A Multivocal Mapping Study on Artifact Traceability Complexities in Practice

Zaki Pauzi[a] and Andrea Capiluppi[b]
University of Groningen, The Netherlands
{*a.z.bin.mohamad.pauzi, a.capiluppi*}*@rug.nl*

Keywords: Software Traceability, Multivocal Mapping Study, Evidence-Based Software Engineering.

Abstract: Artifact traceability is essential for managing the relationships between artifacts produced during the software development lifecycle, yet achieving effective traceability in practice remains a complex challenge. This study explores the multifaceted nature of traceability in real-world settings, providing actionable insights for researchers, practitioners, and tool developers aiming to enhance traceability practices, improve software quality, and support project success. Drawing from 56 academic papers and 15 grey literature sources, this study synthesises findings from scholarly research, industry reports, practitioner experiences, and expert opinions. Key challenges include the lack of standardised processes and tools, difficulties in maintaining traceability over time, balancing automation with human involvement, and fostering effective stakeholder communication and collaboration. Two critical open challenges emerge: achieving semantic interoperability and managing scalability in complex systems. To address these, we recommend targeted efforts towards standardisation and the development of incremental, adaptive techniques for traceability management.

SCITEPRESS SCIENCE AND TECHNOLOGY PUBLICATIONS

1 INTRODUCTION

Multivocal mapping studies (MMSs) in software engineering research offer a unique approach to synthesising diverse perspectives and insights on complex topics within the software engineering field. Unlike traditional systematic mapping studies that focus primarily on academic sources, MMSs incorporate a wide range of voices, including academic literature, industry reports, practitioner experiences, and expert interviews (Garousi et al., 2016; Neto et al., 2019). This was particularly chosen given that synthesising the data including sources from grey literature (e.g., white papers, technical documentation, blog posts) (Lefebvre et al., 2008) has immense value as they provide timely, context-specific, and diverse insights that complement academic research. This supports evidence-based decision-making in industry settings, given that our scope is focused on "in practice" (Garousi et al., 2016; Garousi et al., 2019) – we are focused on experiences and reports. As we look into the complexities of artifact traceability in practice, we considered sources outside of academic literature, albeit through a systematic process of scoping with inclusion/exclusion criteria to reflect high quality data that is in scope for our study. Solely focusing on scientific research will miss out on alternative perspectives and diverse voices from industry practitioners, consultants, and tool vendors. These are not typically published in academic settings.

The following research questions were outlined based on existing research and artifact traceability in practice, and will be assessed as part of the MMS:

RQ1: What are the demographics of reviewed literature?
Rationale: This information gives us an overview of the metadata of our sources. This is particularly important to better understand the impact and quality of our papers in scope.

RQ2: What are the reported key complexities in artifact traceability in practice?
Rationale: Through collating these, we are able to consolidate pain points and challenges. This allows us to understand the perils and pitfalls of artifact traceability in practice, so we can benefit researchers and practitioners alike in identifying these.

RQ3: What are the pertaining existing challenges?
Rationale: From the key complexities identified, we collate the themes and denote these as pertaining (open) challenges. This provides a collection of areas

[a] https://orcid.org/0000-0003-4032-4766
[b] https://orcid.org/0000-0001-9469-6050

Pauzi, Z. and Capiluppi, A.
A Multivocal Mapping Study on Artifact Traceability Complexities in Practice.
DOI: 10.5220/0013458400003928
Paper published under CC license (CC BY-NC-ND 4.0)
In *Proceedings of the 20th International Conference on Evaluation of Novel Approaches to Software Engineering (ENASE 2025)*, pages 754-761
ISBN: 978-989-758-742-9; ISSN: 2184-4895
Proceedings Copyright © 2025 by SCITEPRESS – Science and Technology Publications, Lda.

to focus on for actionable insights to address them, which are not in scope of this paper for now.

This paper aims to tackle these questions by conducting a thoroughly focused, yet comprehensive, multivocal mapping study. The scope of our study specifically focuses on *complexities* and *in practice* – relating to the difficulties of achieving traceability in practice.

2 BACKGROUND

Artifact traceability in practice refers to the ability to systematically document and track the relationships and dependencies between various artifacts created throughout the software development life cycle (SDLC). These artifacts can include requirements, design documents, code modules, test cases, and more. In practice, this involves establishing and maintaining links between these artifacts to ensure that changes made to one are reflected appropriately in others. Secondary studies looking into traceability of software artifacts are sporadic across different domains, with varying directions and focus in terms of reported findings and recommendations. For example, specific to requirements engineering (Tufail et al., 2017; Wang et al., 2018; Lyu et al., 2023; Saleem and Minhas, 2018) and focusing on machine learning applications (Pauzi and Capiluppi, 2023; Aung et al., 2020). To the best of our knowledge, there is not yet a study that includes grey literature to look into reported key complexities in artifact traceability. Our contribution to this area is much needed to formulate what needs to be focused on to address the key challenges pertaining artifact traceability in practice.

3 METHODOLOGY

For our MMS, we followed the published guidelines for conducting multivocal reviews by Garousi et al. (Garousi et al., 2019), namely in the following:

1. Search strategy and source selection (including study quality assessment)
2. Data extraction and synthesis
3. Report results (based on RQs)

Based on these steps, we present Figure 1, which shows the overview of our MMS methodology.

3.1 Search Strategy and Planning

We extracted the content and metadata of each piece of literature using a systematic approach and applied various tools to gather all publications necessary within our scope. For the academic literature, multiple library databases were used, such as ACM Digital Library and Scopus. For the grey literature, we used the same search string as the academic library database search, except a more manual approach had to be done (further reported in this section). This planning was done to ensure comprehensiveness in the study; to address the research questions at hand. Threats to the validity of our study strategy will be discussed in Section 5.

3.2 Search String

Table 1 shows the terms relevant to our search and their synonyms. These were derived to expand the boundaries of semantic keywords that are relevant to the research topics. We have separated the terms according to the relevant theme it belongs to, and only the most relevant synonyms (to our research questions) are shown in Table 1.

For any search strategy, the construction of the string is necessary as is enables transparency for validation and reproducibility. This search string is used for library database searches (further explained in this section) and used in web search engines for non-academic sources. An effective search strategy is usually iterative and benefits from trial searches using various combinations of search terms derived from the research question(s) (Kitchenham and Charters, 2007).

3.2.1 Trial of Potential Candidate Terms

Synonym terms are then evaluated through a robust process. Figure 2 shows the combination of terms that were tested. We grouped the synonyms according to common properties they share, denoted by the ovals. Each of these groups are then evaluated on effectiveness through trials and a decision is then made. Green coloured groups were those chosen.

3.2.2 Decision and Final String Output

- Theme 1: (top-down order) Main term and types of artifacts.
- Theme 2: (top-down order) Main term and offshoot terms.
- Theme 3: (top-down order) Main terms, types of complexities.
- Theme 4: (top-down order) Main term, synonym term (less common), synonym term (more common), parent term.

Figure 1: Filtering of academic literature based on selection process.

Table 1: Terms table.

Theme	Term	Synonyms
Software artifact	software artifact	source code, tests, documentation, requirements
Traceability	traceability	trace link recovery, trace retrieval
Complexities	complexity	difficulty, obstacle, barrier
In practice	in practice	real-world, industry, experience

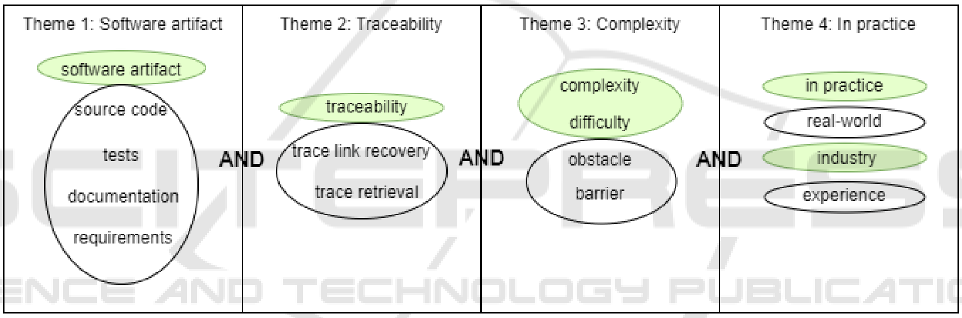

Figure 2: Grouped synonym terms: potential candidates for search string.

We specified the following search string (in order) to extract all related publications within our scope:

> (”software artifact” OR ”software artefact”) AND (”traceability” OR ”trace link”) AND (”complexity” OR ”difficulty”) AND (”in practice” OR ”industry”)

3.3 Source Selection – Inclusion and Exclusion Criteria

To ensure our results are reflective of recent research, we have imposed inclusion criteria in terms of period scope: years 2014 to 2023 inclusive. Spanning a period of a decade (ten years) in consideration, we aim to fill in the gap of studies that predated our start year and focus on more recent complexities of artifact traceability in practice. For exclusion, we have disregarded content that is unrelated to (software engineering) traceability, such as other reviews and non-complexities reported.

For the exclusion criteria, we used the following filters to filter out the papers that are not within our scope:

1. Duplicates: repeated entries
2. Language: non-English papers
3. Data: incomplete (missing) data
4. Reviews: other reviews, surveys, and mapping studies
5. Context: irrelevance to our defined research topics

3.4 Data Extraction for Peer Reviewed Literature

Table 2 shows the literature databases that were used for our first step in data extraction. The aim was to gather all relevant publications related to our study topics by using the search string defined. The extraction was done either by exporting from the web page (via manual extraction using the Web UI) or the API.

Table 2: Details of index databases used.

Database	Extraction type	Results
Scopus	API	3
Google Scholar	API	872
Springer Link	Web UI	51
ACM Digital Library	Web UI	35
IEEE Xplore	Web UI	8
WorldCat	Web UI	4
Total count		973

Google Scholar was first used to get the most results possible: despite the abundance of false positives, it has the potential to considerably extend the outreach of the systematic search (Harzing and van der Wal, 2008). What we observed was that using Google Scholar was enough to capture more than 95% of other results we obtained from the other databases. Regardless, we expanded our search beyond the search engine to ensure the comprehensiveness of our search strategy.

After the cleaning step instrumented by the exclusion criteria, we gathered a total of 56 papers held by libraries worldwide. We have also ensured that all these were peer-reviewed publications. These were extracted, along with the metadata, and compiled into a spreadsheet consisting of all the information and content for each paper. The full list of all selected papers in scope can be found online [1].

3.5 Data Extraction for Grey Literature

Beyond academic literature, we also expanded our search to official technical documentation, white papers, and case studies published by companies and reputable institutions. We specifically chose these outlets as our inclusion criteria for grey literature (first tier literature), following the quality assessment checklist for grey literature (Garousi et al., 2019). However, we make an exception for blog posts (originally categorised as the lowest tier in the guidelines) that are authored by the organisation themselves publishing about their products, as these sources do fulfil the criteria for a Tier 1 source. Given that our scope is targeted to "in practice", we evaluated sources of reports and conclude that only those published officially will be included in our study – this is based on the criteria of authority and outlet type, which complements our academic sources.

On top of the inclusion and exclusion criteria listed above, we added the following exclusion criteria for technical documentation and white papers:

1. Tool/platform does not present or mention traceability.
2. Tool/platform does not support end-to-end traceability.
3. Unofficial documentation (not authored by an official affiliate or endorsed).

For case studies, the following exclusion criteria were added:

1. Post/report not authored by official affiliate or endorsed.
2. Case study does not address fully or part thereof artifact traceability.

We ensured that these do not impede on our pursuit of comprehensive literature sourcing by including grey literature sources, while simultaneously ensuring high-quality reviewing by conforming to these criteria. The complete list of grey literature in the scope of our study is shown online [2], due to space constraints.

4 RESULTS

4.1 RQ1: Demographics

As part of the impact and quality analysis, we look into the sources and publishers of our academic literature. Our pie chart [3] shows the distribution of papers selected in scope for our study. The majority of our papers are from journals and conference proceedings. We have also included students' theses that have made the selection criteria.

For citation count per year, we can see 3 outliers in our box plot [4] – these are top cited publications per year, corresponding to the papers (Guo et al., 2017; Mahmoud and Niu, 2014; Abbas et al., 2022b). Correction has been made on one of the papers recently (Abbas et al., 2022a). Despite the citation count to be, arguably, a weak indicator of research quality for some (Aksnes et al., 2019), for the purpose of our study, we consider citation count as a factor in research impact, and we will analyse these further in Section 5.

[1] https://github.com/zakipauzi/enase-2025/blob/main/papers.csv

[2] https://github.com/zakipauzi/enase-2025/blob/main/gl_source.csv

[3] https://github.com/zakipauzi/enase-2025/blob/main/rq1_pie.png

[4] https://github.com/zakipauzi/enase-2025/blob/main/rq1_box.png

4.2 RQ2: Key Complexities

Based on the academic literature in scope, we have identified and grouped together four key complexities to effective artifact traceability in practice. We chose to only present the most common of them where each of these have been present in the papers (for simplification purposes). Although there are also cases where papers cover more than one complexity, we only present the main complexity that is the most relevant for each paper. Due to space constraints, we have uploaded the mappings between complexities to the papers online [5].

1. Lack of standardised processes and tools
2. Difficulty in maintaining traceability over time
3. Trade-offs between automation and human involvement
4. The need for effective communication and collaboration among stakeholders

4.3 RQ3: Open Challenges

Based on the key complexities identified in the previous section, we have also uncovered the following open challenges that organisations struggle with: *semantic interoperability* and *scalability in complexity*.

Achieving consistent and meaningful links between artifacts across diverse tools and domains remains a challenge due to differences in terminology, evolving artifacts, and varied stakeholder perspectives. Overcoming this challenge requires standardisation efforts, integrated frameworks, and automation techniques to ensure seamless communication and interpretation of traceability information. Managing traceability becomes increasingly difficult in large-scale software projects due to the sheer volume of artifacts, varying levels of granularity, and dynamic nature of software development. To address this challenge, efficient storage and retrieval mechanisms, intuitive visualisation tools, and adaptive traceability techniques are needed to cope with the complexities and scale of traceability information effectively.

5 DISCUSSION

In this section, we discuss the results of our MMS and the threats to the validity of our study.

[5]https://github.com/zakipauzi/enase-2025/blob/main/mapping_complexities.csv

5.1 RQ1: Demographics

As illustrated in our pie chart [6], the distribution of publication types reveals that the majority of selected papers are sourced from journal articles and conference proceedings. This finding underscores the significance of academic research in shaping our understanding of artifact traceability, with peer-reviewed journals and conference venues serving as primary outlets. Additionally, the inclusion of student theses and grey literature meeting our selection criteria highlights the diverse range of sources considered in our study, particularly as they report on case studies and tools. The analysis of citation counts per year further contributes to our understanding of the impact and influence of publications within the domain of artifact traceability. It is notable that three outliers (Guo et al., 2017; Mahmoud and Niu, 2014; Abbas et al., 2022b) emerge as the top-cited publications per year. In all of these papers, the authors employed machine learning in the semantic representation of artifacts to automate traceability.

5.2 RQ2: Key Complexities

The key issue of standardised processes and tools is reported as one of the key hindrances to effective artifact traceability in practice. Without established guidelines and uniform methodologies, organisations struggle to maintain consistency and synchrony in traceability practices across different stages of the software development life cycle (SDLC). The lack of standardisation often results in ad-hoc approaches to trace link creation, leading to inconsistencies, errors, and inefficiencies in traceability management. Consequently, stakeholders face difficulties in tracking and managing trace links, impeding their ability to accurately understand relationships between artifacts and make informed decisions based on traceability information.

The difficulty of maintaining traceability over time commonly relates to the increasing complexities of ever-evolving artifacts and their dependencies. Establishing trace links is commonly focused more during the early stages of the life cycle, and in some cases, visualisation tools are used to represent these traces. The issue becomes compounded when these traces are not maintained, and it becomes laborious and difficult to ensure traceability is updated. Most of the papers tagged to this issue propose tools and solutions to address this difficulty, although the challenge that comes with this does not necessarily dis-

[6]https://github.com/zakipauzi/enase-2025/blob/main/rq1_pie.png

appear. What used to be a manual task, traceability solutions using automated techniques with machine learning have taken the limelight in recent publications. The key issue with this, however, is the compromise between automation and human involvement.

The distribution of papers relevant to each identified issue is fairly equal with the exception of the fourth key issue: The need for effective communication and collaboration among stakeholders, which is also the main reported issue for all the grey literature in our scope. We have also observed that there are multiple instances where these issues overlap and lead to one another.

In answering RQ2, we had to cherry-pick the key complexities surrounding artifact traceability in practice, which is typically not the main focus of these papers, and sometimes can be obscured in the text. Nonetheless, these four that were identified enabled us to map to and derive existing open challenges that were considered to be persisting in practice. Figure 3 shows the mapping of these links.

5.3 RQ3: Open Challenges

5.3.1 Semantic Interoperability

In the context of artifact traceability, achieving semantic interoperability involves establishing and maintaining meaningful links between artifacts across various tools, platforms, and domains involved in the software development process. One of the key complexities contributing to the challenge of semantic interoperability is the heterogeneity of tools and data models used in software development. Different teams and organisations often employ a variety of tools for requirements management, version control, issue tracking, and testing, each with its own terminology and data structures. This diversity makes it challenging to establish meaningful connections between artifacts, as the same concept may be represented differently across different tools.

To address the challenge of semantic interoperability, efforts are needed in several areas, such as the following:

- Standardisation: Developing standardised ontologies, vocabularies, and data models that can be shared and reused across tools and domains to facilitate consistent interpretation and exchange of traceability information.
- Integration and Middleware: Building integration frameworks or middleware layers that enable seamless communication and data exchange between heterogeneous tools and systems, abstracting away the underlying differences in data formats and structures.
- Automation and Machine Learning: Leveraging automation techniques, such as natural language processing (NLP) and machine learning, to automatically infer and maintain traceability links based on textual, structural, and semantic similarities between artifacts (Pauzi and Capiluppi, 2023).

5.3.2 Scalability in Complexity

Scaling with complexities is not unique to artifact traceability, yet it remains as an open challenge that organisations have to handle daily. The sheer volume and dynamic nature of traceability information in large-scale software projects is a major contributing factor. As software systems grow in size and complexity, the number of artifacts, relationships between artifacts, and traceability links increases, posing significant challenges in managing, querying, and visualising traceability information effectively. Moreover, traceability information may need to be captured at various levels of granularity, from high-level requirements to low-level code elements. Managing traceability at different levels of abstraction and detail while preserving meaningful relationships between artifacts adds to the complexity of traceability management. This is also compounded with continuous changes, updates, and revisions throughout the development life cycle. To address scalability in complexity, we recommend innovative approaches and technologies that can do the following:

- Visualisation and Exploration: Develop intuitive visualisation techniques and exploration tools that enable stakeholders to navigate and analyse complex traceability networks, identify dependencies, and gain insights into the relationships between artifacts.
- Incremental and Adaptive Techniques: As traceability is ever evolving, so does the need for maintaining and managing the links. By focusing on techniques and tools that allow incremental and adaptive methods to manage traceability, we reduce the burden of tracing complexities as they evolve real-time. Smaller and more frequent trace link recoveries are much simpler to handle and maintain overtime.

At the backdrop of these challenges that were identified, there is some overlap with the grand challenge in traceability that was published more than a decade ago: making traceability *ubiquitous* in software and systems (Gotel et al., 2012).

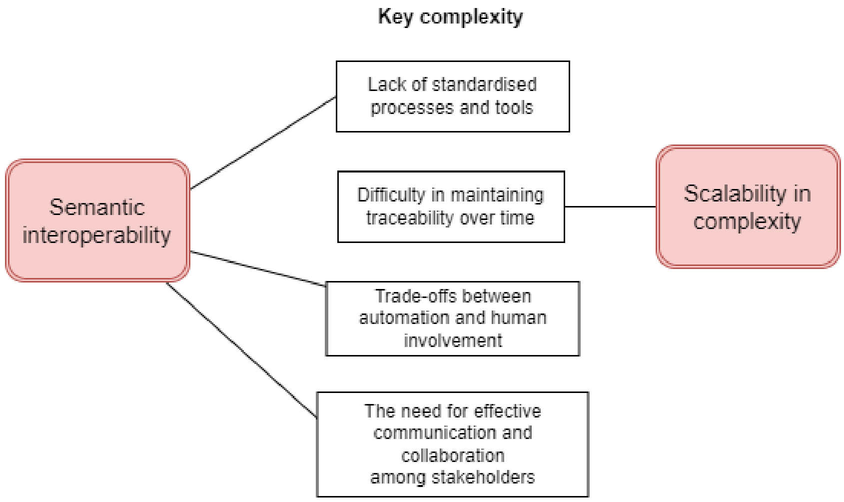

Figure 3: Mapping of key complexities to open challenges.

5.4 Threats to Validity of MMS

In this section, we outline the threats to validity identified throughout our mapping study process. Based on a recent map of threats to validity in systematic literature reviews in software engineering (Zhou et al., 2016), we looked into all possible similar threats that would emerge from conducting our MMS. The following are some key threats identified:

Construct validity – The chosen inclusion and exclusion criteria for evaluating literature may not accurately capture the nuances of complexities in artifact traceability in practice, leading to biased results. This is particularly true for published reports and case studies, given that the majority of published articles are biased towards successes and improvements. Regardless, we focused heavily on these challenges that may not be explicit in the literature; this is done with a thorough analysis and synthesis of available information.

Internal validity – The literature selected for our study may not be representative of the entire body of research on artifact traceability complexities, potentially skewing the conclusions drawn from the available evidence. By introducing grey literature, we expand the search scope to beyond academic literature, which is necessary given that our focus is “in practice”.

External validity – The papers in scope of our MMS may not be representative of the broader population, limiting the generalisability of the findings. Comprehensiveness of search is pivotal to address this threat, and this is why we used a search aggregate engine for literature that indexes multiple databases. Although more work is needed to be done to remove the false positives, we wanted to ensure that our findings can be generalisable.

6 CONCLUSION

In this paper, we conducted a multivocal mapping study (MMS) to explore the complexities of artifact traceability in software engineering practice. Our study addressed three key research questions: (1) the demographics of reviewed literature, (2) reported key complexities in artifact traceability, and (3) existing challenges pertaining to traceability.

Regarding the demographics of the reviewed literature, our analysis revealed that the majority of selected papers are from journal articles and conference proceedings, with a notable inclusion of student theses meeting our selection criteria. Furthermore, an examination of citation counts per year highlighted several top-cited publications. In terms of key complexities, our study identified several common challenges faced in achieving effective artifact traceability in practice. These include the lack of standardised processes and tools, difficulties in maintaining traceability over time, trade-offs between automation and human involvement, and the importance of effective communication and collaboration among stakeholders.

Our exploration of open challenges revealed two significant areas of concern: semantic interoperability and scalability in complexity. These challenges underscore the need for standardisation efforts, integrated frameworks, and automation techniques to address semantic inconsistencies and manage traceability at scale effectively. Overall, our findings provide valuable insights into the current state of artifact traceability in software engineering practice and highlight areas for further research and improvement.

6.1 Future Work

While our MMS has provided a comprehensive overview of the current landscape of artifact traceability, there are several avenues for future research and exploration in this area. First, future studies could delve deeper into specific industries or domains to understand how traceability challenges vary across different contexts. Additionally, further longitudinal studies could investigate the evolution of traceability practices over time and assess the effectiveness of interventions and tools in addressing identified challenges. It is clear that there is an imminent need for continued research and development of innovative tools and techniques to support artifact traceability in practice. This includes the exploration of automated tracing algorithms, integration of traceability mechanisms into existing development workflows, and the development of frameworks for assessing the quality and completeness of traceability information.

REFERENCES

Abbas, M., Ferrari, A., Shatnawi, A., Enoiu, E., Saadatmand, M., and Sundmark, D. (2022a). Correction to: On the relationship between similar requirements and similar software: A case study in the railway domain. *Requirements Engineering*, 27(3):399–399.

Abbas, M., Ferrari, A., Shatnawi, A., Enoiu, E., Saadatmand, M., and Sundmark, D. (2022b). On the relationship between similar requirements and similar software: A case study in the railway domain. *Requirements Engineering*.

Aksnes, D. W., Langfeldt, L., and Wouters, P. (2019). Citations, citation indicators, and research quality: An overview of basic concepts and theories. *SAGE Open*, 9(1):2158244019829575.

Aung, T. W. W., Huo, H., and Sui, Y. (2020). A literature review of automatic traceability links recovery for software change impact analysis. In *Proceedings of the 28th International Conference on Program Comprehension*, ICPC '20, page 14–24, New York, NY, USA. Association for Computing Machinery.

Garousi, V., Felderer, M., and Mäntylä, M. V. (2016). The need for multivocal literature reviews in software engineering: complementing systematic literature reviews with grey literature. In *Proceedings of the 20th International Conference on Evaluation and Assessment in Software Engineering*, EASE '16, New York, NY, USA. Association for Computing Machinery.

Garousi, V., Felderer, M., and Mäntylä, M. V. (2019). Guidelines for including grey literature and conducting multivocal literature reviews in software engineering. *Information and Software Technology*, 106:101–121.

Gotel, O., Cleland-Huang, J., Hayes, J. H., Zisman, A., Egyed, A., Grünbacher, P., Dekhtyar, A., Antoniol, G., and Maletic, J. (2012). *The Grand Challenge of Traceability (v1.0)*, pages 343–409. Springer London, London.

Guo, J., Cheng, J., and Cleland-Huang, J. (2017). Semantically enhanced software traceability using deep learning techniques. In *2017 IEEE/ACM 39th International Conference on Software Engineering (ICSE)*, pages 3–14.

Harzing, A. W. K. and van der Wal, R. (2008). Google scholar as a new source for citation analysis. *Ethics Sci. Environ. Polit.*, 8:61–73.

Kitchenham, B. A. and Charters, S. (2007). Guidelines for performing systematic literature reviews in software engineering. Technical Report EBSE 2007-001, Keele University and Durham University Joint Report.

Lefebvre, C., Manheimer, E., and Glanville, J. (2008). Searching for studies. *Cochrane handbook for systematic reviews of interventions: Cochrane book series*, pages 95–150.

Lyu, Y., Cho, H., Jung, P., and Lee, S. (2023). A systematic literature review of issue-based requirement traceability. *Ieee Access*, 11:13334–13348.

Mahmoud, A. and Niu, N. (2014). On the role of semantics in automated requirements tracing. *Requirements Engineering*, 20(3):281–300.

Neto, G. T. G., Santos, W. B., Endo, P. T., and Fagundes, R. A. (2019). Multivocal literature reviews in software engineering: Preliminary findings from a tertiary study. In *2019 ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM)*, pages 1–6.

Pauzi, Z. and Capiluppi, A. (2023). Applications of natural language processing in software traceability: A systematic mapping study. *Journal of Systems and Software*, 198:111616.

Saleem, M. and Minhas, N. M. (2018). Information retrieval based requirement traceability recovery approaches-a systematic literature review. *University of Sindh Journal of Information and Communication Technology*, 2(4):180–188.

Tufail, H., Masood, M. F., Zeb, B., Azam, F., and Anwar, M. W. (2017). A systematic review of requirement traceability techniques and tools. In *2017 2nd international conference on system reliability and safety (ICSRS)*, pages 450–454. IEEE.

Wang, B., Peng, R., Li, Y., Lai, H., and Wang, Z. (2018). Requirements traceability technologies and technology transfer decision support: A systematic review. *Journal of Systems and Software*, 146:59–79.

Zhou, X., Jin, Y., Zhang, H., Li, S., and Huang, X. (2016). A map of threats to validity of systematic literature reviews in software engineering. In *2016 23rd Asia-Pacific Software Engineering Conference (APSEC)*, pages 153–160, Hamilton, New Zealand. IEEE.