Traceability Research at the Requirements Engineering Conference: Results and Extracted Data

Technical Report

Sunil Nair, Jose Luis de la Vara, and Sagar Sen
Centre for Software V&V, Simula Research Laboratory
P.O. Box 134, 1325 Lysaker, Norway
{sunil, jdelavara, sagar}@simula.no

***Abstract*—Traceability between development artifacts and mainly from and to requirements plays a major role in system lifecycle, supporting activities such as validation, change impact analysis, and regulation compliance. Many researchers have been working on this topic and have published their work throughout editions of the Requirements Engineering Conference. This paper aims to analyze the research on traceability published at this conference and provides insights into its contribution to the traceability area. For this purpose, papers on traceability in the proceedings of the conference have been reviewed for determination of (1) the traceability topics studied, (2) the challenges addressed, (3) the contributions made, (4) the tools features developed to support traceability, (5) the types of systems considered, (6) the types of artifacts traced, (7) the empirical methods used, and (8) the leaders in production. The paper also discusses the evolution of the topic at the conference, compares the results with those reported in other publications, and proposes aspects on which further research should be conducted.**

Keywords-traceability, challenge, artifact, artifact relationship, tool support, empirical validation;

A Review of Traceability Research at the Requirements Engineering Conference

Sunil Nair, Jose Luis de la Vara, and Sagar Sen
Certus Centre for Software V&V, Simula Research Laboratory
P.O. Box 134, 1325 Lysaker, Norway
{sunil, jdelavara, sagar}@simula.no

Abstract—**Traceability between development artefacts and mainly from and to requirements plays a major role in system lifecycle, supporting activities such as validation, change impact analysis, and regulation compliance. Many researchers have been working on this topic and have published their work throughout editions of the Requirements Engineering Conference. This paper aims to analyse the research on traceability published at this conference and provide insights into its contribution to the traceability area. For this purpose, papers on traceability in the proceedings of the conference have been reviewed for determination of (1) the traceability topics studied, (2) the challenges addressed, (3) the contributions made, (4) the tools features developed to support traceability, (5) the types of systems considered, (6) the types of artefacts traced, (7) the empirical methods used, and (8) the leaders in production. The paper also discusses the evolution of the topic at the conference, compares the results with those reported in other publications, and proposes aspects on which further research should be conducted.**

Keywords-traceability, challenge, artefact, artefact relationship, tool support, empirical validation;

I. INTRODUCTION

Traceability can be defined as the degree to which a relationship can be established between two or more products of the development process, especially products having a predecessor-successor or master-subordinate relationship to one another [29]. Tracing in system development can be targeted at different aspects [54][61] such as system verification and validation (V&V), change management, and regulatory compliance. The importance of system traceability has been widely recognised, and it is a practice prescribed in many development standards [11].

Traceability research has greatly focused on requirements traceability, aiming to study how to describe and follow the life of a requirement, in both forward and backward direction [20]. Many researchers have contributed to the area for the last two decades [23][61][63], providing solutions in the form of methods, tools and a better understanding of traceability needs and challenges. Traceability has been an important topic at the requirements engineering (RE) conference since its inception.

The purpose of this paper is provide insights into traceability research at the RE conference and how it has contributed to the area. To this end, we have reviewed papers on traceability published in the proceedings of the main conference. The review has been performed in the form of a systematic literature review (SLR) [34], a documented and repeatable process through which the literature on a given subject is examined and the state of knowledge is recorded.

From a set of 70 papers, we have determined (1) the traceability topics studied, (2) the challenges addressed, (3) the contributions made, (4) the tools features developed to support traceability, (5) the types of systems considered, (6) the types of artefacts traced, (7) the empirical methods used for evaluation, and (8) the leaders in production. This information has also allowed us to analyse how traceability research has evolved and progressed throughout the conference editions.

Related work in literature mainly corresponds to other secondary studies on traceability (e.g., [63]) and papers discussing challenges for traceability (e.g., [23]). What differentiates this paper is its focus on the RE conference. To our knowledge, it is also the most recent SLR on traceability, and the one with the highest number of primary studies. Consequently, we consider that the results presented correspond to the widest and most accurate analysis of traceability research that has been provided up to date.

As shown below, we have used related work as input for discussion in relation to (1) comparison of the research at this conference with that conducted in general within the are of traceability, and (2) determination of challenges that have not been addressed or solved yet. This analysis has allowed us to argue why and how traceability research at the RE conference has contributed to the progress of the area, as well as what aspects should be studied in the future.

The rest of the paper is organized as follows. Section II describes the research method applied. Section III shows the results from the review, whereas Section IV discusses them. Finally, Section V presents our conclusions.

II. RESEARCH METHOD

A SLR is a means of identifying, evaluating and interpreting available literature relevant to a particular research question or topic area [34]. When compared to ad hoc literature search, the main advantage of a SLR is that it provides a higher degree of confidence about covering the relevant literature, and minimizes subjectivity and bias.

The following subsections present the research questions formulated for our study and summarise the procedure for publications selection, data extraction and data synthesis.

A. Research Questions

Our overall goal was to evaluate how traceability research at the RE conference has contributed to the area. We formulated the following research questions (RQs):

RQ1) What topics within the traceability area have been studied?

RQ2) What specific challenges have been addressed?

RQ3) What contributions have been made to address the challenges?

RQ4) What tool features have been developed to support traceability?

RQ5) What types of systems have been considered?

RQ6) What types of artefacts have been traced?

RQ7) What empirical methods have been applied?

RQ8) Who has led research production?

B. *Publications Selection, Data Extraction and Synthesis*

We automatically searched (http://ieeexplore.ieee.org) for papers in the proceedings of the RE main conference that contained the word 'traceability' in the title, abstract or keywords. This resulted in a set of 76 papers.

A data extraction template was created in a spreadsheet with respect to the research questions formulated. For RQ7, we extracted the authors' institutions and the country of the institutions. We also included fields for the tittle of the papers, the conference year, and the authors' names.

The papers were then reviewed by dividing the workload among the three authors. Since our ultimate goal was to evaluate how the publications had contributed to the traceability area, we decided to exclude the papers for which we could not answer RQ3. For those papers, the three authors had to agree upon the exclusion. Six papers were excluded, and we obtained a final set of 70 primary studies.

Once all the papers were reviewed, we revised the spreadsheet in order to harmonize the data extracted by each author. As shown in Section III, we defined categories for grouping the data of RQ1-6. Details about the data extracted can be found in [44].

In relation to the limitations of this procedure, we might have missed some paper and thus some contribution to traceability. However, we consider this to be unlikely. If some contribution (relevant, beyond those made by the final set of primary studies) had been made in another paper, the authors would have very probably included the word 'traceability' in the fields checked in the automatic search.

Identifying the empirical method used in a paper can also be difficult because of the lack of details about the validation. Different authors can also have a different understanding (e.g., about what a case study is). We mitigated this threat by agreeing upon the definition of the empirical methods to distinguish (see Section III.G) before reviewing the papers.

Finally, it is always possible to miss some information in the papers reviewed for a SLR, especially for novices in SLRs or in the area under study. In our case, the first two authors had experience in SLR research [45], and the three authors had researched on RE and on traceability.

III. RESULTS

This section presents the results of the review. A subsection has been created for each RQ. All the papers matching the aspects analysed cannot be referred due to page limitations. Nonetheless, examples are provided.

A. *RQ1: Traceability Topics Studied*

We analysed the various overall topics researched in the conference within the theme of traceability and classified them into 10 groups (Figure 1). Some papers noted more than one group. The groups are as follows.

Post-requirement traceability (50% of the papers): tracking of requirements from their specification through both their development and maintenance lifecycle (e.g., for V&V analysis [26]).

Traceability automation (18.6%): automated traceability activities, such as creation of traces (e.g., [13]).

Pre-requirement traceability (17.1%): tracking of requirements from their specification to their origin (e.g., to the human source [22]).

Traceability in practice (12.9%): traceability management in real industrial settings (e.g., a company's approach for traceability [48]).

Change management (12.9%): management of artefact changes and their traces, and impact analysis (e.g., [66]).

Model traceability (11.4%): traces in and between models (e.g., between requirements models [59]).

Regulatory compliance (5.7%): traceability for demonstrating compliance to some regulation (e.g., for certification against RTCA DO178B [55]).

New approaches for maintaining traceability (4.3%): proposal of new approaches for recording traceability (e.g., through video recordings [62]).

Trade-off analysis (2.9%): traceability management during and for decision-making (e.g., [9]).

Traceability in new development contexts (1.4%): e.g., for chemical and plastic engineering in the automotive industry [31].

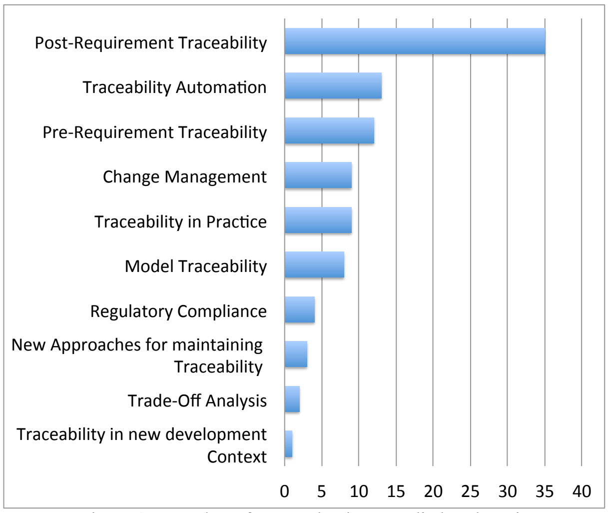

Figure 1. Number of papers that have studied each topic

B. *RQ2: Traceability Challenges Addressed*

We identified eight types of challenges and needs (Figure 2) specific to one or more topics from RQ1. Some papers did not address any specific challenges. We classified the challenges as follows.

Lack of knowledge and understanding about traceability (17.1% of the papers): the general lack of sufficient knowledge when dealing with traceability both in practice and research (e.g., [3]).

Guaranteeing satisfaction of requirements (12.9%): the need for assessing if requirements are met in successor artefacts such as a design specification (e.g., [9]).

Maintaining traceability when requirements evolve (12.9%): the challenge of maintaining traceability in the face of evolving requirements (e.g., [49]).

Effective representation of traceability information (8.6%): the need for presenting the traceability information in a clear and concise fashion (e.g., [59]).

Reducing the cost related to requirements traceability (8.6%): the challenge of cost-effectively maintaining traceability among, from or to requirements (e.g., [39]).

Impact of human factors and judgment (8.6%): the challenges faced when incorporating human judgment for traceability, and its factors (e.g., [13]).

Challenges in practice (5.7%): the various problems that practitioners face in real project settings (e.g., [5]).

Assessing the traceability maintained (1.4%): the importance of evaluating the traces captured (e.g., [16]).

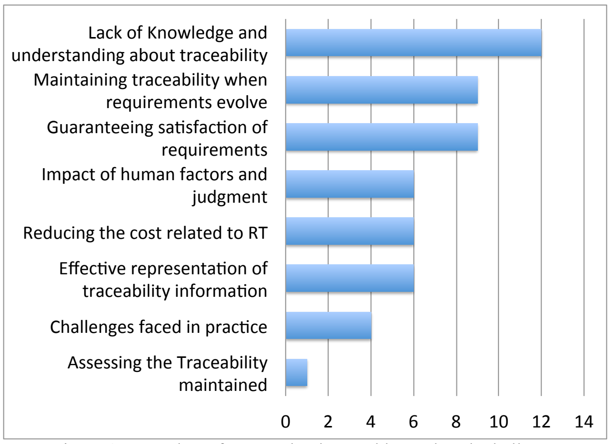

Figure 2. Number of papers that have addressed each challenge

C. *RQ3: Contributions to Traceability*

We classified the contributions made by the papers to address the challenges identified in RQ2 into three broad categories. Some papers had made more than one contribution.

Technical contributions (50% of the papers) have solved challenges by means of technical approaches or methodologies with tool support (e.g., a tool with explicit user defined links through the use of a matrix [9]).

Methodological contributions (31.4%) have solved challenges by means of new methods and approaches without tool support (e.g., a new traceability information model [37]).

Insights into practice and experience reports (21.4%) have provided details about real world traceability (e.g., a case study about the traceability practices in a company and what practices have worked [33]).

It must be noted that, from a general perspective, the contributions made map to the challenges faced and tackled (RQ2) and the tool features developed (RQ4).

D. *RQ4: Tool Features for Traceability*

35 papers presented some traceability tool. We extracted their key features and categorised them. The percentage of the features is the ratio to the total number of papers that proposed some tool. Some tools provided several features.

Traces lifecycle (34.3%): support for creating, maintaining and updating traces between various artefacts. (e.g., Ecolabor is a tool for using hypermedia to maintain traceability between different artefacts [62]).

Maintaining traceability between artefacts specific to requirements specification (28.5%): support for maintaining traces between requirements and managing them (e.g., TOOR is an object-oriented tool for recording traces between requirements [49]).

Automated traceability (20%): support for creating and maintaining traceability information (semi) automatically. (e.g., Poirot [39] implements a probabilistic approach to dynamically generate traceability links; work is still being performed for its extension and improvement).

Change management (11.4%): support for managing and updating changes in artefacts and hence their traceability information (e.g., a tool that extends on DOORS for change management [36]).

Requirements validation with traceability support (8.6%): for assessing and validating requirements with other artefacts and hence maintaining their traces (e.g., RESAT [28] allows user to automatically assess if a design description meets its requirements).

Model management with traceability support (8.6%): for creating and maintaining traces between and in models used in the development process (e.g., a tool for model merging and verification [57]).

Support for regulatory compliance (2.9%): for maintaining traceability towards compliance purposes (e.g., for compliance with DOD-STD-2176A [60]).

Project management (2.9%): features for control and monitoring of a project (e.g., Gantt charts generation [52]).

Traceability visualization (2.9%): support for visualization of traces maintained between artefacts (e.g., CREWS-EVE [24] offers multimedia support and animation to visualize traceability to test cases).

E. *RQ5: Types of Systems Subject to Traceability*

27 papers (38.7%) did not mention any specific type of system subject to traceability. For the rest, these were the types distinguished.

Information systems (32.9% of the papers; e.g., [37]), which store, process, and show data for their users.

Safety-critical systems (17.1%; e.g., [55]), whose failure may cause death or injury to people or harm to the environment.

Real-time embedded system (7.1%; e.g., [51]), which are subject to real-time constraints.

Non-software system (4.2%; e.g., [31]), such as the physical documents managed in an organization.

F. RQ6: Types of Artefacts Traced

For analysis of the types artefacts traced, we extracted information about the source and target of a trace.

Traces between requirements specification artefacts (60% of papers): high-level and low-level requirements (24.3%; e.g., business requirements and software requirements in [2]), requirements and source (17.1%; e.g., [68]), requirements and rationale (8.6%; e.g., the hazard mitigated by a requirement [10]), requirements versions (7.1%; e.g., [60]) requirements and person responsible (5.7%, e.g. [22]), requirements and creator (4.3%; e.g., [21]), requirements and contributor (4.3%; e.g., the people contributing to a model [56]), non-functional and functional requirements (2.9%; e.g., performance requirements [8]), and requirements and conflicts (1.4%; [33]).

Traces between requirements specification artefacts and other types of artefacts (58.6%): design (27.1%; e.g., [40]), testing artefacts (27.1%; e.g., [4]), code (25.7%; e.g., goals and code in [69]), development standards (4.3%; e.g., [27]), formal verification (2.9%; e.g., [57]), and testers (1.4%; [64]).

Traces between other types of artefacts (14.3% of papers): design and code (5.7%; e.g., [25]), design and testing (2.9%; e.g., [43]), design components (1.4%; [51]), design and responsible (1.4%; [51]), design and creator (1.4%; [49]), design and development standard (1.4%; [3]), testing and development standard (1.4%; [3]), and code and development standard (1.4%; [3]).

As shown above, the most frequent traces are between requirements and testing, requirements and design, requirements and code, high-level and low-level requirements, and requirements and source. This is in line with RQ1.

G. RQ7: Empirical Methods

The methods distinguished in the papers are: experiment (validation based on different treatments applied to or by different subjects); survey (validation based on practitioners' opinion and perspectives); field study (validation with data from real projects, but not during the execution of the project); case study (validation in real projects by practitioners different to the authors), and; action research (validation in real projects by the authors themselves).

48 papers (68.6%) have used some empirical method for evaluation. As show in Figure 3, the most frequent method has been **experiment** (20% of the papers; e.g., [16]), followed by **field study** (15.7% e.g., [68]), **action research** (15.7%; e.g., [60]) and **case study** (10%; e.g., [7]). **Survey** (7.1%; e.g., [20]) has been the empirical method least frequently applied. Two primary studies presented evaluations with more than one empirical method.

H. RQ8: Production Leaders

Among all the **institutions** that have published some paper on traceability at the RE conference, the University of Kentucky (9 papers) have the highest number of papers. Followed by the University of Toronto (7 papers), DePaul University (6 papers), Johannes Kepler University (4 papers), and City University London (4 papers).

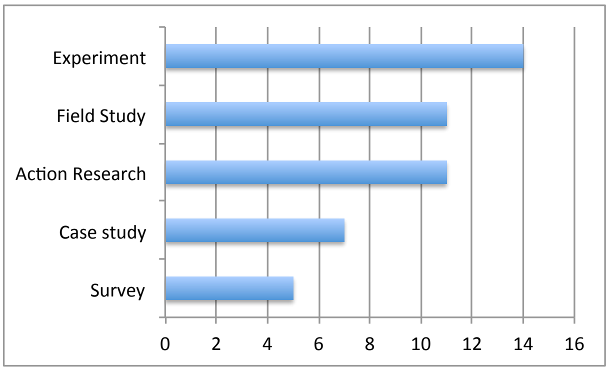

Figure 3. Number of papers applying each empirical method

In relation to the **origin** of the authors, the largest number of contributions comes from academia (70%). Practitioners have also published papers at the RE conference on their own (14.2%) and collaborated with academia (15.8%).

When analysing the **countries**, USA (37 papers), UK (12 papers), Canada (11 papers), and Germany (9 papers) have led production. 14 different countries have contributed to traceability research at the RE conference.

IV. Discussion

This section discusses how traceability research at the RE conference has evolved, how the results of our study relate to those reported in other secondary studies, and what aspects should be addressed by future research.

We would have liked to present a much more detailed discussion. However, this has not been possible due to page limitations and the need of presenting the information above in order to support our arguments.

A. Traceability Research Evolution at the RE Conference

To understand how traceability research has evolved at the RE conference over its 20 years, we performed a comparative analysis between all the editions and the last six editions (from 2007) on different aspects.

A total of 32 papers were published in the last six years, which is almost 46% of the total papers published. This indicates that traceability research has gained more attention at the conference in the near past. We identified that in the last six years there was no paper at the conference in the context of *new approaches for maintaining traceability*. This might mean that this topic has lost relevance and/or its associated challenges have already been tackled in previous research. 10 out of the 13 papers on *traceability automation* have been published since 2007, what shows that more and more effort is being spent on the topic.

We discovered that the challenge related to assessing the traceability maintained emerged during the last five years. This might indicate that interest in traceability quality is growing. Tool features specifically targeted at model management were first published in 2007, what suggests an increasing interest in model-driven engineering as a new framework for traceability.

With regards to the empirical validation, 72% of the experiments identified were conducted in the last six years. This indicates a strong focus on provision of evidence about the contributions made, as well as the maturity that traceability research is achieving.

Finally, we identified that seven papers published in the last six years were a result of industry-academia collaboration, whereas five had been published earlier. This suggests that there exists an increasing interest in innovation ad technology transfer in the area of traceability.

Table I shows what we have regarded as the main highlights related to traceability in each conference edition. It must be noted that no paper was selected from the 2001 edition. It is interesting to see the importance of empirical evaluation and tool support since the beginning of the conference, how some editions have strongly focused on some topics, or that the three most productive institutions did not published any paper until 2003.

B. *Comparison with other Secondary Studies*

As mentioned above, secondary studies related to our study can be found in the literature. We use their results to compare them with those reported in this paper and evaluate how the RE conference has contributed to the development and progress of the traceability area. Comparison with other secondary studies also allows us to compare the maturity of traceability research at the RE conference with the maturity of other fields and RE areas.

In general, the results presented in Section III are in line with those reported in other secondary studies that analysed:

- Traceability motivation (related to RQ1; [50][54][65])
- Traceability challenges (related to RQ2 and RQ3; [41][47] [53][63][65][67])
- Traceability approaches (related to RQ3 and RQ4; [1][18][23][63][65][67])
- Tool features for traceability (related to RQ5; [1][18][35] [58][61][63][65])
- Requirements interdependencies, types of traces, and types of artefacts traced (related to RQ6; [14][17][50][61][65])
- Empirical evaluation in traceability research (related to RQ7; [61][63][65])

It can be argued that traceability research at the RE conference provides an excellent overview of the area, and has significantly contributed to its progress. It is also true that a reason for the coincidence of results, although not as deeply and extensively analysed in this paper, could be that traceability research at the RE conference served as input for the other secondary studies. Even in this case, it shows that the relevance and contribution towards traceability at the conference, serves as a reference for many studies.

One aspect that has not been extensively addressed at the RE conference and that other researchers have regarded as relevant and thus started to focus on is model-driven traceability [1][18][58][67]. The minor presence of this aspect at the conference might be a result of its focus on requirements instead of, for instance, model-driven development approaches and tools in general.

TABLE I. HIGHLIGHTS REGARDING TRACEABILITY AT THE CONFERENCE

Edition	Highlights
1993	First RE edition. Traceability research started to be published.
1994	All the papers published so far at RE had applied empirical methods. Pre-requirements traceability was acknowledged as essential.
1995	Traceability started to be addressed in emerging topics such as goal-oriented RE.
1996	All the papers in this edition presented tool support.
1997	The first case study was published.
1998	A tool for distributed development was presented.
1999	Action research was the most frequent empirical method at RE so far.
2000	DOORS was extended for better traceability support.
2002	Change management was the main topic this year. The first paper applying more than one empirical method is presented (survey and case study).
2003	The first paper evaluating information retrieval techniques was presented. Practitioners were authors of 66% of the papers of the edition. The University of Kentucky published its first paper.
2004	All the papers of the edition presented some tool with features for automated traceability. DePaul University published its first paper.
2005	Lack of knowledge and understanding was the most addressed challenge in the edition. The University of Toronto published its first paper.
2006	The Poirot tool was presented.
2007	The first model management tool with traceability support was presented. Combination of several information retrieval techniques was proposed.
2008	80% of the papers of the edition had authors from different continents. All the papers studied post-requirements traceability.
2009	The most recent survey was published. A paper applied action research and experiment.
2010	Edition with the highest number of papers (seven). All of them applied some empirical method, and five corresponded to action research or case study.
2011	Experiment become the most frequently applied empirical method at RE. It was suggested to analyse traceability practices in other disciplines.
2012	A roadmap for future research was proposed.

The ratio of empirical studies is also higher in the results reported in this paper than in, for instance, [63]. A reason could be that we have considered the most recent traceability research (until 2012), which, as discussed above, has significantly mature in relation to empirical validation in the latest editions of the conference. The frequency in the use of empirical methods also shows that traceability research at the conference is more mature than the overall research in other RE areas (e.g., requirements specification [12]), and much more than other software engineering disciplines (e.g., safety assurance and certification [45]). Nonetheless, we consider that presentation of validation results can be improved. For example, more details should be provided about the studies design to increase rigorousness [30]. A reason for this weakness might be the page limitations at the conference.

Past studies (published before 2007; [1][6][14][20][46] [47][61][65]) discussed challenges and areas for future research, such as pre-requirement traceability, conflict analysis, requirements reuse, automated traceability, trace verification, and tailoring of approaches. We consider that, directly or indirectly, these challenges have been partially or completely tackled at the conference.

Finally, a demographic analysis of RE publications was presented in [15]. When comparing the results with this paper, they are similar. USA, UK, Canada, Germany, the University of Toronto, and City University London are highly ranked in both studies. The study considered publications until 2008. This might be a reason as to why the rest of most productive institutions on traceability at the RE conference do not appear in the study.

C. Aspects for Further Research

Several recent papers have discussed future research in the traceability area [23][32][58][67]. We aim to complement them by focusing on a small set of selected areas, based on our discussion on the review of traceability research at the RE conference and on our own current research. Such research is mainly focused on V&V of business-critical systems (e.g., systems computing taxes) and safety-critical systems (e.g., systems in the automotive, avionics and railway domain), and conducted in close collaboration with industry.

Traceability visualization. One interesting area is visualization of how requirements are realized by a test case for a large, complex system maintaining vast amounts of data. A high-level visualization can reveal requirements holes or gaps in a database or a test set in general. A possible formalism to visualize traceability could be feature models and classification tree models.

Consideration of more artefacts. Research has strongly focused on requirements traceability, but many other artefacts and traces exist in development projects, especially in the context of safety-critical systems [45]. It has to be studied as to how requirements traceability research could be adopted or adapted to a wider spectrum of artefact types.

Traces semantics for impact analysis. Related to the previous aspect, we think that more research on traces semantics for impact analysis is necessary. Practitioners will benefit on more guidance about how to deal with changes and what actions to perform, beyond only being aware of the artefacts potentially affected by a change.

Advanced empirical evaluation. Although this area has been recurrently mentioned in the literature, there are some specific issues that we consider that have not been discussed enough. Firstly, it is necessary to perform more dynamic validation [19], especially of automated traceability. Secondly, it will be important to replicate empirical studies in order to create a larger body of evidence. Approaches should also be compared more commonly, but very few studies have addressed comparison (e.g., regarding trace creation [16] and visualization [38]). Finally, it has been shown that case study and survey have been the empirical methods least frequently used at the RE conference. Since they allow researchers to analyse the perspectives and experiences of practitioners different to them, we think that more research must apply these methods.

Advanced tool support. For adoption in industry, automated traceability must be available or integrated with commercial tools. It is also important to study the confidence that can be placed in automated traceability, if its associated tools should be qualified for use in the development of critical systems, and how to do it.

V. CONCLUSIONS

This paper has presented the results of a SLR on traceability at the RE conference. The review has allowed us to provide new insights into the traceability research published at the conference, its focus, its maturity, its evolution, and its contribution to the traceability area.

The results indicate that traceability research at the conference has greatly focused on post-requirements traceability. The challenges most frequently addressed have been lack of knowledge and understanding about traceability, guaranteeing satisfaction of requirements, and maintaining traceability when requirements evolve. Most of the contributions have been technical, including a wide range of tool features and usually in the context of information or safety-critical systems.

Although both, traceability between requirements and other artefacts, and between requirements have been studied, specific traces of the former type have been most frequently reported. A high percentage of papers have applied empirical methods, and North America has led research production.

Traceability research at the conference has positively evolved. There is an increasing interest in automated traceability, model traceability, traceability quality, experimentation, and academia-industry collaboration. We think that the evolution shows the growth of the area in terms of maturity and interest in technology transfer.

When comparing the results of the review with those reported in other secondary studies, it can be argued that traceability research at the RE conference has provided a very good picture of the advances in the area. It has also shown a high degree of maturity, although need for more rigorousness might be claimed. The RE conference has significantly contributed to the progress of the area, and challenges acknowledged in the literature and based on insights into practice have been regularly tackled.

With regard to the areas for further research, we consider that traceability visualization, impact analysis, and tool qualification must be studied in more depth. We also think that it is necessary to focus on the opinion and experiences of practitioners different to the researchers, conduct dynamic validation, replicate studies, and compare approaches.

These areas represent topics on which we plan to research in the future, especially in the scope of critical systems development. Given the recent advances and its importance for adoption in industry, it might also be relevant to conduct a SLR on automated traceability.

ACKNOWLEDGMENT

The research leading to this paper has received funding from the FP7 programme under the grant agreement n° 289011 (OPENCOSS) and from the Research Council of Norway under the project Certus SFI.

REFERENCES

[1] N. Aizenbud-Reshef, et al., “Model traceability”, IBM Sys. J., vol. 45, no. 3, pp. 515-526, 2006.

[2] J.E. Archer, “Requirement Tracking: A Streamlined Approach”, RE 2003.

[3] P. Arkley and S. Riddle, “Overcoming the Traceability Benefit Problem”, RE 2005.

[4] P. Arkley, S. Riddle, and T. Brookes, “Tailoring Traceability Information to Business Needs”, RE 2006.

[5] D. Becker, “Measuring Requirements Traceability from Multiple Angels at Multiple Lifecycle Entry Points”, RE 2003.

[6] B.H.C. Cheng and J.M. Atlee, “Research Directions in Requirements Engineering”, FOSE 2007.

[7] J. Chisan and D. Damian, “Exploring the role of requirements engineering in improving risk management”, RE 2005.

[8] J. Cleland-Huang, et al., “Automating Speculative Queries through Event-Based Requirements Traceability”, RE 2002.

[9] J. Cleland-Huang, G. Zemont, and W. Lukasik, “A Heterogeneous Solution for Improving the Return on Investment of Requirements Traceability”, RE 2004.

[10] J. Cleland-Huang, et al., “Breaking the big-bang practice of traceability: Pushing timely trace recommendations to project stakeholders”, RE 2012.

[11] J. Cleland-Huang, O. Gotel, and A. Zisman (eds.), Software and Systems Traceability. Springer, 2012.

[12] N. Condori-Fernández, et al. “A Systematic Mapping Study on Empirical Evaluation of Software Requirements Specifications Techniques”, ESEM 2009.

[13] D. Cuddeback, A. Dekhtyar, and J.H. Hayes, “Automated Requirements Traceability: The Study of Human Analysts”, RE 2010.

[14] Å.G. Dahlstedt and A. Persson, “Requirements Interdependencies – Moulding the State of Research into a Research Agenda”, REFSQ 2003.

[15] A. Davis and A. Hickney, “A Quantitative Assessment of Requirements Engineering Publications – 1963-2008”, REFSQ 2009.

[16] A. Dekhtyar, et al., “Technique Integration for Requirements Assessment”, RE 2007.

[17] A. Espinoza, P.P. Alarcón, and J. Garbajosa, “Analyzing and Systematizing Current Traceability Schemas”, SEW 2006.

[18] I. Galvão and A. Goknil, “Survey of Traceability Approaches in Model-Driven Engineering”, EDOC 2007.

[19] T. Gorschek, et al., “A Model for Technology Transfer in Practice”, IEEE Software, vol. 23, no. 6, pp. 88-95, 2006.

[20] O. Gotel and A. Finkelstein, “An analysis of the requirements traceability problem”, RE 1994.

[21] O. Gotel and A. Finkelstein, “Contribution Structures”, RE 1995.

[22] O. Gotel and A. Finkelstein, “Extended Requirements Traceability: Results of an Industrial Case Study”, RE 1997.

[23] O. Gotel, et al., “The Quest for Ubiquity: A Roadmap for Software and Systems Traceability Research”, RE 2012.

[24] P. Haumer, et al. , “Bridging the gap between past and future in RE: a scenario-based approach”, RE 1999.

[25] J.H. Hayes, A. Dekhtyar, J. Osborne, “Improving Requirements Tracing via Information Retrieval”, RE 2003.

[26] J.H. Hayes, et al., “Helping Analysts Trace Requirements: An Objective Look”, RE 2005.

[27] J. Hill and S. Tilley, “Creating Safety Requirements Traceability for Assuring and Recertifying Legacy Safety-Critical Systems”, RE 2010.

[28] E.A. Holbrook, J.H. Hayes, and A. Dekhtyar, “Toward Automating Requirements Satisfaction Assessment”, RE 2009.

[29] IEEE, IEEE Standard Glossary of Software Engineering Terminology, Std. 610.12-1990.

[30] M. Ivarsson and T. Gorschek, “A method for evaluating rigor and industrial relevance of technology evaluations”, Empirical Software Engineering, vol. 16, no. 3, pp. 365-395, 2011.

[31] M. Jarke, et al., “Media-Assisted Product and Process Requirements Traceability in Supply Chains”, RE 2003.

[32] A. Kannenberg and H. Saiedian, “Why Software Requirements Traceability Remains a Challenge”, Cross Talk, vol. 22, no. 5, pp.14-17, 2009.

[33] S. Katz and A. Rashid, “From aspectual requirements to proof obligations for aspect-oriented systems”, RE 2004.

[34] B.A. Kitchenham and S. Charters, “Guidelines for performing systematic literature reviews in software engineering”, Version 2.3, EBSE Technical Report, 2007.

[35] S.F. Königs, et al., “Traceability in Systems Engineering – Review of industrial practices, state-of-the-art technologies and new research solutions”, Adv. Eng. Informatiics, vol. 26, no. 4, pp. 924-940, 2012.

[36] L. Lavazza and G. Valetto, “Enhancing requirements and change management through process modelling and measurement”, RE 2000.

[37] J.C.S.P. Leite and A.P.A. Oliveira, “A client oriented requirements baseline”, RE 1995.

[38] Y. Li and W. Maalej, “Which Traceability Visualization Is Suitable in This Context? A Comparative Study”, REFSQ 2012.

[39] J. Lin, et al. “Poirot: A Distributed Tool Supporting Enterprise-Wide Automated Traceability”, RE 2006.

[40] I.A. Macfarlane and I. Reilly, “Requirements traceability in an integrated development environment”, RE 1995.

[41] F. McCaffery, et al., “Medical Devide Software Traceability”, in [11], pp. 321-339, 2012.

[42] P. Mäder, et al. “traceMaintainer - Automated Traceability Maintenance”, RE 2008.

[43] P. Mäder, O. Gotel, and I. Philippow, “Motivation Matters in the Traceability Trenches”, RE 2008.

[44] S. Nair, J.L de la Vara, and S. Sen, “Traceability Research at the Requirements Engineering Conference: Results and Data Extracted”, Simula Res. Lab., Technical Report, 2013.

[45] S. Nair, et al., “Classification, Structuring, and Assessment of Evidence For Safety: A Systemtic Literature Review”, ICST 2013.

[46] B. Nuseibeh and S. Easterbrook, “Requirements engineering: a roadmap”, FOSE 2000.

[47] R. Oliveto, et al., “Software Artefact Traceability: the Never-Ending Challenge”, ICSM 2007.

[48] M.C. Panis, “Successful Deployment of Requirements Traceability in a Commercial Engineering Organization...Really”, RE 2010.

[49] F.A.C. Pinheiro and J.A.Gogue, “An Object-Oriented Tool for Tracing Requirements”, RE 1996.

[50] K. Pohl, Requirements Engineering. Springer, 2010.

[51] B. Ramesh and M. Edwards, “Issues in the development of a requirements traceability model”, RE 1993.

[52] B. Ramesh, et al., “Implementing requirements traceability: a case study”, RE 1995.

[53] G. Regan, et al., “The Barriers to Traceability and their Potential Solutions: Towards a Reference Framework”, SEAA 2012.

[54] G. Regan, et al., “Traceability – Why Do It?”, SPICE 2012.

[55] G. Romanski, “Requirements, Configuration Management and Traceability for Safety Critical Software”, RE 2003.

[56] M. Sabetzadeh and S. Easterbrook, “An algebraic framework for merging incomplete and inconsistent views”, RE 2005.

[57] M. Sabetzadeh, et al., “Consistency Checking of Conceptual Models via Model Merging”, RE 2007.

[58] I. Santiago, et al., “Model-Driven Engineeirng as a new lanscape for traceability management: A systematic literature review”, Info. And Softw. Tech., vol. 54, no. 12, pp. 1340-1356, 2012.

[59] S. Siegl, K.S. Hielscher, and R. German, “Model Based Requirements Analysis and Testing of Automotive Systems with Timed Usage Models”, RE 2010.

[60] T.J. Simth, “READS: a requirements engineering tool”, RE 1993.

[61] G. Spanoudakis and A. Zisman, “Software traceability: a roadmap”, in Handbook of Software Eng. and Knowledge Enginnering, 2005.

[62] K. Takahashi, et al., “Hypermedia support for collaboration in requirements analysis”, RE 1996.

[63] R. Torkar, et al., “Requirements traceability: a systematic literature review and industry case study”, Int. J. of Softw. Eng. and Knowl. Eng. , vol. 22, no. 3, pp. 1-49, 2012.

[64] E.J. Uusitalo, et al., “Linking Requirements and Testing in Practice”, RE 2008.

[65] A. von Knethen and B. Paech, “A Survey on Tracing Approaches in Practice and Research”, IESE-Report No. 095.01/E, 2002.

[66] A. von Knethen, et al., “Systematic Requirements Recycling through Abstraction and Traceability”, RE 2002.

[67] S. Winkler and J. von Pilgrim, “A survey of traceability in requiremerns engineering and model-driven development”, SoSym, vol. 9, no. 4, pp. 529-565, 2010.

[68] D.P. Wood, M.G. Christel, and S.M. Stevens, “A multimedia approach to requirements capture and modeling”, RE 1994.

[69] Y. Yu, et al., “Reverse engineering goal models from legacy code”, RE 2005.

Extracted Data:

Ref	Institutions (RQ8)	Countries (RQ8)	Area (RQ1)	Challenges (RQ2)	Contributions (RQ3)	Tool Features (RQ4)	Types of system (RQ5)	Artifact Traced (RQ6)	Empirical Evaluation (RQ7)
[49]	Naval Postgraduate School and The United States Naval Surface Warfare Center, Dahlgren Division	US	**(Post requirements traceability) -** maintain traceability of requirements to various outputs or artifacts produced during the systems design process	**(Guaranteeing satisfaction of requirements)** that the design meets the requirement	**(Insights and Experience reports)** preliminary study to explore the traceability needs of various stakeholders	None	Real time embedded systems	Design component & design component, requirement & design, design & responsible, requirement & responsible	Experiment
[55]	Paramax Systems Corp.	US	**(Change management; tractability automation)** Requirements change management, traceability information management and creation	None	**(Technical)** hypertext, database-based tool for requirements management	**Maintaining traceability between artifacts specific to requirements specification, Change Management, support for Regulatory Compliance**	Safety Critical systems (DOD-STD-2176A)	High-level requirement & low-level requirement, requirement & rationale, requirement & testing; requirement version & requirements version; requirement & design	Action research
[63]	Carnegie Mellon Univ. (SEI)	US	**Pre-requirements traceability, traceability automation**	None	**(Technical)** Tool to hierarchize requirements and have access to their source	None	Safety Critical system	Requirement & requirement source	Field study
[19]	Imperial College of Science, Technology & Medicine	UK	Maintaining traceability of requirements **(pre-requirements traceability)**	**(Lack of knowledge and understanding about traceability)** understanding RT, problems attributed to poor requirements traceability due to inadequate pre-RS traceability	**(Insights and Experience reports)** on main barriers in practice, identify relevant areas in which advances have been (or can be) made, and make recommendations for research.	None	Unspecified	Requirements & design; requirements & code	Survey, interview, Literature review
[34]	PUC-Rio, Petrobras	Brazil	**Pre-requirements traceability**	Need for an explicit traceability information model **(Effective representation of traceability information)**	**(Methodological)** Requirements information model, conceptual model, traceability information model.	None	Information system	Requirement & requirement source	None
[66]	Universite Catholique de Louvain	Belgium	**(Model traceability)** Traceability in GORE	None	**(Experience report)** Report on KAOS traceability issues	None	Information system	Requirement & rationale	None
[50]	Naval Postgraduate School, NSWCDD	USA	**(Post-requirements traceability)** Traceability of requirements in weapons systems software	None	**(Insights and Experience reports)** Definition of traceability, traceability of hardware upgrades, Observed high costs of traceability, Steep learning curve for teaching CASE tool, Team responsibility due to audits, Less loss in man-months due to traceability, Improved hardware	**Maintaining traceability between artifacts specific to requirements specification, Project Management**	Safety Critical systems	Requirements & code	Survey

			and Configuration Management IDE			**specification**			
[59]	NTT Software Laboratories, Nippon Telegraph and Telephone Corporation, College of Computing, GATECH, M/Cast Communications, Inc.	Japan, USA	**(New approaches for recording traceability)** Traceability using hypermedia such as audio/video recordings	None	**(Technical)** Ecolabor is a tool for using hypermedia (audio/video) to maintain traceability in different artifacts.	**Traces lifecycle**	Generic	Requirements & source	Action Research
[46]	RWTH Aachen, Informatik V	Germany	**(Pre-requirements traceability)** maintaining traceability of requirements	None	**(Technical)** An environment called PRO-ART which enables requirements pre-traceability.	**Automated traceability**	Unspecified	Requirement & source	None
[45]	Oxford University	UK	Maintaining traceability of requirements **(change management)**	**(Maintaining traceability when requirements evolve)** traceability of requirements and evolving requirements	**(Technical)** A tracing tool that supports requirements evolution and treats requirements and relations among them as objects	TOOR **Maintaining traceability between artifacts specific to requirements specification**	Unspecified	Requirement & creator; design & creator	None
[21]	City University, London	UK	**(Pre-requirements traceability)** Traceability between human sources and artifacts	None	**(Insights & experience reports)** Contribution structures between informal documents	None	Information system	Requirement & source; requirement & responsible	Case study
[47]	Siemens Corporate Research, Inc.	USA	**(New approaches for maintaining traceability)** Web-based traceability tool	None	**(Technical)** Tool to track requirements on the web	**(Maintaining traceability between artifacts specific to requirements specification)** STAR-Track system	Information system	Requirement & source	None
[24]	RWTH Aachen, University of Namur	Germany, Belgium	**(Post-requirements traceability)** Traceability of scenes in engineering	None	**(Technical)** an Integrated tool for traceability and envisionment	**(Traces lifecycle, Traceability visualization)** CREWS-EVE	Generic	High-level requirements & low-level requirements; requirement & source; requirement & testing	Action Research
[33]	CEFRIEL and Politecnico di Milano	Italy	**(Post-requirements traceability; change management)** Improving the effectiveness of requirements management in software development and maintenance	**(Maintaining traceability when requirements evolve)** quantitative assessment of the impact of requirements changes	**(Technical)** A methodological approach to quantitative requirements and change management and tool support for the methodology	**(Requirements Validation with traceability support, Change management)** An extension of DOORS in the SACHER environment	Generic	Requirement & design; requirement & code	Action Research
[61]	Fraunhofer Institute for Experimental Software Engineering, University of Ulm, DaimlerChrysler AG Research and Technology	Germany	**(Change management; requirements versioning)** Application of traceability to requirements recycling	**(Effective representation of traceability information)** how to document relationships of recycling candidates explicitly and on how to use relationships to copy recycling candidates correctly.	**(Methodological & Insights and Experience reports)** Use of conceptual models to determine relationships necessary for correct recycling and the focus on minimizing link setting in traceability & Conceptual models can used to build abstraction and traceability guidelines	None	Safety Critical system	Requirement version & requirement version	Survey, Case study
[7]	University of Illinois at Chicago	US	**(Change management)** impact analysis on software components	**(Maintaining traceability when requirements evolve)** analyze requirements change effect on functional performance	**(Methodological)** a technique for supporting performance related impact analysis amongst heterogeneous software engineering components	None	Unspecified	Non-functional requirements & functional requirements	Experiments
[25]	University of Kentucky	US	**(Traceability automation)** Requirements tracing as an Information retrieval problem	**(Impact of Impact of human factors and judgment)** reduce the number of irrelevant	**(Technical)** improve the state of the art of after the fact requirements tracing	None	NASA Moderate Resolution Imaging Spectroradiometer	High-level requirements & low-level requirements;	Field Study

				potential links that an analyst has to examine when performing requirements tracing				requirement & design; requirement & testing; design and code	
[4]	EDS	US	**Change management, post-requirements traceability**	Traceability measurement in an already running project **(Traceability in practice)**	**(Insights and experience reports)** Insights into traceability solutions in practice	None	Non software Systems	Design & code	Action research
[5]	Guidant Corporation	US	**Post-requirements traceability**	When to stop testing with the claim that the quality required by the software has been demonstrated **(Lack of knowledge and understanding about traceability)**	**(Methodological)** approach called testing with partial traced requirements	None	Unspecified	Requirements & testing	Action research
[51]	Verocel Inc	US	**(Post-requirements traceability; regulatory compliance; traceability in practice)** maintaining traceability of requirements	**(Effective representation of traceability information)** maintaining and presenting requirements traceability for certification to show compliance with DO178B	**(Technical)** Example of how to deal with large amounts of traceability information, and tool support for it	None	Safety Critical Systems	Requirements & code; requirements & design; requirements and testing	None
[1]	Titan Systems Corporation	US	**Post-requirements traceability**	Avoid high degree of formality and informal techniques **(Effective representation of traceability information)**	**(Methodological)** use of requirement statement as requirements unit & requirements hierarchy	None	Unspecified	High-level requirements & low-level requirements	None
[29]	Informatik V, RWTH Aachen	Germany	**(Post-requirements traceability; design trade-off analysis, Traceability in new development context)** Requirements traceability in technical systems in chemical engineering and plastic engineering in automotive industry	Traceability of interdisciplinary design decisions, and corresponding reuse of experiences both from the product and the process perspective **(Traceability in practice)**	**(Technical)** An environment that builds on a decision-oriented and situation-based process meta model	**(Traces lifecycle)** PRIME Process-Integrated Modeling Environment	Non Software Systems Plastic engineering systems	Design decision & design decision	None
[8]	DePaul University	USA	**(Trade-off analysis)** Integration of heterogeneous strategies of traceability for complex systems	Traceability of non-functional requirements with function **(Traceability satisfaction)**	**(Technical)** Integration of a diverse set of trace strategies in one tool. Demonstrated increase in ROI	**(Automated traceability, Change management)** TraCS	Unspecified	Functional requirement & non-functional requirement	Field Study
[30]	Lancaster Univ.	UK	Requirements verification and validation **(post requirements traceability)**	No systematic means to trace the refinement of aspectual requirements through to an aspect-oriented design and implementation. **(Guaranteeing satisfaction of requirements)**	**(Technical)** Framework for generation of proof obligations in standard linear temporal logic.	**(Maintaining traceability between artifacts specific to requirements specification)** extended Ontology with parametric temporal formulas and functions, and extensive treatment of conflicts among requirements	Information system	Requirements & formal verification; requirements & design; design & testing; requirements & conflict	None
[26]	University of Kentucky	USA	**(Post requirements traceability)** Tracing Requirements for Verification and Validation Analysis	**Guaranteeing satisfaction of requirements, Human factors and judgment**	**(Technical)** tool based on analyst responsibilities in the tracing process, new measures for validating requirements,	**(Traces lifecycle)** RETRO candidate link lists	Information system	High-level requirements & low-level requirements	Field Study
[9]	DePaul University	US	**(Post-requirements traceability; change management)** Traceability of requirements for requirements validation and	**(Reducing the cost related to requirements traceability)** Cost and effort of manually constructing and maintaining	**(Methodological)** three strategies for incorporating supporting information into a probabilistic retrieval algorithm in order to improve the performance of	None	Unspecified	Design and code	None

			change management	trace matrices	dynamic requirements traceability				
[52]	Univ. of Toronto	Canada	**(Model traceability)** View merging, model management	Merging Incomplete and Inconsistent Views **(Requirements satisfaction)**	**(Technical)** annotated graphs for merging views, and provide a general algorithm for merging views with arbitrary interconnections.	**Traces lifecycle**	Information system	Requirement & contributor; requirement version & requirement version	None
[6]	Univ. of Victoria	Canada	Benefits of traceability in risk management **(Traceability in practice)**	**(Lack of knowledge and understanding about traceability)**	**(Insights and Experience reports)** Study showing traceability perceived benefits and adoption of traceability practices in planning and SQA	None	Information system	Requirements & rationale; requirements & testing	Case study
[15]	Univ. of Toronto	Canada	**(Pre-requirements traceability)** Traceability to stakeholders	**(Lack of knowledge and understanding about traceability)** No evidence/empirical studies of benefits of viewpoints	**(Insights and experience report)** evidence of "Viewpoints modeling improves traceability to individual stakeholders"	None	Information system	Requirement & contributing stakeholder	Case study
[64]	Univ. of Toronto, PUC-Rio	Canada, Brazil	Traceability between requirements and code in legacy system **(post-requirements traceability)**	Reverse engineering goal models from code **(Guaranteeing satisfaction of requirements)**	**(Methodological)** Method to refactor source code based on comments, Extracting a goal model from the abstract syntax tree, Identifying nonfunctional requirements and derive softgoals based on the traceability between the code and the goal model	None	Information system	Requirement & code	Field study
[2]	University of Newcastle	UK	**(New approaches for recording traceability)** Traceability Benefits Problem	**(Lack of knowledge and understanding about traceability)** Lack of understanding and perceived bureaucracy	**(Methodological)** Method of recording traceability information	None	Generic	Requirement & development standard; design and development standard; code & development standard; testing & development standard	None
[3]	University of Newcaste, BAE Systems	UK	**(Traceability in practice)** Traceability Benefits Problem	None	**(Methodological)** visualization of requirements maturity index, justification of costs , demonstration of how requirements led to tests to customer, link precision and recall	None	Safety Critical systems	Requirements & testing	Field Study
[10]	DePaul University	US	**Traceability in practice**	**(Challenges in practice)** Finding the right traceability process that delivers effective and efficient traceability.	**(Insights and experience report)** explores traceability challenges and solutions for finding the right techniques and process to deliver cost effective traceability within an organization	None	Unspecified	Requirements & responsible	None
[38]	Technical Univ. of Valencia	Spain	**(Post requirements traceability)** Requirements to code traceability	None	**(Methodological)** well defined transformations providing traceability from requirements to implementation and vice-versa	None	Generic	Requirements & code	None
[57]	Lancaster Univ.	UK	**(Pre-requirements traceability)** identification and maintenance of relationships between requirements and the knowledge and information used by analysts to inform the requirements' formulation	**(Lack of knowledge and understanding about traceability)** identification of tacit knowledge, determine requirements that are not firmly derived from source material	**(Technical)** tool for retrospectively identifying pre-requirements traces from requirements to their respective source material.	**(Maintaining traceability between artifacts specific to requirements specification)** establish backwards traces from requirements into extant textual source material	Safety Critical Systems & Information system	Requirements & source	Field study
[35]	DePaul University & Siemens Corporate Research	US	**Traceability automation**	**(Reducing the cost related to requirements traceability)** Cost and effort of manually	**(Technical)** an industry-ready prototype model implementing a probabilistic approach to dynamically generate	Poirot **(Automated traceability)**	Unspecified	Requirements & design; requirements & code	Experiments

				constructing trace matrices	traceability links.				
[32]	DePaul University	US	**(Pre-requirements traceability)** Choosing the right requirements for the system	**(Reducing the cost related to requirements traceability)** Budgetary restrictions and time to market constraints dictate stakeholders to select a subset of requirements for development.	**(Methodological)** propose a semi-automated technique for generating a list of prioritized requirements from a large set of incoming stakeholders' requests, and shows how this prioritized list feeds into the triage process	None	Non Software Systems	High-level & low-level requirements	Experiments
[53]	Univ. of Toronto	Canada	**(Model traceability)** Traceability from source to target models	Global Consistency Checking of several Conceptual Models **(Requirement evolution)**	**(Technical)** implementation of an approach within a logic-based constraint specification framework and automatic generation of traceability information	**(Model Management with traceability support)** integrated environment for model construction, mapping, and merging	Information system	Requirement version & requirement version; requirement & formal verification	None
[13]	Univ. of Kentucky	US	**(Post-requirements traceability; regulatory compliance; traceability in practice; traceability automation)** the requirements traceability matrix (RTM) delivered by the developer must be assessed for accuracy for certification	**(Assessing the traceability maintained)** The current state of the practice is to perform this work manually, Such work is error-prone and person-power intensive.	**(Methodological)** application of Information Retrieval (IR) methods for candidate link generation to the problem of RTM accuracy and completeness assessment.	None	Safety-critical system	Requirements & design; design & code	Experiment
[65]	Open Univ, Univ. Toronto, Univ. Valladolid, Univ. Lille, PUC-Rio	UK, Canada, Spain, France, Brazil	**(Post-requirement traceability)** Code validation against requirements	To validate the modularized code aspects against their very purposes of existence **(Guaranteeing satisfaction of requirements)**	**(Methodological)** framework to trace aspects identified during goal-oriented requirements analysis to code and testing	None	Information system	Requirements & code; requirements & testing	None
[67]	Teknowledge Corp., Johannes Kepler Univ., PSE Siemens Austria & Vienna Univ. of Techn.	US & Austria	**(Post requirement traceability)** maintaining traceability of requirements	**(Reducing the cost related to requirements traceability)** Considering if the traceability is going to be used long-term hence to reduce cost and effort	**(Methodological)** Proposes a value-based approach to software traceability	None	Information systems	Requirements & design; requirements & code	Action research
[41]	Ilmenau Technical University & Pace University	Germany & US	**(Post requirement traceability)** maintaining traceability of different development artifacts	**(Maintaining traceability when requirements evolve)** maintaining a set of traceability relations in the face of evolutionary change	**(Technical)** proposes an approach for the automated update of existing traceability relations after changes have been made to UML analysis and design models	**(Automated Traceability, Change Management)** a prototype tool implemented in Visual Studio .Net and uses the Microsoft XML Parser	Unspecified	Requirements & design	Experiments
[60]	Helsinki University of Technology, The University of Colarado	Finland, USA	**(Post requirement traceability)** Linking requirements and test survey	**(Lack of knowledge and understanding about traceability)** Lack of information and knowledge in testing teams about requirements	**(Insights and Experience reports)** reports on how to improve testers knowledge	None	Generic	Requirements & testing; requirements & tester	Survey
[17]	Univ of Toranto, Open University & Iowa State Univ	Canada, UK and US	**(Model traceability, post-requirement traceability)** The need for support for requirements evolution throughout the software lifecycle, that is, post-	Need for requirements model management when requirements evolve **(Maintaining traceability when requirements evolve)**	**(Technical)** Proposes a framework and tool support for requirement model management.	OpenOME **(Maintaining traceability between artifacts specific to requirements specification, Model Management)**	Generic	Requirements & design; requirements & code	None

			implementation						
[18]	Univ. of Alicante	Spain	**(Traceability automation, model traceability; post requirement traceability)** traceability between requirements and the necessary multidimensional elements	None	**(Technical)** framework establishes a set of formal transformations between a requirement model and a conceptual multidimensional model via the QVT	**Model Management, Maintaining traceability between artifacts specific to requirements specification**	Information system	High-level & low-level requirements	None
[39]	Ilmenau Technical University & Pace University	Germany & US	**(Traceability automation; post requirement traceability)** maintaining traceability of software artifacts	None	**(Technical)** an approach to automatically maintain relations between requirements and sub-sequent artifacts	traceMaintainer **(automated traceability)**	Unspecified	High-level requirements & low-level requirements; requirements & design	Field Study
[62]	Univ of Victoria	Canada	**(Pre requirement traceability)** Maintain requirements denoted in natural language for complex large scale software systems	**(Impact of human factors and judgment)** existing approaches to analyze NL requirements rely on a manual linguistic transformation	**(Technical)** an approach to analyzing requirements by using semantic annotations placed directly into the original NL documents	CREE-tool **(Maintaining traceability between artifacts specific to requirements specification)**	Generic	Requirements & source	Field Study
[40]	Ilmenau Technical University & Pace University	Germany & US	**(Traceability in practice)** traceability practices and problems in industry	**(Lack of knowledge and understanding about traceability)** Previous studies are outdated and date back to over 10 years.	**(Insights and Experience reports)** an exploratory study of the traceability practice and problems within ten companies based predominantly in Germany	None	Safety Critical System Automotive & avionics systems	Requirements & creator; requirements & contributor; requirements & testing; requirements & design; design to testing; requirements & development standards; requirements & code; high-level & low-level requirements	Survey
[28]	Lexmark, University of Kentucky & California Polytechnic State University	US	**(Post requirement traceability)** mapping of natural language textual requirements to natural language design elements	**(Requirement satisfaction)** assessing whether requirements have been satisfied by lower level artifacts such as design	**(Technical)** proposes a 3 step approach for assessing requirement satisfaction and tool support	RESAT - **(Requirements validation with traceability support)**	Unspecified	Requirements & design	Action Research & Experiment
[48]	University of Karlsruhe, Robert Bosch GmBH	Germany	**(Post requirement traceability)** Test and scenario traceability w.r.t requirements	None	**(Technical)** Formalizing Requirements, Checking requirements using CMBC model checker, linking formal analysis and component requirements	Checking requirements **(Requirements validation with traceability support)**	Real time embedded systems	Requirements & testing; high-level requirements & low-level requirements	Case study
[68]	Norwegian University of Science and Technology & University of Strathclyde	Norway & Scotland	**(Post requirement traceability)** identification and addition of traceability information between use cases and source code	Identifying trace links that relate code artifacts and developers to use cases and the relative importance of such links **(Guaranteeing satisfaction of requirements)**	**(Methodological)** presents a relevance indexing approach that enables trace between requirements expressed as uses cases and code artifacts or developer	None	Information system	Requirements & responsible; requirements & code	None
[58]	University of Kentucky	US	**(Post requirement traceability)** Understanding requirements of a system, improving the quality of the requirements	None	**(Methodological)** proposes a swarm technique and simplified ant colony algorithm for tracing textual pairs of requirements artifacts	None	Information system & Safety Critical	High-level requirements & low-level requirements	Action Research
[16]	Johannes Kepler University	Austria	**(Post requirement traceability; traceability automation)** tracing requirements to code	**(Reducing the cost related to requirements traceability)** cost- effectiveness of traceability between requirements and code	**(Technical, insights and experience reports)** Presents two exploratory experiments conducted to trace links for two open source software systems in a controlled environment. The results can be used as bench-mark.	TraceCapture tool **(Traces lifecycle)**	Information system & Safety Critical System	Requirements & code	Experiments

[12]	California Polytechnic State University & University of Kentucky	US	**(Traceability automation)** Automated requirements traceability	**(Human factor and Judgment)** human performance in choosing the right traces from automated requirements traceability matrices	**(Technical)** presents a framework for the study of analyst interaction with artifacts generated automatically during the tracing process	RETRO - **(Traces lifecycle)**	Unspecified	Requirements & testing	Experiments
[54]	University of Erlangen-Nuremberg	Germany	**(Model traceability; post requirement traceability)** Requirements modeling and traceability with test cases	Modeling language for requirements and traceability **(Effective representation of traceability information)**	(**Technical**) Usage models in automotive domain for requirements description, analysis and validation of requirements, linking of test cases to requirements	EXAM approach: **(Maintaining traceability between artifacts specific to requirements specification)**	Real Time Embedded systems	Requirements & testing	Case study
[27]	NASA & Florida Institute of Technology	US	**(Post requirement traceability; regulatory compliance)** safety requirement traceability for recertification of legacy critical systems	None	**(Technical)** describes the steps to perform the software Safety Risk Evaluation (SSRE), and proposes a Legacy Systems Risk Database (LSRD) that maintains requirements traceability	LSRD **(Traces lifecycle)**	Safety-critical (legacy) systems	Requirements & development standard; high-level & low-level requirements; requirement & rationale; requirement & source	Action Research
[56]	CSIRO	Australia	**Post-requirements traceability, change management**	**(Maintaining traceability when requirements evolve)** risk of regression in the system when introducing changes	**(Methodological)** a technique where the coverage of tests can be measured against a suite of system tests, and subsequently these system tests can be traced back to reveal requirement shortfalls.	None	Information system	Requirements & code; requirements & testing	Action research
[44]	Teradyne, Inc.	US	(**Traceability in practice)** maintaining traceability of requirements	**(Lack of knowledge and understanding about traceability)** effective traceability practices	**(Insights and Experience reports)** presents a case study of traceability practice in the authors company	None	Non Software System	High-level requirements & low-level requirements; requirements & testing	Case study
[14]	Cal Poly & University of Kentucky	US	**(Traceability automation)** Automated requirements traceability	**(Impact of human factors and judgment)** which factors have the largest effect on the quality of the final trace when it comes to human analysts performing assisted tracing process	**(Technical)** presents 11 independent variables which may account for the change in final TM accuracy	RETRO (**Traces lifecycle)**	Unspecified	Requirements and code	Experiment
[22]	City University London	UK and US	**Traceability in practice, pre-requirements traceability, post-requirements traceability**	Unique characteristics of requirements traceability in software engineering **(Lack of knowledge and understanding about traceability)**	**(Methodological)** examines tracing and its underlying concepts across a number of disciplines to highlight the specific challenges associated with tracing requirements	None	Unspecified	High-level requirement & lo-level requirements	None
[43]	Fujitsu Laboratories Ltd.	-	**(Model traceability; post requirement traceability)** maintaining traceability of requirements for business application, impact analysis and verification	Achieving quality improvement of the business application's design through establishing traceability links between requirements definition artifacts and design phase artifacts during the design phase **(Guaranteeing satisfaction of requirements)**	**(Technical)** proposes a model-based means to define adequate granularity of artifact particles and to retrieve complete sets of candidate traceability links between the particles	Unnamed tool - **(Traces lifecycle)**	Non Software System	Requirements & design	Field Study
[37]	Mississippi State University	US	(**Traceability automation)** Automated traceability and IR o recover traceability links between software artifacts	**(Effective representation of traceability information)** Front end presentation of the retrieved result	**(Technical)** proposes a candidate traceability clustering method & tool with novel search user interfaces.	TraCter (**Automated traceability)**	Unspecified	Requirements version & requirements version	Experiments

[70]	Univ. of Toronto	Canada	**(Model traceability)** Traceability between models (requirements models with uncertainty, partial models)	Reasoning with traceability relations between models containing uncertainty **(Traceability evolution)**	**(Methodological)** "lift" an existing traceability relation to a partial traceability relation. Each trace link and each of its endpoints can be annotated with MAVO annotations	None	Information system	High-level requirements & low-level requirements	None
[31]	University of Kentucky & California Polytechnic State University	US	(**Traceability automation)** improving traceability practices, in particular automated traceability and TM	**(Human factors and judgment)** Human analyst impact on TM and to develop procedures and software that facilitate accurate assisted tracing	**(Technical)** presents a set of measures that focus on the quality of the analyst working to produce final TMs, visualizing and analyzing analyst trace logs to detect trends	SmartTracer **(Traces lifecycle)**	Information system & Safety Critical System	High-level requirements & low-level requirements	Experiments
[23]	De Paul University, University of Kentucky, City University London, ohannes Kepler University Linz, cole Polytechnique de Montréal	US, UK, Austria & Canada	(**Traceability in practice)** State of the art & practice in traceability and the grand challenge	(**Reducing the cost related to requirements traceability)** traceability that is valued in the near-term, purposed, portable, cost-effective and scalable	**Insights and Experience reports)** a road map of traceability for practice & future research	None	Unspecified	Requirements & design; requirements & rationale; requirements & testing	None
[42]	Mississippi State University	US	(**Traceability automation)** Automated requirements traceability	**(Lack of knowledge and understanding about traceability)** practitioners often fail to implement consistent and effective traceability processes if the traces are maintained manually	**(Methodological)** propose an approach to improving the quality of candidate link generation for the requirements tracing process	None	Information system	Requirements & code; requirements & testing; requirements & design	Experiments
[69]	University of Zurich	Switzerland	**(Post requirement traceability)** maintaining updated requirements	**(Maintaining traceability when requirements evolve)** in practice engineers usually apply changes to the implementation directly and leave requirements unchanged	**(Technical)** proposes an approach for automatically detecting outdated requirements based on changes in the code	RETRO **(Traces lifecycle, Maintaining traceability between artifacts specific to requirements specification, automated traceability)**	Information system	Requirements & code	Field Study or Experiments
[11]	DePaul University & Johannes Kepler University	US & Austria	**(Regulatory compliance; post requirement traceability; pre requirements traceability)** maintaining traceability assures the system to be safe	in practice, traceability links are often created towards the end of the project specifically for approval or certification purposes. This can result in inaccurate and incomplete traces (**traceability in practice)**	**(Methodological)** proposes an approach for generating and pushing timely trace recommendations to developers in order to construct traceability links earlier in the project	None	Safety Critical System	Requirements & design; requirements & code; requirements & rationale; high-level & low level requirements; requirements & testing	None

References

[1] J.E. Archer, “Requirement Tracking: A Streamlined Approach”, RE 2003.
[2] P. Arkley and S. Riddle, “Overcoming the Traceability Benefit Problem”, RE 2005.
[3] P. Arkley, S. Riddle, and T. Brookes, “Tailoring Traceability Information to Business Needs”, RE 2006.
[4] D. Becker, “Measuring Requirements Traceability from Multiple Angels at Multiple Lifecycle Entry Points”, RE 2003.
[5] S. Catrava, “Testing with Partial Traced Requirements: A Necessary Step Towards Higher Quality System Level Verification”, RE 2003.
[6] J. Chisan and D. Damian, “Exploring the role of requirements engineering in improving risk management”, RE 2005.
[7] J. Cleland-Huang, et al., “Automating Speculative Queries through Event-Based Requirements Traceability”, RE 2002.
[8] J. Cleland-Huang, G. Zemont, and W. Lukasik, “A Heterogeneous Solution for Improving the Return on Investment of Requirements Traceability”, RE 2004.
[9] J. Cleland-Huang, et al., “Utilizing Supporting Evidence to Improve Dynamic Requirements Traceability”, RE 2005.
[10] J. Cleland-Huang, “Requirements Traceability - When and How does it Deliver more than it Costs?”, RE 2006.
[11] J. Cleland-Huang, et al., “Breaking the big-bang practice of traceability: Pushing timely trace recommendations to project stakeholders”, RE 2012.
[12] D. Cuddeback, A. Dekhtyar, and J.H. Hayes, “Automated Requirements Traceability: The Study of Human Analysts”, RE 2010.
[13] A. Dekhtyar, et al., “Technique Integration for Requirements Assessment”, RE 2007.
[14] A. Dekhtya, et al. “On human analyst performance in assisted requirements tracing: Statistical analysis”, RE 2011.
[15] S. Easterbrook, et al., “Do viewpoints lead to better conceptual models? An exploratory case study”, RE 2005.
[16] A. Egye, F. Graf, and P. Grünbacher, “Effort and Quality of Recovering Requirements-to-Code Traces: Two Exploratory Experiments”, RE 2010.
[17] N.A. Ernst, et al., “Supporting Requirements Model Evolution throughout the System Life-Cycle”, RE 2008.
[18] O. Glorio, et al., “DaWaRA: An Eclipse Plugin for Using i* on Data Warehouse Requirement Analysis”, RE 2008.
[19] O. Gotel and A. Finkelstein, “An analysis of the requirements traceability problem”, RE 1994.
[20] O. Gotel and A. Finkelstein, “Contribution Structures”, RE 1995.
[21] O. Gotel and A. Finkelstein, “Extended Requirements Traceability: Results of an Industrial Case Study”, RE 1997.
[22] O. Gotel and S.J. Morris, “Out of the labyrinth: Leveraging other disciplines for requirements traceability”, RE 2011.
[23] O. Gotel, et al., “The Quest for Ubiquity: A Roadmap for Software and Systems Traceability Research”, RE 2012.
[24] P. Haumer, et al. , “Bridging the gap between past and future in RE: a scenario-based approach”, RE 1999.
[25] J.H. Hayes, A. Dekhtyar, J. Osborne, “Improving Requirements Tracing via Information Retrieval”, RE 2003.
[26] J.H. Hayes, et al., “Helping Analysts Trace Requirements: An Objective Look”, RE 2005.
[27] J. Hill and S. Tilley, “Creating Safety Requirements Traceability for Assuring and Recertifying Legacy Safety-Critical Systems”, RE 2010.
[28] E.A. Holbrook, J.H. Hayes, and A. Dekhtyar, “Toward Automating Requirements Satisfaction Assessment”, RE 2009.
[29] M. Jarke, et al., “Media-Assisted Product and Process Requirements Traceability in Supply Chains”, RE 2003.
[30] S. Katz and A. Rashid, “From aspectual requirements to proof obligations for aspect-oriented systems”, RE 2004.
[31] W.K. Kong, et al., “Process improvement for traceability: A study of human fallibility”, RE 2012.
[32] P. Laurent, J. Cleland-Huang, and C. Duan, “Towards Automated Requirements Triage”, RE 2007.
[33] L. Lavazza and G. Valetto, “Enhancing requirements and change management through process modelling and measurement”, RE 2000.
[34] J.C.S.P. Leite and A.P.A. Oliveira, “A client oriented requirements baseline”, RE 1995.
[35] J. Lin, et al. “Poirot: A Distributed Tool Supporting Enterprise-Wide Automated Traceability”, RE 2006.
[36] I.A. Macfarlane and I. Reilly, “Requirements traceability in an integrated development environment”, RE 1995.
[37] A. Mahmoud and N. Niu, “TraCter: A tool for candidate traceability link clustering”, RE 2011.
[38] J. Muñoz, et al., “Requirements Engineering for Pervasive Systems. A Transformational Approach”, RE 2006.
[39] P. Mäder, et al. “traceMaintainer - Automated Traceability Maintenance”, RE 2008.
[40] P. Mäder, O. Gotel, and I. Philippow, “Motivation Matters in the Traceability Trenches”, RE 2008.
[41] P. Mäder, O. Gotel, and I. Philippow, “Rule-Based Maintenance of Post-Requirements Traceability Relations”, RE 2009.
[42] N. Niu and A. Mahmoud, “Enhancing candidate link generation for requirements tracing: The cluster hypothesis revisited”, RE 2012.
[43] K. Ohashi, et al., “A means of establishing traceability based on a UML model in business application development”, RE 2011.
[44] M.C. Panis, “Successful Deployment of Requirements Traceability in a Commercial Engineering Organization...Really”, RE 2010.
[45] F.A.C. Pinheiro and J.A.Gogue, “An Object-Oriented Tool for Tracing Requirements”, RE 1996.
[46] K. Pohl, “PRO-ART: Enabling Requirements Pre-Traceability”, RE 1996.
[47] X. Pong, et al., “Lessons Learned from Building a Web-Based Requirements Tracing System”, RE 1998.
[48] H. Post, et al., “Linking Functional Requirements and Software Verification”, RE 2009.
[49] B. Ramesh and M. Edwards, “Issues in the development of a requirements traceability model”, RE 1993.
[50] B. Ramesh, et al., “Implementing requirements traceability: a case study”, RE 1995.
[51] G. Romanski, “Requirements, Configuration Management and Traceability for Safety Critical Software”, RE 2003.
[52] M. Sabetzadeh and S. Easterbrook, “An algebraic framework for merging incomplete and inconsistent views”, RE 2005.
[53] M. Sabetzadeh, et al., “Consistency Checking of Conceptual Models via Model Merging”, RE 2007.
[54] S. Siegl, K.S. Hielscher, and R. German, “Model Based Requirements Analysis and Testing of Automotive Systems with Timed Usage Models”, RE 2010.
[55] T.J. Simth, “READS: a requirements engineering tool”, RE 1993.
[56] C. Stanbridge, “Retrospective Requirement Analysis Using Code Coverage of GUI Driven System Tests”, RE 2010.
[57] A. Stone and P. Sawyer, “Exposing Tacit Knowledge via Pre-Requirements Tracing”, RE 2006.
[58] H. Sultanov and J.H. Hayes, “Application of Swarm Techniques to Requirements Engineering: Requirements Tracing”, RE 2010.
[59] K. Takahashi, et al., “Hypermedia support for collaboration in requirements analysis”, RE 1996.
[60] E.J. Uusitalo, et al., “Linking Requirements and Testing in Practice”, RE 2008.
[61] A. von Knethen, et al., “Systematic Requirements Recycling through Abstraction and Traceability”, RE 2002.
[62] J.H. Weber-Jahnke and A. Onabajo, “Finding Defects in Natural Language Confidentiality Requirements”, RE 2009.
[63] D.P. Wood, M.G. Christel, and S.M. Stevens, “A multimedia approach to requirements capture and modeling”, RE 1994.
[64] Y. Yu, et al., “Reverse engineering goal models from legacy code”, RE 2005.
[65] Y. Yu, et al., “Tracing and Validating Goal Aspects”, RE 2007.
[66] V. L. Axel, R. Darimont, and P. Massonet., "Goal-directed elaboration of requirements for a meeting scheduler: problems and lessons learnt." RE 1995
[67] E. Alexander, et al., "Value-based requirements traceability: Lessons learned." RE 2009.
[68] O. Inah, et al., "Use case to source code traceability: The developer navigation view point." RE 2009.
[69] Charrada, Eya Ben, Anne Koziolek, and Martin Glinz., "Identifying Outdated Requirements Based on Source Code Changes.". RE 2012.
[70] R. Salay, M. Chechik, and J. Horkoff., "Managing Requirements Uncertainty with Partial Models." RE 2012.