee=the software is subject to a fee, free (license)=the software is free, free to use=using the software is free

3. ARCA-tool

This section provides an overview of ARCA-tool. We will discuss how the tool supports distributed retrospectives and the features it includes.

3.1 Overview of ARCA-tool

ARCA-tool is designed to be used when conducting RCA in retrospectives. The tool is open-source (MIT license) and was developed in two subsequent projects on the Aalto University software capstone project course[5] by 15 software engineering students. During the projects, the primary author of this paper acted as the customer and provided the tool requirements. ARCA-tool supports the identification of problems and their causes by providing features particularly suitable for the creation of cause-effect diagrams in software project retrospectives. Among many useful features, the team members can develop process improvement ideas embedded in the detected causes and problems. The tool supports conducting distributed retrospectives, and makes it possible to capture and summarize the findings of a set of retrospectives.

ARCA-tool uses a client-server architecture with push-and-pull technology, i.e., the server and clients transmit and receive messages from one another. The core of the tool is a cloud server. The cloud server ensures that all clients (web browsers) are up-to-date in real-time. This is important during distributed retrospectives as the contribution of team members is immediately visible to the other team members.

3.2 Key features of ARCA-tool

In ARCA-tool, a retrospective facilitator creates a retrospective and shares it with the team members. The team members can join the retrospective from their own computers through a TCP network connection. Thus, the retrospectives do not need to be conducted face-to-face. Additionally, ARCA-tool allows the team members to contribute to the retrospective “before” and “after” the retrospective meeting. This is occasionally important as finding a common time is especially difficult in geographically dispersed projects [8]. However, such approach does not make it possible to ask clarifications about the detected problems from other team members. Then one can only see what the others have found. Respectively, the team members cannot contribute to the findings which are not yet detected. On the other hand, the team members can provide input for others or try to contribute to their findings.

The team members start the retrospective by listing problems that occurred during the unit of analysis, which typically is an iteration [5]. Thereafter, they select problems (which can be done through voting that is supported by the tool or by managerial decision, see “Points” in Figure 1), which are analyzed by using RCA [5]. In order to support RCA, a cause-effect diagram is provided. ARCA-tool uses a directed graph structure to model the cause-and-effect relationships (Figure 1). Such a structure, a cause-effect diagram, has been found to be suitable for software project retrospectives [5, 9]. The team members can enter the problems, the causes of problems and related cause-and-effect relationships to the cause-effect diagram. The tool protects the anonymity of team members.

After the causes are entered, the team members can develop process improvement ideas related to the causes. In ARCA-tool, the team members develop their process improvement ideas for each cause separately. This increases the accuracy of the process improvement ideas as now they are cause specific corrective actions. Additionally, the ideas are visually embedded in the causes. ARCA-tool colors the causes that have correctives actions with a yellow color (see the cause “Lack of commitment” in Figure 1). Embedding is important as it keeps the cause-effect diagram clean and simple. Naturally, for the evaluation of the process improvement ideas, the tool offers a separate view for browsing all or selected improvement suggestions as one list (see Figure 2).

All key features of ARCA-tool are embedded in a radial menu (see Figure 1). The radial menu is activated when a team member selects a cause. Simultaneously, all causes that are directly connected with the cause are emphasized

[5] https://noppa.aalto.fi/noppa/kurssi/t-76.4115/etusivu

(see the edges connected with the cause “Project members do not meet enough” in Figure 1). The key features are, starting from the one o’clock position, and proceeding in counterclockwise order.

- Thumb up = Vote for this cause
- Pencil = Edit this cause
- Trashcan = Delete this cause
- Light bulb = Create process improvement idea
- Arrow left = Link this cause to another existing cause
- + sign = Create a cause that is linked to this cause
- Ticket = Classify this cause

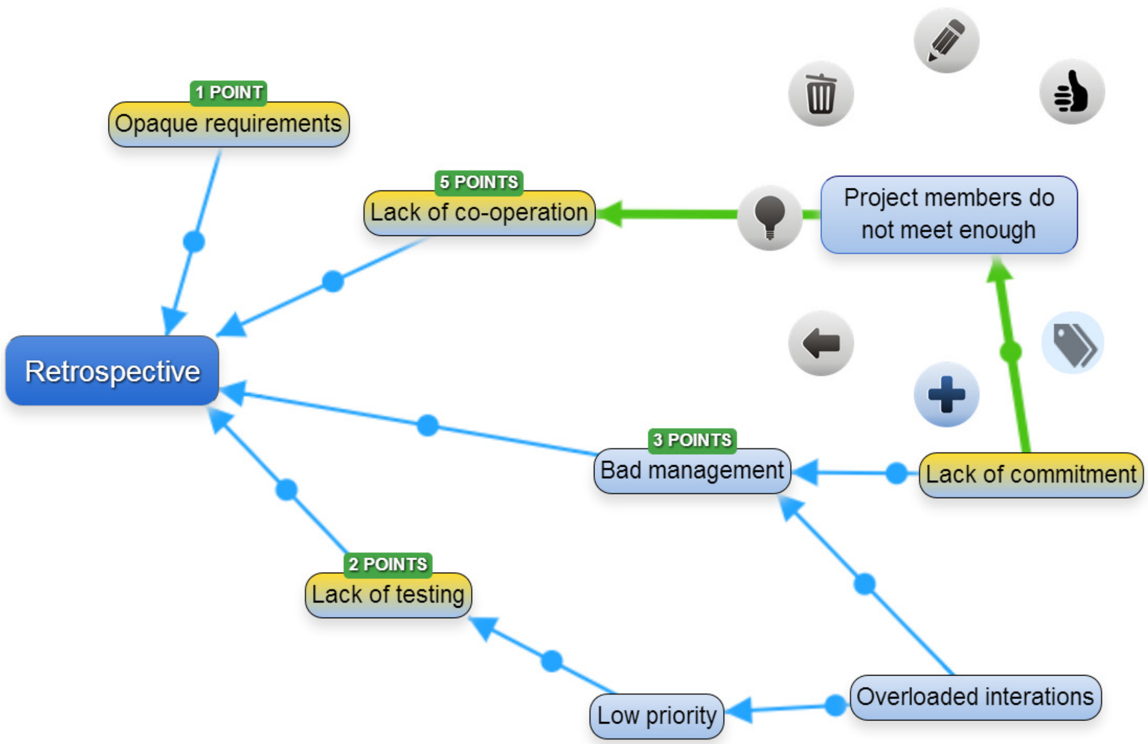

Figure 1. Screen view of ARCA-tool

Export CSV

Corrective actions	Cause	Related case	Status	Likes	Actions
1	Lack of co-operation	To the ARCA paper	Detected	5	-1 +1
0	Bad management	To the ARCA paper	Detected	3	-1 +1
2	Lack of testing	To the ARCA paper	Detected	2	-1 +1
Hire more testers We lack dedicated testers			Idea	1	-1 +1
Motivate developers to unit-test Make sure developers understand that unit-testing is important			Idea	0	+1
1	Opaque requirements	To the ARCA paper	Detected	1	-1 +1
1	Lack of commitment	To the ARCA paper	Detected	0	+1
Daily Scrums Let's keep a daily meeting where all relevant challenges are shortly presented.			Idea	0	+1
0	Retrospective	To the ARCA paper	Detected	0	+1
0	Project members do not meet enough	To the ARCA paper	Detected	0	+1
0	Low priority	To the ARCA paper	Detected	0	+1
0	Overloaded interations	To the ARCA paper	Detected	0	+1

Figure 2. Monitoring view of ARCA-tool showing the causes of Figure 1 and their improvement ideas

3.3 Additional features of ARCA-tool

Voting is occasionally used in retrospectives to focus the attention of the team members to specific problems or causes. Voting is also used to indicate process improvement ideas the team members value the most [9]. In ARCA-tool, the team members can “like” or “dislike” the causes and process improvement ideas (see the “Points” and the thumbnail icon in the radial menu in Figure 1). The amount of likes and dislikes is limited to +/- 1 for the team members while being unlimited for the retrospective facilitator. This way the causes and developed process improvement ideas can be voted on by the team members and emphasized by the facilitator.

Classification of the causes of problems has been used to improve learning and to draw conclusions from detailed and high-volume observations made during RCA, e.g. [10, 12, 13]. In ARCA-tool, the classification can be done during or after the causes are entered in the cause-effect diagram. The tool provides two dimensions for classifying the causes. The pre-existing classification dimensions are the *process areas* and *types* of causes [29]. The process areas express in which parts of the software process the causes occur, whereas the types of causes explain what the causes are. In ARCA-tool, the team members can develop a retrospective specific classification or utilize the classifications used in their prior retrospectives. The tool also provides statistics about the classifications made during the retrospectives. For example, the team members can view the distributions of the detected causes (see Figure 3). They can also view the distributions of liked causes, and causes that include process improvement ideas. The team members can also view the cause-and-effect relationships between the process areas.

In order to support organizational learning, ARCA-tool provides features for monitoring the output of retrospectives, i.e., the causes and process improvement ideas. The tool enables the analysis of an individual retrospective as well as the combination of many retrospectives. This can be highly useful while capturing and refining the lessons learned from many retrospectives. The team members can view the output of all retrospectives they have participated in. The status of the detected causes (*detected, elimination, won’t fix, fixed*) and developed process improvement ideas (*idea, will be implemented, implemented, rejected*) can also be managed. Additionally,

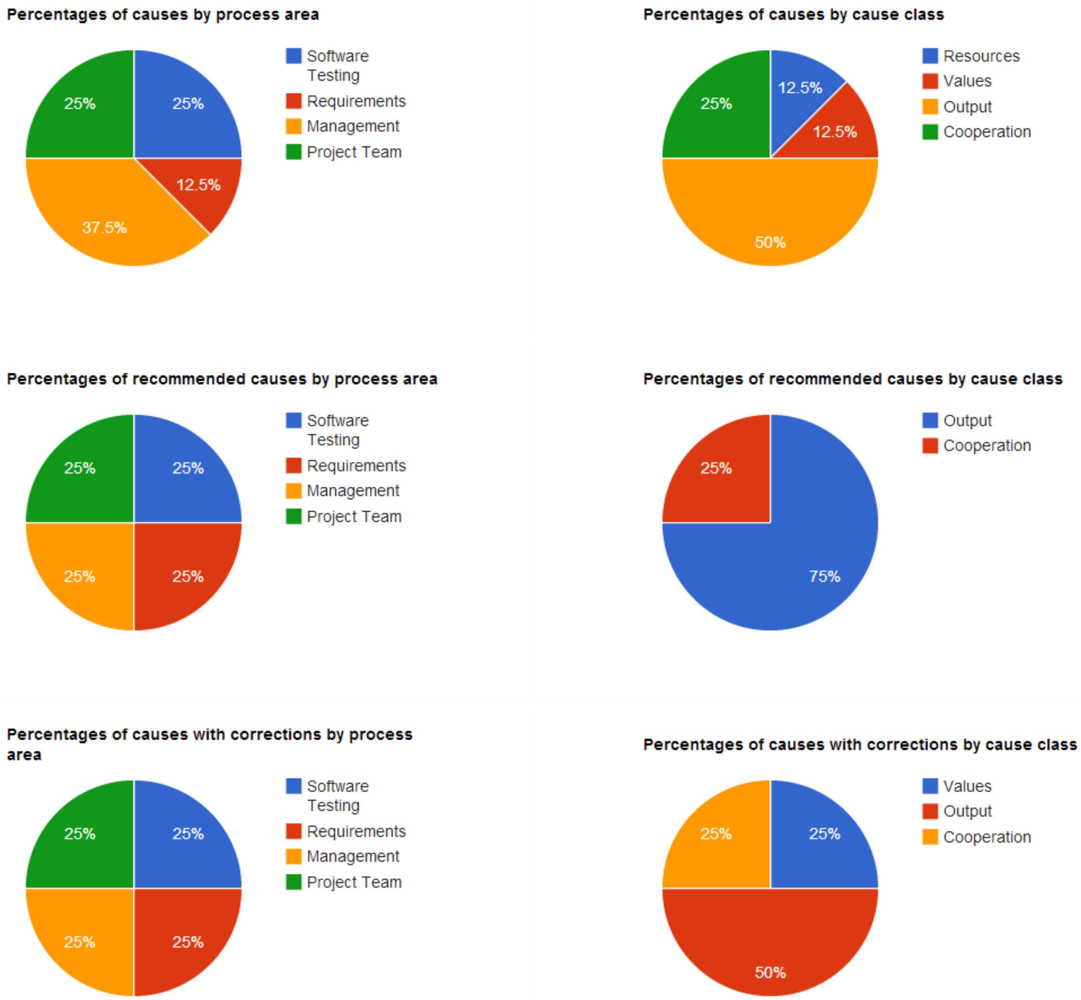

Figure 3. Pie chart view of ARCA-tool presenting the distributions of classified causes shown in Figure 1

the tool provides information about the classified causes. For example, senior managers would like to know what process areas are most often related to the problems analyzed in the retrospectives, see Figure 3. They would also like to know what types of causes are usual in those process areas. In ARCA-tool, the cause-and-effect relationships between the classifications can be automatically visualized for the selected retrospectives. Additionally, the tool provides detailed statistics about the distributions of cause types in process areas. Furthermore, the team members can download a file which includes the detected causes and process improvement ideas from the monitored retrospectives. Thus, the team members can use ARCA-tool to analyze the detailed issues processed in the prior retrospectives and communicate the lessons learned to others.

4. Field study methodology

For the empirical evaluation of ARCA-tool, we used a field study method [37] that allowed us to study the adoption and use of the tool in a real industrial setting. We observed and video recorded four retrospectives conducted by four teams in two companies. After the retrospectives, all participants completed a questionnaire, and selected case participants were interviewed. Thus, we present a rich data set from four industrial software teams, but in contrast to a controlled experiment, we cannot present meaningful statistical comparisons, due to the low number of subjects and the lack of a control group providing an independent baseline for which we could compare our measures. This section presents the research method and context in more detail. The case companies are introduced in Section 4.1 and the retrospective method including the usage of ARCA-tool in Section 4.2. The data collection and analysis methods are shown in sections 4.3 and 4.4.

4.1 Case companies

The empirical part of this study was conducted in two software product companies, as summarized in Table 2. The rationale for the selection of these two case sites was that together they formed an interesting research setting allowing us to evaluate an industrially relevant retrospective method and software tool in collocated and distributed retrospectives. The similarities between the cases made them more comparable whereas the dissimilarities allowed us to evaluate the retrospective method and software tool in different case domains.

The retrospectives of both cases followed a similar retrospective method and each retrospective was computer facilitated by ARCA-tool. The cases were also similar considering the number of retrospective participants, and effort used in the retrospectives. The roles of the case participants were also somewhat similar. Additionally, both cases were conducted in distributed agile software development organizations. Two important differences between the cases were present. First, in Case 1, the used retrospective method was their current method. Instead, the retrospective method was new in Case 2. Similarly, in Case 1, ARCA-tool was used in retrospectives previously. Instead, in Case 2, it was introduced the first time. Second, the participants of Case 1 were experienced with collocated retrospectives, which they used in this study, too. Instead, the case participants in Case 2 were experienced with distributed retrospectives and they followed that approach, respectively. Therefore, we characterize the retrospectives of Case 1 as *collocated* whereas the retrospective of Case 2 was *distributed*. The cases were also different considering the company size and specific target problems analyzed in the retrospectives.

Table 2. Summary of the company cases

	Case 1	**Case 2**
Case company	Software company with > 800 employees	Software company with > 100 employees
SW development organization	Agile with>30 employees	Agile with>70 employees
Case participants	Product owners, scrum masters, architects, and developers. N = 3+ 5 + 3 = 11	Scrum master, architects and developers. N = 5
Evaluation perspective	Evaluation of the *current* method and tool	Evaluation of a *new* method and tool
Retrospective(s)	3 x Collocated	1 x Distributed
Distribution	All persons in a meeting room in Finland	1 person in Romania + 2 in the office in Finland + 2 at home.
Effort Used	1h meeting + 3 x 1h retrospective (3 teams)	1h meeting + 1h retrospective (1 team)
Target problem(s)	*Expectations of product owners do not meet the output of scrum teams*	1) *Lack of pair programming* 2) *Lack of merging the code* 3) *Lack of collaboration*
Causes Found	(23) + (20) + (39) = **82**	(20 + 24 + 15) = **59**

4.1.1 Case 1

Case 1 was conducted in a large-sized international software product company with over 800 employees. The products are highly complex software systems integrated into customized hardware provided by the company partners and to third party software modules. There are around 30 employees working for the core product of the company. The rest of the employees work in localization, integration, customer services and sales. Our study context, the software development organization of the core product is divided into two development teams, which are geographically distributed over several European countries.

The organization follows agile software development practices, based on the Scrum methodology [22]. The development work is divided into sprints each lasting two weeks. In order to facilitate continuous improvement, the Scrum teams conduct 60 minutes face-to-face retrospectives regularly. These are conducted at the same time as the sprint demonstration and the planning of the upcoming sprint. The teams have found using RCA and ARCA-tool in the retrospectives to be useful. The retrospectives are conducted with the following procedure. The team members start by listing positive and negative experiences with ARCA-tool. Then they conduct RCA for some of the voted negative experiences. During RCA, the team members first list underlying causes to ARCA-tool. Then they discuss the findings and try to detect deeper level causes. Corrective actions are developed either during or after the retrospectives for the selected root causes. The problem of the current practice has been the fact that the team members have been forced to travel to the same physical location regularly, a challenge for many team members. In order to reduce the need for travelling, distributed retrospectives have been considered as a substitute.

We conducted our field study in the context of three teams, two distributed development teams, and one product owner team. In both development teams, members include approximately five software developers (software developers and architects) and one scrum master (team leader). The work of the teams is overseen by several product owners (business and product managers). The product owner team had three members, all product owners, representing the needs of customers in different countries. Each product owner is responsible for steering the customer needs to both development teams. The knowledge sharing between the development teams and product owners occurs mainly in the sprint planning sessions. It is assumed that all needed information about the customer needs is communicated during the sessions. However, the developers can ask for the product owners to give clarifications to the customer needs during the sprints.

We were invited to observe the retrospectives of these three teams. The goal of the retrospectives was to analyze why the expectations of the product owners did not meet the output of the development work. The goal was defined by two software development managers before the retrospectives (see Section 4.2).

4.1.2 Case 2

Case 2 was conducted in a medium-sized international software product company with over 100 employees. The company products are large and complex software systems released four times a year. The software development organization includes approximately 70 people. The development work is divided into seven teams, each including about ten people. The team members are geographically distributed over several countries in Europa and Asia.

Like Case 1, the organization follows agile software development practices, based upon the Scrum methodology [22] and the teams conduct 60 minutes distributed retrospectives regularly. Unlike Case 1, the duration of sprints varies between two and four weeks. Additionally, the retrospectives are conducted in a distributed fashion using an online audio and video bridge. In the retrospectives, problems that have occurred are discussed, and process improvement ideas are developed. The teams do not use RCA in their retrospectives. Instead, the team members discuss positive and negative experiences and try to figure out how to make improvements in their development work activities. The retrospectives are occasionally summarized to the company's intranet pages. The problems of the current practice include informal discussions resulting in unfocused discussions and dominating team members who have spoken over the others. Thus, the team members have considered alternative practices, which may be more feasible for their needs.

Our field study was conducted in a context of one distributed software team including the development roles of scrum master, software developers, and architects. We observed a distributed retrospective meeting, where the team members used ARCA-tool and the retrospective method which we introduced to them. Three problems were analyzed in the retrospective. The problems were identified in a separate meeting, which was conducted by the team members before the retrospective (see Section 4.2). The first problem was lack of pair programming, which the team members thought was not used enough. The second problem was merging the code between different work branches. The merge status was unclear, additionally; merging was not done often enough. The third problem was lack of collaboration with other teams in the company.

4.2 Retrospective method used in the cases

Each of the retrospectives across both cases was initiated by a separate meeting, where a high-level target problem for each retrospective was defined. The separate meeting lasted approximately 1 hour. The meeting was conducted by the company representatives who wanted to give a specific goal for the retrospective. In Case 1, the representatives included a product owner and scrum master. In Case 2, the representatives included the scrum master and few software developers of the team. In the meeting, the representatives discussed about problems that had occurred in the development work. Based on the discussion, the representatives concluded the goal of the retrospective, i.e., to explain one (Case 1) or several (Case 2) high-level problems (see Table 2). Thereafter, the retrospective was arranged. The retrospective lasted approximately 1 hour, and it was facilitated by a company representative. At the beginning of each retrospective, the facilitator briefly introduced the specific goal of the retrospective for the participants. In Case 2, the facilitator also shortly introduced the retrospective method and ARCA-tool. The used retrospective method is summarized in Figure 4. ARCA-tool was used by all participants in every step of the retrospective. Each retrospective resulted in a cause-effect diagram emphasizing the most important root causes.

In Case 1, three retrospectives were conducted for a single problem. The first two retrospectives were conducted with each development team having participants from all different roles of the development team including the scrum master, developers, and architects. The third one was conducted with the product owners. The facilitator in Case 1 was the scrum master of one development team. The facilitator steered the retrospectives and led the implementation. The retrospectives were conducted face-to-face at the same physical location. Each retrospective was conducted by using the following procedure. First, the participants were given 5 minutes to enter problems related to the target problem in ARCA-tool. At this stage, all participants used their own computers. During the next

15 minutes, each participant explained the problems entered to the tool. The other participants simultaneously commented and discussed the findings. Thereafter, the participants were given 5 minutes to enter underlying causes that explained the detected problems. This was also done simultaneously in ARCA-tool by all participants, working on their own computers. Then, during the next 15 minutes, each participant explained the underlying causes entered to the tool. The other participants commented on and discussed the findings. They also entered additional causes discovered during the discussion. Furthermore, they used the tool to note if some cause explained other causes. At the end of the retrospective, the participants held a summarizing discussion about the problems and causes entered to the tool. They also voted on the most controllable causes by using the liking feature of the tool.

In Case 2, one retrospective was conducted and it was facilitated by the scrum master of the team who steered the retrospective and led the implementation. The retrospective was conducted as distributed with geographically dispersed participants. The participants included all roles of the development team (a scrum master, software developers and architects) and they used ARCA-tool to document and share their findings about problems and related causes, working on their own computers in their own locations in two European countries. Google+ was used as an audio and video bridge. Thus, the participants were able to discuss and see each other. The retrospective followed the same outline as the one in Case 1, see Figure 4.

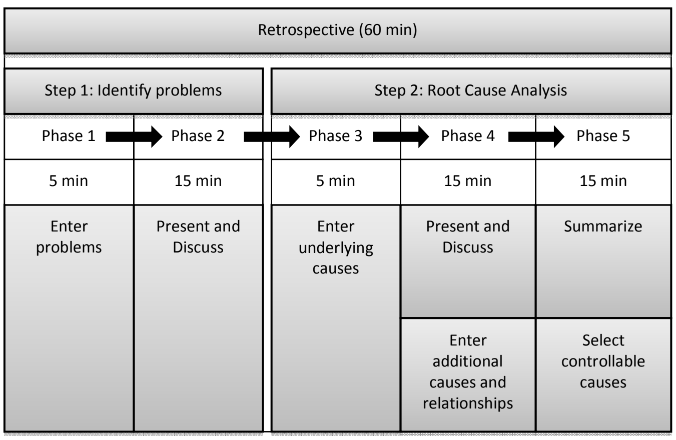

Figure 4. The retrospective method used in the study

4.3 Data collection

The feedback was collected from the case participants using interviews and questionnaires, see Appendix 2 and 3. Additionally, we used observations combined with video recording. The interviews were executed by the 2nd (Case 1) and 3rd (Case 2) author. The primary author observed the interviews. He wrote notes and ensured that the questionnaires were filled in by the case participants. Additionally, the retrospectives were video recorded. Thus, we were able to check if something was missing during the data analysis.

A total of 16 case participants filled in the questionnaires. In addition, we interviewed eight participants. Our aim was to collect feedback about the introduced retrospective method and usefulness and ease of use of ARCA-tool. In Case 1, one participant from each retrospective was interviewed. In Case 2, we interviewed all retrospective participants. Interviews at Case 1 were conducted face-to-face, whereas the interviews at Case 2 were conducted as distributed by using online chat for three participants and face-to-face for two participants. The chat was used in interviews, because it was easier for the interviewees being geographically dispersed. Furthermore, in the questionnaires, the participants of Case 1 evaluated mostly their current retrospective method as the introduced retrospective method was very similar with it. In contrast, in Case 2, the participants compared the introduced retrospective method with their current methods being different than the introduced one. The scale in the questionnaires was a symmetric 5-point Likert scale.

4.4 Data analysis

Both cases were analyzed separately as the questions asked in the questionnaires and interviews varied slightly between the cases. This was due to differences in the company context. Case 1 had used RCA and ARCA-tool previously while Case 2 had not. We transcribed and coded the interviews accordingly. We calculated the means, standard deviations, and medians of the questionnaires. Finally, we summarized the interviews and questionnaires in order to conclude whether the findings were similar between the cases.

We are aware of the controversy of presenting means from a Likert scale. If the interval between the Likert scale items cannot be presumed equal, calculating means with standard deviations is "inappropriate", as stated by Jamieson [38]. In our study, the interval between the Likert scale items can be presumed equal as the scale was symmetric and only the extreme values had a textual representation. In Case 1, the scale was: 1=very minor, 2, 3, 4, 5=very major, and in Case 2, the scale was: 1=very low, 2, 3, 4, 5=very high. Furthermore, mean contains more information in small samples, such as ours, than median, e.g., three responses with values 5, 5, and 1 give the median of 5 but mean of 3.67. The latter is closer to the "truth" because the opinions were highly polarized and the median would only represent the opinion of the middle respondent.

5. Results

In this section, we present the field study results. Feedback from ARCA-tool (see Section 3) is presented in Section 5.1 and the feedback from the retrospective method including the RCA method (see 4.2) is summarized in Section 5.2. Furthermore, Table 3 summarizes the feedback from the questionnaires, and Tables 4 and 5 summarize the results from the interviews. The tables separate the results regarding the research questions. While RQ1 and RQ2 aim to evaluate ARCA- tool, RQ3 evaluates the retrospective method.

5.1 ARCA-tool

To summarize, our results indicate that ARCA-tool increases the cost-efficiency of retrospectives and it is perceived as essential in distributed retrospectives. Additionally, the tool is perceived easy to use and learn. Therefore, we believe that the tool supports the process of the retrospective method (see Section 4.2) and helps the participants to conduct the tasks of retrospectives.

Regarding usefulness, the participants from both cases evaluated in questionnaires (see Table 3) that the tool helped to detect the causes of problems. The participants in Case 1 also evaluated that the retrospective would be less efficient and more difficult without the tool. Respectively, in Case 2, the participants evaluated that the cost efficiency of the retrospective increased with ARCA-tool. Furthermore, the interview results from both cases (see Tables 4 and 5) indicate that the tool is essential in distributed retrospectives. Our results from Case 1 also indicate that when the retrospective is conducted face-to-face, the tool can be substituted with a whiteboard and postIT notes, but in that case the analysis is not as efficient as it is with the tool. According to the interviews at Case 2, ARCA-tool should also be improved. It was said that the tool needs slight improvements while the detected causes are organized. Some participants perceived that currently the tool does not support the visualization of cause groups enough. Perhaps it would be useful to organize similar causes into the same set of causes to be visually represented well on the cause-effect diagram.

Regarding ease of use, the participants from both cases evaluated in questionnaires (see Table 3) that the tool is easy to use and learn. This indicates that ARCA-tool supports the process of retrospective as it helps the participants to conduct the tasks of retrospectives easier, i.e., to detect and analyze the causes of target problems (see Table 2). In Case 1, the ease of use and learning the tool were both evaluated with a very high value. In Case 2, the values were also high, but less than in Case 1. We assume that this was because the tool was new to the participants of Case 2. Furthermore, also the interviews indicate that the tool is easy to use (see Tables 4 and 5). The interviews at Case 1 indicate that the tool makes it easier to visualize the detected causes. Respectively, the participants in Case 2 claimed that the user experience is "intuitive" and the tool is "relatively easy to use". Furthermore, regarding the results from Case 2, there is no feature overload, but all essential features are included in the tool. It was also noted that the difficulty of analysis correlates with the number of causes of problems. The number of detected causes in the retrospectives was around 20 to 59 (see Table 2).

Table 3. Summary of questionnaires

		Case 1 (Collocated)									Case 2 (Distributed)			All Teams [a]			
		ScrumT1 (N=3)			ScrumT2 (N=5)			Product Owners (N=3)			ScrumT3 (N=5)						
		$\bar{x}$	σ	$\tilde{x}$	$\bar{x}$	σ	$\tilde{x}$	$\bar{x}$	σ	$\tilde{x}$	$\bar{x}$	σ	$\tilde{x}$	N	$\bar{x}$	σ	$\tilde{x}$
RQ1 Usefulness of ARCA-tool	Retrospective efficiency without the tool	1.3	0.6	1	2.2	0.8	2	1.3	0.6	1	-	-	-	11	1.7	0.8	2
	Tool's cost efficiency compared with previous practices	-	-	-	-	-	-	-	-	-	4.0	1	4	5	4.0	1.0	4
	Assistance of the tool for cause detection	4.3	0.6	4	4.2	0.8	4	4.7	0.6	5	4.0	0.7	4	16	4.3	0.7	4
	Ability to detect the causes without the tool	3.3	0.6	3	3.2	0.4	3	3.0	1.0	3	3.2	0.4	3	16	3.2	0.5	3
	Retrospective ease of use without the tool	1.7	0.6	2	1.8	0.4	2	1.0	0	1	-	-	-	11	1.5	0.5	2

RQ2 Ease-of-use of ARCA-tool	Easiness to collect causes	3.7	0.6	4	4.2	0.4	4	4.7	0.6	5	4.0	0.7	4	16	4.1	0.6	4
	Easiness to detect root causes	4.0	1.0	4	3.8	0.4	4	4.3	0.6	4	3.0	0.7	3	16	3.7	0.8	4
	Ease of use of the tool	5.0	0	5	4.6	0.5	5	5.0	0	5	4.0	0.7	4	16	4.6	0.6	5
	Learnability of the tool	4.7	0.6	5	4.6	0.5	5	5.0	0	5	4.0	1.0	4	16	4.5	0.7	5
RQ3 Retrospective method	Personal contribution	4.0	0	4	3.8	1.1	4	3.3	0.6	3	3.2	0.4	3	16	3.6	0.7	4
	RCA cost efficiency compared with prior practices	-	-	-	-	-	-	-	-	-	4.2	0.4	4	5	4.2	0.4	4
	RCA ease of use compared with prior practices	-	-	-	-	-	-	-	-	-	4.4	0.5	4	5	4.4	0.5	4
	Correctness of detected causes	4.0	0	4	4.2	0.4	4	4.0	1.0	4	3.8	0.4	4	16	4.0	0.5	4
	Impact of the detected causes	3.7	0.6	4	3.6	1.1	4	4.7	0.6	4	3.8	0.8	4	16	3.9	0.9	4
	Openness in communication	3.0	1.0	3	4.4	0.5	4	5.0	0	5	4.8	0.4	5	16	4.4	0.9	5

a. N=the number of respondents, $\bar{x}$ =mean, σ =standard deviation, $\tilde{x}$ =median, Scale: 1=very minor/low; 2, 3, 4, 5=very major/high

Table 4. Summary of interviews in Case 1

Question	Summary	Quotes from the Interviews
Would we have found the same problems and causes without the tool? (RQ1-2)	Similar causes could have been detected also by using a whiteboard, as an example. However, ARCA-tool improves the efficiency of the analysis. In geographically distributed teams, ARCA-tool is essential.	*"We could have detected the same causes by using a whiteboard, however, ARCA-tool made the analysis easier."* (person 1) *"The required effort by using the whiteboard would be higher"* (person 1) *"ARCA-tool improves the visualization of the detected causes."* (person 2) *"ARCA-tool spares time when documenting the results."* (person 2) *"ARCA-tool is essential when some participants are geographically dispersed."* (person 1)
Did this retrospective method help us to find the causes of the problems? (RQ2-3)	The key to finding the causes was the RCA method. ARCA-tool helped to visualize the causes of the problem. However, the tool itself was not perceived as the key to success.	*"The RCA method helped to found these causes. ARCA-tool itself is not the key to success, but the structured approach of the RCA method is."* (person 1) *"The tool made it easy to see the big picture related to the problem causes. Each team member was additionally able to see what the other participants have detected."* (person 2)
Do you think that we found the most critical problems? (RQ3)	The most critical causes of the target problem were found.	*"We did find the most important root causes"* (person 1) *"We did find the most critical problems"* (person 2) *"I think that we found most of the causes."* (person 3)

Table 5. Summary of interviews in Case 2

Question	Summary	Quotes from the interviews
In contrast to the company practices used to detect the causes of problems, do you consider ARCA-tool as useful? (RQ1)	ARCA-tool improves the company practices. The tool improves the analysis of the causes of problems and their relationships. Additionally, the tool is perceived as enjoyable.	*"It works!"* (person 4) *"The tool improves understanding about the causal relationships between the problems, which I consider as useful."* (person 5) *"I found it very useful *and* fun to do. It certainly is a better practice than having an online video meeting like we had in the past."* (person 6) *"The tool challenges the participants to consider the causes of problems deeper."* (person 7)
Do you consider ARCA-tool as cost efficient when compared with RCA which is conducted by using postIT notes or Google Docs drawings? (RQ1)	ARCA-tool works well with distributed teams. This is because of the online automation and features supporting organizing the causes easily. In contrast to Google Docs drawings, the tool should support the grouping of causes.	*"In our case, the postIT notes do not work at all. This is because of the distributed team members."* (person 4) *"In Google Docs drawings, a lot of time is spent to organize the causes and their relationships"* (person 7) *"Grouping the detected causes with ARCA-tool is currently difficult."* (person 8)
Do you consider ARCA-tool as easy to use when compared with RCA which is conducted by using postIT notes or Google Docs drawings? (RQ2)	ARCA-tool is learnable and intuitive. There is no feature overload either. The layout automation improves usability. On the other hand, when the number of causes increases, the difficulty of the analysis increases.	*"The tool is relatively easy to use and much more flexible than RCA which is conducted by using the postIT notes."* (person 4) *"The user experience was intuitive."* (person 5) *"Layout automation is good."* (person 7) *"Outlining a high number of causes is somewhat difficult."* (person 8)
In contrast to our process improvement practices, do you consider the RCA method as easy to use? (RQ3)	The RCA method fits the retrospectives well. It is learnable, simple, intuitive, and formal.	*"After little practice it definitely helps us to improve the efficiency of the work."* (person 4) *"The RCA method is not difficult to use."* (person 7) *"It is based on intuitive and simple idea"* (person 5) *"Yes, because the RCA method is structural and straight forward"* (person 8)

In contrast to our process improvement practices, do you consider the RCA method as cost efficient? (RQ3)	The RCA method improves current practices by providing deeper analysis with its structural approach. It also improves the collaboration and conceptualization related to the causes of problems.	*"The RCA method works."* (person 4) *"I think that the visualization of the causes is important."*(person 4) *"The method improved the discussions and helped to consider the problem more deeply."* (person 5)

5.2 The retrospective method

Considering the results from the interviews, using RCA in retrospectives was perceived as useful in both cases. This was due to the structured approach that the retrospective method followed and the in-depth analysis which improved collaboration.

In Case 1, the participants said that the structured approach of the RCA method helped to detect the causes of problems. In Case 2, the participants said that the structured approach of the RCA method resulted in deeper understanding about the causes of problems which makes improvement to their current practices. Considering the questionnaires, the participants from both cases evaluated the easiness to collect causes and detect root causes as high (see Table 3). Furthermore, they evaluated that the correctness and impact of the detected causes was high.

The participants of Case 2 perceived that the RCA method improved collaboration. Additionally, they said that the RCA method is easy to use and learn. They explained that the method is based on "an intuitive and simple idea". The results from questionnaires are in line with these results. The openness in communication was evaluated with high values in both cases (see Table 3). Additionally, the participants evaluated their personal contribution with high values.

6. Discussion

In this section, we answer the research questions and discuss our findings and possible threats to the validity of this study.

6.1 Answering the research questions

RQ1: Is ARCA-tool perceived as useful in the distributed retrospectives of agile software teams? In Case 1, ARCA-tool had already been found to be useful in collocated retrospectives. The tool was new to the participants of Case 2, but they were experienced in conducting distributed retrospectives. In order to answer this research question, we use the results from Case 2 and compare them to Case 1. Regarding ARCA-tool we claim the following:

- The tool enables the team members to contribute to the retrospective simultaneously. This improves the communication as the team members can write simultaneously while speaking simultaneously is not possible. This also reduces the risk that the participants forget some important comments if they are not written down.
- The cause-effect diagram structure provided by ARCA-tool improves the way the findings are visualized. This encourages the team members to consider the findings in-depth, as proposed in Case 2 (see Table 5).

Our results support these claims. In both cases, the tool was evaluated as efficient (see Table 3), but in the distributed retrospective of Case 2 (see Table 5), the tool was characterized as essential. Similar comments about distributed retrospectives were also given in the interviews with the participants of Case 1. In Case 1, ARCA-tool was used previously and the participants perceived that they would like to use the tool in their upcoming retrospectives too. Obviously, the tool was found to be useful in face-to-face retrospectives. A comparison of the results from Case 2 to Case 1 indicates that ARCA-tool is also useful in distributed retrospectives. In the distributed retrospective of Case 2, the tool was perceived as useful when it was compared with the current practices (see Table 3). Thus, regarding Case 2, ARCA-tool improves distributed retrospectives where only audio and video bridges are used (see Section 4.1.2).

Furthermore, in collocated retrospectives of Case 1, the participants proposed that the tool made it possible to note what the other participants have found (see Table 4). It was also perceived that the visualization of the causes helped to outline the detected causes. In the distributed retrospective of Case 2, the participants perceived that the visualization of the detected causes is important and the tool helped to organize them (see Table 5). It was also claimed that the tool improved the analysis of the causes of problems and their relationships (see Table 5), probably one of the main advantages of RCA.

To summarize, it seems that ARCA-tool is perceived useful in synchronous distributed retrospectives of small agile software teams. Probably we still need to continue its development by making slight improvements to it (see

Section 5.1). However, the tool improves the contribution of participants and challenges them to consider the findings in-depth.

RQ2: Is ARCA-tool perceived as easy to use in the distributed retrospectives of agile software teams? Considering the ease of use, ARCA-tool was designed to be used in distributed retrospectives [8]. Additionally, we required that it enables conducting RCA [9]. Our aim was not to develop software supporting all kinds of different modeling needs, e.g., making complex software models [30]. Instead, we wanted to make a lightweight tool which is simple and easy to use in a small group of individuals, i.e., less than ten participants use the tool in a synchronous retrospective collaboratively, as introduced in [9].

ARCA-tool was perceived as easy to use in both cases. The number of participants was between three and five. In the collocated retrospectives of Case 1, the participants perceived that the tool made the analysis easier (see Table 4). In the distributed retrospective of Case 2, the participants perceived that the tool is learnable and intuitive (see Table 5). They also appreciated that only the necessary features are included in the tool. Additionally, it was noted in Case 2 that the way the tool automates the cause-and-effect structure improves its usability. Additionally, in the questionnaires, the participants from both cases evaluated the ease of use and learnability of the tool as high (see Table 3). It seems that the participants of Case 1 evaluated the ease of use and learnability with higher values than in Case 2. It is possible that this was due to the fact that the tool was new to the participants of Case 2, whereas the participants of Case 1 were already familiar with it. It is also possible that in distributed retrospectives, the perceived ease of use decreases. The participants are geographically dispersed, and therefore, asking assistance from others becomes more difficult. However, we did not observe such problems in the distributed retrospective of Case 2.

To summarize, it seems that using ARCA-tool in distributed retrospectives does not make a major difference to its ease-of-use in collocated retrospectives. The participants learn using the tool with a short introduction, and during the distributed retrospective they perceive that it is easy to use.

RQ3: Is RCA perceived as a good approach to use in the distributed retrospectives of agile software teams? Both cases resulted in a similar finding. RCA was perceived as a good approach for retrospectives. This conclusion is well in line with prior studies. Problem prevention requires controlling the causes that create the problem. RCA makes it possible to detect the causes of the problem systematically and in-depth. In retrospectives [5, 28], also with distributed settings, RCA helps the team members to consider the causes of their problems. This is important in order to make improvements in the team.

Case 1 has used RCA previously, which indicates that the case organization have already found such an approach as useful in collocated retrospectives. Instead, the RCA approach was new to the participants of Case 2, but they were experienced with distributed retrospectives. In order to answer this research question, we use the results from Case 2 and compare them with Case 1.

Regarding the interviews at Case 1, the key for finding the causes was the RCA method (see Table 4). The participants of Case 1 also perceived that the most critical causes of the target problem were found. Thus, the outcome of RCA was perceived accurate and useful in the collocated retrospectives of Case 1. Similarly, it was proposed in Case 2 that the structural approach of RCA improves their current practices by providing in-depth analysis. It was also noted that the RCA method improves the collaboration and conceptualization of the causes of problems. The participants of Case 2 also evaluated in the questionnaires that the detected causes were correct and their impact was “high” (see Table 3). Additionally, the participants of Case 2 evaluated that in contrast to their current practices the RCA approach is cost-efficient and easy to use (see Table 3).

The core of RCA is the cause-effect diagram. Retrospectives using discussions only are concerned with the problem of it being difficult to remember all relevant findings and outline the findings as a whole. Level of detail and the coverage of the discussions are dependent on human memory. Retrospectives using RCA do not suffer the memory problem as the cause-effect diagram keeps the attention on relevant causes, but simultaneously helps the team members to remember the findings as they are registered to the diagram. In synchronous distributed retrospectives, this means that the cause-effect diagram has to be simultaneously reachable by all distributed team members. Otherwise conducting collaborative RCA would likely be difficult. In the distributed retrospective of Case 2, the RCA method was characterized as learnable, simple, intuitive, and straightforward (see Table 5). The results from the questionnaires are in line with the results from the interviews. The participants evaluated that the retrospective method helped to detect the causes of the target problems (see Table 3).

The retrospectives of Case 1 were collocated and the retrospective of Case 2 was distributed. In both cases, ARCA-tool made the cause-effect diagram reachable for all participants. RCA worked well in the collocated retrospective of Case 1 and in the distributed retrospective of Case 2. There were no major differences in the evaluations of the case participants between the cases either. The empirical results from Case 2 are very similar with the results from Case 1. Thus, to summarize, we conclude that the RCA worked well in the synchronous distributed retrospective of Case 2. However, it required the tool for collaborative cause-effect diagramming.

6.2 Comparison to prior studies

Regarding the scrum methodology [22], retrospectives are valuable and they should be conducted at the end of iterations. Our results are in line with this claim as both of our cases have used retrospectives accordingly and found them useful. Furthermore, the prior studies [5, 28] introduce RCA as a part of retrospectives. Our results consolidate the prior studies by indicating that RCA is an important part of the retrospectives of small agile teams. The retrospective method used in this study is similar to the prior method called “postmortem review” [4] that also includes the step of RCA. Such method has been introduced as lightweight and useful for small software teams [5]. Respectively, Case 1 has used the method previously and found it useful. Furthermore, considering Case 2, their prior practices did not include RCA. They discussed positive and negative experiences and they tried to figure out how to make improvements in their development work activities, as recommended in the scrum methodology [22]. However, they did not create cause-effect diagrams or otherwise registered the causal structures of problems. The problems of the prior practices included informal discussions resulting in unfocused discussions and dominating team members who spoke over the others. When RCA was used in their distributed retrospective (see Section 4.2), the participants perceived that the method was better than their current practices.

Prior work has also been conducted in the area of Group Support System (GSS). GSSs are systems whose main aim is to help individuals to arrive at correct decision in meetings effectively. GSS systems, such as one presented in [39], include features from three dimensions: 1) “communication support,” 2) “process structuring” and 3) “information processing” [40]. The features of communication support help in the information exchange between the participants [40]. The features of process structuring keep the meeting progressing according to the agenda [40]. Furthermore, the features of information processing provide access to important information, and enable sharing, aggregating, structuring, and evaluating the information [40].

The retrospective method together with ARCA-tool fulfills the three dimensions of GSS. Regarding the usefulness and ease-of-use of ARCA-tool, we hypothesize that the tool provides “communication support” [40], especially in distributed retrospectives. The tool improves the information exchange around the problems and their causes. Additionally, the tool includes the features of the parallel communication, the anonymity of participants, and “group memory” [41]. Although our tool does not provide access to internal or external databases, the tool does makes it possible to model the important knowledge of participants through cause-effect diagrams, voting, cause classifications, and corrective actions. Therefore, we see that the tool also provides features for “information processing” [40]. Finally, we hypothesize that the retrospective method provides “process structure” [41] as it includes the rules for communication and process steps that are steered by a facilitator. ARCA-tool also records a cause-effect diagram that is a partial record of the meeting interaction and part of process support.

The prior approach for distributed retrospectives [8], using a combination of emails, spreadsheets, and an audio bridge, does not provide anonymity or parallel information exchange. Sending emails between the participants is not an anonymous approach to exchange information. Furthermore, using spreadsheets does not provide parallel contribution to the outcome of retrospective. All individual spreadsheets need to be combined together. Additionally, describing and analyzing cause-effect relationships with spreadsheets is difficult [9]. Therefore, the distributed retrospectives also require collaborative cause-effect diagrams. Thus, we conclude that the retrospective method combined with ARCA-tool makes an improvement to the approach introduced in the prior work [8]. RCA is an important part of retrospectives and ARCA-tool improves them by providing communication support and information processing.

6.3 Evaluation of the research

This section discusses the validity of our empirical results using a validation scheme presented by [42]. Furthermore, as our results are based on the social construction of case companies, we will also use the evaluation principles of Interpretive Field Studies [43] in the validation scheme.

6.3.1 Construct validity

Construct validity reflects the extent to which the studied operational measures represent what is investigated according to the research questions [42]. The participants represented experts while considering the current practices used in the retrospectives of their teams. Thus, we believe that they were able to compare the introduced retrospective method with their current practices. Additionally, the participants covered most of the organization members, i.e., the organization members of Case 1 and the team members of Case 2. Therefore, we believe that the research data was not biased by a homogenous group of individuals. Instead, various interpretations about the RCA approach and ARCA-tool were captured. This enabled us to draw out multiple interpretations about the study results, an important aspect for validity introduced in [43]. Using interviews and questionnaires were therefore reasonable data collection methods, which increases the construct validity [42]. However, our results are not based on the comparison of the outputs between the previous and the introduced retrospective method, as such information was not available for our purposes. Thus, even though the feedback from all case participants was highly positive, it should be noted that these evaluations are based on perceptions.

Separating the effect of RCA from the use of ARCA-tool was also difficult. In the interviews with both cases and in the questionnaire of Case 2, we asked the participants to evaluate the tool and RCA approach separately. Instead, the questionnaire used in Case 1 asked the participants to evaluate ARCA-tool and the output of RCA thoroughly, but there were no questions about the RCA approach itself. Thus, in Case 1, separating the evaluations of the tool from the evaluations of the RCA approach was difficult. It was based on the interviews only. Therefore, regarding the RCA approach, we were not able to compare the questionnaire results between the cases.

6.3.2 External validity

External validity is concerned with whether it is possible to generalize the findings of the study and to what extent they can be generalized [42]. "Contextualization" has been presented as an important principle for generalizing the study results [43]. Both cases varied and thus they evaluated RCA and ARCA-tool from slightly different perspectives. This increases the external validity [42]. The participants of Case 1 were experienced with the used retrospective method and ARCA-tool, but inexperienced on using it in a geographically distributed setting. Instead, the participants of Case 2 were experienced on conducting retrospectives in a geographically distributed setting, but inexperienced in using the retrospective method and ARCA-tool. The feedback from both cases, however, was very similar. Naturally, the evaluations of the case participants reflected the advances of the introduced retrospective method in comparison with the current practices. If the companies would have previously used the existing RCA software tools, perhaps, the feedback of ARCA-tool would have been different.

Furthermore, we had only two cases in which one fully investigated the intended research questions. All of the retrospectives were conducted at the team level and the number of case participants in each retrospective was between three and five. Four teams were studied. DeSanctis and Gallupe [44] present in the study of group decision support systems that the "nature of technological support" is dependent on three important aspects: "group size," "membership proximity," and "the task confronting the group". Our case contexts included only small groups, but the member proximity covered both extremes "face-to-face" and "dispersed" settings [44]. Furthermore, the tasks the groups confronted included analyses of problems faced at the agile software development organizations and teams. Thus, we cannot generalize our findings to organization wide distributed heavy-weight retrospectives using different RCA methods [9] and a higher number of participants. We can only conclude that our results are likely valid in similar case contexts to ours, i.e., geographically dispersed small agile software teams using retrospectives regularly in order to create continuous learning and improvements (see Section 4.1).

We cannot conclude that the distributed retrospectives can fully substitute the face-to-face retrospectives either. Building trust in global software teams is crucial for success, which requires frequent communication, face-to-face meetings, and socialization [45]. The tool support for distributed retrospectives likely enables conducting retrospectives more frequently, which we assume would improve the communication. However, if the team members communicate on distributed settings only, then the risk for decreased information exchange and feedback increases [45].

Finally, considering the uniqueness of ARCA-tool, the evaluation of the prior RCA tools in Section 2.3 was not complete due to an excessive number of hits to our search strings in Google. However, we used five additional data sources. We studied all the tools listed in the two websites of "useful RCA tools". Additionally, we searched for RCA tools in Sourceforge.com, but did not find any tools suitable for doing RCA. Finally, we searched for RCA tools in Google Scholar and Scopus. The data from these six sources resulted in 35 RCA tools that we compared with our ARCA-tool. In contrast to ARCA-tool, none of the other tools matched all the seven aspects used in our comparison. However, it is still possible that a similar tool to our ARCA-tool exists. Nevertheless, according to the authors' best knowledge our comparison of 35 RCA software tools is the largest one existing.

6.3.3 Reliability

Reliability is concerned with the extent to which data and analysis are dependent on a specific researcher [42]. Klein & Myers [43] state that the social tie between the researchers and participants should be critically reflected in order to evaluate the validity of results. The retrospectives were conducted by the employees of the companies. Thus, it is possible that the case participants overstated the goodness of the retrospective method in the questionnaires and interviews as they conducted the method by themselves. It is also possible that the social tie between the researchers and participants biased the results. We controlled this risk by using triangulation in data collection [46] through observations, video recording, questionnaires, and interviews which increases the reliability of our results. In the observations, we did not note any practical issues during the retrospectives. Additionally, our observations indicate that the case participants truly liked the used retrospective method.

Furthermore, considering the data analysis, there is a slight risk for researcher bias which is a common problem in qualitative data analysis. While the number of interviews increases, summarizing the results becomes challenging as people answer the same questions differently. We controlled this risk by using questionnaires. Similar responses from the questionnaire forms make it unlikely that researcher bias would have had large effect on the qualitative results. Additionally, our conclusions were based on both 1) the analysis of individual parts of research data and 2) the analysis of all research data combined together, the key principle in Interpretive Field Research, called “Hermeneutic Circle” [43]. Our conclusions are also in line with prior literature (see Section 2.1). The approach of RCA has already been introduced as valuable for retrospectives [5, 28]. The prior literature did not “tell the story behind our results”, a threat to validity introduced in [43], but consolidate our findings. Our conclusions are explicitly derived from the results (see Section 5), as can be seen in Section 6.1.

7. Conclusions and future work

This paper proposed a real-time cloud-based tool for solving the problem of being infeasible to conduct collocated retrospectives in geographically distributed software teams [8]. ARCA-tool enables conducting collocated and distributed retrospectives with RCA. The most important feature of the tool is the up-to-date real-time view of the retrospective outcome. Additionally, the tool provides features for the co-creation of cause-effect diagrams, the development of improvement ideas, the voting of the causes and improvement ideas, and support for organizational learning by allowing the data exploration of past retrospectives. Finally, our analysis of 35 prior RCA tools showed that none of the prior tools had all the main features of our ARCA-tool (see Section 2.3).

We evaluated the tool and RCA approach in industrial field studies with four different teams in two different software companies. Although our field study context differed (See Section 4.1), the results are remarkably similar in both contexts. The results indicate that using ARCA-tool in the synchronous collocated and distributed retrospectives of small agile software teams is useful and easy. The tool was perceived as useful and highly easy to use and learn. It was claimed that the tool increases the efficiency of retrospectives and helps to visualize the causes of problems. Additionally, the field study evaluated the use of RCA in synchronous distributed retrospectives. RCA was perceived as highly useful because of its structural approach, which improves collaboration and provides deeper analysis challenging the team members to consider the problems in-depth.

After this case study was conducted, both case organizations have continued using the RCA approach with ARCA-tool in their retrospectives. In addition, Case 1 substituted all of their collocated retrospectives with distributed retrospectives. In the future, we are planning to continue the development and evaluation of ARCA-tool and RCA as other companies have also expressed interest in them. During the first year after the release, ARCA-tool website[6] has had over 63000 page views with 1357 unique visitors (63.5% are returning visitors) with an average visiting time of slightly less than 8 minutes. Our motivation for the development of ARCA-tool was academic, i.e., to develop and evaluate an open source solution, which is freely available for anyone who needs it. Obviously, ARCA-tool also has business potential through SAAS business models. However, replicating studies are needed. The tool and the RCA method should be evaluated with different case contexts including larger group sizes and various target problems. Prior literature on group decision support systems [40, 41, 44] could help to evaluate the tool from various important aspects. ARCA-tool was published under MIT license in order to enable replication studies and future business applications of the tool in a range of settings.

8. Acknowledgements

The authors would like to thank the companies participating in the field studies and software engineering students implementing ARCA-tool, in alphabetical order: Helin Anssi Matti, Hovi Roope, Jaanto Jari, Kekäle Mika, Kere Markus, Koistinen Joona, Laukkanen Eero, Patana Jussi, Rihtniemi Pekka, Saarinen Jerome, Sevenius Toni, Valjus Mikko, and Viitanen Jonne.

References

[1] K. C. Desouza, T. Dingsøyr and Y. Awazu, Experiences with conducting project postmortems: reports versus stories, Software Process Improvement and Practice 10 (2005) 203-215.

[6] http:// wirca.soberit.hut.fi/

[2] R. J. Latino and K. C. Latino, Eds., Root cause analysis: improving performance for bottom-line results. 6000 Broken Sound Parkway NW, Suite 300 Boca Raton, FL 33487-2742: CRC Press, 2006.

[3] T. Dingsøyr, N. B. Moe and Ø. Nytrø, Augmenting experience reports with lightweight postmortem reviews, PROFES '01 Proceedings of the Third International Conference on Product Focused Software Process Improvement, 2001, pp. 167-181.

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

[5] F. O. Bjørnson, A. I. Wang and E. Arisholm, Improving the effectiveness of root cause analysis in post mortem analysis: A controlled experiment, Information and Software Technology 51 (1) (2009) 150 - 161.

[6] J. D. Herbsleb and D. Moitra, Global software development, Software, IEEE 18 (2001) 16-20.

[7] T. Jaanu, M. Paasivaara and C. Lassenius, Near-synchronity and distance: Instant messaging as a medium for global software engineering, 2012 IEEE Seventh International Conference on Global Software Engineering, 2012, pp. 149-153.

[8] J. Terzakis, Virtual retrospectives for geographically dispersed software teams, IEEE Software 28 (2011) 12-15.

[9] 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.

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

[11] M. Leszak, D. E. Perry and D. Stoll, A case study in root cause defect analysis, Proceedings of the 2000 International Conference on Software Engineering, 2000, pp. 428 - 437.

[12] R. B. Grady, Software failure analysis for high-return process improvement decisions, Hewlett-Packard Journal 47 (4) (1996) 15 - 25.

[13] P. Jalote and N. Agrawal, Using defect analysis feedback for improving quality and productivity in iterative software development, Proceedings of the Information Science and Communications Technology (ICICT 2005), 2005, pp. 701 - 714.

[14] B. Andersen and T. Fagerhaug, Eds., Root cause analysis: simplified tools and techniques. United States, Milwaukee 53203: Tony A. William American Society for Quality, Quality Press, 2006.

[15] News and notes from the Google Drive and Docs teams, "Introducing Google Docs drawings," *Docs Blog,* 13 th April 2010.

[16] M. Kalinowski, G. H. Travassos and D. N. Card, Towards a defect prevention based process improvement approach, Proceedings of the 34th EUROMICRO Conference on Software Engineering and Advanced Applications, Parma, Italy, 2008, pp. 199 - 206.

[17] R. G. Mays, Applications of defect prevention in software development, IEEE Journal on Selected Areas in Communications 8 (1990) 164-168.

[18] M. Siekkinen, G. Urvoy-Keller, E. W. Biersack and D. Collange, A root cause analysis toolkit for TCP, Computer Networks (2008) 1846-1858.

[19] T. Stålhane, Root cause analysis and gap analysis - A tale of two methods, EuroSPI 2004, Trondheim, Norway, 2004, pp. 150 - 160.

[20] I. Bhandari, M. Halliday, E. Tarver, D. Brown, J. Chaar and R. Chillarege, A case study of software process improvement during development, IEEE Transactions on Software Engineering 19 (12) (1993) 1157-1170.

[21] Z. X. Jin, J. Hajdukiewicz, G. Ho, D. Chan and Y. Kow, Using root cause data analysis for requirements and knowledge elicitation, International Conference on Engineering Psychology and Cognitive Ergonomics (HCII 2007), Berlin, Germany, 2007, pp. 79 - 88.

[22] K. Schwaber and J. Sutherland, Scrum guide, Scrum Alliance (2011) .

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

[24] F. T. Anbari, E. G. Carayannis and R. J. Voetsch, Post-project reviews as a key project management competence, Technovation 28 (2008) 633-643.

[25] R. L. Glass, Project retrospectives, and why they never happen, IEEE Software 19 (2002) 111-112.

[26] L. Williams, What agile teams think of agile principles, Communications of the ACM 55 (2012) 71-76.

[27] W. F. Boh, S. A. Slaughter and A. J. Espinosa, Learning from experience in software development: A Multilevel Analysis, Management Science 53 (2007) 1315-1331.

[28] T. Stålhane, T. Dingsøyr, G. Hanssen and N. Moe, Post mortem–an assessment of two approaches, Empirical Methods and Studies in Software Engineering (2003) 129-141.

[29] 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 '11 Proc. of the 2011 International Symposium on Empirical Software Engineering and Measurement, 2011, pp. 388-391.

[30] F. Lanubile, C. Ebert, R. Prikladnicki and A. Vizcaíno, Collaboration tools for global software engineering, Software, IEEE 27 (2010) 52-55.

[31] M. Jiménez, M. Piattini and A. Vizcaíno, Challenges and improvements in distributed software development: a systematic review, Advances in Software Engineering 2009 (2009) 3.

[32] F. Q. da Silva, C. Costa, A. C. C. França and R. Prikladinicki, Challenges and solutions in distributed software development project management: A systematic literature review, Global Software Engineering (ICGSE), 2010 5th IEEE International Conference on, 2010, pp. 87-96.

[33] L. Williams, D. Grayson and J. Gosbee, Patient safety—incorporating drawing software into root cause analysis software, Journal of the American Medical Informatics Association 9 (2002) S52-S53.

[34] W. Vantine, K. Benfield, D. Pritts and K. Ballard, Evaluating and incorporating new age software technology for identifying systemic root causes, Joint ESA-NASA Space-Flight Safety Conference, 2002, pp. 369.

[35] L. Huertas-Quintero, P. Conway, D. Segura-Velandia and A. West, Root cause analysis support for quality improvement in electronics manufacturing, Assem. Autom. 31 (2011) 38-46.

[36] S. Lee, J. F. Courtney and R. M. O'Keefe, A System for organizational learning using cognitive maps, Omega, the International Journal of Management Science 20 (1992) 23-36.

[37] T. C. Lethbridge, S. Elliott Sim and J. Singer, Studying software engineers: Data collection techniques for software field studies, Empirical Software Engineering 10 (3) (2005) 311-341.

[38] S. Jamieson, Likert scales: how to (ab) use them, Med. Educ. 38 (2004) 1217-1218.

[39] J. F. Nunamaker, A. R. Dennis, J. S. Valacich, D. Vogel and J. F. George, Electronic meeting systems, Communications of the ACM 34 (1991) 40-61.

[40] I. Zigurs and B. K. Buckland, A theory of task/technology fit and group support systems effectiveness, MIS Quarterly (1998) 313-334.

[41] A. R. Dennis, C. K. Tyran, D. R. Vogel and J. F. Nunamaker Jr, Group support systems for strategic planning, J. Manage. Inf. Syst. 14 (1997) 155-184.

[42] P. Runeson and M. Höst, Guidelines for conducting and reporting case study research in software engineering, Empirical Software Engineering (14) (2008) 131-164.

[43] H. K. Klein and M. D. Myers, A set of principles for conducting and evaluating interpretive field studies in information systems, MIS Quarterly (1999) 67-93.

[44] G. DeSanctis and R. B. Gallupe, A foundation for the study of group decision support systems, Management Science 33 (1987) 5.

[45] N. B. Moe and D. Šmite, "Understanding lacking trust in global software teams: A multi-case study," in *Product-Focused Software Process Improvement*, Springer, 2007, pp. 20-34.

[46] T. D. Jick, Mixing qualitative and quantitative methods: Triangulation in action, Adm. Sci. Q. 24 (1979) 602-611.

Appendix 1: Raw data of RCA tool comparison

Software	Client: native/browser	Real-time collaboration	Cause-effect diagram	Idea development	Voting	Knowledge management	Costs	From
ARCA tool wirca.soberit.hut.fi	Browser	Yes	Yes, graph	Yes	Yes	Yes	Free (MIT)	-
Google Docs Drawing drive.google.com	Browser	Yes	Yes, graph Various drawing capabilities are provided, e.g., shapes with text and arrows. These can be used to create a CED graph.	Yes By using different shapes for causes and ideas.	Yes By using shapes with numbers.	NO	Free to use	Author
TapRooT Enterpriced. www.taproot.com	Both Native software + IE7 support (if activeX components)	(Yes) Likely not real-time, but accessible and editable. "collaboration of multiple investigators at separate locations over the net" AND "the user has access to edit or view all Audit/Investigation data as well as the ability to status Corrective Actions. When an Audit/Investigation is created or edited the user is provided the 7-Step Process Flow where the progress of the audit/investigation can be tracked and each technique can be viewed and edited"	Yes, tree	Yes	No	Yes "report incidents, analyze root causes, develop corrective actions, write and approve reports, track fixes, validate the effectiveness of the fixes, and trend performance"	Fee	Google*, Open-tube & Root Cause Live
REASON www.rootcause.com http://adsabs.harvard.edu/full/2002ESASP.486..369V	Browser "reason 9 is an online based software" AND "the software runs in a web browser"	No Individual investigator.	No Timeline is used instead. The whole process resulting to the issue is modeled. Many timelines can be created. The timelines result to a causal model which is a tree of causes automatically created.	Yes	No	Yes "preserve and communicate the knowledge learned" "enables you to manage and track your corrective action plans and communicates the lessons learned from your problem solving activities"	Fee	Google*#, Open-tube & Root Cause Live, Scopus
XFRACAS www.reliasoft.com/xfracas/	Browser "The system's Web-based user interface allows for easy access, collaboration and deployment throughout multiple sites, suppliers and dealers."	(Yes) Support for real-time system is not explicitly indicated, however, people can access to the analysis from various computers.	No Trees or graphs are not provided. "XFRACAS allows you to "categorize the incident with the Function > Failure > Effect > Cause that will "map" the event to a new or existing FMEA"	Yes "analysis and corrective action software" AND "configure XFRACAS to support any problem resolution methodology, from 4 to 8 steps, such as the four step DCOV process, the five step Six Sigma DMAIC process or the eight step 8D."	NO	Yes "build a "knowledge base" of lessons learned that will be instrumental to future troubleshooting and development efforts"	Fee	Open-tube & Root Cause Live

RCAT Software nsc.nasa.gov/RCAT/	?	?	Yes, tree "Create and edit a fault tree, Create and edit an event and causal factor tree"	?	?	Yes "perform and document root cause analysis, identify corrective actions, perform trending, and generate data usable in precursor analysis and probabilistic risk assessment"	? "available free to government Agencies and contractors"	Open-tube
PathMaker www.pathmaker.com/pathmaker/pathhome.asp	Native The tool is native software, which requires installation for each PC.	YES "use the open floor option to allow anyone using the same PathMaker project as you to enter their ideas with yours in real time" AND " This tool's design, based on the classic brainstorming method.., allows the team recorder to keep pace with group thinking. "	Yes, tree "PathMaker's cause and effect diagram, or Ishikawa diagram tool helps users discover the root causes of problems."	Yes Ideas can be entered to the tool.	No	Yes "We use PathMaker for our Process Improvement Program and have implemented an SPC program which has reduced our process cycle times by 25-50%. Also use for our organizational Strategic Planning process." - John Heinrich	Fee	Open-tube & Root Cause Live
RealityCharting (Cause link) www.sologic.com/rca-products-services/charting-software	Native Web-browser based application only for reports. Native software for analysis.	No You need to share your analysis by exporting a file. Manual updating is required. Users need to click a button to refresh the content. CED can be created only at the client software.	Yes, tree At the client software, a tree diagram can be created.	Yes "identify effective solutions" Solutions can be entered to the tree diagram which embeds the ideas in the causes.	No	Yes "Store RCAs and data. Maintain causal relationships. Knowledge management"	Fee	Google*# & Open-tube
Solve Gpw Computer Consulting, Inc www.gpwcomputerconsulting.com/solverootcauseanalysis.htm The found link to the page does not work. Additionally, the tool cannot be found from Google.	?	?	Yes, tree	?	?	?	?	Open-tube & Root Cause Live
SIM® software (tripod beta) www.advisafe.com/software/incident-management/simr-simple-incident-analysis-method	Native "incident analysis report is automatically generated by the SIM® software, which can be further edited in MS-Word"	(Yes) Support for real-time system is not explicitly indicated. "the incident will be analysed by the people within the department in which the incident took place"	Yes, tree "The program will ask you to indicate why the event could take place. After that, the same question will be repeated 4 more times, so that the analysis tree can indicate 5 layers of causation."	Yes "The corrective actions are a unique part of the SIM® analysis. "	?	Yes Provides a report which can be further edited in MS-Word.	Fee	Open-tube & Root Cause Live
PROACT www.reliability.com/industry/proact_templates.html	Native Native client installation is required.	(Yes) Real-time capabilities are not explicitly stated. However, it seems that the tool provides some team work features through sharing and permissions. "Robust Root Cause Analysis Process for Collaboration, Trending, Streamlining & Standardization of Analyses" AND "Team Permissions - Access, Read-Only, Delete"	Yes, tree	(Yes) "PROACT® RCA provides the tools for the RCA analyst to easily document, validate, report and track findings and recommendations."	?	Yes "PROACT automatically builds your knowledge management database of completed analyses creating your own customized and interchangeable templates for future incident investigations."	Fee	Google*#, Open-tube & Root Cause Live

Investigation Catalyst code.google.com/p/meslib/source/checkout	Native Native client Installation is required. The software works only on mac. However, data entries can be made with web-browsers. This requires using third party services. The entries need to be imported to the system.	No "To use computers for remote data entry with Web Browsers on computers with Windows, Linux or Mac operating systems, contact Starline to set up a private server URL for your password-protected project files and designate an e-mail account to which data entered remotely will be forwarded for importing into Mac project work files."	No MES worksheet matrixes are used instead. "A major difference between the MES investigation system and current investigation paradigms is that MES uses a "process" model of phenomena, instead of a "causation" model."	(No) The system is introduced as a solution for analyzing problems. This in turn helps to make improvements. However, making the improvements is not introduced as a part of the system.	No	No Single case reports are provided, however, you cannot combine many reports together.	Free (GPL)	Open-tube & Root Cause Live
Blackbox (tripod beta) www.advisafe.com/software/incident-management/blackbox	Native Native client installation is required.	?	Yes, tree	(Yes) Not explicitly stated. "Blackbox automatically creates a clear and standardized report, including an incident report, a cause tree and recommendations"	?	Yes Provides a report including recommendations and causes detected.	Fee	Google# & Open-tube*
Investigator 3 (tripod beta) http://www.advisafe.com/software/incident-management/investigator-3	Native Native client installation is required.	?	Yes, tree	Yes "Investigator 3 supports all the stages of the incident investigation process, from initially identifying what happened, through the analysis process and to writing the recommendations."	?	Yes "Investigator 3 supports a perfectly editable native Word export to make your report meet your organizations standards."	Fee	Google# & Open-tube
Track (tripod beta) www.advisafe.com/software/incident-management/track	Native Native client installation is required.	No	No The tool is based on questionnaires about incidents. Questionnaires are followed with "it follows" questions.	No	No	Yes The software outputs a track incident report which includes the distributions of cause types and actual causes organized as a structural list.	Fee	Open-tube
Web-based Quality Management Tool www.qitconsulting.com/CorrectiveAction.htm	Browser (works only in IE 6 or higher) "A web-based quality system for Manufacturing, Automotive, OEM/ODM, Food and Drug, and Service industries"	Yes "Collaborating suppliers, departments, and divisions in global scale " AND "Real-time corrective/preventive tracking and reporting"	No Forms to conduct RCA are provided. The causes are not organizer as CED.	Yes "Sharing and monitoring improvement activities"	No	Yes "Managing ALL types of corrective/preventive actions and monitor action progress online"	Fee	Google*#

RealityCharting® Software www.realitycharting.com/software	Both (Internet Explorer 7+ standalone client for mac and Windows)	Yes “The Track Changes tool allows groups of people to work together and share constructive input with the visible notification of any addition, deletion, reposition, or text change of a cause.”	Tree	Yes “The solution generation process systematically moves from cause to cause allowing you to propose solutions until each cause has been reviewed.”	Yes “The assessment evaluates each solution against the 5 default criteria. To change a default criteria setting, select the related field and type in your own criteria entry.”	Yes “The Action Item Report stores automatically generated action items from evidence fields and cause path endings.”	Fee	Google*#, Open-tube & Root Cause Live
ABS Consulting Root Cause Map™ www.absconsulting.com/ root-cause-analysis/root-cause-analysis-software.cfm	?	Yes “The web-based system is designed to capture, analyze and report all adverse impacts to your organization.” AND “Real-time reporting and management dashboards ”	?	Yes “Corrective Actions / Preventive Actions (CAPA) Management ”	?	Yes “Centralized system (multiple language support)—One location for all incidents, events, investigations, recommendations, root causes and action items ” AND “Flexible classification system—OSHA, EU, ABS Consulting or any other standards ”	Fee	Google*# & Root Cause Live
ROOT-CAUSE-ANALYSIS-SOFTWARE 5.1 www.sqaki.com/17/ ROOT-CAUSE-ANALYSIS/	Native Native client installation is required.	No	Tree	No	No	No	Fee	Google*#
ThinkReliability Excel Template www.thinkreliability.com/excel-tools.aspx	Native MS Excel is required in order to run this template.	No This is an excel sheet only. Furthermore, it does not work in Google Drive and thus real-time collaboration is not an option.	Tree	Yes	No	No	Free (MS Excel)	Google*#
Enablon IMS www.enablon.com	?	?	Tree	Yes “Creation of corrective and preventive action plans”	?	Yes “Enablon IMS meets all event (incidents, accidents, etc.) reporting, management and monitoring needs, both for individual sites and the Group as a whole.” AND “Management & monitoring of corrective & preventive action plans”	Fee	Google*#
Smartdraw http://www.smartdraw.com/	Native Requires to install client software	No This software is used from one PC. The user can export a file, which can be opened from other PC.	Tree	No Only “causes” can be added to the diagram.	No “Only comments can be added”	No The software is only for drawing, not for analysis of many drawings.	Fee	Google #

Set-Based Thinking http://www.targetedconvergence.com/tcc-can-help/software-features/	Native Based on the screenshots, the tool requires to install client software	No A3 reports are used to "collect together a set of visual models which concisely tell the story of the ongoing discussion."	Graph "packages are designed to chart the particular relationship they are analyzing just fine; but generally we need to pull together analyses from many different tools of many different relations that must all be considered when making a decision"	Yes The corrective actions are called as "decisions". The feature list of the tool states that "identify what decisions must change to implement those remedies [of causes]"	?	Yes The tool is used to combine knowledge from various sources (e.g. individuals). This includes results from root cause analysis and decisions made.	Fee	Google #
PHRED http://www.phredsolutions.com/customerfocus.html	(Browser) "PHRED is a web-based problem solving system. It makes it easy for you, your suppliers and contract manufacturers to enter, edit and manage problems."	(Yes) "Problem solvers, experts and managers share a common process and information"	(Yes) The causes are detected by using questions only. In the end, software provides a report which is a tree based diagram.	Yes "PHRED takes you through outlining a solution and presenting it for agreement, sign-off and implementation. PHRED tracks the multiple implementation actions."	?	Yes The tool includes a database which is used to "Share Root Cause information between people and plants." AND "Standard reports, individually defined user query reports, management summaries and charts. Export the information into Excel or PDF. Send only the information you want to share with your customers and suppliers."	Fee	Google*
BowTieXP http://www.cgerisk.com/software/incident-analysis/incidentxp/rca	Native The screenshot of the tool reveals that the software requires client installation	No	Tree	(No) There is no evidence that the tool is used to create corrective actions. It seems that the tool is used to detect cause-and-effects only.	?	Yes The output of RCA can be refined and stored. "Ongoing effort shall be made to examine ways in which a similar improved learning from incidents can be realized by correlating RCA with bowtie analysis. One could think about classification of events, and perhaps correlation of events to barriers in the bowtie diagrams."	Fee	Google*
FMEA Software http://www.fmea.co.uk/APIS_FMEA_software.html	Native Requires client installation in order to be used.	(No) Shares the information through the web, but the analysis is conducted on a single PC.	Tree	Yes The tool provides features for developing and registering corrective actions.	No	Yes "procedure of focusing on what can go wrong, what possibly could cause it and what are the potential effects. Quantification of the risk, taking into account the current controls, then indicates areas of weakness. It is widely used in manufacturing industries in various phases of the product life cycle. Failure causes are any errors or defects in process, design, or part, especially ones that affect the customer."	Fee	Google*

Systems2win http://www.systems2win.com/solutions/brainstorming.htm	Native Is an Excel template.	No Is an Excel template which is used to substitute whiteboards in "brainstorming sessions"	Graph	No	No	No	Fee	Google*#
iReliability Root Cause Analysis http://ireliability.com/product/root-cause-analysis/	Browser	No The analysis is conducted on a single PC.	Tree	Yes Track and facilitate implementation activities	No	(No) The existing RCAs are stored and the user can view and refine them. However, there is no evidence that the user can share the information with other users (e.g. organization's members)	Fee	Google*
FMECA Software http://www.itemsoft.com/fmeca.html?gclid=CNbprZTd9bkCFWd7cAodG2cAjQ	Native Require client installation	No The analysis is conducted on a single PC.	Tree The screenshots indicate that the tool supports tree diagrams. Additionally, "an enhanced hierarchy tree and tabular views" AND "Graphically constructed system hierarchy diagrams"	(No) Nothing indicates that corrective actions are registered with the tool.	(No)	(Yes) This feature is not explicitly stated, however, it is stated that "build and open multiple systems and project files" AND "Powerful reporting and charting facilities".	Fee	Google*
Rapid Problem Isolation http://www.neebula.com/it-incident-management-tool/root-cause-analysis-software/	Native "download and install a small-footprint collector that communicates securely with our cloud-based application."	No This software is used to link technical solutions to the business application, automatically.	Tree	No The software is used to view the technical cause-and-effect linkages only.	No	(Yes) The personnel get information about the technical solution linkage to the business solutions. However, it seems that this information needs to be shared manually.	Fee	Google*
Lassale http://www.ncbi.nlm.nih.gov/pmc/articles/PMC419418/#!po=75.0000	Native Screenshots indicate that a client installation is required. Additionally, it is stated that "designed for Visual Basic"	No	Tree	No The software is used to create cause-and-effect diagrams only.	No	(No) The analysis is run on a single PC. However, it is stated that "The database backend is SQL Server".	?	Google*, Google Scholar
CA Spectrum http://www.ca.com/us/root-cause-analysis.aspx	?	?	?	?	?	?	Fee	Google*
RootCause http://www.rlsolutions.com/Root_Cause_Analysis.aspx	?	No It seems that the analysis is conducted on a single PC and the results can thereafter be used in future analyses.	(No) The tool uses questions answered by the user. Thereafter, a report is made. No indication that there is a cause-and-effect diagram available.	Yes "Send action items to multiple recipients and track their progress"	?	(Yes) Not explicitly stated. However, "Import RL6:Risk data into your root cause analysis, to reduce rework and reduce the chance of errors" AND "Monitor your ongoing improvement in frequency of process failures with the RL6 Report Center"	Fee	Google*
Speechminer http://www.utopy.com/speechminer/	?	?	?	?	?	?	Fee	Root Cause Live

Root Cause Analyst http://www.ccdsystems.com/Products/RootCauseAnalyst/RCAProductDescription.aspx	(Native) "Programmed in Visual Basic"	?	(Tree) Seems to be a tool that uses questions which the user answers. Thereafter, the user can view the cause-and-effects as a tree. "One-button generation of flowcharts & factor trees"	?	?	(No) There is no evidence that the tool provides any features for refining and sharing prior analyses over the tool. However, it is stated that "Import & export analyses and Factor Guides" AND "All reports generated in Microsoft Office format".	Fee	Root Cause Live
RCA GUI http://www.emeraldinsight.com/journals.htm?articleid=1907212&show=abstract	Native Screenshots reveal that the software require client installation	No The application is used by a user, who uses the tool by asking experts over the root causes detected. "Experts in the PCA manufacturing industry were questioned over the most likely root cause of the problem from those provided by the RCA module"	Tree The application creates a tree diagram based on the information gathered automatically from a technical system. ""	Yes "the user can.. ..propose it as a design change to eliminate the manufacturing defect investigated."	No	Yes The tool seems to be integrated to a knowledge management system. Additionally, the system provides assistance for the user based on prior knowledge, e.g., "the software provides guidance to the user on the investigation of a defect based on previous knowledge formalized in the form of integrated models."	?	Scopus

No=this feature is not available in the software tool, Yes=this feature is available in the software tool, (No)=it is likely that this feature is not available in the software tool, but we were not able to verify that, (Yes)= it is likely that this feature is available in the software tool, but we were not able to verify that, ?=we were not able to find any evidence on the occurrence of this feature, Fee=the software is subject to a charge, Free (license)=the software is free, Free to use=using the software is free
Google* = Found from our Google search "root cause analysis software"
Google# = Found from our Google search "root cause analysis software" free
Open-tube = Found from http://open-tube.com/10-best-software-tools-to-conduct-root-cause-analysis-and-solve-complex-problems/
Root Cause Live = found from http://www.rootcauselive.com/library/Software.htm
Scopus = found from scopus.com
Google Scholar = found from Google Scholar

Appendix 2: Questionnaire used in Case 1

1. What is your title?
2. Select the roles that describe your responsibility best
    a. Manager
    b. Product Owner
    c. Developer
    d. Something else, what?
3. How long have you worked in this role(s)?
4. How long have you worked at the company?
5. Target Problem
   Give a value (1=very minor, 2, 3, 4, 5=very major) that corresponds the question best
    a. Effort the company has used to try to prevent the target problem earlier
    b. The internal impact of the target problem for the company
    c. The external impact of the target problem for the company
    d. The impact of the target problem for team's communication
6. Target problem causes
   Give a value (1=very minor, 2, 3, 4, 5=very major) that corresponds the question best
    a. The correctness of the detected causes
    b. The correctness of the detected root causes
    c. Impact of resolving the found causes of the problems
7. Retrospective method
   Give a value (1=very minor, 2, 3, 4, 5=very major) that corresponds the question best
    a. The easiness to collect the causes

b. The easiness to detect the root causes
c. The easiness to organize the causes
d. The easiness to detect the root causes of the target problem
e. My own contribution in the retrospective
f. The openness of the communication in the retrospective

Give a value (1=absolutely NO, 2, 3, 4, 5=absolutely YES) that corresponds the question best

g. Is the ARCA tool easy?
h. Is the ARCA tool learnable?
i. Did the ARCA tool help in finding problems and causes?
j. Would you have found the same causes without the tool?
k. Would the retrospective have been easier without the tool?
l. Would the retrospective have been more effective without the tool?

Appendix 3: Questionnaire used in Case 2

1. What is your title
2. Select the roles that describe your responsibility best
3. Target Problem

   Give a value that corresponds the question best [1=very low, 2, 3, 4, 5=very high]

   a. Effort the company has used to try to prevent the target problem (or similar ones) earlier
   b. Internal impact of the target problem for the company
4. Target problem causes
   a. Correctness of the detected causes
   b. Impact of resolving the found causes of the problems
5. Retrospective method

   Give a value that corresponds the question best [1=very low, 2, 3, 4, 5=very high]

   a. Easiness to collect the causes
   b. Easiness to detect the root causes
   c. My own contribution in the RCA session was
   d. Openness of the communication in the RCA session was
   e. Compared to our team's current process improvement practices, do you find using the RCA method cost efficient?
   f. Compared to our team's current process improvement practices, do you find using the RCA method easy?
   g. Compared to our team's previous practices to find causes behind problems or issues, do you find using the RCA method useful?
   h. The easiness to detect the causes of the target problem was..
   i. To solve the target problem, was detecting causes for the target problem useful?
   j. In contrast to the company's practices, was the method used to detect target problem causes useful?
   k. Was it easy to detect the target problem causes?
6. The ARCA tool

   Give a value that corresponds the question best [1=very low, 2, 3, 4, 5=very high]

   a. Compared to an RCA session done by using post-it notes, do you find using the online ARCA-tool cost efficient?
   b. Compared to an RCA session done by using post-it notes, do you find the online ARCA-tool easy to use?
   c. Compared to our team's previous process improvement practices, do you find the online ARCA-tool useful?
   d. Is the online ARCA tool easy to use?
   e. Is the online ARCA tool easy to learn?
   f. Did the online ARCA tool help to find problem causes?
   g. Would we have found the same problem causes without the tool?
   h. Was it difficult to organize the problem causes with the online ARCA tool?
   i. In contrast to our company practices, is the online ARCA tool cost efficient to detect process improvement targets?