# Grey Literature Versus Academic Literature in Software Engineering: A Call for Epistemological Analysis

Garousi, V., & Rainer, A. (2020). Grey Literature Versus Academic Literature in Software Engineering: A Call for Epistemological Analysis. *IEEE Software*. Advance online publication. https://doi.org/10.1109/MS.2020.3022931

# Grey literature versus academic literature in software engineering: A call for epistemological analysis

Vahid Garousi, Austen Rainer, Queen's University Belfast

**Abstract**:

To learn about novel software engineering (SE) trends, where do you refer to? In order to document and disseminate their experience / knowledge, many SE practitioners prepare technical materials and share them online as blog posts, white papers and videos. Such materials are often called "grey literature" because they are not formally peer reviewed. By contrast, SE researchers write technical papers that are peer-reviewed and published as academic literature. We observe that, in general, these two communities mostly read literature that is only written by and published within their respective communities. This situation has led to a form of "knowledge divide" between the two communities that, we believe, hurts both communities. By characterizing and contrasting the two types of literature, grey and academic, we discuss how each literature can complement the other and can lead to a richer and more integrated knowledge exchange and dissemination in SE.

**Keywords:**

Grey literature; academic literature; information needs; evidence-based software engineering

## 1 Introduction

Software engineering (SE) practitioners and SE researchers form two large communities. According to one recent report (bit.ly/GlobalDeveloperDemographic Study), there were about 23 million software developers worldwide in 2018, and the number is estimated to reach 28 million by 2023. Furthermore, according to [1], as of 2012, about 4,000 individuals were actively publishing in major SE journals. These statistics show that the size of the SE researchers' community is very small compared to the size of the practitioners' community, a ratio of about 1:5,750.

To disseminate new approaches, knowledge and experience, members of each community write and share technical materials. Researchers write and publish research papers as peer-reviewed Academic Literature (AL), often as a required output of their research. On the other hand, many practitioners write articles and other online materials (such as blog posts and videos) and disseminate them. Such materials are collectively referred to as "Grey Literature" (GL). GL constitutes material that is published without the academic peer-review processes.

At the global level, there are some collaborations (mostly in the form of research projects) between the two communities of practitioners and researchers, however the number of collaborations is small relative to the size of the two communities [1]. Also, some ongoing positive efforts have been made to bring the two communities closer, e.g., establishing "industrial" paper tracks in several SE conferences, such as ICSE.

Similar to other fields, such as Information Systems (IS) [2], due to the limited level of collaborations, there is a limited level of knowledge exchange between the two communities. This is partly because each community has its own conferences and journals. For example, in software testing, one sub-area of SE, there are several academic conferences such as ICST (International Conferences on Software Testing), and ISSTA (International Symposium on Software Testing and Analysis), while there are many more industrial conferences (www.testingconferences. org), such as the STAR (Software Testing, Analysis and Review Conference) and GTAC (Google Test Automation Conference). Studies have pointed out that the focus areas of the two types of conferences are often disjoint [3], e.g., while most academic literature in testing tends to focus mostly on test-case design, most talks in industry conferences tend to focus on test automation. On a related note, participation in a given conference by members of the other community is, unfortunately, usually very low.

Practitioners rarely publish in academic periodicals, because they often cannot meet the standards of "rigor" required by the review process of academic venues. Academics tend not to publish in practitioner trade magazine because such magazines are not refereed and, therefore, do not usually "count" for academic CVs. As a result, the limited scale of knowledge flow, in both directions, hurts both communities. Many good ideas from SE research do not get disseminated into practice and vice versa. For example, by not being fully aware of industrial trends and challenges, many researchers work on topics that are often criticized for having low practical relevance [1].

To make the matters worse, practitioners also often do not read the AL [4]. This observation has also been reported in other fields, e.g., a paper in *Nature* [5] found that scientific papers "*are getting harder to read*" for practitioners; and a paper in medicine [6] argued that "*scientific papers are not written to disseminate information [to practitioners]*". Similarly, researchers often do not read (or cite) the GL, because they often do not consider such sources "scientific" or peer reviewed. They instead generally trust information that is published in journals using the peer-review process (bit.ly/GreyLiterature). As a result, the two communities and the two types of literature are unfortunately, to a large extent, disjoint. But studies have shown that using GL could clearly benefit SE research [7]. As we passed the 50th anniversary of SE (in 2018), there have been many recent papers addressing such themes as "*bridging the gap*", and linking "*research to practical advice*" [8].

We and many others wonder: Why there is such a knowledge divide between SE research and practice? The authors were motivated to write this paper after years of reading both academic literature and GL in SE and observing the differences between the two, and also from a recent Multi-vocal Literature Review (MLR) on practical relevance of SE research [1]. To shed light on the issue, we take an *epistemological* look at the two types of literature, and we "call" for further epistemological analysis on the topic. Epistemologists deal with questions such as: What is knowledge? How do we acquire knowledge? Those are among the questions that rise when comparing the industrial versus academic literatures types.

We characterize and contrast the two types of literature, so as to raise awareness of the benefits of reading both types of literature, and to encourage both communities (practitioners and researchers) to take the "knowledge divide" between the two communities more seriously, and to better appreciate the contributions of both literature types. Our vision is for a more inter-connected SE community in which both practitioners and researchers together build a more complete body of knowledge and practice for SE.

This paper draws upon the 45+ combined years of the authors' experience in working with, publishing and analysing both types of literature, e.g., [7, 9], and also from our discussions about the issue with numerous practitioners and researchers.

## 2 TWO COMMUNITIES, DIFFERENT MIND-SETS, DIFFERENT OBJECTIVES

As literature is produced by the respective communities, it makes sense to first briefly characterize those communities. Figure 1 presents a visualisation of the two communities, indicating how they relate to each other. Our general observation is that, while members of each community are busy writing and sharing technical materials, and also reading and learning from materials in their own community, they tend to read much less of the materials written by members of the 'other' community.

It is important to remind ourselves of the main goals of each community [10]. The main goals of practitioners is to develop software systems with reduced costs and higher quality. However, academic researchers' main job responsibility is to conduct research, and publish rigorous papers. Studies have shown that research "rigor" and industrial relevance can often be conflicting goals [1]. Of course, some researchers find ways to collaborate with industry and conduct research which is both rigorous and industrially relevant.

It is also important to characterise and contrast the decision-making models of the communities. Industrialists often follow a "naturalistic" decision-making approach [11] and the "satisficing" heuristic [12], in their demanding, real-world situations. These include situations marked by limited time, uncertainty, high stakes, team and organizational constraints, unstable conditions, and varying amounts of experience; conditions which are often the case in large-scale SE projects.

This is a different approach to the traditional approach of science and academia in which researchers identify all the options (alternatives) and criteria and then systematically evaluate each. Thus, academics follow analytic, optimizing, quantitative, 'God's-eye' view, "maximizing" decision-making, and exhaustive approaches, which is often in complete contrast to industry. We [1] and many others [13] have argued that, most academic SE papers so far have not adequately taken into account many of the real-world SE issues such as limited resources, importance of cost-effectiveness, team and organizational constraints, and unstable conditions, but instead have focused mostly on research "rigor" and "green-field" SE. Practitioners often complain [14] that "over-rigor" in academic studies goes against their satisficing decision-making strategy. On a similar note, we recognise that applied SE research undertaken with industry is often seen as not rigorous "enough" because it does not sufficiently generalise the problem and solution, and can therefore generally be harder to publish in academic journals [1].

Practitioners often do not have access to peer-reviewed journals, or the time or expertise to wade through scientific text. As a result, they are more likely to turn to other sources of information that is closer to their "context" and to which they can better "relate" (bit.ly/GreyLiterature). A recent IEEE Software paper [15] studied how software engineers form their opinions and found that personal experience ranked highest in terms of the source of knowledge. The next main sources of knowledge were peer opinion, mentor or manager, trade journals, and only lastly: research papers. The study observed that: "*research papers counted so little toward forming the opinions of professional practitioners*", an observation, which should be "*profoundly distressing*" to empirical SE researchers.

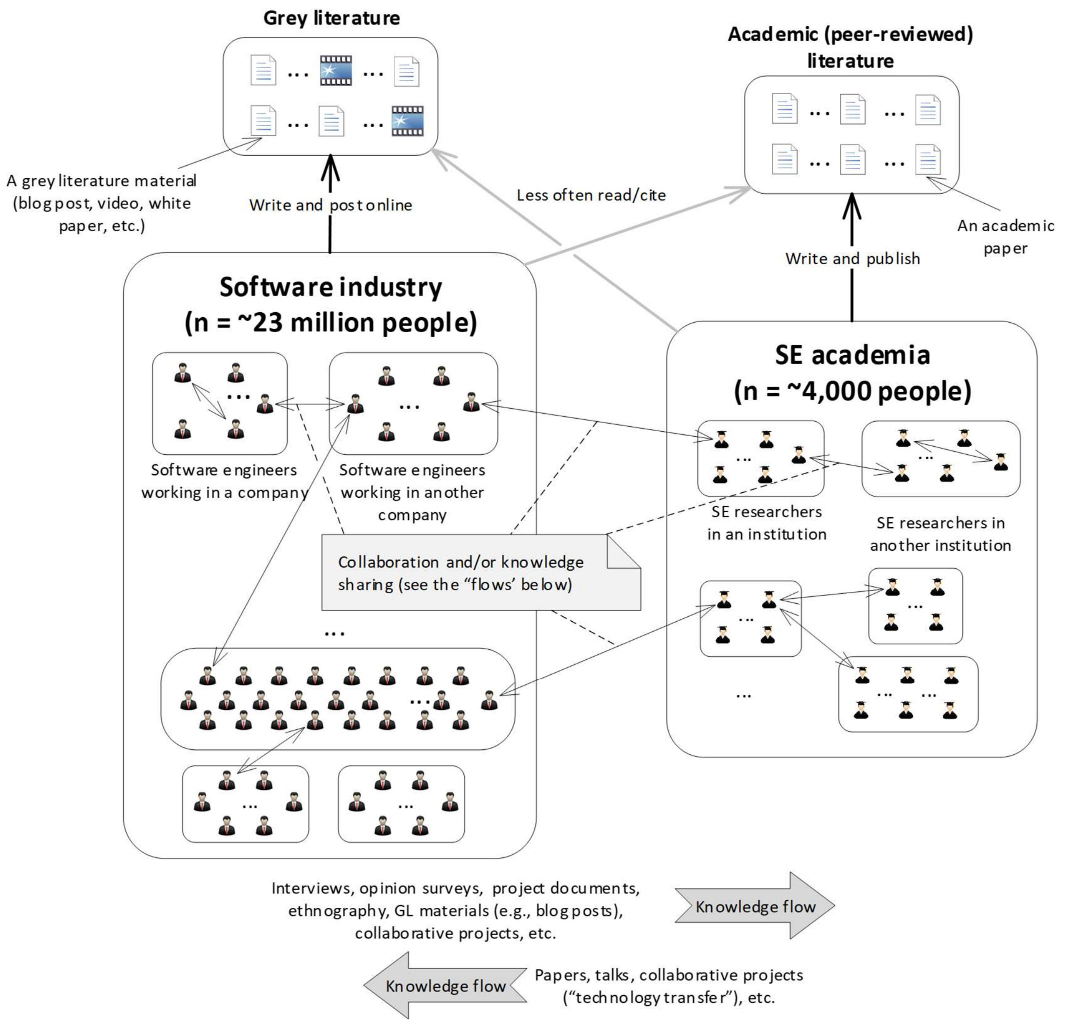
Grey literature
Academic (peer-reviewed) literature
A grey literature material (blog post, video, white paper, etc.)
Write and post online
Less often read/cite
An academic paper
Write and publish
Software industry (n = ~23 million people)
SE academia (n = ~4,000 people)
Software engineers working in a company
Software engineers working in another company
SE researchers in an institution
SE researchers in another institution
Collaboration and/or knowledge sharing (see the "flows' below)
Interviews, opinion surveys, project documents, ethnography, GL materials (e.g., blog posts), collaborative projects, etc.
Knowledge flow
Knowledge flow
Papers, talks, collaborative projects ("technology transfer"), etc.

**Figure 1- Visualizing the two communities (practitioners and researchers) and the knowledge literature produced by them**

## 3 Comparing the two literature types

We characterise and compare the two types of literature in Table 1. Note that the values presented in the table represent typical/generalized values. The comparison criteria were derived from the collective experience of the authors in working in academia and industry. To settle on the criteria listed in Table 1, the authors discussed a set of candidate criteria and iterated through several versions of the criteria. More specifically, characterization of the criteria in Table 1 was informed by findings from a recent MLR on relevance of SE research [1] and also the set of AL and GL sources from another MLR study [16]. For example, for the criterion "Rigor", many studies (as reviewed in [1]) have shown that GL materials have generally Low rigor, while AL sources are generally of High rigor.

Let us highlight that literature types have a *continuum*, thus outlets such as *IEEE Software* would be 'positioned' between the two types of literature shown, since they intend to bridge and serve the two communities.

Most of the criteria in Table 1 are self-explanatory. We therefore only discuss a few note-worthy items below. In terms of readability and understandability for practitioners, various papers, such as one in *Nature* [5] reported that scientific papers "*are getting harder to read*" for practitioners. On the other hand, since GL literature is prepared by practitioners for practitioners, and they do not have to satisfy the academic writing norms, the GL materials are often written in the most readable and understandable form. As an example, an IEEE Software paper [3] compared the focus areas of industrial and academic conferences in software testing and, by calculating quantitative readability metrics, it found that talk titles in industrial conferences were more readable than academic paper titles in testing.

In terms of type of advice offered, GL often provides concrete suggestions and insights, and they usually match what practitioners are looking for, i.e., know-how (how to do things). For instance, during our survey on when to automate testing [16], we found many such materials, e.g., in a presentation (goo.gl/QJNfes) entitled "*Choosing what to automate*", a practitioner shared concrete suggestions on the topic: "*Good candidates for test automation are short or simple tests, many data combinations, When expected results are stable, and when tasks that are difficult to do manually*". On the other hand, the AL in this area has proposed systematic approaches to the problem, and while these approaches could be useful, they will need effort to be tailored to and be applied in specific contexts [11]. In our discussions with several SE practitioners, we have seen that it is not always easy to convince them to apply such systematic techniques, which are often costly to apply.

As shown in Table 1, there are other limitations with AL which make it unattractive for practitioners to read, e.g., "*software developers need actionable guidance, but researchers rarely integrate diverse types of evidence*" in a way that meets those needs [8]. Also, "*it is common to find [SE research] results that are not directly applicable or are not relevant to the practice. Not to mention that when they are applicable, the results are usually buried in a highly technical propositional text written with the maximum rigor in mind, but not necessarily readability*" (pasemes.github.io/blog/what-is-evidence-based-practice).

As another perspective, a recent paper [17] distinguished two paradigms of SE research: rational and empirical. These two paradigms are "*disparate constellations of beliefs*" about how software "*should be [developed]*" (rational) versus how "*it is [actually] developed*" (empirical). It has been widely argued [18] that most SE papers have been written in the rational paradigm, thus making them less attractive and less realistic for practitioners. However, we could say that almost all GL literature is written in the empirical paradigm.

Furthermore, the paper [3] which compared the focus areas of industrial and academic conferences in software testing, found that many industrial conference talks present how-to advice, e.g., with titles such as "*How to do realistic mobile app load testing: device and network impact*". By contrast, academic papers often present new technologies or insights that, in most cases, have not been evaluated in real-world settings in terms of constraints and also considering the costs and benefits of applying a proposed SE technique [18].

Another important issue is that for the case of GL, it is generally hard to verify how accurately the shared opinions correspond to the reality. In AL, this risk is generally smaller.

**Table 1-Comparing the types of literature (showing typical or "generalized" cases)**

| **Criteria** | | ***Continuum* of literature** | |
|---|---|---|---|
| | | **Grey literature (GL)** | **Academic literature (AL)** |
| **Example types/venues** | | Blog posts, white papers and videos, talks in conferences | Journal papers, conference papers |
| **Type of problem rooted in** | | Almost always rooted in real problems | Rooted in abstracted SE problems, not often directly observed in practice. Also, often curiosity-driven |
| **Motivation to write / publish** | | To share experience, personal branding, evangelism and recruitment, personal knowledge repository, and soliciting feedback [19] | To share new knowledge ("advocacy" research [20]), to achieve academic targets |
| **Source of knowledge** | | Practice | Research / study |
| **Type of advice provided** | | (Often) Concrete: How-to's / insights | (Often) Generic (abstract). New techniques / approaches to do things with generic, vague and often unrealistic practical assumptions [1] |
| **Type of knowledge** | | Practical / tacit knowledge | Propositional / "Explicit" knowledge |
| **Distance / relevance to practice** | | Close / High relevance | (Typically) 'remote' [1] |
| **Dependency on specific industry context** | | Context-dependent | (Often) context-independent (generic) |
| **Rigor** | | Low | High |
| **Accessibility** | | High | Low (paywalls). The good news is that researchers increasingly share free pre-prints. Also, open-access journals help solve this issue. |
| **Readability and understandability, from practitioners' perspective** | | High | Low |
| **Quality-assurance** | | Online comments after publication. Page hits. Higher page-rank achieved by back-links | Peer-reviewed prior to publication |
| **Decision-making strategy** | | Intuitive: satisficing [12], qualitative, non-exhaustive, constraint-based (e.g., by time) | Analytic: optimizing, quantitative, exhaustive |
| **Volume of the materials** | | No exact estimate. In millions perhaps | 71,000+ papers as of 2016 [21] |
| **Motivation to read** | **Practitioners' motivation** | To inform decisions and actions | To see whether academic approaches can be used in practice |
| | **Researchers' motivation** | To see the latest trends in SE practice | To understand the past literature |

A recent experience report on applying software testing academic results in industry was published in [13]. The author of the study had tried to apply model-based testing, but one problem was that: "*None of the companies*" that he "*worked for in the past five years used any sort of modeling*". He then argued that "*it is important to always state where the models [to be used in model-based testing] come from: are they artificial or did they already exist before the experiments*" and that "*one has to argue and evaluate if the time and effort in developing and maintaining such models for a given system does pay off in the end*".

In summary, we should note that the differences between AL and GL cannot be taken for granted, without asking the question of why they are so significant. Many possible and often historical reasons have led to those differences over the decades since the inception of SE in the late 1960's [1], e.g., heavy emphasis on, and expectation of, high rigor over relevance in SE papers, *advocacy* research by SE academia, i.e., how software *should be* developed versus how software *is* developed in practice. We leave those questions to be investigated by future studies.

## 4 INTEGRATING KNOWLEDGE FROM GL AND AL

One way to integrate GL and AL is to survey and synthesize knowledge from the two literature types using Multi-vocal Literature Reviews (MLRs) [7]. A MLR is a form of a Systematic Literature Review (SLR) which includes the GL (e.g., blog posts, videos and white papers) in addition to the AL (e.g., journal and conference papers). MLRs are popular in other fields and have recently started to appear in SE. Examples of such studies are: a MLR on when/what to automate in testing [16], and a MLR on smells in software test code [22].

MLRs are useful for both researchers and practitioners since they provide summaries of both the state-of-the art and –practice in a given area. For example, when we recently asked a practitioner about our MLR on when/what to automate in testing [16], he mentioned: "*It is great for me and other test engineers in my team to read this MLR paper, since instead of reading all the 78 long documents (26 papers and 52 GL sources) reviewed by this study, we could read one single document which has nicely synthesized all the technical recommendations of those 78 documents. This is a valuable source of knowledge!*".

We extracted the number of GL and AL sources studied in nine previous MLRs in SE and depict the data in Figure 2. As Figure 2 shows, the MLRs cover different topics in SE. Depending on the SE topic under study, the studies have found different scales and ratio of GL versus AL. This indicates that while some areas have received more attention from practitioners (in terms of number of online materials written on the topic), other topics have received more attention from researchers. For example, the topic of the relationship of DevOps to Agile deployment, as expected, seems to have more GL than AL.

In the MLR studies that the first author was involved in (e.g., the ones on when/what to automate in testing [16], and smells in software test code [22]), we observed first-hand the differences between the two types of literature (Table 1). For example, for the decision-making strategies for when/what to automate in testing [16], the GL sources provided advice, which were intuitive, satisficing, qualitative, and non-exhaustive (in terms of all possible alternatives test cases). On the other hand, AL papers were almost all analytical, sophisticated, quantitative, and exhaustive. The fundamental issue here is that since practitioners mostly satisfice in their daily tasks, analytical and exhaustive approaches are just too "heavy" to be used in their context (as per the feedback that we have received from our industry partners [14]).

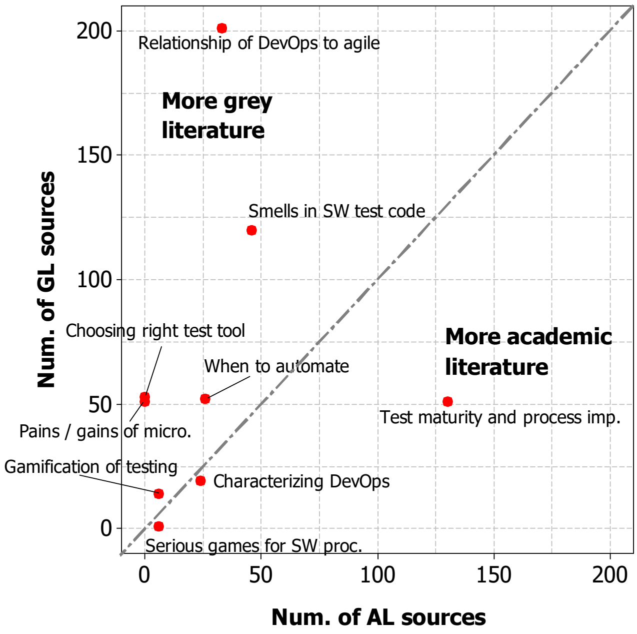

**Figure 2- Number of AL vs GL sources extracted from nine previous MLRs in SE**

## 5 Concluding remarks

With this paper, we encourage the community to pay more attention to the 'knowledge divide' and take further steps to address that divide and so connect the two types of knowledge. Different steps could and should be taken by the two communities (practice and research) to make a difference, i.e., a more inter-connected SE community in which practitioners and researchers advance the SE knowledge *together* and learn more actively from each other.

One action-item would be to increase the accessibility, readability, understandability and usefulness of AL for practitioners. Rather simpler issues such as practitioners hitting paywalls when trying to access papers are being addressed with open repositories such as ResearchGate and arXiv, and institutional repositories; but most other fundamental aspects in Table 1 should be addressed to increase knowledge flow in both directions. For instance, to 'encourage' practitioners to read more of the academic literature, researchers need to improve various aspects of their papers, e.g., while some academic papers have focused on sharing of experience (practical and experiential knowledge), which practitioners find useful, we still have many papers which report accumulation of propositional knowledge. We also need more papers to share experiences of applying SE techniques in the real-world.

Another action-item is for researchers to utilize more GL in their research, e.g., by including data and evidence from the huge number of blog posts written by practitioners, and simply attending or even watching online talk videos of industrial SE conferences which would help researchers to gain insights into the challenges and contemporary topics in SE industry. More discussions about the benefits of GL in SE research can be found in [23].

**Acknowledgements:** The authors would like to thank the anonymous reviewers for their insightful comments.

## References

[1] Vahid Garousi, Markus Borg, and Markku Oivo, "Practical relevance of software engineering research: Synthesizing the community's voice," *Empirical Software Engineering Journal, In Press,* vol. 25, pp. 1687–1754, 2020.

[2] Daniel L Moody, "Building links between IS research and professional practice: improving the relevance and impact of IS research," in *Proceedings of the international conference on Information systems*, 2000, pp. 351-360

[3] Vahid Garousi and Michael Felderer, "Worlds apart: industrial and academic focus areas in software testing," *IEEE Software,* vol. 34, no. 5, pp. 38-45, 2017.

[4] Sarah Beecham, Padraig OLeary, Ita Richardson, Sean Baker, and John Noll, "Who are we doing global software engineering research for?," in *IEEE International Conference on Global Software Engineering*, 2013: IEEE, pp. 41-50

[5] Philip Ball, "It 's not just you: science papers are getting harder to read," *Nature News,* 2017.

[6] Michael O'Donnell, "Why doctors don't read research papers: scientific papers are not written to disseminate information," *BMJ,* vol. 330, no. 7485, pp. 256-256, 2005.

[7] Vahid Garousi, Michael Felderer, and Mika V. Mäntylä, "Guidelines for including grey literature and conducting multivocal literature reviews in software engineering," *Information and Software Technology,* vol. 106, pp. 101-121, 2019.

[8] C. L. Goues, C. Jaspan, I. Ozkaya, M. Shaw, and K. T. Stolee, "Bridging the Gap: From Research to Practical Advice," *IEEE Software,* vol. 35, no. 5, pp. 50-57, 2018.

[9] Austen Rainer and Ashley Williams, "Heuristics for improving the rigour and relevance of grey literature searches for software engineering research," *Information and Software Technology,* vol. 106, pp. 231-233, 2019.

[10] A Davis, *Great software debates*. Wiley-IEEE Press, 2004.

[11] Akash Manjunath, Manoj Bhat, Klym Shumaiev, Andreas Biesdorf, and Florian Matthes, "Decision making and cognitive biases in designing software architectures," in *IEEE International Conference on Software Architecture Companion*, 2018, pp. 52-55

[12] Antony Tang and Hans van Vliet, "Software designers satisfice," in *European Conference on Software Architecture*, 2015: Springer, pp. 105-120

[13] Andrea Arcuri, "An experience report on applying software testing academic results in industry: we need usable automated test generation," *Empirical Software Engineering,* 2017.

[14] Vahid Garousi, Matt M. Eskandar, and Kadir Herkiloğlu, "Industry-academia collaborations in software testing: experience and success stories from Canada and Turkey," *Software Quality Journal,* vol. 25, no. 4, pp. 1091–1143, 2017.

[15] Prem Devanbu, Thomas Zimmermann, and Christian Bird, "Belief and Evidence: How Software Engineers Form Their Opinions," *IEEE Software,* vol. 35, no. 6, pp. 72-76, 2018.

[16] Vahid Garousi and Mika V. Mäntylä, "When and what to automate in software testing? A multivocal literature review," *Information and Software Technology,* vol. 76, pp. 92-117, 2016.

[17] Paul Ralph, "The two paradigms of software development research," *Science of Computer Programming,* vol. 156, pp. 68-89, 2018.

[18] Robert L. Glass, "A deja-vu look at software engineering researchers who care about practice," *Communication of ACM,* vol. 50, no. 8, pp. 21-23, 2007.

[19] Chris Parnin, Christoph Treude, and Margaret-Anne Storey, "Blogging developer knowledge: Motivations, challenges, and future directions," in *IEEE International Conference on Program Comprehension*, 2013, pp. 211-214

[20] Robert L Glass, "The software-research crisis," *IEEE Software,* no. 6, pp. 42-47, 1994.

[21] Vahid Garousi and Mika V. Mäntylä, "Citations, research topics and active countries in software engineering: A bibliometrics study " *Elsevier Computer Science Review,* vol. 19, pp. 56-77, 2016.

[22] Vahid Garousi and Barış Küçük, "Smells in software test code: A survey of knowledge in industry and academia," *Journal of Systems and Software,* vol. 138, pp. 52-81, 2018.

[23] Vahid Garousi, Michael Felderer, Mika V Mäntylä, and Austen Rainer, "Benefitting from the grey literature in software engineering research," in *Contemporary Empirical Methods in Software Engineering*: Springer, 2020.

## AUTHOR BIOGRAPHIES

Vahid Garousi is an Associate Professor in Queen's University Belfast, and also the Director of Bahar Software Engineering Consulting Corporation. His areas of research and practice include automated software testing, empirical software engineering and industry-academia collaborations. He received a PhD in software engineering from Carleton University. Contact him at v.garousi@qub.ac.uk. More info at: www.vgarousi.com

Austen Rainer is a Professor in Queen's University Belfast, UK. His research interests include behavioural software engineering. He received his PhD in software engineering from Bournemouth University. Contact him at a.rainer@qub.ac.uk