The Benefits of Pre-Requirements Specification Traceability

Julia Krause
Department of Computer Science
Friedrich-Alexander University Erlangen Nürnberg
Erlangen, Germany
julia.krause@fau.de

Andreas Kaufmann
Department of Computer Science
Friedrich-Alexander University Erlangen Nürnberg
Erlangen, Germany
andreas.kaufmann@fau.de

Dirk Riehle
Department of Computer Science
Friedrich-Alexander University Erlangen Nürnberg
Erlangen, Germany
dirk@riehle.org

Martin Jung
Software Engineering Consulting
develop group
Erlangen, Germany
jung@develop-group.de

***Abstract*—Requirements traceability is the ability to trace requirements to other software engineering artifacts. Traceability can be classified as either pre- or post-requirements specifications (RS) traceability. Pre-RS traceability is the ability to trace between requirements and their origin. However, the benefits of pre-RS traceability are often not clear. In this article, we systematically lay out the benefits of pre-RS traceability. We present results from both a literature review and a qualitative survey of practitioners involved with documenting and utilizing such trace links. We find that the benefits strongly depend on the practitioners, their tasks, and the project environment. Awareness of these relationships supports a clearer understanding of the benefits of pre-RS traceability and thus motivates successful implementation of the required practices. The results of our research motivates the adoption of pre-RS traceability and present problem areas for future research.**

***Index Terms*—Requirements Engineering, Requirements Traceability, Pre-Requirements specification Traceability, Literature review, Qualitative Survey**

I. Introduction

With requirements traceability, users have the ability to trace requirements from their origins through their definition and evolution until their eventual implementation and validation by creating and using trace links. There exist two types of requirements traceability. Linking requirements with artifacts that are based on the requirements specification (RS) is done by trace links referring to post-RS traceability. A lot of research is being done in this area. Pre-RS traceability is the ability to link requirements with their origin. Questions like *What is the reason behind a particular requirement?*, *Why was this requirement changed?* and *Who was involved?* can be answered by pre-RS traceability. As early as 1992, Gotel stated that *"[pre]-RS traceability has the potential for much greater leverage than post-RS traceability on costs and quality of software products"* [1]. Yet, to this date, the space of research literature on pre-RS traceability has not been systematically laid out.

Therefore, we performed a systematic literature review (SLR) to identify challenges, existing solutions and use cases of pre-RS traceability [2]. In the initial stages of the SLR we learned that requirements engineers and other people working with requirements frequently had an unclear understanding of the benefits of pre-RS traceability and how to achieve them. As a consequence, in many cases they did not implement and use pre-RS traceability and associated trace links, missing out on its many benefits. Furthermore, the expected effort to document trace links or to maintain existing trace links is often perceived as too high and not justified by the expected benefits [3]–[5].

In this article, we present an overview of the benefits of pre-RS traceability. However, since pre-RS traceability is not equally beneficial for every use case, our first research question aims to identify the use cases and their organizational contexts. Relating use cases with expected benefits reveals two influencing factors that must also be taken into account: the types of practitioners involved and the characteristics of the particular project environment.

Different practitioners have different perspectives on traceability. They apply traceability practices in different use cases, and they expect different benefits from using trace links [6]. For example requirements engineers use trace links between requirements and their origin to identify missing requirements or past decisions and software developers may follow trace links to gain more contextual information on the intent of a requirement.

To get more insight into practitioners, their use cases, and expectations we triangulate and extend our results from the SLR with a qualitative survey consisting of ten semi-structured interviews with practitioners from industry.

The contributions of the paper are:

- Presentation of a collection of use cases where pre-RS traceability is beneficial

- Clarification of the benefits of pre-RS traceability, specifically for different practitioner roles and related characteristics of the project environment
- Identification of topics for future research

The paper is structured as follows. Section II provides an overview of similar articles predominantly in the area of requirements traceability. The research design, including the research questions, is presented in section III followed by the results in section IV. Afterwards in section V we discuss our findings and we highlight new directions of research. We close the paper with section VI about limitations and a conclusion in section VII.

II. Related Work

Requirements traceability has long been the subject of research. Gotel and Finkelstein [7] introduced the distinction between pre- and post-RS traceability. The results of their empirical study lead to a strong recommendation to put more effort into pre-RS traceability because it has a significant influence on a project's success.

Since then, much research has been carried out to examine requirements traceability by combining literature reviews and studies within the industry to expand and evaluate findings from literature with insights from the industry [8]–[12]. Most of them stated that pre-RS traceability needs more attention. We follow the suggestion of this previous research by combining a literature review with a qualitative survey focused specifically on the topic of pre-RS traceability.

Ahmad and Ghazali [11] identified problems of requirements traceability within small projects and developed guidelines for documenting trace information. They find that taking characteristics of the project and the company into account is important to develop a suitable traceability strategy.

Altaf et al. [3] developed visualized metaphors to support the reputation of benefits of pre-RS traceability. As one part of their research, they created a list of benefits of requirements traceability based on the literature. The visualized metaphors they developed were evaluated in an experiment within a real-world project. Our study builds on these findings, but instead of directly presenting a solution, we focus on the benefits of pre-RS traceability and complementing them with insights from interviews with practitioners from different industry domains. The results of our research could also improve the visualizations and individualize them for different roles and contexts.

Gotel and Finkelstein [7] recommend that roles and related individual needs of practitioners involved have to be taken into account when developing traceability strategies. Therefore, they [13], [14] modeled the contribution structure underlying requirements artifacts considering different roles involved.

Previous studies [4], [15]–[18] deepened the findings on characteristics of roles and how they influences requirements traceability. For example Gotel and Fineklstein [1], [7] distinguish between *providers* who have the ability to create trace links and *end-users* who uses the trace links. In contrast, Ramesh [4] and Mäder et al. [16] distinguish between different user groups based on the underlying motivation and practice. Our study considers these findings and therefore we conducted interviews with people who have different roles such as consultants or requirement engineers.

III. Research Design

Our initial research goal was to identify open and unsolved issues in pre-RS traceability. During a literature review we identified the benefits of pre-RS traceability in different contexts as a relevant but in many respects open question. Therefore, our research questions became:

RQ1: Which use cases benefit from pre-RS traceability?
RQ2: What are the expected benefits?

As a consequence, our research design is structured into two parts. We performed a literature review, the results of which were supplemented and expanded by a qualitative survey with practitioners from the industry.

A. Literature review

The literature review was part of a structured literature review (SLR) according to Kitchenham et al. [19] on pre-RS traceability. A more detailed description of the method and the results can be found in our technical report [20]. A research protocol was drawn up in the beginning to describe the background and to define all important cornerstones like research questions, search strategy, selection and quality criteria, research process, the data extraction process, and a work program. Regular peer debriefings [21] with the second author were conducted to ensure high quality during the procedure and a good fit of research questions to research design.

1) Search strategy: To find relevant articles we used two different search strategies. We first performed a pilot keyword search based on the search term "pre-requirements specification traceability". Afterwards we did snowballing by performing forward an backward search to identify additional search terms [22]. Finally, We decided on three individual search terms that cover different designations of pre-RS traceability. The keywords with the associated number of articles found per database are shown in table I. In order not to miss relevant literature despite the selection of search terms due to varying terminology, we carried out an additional forward an backward search on relevant articles.

2) Selection criteria: To identify relevant articles we defined the following selection criteria:

a) the article is written in English
b) the article has been peer-reviewed
c) the article is about techniques to link RS with their origin
d) the article is about an overview which presents different techniques, issues and/or problems to link RS with their origin
e) the article is about an evaluation of a technique which links RS with their origin

If criteria a) and b) apply and at least one of the criteria c), d) and e) applies, then the article was included in our further research.

TABLE I
SEARCH TERMS AND RESULTS PER DATABASE

Search term	Google Scholar	IEEE Xplorer	ACM	Web of Science
"pre-requirements specification traceability"	55	1	3	1
"pre-requirements specification"	124	5	6	0
"requirement provenance"	36	2	1	4

TABLE II
NO. OF ARTICLES PER STAGE OF SELECTION

Selection stage	No. of articles
Based on keyword search (sum of found articles based on table I)	238
Included articles from keyword	36
Included articles from forward and backward search along referenced articles	31
Included articles total	67

3) Data extraction and synthesis: To organize our data and create an overview of their properties, we developed a table that contains the whole collection of articles and their state of analysis. We performed an iterative data extraction process by starting with a randomly selected sample of articles to improve the research protocol.

We first analyzed the title, abstract, and potentially relevant parts of the articles based on the selection criteria. Table II summarizes the number of articles at the different stages. After applying the selection criteria, we identified 36 relevant articles by keyword search and 31 relevant articles by backward & forward search. Finally, we included 67 articles (list in appendix VIII-A) into our SLR.

The included articles were published between 1992 and 2020. Figure 1 shows the number of articles published each year. We identified 45 relevant conference papers and 14 relevant journal articles. Eight relevant documents came from books, reports, and workshops. More than half of the articles (52%) came from the *International Requirements Engineering Conference (RE)* and 13% from *Traceability in Emerging Forms of Software Engineering (TEFSE)*. The 14 journal articles originate from 13 different academic journals. The large number of different sources demonstrates the need for this SLR.

Articles identified as relevant were analyzed by performing **qualitative data analysis (QDA)**. For this purpose, relevant segments of text were annotated with abstract concept labels, so called codes and these codes were combined in a hierarchical code system. During the first pilot sample, a basic code system was developed and was continuously refined. The coding process is iterative. Each iteration consists of coding one article based on the three steps: annotating relevant segments of text (open coding), restructuring the code system

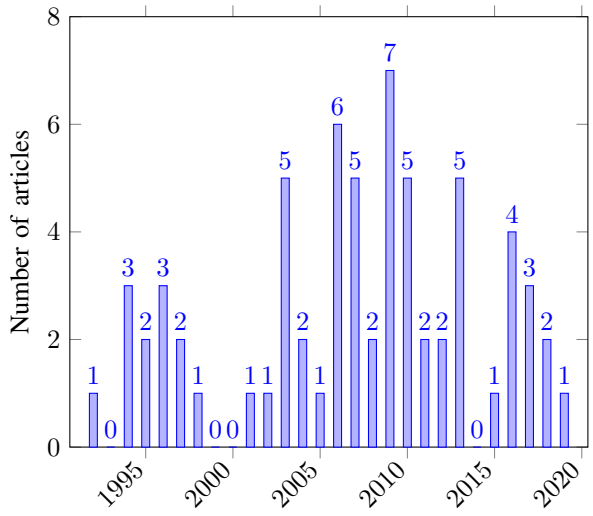

Fig. 1. Number of articles published per year

(axial coding), and arranging and relating our codes to the core category and thereby refining focus (selective coding) [23]. Saturation is reached if there are no significant changes in the code system over the past four iterations. Table III shows the main codes of the hierarchical code system and the number of assigned segments of text. The blue highlighted codes including there sub-codes are in the scope of this article, the other codes and their sub-codes can be found in our technical report [20].

TABLE III
MAIN CATEGORIES OF THE RESULTING CODE SYSTEM AND RELATED NUMBER OF CODINGS

Code	No. of codings
Requirements Traceability (RT)	5
Pre-RS traceability general	94
Traceability users	31
Use cases & Benefits	0
↪ Use cases	168
↪ Benefits	52
Problems & Challenges	242
Solutions & Suggestions	249
Traceability tool	47
Traceability models/techniques	193
Consequences of inadequate pre-RS traceability	4

Use cases & Benefits groups sub-codes and was not used to assign segments of text therefore the number of codings is zero. The sub-code *Use cases* corresponds to RQ1 and groups sub-codes representing specific use cases. The same applies to the sub-code *Benefits* corresponding to RQ2. We furthermore identified the code *Traceability users* as relevant because we identified that various users and their roles within a use case experience different benefits. Therefore the code *Traceability users* is relevant to answer RQ2.

4) Quality assurance: In addition to a continuous professional exchange between all co-authors, we carried out regular peer debriefings guided by the description of Spall [21] to ensure high quality during the procedure and a good fit of research questions to research design. We performed three peer debriefings, one after the first half and one during the second half of the QDA. The third peer debriefing was conducted during the data synthesis. Critical points were discussed together in order to then make a decision on how to proceed. Each debriefing was documented in a peer debriefing protocol.

B. Qualitative survey

To add more practical insight into the results of the literature research we then conducted a qualitative survey guided by Jansen [24]. We performed the following steps:

1) Defining knowledge aims: The aim is to answer the RQ1 and RQ2. Furthermore, we want to identify the gap between scientific literature and industry.

2) Defining a sampling model: Our sampling model covers the following roles to address different perspectives for working with requirements:

- *creators of RS* such as requirements engineers, product owners, analyst or strategy developer
- *RS-users* such as product owners or software developer
- *Consultants* such as RE consultants, software engineering (SE) consultants or trainer provided by consulting firms or self-employed

Further, our sampling model includes the consideration of the characteristics of projects in which our participants work (agile, plan-driven, development of safety-critical systems). Due to the exploratory nature of the study, we did short-term interactions by starting with a small sample, analyzing this sample, and deciding to adopt the sampling strategy to discover new samples.

3) Sampling according to the model: We sampled people who were recruited through the career networks LinkedIn[1] and Xing[2] and our direct contacts to the industry. We were looking for candidates who have a role covered by the roles in the sampling model, or who have at least IREB's CPRE Foundation Level certification[3] and are currently working with requirements. Based on the results found, it was not always possible to get information about the type of role and whether they are involved in agile, plan-driven, or other project types. So, we invited potential candidates to fill out a preliminary questionnaire. This questionnaire is presented in the appendix VIII-B. Table IV summarises the number of candidates at the different stages of sampling.

TABLE IV
NUMBER OF CANDIDATES AT THE DIFFERENT STAGES OF SAMPLING

Sampling step	No. of candidates
Initial identified candidates	39
Candidates contacted	20
Candidates who completed the preliminary questionnaire	13
Participating candidates	10

Based on the 13 candidates who completed the preliminary questionnaire, we selected ten candidates that collectively covered the widest variance, and thus our sample model covered the widest.

4) Interviews with sampled instances: We conducted semi-structured interviews to maintain the possibility of exploring topics more deeply. The duration of the interviews ranged from 45 to 90 minutes. The interview guidelines were created based on the results of the preliminary questionnaire. The template on which the basic structure of our interview guideline was created can be found in the appendix VIII-C. We conducted 10 interviews with candidates from five different companies (table V). Table VI presents the roles of the interviewees.

5) 5. Analysis of interviews for theory: We performed a QDA on the transcripts of the interview. After the analysis of five interviews, we performed an interrater reliability session with two researchers to improve and ensure the quality of our analysis by performing investigator triangulation. Within this session, each researcher recoded two interviews. One of the two interviews was recoded by both researchers in order to subsequently stimulate a joint discussion afterwards. Each researcher made notes during the coding about missing codes, unclearly described codes, or general improvements. These notices were discussed afterwards in a meeting to define actions to improve and restructure the code system. A discussion about the 11 suggested changes to the code book resulted in significant improvements to the definitions of our codes, lead to five previously missed concepts and fostered a shared understanding between all three coders.

IV. RESEARCH RESULTS

The literature and the interviews reveal 13 use cases and nine benefits of pre-RS traceability. *The application of one or mutable [use cases] by a [user] within an [environment] can lead to one or multiple [benefits].* Figure 2 visualizes the relationships between these variables as formula. Answering RQ1 we identified use cases which correspond to the blue box on the left side. From these we abstracted three main categories of use cases. To understand the context in which the benefits (answering RQ2) can materialize, it is important to

[1]LinkedIn https://www.linkedin.com/
[2]Xing https://www.xing.com/
[3]CPRE Foundation Level https://www.ireb.org/en/cpre/foundation/

TABLE V
CHARACTERISTIC OF COMPANIES AND THE RELATED NUMBER OF INTERVIEWED CANDIDATES

ID	No. of candidates	Company focus	Company size
Comp_01	4	IT consulting	Medium (< 250 employees)
Comp_02	2	Software development	Large (> 249 employees)
Comp_03	2	Automotive	Large (> 249 employees)
Comp_04	1	RE training and consulting	Medium (< 250 employees)
Comp_05	1	Communication and IT consulting	Large (> 249 employees)

TABLE VI
ROLES OF INTERVIEWEES

ID	Role	Company
Inter_01	Software engineering (SE) consultant	Comp_01
Inter_02	RE & Strategy consultant	Comp_03
Inter_03	RE consultant & Trainer	Comp_04
Inter_04	SE consultant	Comp_01
Inter_05	RE & Strategy consultant	Comp_05
Inter_06	Architect & SE consultant	Comp_01
Inter_07	Strategy developer	Comp_03
Inter_08	RE & SE consultant	Comp_01
Inter_09	Business analyst	Comp_02
Inter_10	Product owner	Comp_02

not only consider the use case itself but also the corresponding environmental and human (middle blue boxes) factors which appear to be conducive to the benefits (right blue box).

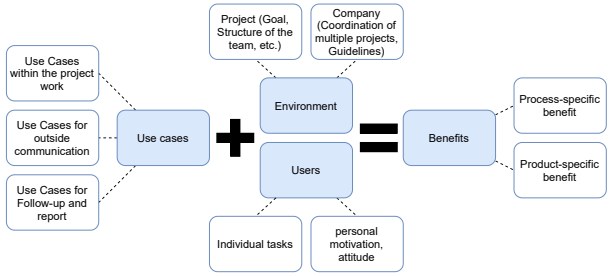

Fig. 2. Formula about influencing factors for the benefits of pre-RS traceability: *The application of [UCX] by a [User] within an [Environment] can lead to [Benefit].*

A. Which use cases benefit from pre-RS traceability? (RQ1)

Based on the literature and the qualitative survey we identified 13 use cases that benefit from pre-RS traceability.

The interviews show that consultants have a greater awareness of use cases and their benefits from pre-RS traceability than people working specifically in projects. There are probably two reasons for this:

- The experience of consultants is usually based on many different projects from which they can synthesize best practices and avoid repeating problems.
- Consultants are usually called in when problems have already occurred or if special expertise is required. To familiarize themselves with the project, in many cases they have to trace requirements.

The following use cases show that pre-RS traceability not only supports problem solving or familiarization, but also supports day-to-day work. The identified use cases are divided into three categories as follows:

a) Use cases within the project team: focus on the day-to-day work of project members or other practitioners involved within the company.

UC1 *(Mentioned in literature [3], [7]–[9], [11], [13], [14], [16], [25]–[36] and mentioned in interviews Inter_01, Inter_03, Inter_04, Inter_05, Inter_07, Inter_08, Inter_10)* Finding the origin of requirements to get context, to get clarity about ambiguous requirements, and to determine the importance of requirements.

UC2 *(Mentioned in literature [3], [4], [7]–[9], [11], [14], [26], [29], [32]–[34], [37], [38] and mentioned in interviews Inter_01, Inter_04)* Finding people who can answer questions about particular requirements.

UC3 *(Mentioned in literature [3], [7], [26], [30], [31], [39]–[44] and mentioned in interviews Inter_01, Inter_06, Inter_10)* Knowing or remembering past decisions. Knowledge about past decisions supports future decisions.

UC4 *(Mentioned in literature [3], [29], [36], [45]–[47] and not explicit mentioned in interviews)* Managing the system evolution or supporting of maintenance tasks through transparent documentation of the system's creation. Such creation information includes, among other things, the reason for a requirement, which can play a major role in the revision.

UC5 *(Mentioned in literature [11], [31] and mentioned in interviews Inter_08)* Prioritizing of requirements by assessing their importance based on their origin. For example, a requirement that addresses a problem for multiple stakeholders can be more important than a problem that only one stakeholder has.

UC6 *(Not explicit mentioned in literature and mentioned in interviews Inter_08)* Assigning requirements to a project-specific category in the RS such as hardware or software components through transparent documentation of their creation.

b) Use Cases for outside communication: focus on communicating with parties outside of the project team, usually with stakeholders.

UC7 *(Mentioned in literature [3], [8], [10], [16], [29], [40], [48]–[50] and mentioned in interviews Inter_02, Inter_06, Inter_07)* Proving the fulfillment of regulatory compliance by tracing back to the regulation from requirements.

UC8 *(Mentioned in literature [3], [11], [27], [29], [31], [50] and mentioned in interviews Inter_01, Inter_02, Inter_05, Inter_10)* Proving the fulfillment of stakeholder needs by tracing back to statements about needs from requirements. This supports coverage analysis.

UC9 *(Mentioned in literature [3], [51] and mentioned in interviews Inter_01)* Assistance in negotiating requirements through knowledge of origin, history and people who can answer questions.

c) Use Cases for follow-ups and reports: focus on the evaluation of projects or particular project steps in order to gain knowledge and monitor progress.

UC10 *(Mentioned in literature [4], [8], [9], [11], [26], [27], [29], [31], [33], [40], [52] and mentioned in interviews Inter_07)* Identification of potential influencing factors for changes in requirements in order to be able to predict future changes. This is particularly important in the case of long-term requirements over several years.

UC11 *(Mentioned in literature [3], [8], [9], [28], [30] and mentioned in interviews Inter_01, Inter_03, , Inter_06, Inter_07, Inter_08)* Storing and reviewing the history of requirements for example to identify frequently changing requirements. This requires efficient versioning of requirements.

UC12 *(Mentioned in literature [3]–[5], [8], [16], [29], [31] and mentioned in interviews Inter_10)* Determining the degree of completion of the RS by analyzing which stakeholder information is already addressed in the RS and which information is not yet considered. This supports comprehensive requirements management.

UC13 *(Mentioned in literature [3], [45] and not explicit mentioned in interviews)* Creating a knowledge management system to learn from the past and derive best practices.

People working within a project team are most interested in the fulfillment of the stakeholder needs, they mentioned *UC8* most frequently. In contrast, consultants focus more on the improved understanding using pre-RS traceability. The most frequently mentioned use cases in this user group are *UC3* and *UC11*.

The subsequent comparison of the number of mentions of use cases in the literature to the interviews shows a similar distribution. However, there are three use cases which were mentioned only either in the literature or in the interviews. Table VII presents an overview.

Within our 10 interviews, we did not go through a complete checklist of all already compiled use cases in order to focus on those aspects that were important to the interviewees themselves and have them lead the discussions from their unique perspective. However, the mentioned and unmentioned use cases point to a gap which needs further investigation for example by conducting and analysing a larger set of interviews.

TABLE VII
RANKING OF USE CASES BASED ON MENTIONS IN THE LITERATURE COMPARED TO THE INTERVIEWS

Top 3 UCs in literature	Top 3 UCs in interviews
UC1: Finding source	UC1: Finding source
UC2: Find people	UC3: Knowing or remembering past decisions
UC7: Fulfillment of regulatory compliance	UC11: Storing and reviewing the history of requirements
Not mentioned in literature	not mentioned in interviews
UC6: Assigning requirements	UC4: Managing the system evolution
	UC13: Creating a knowledge management system

B. What are the expected benefits? (RQ2)

The literature review and qualitative survey revealed two primary factors that influence the benefit of pre-RS traceability for the identified use cases: characteristics of the project environment and the practitioners involved (figure 2). The following list of identified benefits considers these influencing factors. Furthermore, we related use cases to the benefits, because not each use case is beneficial in every context. The benefits are divided into two different groups: process- and product specific benefits.

a) Process-specific benefits: support internal processes and workflows within a team and a company.

B1 **Gain knowledge for future development** *(Mentioned in literature [3], [4], [28], [30], [31] and mentioned in interviews Inter_06, Inter_07, Inter_10)* Almost all roles can benefit from it, especially in long-term projects. Reviewing past actions, especially decisions reduces the repetition of wrong decisions and builds experience (*UC3*, *UC13*). Furthermore, tracing between high-level requirements and project-specific requirements supports monitoring of project progress, and helps to visualize the status of implementation (*UC12*). Inter_10, as a product owner, mentioned reviewing the creation and implementation of requirements by tracing back to the stakeholder wishes, brings certainty that these wishes have been correctly addressed or leads to opportunities for improvement. Further, trace links can be used to make success measurable.

> *"And of course, as a product owner, I also get certainty that my interpretation of the stakeholder statement is correct [...] I do believe that it is very important, especially concerning the measurability of success, to always trace this path of creation back to the origin"* Translated quote1 in VIII-D from Inter_10

B2 **Improve communication and collaboration** *(Mentioned in literature [3], [14], [42], [45] and mentioned in Interviews Inter_01, Inter_08)* Pre-RS trace links show knowledge carriers (*UC2*) and support transparent documentation. This improves communication within the team and to the stakeholders, especially in distributed work environments. Inter_01 and Inter_08 mentioned that having a clear derivation path of requirements avoids conflicts when questions arise concerning why something was implemented the way it was (*UC3*, *UC5*).

"If you have the clean derivation paths, there's no bad blood afterwards because someone can debate why the piece of software looks the way it does." Translated quote2 in VIII-D from Inter_01

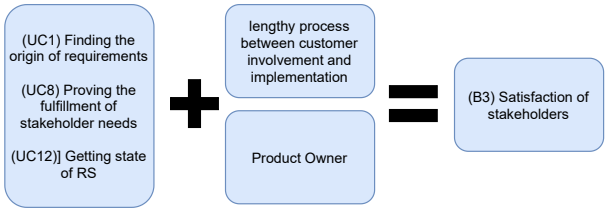

Fig. 3. Formula: *The application of UC1, UC8, UC12 by a Product Owner within a lengthy process between customer engagement and implementation can lead to (B3) stakeholder satisfaction* by avoiding misinterpretations.

B3 **Satisfaction of stakeholders** *(Mentioned in literature [8], [11] and mentioned in Interview Inter_10)* A proven way to satisfy stakeholders is to give them the opportunity to follow the project progress. In the case of pre-RS traceability, the stakeholder has the ability to understand and see which needs are satisfied by which requirements (UC8, UC12). The corresponding trace links relate the requirements to the original stakeholder statement. Thereby, even in the face of interpretations and adaptations during information transfer, the intention of a stakeholder remains traceable and readable for everyone (UC1). This is particularly suitable in cases where there is a lengthy process between customer involvement and implementation. Figure 3 presents the related formula: *The application of UC1, UC8, UC12 by a Product Owner within a lengthy process between customer engagement and implementation can lead to (B3) stakeholder satisfaction* by avoiding misinterpretations.

B4 **Reduction of maintenance costs** *(Mentioned in literature [29] and mentioned in Interviews Inter_03, Inter_07)* In the case of rewriting and refinement of requirements, pre-RS traceability provides additional background information, refinement history, or persons involved.

b) Product-specific benefits: support the quality of a RS and thus the quality of the product.

B5 **Improve product/software quality** *(Mentioned in literature [3], [10], [28], [29], [31], [43], [45], [46] and mentioned in Interview Inter_07)* Keeping the RS updated and on a high level of quality is essential, especially if the system becomes more complex (UC4). Inter_07 reported that an important quality aspect is the reproducibility of requirements. Inter_07 reports about the development of strategic requirements by having a factual base incl. their input parameters on which the strategic expression builds.

"It always starts with a reliable basis of facts, then there is a strategy on top of it [...]. This reliable basis of facts, it would help and also be necessary to have this reproducible, that always the same basis comes out if you give in the same input parameters." Translated quote3 in VIII-D from Inter_07

B6 **Support reusability of requirements** *(Mentioned in literature [3], [4], [11], [29], [31] and mentioned in Interview Inter_10)* Knowing the origin of a requirement (UC1) allows to conclude whether the requirement can be applied to other products, especially within a company that provides multiple related products (UC12).

B7 **Reveal tacit knowledge** *(Mentioned in literature [44], [53], [54])* On the one hand, tacit knowledge is the knowledge that is only in the mind of a stakeholder and remains unspoken, and never explicitly documented. On the other hand, it can be knowledge of the requirements engineer that flows into the RS. Literature uncovered that requirements without trace links to their origin artifacts can be based on tacit knowledge. Pre-RS traceability makes such implicit knowledge explicit and allows to build upon it. But we could not find evidence for this benefit within the interviews, one reason may be that the interviewees are not aware of this benefit. However, further interviews are necessary here.

B8 **Finding missing requirements** *(Mentioned in literature [29] and mentioned in Interview Inter_06)* The presence of relevant stakeholder statements or other sources not linked to the RS may reveal missing requirements and therefore need further investigation.

B9 **Finding unnecessary requirements** *(Mentioned in literature [29])* Pre-RS traceability uncovers requirements without a trace back to their origin. These requirements need further analysis and if they are not based on tacit knowledge they may be unnecessary requirements. No interviewee mentioned that benefit, but Inter_01, Inter_02, Inter_04 and Inter_06 report requirements received from the stakeholder that were not relevant for the desired product.

V. Discussion

The motivation of users to adopt pre-RS traceability is driven by their expectation. [16] reported that *"pre-requirements traceability appears slightly more supported, but there are still multiple expectations as to what traceability should assist with at a practical project level and unproven actual benefits."* Within this article we sorted through

the published literature to identify those benefits that peer-reviewed research have corroborated as evident and linked them to specific use cases in which they manifest. Interviews with affected people expanded the results from the literature. We identified benefits for pre-RS traceability in relation with characteristics of users and environment.

Analyzing the literature reveals that pre-RS Traceability is often only handled as sub topic of traceability or research focus on specific solutions of realizing pre-RS traceability. But the interview results show how few concrete solutions creating and maintaining trace links to sources of requirements are used because no suitable strategy was found. Similar to [3] this paper describes benefits and use cases of pre-RS traceability, but focuses more on the current situation, because we need more research to deepen and expand the knowledge about the usage and benefits of pre-RS traceability within the industry. Therefore, further research needs to be done to provide a solid understanding as a basis for comprehensive pre-RS traceability solutions for industrial implementation to narrow the gap.

What surprised us during the SLR was that while use cases and benefits have been subject of prior research, another possible motivation was mostly lacking from the body of research, which are the consequences of poor pre-RS traceability. Studies uncovered that the main reason for poor requirements traceability and consequentially, for poor post-RS traceability, is a lack of pre-RS traceability, especially in long-term projects. Missing documentation about the creation process or the context leads to "black box" requirements, which prevent adequate requirements evolution as one consequence [5], [7], [13], [14].

However, the benefits gained by pre-RS traceability depend on the environment and the users involved. Therefore, both factors (users and environment) need to be observed and construed in order to successfully develop or implement a traceability strategy.

A. Future research topics

Based on the results of our research, the following research topics can be derived.

a) Measuring the impact: Some of the frequently presented categories that motivate pre-RS traceability adoption may seem easy to convey to practitioners considering traceability practices in principle. But the inevitable question, frequently asked in the interviews that ultimately convinces decision makers to commit to full traceability is *how impactful are these changes in our process metrics and measures for software quality?*. Some work on such a quantification, and a cost/benefit analysis is reported in [29].

b) Consequences of poor/lacking pre-RS traceability: Related to the cost/benefit analysis mentioned in the previous paragraph, it is important to explicate the costs related to poor documentation of traces. This is partly implied by benefit B4, the reduction of maintenance costs.

c) Appropriate traceability strategies: It would be of benefit to gain a deeper understanding of details about particular relations between use cases, characteristics of projects and users and benefits. Which strategies of applying pre-RS traceability are suitable dependent on particular characteristics of projects and users involved? An exemplary assumption that is in need of rigorous analysis would be *What project characteristics require a specific pre-RS traceability strategy?* Ahmad and Ghazali for example analyzed requirements traceability in the context of small projects with limited budget [11].

d) Making success measurable: Inter_10 suggested using trace links to validate requirements against stakeholder needs, in order to get a base for measurable success. This is particularly helpful when stakeholder involvement is time-consuming and expensive and the requirements engineer wants to validate requirements in advance. A deeper dive into the topic of using pre-RS traceability to measure project success and how it works in the industry highlights a possible research direction.

e) Deepen the knowledge of benefits: Two benefits, the explication of tacit knowledge (B7) and the identification of unnecessary requirements (B9) are identified as benefits of pre-RS traceability by the literature. In contrast, no participant of the interviews mentioned these benefits. We are aware that our interview data is limited with regards to possible claims to generalizability in this regard, but it still gives indication for potential future research. How can we make users aware of these benefits and support them in achieving these benefits?

VI. Threats to Validity

To identify and address possible limitations and threats to validity we built on the insights of Zhou et al. [55].

a) Construct Validity: During the SLR about pre-RS traceability, we identified that benefits of pre-RS traceability are often not clear. This appeared to be tightly related to the benefits being highly dependent on different use cases and other influencing factors. We defined two research questions to investigate on this topic. Based on these research questions we may not cover all influencing factors of benefits of pre-RS traceability, but the results provide a comprehensive list of use cases and benefits as a base for future research. Construct confounding is a possible threat to construct validity that is especially prevalent in an SLR because multiple authors may use differing terminology for the same constructs, but can also naturally occur in a qualitative survey. We mitigated this threat by having our coding checked through intercoder agreement, and by doing member checking in the qualitative survey.

b) Internal Validity: In the context of the SLR, the selection of search terms may not cover all relevant articles. We addressed this through snowballing by backward and forward search to identify more relevant articles. Regarding the qualitative survey, the sampling of interview candidates was driven by theoretical sampling as much as possible to cover a variety of different roles of users and types of industry. However, some availability constraints and our access to the field remains a limitation for a potential sampling bias. We didn't sample by industry, which would also have been a possible approach to focus on the different environments. This could be a starting point for further research. To increase

the internal validity of our research, we took the following measures. We triangulated our research results using multiple methods by combining the SLR and a qualitative survey (method- and data-triangulation), and avoided researcher bias by having multiple peer debriefing [21] sessions about the research design. In these peer debriefings, the audit trail of our research was thoroughly questioned. We also implemented investigator triangulation by having two interrater reliability sessions on a sample of the interviews with two researchers each, which helped refine the code system with better definitions of codes and clearer distinctions between different themes.

c) External validity: Regarding the SLR, we included articles published between 1992 and 2020. Articles published outside of this time span may affect the generalizability of the SLR results. Regarding the qualitative survey, we were more likely to talk to practitioners who are motivated to share their knowledge and make changes, and less to those who are already frustrated by the topic, and hesitant to engage in it further within the scope of our research project. We explicitly tried to cover as much space as possible w.r.t. industries included in our survey (i.e. automotive, finance, e-commerce). However the sample size was simply not large enough to allow for broad generalization, hence the sample, to an extent, impedes external validity.

d) Conclusion validity: The whole research process was defined and documented by a research protocol. The research protocol was continuously reviewed by an experienced researcher within peer debriefing sessions.

VII. Conclusions

Pre-requirement specification (pre-RS) traceability links requirements with their source and post-RS traceability links requirements with artifacts build on the requirements specification (RS). During a systematic literature review (SLR) on challenges and existing solutions of pre-RS traceability, we identified that the benefits are not clear. So we defined two research questions to address this issue, *RQ1: Which use cases benefit from pre-RS traceability?* and *RQ2: What are the expected benefits?*

To answer the research questions we did a literature review and conducted a qualitative survey consisting of ten semi-structured interviews with practitioners, like requirements engineers, product owners, and consultants to get more insight from the industry.

The benefits of pre-RS traceability depend on particular use cases, the practitioners involved within a project, and the project environment. We identified 13 use cases and nine benefits of pre-RS traceability. The use cases can be divided into three categories: use cases within the project team, for outside communication, or for follow-ups and reports. For a use case to be beneficial, the characteristics of practitioners involved and the project environment must be taken into account. The qualitative survey reveals, consultants in particular are more aware of the benefits of pre-RS traceability than others, as they can familiarize themselves more quickly with the help of trace links between requirements and their origin and thus also provide faster and better support.

Our study provides an overview of the benefits of pre-RS traceability and how they can be achieved. More research needs to be done to identify more insight about the relation of use cases, characteristics of practitioners and environment, and benefits. Research to find a specific and suitable strategy of pre-RS traceability can build on it.

Acknowledgment

This work was supported by a grant from Software Campus through the Federal Ministry of Education and Research[4]. Furthermore, this work should be part of a cumulative dissertation. We would like to thank all anonymous interview partners. Finally, thanks to Georg Schwarz for participating in the interrater reliability session and for providing feedback on the paper to improve its presentation. We are also deeply thankful to Barbara Paech for reviewing a draft of this paper and providing valuable feedback.

References

[1] O. C. Z. Gotel, "Requirements Traceability," main Report, Centre for Requirements and Foundations, Oxford University Computing Lab, Oxford, 1992.

[2] A. for review, "The code system of a systematic literature review on pre-requirements specification traceability," Tech. Rep. Finde article in complementary data, Anonymized for review, 2020.

[3] S. Altaf, A. Shah, N. Imtiaz, A. S. Shah, and S. F. Ahmed, "Visualization representing benefits of pre-requirement specification traceability," *International Journal of Engineering & Technology*, vol. 7, p. 44, 2018.

[4] B. Ramesh, "Factors Influencing Requirements Traceability Practice," *Communications of the ACM*, vol. 41, no. 12, pp. 37–44, 1998.

[5] R. Ravichandar, J. D. Arthur, and M. Pérez-Quiñones, "Pre-Requirement Specification Traceability: Bridging the Complexity Gap through Capabilities," *International Symposium on Grand Challenges in Traceability, TEFSE/GCT 2007*, p. 10, 2007.

[6] G. Spanoudakis and A. Zisman, "Software traceability: a roadmap," in *Handbook of Software Engineering and Knowledge Engineering*, pp. 395–428, WORLD SCIENTIFIC, 2005.

[7] O. C. Z. Gotel and C. W. Finkelstein, "An analysis of the requirements traceability problem," in *Proceedings of IEEE International Conference on Requirements Engineering*, pp. 94–101, 1994.

[8] R. Torkar, T. Gorschek, R. Feldt, M. Svahnberg, R. Uzair Akbar, and K. Kamran, "Requirements Traceability : A Systematic Review and Industry Case Study," *International Journal of Software Engineering and Knowledge Engineering*, vol. 22, no. 3, pp. 385–433, 2012.

[9] E. Bouillon, P. Mäder, and I. Philippow, "A Survey on Usage Scenarios for Requirements Traceability in Practice," in *Requirements Engineering: Foundation for Software Quality*, Lecture Notes in Computer Science, pp. 158–173, Springer, 2013.

[10] P. Rempel, P. Mäder, and T. Kuschke, "An empirical study on project-specific traceability strategies," in *2013 21st IEEE International Requirements Engineering Conference (RE)*, pp. 195–204, IEEE, 2013.

[11] A. Ahmad and M. A. Ghazali, "Documenting Requirements Traceability Information for Small Projects," in *2007 IEEE International Multitopic Conference*, pp. 1–5, IEEE, 2007.

[12] D. M. Fernández, S. Wagner, M. Kalinowski, M. Felderer, P. Mafra, A. Vetrò, T. Conte, M.-T. Christiansson, D. Greer, C. Lassenius, T. Männistö, M. Nayabi, M. Oivo, B. Penzenstadler, D. Pfahl, R. Prikladnicki, G. Ruhe, A. Schekelmann, S. Sen, R. Spinola, A. Tuzcu, J. L. de la Vara, and R. Wieringa, "Naming the pain in requirements engineering," *Empirical Software Engineering*, vol. 22, no. 5, pp. 2298–2338, 2017.

[13] O. Gotel and A. Finkelstein, "Contribution structures [Requirements artifacts]," in *Proceedings of 1995 IEEE International Symposium on Requirements Engineering (RE'95)*, pp. 100–107, IEEE, 1995.

[4]BMBF: Bundesministerium fuer Bildung und Forschung https://www.bmbf.de/en/index.html

[14] O. Gotel and A. Finkelstein, "Revisiting requirements production," *Software Engineering Journal*, vol. 11, pp. 166–182, May 1996.

[15] B. Ramesh and M. Jarke, "Toward reference models for requirements traceability," *IEEE Transactions on Software Engineering*, vol. 27, no. 1, pp. 58–93, 2001.

[16] P. Mäder, O. Gotel, and I. Philippow, "Motivation Matters in the Traceability Trenches," in *2009 17th IEEE International Requirements Engineering Conference*, pp. 143–148, Aug. 2009.

[17] J. Goguen, "Social issues in requirements engineering," in *[1993] Proceedings of the IEEE International Symposium on Requirements Engineering*, pp. 194–195, IEEE, Jan. 1993.

[18] M. Serrano and J. C. S. do Prado Leite, "A Rich Traceability Model for Social Interactions," in *Proceedings of the 6th International Workshop on Traceability in Emerging Forms of Software Engineering*, TEFSE '11, pp. 63–66, ACM, 2011.

[19] B. Kitchenham, "Procedures for Performing Systematic Reviews," Technical Report 1.0, Keele University, 2004.

[20] J. Krause, A. Kaufmann, and D. Riehle, "The code system of a systematic literature review on pre-requirements specification traceability," Tech. Rep. CS-2020-02, Technische Fakultät, 2020.

[21] S. Spall, "Peer Debriefing in Qualitative Research: Emerging Operational Models," *Qualitative Inquiry*, vol. 4, no. 2, pp. 280–292, 1998.

[22] J. Webster and R. T. Watson, "Analyzing the Past to Prepare for the Future: Writing a Literature Review," *Management Information Systems Research Center*, vol. 26, no. 2, pp. 13–23, 2002.

[23] J. M. Corbin and A. Strauss, "Grounded theory research: Procedures, canons, and evaluative criteria," *Qualitative Sociology*, vol. 13, pp. 3–21, Mar. 1990.

[24] H. Jansen, "The Logic of Qualitative Survey Research and its Position in the Field of Social Research Methods," *Forum Qualitative Sozialforschung / Forum: Qualitative Social Research*, vol. 11, no. 2, 2010.

[25] D. Wood, M. Christel, and S. Stevens, "A multimedia approach to requirements capture and modeling," in *Proceedings of IEEE International Conference on Requirements Engineering*, pp. 53–56, IEEE, 1994.

[26] S. Jayatilleke and R. Lai, "A systematic review of requirements change management," *Information and Software Technology*, vol. 93, pp. 163–185, 2018.

[27] J. Dick, E. Hull, and K. Jackson, *Requirements Engineering*. Switzerland: Springer International Publishing, 4 ed., 2017.

[28] K. Souali, O. Rahmaoui, and M. Ouzzif, "An Overview of Traceability: Towards a general multi-domain model," *Advances in Science, Technology and Engineering Systems Journal (ASTES)*, vol. 2, no. 3, pp. 356–361, 2017.

[29] K. E. Wiegers and J. Beatty, *Software Requirements*. Redmond: Microsoft Press, 2013.

[30] A. Finkelstein, "Requirements engineering: a review and research agenda," in *Proceedings of 1st Asia-Pacific Software Engineering Conference*, pp. 10–19, IEEE, 1994.

[31] S. Winkler and J. von Pilgrim, "A survey of traceability in requirements engineering and model-driven development," *Software & Systems Modeling*, vol. 9, pp. 529–565, Sept. 2010.

[32] F. A. C. Pinheiro, "Requirements Traceability," in *Perspectives on Software Requirements*, pp. 91–113, Springer US, 2004.

[33] J. Castro, R. Pinto, A. Castor, and J. Mylopoulos, "Requirements Traceability in Agent Oriented Development," in *Software Engineering for Large-Scale Multi-Agent Systems* (A. Garcia, C. Lucena, F. Zambonelli, A. Omicini, and J. Castro, eds.), Lecture Notes in Computer Science, pp. 57–72, Springer, 2002.

[34] S. Haidrar, A. Anwar, and O. Roudies, "Towards a generic framework for requirements traceability management for SysML language," in *2016 4th IEEE International Colloquium on Information Science and Technology (CiSt)*, pp. 210–215, IEEE, 2016.

[35] H. Ossher, D. Amid, A. Anaby-Tavor, R. Bellamy, M. Callery, M. Desmond, J. De Vries, A. Fisher, S. Krasikov, I. Simmonds, and C. Swart, "Using tagging to identify and organize concerns during pre-requirements analysis," in *2009 ICSE Workshop on Aspect-Oriented Requirements Engineering and Architecture Design*, pp. 25–30, IEEE, May 2009.

[36] K. Pohl, "PRO-ART: enabling requirements pre-traceability," in *Proceedings of the Second International Conference on Requirements Engineering*, pp. 76–84, 1996.

[37] O. Gotel and A. Finkelstein, "Extended requirements traceability: results of an industrial case study," in *Proceedings of ISRE '97: 3rd IEEE International Symposium on Requirements Engineering*, pp. 169–178, IEEE, 1997.

[38] F. Pinheiro and J. Goguen, "An object-oriented tool for tracing requirements," *IEEE Software*, vol. 13, no. 2, pp. 52–64, 1996.

[39] H. Kitapci and B. W. Boehm, "Formalizing Informal Stakeholder Decisions–A Hybrid Method Approach," in *2007 40th Annual Hawaii International Conference on System Sciences (HICSS'07)*, pp. 283c–283c, IEEE, 2007.

[40] V. Shukla, G. Auriol, and C. Baron, "Integrated requirement traceability, multiview modeling, and decision-making: A systems engineering approach for integrating processes and product," in *2012 IEEE International Systems Conference SysCon 2012*, pp. 1–5, IEEE, 2012.

[41] H. M. Hao and A. Jaafar, "Tracing user interface design pre-requirement to generate interface design specification," in *2009 International Conference on Electrical Engineering and Informatics*, vol. 01, pp. 287–292, IEEE, 2009.

[42] R. Wohlrab, J.-P. Steghöfer, E. Knauss, S. Maro, and A. Anjorin, "Collaborative Traceability Management: Challenges and Opportunities," in *2016 IEEE 24th International Requirements Engineering Conference (RE)*, pp. 216–225, 2016.

[43] A. Espinoza, P. P. Alarcon, and J. Garbajosa, "Analyzing and Systematizing Current Traceability Schemas," in *2006 30th Annual IEEE/NASA Software Engineering Workshop*, pp. 21–32, Apr. 2006.

[44] A. Stone and P. Sawyer, "Identifying tacit knowledge-based requirements," *IEE Proceedings - Software*, vol. 153, no. 6, pp. 211–218, 2006.

[45] K. Mohan and B. Ramesh, "Managing variability with traceability in product and service families," in *Proceedings of the 35th Annual Hawaii International Conference on System Sciences*, pp. 1309–1317, IEEE, 2002.

[46] C. Lee, L. Guadagno, and X. Jia, "An agile approach to capturing requirements and traceability," in *Proceedings of the 2nd International Workshop on Traceability in Emerging Forms of Software Engineering (TEFSE 2003)*, vol. 20, 2003.

[47] G. Urrego-Giraldo, "Agent-based knowledge keep tracking," in *Proceedings Fifth IEEE Workshop on Mobile Computing Systems and Applications*, pp. 8–16, IEEE, 2003.

[48] K. Souali, O. Rahmaoui, and M. Ouzzif, "An overview of traceability: Definitions and techniques," in *2016 4th IEEE International Colloquium on Information Science and Technology (CiSt)*, pp. 789–793, 2016.

[49] H. Dubois, M.-A. Peraldi-Frati, and F. Lakhal, "A Model for Requirements Traceability in a Heterogeneous Model-Based Design Process: Application to Automotive Embedded Systems," in *2010 15th IEEE International Conference on Engineering of Complex Computer Systems*, pp. 233–242, IEEE, 2010.

[50] J. Hayes, A. Dekhtyar, and S. Sundaram, "Advancing candidate link generation for requirements tracing: the study of methods," *IEEE Transactions on Software Engineering*, vol. 32, no. 1, pp. 4–19, 2006.

[51] P. Grunbacher, M. Halling, S. Biffl, H. Kitapci, and B. Boehm, "Repeatable quality assurance techniques for requirements negotiations," in *36th Annual Hawaii International Conference on System Sciences, 2003. Proceedings of the*, pp. 9 pp.–, IEEE, 2003.

[52] S. Imtiaz, N. Ikram, and S. Imtiaz, "Impact Analysis from Multiple Perspectives: Evaluation of Traceability Techniques," in *2008 The Third International Conference on Software Engineering Advances*, pp. 457–464, IEEE, 2008.

[53] G. R., M. L., N. B., P. P., d. R. A. N., R. M., S. P., W. A., and Y. H., "Making Tacit Requirements Explicit," in *2009 Second International Workshop on Managing Requirements Knowledge*, pp. 40–44, IEEE, 2009.

[54] A. Stone and P. Sawyer, "Exposing Tacit Knowledge via Pre-Requirements Tracing," in *14th IEEE International Requirements Engineering Conference (RE'06)*, pp. 353–354, IEEE, 2006.

[55] X. Zhou, Y. Jin, H. Zhang, S. Li, and X. Huang, "A Map of Threats to Validity of Systematic Literature Reviews in Software Engineering," in *2016 23rd Asia-Pacific Software Engineering Conference (APSEC)*, pp. 153–160, Dec. 2016.

[56] M. C. Panis, "Successful Deployment of Requirements Traceability in a Commercial Engineering Organization...Really," in *Proceedings of the 2010 18th IEEE International Requirements Engineering Conference*, RE '10, pp. 303–307, IEEE Computer Society, 2010.

[57] A. Bogner, B. Littig, and W. Menz, *Interviews mit Experten: Eine praxisorientierte Einführung*. Springer-Verlag, July 2014.

VIII. Appendix

A. List of articles included

The list of articles included in the literature review is sorted alphabetically.

[49] A Model for Requirements Traceability in a Heterogeneous Model-Based Design Process: Application to Automotive Embedded Systems
[25] A multimedia approach to requirements capture and modeling
[18] A rich traceability model for social interactions
[31] A survey of traceability in requirements engineering and model-driven development
[9] A Survey on Usage Scenarios for Requirements Traceability in Practice
[26] A systematic review of requirements change management
[50] Advancing candidate link generation for requirements tracing: the study of methods
[47] Agent-based knowledge keep tracking
[46] An Agile Approach to Capturing Requirements and Traceability
[7] An analysis of the requirements traceability problem
[10] An empirical study on project-specific traceability strategies
[38] An object-oriented tool for tracing requirements
[48] An overview of traceability: Definitions and techniques
[28] An Overview of Traceability: Towards a general multi-domain model
[43] Analyzing and Systematizing Current Traceability Schemas
[42] Collaborative Traceability Management: Challenges and Opportunities
[13] Contribution structures [Requirements artifacts]
[11] Documenting Requirements Traceability Information for Small Projects
[54] Exposing Tacit Knowledge via Pre-Requirements Tracing
[37] Extended requirements traceability: results of an industrial case study
[4] Factors influencing requirements traceability practice
[39] Formalizing Informal Stakeholder Decisions–A Hybrid Method Approach
[44] Identifying tacit knowledge-based requirements
[52] Impact Analysis from Multiple Perspectives: Evaluation of Traceability Techniques
[40] Integrated requirement traceability, multiview modeling, and decision-making: A systems engineering approach for integrating processes and product
[53] Making Tacit Requirements Explicit
[45] Managing variability with traceability in product and service families
[16] Motivation Matters in the Traceability Trenches
[5] Pre-Requirement Specification Traceability: Bridging the Complexity Gap through Capabilities
[36] PRO-ART: enabling requirements pre-traceability
[51] Repeatable quality assurance techniques for requirements negotiations
[27] Requirements Engineering (Fourth Edition)
[30] Requirements engineering: a review and research agenda
[1] Requirements Traceability
[32] Requirements Traceability
[33] Requirements Traceability in Agent Oriented Development
[8] Requirements traceability state-of-the-art: a systematic review and industry case study
[14] Revisiting requirements production
[29] Software requirements
[56] Successful Deployment of Requirements Traceability in a Commercial Engineering Organization...Really
[15] Toward reference models for requirements traceability
[34] Towards a generic framework for requirements traceability management for SysML language
[41] Tracing user interface design pre-requirement to generate interface design specification
[35] Using tagging to identify and organize concerns during pre-requirements analysis
[3] Visualization representing benefits of pre-requirement specification traceability

B. Preliminary questionnaire

To get more information about potential interview candidates we create an preliminary questionnaire that included questions about contact information and role information.

- What is your name? *Response type: free text*
- Which email would you like to be contacted at? *Response type: email address*
- Do you agree to the audio track of the interview being recorded? *Response type: yes or no*
- Which activities are part of your current work with requirements? *Response type: multi selection*
  - I analyze customer requirements/information and write requirement specifications
  - I work with requirements specifications that have already been written in order to improve them in terms of quality.
  - I advise other people on their work in requirements engineering within my current company.
  - I advise other people on their requirements engineering work in other companies.
  - Other... (free text input possible)
- What type of project are you involved in? *Response type: multi selection*
  - Agile project(s)
  - Plan-driven project(s)
  - Other... (free text input possible)
- Do you have any comments? *Response type: free text*

C. Template for semi-structured interviews

Based on the guidelines of Bogner et al. [57] we created a template for the semi-structured interviews, which we adjusted for each interviewee based on the results of the preliminary questionnaire.

1) Introduction consisting of some general information about the interviewer, the topic of the interview, conditions (duration, audio recording, etc.), processing of interview data, etc.
2) Main part consisting of questions addressing following topics
   - RS creation activities
     - Analyzing & extracting relevant information from different sources (*What are the practices to identify, analyse and extract relevant information?*
     - Synthesize the relevant information into natural language requirements (*What are the practices to create natural language requirements?*)
   - RS quality criteria
     - Techniques to achieve quality criteria: completeness, consistency & traceability (*What are practices to achieve completeness, consistency, and traceability?*)
     - Willingness to make efforts to achieve quality (*When is it worth spending more time writing high-quality requirements?*)
3) End of interview containing following open questions
   - *Can you recommend other people to me?*
   - *Do you have the impression that we have forgotten points that you think are relevant? Anything else you would like to add?*

D. Quotes from interviews

The interviews of the qualitative survey were performed in German. Quotations have been translated in the article for a legible text flow. Below you will find the quotations in their original wording.

Quote1: *"Und ich natürlich als Product Owner schlichtweg auch Sicherheit bekomme ist das wirklich so angekommen, wie es ursprünglich auch formuliert wurde und es gibt [...] auch dem Entwicklungsteam beispielsweise Sicherheit, also, ich glaube schon, dass es vor allem im Hinblick auf Messbarkeit von Erfolgen sehr wichtig ist, immer wieder dann auch diesen Pfad bis zum ursprünglichen Entstehen zurückverfolgen zu können und dann halt auch wirklich Aussagen zu können, hat es jetzt die Welt verbessert, war das ein Schuss in den Ofen, [...]"* (Inter_10)

Quote2: *"[...] wenn man die sauberen Ableitungspfade hat, gibt's hinterher kein böses Blut, weil jemand drüber diskutieren kann, warum das Stück Software so aussieht, wie es aussieht, weil ich zu jedem bisschen sagen kann. Das gibt's weil du damals am 25. gesagt hast, du willst es so haben und es wird auch schneller gehen."* (Inter_01)

Quote3: *"Das heißt es fängt immer an mit einer Faktenbasis, dann sitzt eine Strategie drauf, worauf man dann auch eine strategische Ausprägung macht und das ganze gestaltest. Und diese Faktenbasis, da wäre es wirklich Hilfe und auch notwendig, dieses reproduzierbare zu haben, dass immer die gleiche Basis raus kommt, wenn du die gleichen Input Parameter rein gibst."* (Inter_07)