Empirical Software Engineering Experts on the Use of Students and Professionals in Experiments

Davide Falessi[1*], Natalia Juristo[2], Claes Wohlin[3], Burak Turhan[4], Jürgen Münch[5, 6], Andreas Jedlitschka[7], Markku Oivo[8]

[1]*California Polytechnic State University, CA, USA*

[2]*Universidad Politécnica de Madrid, Spain*

[3]*Blekinge Institute of Technology, Sweden*

[4]*Brunel University London, UK*

[5] *University of Helsinki, Finland*

[6]*Reutlingen University, Germany*

[7]*Fraunhofer Institute for Experimental Software Engineering, Germany*

[8]*University of Oulu, Finland*

Email: dfalessi@calpoly.edu, natalia@fi.upm.es, claes.wohlin@bth.se, Burak.Turhan@brunel.ac.uk, Juergen.Muench@reutlingen-university.de, Andreas.Jedlitschka@iese.fraunhofer.de, Markku.Oivo@oulu.fi.

* = *corresponding author*.

Abstract.

[Context] Controlled experiments are an important empirical method to generate and validate theories. Many software engineering experiments are conducted with students. It is often claimed that the use of students as participants in experiments comes at the cost of low external validity while using professionals does not. [Objective] We believe a deeper understanding is needed on the external validity of software engineering experiments conducted with students or with professionals. We aim to gain insight about the pros and cons of using students and professionals in experiments. [Method] We performed an unconventional, focus group approach and a follow-up survey. First, during a session at ISERN 2014, 65 empirical researchers, including the seven authors, argued and discussed the use of students in experiments with an open mind. Afterwards, we revisited the topic and elicited experts' opinions to foster discussions. Then we derived 14 statements and asked the ISERN attendees excluding the authors, to provide their level of agreement with the statements. Finally, we analyzed the researchers' opinions and used the findings to further discuss the statements. [Results] Our survey results showed that, in general, the respondents disagreed with us about the drawbacks of professionals. We, on the contrary, strongly believe that no population (students, professionals, or others) can be deemed better than another in absolute terms. [Conclusion] Using students as participants remains a valid simplification of reality needed in laboratory contexts. It is an effective way to advance software engineering theories and technologies but, like any other aspect of study settings, should be carefully considered during the design, execution, interpretation, and reporting of an experiment. The key is

to understand which developer population portion is being represented by the participants in an experiment. Thus, a proposal for describing experimental participants is put forward.

Keywords: Experimentation, threats to validity, generalization, subjects of experiments, participants in experiments.

1 Introduction

Controlled experiments are an important empirical method to generate and validate software engineering theories. Many experiments in software engineering are conducted with students. This happens for several reasons, including easy accessibility and low cost compared to professionals. However, the use of students in software engineering experiments is often criticized to come at the cost of low external validity because the technology under study is to be used by professionals rather than students. The type of participants is often a concern when assessing the quality of an experiment. The use of students is often seen as a weakness, whereas the use of professionals is seen as a strength. We have wondered if such beliefs are correct.

To advance the use of experiments to generate knowledge about software engineering, it is essential that we gain insights regarding the generalization of results from participants' samples to populations for example to answer questions such as to what extent the results obtained from an experiment conducted with students hold for professional software engineers. We believe that answering the question to what extent the results obtained from an experiment conducted with professionals hold for all software engineers is equally relevant. We lack knowledge when it comes to understanding: 1) under which circumstances students might behave similarly to professionals and 2) which students might behave similarly to which professionals. Are there situations where the behavior is similar? On what does it depend? Does education affect this issue? Some universities stress project work and try to mimic industry projects [1] [2], and hence students from this type of environment could be better proxies for novice industrial software engineers than students not exposed to such training. Is it possible for students in certain situations to understand the industrial context [3], and hence behave similarly to professional software engineers? Are there situations where students might even outperform professionals [4]?

The objective of this paper is to report the opinions of a group of experts on the use of students and professionals as participants in software engineering experiments. Our aim is to reflect on the pros and cons of using students and professionals in software engineering experiments. Our discussion aims at supporting both researchers during the generalization of experiment results and reviewers during the evaluation of experiments. The scope of this paper is human-based experimentation. Other experiments, like those conducted in mining repositories research, are out of scope even if they are still very relevant for advancing the software engineering body of knowledge.

We followed an unconventional focus group approach consisting of two steps. First, we had a session at ISERN[1] 2014 where 65 empirical researchers, including the seven authors, argued on and discussed the challenges with students as subjects in software engineering experiments. Second, we applied thematic synthesis for a more in-depth rethinking and elicitation of experts' opinions. The authors consolidated and evolved the main findings into a set of 14 statements. After the focus group, we performed a survey in the fall of 2016 where the ISERN session attendees, excluding the authors, were asked to provide their personal level of agreement with these statements. We used the results from the survey to refine the statements.

[1] http://isern.iese.de/

The overall theme of the paper is that not considering students as representative of professional developers seems to be an oversimplification. Considering the use of professionals in experiments as a panacea is another oversimplification. According to our survey results, the experts agreed with most of the statements but highly disagreed about the drawbacks of professionals. It is worth noting that in order not to bias the experts, we provided the respondents only with the statements and not with their rationale. Therefore, some statements may have been perceived slightly differently than intended, given that the invited experts did not have the full background, for example the rationale for each statement.

In order to avoid misunderstandings, in the remainder of this paper the term "**experts**" is used to identify the ISERN members having long-time experience in performing experiments. The term "**authors**" is used to identify the seven experts authoring this paper. Although the authors are also experts, when we refer to experts here we do not include the authors.

The remainder of this paper is structured as follows. Section 2 reports the related work and Section 3 describes the research method. Section 4 presents our statements. Section 5 reports our survey results, and an analysis of the findings is presented in Section 6. In Section 7, a solution for better characterizing experimental subjects by providing a characterization scheme of subject experience is proposed. Section 8 compares and contrasts the results with related work. The threats to validity are discussed in Section 9. Section 10 concludes the paper with some recommendations to both authors and reviewers.

2 Related work

One simplification of reality aimed at allowing it to be studied under laboratory conditions (in a controlled and systematic manner) is the use of participants who are not professional software engineers. In software engineering, in particular, we commonly use students rather than professionals. The use of proxies in experiments is also common in other experimental disciplines, although our case might have a special twist. In software engineering, we need to deal with the differences in competencies between subjects developing software in the real world and subjects developing software as an academic exercise, which is a specific validity issue pertaining to our discipline.

Some initial evidence is available about the similarities of results obtained in experiments run with students and professionals. Understanding the differences between such results should help us to understand the level of generalization that can be expected regarding results obtained from experiments with students. In 1995, Porter et al. published a replicated experiment conducted with students [5]. Three years later, they published a further replication with professionals [6]. In their conclusion covering the results from both experiments, the authors state that "the fault detection rate when using scenarios was superior to that obtained with ad-hoc or checklist methods—an improvement from 21% to 38% in the professional population and from 35% to 51% in the student population." They argue that there is basically no difference between the effects identified, arriving at the conclusion that "students should not be automatically discounted" and that for economic reasons, experiments with students can contribute "to validate fundamental research recommendations". Although the effect was similar, the absolute detection rates differ. From this work, it cannot be concluded that students behave identically to professionals with respect to the defect detection rate, but rather that they behave similarly regarding the effect. So when it comes to performance improvement as a result of using checklists versus doing ad-hoc inspections, experiments with students seem to be valid and the results (effect) could be generalized, at least to the type of professionals they used [6].

Höst et al. [7] investigated the appropriateness of last-year students as subjects by comparing their judgments of factors impacting lead time to those of professionals. The authors only found minor differences with respect to the conception and no differences with respect to correctness. Consequently, they conclude that last-year students, if well trained, can be considered as appropriate subjects in this type of experiments.

Svahnberg et al. [3] investigated "whether students have a good understanding of the situation in industry" in the context of requirements selection for release planning. From their results, the authors conclude that "it may be possible to influence students to provide answers that are in line with industrial practice."

Berander [8] compared results from an experiment with classroom students to results from further studies (with students having project experience and an industrial case study) in the area of requirements prioritization. He found that a high level of commitment, rather than of experience, makes the students more comparable to professionals.

Salman et al. [9] measured the code quality of several tasks implemented by professionals and students, and checked differences between the two groups. Their results show no major differences between the two groups. Specifically, they report that professionals produce larger, yet less complex, methods when they use their traditional development approach, whereas both subject groups perform similarly when applying a new approach for the first time.

However, there is also some evidence that students perform differently. For example, in the area of production scheduling decision making, Remus [10] found that "undergraduate students made more costly decisions, used less effective decision heuristics, and were more erratic than the managers." In the area of software testing, Basili et al. [4] reported higher levels of effectiveness among undergraduate and graduate students than among professionals.

In the area of inspection techniques, McMeekin et al. [11] found that professional developers performed significantly better than student participants for checklist-based, usage-based, and use case reading. Taking into account further qualitative information, the authors claim that professionals perform differently from students and emphasize the importance of studies with professionals.

Ciolkowski [12] quantitatively aggregated the results from experiments concerning perspective-based reading. He identified 12 publications reporting 21 runs of experiments, of which 15 used students and six used professionals as subjects. He investigated several influencing factors, among them the kind of subjects used in the experiment. He found that "for experiments with professional developers, perspective-based reading was more effective than checklist-based reading or ad-hoc reading", whereas for experiments with "students, there was no consistent effect." One possible explanation provided is that the professionals actually had the roles used in perspective-based reading, while the students were assigned the roles.

Runeson [13] investigated the performance of undergraduate (freshmen) and graduate students in the context of the personal software process (PSP). He found that undergraduate students needed significantly more time to complete a task than graduate students. Comparing his results to the results from an experiment conducted with professionals by other authors, Runeson stated that "the data is not sufficient to evaluate similarities or differences between industry people and graduate students". Runeson also confirms the results by Wesslén [14], who conducted a study with PhD students and compared the results to those of the original study made by the Software Engineering Institute.

Some studies specifically investigated the difference between students and professionals in performing specific tasks. In the context of UML diagrams, Ricca et al. [15] describe the results of four

experiments with subjects having different experience levels. Their results show that the treatment under consideration (i.e., UML diagrams) only improved the performances of subjects with low experience. Salviulo and Scaniello [16] analyzed how comprehension and maintenance practices change across subjects with different levels of experience.

In conclusion, there are some studies that compare students versus professionals in terms of performance in software engineering experiments, but the picture is far from clear. A clear conclusion cannot be stated because very often, different development tasks as well as different experimental settings are used by students and professionals. So many more comparative studies are needed before we obtain an answer on whether students are good proxies of professionals in software engineering experiments. Moreover, we should always differentiate between relative comparisons and absolute numbers as exemplified above with the studies by Porter et al. and their conclusions based on the two studies [6].

A straightforward way to compare the performance of students and professionals is to conduct experiments with both types of subjects and include the subject type as a factor in order to study the interaction of the subject type with the treatment and its impact on the studied effect. There exist some experiments in the literature that have used both students and professionals. However, those experiments did not mainly aim at investigating the difference between groups of subjects. We encourage researchers who have both students and professionals in their experiments (if the number of participants allows it) to include the comparison of the subjects' type performance in their research and add it to the keywords or abstract to allow their contribution regarding this issue to be easily identified.

Our approach is different from those used in the related work and aims to complement the studies referred to above. We investigate the topic qualitatively to learn the views of researchers with experience in conducting experiments with both types of subjects.

3 Research method

Figure 1 summarizes our research method, which consists of a focus group followed by a survey. We planned the focus group in two sessions. The first session took place during the annual meeting of ISERN in 2014 and the participants were ISERN attendees (a total of 65 participants including all the authors of this paper). The second session was remote and asynchronous, and conducted among the authors of this paper. The analysis of materials collected resulted in themes and provided the input to the second session. The second session resulted in 14 statements, which were obtained after a thorough discussion among the authors including steps to consolidate and evolve the main findings. After the focus group sessions, we conducted a survey with ISERN 2014 attendees to evaluate the outcomes of the focus group sessions, i.e., the 14 statements. Incorporating the feedback from the survey, we then fine-tuned the statements into their final form. The following subsections provide details about the focus group and the survey.

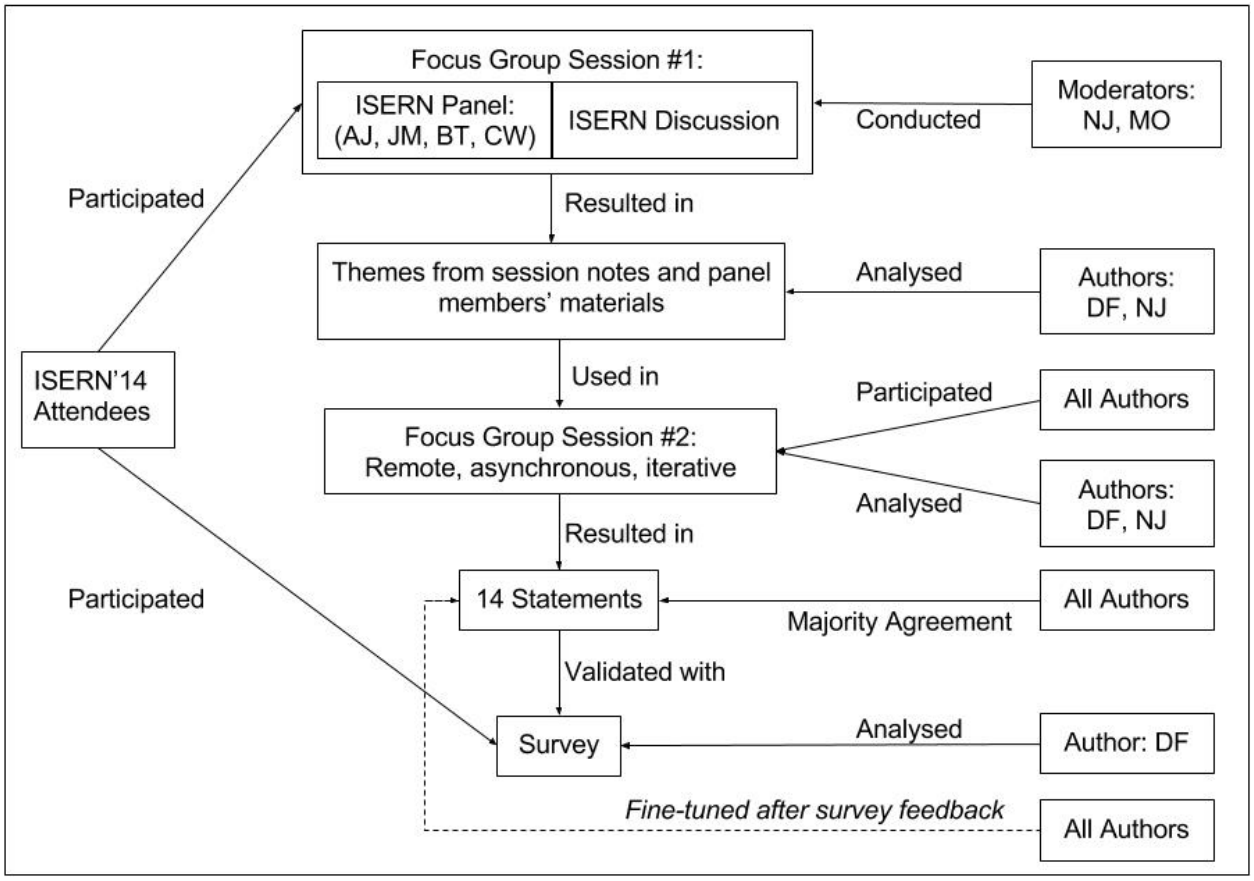

Figure 1 Research Methodology. Initials represent the authors of this paper.

3.1 Focus Group

The first step in the research methodology was the use of a focus group, which is an established methodology commonly utilized in social sciences [17]. In software engineering, researchers have used focus groups to investigate various issues based on eliciting expert opinions. Focus groups have, for example, been used to study (de-)motivation factors in software process improvement [18] and the effectiveness of agile methodologies [19], to understand corporate risk management practices and requirements prioritization challenges and usability evaluation [20]. Kontio et al. [20] refer to focus groups as a fast and cost-effective method for gaining qualitative insights. Apart from these advantages, the reason for our choice was that focus groups are suitable for fulfilling our research purpose, as they:

- are exploratory in nature [21], yet the "... real strength of focus groups is not simply in exploring what people have to say, but in providing insights into the sources of complex behaviors and motivations [22]",
- build upon interaction among group participants and allow "discovery of new insights" and "aided recall" through group discussions [23],
- are inquisitive and a first-order data collection method that provides general understanding and opinions, though subjective, on a particular issue of interest [24], and
- can be used as a "self-contained" method or in combination with other methods, i.e., for further data collection or analysis [17].

Kontio et al. [23] provide guidelines on the steps for conducting focus group studies in a software engineering context. Below, we follow their guidelines in reporting our approach.

Step 1: Defining the research problem. The objective is to discuss the use of students and professionals as subjects in software engineering experiments. We intend to better understand the pros and cons of using one or the other. Furthermore, we would like to obtain further insights about whether one of the two types is preferable over the other, and reflect on the opinions related to the use of students and professionals as subjects in software engineering experiments. In particular, we want to understand if experiments with students are somehow second-class and the use of professionals should always be preferred for software engineering experiments. Finally, the objective is to propose a characterization scheme for subjects in software engineering experiments that takes us away from the simplistic view of dividing participants in experiments into professionals and students. These objectives are summarized in the following three research questions:

RQ1: What are the pros and cons of using professionals and students as subjects in software engineering experiments?

RQ2: As a community, what do we agree and disagree on with respect to subjects in software engineering experiments?

RQ3: Is it possible to characterize subjects using a different classification than professionals or students?

Step 2: Selecting the participants. We targeted the participants for the first session of our focus group with convenience sampling using two basic criteria: i) experience in conducting software engineering experiments, and ii) experience in conducting experiments with students as well as experiments in industry with professionals, as the nature of our inquiry requires. In order to reach out to suitable candidates, we approached scholars who are members of ISERN[1] by organizing a plenary session during its annual meeting (for details, see next step). For the second session of the focus group, the people listed as the authors of this paper furthermore agreed, after the ISERN session, to participate in this research and to contribute to the paper. All authors have extensive experience in conducting experiments with professionals and students, e.g., the number of documented experience of the authors, in terms of peer-reviewed publications, varies from six to 15 experiments.

Step 3: Conducting focus group sessions. Once potential participants had been identified and approached, we were ready to conduct the first session of the focus group. The first session was embedded within a plenary session organized by Natalia Juristo and Markku Oivo during the 2014 annual meeting of ISERN[2]. The plenary session was entitled "Students vs. practitioners". This session in turn had two parts. First part was a panel with invited speakers who are known to have experience in conducting experiments with students and professionals. Andreas Jedlitschka, Jürgen Münch, Burak Turhan and Claes Wohlin were invited as panelists to provide their view on the matter. The second part was the open discussion with all ISERN members who have attended the plenary session. While the first part provided the motivation, and introduced discussion points and a summary of expert opinions, the second part provided an interactive discussion forum where the audience had the chance to actively participate and shape the discussions. Hence, when considered collectively, both parts make up the first session of the focus group. The goals of the plenary session were announced as follows:

[2] http://softeng.polito.it/ESEIW2014/ISERN/program.html#Students

- Identify how we can proceed to learn more about the type of experimental subjects in software engineering experiments and the kind of generalization we can expect for each type;
- Identify ISERN members' experience about the topic;
- Identify interested contributors to research on a more profound understanding of the differences between professionals and students.

The first session was moderated by the organizers (NJ and MO) and the goals listed above provided a generic starting point. The panelists were informed about the expectations of the session as a guideline for stating their views and for initiating further discussions with the audience. Specifically, the panelists were provided with the expectations regarding the outline, i.e., "what experience do you have on the topic, what is your position, how to proceed... ". Each panelist was expected to talk about his view on what degree of representativeness can be expected from students as experimental subjects. The panelists were also reminded to first state where their experience comes from and to then present their position.

The objective of the session was to have an open discussion. The moderators started by explaining the motivation for the session (10 minutes), then the panelists presented their views (5 minutes each). This was followed by a discussion among the panelists and with the audience (30 minutes). The moderators took notes of the discussions. The session lasted for 60 minutes in total. After the session, in addition to collecting the organizer's notes, the panelists were asked to provide their materials (written material/presentation slides). The first author of this paper volunteered take part at that stage.

The second session of the focus group took place remotely in an asynchronous way. While remote focus groups are common [25], using asynchronous mode could be considered a threat to the interactive nature of focus groups. However, when reporting their experiences in conducting (computer-mediated, electronic) focus groups, Kontio et al. state that "... the session could have been conducted online as well (i.e. not necessarily being at the same place nor at the same time)" [20]. The interaction among the participants was preserved as the discussions continued to provide ideas and experiences in several rounds. Davide Falessi and Natalia Juristo consolidated the collected information from the first session and drafted the first version of this paper. Interaction continued in the form of comments and contributions to this research paper. While unconventional, our approach also fits in with the notion that focus group interactions should take place until no new information emerges, which in our case translates to no new ideas or comments arising. We reached the final version of this research paper by performing three iterations. These iterations took about 16 months to complete.

Step 4: Analyzing the data. Since we have two focus group sessions, and the latter is iterative, the data analysis was interleaved with the previous step. As mentioned earlier, the analysis of materials collected in the first session provided input to the second session. Similarly, the analysis activities for the second session took place after each iteration. We started with a thematic synthesis by analyzing the organizer's notes of the session and panelists' materials (slide decks and written summaries) from the first session. We recall that during the ISERN session, each author talked about the topics they preferred. It was expected that the authors would focus on the topic impacting them the most or the one most recently or frequently encountered. Most of these topics turned out to be complementary. For example, one author focused on trying to identify which professionals' populations are better represented by students (see Section 4.3), whereas another author focused on the difficulties occurring when using professionals as experimental subjects (see Section 4.2.1). Despite each author focusing on a specific and different topic, it was observed that the authors had strong and specific opinions on several topics. Therefore, after the ISERN session, the authors'

opinions on all topics were collected. Specifically, the first two authors initiated the collection and organized the topics, and then asked the other authors to agree, disagree, or refine all the topics by providing examples of concrete experiences. During this process of refining the authors' opinions, some new opinions and topics arose. For example, one author proposed as a new topic the eternal trade-off between external and internal validity (see Section 4.3). We did not enforce a consensus among the authors. We decided to define the final topics based on agreement of the majority of the authors. This resulted in 14 statements, which we externally evaluated using the survey design described in the next section.

3.2 Survey

Although the authors are experts in performing, analyzing, and reporting controlled experiments, they are not the only experts. Moreover, the authors felt that some statements were more controversial than others. Therefore, we proceeded by designing a survey targeting the ISERN session attendees with the exclusion of the authors. One option could have been to open the survey to the entire software engineering research community, but we were afraid that by doing so, we would deviate from the target population, i.e., experts in conducting experiments. Furthermore, it was considered to be important to target researchers who participated in the ISERN session because this formed the basis for the later discussions. The survey and its invitation letter are reported in Appendix A.

The invitation letter was provided via email and contained a unique, individual link for participating in the survey. In the invitation letter we stressed that the results would be anonymous, that we cared about the invitees' opinion, and that they should not hesitate or be afraid to disagree with us.

The first question of the survey characterizes the experience of the respondent. In fact, despite ISERN being one of the main research communities for empirical software engineering, not all attendees necessarily have extensive experience in performing experiments. Providing an answer to this question was mandatory. The respondents could choose among the following three options: a) Expert: more than 3 published experiments; b) Novice: between 1 and 2 published experiments; or c) Amateur: no published experiment or you do not know what an experiment is.

The main part of the instrument consists of 14 statements resulted from focus groups. The respondents were asked to rate their level of agreement/disagreement with each statement. The following five options were possible: a) Completely agree, b) Partially agree, c) Partially disagree, d) Completely disagree, or e) I do not know or I do not want to answer. The 14 statements were organized as rows and the possible answers were organized as columns. Providing an opinion about each statement was mandatory.

At the end, we asked for the respondent's last name in case the respondent wished to be acknowledged in this paper. Providing this information was optional.

4 Focus Group Results: 14 Statements

This section details the statements we distilled from the ISERN session and the related rationales. We included statements when a majority of the authors agreed on them. The statements are organized into topics. Each of the following subsections presents a specific topic and the rationale behind each statement. The statements are enumerated according to the order of their presentation to facilitate traceability; thus, the order does not represent their level of importance or agreement. The statements are identified using the tag "STn", where "n" is a number between 1 and 14.

4.1 Clarifications

4.1.1 Misleading terminology

Students and professionals are tags broadly used to characterize participants in software engineering experiments. However, some authors think that these terms are not appropriate for characterizing participants in experiments. They provided some examples that illustrate this view.

Professionals can be defined as persons working in a software-intensive company or a company where software is an essential part of their offering. Students can be defined as persons following university lessons. However, it is common to have professionals going back to university or working and studying at the same time. One of the authors reported that during the summer, almost all computer science students in California perform internships in top software companies like Apple, Google, IBM, Intel, or Microsoft. Other authors reported that in Spain or Germany, students in their last year (or during their master studies) are already working while continuing to study. Thus, the term "student" does not preclude the possibility of having industrial experience. In fact, interns in industry or novice professionals can have less industrial experience than many students.

Also, there are borderline cases that are hard to classify. For example, it is hard to decide if practitioners having less than two years of experience should be classified as students or professionals. In other words, the threshold from which we can start considering someone a professional is unclear. Assuming the threshold is as soon as a company hires a student, does this mean that she is different from an experiment viewpoint than the day before?

So the terms professionals and students are misleading for at least three reasons:

1. They overlap, i.e., the same person can be a student and a professional.
2. They are not accurate, i.e., the work experience of a student can be higher than that of a professional just hired.
3. The inclusion criteria for the two types of subjects are fuzzy.

Therefore, our first statement is: **ST1 - Classifying experiment participants using a binary scale (students or professional) is an inappropriate approach.**

Ideally, subject characterization should be based on relevant attributes, and what these are is an open question. How to address the experience of subjects in a less misleading and more fine-grained way will be proposed and described in Section 7.

4.1.2 Threats to validity trade-off

One of the major objectives of the design phase of an experiment is to reflect on potential threats to validity and how to mitigate them. This is enacted, e.g., through systematic selection of the context in which the experiment is to take place, sound operationalization and instrumentation, systematic sampling, and proper application of design principles [26] [27]. Design principles include randomization (to cancel out, e.g., natural variation in human performance), blocking (to systematically eliminate undesired effects because of known/assumed variation factors), and balancing (to achieve, e.g., equal accuracy, power, or confidence interval width for treatment comparisons) [28].

Our perception is that many researchers support the following statement, and hence it was added as a statement to the survey: **ST2 - Experiments with students might exhibit lower external**

validity than experiments with professionals. However, some of the authors raised the issue of treatment conformance and internal validity, which led to the following statement: **ST3 - Conducting experiments with professionals entails a higher treatment conformance threat to validity than experiments with students**. This statement will be further elaborated in Section 4.2.3.

The experimental paradigm requires a trade-off between internal and external validity. It is infeasible to maximize both internal and external validity at the same time. Berg et al. [28] state quite clearly: "The higher the degree of one type of validity, the amount the [sic] other type lessens". But in software engineering there does not seem to be broad acknowledgment of this trade-off. A recent paper [29] attempted to determine if there is consensus in the community as to whether to focus on internal or external validity, and found that there is no consensus. From our perspective, this result makes a lot of sense. The authors agree that a choice cannot be made between one and the other. A minimum of both is required. Therefore, the authors agree that **ST4 - Internal and external validities are of equal importance.**

Even in experiments with professionals, internal validity is required. That is, the design must guarantee a level of control that assures that effects in the dependent variable(s) are caused by the independent variable(s). If this is not shown reliably, high generalizability will not save experiments from being flawed and will not save the results from being unreliable. Similarly, experiments with students need to have some minimum level of external validity. That is, the sample must resemble developers (or whichever population is targeted). For example, an experiment conducted with ordinary people (rather than students of a software development related program) would not achieve the minimum degree of external validity required.

Summarizing, we can conclude that if the objective of a specific study is taken into account, an experiment with students is neither intrinsically wrong nor worse than one with professionals. If an experiment with students exhibits high internal validity, but low (greater than the minimum required) external validity, it can be considered a first crack in the wall of evidence for a piece of knowledge. Therefore, **ST5 - Experiments with students are not of lower relevance than experiments with professionals**. Moreover, **ST6 - Experiments with students are not of less interest than experiments with professionals.**

4.1.3 Convenience sampling

The use of students is not only a simplification for developers required to simulate the real world in the laboratory. They are also a convenient sample (i.e., participants are selected only because they are easy to approach). Convenience sampling constitutes a threat to an experiment's external validity. However, experiments with professionals equally suffer from this type of external validity threat. When experiments are conducted in a company, the participants are also a convenient sample because they are not selected randomly from the population of all developers and very often they are not even a random sample of the developers at the company. Most often professionals participating in an experiment run at a company are even more convenient than a student sample. Consider a situation where a project manager selects the participants for an experiment. The chosen participants might not be people who are heavily involved in projects. In other cases, enrolment is voluntary, so those professionals who are most eager to learn sign up. We can easily find more of these examples. All such cases are also convenient samples and, therefore, the threat of low external validity due to the lack of representativeness of the whole population of developers affects the results, even if the experiments are run with professionals. As in the case of experiments conducted with students, experiments run with convenience sampling of professionals represent a certain sub-type of the developer population. Therefore, the authors agreed that **ST7 -**

Students participating in an experiment are a convenient sample. Moreover, **ST8 - Professionals participating in an experiment are a convenient sample**.

4.1.4 Characterizing subjects at design time

Even using convenience sampling (because very often this is all we have), a careful evaluation of the subject population at design time is a must, so that the limitations on generalization are understood from the beginning and a decision can be made whether to proceed or not.

Too often, researchers design experiments using convenience sampling and only retrospectively start to reason about external validity. Many studies use convenience sampling and then consider representativeness and validity *after* the experiment has been conducted. Instead, we should evaluate from the beginning what such convenient sample would represent and whether it is of interest. Thus, the authors agreed that **ST9 - We should think about population and validity already before conducting the experiment, at the time when we are planning to use convenience sampling.**

We encourage researchers to consider the choice of subjects already during the design of an experiment [27]. We suggest that researchers carefully plan experiments when it comes to the subjects and capture their experience in the best possible way. This should allow for a better understanding of the sample being used and the population the participants belong to. If the population is not an adequate one for the technology being studied, it might be necessary to abort the experiment.

For any experiment, researchers should carefully consider the population in relation to the research questions, and hence for which population the results may be claimed to be valid. If software engineers are our target population, then we have a very broad population and one that is not very specific, which is a benefit on the one hand (we could use everyone with a knowledge of software engineering), but a drawback on the other hand (we would not be able to collect a large enough sample to capture all aspects of software engineering, from information systems to embedded systems, etc.). If testers are our target population, the population gets smaller and more specific. And we could continue making the target population even more specific. When conducting an experiment, we could derive characteristics of the population for each step. For example: What would be the expectations when hiring a tester? Knowledge in xyz, experience in uvw, or certificates from abc? In answering these questions, we could think about the appropriateness of the students participating in the experiment and at the same time about the kind of professionals we could compare them with, that is, the professionals to whom we will (potentially) be able to generalize the results obtained in the experiment with students.

It might be that the closer the experimental task is to reality, the more experience plays a role. If we are experimenting on new things, it makes sense (like we have seen in some of the related work) that students perform even better than professionals because they might be more willing to learn new things and the experience of professionals does not really account for their performance. But if we ask the experiment participants to do some task in a way in which professionals already have experience, it becomes relevant and it might be that students perform worse than professionals. Note that this example discusses only the willingness to learn and experience in the technology, but there might be several other characteristics that merit attention when struggling to understand the representativeness of the sample we got. It is still subject to research what such characteristics are.

These reflections on sub-populations, experimental tasks, and participant characteristics (as measured by experience) should be done at design time to decide which population the sample in

the experiment represents, if any. Note that oversimplifying subject characteristics can lead to wrong conclusions, so caution is important when performing this task.

4.2 On the drawbacks of professionals

Experimentalism is a paradigm encompassing different types of experiments aimed at getting a valid piece of knowledge out of the lab. To obtain experiment-backed knowledge that is valid in the real world, a particular path needs to be followed. This path is similar, yet not identical, to that used in different other experimental disciplines. For example, in medicine the path starts with lab experiments in vitro, followed by experiments in vivo (going from mice to apes), then it goes out of the laboratory with volunteer trials, and the path ends with clinical trials (experiments with real patients).

The experimental path starts in a very controlled environment (i.e., the lab) and ends in a less controlled, but real, environment (i.e., the field). Laboratory experiments aim to have high internal validity even though they have low external validity. Field experiments work the other way around. The authors phrased this idea in different ways. In the words of one author: To get high internal validity, laboratory experiments need to make simplifications of reality; the use of students is one of the many simplifications of reality done in software engineering laboratory experiments (others being toy tasks, short time, class environment, etc.). In another author's words: In order to mature knowledge pieces, we first need to explore whether there is a cause-effect relationship, starting in an environment with maximum control (i.e., the laboratory). In later stages of experimental research, we seek to generalize such preliminary laboratory results, which means going out of the laboratory into the real world (i.e., the field) and accepting that we lose control (or a least that there is less control).

All authors agreed that, as a stage in the experimental path, there is nothing intrinsically wrong with experiments using students [30].

Another author recalled the NASA/SEL approach [31][32], where experiments with students were the first step in the software engineering experimental path, while professionals were the second step. One of the authors described this path as: "Once a technology gets promising results in laboratory experiments, it might be time to advance in the path and call for practitioners" [33].

One author pointed out that for software engineering, there might be alternatives to the classical experimental path: Starting with controlled experiments in the lab might need to be rethought for software engineering because there are other alternative paths that could also work. For example, starting outside the laboratory to understand the complex contexts where real software development is done with the help of a case study, then going back to the controlled environment of the laboratory to simulate the context learned. In any case, experiments run in a very controlled environment (a laboratory using students as proxy of professional developers) play a role in the path to gain software engineering knowledge backed by empirical evidence.

Thus, conducting experiments with professionals is an important step in the experimental path of software engineering. However, professionals come at a cost. Would it be a desirable situation that in software engineering we conduct experiments only with professionals and discard students as participants? Some of the authors highlighted the challenges they encountered when working with professionals in experiments, challenges they did not face when conducting experiments with students.

4.2.1 Hard to get, even harder to improve

Professionals are surely more difficult to get involved in an experiment than students. Most researchers have easier access to students; they interact with them in courses and very often can easily get them involved in experiments either as a practical part of the course or on a voluntary basis. The effort expended for conducting a single experiment with professionals might be the same as conducting multiple experiments with students, which would allow multiple cycles of knowledge discovery, assessment, refinement, etc.

One author commented on the time needed to run an experiment with companies [34]. For an experiment embedded into a research project for which several companies had enrolled, some companies needed just a few months (3-6) while others needed a couple of years for the same experiment. In academia, the very same experiment took only a few months. However, because experiments in academia are often conducted during a course, researchers need to wait one full semester (or two if the course is taught on an annual basis) for the results. Given this reasoning, it is hard to state firmly that experiments in academia or in industry are better or worse from a calendar time perspective. But an issue more threatening than time is flexibility. A laboratory experiment run in academia can be improved annually. Even if we all aim to get a good experiment from the beginning, this might not be the case for the first run, but then we will have a second chance in the next semester (or year). With companies, such flexibility does not exist. There are very few chances to run the same experiment more than once in the same company. If something in the design is faulty and this is discovered after the experiment has been conducted (as so often happens in lab experiments), there will seldom be a second chance to do another field experiment with the same company. This restriction inhibits the iterative nature of experimentation (or makes it really hard). From this perspective, the role of laboratory experiments with students as a preliminary step in the experimental path to mature experiment design and protocol is becoming more important. Therefore, the authors agreed that **ST10 - The use of students supports the improvement of experiment design and protocol better than the use of professionals.**

4.2.2 Small sample size

Even when a researcher succeeds in getting professionals involved in experiments, it most often results in only a small number of participants. One author gave us the specific numbers of professionals participating in an experiment conducted at four different sites: 7, 11, 4, and 9, while the initially committed participants in the runs were: 5, 15, 12, and 9. This means that in one case, only 33% of the committed participants finally showed up for the experiment. In this author's experience, "no shows" happen more often with professionals than with students.

From a statistical perspective, a higher number of subjects allows for more data points, which facilitates the possibility to obtain statistically relevant results.

Unfortunately, in software engineering, publishable results are still those that yield statistically significant results. One author raised the issue that an experiment resulting in the same effect size would have (surprisingly enough) a much greater chance of being published if it used students rather than professionals because the former would allow for a larger sample size.

Regardless of the ability to publish the study, a result lacking statistical significance might discourage researchers from continuing to run replications of such an experiment even if the result was due to a small sample size. Therefore, the authors agreed that **ST11 - Conducting experiments with professionals as a first step should not be encouraged unless high sample sizes are guaranteed.**

4.2.3 Treatment conformance

In a typical human-based software engineering experiment, the subjects are given limited time to work on a specific task. One author said that, in some of his experiments, professionals were less productive than students. He believes that students have the (unconscious) ability to focus on the assigned task, driven by the goal of "reaching the end" and willing to make trade-offs. They are used to an examination culture, even when the instructors explicitly state that their performance (in the experiment) will not affect their grades. Professionals are used to working under pressure on different timescales than the experimental contexts, hence they are not as committed/responsible in an experiment as in their daily work. Their natural focus is on solving problems rather than on reaching the end within the limited time provided.

The same author has also observed that students in his experiments adhered to the treatment more closely than professionals did. He provided some hypotheses of why this might be happening: It could be the result of unspoken authority in the class, of the students' willingness to learn a new technology, or simply of a "monkey see – monkey do" approach to learning. In contrast, it is hard to enforce any (treatment) on professionals by the book, as they are likely to behave in a more independent way based on their interpretation of the topic under study – which results in multiple realizations of the same concept in practice. Another author commented that even asking professionals to apply a certain technique was a bit controversial. He explanation: since in their daily work (particularly in organizations applying agile practices), developers have the freedom to choose how to develop software, they felt somewhat forced and uncomfortable if they are "obliged" to apply a specific technique. Therefore, the authors agreed that: *ST3 - Conducting experiments with professionals entails a higher treatment conformance threat to validity than experiments with students* (see Section 4.1.2).

4.2.4 Cycles too long

One author commented that experimentation in software engineering is often a trade-off between a high degree of validity and the pace of getting results. Having a sufficient level of external validity with students as experimental subjects often seems to be a better solution than aiming at higher external validity at the cost of much longer experimental cycles due to negotiation with companies and allocation of time for the experiment, which very often is delayed several times due to pressing deadlines in industry. The contexts are typically changing so fast that generalizable results are useless if they do not match the context anymore. Accelerating learning cycles can be seen as an engineering principle and getting sufficiently valid results with students might support this goal more efficiently. Therefore, the authors agreed that: **ST12 - Conducting experiments with professionals entails longer learning cycles than experiments with students**.

4.2.5 Resistance to change

Practitioners might use defensive response styles, which systematically underestimate the effect induced by a new technology. Additionally, it is possible that practitioners might assume a rather critical or opposing position against new methods because it is not (yet) clear whether these new methods are beneficial and because they want to avoid the high effort required for learning these methods. Therefore, the authors agreed that: **ST13 – When conducting experiments with professionals, the positive effects of a new technology are underestimated more than in experiments with students**.

4.3 The representativeness issue

Several authors highlighted that the suitability and the representativeness of students as proxies for professional developers change with different contexts and, most importantly, with different types of population.

One author said that based on his experience, students are reliable proxies for the developer population of typical start-up environments or SMEs [2]. Start-ups often hire young developers or graduates from universities. One of the reasons might be that software development in start-ups is mainly done to validate a business model with so-called minimum viable products (MVP) until the product is fit for the market. Developing such MVPs usually does not require advanced development skills and deep expert knowledge or experience.

Another author believes that students are reliable proxies for the developer population of OSS ecosystems, i.e., developers who are involved in the development and maintenance of open source software. With appropriate mentoring, students are able to quickly achieve a similar performance level as average OSS developers or even outperform them.

The authors acknowledge that there might be populations where students are not reliable proxies. This is particularly the case in domains where the developers need to have a significant amount of domain knowledge (such as the aerospace domain or the automotive domain). Students usually do not have such domain knowledge and it is difficult to train them in a way that they get a comparable level of knowledge.

Another author noticed that, in several domains, software developers in small and large companies (exception: critical domains) are quite young and therefore comparable to students. This has to do with the typical career path of a developer. This author has seen very few software developers who continue being developers for more than ten years, although exceptions exist. Specifically, and particularly in Europe, developers move on into other roles such as product managers or coaches. Therefore, there are not so many highly skilled professionals who really do the development, and so students are reliable proxies of the software developer population.

In summary, every subject sample is representative of a certain population. Moreover, the technologies need to be assessed for all populations that are candidates for their use. So far, we have treated professionals as belonging to one population and students as belonging to another. We need much more research aimed at understanding the subpopulations, for some of which students can be proper representations. Representativeness in general is a matter of whether the sample (the participants in an experiment) is representative of the population of interest. Thus representativeness is not only an issue in experiments with students. Note that a certain set of professionals participating in an experiment can also have low representativeness for the whole developer population, e.g., if the professional participants are more experienced than the average developer population.

There is no general answer to the question whether students are representing practitioners from industry well. However, there are occasions (i.e., experience or context) where students are well suited as representative subjects. Furthermore, the representativeness of students is highly dependent on the treatment and context. What is often missing is the viewpoint of the effect of the treatment on the subjects' representativeness. Specifically, it is not so important if students perform as well as professionals in absolute terms, but the effect of the treatment and whether it is similar with students and with professionals (as shown in the examples in Section 2) is. Therefore, the authors agreed that: **ST14 - The suitability and representativeness of students as proxies for professional developers change with different contexts and with different types of population.** The 14 statements listed in this section formed the input to the survey.

5 Survey Results

In order to investigate the level of agreement of experts other than the authors of this paper, we used a web survey to ask the ISERN members to rate our 14 statements (more methodological details are available in Section 3.2). Our survey response rate was 64%. In fact, out of 58 invitations, 37 subjects answered our survey. In order to analyze the results, we first removed incomplete answers; in total, three answers were removed. Because we are interested in experts' opinions only, we then removed the answers from respondents claiming not to be experts (i.e., claiming to be novices or amateurs); this resulted in the removal of another seven responses. Thus, our survey analysis concerns the answers of 27 subjects who are experts in performing software engineering experiments.

Figure 2 reports the levels of agreement of the experts on different statements. In order to facilitate analyzing the results, we computed two scores for each statement: Agreement and Opinion level. We computed the agreement level as the number of agreements (partial or complete) divided by the number of agreements and disagreements (partial or complete). The agreement score describes how many times an expert with an opinion (i.e., excluding "I do not know" answers) agreed with the statements. Then we classified the statements according to the agreement score as High (agreement >75%), Low (agreement < 60%), or otherwise Medium. We defined the thresholds of the different levels of agreement (i.e., 75% and 60%) with the aim of achieving the same population size for each level. Regarding the Opinion level, given that the statements are clearly in two different clusters, they were classified as High when the number of "I do not know" answers was lower than 15%, and Low otherwise.

Table 1 reports the values of the Agreement and Opinion level for each statement in descending order. Table 1 has been computed by applying the thresholds described above to the data reported in Figure 2. The data will be further analyzed in the following section.

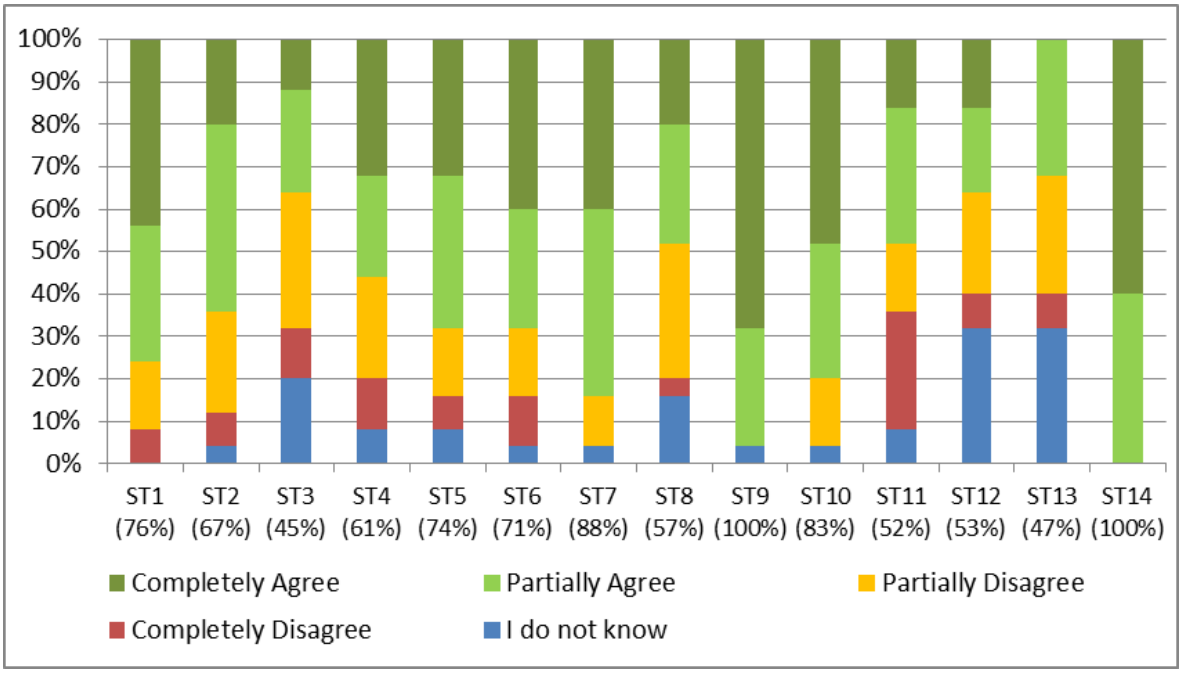

Figure 2: Experts' answers for each statement. The percentage near each statement ID is the percentage of times an expert agreed (partially or completely) over the times they expressed an opinion.

6 Analysis

6.1 Overall results

Before going into the results of each specific statement, it is interesting to note that the average percentage of "I do not know" answers is about 9% and therefore we can claim that the experts had, in general, a very good understanding of the statements for which they were asked to provide an opinion.

There are four statements on which none of the experts completely disagreed (i.e., S7, S9, S10, S14); in two of them, they did not even partially disagree (i.e., S9 and S14). Moreover, the agreement is higher than 50% in all statements except two (i.e., S3 and S13). Thus, we conclude that the experts are mostly in agreement with our statements.

It is interesting to note that the two statements where there is disagreement are both about highlighting the drawbacks of using professionals. In the same vein we note that *the only statements having a low Agreement level or a low Opinion level are all related to the use of professionals as subjects in software engineering experiments*. A possible reason for the low Opinion level may be that experiments with professionals are scarce in number and hence the experts are more confident providing opinions about students rather than about professionals. A possible reason for the low Agreement level may be that experts tend to see professionals as more appropriate than students as participants in experiments. However, because experiments with professionals are scarce in number, we highly question the validity of this belief. Moreover, if some experts really believe that professionals are more appropriate than students in performing experiments, then this paper makes a valuable contribution to the discussion about the pros and cons of using professionals as subjects in experiments.

6.2 Added statements

Two statements achieved an agreement of 100%. Thus, based on the results, we identified two additional statements, which particularly target papers describing experiments.

The first statement that received 100% agreement is: *ST9 - We should think about population and validity already before conducting the experiment, at the time when we are planning to use convenience sampling.* This type of analysis and motivation is missing in several papers on experiments. Because this statement achieved 100% agreement, we can claim that **ST15 - Papers should report a precise description of how convenience sampling was determined during the planning phase of the experiment.**

The last statement in the survey was: *ST14 - The suitability and representativeness of students as proxies for professional developers change with different contexts and with different types of population.* Given that this statement achieved 100% agreement, we can claim that the community acknowledges that **ST16 - Papers should provide a precise description of the reasons why the use of students as subjects in the study was appropriate under the specific circumstances of the experiment reported**. On the other hand, reviewers should criticize the use of students not as a general statement valid for all experiments, but by detailing why it was inappropriate under the specific circumstances in the experiment presented.

Table 1: Agreement and Opinion levels of each statement.

Statements	Agreement level	Opinion level
ST1 - Classifying experiment participants using a binary scale (students or professional) is an inappropriate approach.	High	High
ST7 - Students participating in an experiment are a convenient sample.	High	High
ST9 - We should think about population and validity already before conducting the experiment at the time when we are planning to use convenience sampling.	High	High
ST10 - The use of students supports the improvement of experiment design and protocol better than the use of professionals.	High	High
ST14 - The suitability and representativeness of students as proxies for professional developers change with different contexts and with different types of population.	High	High
ST2 - Experiments with students might exhibit lower external validity than experiments with professionals.	Medium	High
ST4 - Internal and external validities are of equal importance.	Medium	High
ST5 - Experiments with students are not of lower relevance than experiments with professionals.	Medium	High
ST6 - Experiments with students are not of less interest than experiments with professionals.	Medium	High
ST3 - Conducting experiments with professionals entails a higher treatment conformance threat to validity than experiments with students.	Low	Low
ST8 - Professionals participating in an experiment are a convenient sample.	Low	Low
ST11 - Conducting experiments with professionals as a first step should not be encouraged unless high sample sizes are guaranteed.	Low	High
ST12 - Conducting experiments with professionals entails longer learning cycles than experiments with students.	Low	Low
ST13 – When conducting experiments with professionals, the positive effects of a new technology are underestimated more than in experiments with students.	Low	Low

6.3 Reformulated statements

Although we do not know the precise reasons why specific experts disagreed with specific statements, we tried to find an explanation. Therefore, in the following we will provide a revised set of statements that, if subjected to another survey, will be very likely to meet with more agreement on the part of the experts.

ST3 - Conducting experiments with professionals entails a higher treatment conformance threat to validity than experiments with students. This statement has a low level of agreement and a low opinion level. However, we expected complete agreement. We recognized that there could be cases where professionals understand the authority of the researcher and are willing to follow the guidelines like students. Again, these cases are rare but we acknowledge that they do exist. Therefore, we revise ST3 to **ST3* - *Conducting experiments with professionals often entails a higher treatment conformance threat to validity than experiments with students.***

ST4 - Internal and external validities are of equal importance. This statement has a high opinion level and only medium agreement. However, we expected complete agreement. Now we realize that "equal" has a strong meaning and that what we meant is better phrased as: ***ST4* – Internal and external validities often have the same importance****".* However, some experts might still disagree with us by preferring experiments with high external validity (e.g., by using professionals rather than students) over those with high internal validity. In general, there is a tendency to prefer experiments with professionals, which indicates that external validity seems to be more important. For example, the authors of this paper have had better luck publishing papers with high external validity and low internal validity than vice versa, i.e., we claim that publishing papers with professionals was intrinsically easier, even though the overall validity was equal to that of papers rejected because the studies were conducted with students as subjects.

ST7 - Students participating in an experiment are a convenient sample. And *ST8 - Professionals participating in an experiment are a convenient sample.* These statements have a high, respectively low level of agreement. However, we expected complete agreement on both. Given the outcome, however, we recognized that there might be cases where students or professionals participating in an experiment are not a convenient sample, for example if they are randomly selected from all courses at the university or from all employees in a company. These cases are rare, but we acknowledge that they do exist. Therefore we revise ST7 to **ST7* - Students participating in an experiment are very often a convenient sample.** And ST8 to **ST8* - Professionals participating in an experiment are very often a convenient sample.**

ST10 - The use of students supports the improvement of experiment design and protocol better than the use of professionals. This statement has a high level of agreement and nobody completely disagreed with it. However, we expected complete agreement. We do acknowledge that there might be cases where one could tune the experiments by using professionals. Again, these cases are rare, but we acknowledge that they do exist. Therefore we revise ST10 to **ST10* - The use of students very often supports the improvement of experiment design and protocol better than the use of professionals.**

ST11 - Conducting experiments with professionals as a first step should not be encouraged unless high sample sizes are guaranteed. This statement has a low level of agreement and a high opinion level. Based on the results, we realize that there might be cases where professionals are as available and as cheap as students and therefore an experiment could start with professionals. Again, these cases are rare, but we acknowledge that they do exist. Therefore we revise ST11 to **ST11* - Conducting experiments with professionals as a first step should not be encouraged unless high sample sizes are guaranteed or performing replications is cheap.**

6.4 Non-reformulated statements with disagreement

We note that one of the reasons behind the survey is to recognize the existence of disagreement among experts about the topic. However, although we do not want to convince all experts of all our statements, there are statements where we are convinced that they are correct despite some disagreements. Therefore, we believe that the observed low level of agreement is either due to the absence of context (which would have increased agreement) or due to the limited experience of the experts in performing experiments with both populations of subjects. Therefore, in this section we report the statements for which we are still convinced that the statements are correct although the observed agreement is low.

ST1 - Classifying experiment participants using a binary scale (students or professional) is an inappropriate approach. This statement has a high level of agreement and nobody completely disagreed with it. However, we expected complete agreement, which is why in the next section we will propose a new characterization scheme. On the one hand, we know that the statements could have received more agreement if the term "inappropriate" had been replaced by "not completely appropriate". However, imperfection is the norm and hence, formally, everything can be considered as being not completely appropriate. Thus, the revised statement "*Classifying experiment participants using a binary scale (students or professional) is an approach that is not completely appropriate"* would have no meaning. It is our opinion that a binary classification is weak (see Section 4.1.1) and hence future efforts should be spent on developing better characterization schemes.

ST2 - Experiments with students might exhibit lower external validity than experiments with professionals. This statement has a medium level of agreement and a high opinion level. This is the statement where we are most surprised that agreement is not complete. The only reason we could think of why experts disagreed is that they would have preferred substituting "might" with "always". However, the statement "*Experiments with students always exhibit lower external validity than experiments with professionals"* is wrong because there could be cases where professionals are less representative than students, like for instance when professionals are different (i.e., particularly younger or older) from the average professionals.

ST5 - Experiments with students are not of lower relevance than experiments with professionals, and *ST6 - Experiments with students are not of less interest than experiments with professionals.* Both these statements have a high level of opinion and only a medium level of agreement. However, we strongly believe they are correct. Possible reasons for the agreement to be only medium could be the absence of context and also the use of negation in both sentences, which could have tricked the experts' reasoning. Another possible reason is that the majority of experts prefer professionals to students as subject of experiments. This could be because they do not have experience in using professionals and therefore they do not know that, although using professionals does not have the drawbacks encountered with students, it has drawbacks of its own (see Section 4.2).

ST13 – When conducting experiments with professionals, the positive effects of a new technology are underestimated more than in experiments with students. This statement has a low level of agreement and a low opinion level. Moreover, no expert completely agreed with it. However, we expected complete agreement. One possible reason for the disagreement is the absence of context explaining why professionals are less open to change technology than students. Another reason is that the statement is true only if the subject of the experiment is a technology that is new to the professionals. However, this condition is already well phrased and therefore we do not see how to improve on the sentence. Another possible reason is that the experts never reflected on this specific point. With this paper we aim to encourage readers to reflect about the dichotomy of students versus professionals that researches have not reflected about sufficiently.

7 Towards a characterization of experience

So the key to whether a set of participants in an experiment is representative of a population lies in an appropriate characterization of the subjects. Unfortunately, we do not yet understand which factors are crucial for characterizing experimental subjects. "Students vs. professionals" as a characterization is too simplistic and does not necessarily capture the most important aspects when it comes to participants in an experiment. Most human-based experiments only report the amount of experience without detailing other important characteristics. There exist efforts aimed at coming up with proper ways to characterize the experience of people in software development, for example the work on measuring programming skills [35]. Admittedly, it is a challenging task to measure skills. One of our experts suggested the R^3-characterization scheme as a step forward to focus on the characterization of the actual experience of a subject rather than using simplistic role-oriented labels such as student or professional. The suggestion is based on the experience of the proposer and the other experts on experimentation in software engineering have reviewed the scheme.

The R^3 scheme characterizes Real, Relevant, and Recent experience, hence the name R^3. It is important to highlight that these dimensions are not easily measured and should be judged in the context of every experiment and with respect to the objective of the experiment in terms of external validity. The three dimensions are hierarchical in the sense that recent experience is a subset of relevant experience (or potentially exactly the same), and relevant experience is in a similar way a subset of real experience. If possible, it would be preferable if information about these three dimensions were collected through interviews (rather than questionnaires) because this would be likely to reflect the most accurate representation of the experience. It would also give the researcher an opportunity to pose follow-up questions to capture the actual experience in the best possible way. However, this might not be possible, for example because there are too many subjects or permitted access to the subjects is not sufficient (in particular in an industrial setting). These three dimensions of experience can be summarized as follows:

- **Real** – to what extent does a subject have real experience?

  This dimension captures whether or not the subject has real experience, and if so how much. The first step is to define what is meant by real experience. This could, for example, be experience such as: participation in industrial software development projects, project work in a student setting, participation in open source projects, or any other form of experience judged relevant by the researcher in dialogue with each subject. The type of real experience should be listed when reporting this dimension. The amount of real experience is best divided into classes, for example: none, short, medium, and long, where short may be 0-2 years, medium 2-5 years, and long experience may be classified as more than 5 years. Whether these classes are the best remains to be researched. If a subject does not have real experience, the following two dimensions become non-applicable, as they are subsets of real experience.

- **Relevant** – to what extent is the real experience of a subject relevant?

  Industrial experience (or any other real experience) is not necessarily relevant for the task in the experiment, and hence the relevance of the experience should be evaluated. Relevant may also be knowledge of the domain or other knowledge that is considered important for conducting the experiment and judging a subject's representativeness for the intended population. It is most likely impossible to define relevant experience in general, and hence it is important that researchers clearly report what is determined as being relevant for each specific experiment. In a similar way as for the previous dimension, it is recommended

dividing relevant experience into classes, for example: N/A (if subject has no real experience as captured by the first dimension); none (if subject has real experience, but it is not relevant for the current experiment); short, medium, and long, where short may be 0-2 years, medium 2-5 years, and long experience may be classified as more than 5 years. Once again, further research is needed to determine whether these classes are the best.

- **Recent** – to what extent does a subject have recent (relevant) experience?

  It may be the case that subjects have experience that is real and relevant for the experiment, but the experience is getting old. A subject with old experience is not likely to perform as well as a subject with more recent experience. This was observed as part of the analysis of the experiment reported by Bratthall et al. [36], where one subject had real and relevant experience, but it was acquired ten years before the experiment was conducted, and hence the subject did not perform as well as could have been expected. The definition of recent must be judged in each case, but a characterization is needed, for example: N/A (if subject has no real experience as captured by the first dimension, or if subject has real experience but no relevant experience as captured by the second dimension); more than 5 years ago, 2-5 years ago, and 0-2 years ago, which may be denoted as old, medium, and new, respectively. Here, it should be noted that a lower number of years is better than a higher number of years. As for the other dimensions, there is a need for further research to determine whether these classes are better than other options.

The proposal suggests scales and intervals to measure the three characteristics: Real, Relevant, and Recent experience. The suggestions are based on experience, but their actual validity has not been evaluated. Thus, the suggestions should not be necessarily taken as the right intervals. More research is needed to determine suitable scales and intervals. At this stage, the primary recommendation is that researchers should use experience scales with respect to the three characteristics and clearly report which intervals they used. Based on this, it is hopefully possible to converge on suitable scales and intervals, although they may be different for different types of software engineering experiments. However, by clearly reporting the scales and intervals used, it is possible to use them in replications as well as to challenge the choices made. Thus, without further experiences and evidence it is close to impossible to generally define the three characteristics in such a way that they are perceived as valid for all different types of experiments in software engineering.

We recommend capturing the experience for each individual subject and then summarizing the results. The individual subject experience can be reported in a table as illustrated in Table 2. In the example, the experience of six subjects is shown using the scales and intervals above. In the example, the following should be noted:

- SU1 does not have any real experience, and hence relevant and recent are not applicable for S1. This is typically a student without any real experience.
- SU2 has some real experience, but it is not relevant for the current experiment. Thus, the recent experience is listed as non-applicable in Table 2.
- SU3 has medium real experience, and some of it is relevant. However, it was some time ago (more than five years ago).
- SU4 has medium real experience (2-5 years), and it includes medium relevant experience (2-5 years) and is also recent (some of it was less than two years ago).
- SU5 has long experience (more than five years), some of it is judged as relevant for the experiment (in this case 2-5 years), and it was also quite recent (2-5 years ago).

- SU6 has long and relevant experience (more than five years of relevant and real experience), and some of it is also recent (0-2 years ago). This is an experienced software engineer who ought to be regarded as a very good subject for the experiment.

Table 2: A subject table for the R³-characterization scheme.

Subject	Real	Relevant	Recent
SU1	None	N/A	N/A
SU2	Short	None	N/A
SU3	Medium	Short	Old
SU4	Medium	Medium	New
SU5	Long	Medium	Medium
SU6	Long	Long	New

Furthermore, we recommend providing a separate summary in a table. A summary table for the example is illustrated in Table 3. In Table 3, it can be seen how the six individuals are spread over the three characteristics: Real, Relevant, and Recent experience. It should be noted that the sum of each row should be equal to the number of subjects in the experiment, which in this case is six.

Table 3: A summary table of the example for the R³-characterization scheme.

Dimension					
Real		None: 1	Short: 1	Medium: 2	Long: 2
Relevant	N/A[3]: 1	None: 1	Short: 1	Medium: 2	Long: 1
Recent	N/A[4]: 2		Old: 1	Medium: 1	New: 2

Preferably, software engineering experimenters should report the experience of subjects in a table as illustrated in Table 2 and also discuss and justify the relevance of the experience for the experiment at hand. In Table 3, it is preferable having the higher numbers in the right part of the table. Very often we want participants in experiments with long, relevant, and recent experience that is independent of the formal role they may have, such as students or professional. Notice, however, that for some specific experiments, the appropriate experience might be different for different subjects. For example, if the experiment investigates the effect of a technology in novices or more experienced developers or another specific subset of developers, then the experiment participants should be representative of such a subgroup and their experience should match that of the subgroup.

We suggest using Table 2 and Table 3 as templates for researchers conducting software engineering experiments with human subjects. If possible, both tables ought to be included in papers presenting such software engineering experiments. If this is impossible, for example due to page restrictions, the individual characterization should preferably be made available online to

[3] This would be the same number as "none" in the first row.
[4] This would be the sum of "N/A" and "none" in the second row.

ensure transparency regarding the actual characterization, and the summary table should always be included in the paper describing the experiment.

The R^3-characterization scheme would allow for a more transparent characterization of subjects in software engineering experiments. Based on our experience, it also captures better what actually matters with respect to a subject's experience. This also means that the characterization should preferably be done upfront, as it also provides an important input when assigning subjects to groups and treatments (i.e., blocking, see Wohlin et al. [27]).

With this type of characterization of R^3-experience, it becomes easier to argue and discuss the comparison of different types of subjects vs. the target population (i.e., often professional software engineers conducting the type of task being investigated via an experiment). For example, it would become easier to compare master students and software consultants being hired to participate in an experiment. The latter may be exemplified by the research conducted by Sjøberg et al. [37] to make software engineering experiments more realistic. However, it is unknown how the subjects in the experiment by Sjøberg et al. would be characterized according to the three dimensions listed above given that the information is not available in [37].

There are several other characteristics of experimental subjects that may be relevant for characterizing them because they might affect experiment results. For example, motivation is a well-known attribute affecting human performance. However, motivation is harder to capture in a fair and non-intrusive way than experience.

8 Threats to validity

In this section, we report the threats to validity related to our study. The threats are organized by type (i.e., Conclusion, Internal, Construct, and External) and by the method we applied (first the focus group and then the survey).

Conclusion validity regards issues that affect the ability to draw accurate conclusions about relations between the treatments and the outcome of an experiment [27]. We do not see any major threats of this type because we applied the expected research methods to get the experts' opinions (i.e., focus group and survey). Regarding the focus group, it perfectly matches our need to stimulate discussion and brainstorming.

Internal validity regards the influences that can affect the independent variables with respect to causality [27]. The asynchronous mode of our focus group could be considered as a threat to the interactive nature of focus groups. However, while reporting their experiences in conducting (computer-mediated, electronic) focus groups, Kontio et al. state that "... the session could have been conducted online as well (i.e. not necessarily being at the same place nor at the same time)" [20]. Moreover, the use of an asynchronous focus group is justified by our aim to elicit as many valid statements as possible, and to iterate until no new ideas emerge. Another threat to internal validity, related to the focus group, concerns the thematic analysis performed by the first two authors of the paper. Specifically, there could be opinions that did not emerge or that we failed to collect during the ISERN 2014 meeting. We believe that we sufficiently mitigated this threat by having numerous rounds of discussion among the seven authors of the paper.

Construct validity regards the ability to generalize the results of an experiment to the theory behind the experiment [27]. We do not see any major threat to construct validity in the focus group, whereas some threats exist in the survey. The main threat to construct validity regards the phrasing we adopted to define our statements. In order to mitigate this threat, we tried to avoid ambiguities and misunderstanding by performing a pilot survey in which all seven authors

participated and refined the language in the statements. Moreover, we were careful in interpreting disagreement answers as a result of ambiguity in the description. As a result, we revised the formulations of six statements. Another threat to construct validity is the fact that we (both authors and experts), being authors and mostly academics, have more access to students than professionals and hence we are writing this paper to prove we can use students; in other words, we are trying to improve our chances of getting our papers accepted. Therefore, we might have unconsciously favored the opinion against professionals. However, we note that all authors have experience in performing experiments with professionals. When performing experiments with professionals, we expected a lot (as we had been told they would be the perfect experimental subjects), but we encountered subjects with even more problems than students, who in many cases jeopardized the validity of our experiments.

External validity regards the extent to which the research elements (subjects, artifacts, etc.) are representative of actual elements [27]. The main threat to the external validity of this work concerns both the focus group and the survey, and is about the representativeness of the experts from which our conclusions are derived. Specifically, we do recognize that our sampling of experts was based on convenience. It was impossible to reach all experts, but because ISERN is the research network on Empirical Software Engineering, we believe the sample is highly representative of experts on software engineering experiments.

9 Discussion

Related work as presented in Section 2 is primarily related to having different types of subjects and then comparing the outcomes in the experiments conducted. In this paper, the focus has been on capturing the experience of a number of experts. Seven experts (the authors of the paper) set out to address the challenge. This was then complemented by a survey performed with a set of experts on software engineering experimentation. Thus, the research presented in this paper focuses on synthesizing the experience and expertise in the research community in a qualitative manner rather than creating an additional quantitative data point by running an experiment with different categories of subjects.

Throughout the paper we have argued that the approach of dividing subjects into professionals and students is too simplistic. These two categories are not necessarily mutually exclusive. Students may have industrial experience or may even be working in parallel to their studies. Professionals may pursue studies in parallel to their regular work in industry. So, who is a student and who is a professional? This challenge was the starting point for our research, which was summarized in three research questions in Section 3.1. These questions are repeated here. In addition, we also summarize our main observations in relation to the questions.

RQ1: What are the pros and cons of using professionals and students as subjects in software engineering experiments?

This research question has been addressed throughout the paper and in particular in Section 4, which also forms the basis for the survey among experts. There is no general agreement that either one of these two categories of subjects is always better than the other. Professionals may be more representative, but they often come at a higher cost and might not be as accessible as students taking courses at a university. Furthermore, depending on the maturity of the experimental design as well as the technologies or methods being evaluated, students may be better as subjects because an experiment can be a learning experience for them, while for professionals the actual outcome is most likely the main concern for them. Thus, in summary, both professionals and students have their pros and cons, and it is impossible to state that one is

always better than the other. The subjects in an experiment should be determined based on the objectives of the experiment, for example in relation to its intention with respect to generalizability.

RQ2: As a community, what do we agree and disagree on with respect to subjects in software engineering experiments?

In Section 6, the analysis of the survey of experts on software engineering experimentation is presented. The experts are generally in agreement, although agreement is higher with respect to some statements than to others. In particular, there was very good agreement regarding two statements: *ST9 - We should think about population and validity already before conducting the experiment, at the time when we are planning to use convenience sampling,* and *ST14 - The suitability and representativeness of students as proxies for professional developers change with different contexts and with different types of population.* ST9 aims at complementing current guidelines on how to report experiments [27] and [38]. ST14 is in line with the disagreement among related work, specifically, several studies showed similarities [6], [7], [3], [8] and [9], and other studies showed dissimilarities [4], [10], [11], [12], [13], [14], [15] and [16] of results obtained in experiments run with students and professionals. Thus, students are clearly neither always nor never suitable as proxies for professional developers; it depends on different contexts and with different types of population. Based on the high level of agreement on ST9 and ST14, we added ST15 and ST16 (see Section 6.2), which aim at putting explicit requirements on research papers presenting experiments. For some other statements, agreement was not as high as expected, which is believed to be due to two main reasons: 1) the actual formulations of the statements, which resulted in some statements being reformulated (see Section 6.3), and 2) the fact that the context of the statements was not provided in the survey, and hence more discussions are needed to potentially obtain higher agreement (see Section 6.4).

RQ3: Is it possible to characterize subjects using a different classification than professionals or students?

Given that professionals and students are not mutually exclusive as subject categories, a scheme for characterizing subjects related to three perspectives on experience has been proposed. The scheme suggests that subjects should be characterized with respect to: real experience, relevant experience, and recent experience. The scheme is elaborated in Section 7. The objective of the scheme is to capture the actual experience of the subjects rather than classifying them into general categories such as professionals and students.

In summary, there is a continued need to better understand how different subjects and their characterization can help us make even better use of the outcomes of software engineering experiments. This understanding is needed to ensure that results from experiments can be generalized to the population of interest, and also to support the synthesis of findings from different experiments.

10 Conclusions and future work

On the one hand, we have observed too many times that our papers were rejected because we used students as subjects, and hence the software engineering experiments were judged as having severe threats to validity. On the other hand, reviewers have rated our papers with professional subjects as having high validity, probably too high in relation to the actual validity. Thus, we believe that a deeper understanding is needed regarding the internal and external validities of software engineering experiments conducted with students, respectively with professionals.

10.1 Conclusions

During a session at ISERN 2014, 65 empirical researchers (the seven authors and 58 other experts) argued and discussed this issue with an open mind. Afterwards, we revisited the topic, elicited the experts' opinions, and fostered discussion. Then we derived 14 statements and asked 58 experts to provide their level of agreement. Finally, we analyzed the opinions of the experts and refined the statements by adding two new statements, reformulated several of them, and finally provided more in-depth motivation and context for those original statements where we found disagreements between the authors and the experts responding to the survey. Our survey results showed that, in general, the experts responding to the survey disagreed with us about the drawbacks of using professionals. In contrast we strongly believe that no population (students, professionals, or others) can be deemed better than another in absolute terms.

The statements we as authors want to provide to the community are reported below. An * indicates that the statement was revised after the survey. Statements 15 and 16 were added after the survey:

- ST1 – Classifying experiment participants using a binary scale (students or professional) is an inappropriate approach.
- ST2 – Experiments with students might exhibit lower external validity than experiments with professionals.
- ST3* – Conducting experiments with professionals often entails a higher treatment conformance threat to validity than experiments with students.
- ST4* – Internal and external validities often have the same importance.
- ST5 – Experiments with students are not of lower relevance than experiments with professionals.
- ST6 – Experiments with students are not of less interest than experiments with professionals.
- ST7* – Students participating in an experiment are very often a convenient sample.
- ST8* – Professionals participating in an experiment are very often a convenient sample.
- ST9 – We should think about population and validity already before conducting the experiment, at the time when we are planning to use convenience sampling.
- ST10* – The use of students very often supports the improvement of experiment design and protocol better than the use of professionals.
- ST11* – Conducting experiments with professionals as a first step should not be encouraged unless high sample sizes are guaranteed or performing replicas is cheap.
- ST12 – Conducting experiments with professionals entails longer learning cycles than experiments with students.
- ST13 – When conducting experiments with professionals, the positive effects of a new technology are underestimated more than in experiments with students.
- ST14 – The suitability and representativeness of students as proxies for professional developers change with different contexts and with different types of population.
- ST15 – Papers should report a precise description of how convenience sampling was determined during the planning phase of the experiment.
- ST16 – Papers should report a precise description of the reasons why the use of students as subjects of software engineering experiment was appropriate in the specific circumstances.

In conclusion, using students as participants in software engineering experiments remains a valid simplification of reality needed in laboratory contexts. It is an effective way to advance software engineering theories and technologies but, like any other aspect of the study setting, should be carefully considered during experiment design, execution, interpretation, and reporting. The key is to understand which developer population portion is being represented by participants in an experiment. To better capture the experience of subjects (rather than using the oversimplification of dividing subjects into two categories only: students and professionals), we have proposed a characterization scheme for participants in experiments that – even if not perfect – can be used, evaluated, and improved to promote replication and generalization. We feel that reviewers might overestimate the validity of experiments using professionals. For example, professionals must, in many cases, be paid to be subjects in an experiment, which per se is a strong threat to validity. With this paper, we aim to support readers in reflecting about the dichotomy of students versus professionals that researches have not sufficiently reflected about.

10.2 Future work

Based on the experiences from the research reported herein, we summarize the future work as follows:

- We need to better understand the relationship between students and professionals as subjects in software engineering experiments. The 16 statements listed above is a start, but there is still a need to deepen our understanding with respect to in particular statements ST2- ST6, and ST13-ST14. Thus, we encourage further investigations into these statements. Several of the other statements are either a description of the current situation or recommendations for conducting software engineering experiments, and hence there is less need for further research on these.
- As a complement to keeping tracks of the background in terms of students and professionals of subjects, we recommend that future studies should report the experience according to the R3-characterization scheme. The objective being to enable to identify patterns and also facilitate comparisons between studies and their outcomes.
- The importance of each dimension in the R3-characterization scheme needs to be further investigated. It is important to try to understand their relative importance for the outcome of software engineering experiments. Are any of the dimensions more important for the outcome than the others, or is it the interplay between them that is important? Furthermore, there is a need to investigate if the proposed characterization scheme creates a better understanding than the somewhat simplistic characterization into students and professions.
- Within the R3-characterization scheme, there is a need to investigate the suggested classes of experience. What are suitable groupings, for example what is short and long relevant experience?

The future work listed above is not intended only as future research for the authors. The items listed are challenges for the empirical research community conducting software engineering experiments with human subjects, and hence we are looking forward to a community effort to increase our understanding of subjects in software engineering experiments.

Acknowledgments

We would like to thank all the ISERN 2014 participants for the inspiring and energetic discussions. We would like to thank both the anonymous experts and the following non-anonymous experts for participating in the survey: Paris Avgeriou, Teresa Baldassarre, Victor Basili, Giovanni Cantone, Jeff Carver, Tore Dybå, Hakan Erdogmus, Vladimir Mandic, Manuel Mastrofini, Daniel Mendez, Oscar Pastor, Guilherme Horta Travassos, Stephan Wagner, Qing Wang, Roel Wieringa, and Dietmar Winkler. We thank Sonnhild Namingha for proof reading the manuscript. This research is supported in part by the Academy of Finland Project 278354.

Appendix A

Invitation letter

Dear First name Last name,

During an ISERN 2014 session, we discussed the uses of students and professionals as subjects of software engineering experiments. Afterwards, Davide Falessi, Natalia Juristo, Claes Wohlin, Burak Turhan, Jürgen Münch, Andreas Jedlitschka, and Markku Oivo revisited the topic, elicited experts' opinions, fostered discussion, and derived some conclusions. These conclusions are currently summarized in a paper, we aim to publish in EMSE. The aim of this research is to identify pros and cons of using students and professionals in software engineering experiments. Our results aim to support researchers during generalization of results from experiments as well as reviewers during the evaluation of experiments.

As an attendee of ISERN 2014 meeting, we care about knowing your degree of agreement with the reached conclusions. We would appreciate if you could share with us your level of agreement on our conclusions. The aim of this survey is to add the community view on our conclusions. Because we care about your opinion, you should not hesitate or be afraid to disagree with our conclusions.

The survey takes approximately 10 minutes to be completed. Your answers will be confidential and will only be reported in an aggregated form. Reminders will be sent to non-respondents only. You will be personally acknowledged in the paper.

As a member of ISERN, please collaborate with us by participating to this survey by using this URL:

The deadline is in two weeks from now.

If you experience any troubles or you have any comment, please contact Dr. Davide Falessi at DFalessi@calpoly.edu

Best regards,

Davide Falessi, Natalia Juristo, Claes Wohlin, Burak Turhan, Jürgen Münch, Andreas Jedlitschka, Markku Oivo

Introduction

Thank you for participating in our survey on students and professionals in software engineering experiments.

The aim of this research is to identify pros and cons of using students and professionals in software engineering experiments. Our results aim to support researchers during generalization of results from experiments as well as reviewers during the evaluation of experiments.

The survey takes approximately 10 minutes to be completed. Your answers will be confidential and will only be reported in an aggregated form.

If you experience any troubles or you have any comment, please contact Dr. Davide Falessi at DFalessi@calpoly.edu

Experience

1. What is your level of experience in performing (designing, running, analyzing, reporting) controlled experiments?

-Expert: more than 3 published experiments

-Novice: between 1 and 2 published experiments

-Amateur: no published experiment or you do not know what an experiment is

Agreement

2. For the following statements, we would like to get your level of agreement (Completely agree, Partially agree, Partially disagree, Completely disagree, I do not know or I do not want to answer)

1. Experiments with students are not of lower relevance than experiments with professionals.
2. Experiments with students are not of less interest than experiments with professionals.
3. Experiments with students might exhibit lower external validity than experiments with professionals.
4. Internal and external validities are of equal importance.
5. Classifying experiment participants using a binary scale (students or professional) is an inappropriate approach.
6. Students participating in an experiment are a convenience sampling.
7. Professionals participating in an experiment are a convenient sampling.
8. We should think about population and validity already before conducting the experiment at the time when we are planning to use convenience sampling.
9. The use of students better supports the improvement of experiment design and protocol than professionals.
10. Conducting experiments with professionals as a first step should not be encouraged unless high sample sizes are guaranteed.
11. Conducting experiments with professionals entails a higher treatment conformance threat to validity than experiments with students.
12. Conducting experiments with professionals entails longer learning cycles than experiments with students.
13. Conducting experiments with professionals underestimate the positive effects of new technology than experiments with students.

14. The suitability and representativeness of students as proxies for professional developers change with different contexts and with different types of population.

Thank you

3. We appreciate the time and effort you spent in answering this survey. Please report your last name if you want to be personally acknowledged in the paper.

References

[1] B. Turhan and A. Bener, "A template for real world team projects for highly populated software engineering classes," in *Proceedings - International Conference on Software Engineering*, 2007, pp. 748–751.

[2] F. Fagerholm, N. Oza, and J. Munch, "A platform for teaching applied distributed software development: The ongoing journey of the Helsinki software factory," in *2013 3rd International Workshop on Collaborative Teaching of Globally Distributed Software Development, CTGDSD 2013 - Proceedings*, 2013, pp. 1–5.

[3] M. Svahnberg, A. Aurum, and C. Wohlin, "Using students as subjects - an empirical evaluation," in *Proceedings of the Second ACMIEEE international symposium on Empirical software engineering and measurement*, 2008, pp. 288–290.

[4] V. R. Basili and R. W. Selby, "Comparing the Effectiveness of Software Testing Strategies," *IEEE Trans. Softw. Eng.*, vol. SE-13, no. 12, pp. 1278–1296, Dec. 1987.

[5] A. A. Porter, L. G. Votta, and V. R. Basili, "Comparing detection methods for software requirements inspections: a replicated experiment," *IEEE Trans. Softw. Eng.*, vol. 21, pp. 563–575, 1995.

[6] A. Porter and L. Votta, "Comparing detection methods for software requirements inspections: A replication using professional subjects," *Empir. Softw. Eng.*, vol. 3, pp. 355–379, 1998.

[7] M. Höst, B. Regnell, and C. Wohlin, "Using Students as Subjects—A Comparative Study of Students and Professionals in Lead-Time Impact Assessment," in *Empirical Software Engineering*, vol. 5, 2000, pp. 201–214.

[8] P. Berander, "Using students as subjects in requirements prioritization," in *Proceedings - 2004 International Symposium on Empirical Software Engineering, ISESE 2004*, 2004, pp. 167–176.

[9] I. Salman, A. T. Misirli, and N. Juristo, "Are Students Representatives of Professionals in Software Engineering Experiments?," in *International Conference on Software Engineering (ICSE)*, 2015, pp. 666–676.

[10] W. Remus, "Using students as subjects in experiments on decision supportsystems," *[1989] Proc. Twenty-Second Annu. Hawaii Int. Conf. Syst. Sci. Vol. III Decis. Support Knowl. Based Syst. Track*, vol. 3, 1989.

[11] D. A. McMeekin, B. R. Von Konsky, M. Robey, and D. J. A. Cooper, "The significance of participant experience when evaluating software inspection techniques," in *Proceedings of the Australian Software Engineering Conference, ASWEC*, 2009, pp. 200–209.

[12] M. Ciolkowski, "What do we know about perspective-based reading? An approach for

quantitative aggregation in software engineering," in *2009 3rd International Symposium on Empirical Software Engineering and Measurement, ESEM 2009*, 2009, pp. 133–144.

[13] P. Runeson, "Using students as experiment subjects–an analysis on graduate and freshmen student data," *Proc. 7th Int. Conf. Empir. Assess. Eval. Softw. Eng.*, pp. 95–102, 2003.

[14] A. Wesslén, "Replicated empirical study of the impact of the methods in the PSP on individual engineers," *Empir. Softw. Eng.*, vol. 5, pp. 93–123, 2000.

[15] F. Ricca, M. Di Penta, M. Torchiano, P. Tonella, and M. Ceccato, "How developers' experience and ability influence web application comprehension tasks supported by UML stereotypes: A series of four experiments," *IEEE Trans. Softw. Eng.*, vol. 36, pp. 96–118, 2010.

[16] F. Salviulo and G. Scanniello, "Dealing with Identifiers and Comments in Source Code Comprehension and Maintenance: Results from an Ethnographically-informed Study with Students and Professionals," in *Proceedings of the 18th International Conference on Evaluation and Assessment in Software Engineering*, 2014, p. 48:1--48:10.

[17] D. L. Morgan, "Focus Groups," *Annu. Rev. Sociol.*, vol. 22, no. 1, pp. 129–152, Aug. 1996.

[18] S. Beecham, T. Hall, and A. Rainer, "Software Process Improvement Problems in Twelve Software Companies: An Empirical Analysis," *Empir. Softw. Eng.*, vol. 8, no. 1, pp. 7–42, Mar. 2003.

[19] M. Lindvall, V. R. Basili, B. W. Boehm, P. Costa, K. Dangle, F. Shull, R. Tesoriero, L. A. Williams, and M. V. Zelkowitz, "Empirical Findings in Agile Methods," in *Second XP Universe and First Agile Universe Conference on Extreme Programming and Agile Methods - XP/Agile Universe 2002*, 2002, pp. 197–207.

[20] J. Kontio, L. Lehtola, and J. Bragge, "Using the Focus Group Method in Software Engineering: Obtaining Practitioner and User Experiences," in *International Symposium on Empirical Software Engineering*, 2004, pp. 271–280.

[21] M. Tremblay, A. Hevner, and D. Berndt, "Focus Groups for Artifact Refinement and Evaluation in Design Research," *Communications of the Association for Information Systems*, vol. 26, no. 1. 2010.

[22] D. L. Morgan and R. A. Krueger, "When to use focus groups and why.," in *Successful focus groups: Advancing the state of the art.*, 1993, pp. 3–19.

[23] K. Jyrki, J. Bragge, and L. Lehtola, "The Focus Group Method as an Empirical Tool in Software Engineering," in *Guide to Advanced Empirical Software Engineering*, 2008, pp. 93–116.

[24] T. C. Lethbridge, S. E. Sim, and J. Singer, "Studying Software Engineers: Data Collection Techniques for Software Field Studies," *Empir. Softw. Eng.*, vol. 10, no. 3, pp. 311–341, Jul. 2005.

[25] G. Easton, A. Easton, and M. Belch, "An experimental investigation of electronic focus groups," *Inf. Manag.*, vol. 40, no. 8, pp. 717–727, Sep. 2003.

[26] N. Juristo and A. M. Moreno, *Basics of Software Engineering Experimentation*, vol. 5/6. 2001.

[27] C. Wohlin, P. Runeson, M. Höst, M. C. Ohlsson, B. Regnell, and A. Wesslén, *Experimentation in software engineering: an introduction*. Springer, 2012.

[28] K. E. Berg and R. W. Latin, *Essentials of Research Methods in Health, Physical Education, Exercise Science, and Recreation*. LWW, 2003.

[29] J. Siegmund, N. Siegmund, and S. Apel, "Views on Internal and External Validity in Empirical Software Engineering," in *International Conference on Software Engineering (ICSE)*, 2015.

[30] M. H. and C. M. C. Wohlin, A. Gustavsson, “A Framework for Technology Introduction in Software Organizations,” in *Software Process Improvement Conference*, 1996, pp. 167–176.

[31] V. R. Basili, F. E. McGarry, R. Pajerski, and M. V. Zelkowitz, “Lessons learned from 25 years of process improvement: The rise and fall of the NASA software engineering laboratory,” in *Proceedings - International Conference on Software Engineering*, 2002, pp. 69–79.

[32] V. R. Basili and R. W. Reiter, “A Controlled Experiment Quantitatively Comparing Software Development Approaches,” *IEEE Trans. Softw. Eng.*, vol. SE-7, no. 3, pp. 299–320, May 1981.

[33] T. Gorschek, P. Garre, S. Larsson, and C. Wohlin, “A model for technology transfer in practice,” *IEEE Softw.*, vol. 23, no. 6, pp. 88–95, 2006.

[34] S. Vegas, O. Dieste, and N. Juristo, “Difficulties in Running Experiments in the Software Industry: Experiences from the Trenches,” in *Proceedings - 3rd International Workshop on Conducting Empirical Studies in Industry, CESI 2015*, 2015, pp. 3–9.

[35] G. R. Bergersen, D. I. K. Sjoberg, and T. Dyba, “Construction and Validation of an Instrument for Measuring Programming Skill,” *IEEE Trans. Softw. Eng.*, vol. 40, no. 12, pp. 1163–1184, 2014.

[36] L. Bratthall, E. Johansson, and B. Regnell, “Is a Design Rationale Vital when Predicting Change Impact? – A Controlled Experiment on Software,” in *2nd International Conference on Product Focused Software Process Improvement (PROFES 2000)*, 2000, pp. 126–139.

[37] D. I. K. Sjoberg, B. Anda, E. Arisholm, T. Dyba, M. Jorgensen, A. Karahasanovic, E. F. Koren, and M. Vokac, “Conducting realistic experiments in software engineering,” *Proc. Int. Symp. Empir. Softw. Eng.*, 2002.

[38] A. Jedlitschka and D. Pfahl, “Reporting guidelines for controlled experiments in software engineering,” in *2005 International Symposium on Empirical Software Engineering, ISESE 2005*, 2005, pp. 95–104.