Guidelines for Knowledge Translation in Software Engineering

Deepika Badampudi[a,*], Claes Wohlin[a], Tony Gorschek[a]

[a]*Blekinge Institute of Technology, Karlskrona, Sweden*

Abstract

Context - Transfer of software engineering research into industrial practice is a challenge; academic articles are not the best enabler for industrial adoption of research. Furthermore, synthesis of knowledge from multiple research studies is needed to provide evidence-based decision-support for industry. However, translation of research outcomes (for e.g., results of a systematic review) to provide recommendations for practitioners is seldom practiced.

Objective - The objective of this paper is to provide a knowledge translation framework in software engineering research, in particular to translate research evidence into practice by combining contextualized expert opinions with research evidence.

Method - We adopted the framework of knowledge translation from healthcare research and combined it with the Bayesian synthesis method for use in software engineering research and practice. We evaluated the outcome of the knowledge translation framework along with the effectiveness of the interventions undertaken as part of knowledge translation in two cases.

Results - The framework provided in this paper includes a detailed description of each step of the knowledge translation. In the evaluation of the knowledge translation framework we found that, in comparison to the prior opinions practitioners become more consensual after the integration of opinions and knowledge from research studies.

Conclusions - Knowledge translation using Bayesian synthesis provides a

*Corresponding author

systematic approach towards contextualized, collaborative and consensus-driven application of research results. Dissemination of research results, transfer of research outcomes and sharing tacit knowledge has been done to some extent to provide explicit knowledge. However, there is limited research on how to translate research outcomes to make the research more contextualized and hence more easy, accessible and acceptable by industry. In conclusion, this paper contributes towards the application of knowledge translation in software engineering through its framework and a first evaluation of its use.

Keywords: Knowledge translation, Bayesian synthesis, decision-making
2010 MSC: 00-01, 99-00

1. Introduction

The knowledge produced by researchers should be applied in practice [1, 2]. It should be translated beyond the dissemination and towards the application of the knowledge in practice. Budgen et al. [2] suggest that knowledge should be presented in the form of recommendations that enable and support evidence-informed decision-making in software engineering (SE) practice. This is reflected in the fourth step of the five-step process for adapting the practices of evidence-based software engineering (EBSE) which is referred to as knowledge translation (KT) [1, 2]. A large survey at Microsoft indicates that the developers rely on and are more influenced by their "personal experience" than evidence from research [3]. Devanbu and Zimmerman identify this as a problem and suggest that greater efforts should be made to translate knowledge so that practitioners are informed and rely on verified evidence, rather than prior opinions which could be "biased, error-prone and spotty" [3].

Greenhalgh and Wieringa [4] argue that "*objective, impersonal research findings*" are unhelpful. In addition, research might not always be straightaway applicable, relevant and suitable for practice according to Cartaxo et al. [5]. Therefore, KT should not be viewed as just supplying the research outcomes to professionals rather aligning the knowledge to a specific context.

In addition, Devanbu and Zimmerman [3] point out that by not accepting research results software practitioners might be at risk of relying too much on their personal opinion and experience. They regard subjective and personal recollection to be notoriously error-prone [3].

It is clear that both objective and impersonal knowledge from research studies and subjective and personal opinions of practitioners together should be considered for impactful implementation of research results. Therefore, knowledge translation should be an integration of knowledge from research studies and practitioners' opinions. However, it is not often practiced [5]. *Kitchenham et al. consider KT as a research activity involving researchers, subjective opinions of practitioners and policy-makers/decision-makers to make evidence-informed decisions [1].*

What is KT? - KT in SE is defined as "the exchange, synthesis and ethically sound application of knowledge - within a complex system of interactions between researchers and users - to accelerate the capture of the benefits of research through better quality software and software development processes" [1]. The three main aspects of KT are "exchange", "synthesis" and "application". The Bayesian synthesis method proposed in [6] supports KT as it synthesizes data and provides interpretation of the research outcome in the application context by incorporating knowledge from research studies and experience of intended users.

What is not KT? - Different processes may be undertaken after the completion of a research study, for example, dissemination, knowledge transfer and KT. Dissemination is spreading the knowledge in different scientific channels. Whereas, knowledge transfer is an activity of making knowledge explicitly available in a form that allows users to use explicit information that is applicable to them in their specific context. The key aspect that differentiates KT is that it focusses on the application of contextualized knowledge in practice. Therefore, it is beyond dissemination and making knowledge available (knowledge transfer) to practice. It is also different from technology transfer as it involves the subjective opinions of practitioners and it involves the process of actual use of knowledge in practice.

Why KT? - Based on the classic two-community theory, the two communities of academia and practice are not always synced due to differences in perspectives and cultures [7]. A survey indicates that developers' knowledge is formed based on personal experience and opinion and far less on research results [3]. This is also identified by Ayala et al., they found a gap between decision-making processes proposed in research and the processes used in industry. [8]. As a result, knowledge produced in research is too rarely implemented in practice. This is even more problematic if the studies do not provide explicit recommendations or guidelines to practice. Very few secondary studies in SE provide recommendations for practitioners [9, 2].

Due to the differences in software engineering research and practice, the translation of knowledge into practice is a complex research activity and not always straightforward. Practitioners are often busy which makes it difficult for them to keep up with research. However, even if practitioners are presented with strong and reliable evidence, it is often difficult for practitioners to accept the evidence and agree that it applies to them [10]. The need to transform evidence to local opinion [10] and the need to put more effort and systematize the dissemination of research findings is identified [3]. Budgen et al. [2] state that KT in SE is done in an ad-hoc manner and lacks adequate documentation. In addition, they highlight that "*KT should itself be systematic and repeatable as possible, and it should also reflect the needs and mores of practitioners as well as of the different forms of organizational context within which they work*" [1, 2]. The need to develop guidelines for undertaking KT in SE has been identified [2]. The aim of this paper is to provide guidelines in the form of a knowledge translation framework using Bayesian synthesis.

The remainder of the paper is structured as follows. Section 2 describes the background and the related work. We then provide the description of the KT framework using Bayesian synthesis in the context of SE in Section 3. The evaluation and reflections from application of KT in a case study are presented in Section 4. Finally in Section 5 we discuss our findings in relation to previous research and conclude in Section 6.

2. Background and Related Work

Different approaches to support software practice have been proposed. We discuss different approaches from knowledge sharing to knowledge translation in this section.

2.1. Sharing tacit knowledge

Frameworks like the quality improvement paradigm (QIP) [11] and the experience factory [12] have been proposed to utilize the knowledge and experience of software practitioners. The QIP framework [11] consists of two cycles - control cycle and capitalization cycle. The control cycle is about project learning to provide immediate solutions to the problems for on-going projects. The capitalization cycle is about capturing experience and transfer learning from one project to other applicable projects. The five steps in the capitalization cycle are - analysis of project results, packaging and storing practitioners' experience, characterizing and understanding the experience, setting goals for improvement and finally choosing processes, methods, techniques and tools for implementing the process improvement.

The experience factory [12] is a repository to store experiences and lessons learned from working in projects and providing the experience to applicable projects on demand.

2.2. Academia-industry collaborations

Studies point out that opinions based on personal experience might be biased, subjective and error-prone [10] and [3]. Therefore, relying just on personal opinions and experience might not be the best approach for practitioners. In addition, the importance of interaction between researchers and practitioners to integrate evidence from research results (knowledge) and practitioners' opinion and experience has been identified. Evidence-based software engineering guidelines emphasize such interactions [1].

Academia and industry collaborations are important for better utilization of research results and for guiding future research directions [13] and [14]. Initiation from both researchers and practitioners is required for effective research collaboration [13]. According to a study [14], regular and continuous collaboration among researchers and practitioners with major contribution from researchers leads to better implementation of research in practice.

Case studies have been identified as an effective research method for implementation of research in practice [15]. According to an experiment conducted in Microsoft [16], 71% of practitioners had a positive opinion on software engineering research studies. An updated study [17] on research published in the empirical software engineering and measurement conference, indicates that practitioners are receptive towards research. However, even though the research results are sound, it still might be difficult to convince practitioners to implement the results [10], [3].

2.3. Technology transfer

Technology transfer in software engineering has been recognized as an important activity [18]. Over the years it has been improved and refined to make it more collaborative and a technology pull rather than a technology push activity [19]. Gorschek et al. proposed a model that validates a candidate solution in academia followed by a static and dynamic validation in practice before releasing as a solution. A need to make the technology transfer activity more efficient and effective has been identified [20]. Mikkonen et al. emphasize the importance of co-creation and co-learning instead of a one-way transfer from research to practice. The KT framework proposed in this paper is a complementary solution to the technology transfer model proposed in [20]. However, KT framework focuses on adaptation and translation of research results in general and not only transfer of a technology or a candidate solution.

2.4. Knowledge transfer and translation

To aid the implementation of research in practice, an approach to capture and share the tacit knowledge to make it explicit and widely available has been proposed by Cartaxo et al. [5]. They present and evaluate a template called evidence briefings to be used to summarize the findings from systematic reviews. Evidence briefings transfer the research results in an attractive format for practitioners. However, it does not translate the knowledge into the context and does not focus on application of the knowledge in practice. Although, evidence briefings can be used to facilitate knowledge translation. We have depicted the use of knowledge transfer (evidence briefings) in knowledge translation in Figure 1. The key difference in knowledge translation is that a relevance jury consist-

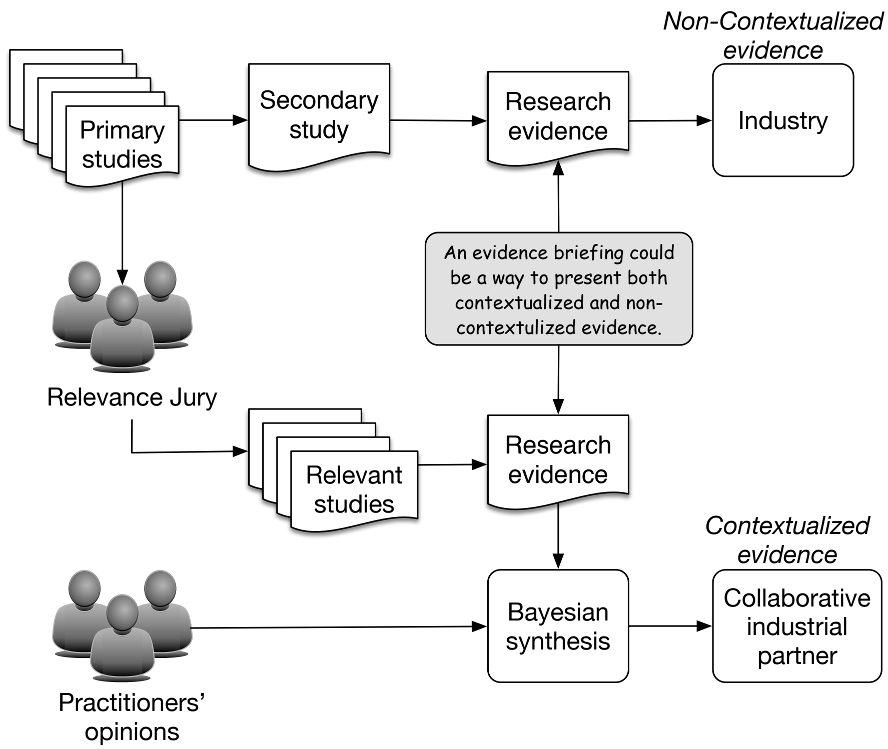

Figure 1: The use of knowledge transfer (evidence briefings) in knowledge translation.

ing of both researchers and practitioners discuss the relevance of the primary studies to identify relevant studies. The findings from these relevant studies is then aligned to the local context.

In healthcare Bayesian approaches to integrate qualitative and quantitative evidence with opinions and knowledge have been proposed [21] and [22] and applied with some customization in [18], [23], [24] and [25]. A customization and application of Bayesian synthesis to align external knowledge to local context by integrating opinions and evidence in software engineering has been proposed [6]. Bayesian synthesis consists of three steps, in the first step the prior opinions of practitioners is collected. In the second step, the evidence from the research results is presented to the practitioners. This evidence could be presented in the format of evidence briefings. In the last step, the practitioners are asked about their opinion in the light of the knowledge from the research results. We have used Bayesian synthesis in our KT framework to align the external knowledge from research studies to practitioners local context.

3. Description of the KT framework

3.1. Overview of the KT framework

In this section, we describe the KT framework which is adopted from healthcare research [7]. Figure 2, represents the overview of the knowledge to action activity. It consists of two main parts: knowledge creation (represented as knowledge creation funnel) and knowledge application (represented as knowledge translation cycle). The knowledge creation funnel shown in Figure 2 consists of the following steps -

- Knowledge inquiry - It is done through primary studies.
- Knowledge synthesis - It is done through secondary studies.
- Knowledge tools/products - It presents the knowledge in end-user friendly format based on the needs of the end-user (e.g. - evidence briefings [5]).

As the knowledge passes through the knowledge creation funnel it gets more tailored to the end-users' needs. The tailoring in the knowledge creation funnel is done by filtering the knowledge that could be applicable to the end-users. For exmaple, the tailoring is done by filtering information that is not relevant to the

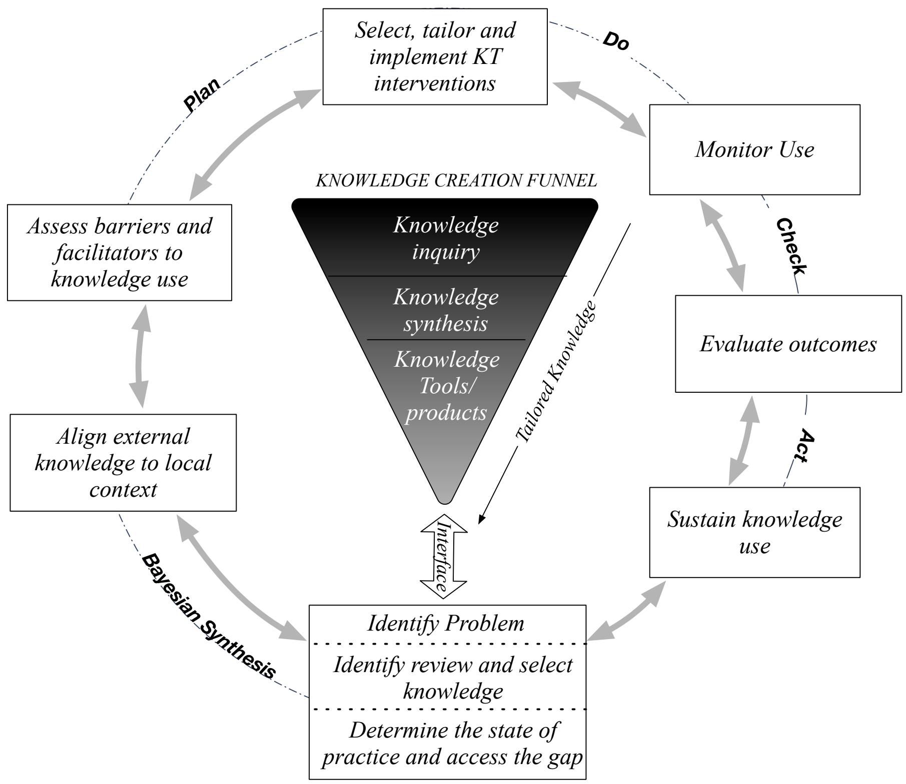

Figure 2: Knowledge translation framework (Adapted from Graham et al. [7]).

users such as methodological details. It is different from the tailoring based on the context of knowledge application which is part of knowledge translation.

The knowledge creation funnel not only provides input to the KT cycle, it also accepts input from the KT cycle hence, depicted by the double ended arrow (<==>) and labeled as "interface" between the knowledge creation funnel and KT cycle. This is further elaborated in Section 3.2 and Figure 3. The main purpose of the KT cycle is to implement knowledge in practice. As mentioned earlier, KT is a research activity that goes beyond dissemination and towards the application of the knowledge.

The KT cycle is related to the application of the knowledge/evidence. In Figure 2, the KT cycle is represented in relation to Bayesian synthesis [6] and generic improvement paradigms such as the PDCA cycle (Plan-Do-Check-Act) [11]. Due to the uncertainty involved in software engineering practice, it is important to adapt the knowledge to the local context before it is implemented in practice. Therefore, it is important to customize the external knowledge and adapt it to the local context, this is mainly done through the Bayesian synthesis part of the KT cycle (discussed in detail in Section 3.2). Once, the contextualized knowledge is identified, the next step is to implement the contextualized knowledge using the PDCA cycle. The implementation is done by assessing the current situation and identifying interventions that are necessary for knowledge implementation and then monitor and evaluate before sustaining it as an industrial practice.

3.2. Detailed working of the KT framework

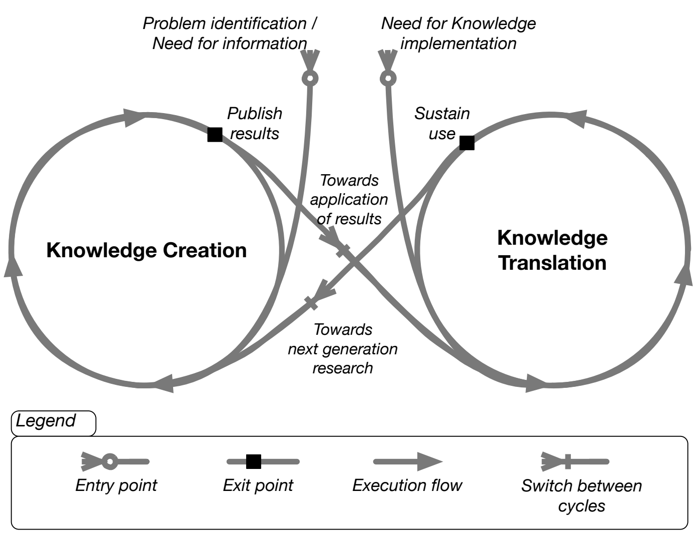

Figure 3: Knowledge creation cycle (Adapted from Straus et al. [26]).

The steps involved in producing primary and secondary studies or tools (within knowledge creation) are cyclic in nature. Therefore, the knowledge creation funnel in Figure 2 is represented as a cycle together with the knowledge translation cycle in Figure 3. As see in Figure 3 the knowledge creation cycle and

KT cycle are connected and provide input to each other. The entry point for the cycles depends on the need for knowledge creation or implementation indicated by an arrow head with a circle and labeled as "entry point". Apart from these entry points the execution of the cycles can continue from the previous cycles as indicated by the arrows with a line and labeled as "switch between labels". The arrows indicate the execution flow in the knowledge creation cycle and the knowledge translation cycle. Each cycle also has exit points indicated by a "square" which marks the end of the cycle. For example, publishing research is the exit step of the knowledge creation cycle.

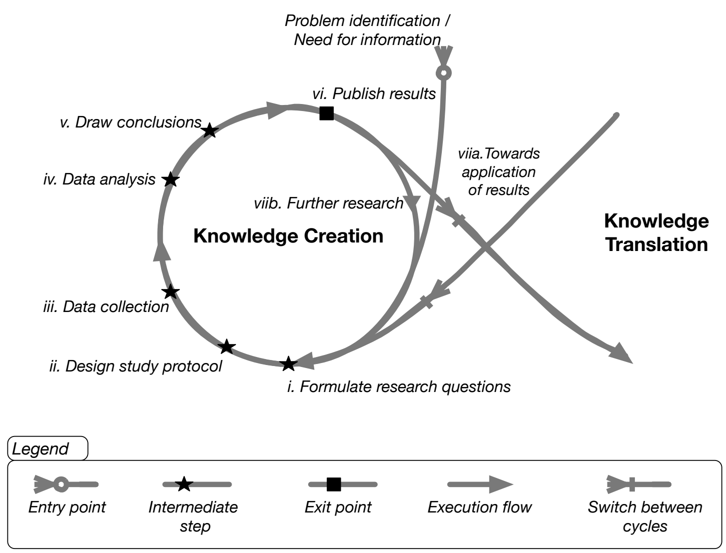

Figure 4: Knowledge creation cycle (Adapted from Straus et al. [26]).

Steps i-vi in Figure 4 are part of the knowledge creation cycle, Step vii is a switch which points either towards application of knowledge (Step viia) or a continuation in the knowledge creation cycle (Step viib). Since the focus on this paper is not on knowledge creation we will focus on the KT cycle i.e. Steps 1-9 in Figure 5.

These steps can be further categorized into steps (1-4) to make knowledge contextualized using Bayesian synthesis and steps 5-9) to carry out the knowl-

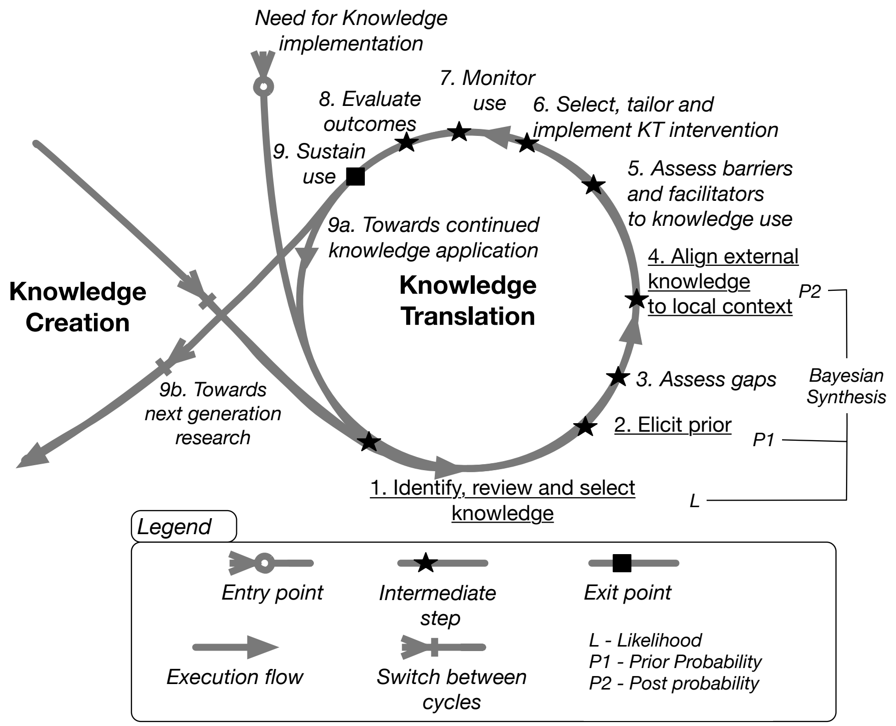

Figure 5: Knowledge translation cycle using Bayesian synthesis (Adapted from Straus et al. [26]).

edge implementation. Step 9 is a switch, which points either towards continued application of knowledge (Step 9a) or towards the creation of the next generation research based on the lessons learned from implementing knowledge in practice (Step 9b).

Figure 5, depicts the detailed working of KT in connection with the use of Bayesian synthesis [6].

Entry point for knowledge translation - Need for knowledge implementation: It is done by identifying the need for improvement in practice for which relevant knowledge from literature exists or has been created that could be used in practice. In order to implement evidence in practice, it is useful if there is a local champion and there is support from management. Once the topic is identified then the required knowledge is identified, reviewed and selected.

The description of Steps 1 to 9 in Figure 5 are as follows:

1. **Identify, review and select knowledge:** In this step, the knowledge that could be used in practice to address the need for improvement is identified, reviewed and selected. In addition, the quality, strength and applicability of the knowledge is determined so that the knowledge is correctly interpreted. The quality of the research studies can be evaluated using the rigor and relevance criteria [27] and the strength of knowledge/evidence can be determined using guidelines in [28] and [29]. It is important to determine if the knowledge is based on a general context or a specific context. A simple literature review or a systematic literature review can be conducted to identify, review and select knowledge.

2. **Elicit prior (Prior probability):** In this step, the current state-of-practice in terms of probabilities are elicited from the relevant stakeholders. It involves the following two sub-steps:

   - Selecting individuals - In SE, the subjective opinions of practitioners, decision-makers/policy-makers are relevant. The selection of individuals who will be the users' of the knowledge is important. For example, if a decision needs to be made in a software project then, all the practitioner roles that should be involved in making the decision should be selected to elicit their subjective opinions.

   - Eliciting opinions - Opinions and experiences are elicited to collect prior probability. Opinions and experiences of practitioners related to the area of improvement should be elicited. For example, if the test automation process needs to be improved then the practitioners' experiences and opinions on working with test automation are to be elicited. Prior probability can be captured in terms of percentage ranging from 0 to 100 % or in terms of absence/presence (0/1) of a parameter value.

3. **Assess gaps:** In this step, the size and nature of the gap between the current and desired knowledge, skills, and outcomes are assessed. Quality

indicators can be used to assess the gap. For example, what is the current and desired level to achieve the goal?

4. **Align external knowledge to local context:** In this step, individuals or groups of decision-makers go through the knowledge identified in Step 1 and determine the value, usefulness and validity of the knowledge in their context. Before knowledge from research studies is provided to the practitioners, the terminology used in the literature should be adapted to the language and terminology followed in the company. This adaptation can be done with the help of the local champion.

   The knowledge can be summarized in the form of likelihood calculations. The likelihood is the representation of what is known. In other words, it is the summary of all the research studies within a specific research objective. The likelihood is calculated as the percentage of the research studies reporting a finding. For example, the likelihood of 60 % indicates that 6 out of 10 research studies are reporting a particular finding. More examples on how the likelihood can be presented is mentioned in our previous work [6]. The likelihood calculation is not always the indication of the importance of a finding. For example, if the likelihood of Finding F1 is 80% and Finding F2 is 70%, it does not mean that F1 is more important. It only means that Finding F1 is more often researched. Therefore, the likelihood does not always indicate importance. The importance of a research finding needs to be judged by the practitioners, and hence findings from research may be seen as a checklist for contextualizing the knowledge related to different research findings.

5. **Assess barriers and facilitators to knowledge use:** In this step the barriers and facilitators that restrict or help implement knowledge are identified. The barriers and facilitators are useful in understanding the intentions of the practitioners. Barriers for knowledge use could be related to knowledge, attitudes, skills, habits or the like of the potential adopters according to Graham et al. [7]. The possible interventions for such barriers could be interactive educational workshops or training and

documenting the knowledge in the form of company standards.

6. **Select, tailor and implement KT intervention:** Based on the identified barriers and facilitators the KT intervention for example training is selected, tailored and implemented in this step.
7. **Monitor use:** Once the knowledge is in use or has been applied, it should be monitored. To be able to monitor the knowledge use, it is necessary to know the type of knowledge use. Using the classification scheme by Straus et al. [26] we identify the use and monitoring of use in software engineering as shown in Table 1.

Table 1: Types of knowledge use and monitoring

Knowledge use	**Description**	**Examples of measures**	**Strategy for data collection**
Conceptual	Use of knowledge to change the levels of knowledge, understanding or opinions of the software practitioners.	Practitioners' knowledge, attitudes and/or intentions to change.	Survey and/or interviews
Instrumental	Use of knowledge to change a software practice or practitioners' behavior.	Adherence to recommendations (e.g. change in the testing process or adoption of an innersource strategy).	Archival or document analysis
Persuasive	Use of knowledge as ammunition to attain specific power or profit goals. For example, use of knowledge to justify a decision.	Decision-makers' or policy-makers knowledge	Survey and/or interviews

Persuasive knowledge use is related to the use of knowledge as a ammunition. Conceptual and instrumental use of knowledge refers to the use

Table 2: Outcome of knowledge use

Outcome	**Description**	**Examples of measures**	**Strategy for data collection**
Process or organization level	Impact on process or organization of using or applying the knowledge.	Effectiveness and efficiency of the process (e.g. testing).	Project outcomes/artifacts, document analysis, survey and/or interviews
Practitioner level	Impact on practitioners of using or applying the knowledge.	Practitioner satisfaction, confidence or knowledge.	Survey and/or interviews
Customer level	Impact on customers of using or applying the knowledge.	Customer satisfaction	Survey and/or interviews

of knowledge to make changes in practice. The key difference between conceptual and instrumental use is that in conceptual use, knowledge can be used to make an informed decision but not change practice unlike instrumental use where the process is changed and standards are defined. Monitoring knowledge use is important to understand if the desired level of knowledge use is attained. If the knowledge use is not adequate then, the barriers and facilitators of knowledge use need to reassessed to understand the intentions of the practitioners' to use the knowledge.

8. **Evaluate outcomes:** The evaluation is based on the impact of the knowledge (outcome evaluation) and the evaluation of the KT intervention used to make the knowledge available to all the potential users of the knowledge.

    - **Evaluation of knowledge use outcomes:** The outcome of knowledge use could be on the process/organization level, practitioner lever or customer level. These outcomes are presented in Table 2.

    - **Evaluation of KT intervention:** The effectiveness of the KT intervention selected in Step 6 could be conducted using quantitative and qualitative evaluation methods. Quantitative measure can be such as randomized trials or a interrupted time series method that indicate the effectiveness of the KT intervention for example, the testing process improved after all the involved stakeholders participated in the training workshop. At the same time, qualitative information such as feedback from workshop participants is important in evaluating the KT intervention. For example, the participants might have different level of knowledge, experienced participants might not benefit in the same way as participants with less experience. Such cost-benefit analysis can be done through qualitative evaluation of the KT intervention. Another important aspect of KT intervention evaluation is determining the extent to which practitioners were exposed to the intervention [26].

9. **Sustain use:** Sustainability of knowledge is continuation of knowledge use after the initial adoption. In order to sustain knowledge use, sustainability-oriented action plans need to be developed. There should be consensus on the implementation need and benefits of the knowledge use. Once, the knowledge is used in an on-going implementation, the long-term implementation of the knowledge use needs to be evaluated. For example, can the changes be implemented organization wide? What support is required from the management for continued implementation? It is also important to respond to the shift in context while spreading the knowledge implementation. Along with ensuring sustained use of knowledge, it is also important to allow integration of new ideas that emerge from knowledge use.

4. KT framework evaluation

Case description - The KT framework was used to translate evidence from research studies related to three areas: when to begin automation, what are the testability requirements and what is waste in test automation? In particular, the unit of analysis were the *criteria* for when to automate, testability *requirements* and *factors* associated with waste. From here onwards we refer the criteria, requirements and factors as *variables*. The outcome of the knowledge translation with respect to the three variables is depicted in Figures 6, 7 and 8 and elaborated in the steps below.

Entry point - Need for knowledge implementation: The need for improvement in practitioners' opinion and knowledge related to GUI test automation was identified together with the local champion in a expert panel discussion. The expert panel consisted of a test expert (local champion), a professor, a PhD student and a master student. The test expert was the local champion in the case organization who was interested to know the perceptions of the team members as well as about the evidence in terms of the research outcomes. The main purpose of the discussion was to delimit the scope.

The steps followed in the KT cycle were as follows:

1. **Identify review and select knowledge (Likelihood):** A literature review was conducted to identify relevant knowledge from research studies. Overall 19 studies related to the test automation areas were found. We extracted the frequency and qualitative information related to the variables from the identified studies. The frequencies in the form of percentages of the research studies that mention any one of the areas were calculated. For example, if a criterion for when to automate was mentioned in 5 out of 19 identified studies then, the likelihood was calculated as 26 %. The likelihood is indicated by dashed lines (- - - -) in Figures 6, 7 and 8. In addition, qualitative information about each variable was extracted. Qualitative information includes the name, description and references of the papers that mention the criterion, testability requirements and factors associated with waste.

   Observations on the Likelihood - The likelihood calculation might be based on the percentage of papers that discuss a particular variable. The papers might focus on a particular subset of criteria and not all possible criteria. Therefore, 10% likelihood does not necessarily mean that only 10% of the papers found the criterion. It merely indicates the research focus which might indicate the significance of the criteria but not necessarily.

2. **Elicit prior (Prior probabilities):**

   - Selecting individuals: The practitioners related to the testing activity were selected. The roles investigated were test lead, test architect, developer, tester, design lead and product owner. In this particular case, all roles involved in the project were selected to understand all different perceptions. In total, 12 interviews were conducted.

   - Elicit opinions: The state-of-practice in terms of prior probabilities are elicited through interviews from the above mentioned stakeholders. We asked the practitioners about their testing process. In particular, the following questions were asked: what criteria do you use to decide when to automate? What testability requirements do you consider? What are the factors that you associate with waste? The information on the research outcomes (likelihood) was not provided to the interviewees while eliciting prior probabilities. Similar to the likelihood calculations, the percentage of the interviewees mentioning the factor was calculated. The prior probability is indicated by dotted lines (....) in Figures 6, 7 and 8.

   Observations on the prior probability: Not all interviewees mentioned the same variables. For example, only F2 in Figure 7 was mentioned by all interviewees. In most cases only a subset of the interviewees identified the factors. Out of the total number of possible responses (12 interviewees x 10 factors = 120) obtained from interviewees only 42% of the interviewees' responses accounted for the identified criteria in Figure 7 for example. This indicates that all stakeholders are not equally informed or are not able to articulate or communicate the same variables when they are asked. This further implies that when decisions are made or processes are defined, some of the variables might not be considered or communicated.

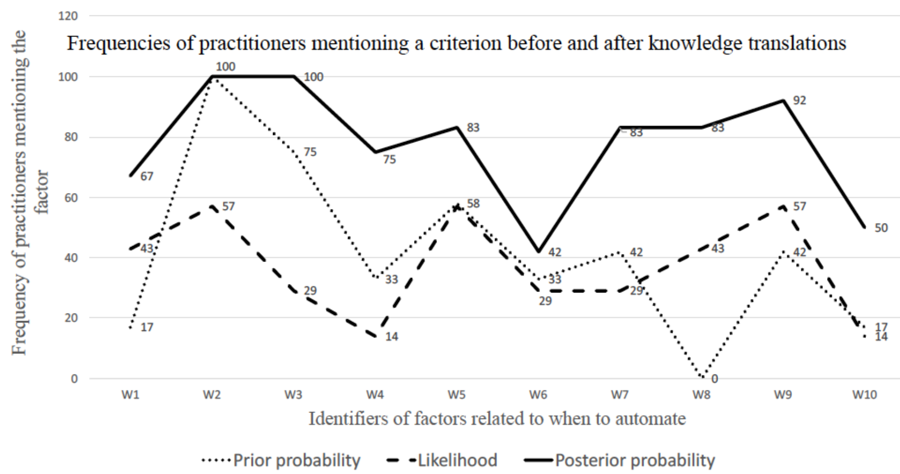

Figure 6: Criteria for when to automate

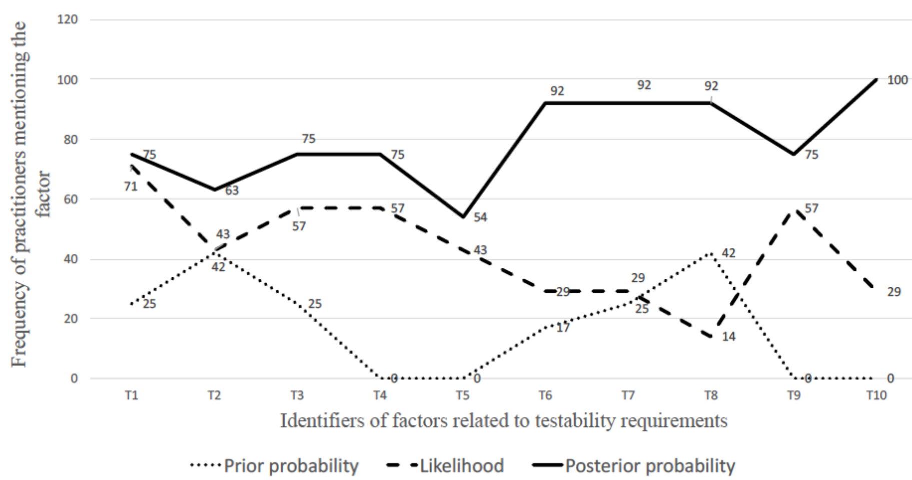

Figure 7: Testability criteria

3. **Assess the gap**: The local champion wanted to know the perceptions and also about the evidence in research studies. In addition, the desired state was that all team members shared the same knowledge.
4. **Align external knowledge to local context**: The knowledge iden-

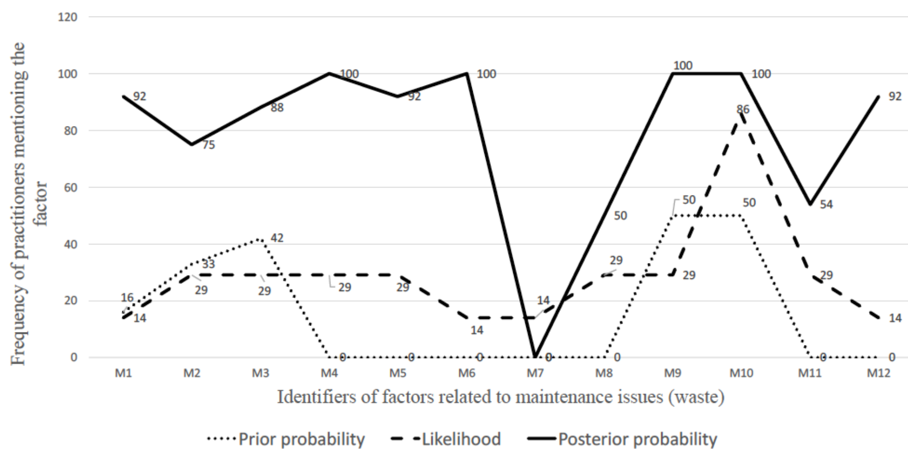

Figure 8: Factors associated with waste

tified, reviewed and selected in Step 1 is presented to the interviewees and their opinions (posterior probabilities) were elicited again in the light of knowledge that was presented to them. The posterior probability is indicated by solid lines (—) in Figures 6, 7 and 8.

Observations on the posterior probability:

- Posterior probability is not always 100%, though the results show that the respondents became more consensual, it still shows that complete consensus is not achieved. In some cases, the likelihood is not applicable or aligned to the context hence, there is no difference between posterior probability and prior probability.
- The average posterior probability of the variables that were mentioned by a subset of the interviewees and also mentioned in the literature was 81% whereas the average prior probability was only 39%. Which indicates that after the knowledge translation the interviewees became more consensual and more informed by having the variables available from literature.
- Even when the interviewees did not mention the variables in their prior probability, the final probability increased drastically. The av-

erage posterior probability of such factors was 73% even when the likelihood percentage was rather low i.e. 26% on an average. If the research result is applicable to the context then the interviewees revised their opinions i.e. posterior probability increased however, the research that is applicable to a specific context might not be heavily researched. Therefore, it is not surprising that there was a significant increase in posterior probability in spite of the likelihood being low.

5. **Assess barriers and facilitators**: The possible barrier anticipated with the one-on-one interviewees was that the interviewees might tend to agree more to avoid discussions or due to time constraints. On the other hand, the interviewees might not agree or not know about certain aspects that are related to other roles or tasks. In order words, the interviewees' opinions and knowledge might not be aligned. The main facilitator was the local champion who identified the need for improvement and wanted to change and improve the overall understanding of the practitioners.

6. **Select, tailor and implement KT interventions**: Due to the identified barriers and the need to have a joint discussion, we decided to conduct a workshop with the interviewees. We decided to tailor the workshop by only using aggregated results and not asking individual questions on why they did not mention certain aspects. Rather than discussing what they did not mention in the interview we discuss what other companies and researchers think and if they agree or disagree to others. This format of the workshop encourages discussion rather than focusing on individual responses. In addition, the intention of conducting a workshop was to facilitate discussions among workshop participants that had different roles in order to understand the big picture of the end-to-end test automation process.

7. **Monitor use**: Since, the identified need for improvement was to improve the practitioners' knowledge and opinions, the knowledge use was on the conceptual level. Therefore, in the workshop we collected informa-

tion about the levels of knowledge and attitudes before and after being informed about the research results.

We asked about the variables that were not mentioned by all the interviewees. Based on the response of the workshop participants we conclude that the practitioners' opinions before knowledge translation are driven by a number of factors: the tasks practitioners are involved in at the time of the interview, their role and the background such as education and work experience. If practitioners miss out on mentioning an aspect, then it might be due to the fact that it is not a priority for the practitioner working on a particular task at that time of the interview. For example, as a tester, the task was to automate everything, so when asked about criteria for when to automate, the tester is likely to mention the criteria that are relevant to his/her task. Criterion such as feasibility assessment of test automation might be a criterion that is considered by the architect and most likely the tester will not mention it right away.

In addition, the question regarding the intention to change based on the knowledge from research results were asked. The workshop participants were already following most of the practices mentioned in the literature however, we found some new insights which were not considered by the practitioners prior to the KT process. However, the practitioners did not perceive it to have much impact. Although, this is a reflection on the results from research studies and not so much on the KT process itself.

8. **Evaluate outcomes**:

   - Evaluating knowledge use outcomes: The need for improvement was on practitioner level, therefore, we collected the practitioner's satisfaction, confidence and knowledge in the workshop. The evaluation of the KT outcome is based on the time, usefulness and value of KT as perceived by the workshop participants.

     Time: Practitioners hardly have time to find answers in research studies. Therefore, it was appreciated that the research results were summarized to them and it was considered to be a time efficient process. In addition, scientific results were valued more than the external information practitioners access (usually technical blogs).

Usefulness: The KT process helped in aligning and streamlining the opinions of the workshop participants. They appreciated the external input and the knowledge as sometimes they can get too influenced by their own tasks and environment. They also identified the most use of KT process in supporting decision-making. Particularly when they do not have all the information needed to evaluate all the alternatives.

Value: The workshop participants expressed an increase in confidence as they got a confirmation that they are following good practices and that they are not missing any crucial knowledge from the research.

- Evaluation of KT intervention: The interventions used in the KT process were interviewees and a workshop. The combination of interviews and workshop was appreciated. The interviews allowed the practitioners to express their individual opinions without being influenced by others. At the same time, after expressing their opinions, it was good to discuss with the other practitioners and reflect more on it. In addition to the conversation between the researchers and practitioners during the workshop, the practitioners asked questions to other practitioners and discussed among themselves as well. The practitioners did more reflecting and discussed the aspects that they did not mention in the interview.

5. Discussion

In this section we discuss the adaptation to the KT framework based on the evaluation. In addition, we compare the KT evaluation result with the related work.

Updates in the KT framework based on the challenges in the pilot study:

The difference in terminology: The terminology used in the research studies was used straightaway, therefore, it created problem in communicating the knowledge and created problems for practitioners in interpreting the results. Hence, after the information is retrieved from the research studies in Step 1, it is important to discuss the results with the local champion. The results summarized in the terminology that is used in the company will be effective in aligning it to the local context in Step 4.

Allowing time between collecting prior and posterior opinions: After collecting the prior opinions, the practitioners wanted to get the research results beforehand so that they can try to better understand the research results and discuss more concretely. The practitioners did not get enough time to think and discuss the research results as they were presented in the same interview. The practitioners were asked to give their opinion immediately after knowing the research results. Getting prior access to the research results will give practitioners time to think and reflect on it. In addition, the results tailored to the terminology that practitioners are used to, will make it clearer. We agree that allowing time between collecting prior and posterior probability will help in capturing true opinions and reflections. Although, booking two different interviews might be a challenge. In particular, it might be difficult if the benefits of the participation are not convincing or clear to the practitioners.

Participating researchers: In order to facilitate good discussions, the researchers participating in the knowledge translation process should be well informed about the topic. Thus, it may not be suitable to use students, unless being PhD students focusing on the area being translated to industry.

Discussion on the evaluation results in related to the related work: The practitioners involved in the evaluation value results from research studies. This is consistent with the study conducted in Microsoft where practitioners were positive towards results from software research studies [16].

The evaluation results indicate that the practitioners have different opinions and knowledge. The practitioners might not always be aware of each others' opinions and knowledge. This is evident by the prior probability line (....) in

Figures 6, 7 and 8. This creates misalignment which might be a problem in cases where alignment and shared knowledge is necessary. This is supporting the problem identified in related works [10] and [3] that suggest replying solely on practitioners' opinion and knowledge might be error-prone.

In the previous study, Devanbu et al. mention that practitioners are heavily influenced by their prior beliefs which impacts their response to new evidence [3]. Although, through the several interactions in the KT interviews and workshops we noticed that in the posterior probability a broader view is often taken than when discussing the prior probabilities. Therefore, the practitioners were not biased and heavily influenced by their prior beliefs. Another finding in the previous study was on the relation between level of agreement with the strength of evidence. They conclude that "the level of agreement didnt always correspond very well with the strength of evidence in regards to the claim [3]." In the previous study [3], the practitioners were asked to respond to the new evidence through a survey, which might explain why practitioners were more influenced by personal opinions than evidence. However, in our evaluation of the KT process we found that the practitioners did considerably revise their opinions although, only when the results from research studies are valid to their context.

6. Conclusions

KT in software engineering research is an important activity for application of research results in practice. This paper outlines the steps in the KT framework and the evaluation results indicate that KT can be valuable in addressing the practice-research gap.

In conclusion, KT framework presented and evaluated in this paper is:

- Systematic,
- Repeatable,
- Reflective of the needs and mores of the practitioners,
- Iterative,

- Collaborative and consensus-driven.

Beecham et al. [30] discuss the practice-research paradox where practitioners do not regularly look at academic literature even though they perceive it to be valuable. We discuss the results of the KT evaluation in relation to the three reasons identified by Beecham et al. [30] for the practice-research paradox: accessibility, credibility and relevance.

The KT framework starts by identifying and summarizing the research results (Step 1) in order to make it accessible to the practitioners. These results are presented to the practitioners by the researchers in one of the step in KT framework (Step 4) therefore making it accessible. It incorporates expert opinions of the researchers as well as elicits practitioners' knowledge and opinion (Step 2) thereby making the results more credible for the practitioners who prefer advice from people with "skin in the game" [30]. In addition to making the research results accessible, the KT framework also facilities aligning the external knowledge from research studies to local context, thereby making the research relevant to the context.

7. References

[1] B. A. Kitchenham, D. Budgen, P. Brereton, Evidence-Based Software Engineering and Systematic Reviews, Vol. 4, CRC Press, 2015.

[2] D. Budgen, B. Kitchenham, P. Brereton, The case for knowledge translation, in: Empirical Software Engineering and Measurement, 2013 ACM/IEEE International Symposium on, IEEE, 2013, pp. 263–266.

[3] P. Devanbu, T. Zimmermann, C. Bird, Belief & evidence in empirical software engineering, in: Proceedings of the 38th international conference on software engineering, ACM, 2016, pp. 108–119.

[4] T. Greenhalgh, S. Wieringa, Is it time to drop the knowledge translation-metaphor? a critical literature review, Journal of the Royal Society of Medicine 104 (12) (2011) 501–509.

[5] B. Cartaxo, G. Pinto, E. Vieira, S. Soares, Evidence briefings: Towards a medium to transfer knowledge from systematic reviews to practitioners, in: Proceedings of the 10th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement, ACM, 2016, p. 57.

[6] D. Badampudi, C. Wohlin, Bayesian synthesis for knowledge translation in software engineering: Method and illustration, in: 42th Euromicro Conference on Software Engineering and Advanced Applications (SEAA), IEEE, 2016, pp. 148–156.

[7] I. D. Graham, J. Logan, M. B. Harrison, S. E. Straus, J. Tetroe, W. Caswell, N. Robinson, Lost in knowledge translation: time for a map?, Journal of continuing education in the health professions 26 (1) (2006) 13–24.

[8] C. Ayala, Ø. Hauge, R. Conradi, X. Franch, J. Li, Selection of third party software in off-the-shelf-based software developmentan interview study with industrial practitioners, Journal of Systems and Software 84 (4) (2011) 620–637.

[9] D. Budgen, S. Drummond, P. Brereton, N. Holland, What scope is there for adopting evidence-informed teaching in se?, in: Proceedings of the 34th International Conference on Software Engineering, IEEE Press, 2012, pp. 1205–1214.

[10] A. Rainer, T. Hall, N. Baddoo, Persuading developers to" buy into" software process improvement: a local opinion and empirical evidence, in: Empirical Software Engineering, 2003. ISESE 2003. Proceedings. 2003 International Symposium on, IEEE, 2003, pp. 326–335.

[11] V. R. Basili, G. Caldiera, Improve software quality by reusing knowledge and experience, MIT Sloan Management Review 37 (1) (1995) 55.

[12] V. R. Basili, G. Caldiera, H. D. Rombach, Experience factory, Encyclopedia of software engineering.

[13] V. Garousi, M. Felderer, Living in two different worlds: A comparison of industry and academic focus areas in software testing, IEEE Software.

[14] L. J. Osterweil, C. Ghezzi, J. Kramer, A. L. Wolf, Determining the impact of software engineering research on practice, Computer 41 (3).

[15] J. Segal, The nature of evidence in empirical software engineering, in: Software Technology and Engineering Practice, 2003. Eleventh Annual International Workshop on, IEEE, 2003, pp. 40–47.

[16] D. Lo, N. Nagappan, T. Zimmermann, How practitioners perceive the relevance of software engineering research, in: Proceedings of the 2015 10th Joint Meeting on Foundations of Software Engineering, ACM, 2015, pp. 415–425.

[17] J. C. Carver, O. Dieste, N. A. Kraft, D. Lo, T. Zimmermann, How practitioners perceive the relevance of esem research, in: Proceedings of the 10th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement, ACM, 2016, p. 56.

[18] S. L. Pfleeger, Understanding and improving technology transfer in software engineering, Journal of Systems and Software 47 (2-3) (1999) 111–124.

[19] T. Gorschek, P. Garre, S. Larsson, C. Wohlin, A model for technology transfer in practice, IEEE software 23 (6) (2006) 88–95.

[20] T. Mikkonen, C. Lassenius, T. Männistö, M. Oivo, J. Järvinen, Continuous and collaborative technology transfer: Software engineering research with real-time industry impact, Information and Software Technology.

[21] C. Pope, N. Mays, J. Popay, Synthesising qualitative and quantitative health evidence: A guide to methods: A guide to methods (2007).

[22] D. J. Spiegelhalter, K. R. Abrams, J. P. Myles, Bayesian approaches to clinical trials and health-care evaluation, Vol. 13, John Wiley & Sons, 2004.

[23] J. L. Crandell, C. I. Voils, M. Sandelowski, Bayesian approaches to the synthesis of qualitative and quantitative research findings, Synthesizing Qualitative Research: Choosing the Right Approach (2012) 137–159.

[24] K. A. Roberts, M. Dixon-Woods, R. Fitzpatrick, K. R. Abrams, D. R. Jones, Factors affecting uptake of childhood immunisation: a bayesian synthesis of qualitative and quantitative evidence, The Lancet 360 (9345) (2002) 1596–1599.

[25] C. Voils, V. Hasselblad, J. Crandell, Y. Chang, E. Lee, M. Sandelowski, A Bayesian method for the synthesis of evidence from qualitative and quantitative reports: the example of antiretroviral medication adherence, Journal of health services research & policy 14 (4) (2009) 226–233.

[26] S. Straus, J. Tetroe, I. D. Graham, Knowledge translation in health care: moving from evidence to practice, John Wiley & Sons, 2011.

[27] M. Ivarsson, T. Gorschek, A method for evaluating rigor and industrial relevance of technology evaluations, Empirical Software Engineering 16 (3) (2011) 365–395.

[28] T. Dybå, T. Dingsøyr, Strength of evidence in systematic reviews in software engineering, in: Proceedings of the Second ACM-IEEE international symposium on Empirical software engineering and measurement, ACM, 2008, pp. 178–187.

[29] C. Wohlin, An evidence profile for software engineering research and practice, in: Perspectives on the Future of Software Engineering, Springer, 2013, pp. 145–157.

[30] S. Beecham, P. O'Leary, S. Baker, I. Richardson, J. Noll, Making software engineering research relevant, Computer 47 (4) (2014) 80–83.