Industrial comparability of student artifacts in traceability recovery research

- An exploratory survey

Markus Borg, Krzysztof Wnuk, Dietmar Pfahl
Department of Computer Science
Lund University
Lund, Sweden
{Markus.Borg, Krzysztof.Wnuk, Dietmar.Pfahl}@cs.lth.se

***Abstract*— About a hundred studies on traceability recovery have been published in software engineering fora. In roughly half of them, software artifacts developed by students have been used as input. To what extent student artifacts differ from industrial counterparts has not been fully explored in the literature. We conducted a survey among authors of studies on traceability recovery, including both academics and practitioners, to explore their perspectives on the matter. Our results indicate that a majority of authors consider software artifacts originating from student projects to be only partly representative to industrial artifacts. Moreover, only few respondents validated student artifacts for industrial representativeness. Furthermore, our respondents made suggestions for improving the description of artifact sets used in studies by adding contextual, domain-specific and artifact-centric information. Example suggestions include adding descriptions of processes used for artifact development, meaning of traceability links, and the structure of artifacts. Our findings call for further research on characterization and validation of software artifacts to support aggregation of results from empirical studies.**

Keywords- survey, traceability, software artifacts, empirical study

I. INTRODUCTION

To advance both the state-of-art and state-of-practice in software engineering, empirical studies (i.e., surveys, experiments, case studies) have to be conducted [8, 23]. In order to investigate the cause-effect relationships of introducing new methods, techniques or tools in software engineering, controlled experiments are commonly used as the research method. Despite the benefits resulting from the controlled environment that can be created in this fixed research design [22, 28], controlled experiments are expensive due to the involvement of human subjects. Therefore, controlled experiments are often conducted with university students – and not with engineers working in industry.

Several researchers have studied the differences between using students and practitioners as subjects in software engineering studies, since the inadequate choice of subjects might be a considerable threat to the validity of the study results [10]. A number of publications report that the differences are only minor, thus using students is reasonable under certain circumstances [2, 10, 11, 16, 27]. Senior students, representing the next generation of professional software engineers, are relatively close to the population of interest in studies aiming at emulating professional behavior [16]. Further, for relatively small tasks, trained students have been shown to perform comparably to practitioners in industry [10, 27].

However, the question whether software artifacts (referred to as only 'artifacts' in this paper) produced by students should be used in empirical studies has been less explored. How useful are results from such studies when it comes to generalizability to industrial contexts? Using student artifacts is often motivated by low availability of industrial artifacts due to confidentiality issues. A qualification of the validity of student artifacts is particularly important for the domain of traceability recovery, since student artifacts frequently have been used for tool evaluations [6, 7, 20]. Since several software maintenance tasks (such as change impact analysis and regression testing) depend on up-to-date traceability information [14], it is fundamental to understand the nature of experimental artifact sets.

Furthermore, as presented in more detail in Section II, the reported characterization of artifact sets used as input to experiments on traceability recovery is typically insufficient. According to Jedlitschka *et al*., inadequate reporting of empirical research commonly impedes integration of study results into a common body of knowledge [15]. This applies also to traceability recovery research. First, insufficient reporting makes it harder to assess the validity of results using student artifacts (even if artifacts have been made available elsewhere). Second, it hinders aggregation of empirical results, particularly when closed industrial artifact sets have been used (that never can be made available).

In this paper, we present a study exploring differences between Natural Language (NL) artifacts originating from students and practitioners. We conducted a questionnaire-based survey of researchers with experience in doing experiments on traceability recovery using Information Retrieval (IR) approaches. The survey builds upon results from a literature study, which will be published in full detail elsewhere.

This paper is structured as follows. Section II presents background, including a short overview of related work on IR-based traceability recovery and using students in software engineering experiments. Section III presents the research design and how the survey was conducted. In Section IV we present and analyze our results in comparison to the literature. Section V describes threats to validity. Finally, Section VI concludes the paper and discusses future work.

II. BACKGROUND AND RELATED WORK

Using students as subjects in empirical software engineering studies has been considered as reasonable by several researchers [2, 10, 11, 16, 27]. Kitchenham *et al.* claim that students are the next generation of software professionals and that they are relatively close to the population of interest (practitioners) [16]. Kuzniarz *et al.* concluded that students are good subjects under certain circumstances and proposes a classification of the possible types of students used in an experiment [17]. Svahnberg *et al.* investigated if students understand the way how industry acts in the context of requirements selection [27].

Höst *et al.* [11] investigated the incentives and experience of subjects in experiments and proposed a classification scheme in relation to the outcome of an experiment. Although Höst *et al.* distinguished between artificial artifacts (such as produced by students during a course) and industrial artifacts as part of the incentives in the proposed classification, guidelines on how to assess the two types of artifacts are not provided in this work. Moreover, none of the mentioned studies investigate whether artifacts produced by students are comparable to artifacts produced in industry and how possible discrepancies could be assessed.

Several empirical studies on traceability recovery were conducted using student subjects working on artifacts originating from student projects. De Lucia *et al.* evaluated the usefulness of supported traceability recovery in a study with 150 students in 17 software development projects at the University of Salerno [7]. They also conducted a controlled experiment with 32 students using student artifacts [6]. Natt och Dag *et al.* conducted another controlled experiment in an academic setting, where 45 students were solving tracing tasks on artifacts produced by students [20].

During the last decade, several researchers proposed expressing the traceability challenge, i.e., identifying related artifacts, as an IR problem [1, 6, 19]. The approach suggests traceability links based on textual similarity between artifacts, since text in NL is the common format of information representation in software development [19]. The underlying assumption is that developers use similar language when referring to the same functionality across artifacts.

In an ongoing literature review, we have identified 59 publications on IR-based traceability recovery of NL artifacts. Figure 1 shows the reported origin of artifacts used in evaluations of traceability recovery tools, classified as industrial artifacts, open source artifacts, university (artifacts developed in university projects, role of developers unspecified) or student (deliverables from student projects). Several publications use artifacts from more than one category, some do not report the origin of the artifacts used for evaluations. As Figure 1 shows, a majority of the artifacts originate from an academic environment, i.e. they have been developed in university or student projects.

The Center of Excellence for Software Traceability[1] (COEST) has collected and published four artifact sets (see Table 1), that constitute the de-facto benchmarks for IR-based traceability recovery research. In Figure 1, darker grey color represents artifact sets available at COEST. Among the 59 identified publications, the most frequently used artifact sets in traceability recovery studies are EasyClinic (marked with a letter D, 10 publications), CM-1 (B, 9 publications) and MODIS (A, 6 publications).

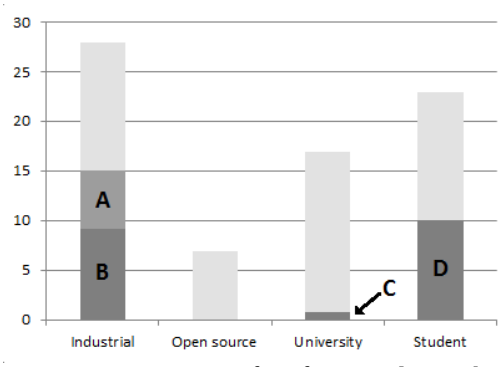

Figure 1. Origin of artifacts used in IR-based traceability recovery evaluations. Artifact sets in darker grey are available at COEST (A=MODIS, B=CM-1, C=Waterloo, D=EasyClinic).

Artifact set	Artifact types	Origin / Domain	Size (#artifacts)
MODIS	Requirements	NASA / Embedded	48
CM-1	Requirements	NASA / Embedded	555
Waterloo	Requirements	23 Student projects	23 x ~75
EasyClinic	Requirements, code, test cases	Student project	160

Table 1. Publicly available artifact sets at COEST (Oct 23, 2011).

In 2005, Huffman Hayes and Dekhtyar proposed "A framework for comparing requirements tracing experiments" [12]. The framework focuses on developing, conducting and analyzing experiments, but also suggests information about artifacts and contexts that are worth reporting. They specifically say that the average size of an artifact is of interest, but that it rarely is specified in research papers. Furthermore, they propose characterizing the quality of the artifacts and the importance of both the domain and object of study (on a scale from convenience to safety-critical).

Moreover, even though the framework was published in 2005, our review of the literature revealed that artifact sets often are presented in rudimentary fashion in the surveyed papers. The most common way to characterize artifact sets in the surveyed papers is to report its origins together with a brief description of the functionality of the related system, its size and the types of artifacts included. Size is reported as the number of artifacts and the number of traceability links between them. This style of reporting was applied in 49 of the 59 publications (83%). Only three publications thoroughly describe the context and process used when the artifacts were developed. For example, Lormans *et al.* well describe the context of their case study at LogicaCMG [18].

[1] www.coest.org

	Research question	Aim	Example answer
RQ1	When used as experiment inputs, how comparable are artifacts produced by students to their industrial counterparts?	Understand to what degree respondents, both in academia and industry, consider industrial and student artifacts to be comparable.	"As a rule, the educational artifacts are simpler."
RQ2	How are artifacts validated before being used as input to experiments?	Survey if and how student artifacts are validated before experiments are conducted.	"Our validation was based on expert opinion."
RQ3	Is the typically reported characterization of artifact sets sufficient?	Do respondents, both in academia and industry, consider that the way natural language artifacts are described is good enough.	"I would argue that it should also be characterized by the process by it was developed."
RQ4	How could artifacts be described to better support aggregation of empirical results?	Explore whether there are ways to improve the way natural language artifacts are presented.	"The artifacts should be combined with a task that is of principal cognitive nature."
RQ5	How could the difference between artifacts originating from industrial and student projects be measured?	Investigate if there are any measures that would be particularly suitable to compare industrial and student artifacts.	"The main difference is the verbosity."

Table 2. Research questions of the study. All questions are related to the context of traceability recovery studies.

Apart from mentioning size and number of links, some publications present more detail regarding the artifacts. Six publications report descriptive statistics of individual artifacts, including average size and number of words. Being even more detailed, Hayes *et al.* reported two readability measures to characterize artifact sets, namely Flesch Reading Ease and Flesch-Kincaid Grade Level [13]. Another approach was proposed by De Lucia *et al.* [5]. They reported subjectively assessed quality of different artifact types, in addition to the typical size measures. As stressed by Jedlitschka et al. proper reporting of traceability recovery studies is important, since inadequate reporting of empirical research commonly impedes integration of study results into a common body of knowledge [15].

III. Research Design

This section presents the research questions, the research methodology, and the data collection procedures used in our study. The study is an exploratory follow-up to the ongoing literature review mentioned in Section II. Table 2 presents the research questions governing this study. The research questions investigate whether the artifacts used in the reported studies are considered comparable to their industrial counterparts by our respondents. Moreover, the questions aim at exploring how to support assessing the comparability by augmenting the descriptions of the used artifacts.

For this study, we chose a questionnaire-based survey as the tool to collect empirical data, since it helps reaching a large number of respondents from geographically diverse locations [25]. Also, a survey provides flexibility and is convenient to both researchers and participants [8]. The details in relation to survey design and data collection are outlined in the section that follows.

A. Survey design

Since the review of literature resulted in a substantial body of knowledge on IR-based approaches to traceability recovery, we decided to use the authors of the identified publications as our sample population. Other methods to recover traceability have been proposed, including data mining [26] and ontology-based recovery [29], however the majority of traceability recovery publications apply IR techniques. Furthermore, it is well-known that IR is sensitive to the input data used in evaluations [4].

The primary aim of this study was to explore researchers' views on the comparability between NL artifacts produced by students and practitioners. We restricted the sample to authors with documented experience, i.e., published peer-reviewed research articles, of using either student or industrial artifact sets in IR-based traceability recovery studies. Consequently, we left out authors who exclusively used artifacts from the open source domain.

The questionnaire was constructed through a brainstorming session with the authors, using the literature review as input. To adapt the questions to the respondents regarding the origin of the artifacts used, three versions of the questionnaire were created:

- **STUD**. A version for authors of published studies on traceability recovery using student artifacts. This version was most comprehensive since it contained more questions. Thus it was sent to authors, if at least one publication using student artifacts had been identified.
- **UNIV**. A version for authors using artifacts originating from university projects. This version included a clarifying question on whether the artifacts were developed by students or not, followed by the same detailed questions about student artifacts as in version STUD. This question was used to filter out answers related to student artifacts.
- **IND**. A subset of STUD, sent to authors who only had published traceability recovery studies using industrial artifacts.

We piloted the questionnaire using five senior software engineering researchers, including a native English speaker. The three versions of the questionnaire were then refined, the final versions are presented in the Appendix. The mapping between research questions and questions in the questionnaire is presented in Table 3.

Research questions	Questionnaire questions
RQ1	QQ1, *QQ4*, QQ6
RQ2	*QQ4*, QQ5
RQ3	QQ2
RQ4	QQ3
RQ5	*QQ4*, QQ7

Table 3. Mapping between research questions and the questionnaire. QQ4 was used as a filter.

B. *Survey execution and analysis*

The questionnaires were distributed via email, sent to the set of authors described in Section III.A. As Figure 2 depicts, in total 90 authors were identified. We were able to send emails that appeared to reach 75 (83%) of them. Several mails returned with no found recipient and in some cases no contact information was available. In those few cases we tried contacting current colleagues; nevertheless there remained 15 authors (17%) we did not manage to send emails successfully. The mails were sent between September 27 and October 12, 2011. After one week, reminders were sent to respondents who had not yet answered the survey.

24 authors (32%) responded to our emails; however four responses did not contain answers to the survey questions. Among them, two academics referred to other colleagues more suitable to answer the questionnaire (all however already included in our sample) and two practitioners claimed to be too disconnected from research to be able to answer with a reasonable effort. Thus, the final set of complete answers included 20 returned questionnaires. This yielded a response rate of 27%.

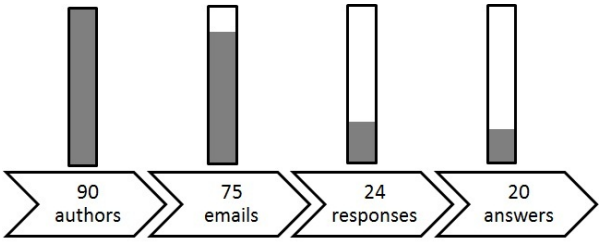

Figure 2. Survey response rate.

The survey answers were analyzed by descriptive statistics and qualitative categorization of the answers. The results and the analysis are presented in Section IV.

IV. Results and Analysis

In this section, the results from the survey of authors are presented and discussed.

A. *Demographics and sample characterization*

For the 20 authors of publications on IR-based traceability recovery who answered the questionnaire, Figure 3 depicts the distribution of practitioners and academics based on current main affiliation. 40% of the respondents are currently working in industry. Our respondents represent all combinations of academics and practitioners from Europe, North America and Asia. Table 4 presents how many answers we received per questionnaire version. Both respondents answering UNIV reported that students had developed the artifacts in their university projects (QQ4), thus at least twelve of the respondents had experience with student artifacts in traceability recovery experiments.

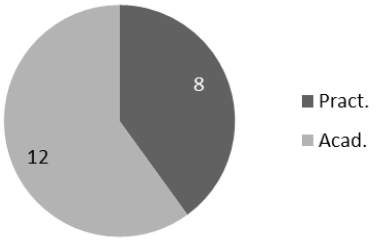

Figure 3. Current main affiliation of respondents.

	STUD	UNIV	IND
Academics	8	2	2
Practitioners	3	0	5
Total	11	2	7

Table 4. Respondents, organized by questionnaire versions and current main affiliation.

B. *Representativeness of student artifacts (RQ1)*

In this subsection, we present the view of our respondents on the representativeness of software artifacts produced by students. In this case, we investigated if our respondents agree with the statement that software artifacts produced by students are representative of software artifacts produced in industry (see QQ1, and QQ6 filtered by QQ4 in the Appendix). QQ6 overlaps QQ1 by targeting specific publications. To preserve the anonymity of the respondents, the analyses of the questions are reported together.

Figure 4 shows survey answers to the statement "Software artifacts produced by students (used as input in traceability experiments) are representative of software artifacts produced in industry" (QQ1). Black color represents answers from practitioners, grey color answers from academics. Half of the respondents fully or partly disagree to the statement. Academics answered this question with a higher degree of consensus than practitioners. No respondent totally agreed to the statement.

Several respondents decided to comment on the comparability of student artifacts. Two of them, both practitioners answering QQ1 with '4', pointed out that trained students actually might produce NL artifacts of higher quality than engineers in industry. One of them clarified: "In industry, there are a lot of untrained `professionals` who, due to many reasons including time constraints, produce `flawed` artifacts". Another respondent answered QQ1 with '2', but stressed that certain student projects could be comparable to industrial counterparts, for instance in the domain of web applications. On the other hand, he explained, would they not at all be comparable for domains with many process requirements such as finance and aviation. Finally, one respondent mentioned the wider diversity of industrial

artifacts, compared to artifacts produced by students: “I’ve seen ridiculously short requirements in industry (5 words only) and very long ones of multiple paragraphs. Students would be unlikely to create such monstrous requirements!” and also added “Student datasets are MUCH MUCH smaller (perhaps 50-100 artifacts compared to several thousands)”.

Three of our respondents mentioned the importance of understanding the incentives of the developers of the artifacts. This result confirms the findings by Höst *et al.* [11]. The scope and lifetime of student artifacts are likely to be much different for industrial counterparts. Another respondent (academic) supported this claim and also stressed the importance of the development context: “The vast majority [of student artifacts] are developed for pedagogical reasons – not for practical reasons. That is, the objective is not to build production code, but to teach students.” According to one respondent (practitioner), both incentives and development contexts are playing an important role also in industry: “Industrial artifacts are created and evolved in a tension between regulations, pressing schedule and lacking motivation /---/ some artifacts are created because mandated by regulations, but no one ever reads them again, other artifacts are part of contracts and are, therefore, carefully formulated and looked through by company lawyers etc.”

These results are not surprising, and lead to the conclusion that NL artifacts produced by students are understood to be less complex than their industrial counterparts. However, put in the light of related work outlined in Section 2, the results can lead to interesting interpretations. As presented in Figure 1, experiments on traceability recovery frequently use artifacts developed by students as input. Also, as presented in Table 1, two of four publicly available artifact sets at COEST originate from student projects. Nevertheless, our respondents mostly disagreed that these artifacts are representative of NL artifacts produced in industry.

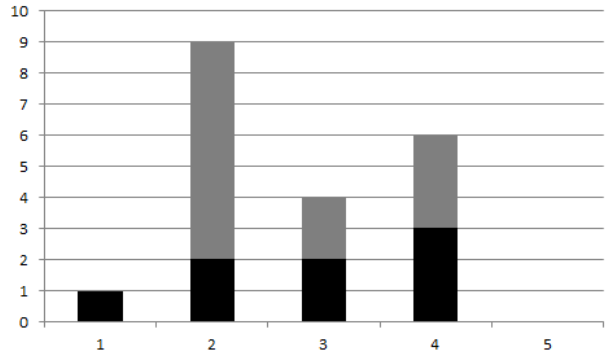

Figure 4. Are student artifacts representative to industrial counterparts? (1 = totally disagree, 5 = totally agree) (QQ1)

C. Validation of experimental artifacts (RQ2)

In this subsection, we present the results from QQ5 which is related to research question RQ2. QQ5, filtered by QQ4, investigates whether student artifacts, when used in traceability recovery experiments, were validated for industrial representativeness.

We received answers to QQ5 from 13 respondents (questionnaire versions STUD and UNIV). The distribution of answers is depicted in Figure 4. Among the five respondents who validated student artifacts being used as experimental input, three respondents focused on robustness of the experiment output (of the experiment in which the artifacts were used as input). The robustness was assessed by comparing experimental results to experiments using industrial artifacts. As another approach to validation, two respondents primarily used expert opinion to evaluate the industrial comparability of the student artifacts. Finally, three respondents answered that they did not conduct any explicit validation of the industrial comparability at all.

Neither answering ‘yes’ nor ‘no’ to QQ5, five respondents discussed the question in more general terms. Two of them stressed the importance of conducting traceability recovery experiments using realistic tasks. One respondent considered it significant to identify in which industrial scenario the student artifacts would be representative and said “The same student artifacts can be very `industrial´ if we think at a hi-tech startup company, and totally `unindustrial´ if we think at Boeing”. Another respondent claimed that they had focused on creating an as general tracing task as possible.

Only a minority of researchers who used student artifacts to evaluate IR-based traceability recovery explicitly answered with `yes´ to this question, suggesting that it is no widespread common practice. Considering the questionable comparability of artifacts produced by students, confirmed by QQ1, this finding is remarkable. Simply assuming that there is an industrial context where the artifacts would be representative might not be enough. The validation that actually takes place appears to be ad-hoc, thus some form of supporting guidelines would be helpful.

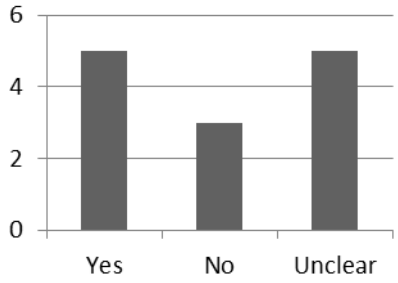

Figure 5. Were the student artifacts validated for industrial comparability? (QQ5)

D. Adequacy of artifact characterization (RQ3)

In this subsection, we present the results from asking our respondents whether the typical way of characterizing artifacts used in experiments (mainly size and number of correct traceability links) is sufficient. In Figure 6, we present answers to QQ2 which is related to RQ3. Black color represents practitioners, grey color academics. Two respondents (both academics) considered this to be a fully sufficient characterization. The distribution of the rest of the answers, both for practitioners and academics, shows mixed opinions.

Respondents answering with ‘1’ (totally insufficient) to QQ2 motivated their answers by claiming: simple link

existence being too rudimentary, complexity of artifact sets must be presented and the meaning of traceability links should be clarified. On the other hand, six respondents answered with '4' or '5' (5=fully sufficient). Their motivations included: tracing effort is most importantly proportional to the size of the artifact set and experiments based on textual similarities are reliable. However, two respondents answering with '4' also stated that information density and language are important factors and that the properties of the traceability links should not be missed.

More than 50% of all answers to QQ2 were marking options '1', '2' or '3'. Thus a majority of the respondents answering this question either disagree with the question statement or have a neutral opinion. This result is contrasting with published literature, in which we found that characterization of input artifacts in traceability experiments is generally brief (see Section II). There may be two possible explanations of this misalignment. Either the authors don't see the need of providing more descriptions of the used artifact sets (this may be the remaining minority of the answers), or the complementary metrics and important characterization factors are unknown. We believe that the result supports the second explanation as only limited work including explicit guidance has been published to date. Two respondents answered with '4' without motivating their choices. To conclude, since our review of literature found that the characterization of input artifacts in traceability experiments is generally brief (see Section II), this result justifies our research efforts and calls for further explanatory research.

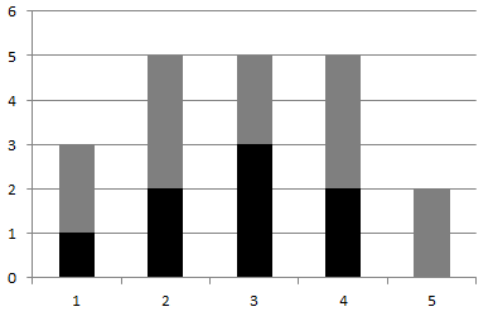

Figure 6. Is size and traceability link number sufficient to characterize an artifact set? (1 = totally insufficient, 5 = fully sufficient) (QQ2)

Our results indicate that authors are aware that there are other significant features of artifact sets than the typically reported size and total number of links (see also results in Section IV.E). Apparently, there seems to be a gap between what is considered a preferred characterization and what actually is reported in publications. The gap could have been partly mitigated if the research community to a higher extent had accepted "A framework for requirements tracing experiments", since it partly covers artifact set descriptions [12]. However, the results also indicate that parts of the research community think that the basic reporting is a good start.

E. Improved characterization (RQ4)

In this section we provide results for the RQ4, exploring ways to improve the way NL artifacts are reported, addressed by QQ3. Eleven respondents, six academics and five practitioners, suggested explicit enhancements to artifact set characterization, other respondents answered more vaguely. Those suggestions are collected and organized below into the three classes Contextual (describes the environment of the artifact development), Link-related (describes properties of traceability links) and Artifact-centric (describes the artifacts). In total, 23 aspects to additionally characterize artifacts were suggested.

Contextual aspects:

- Domain from which the artifacts originate
- Process used when artifact was developed (agile/spiral/waterfall etc., versioning, safety regulations)
- When in the product lifecycle the artifacts where developed
- Maturity/evolution of the artifacts (years in operation, #reviews, #updates)
- Role and personality of the developer of the artifacts
- Number of stakeholders/users of the artifacts
- Tasks that are related to the artifact set

Link-related aspects:

- Meaning of a traceability link (depends on, satisfies, based on, verifies etc.)
- Typical usage of traceability links
- Values and costs of traceability links (Value of correct link, cost of establishing link, establishing false link, missing link)
- Person who created the golden standard of links (practitioners, researchers, students, and their incentives)
- Quality of the golden standard of traceability links
- Link density
- Distribution of inbound/outbound links

Artifact-centric aspects:

- Size of individual artifacts
- Language (Natural, formal)
- Complexity of artifacts
- Verbosity of artifacts
- Artifact redundancy/overlap
- Artifact granularity (Full document/chapter/page/ section etc.)
- Quality/maturity of artifact (#defects reported, draft/reviewed/released)
- Structure/format of artifact (structured/semi-structured/unstructured information)
- Information density

As our survey shows, several authors have ideas about additional artifact set features that would be meaningful to report. Thus most authors both are of the opinion that

artifact sets should be better characterized, and also have suggestions for how it could be done. Still, despite also being stressed in Huffman Hayes and Dekhtyars framework from 2005, it has not reached the publications. However, we collected many requests for "what" to describe, but little input on the "how" (i.e. 'what' = state complexity / 'how' = how to measure complexity?). This discrepancy can be partly responsible for the insufficient artifact set characterizations. A collection of how different aspects might be measured, tailored for reporting artifact sets used in traceability recovery studies, appears to be a desirable composition.

One might argue that several of the suggested aspects are not applicable to student projects. This is in line with both what Höst *et al.* [11] and our respondents stated, purpose and lifecycle of student artifacts are rarely representative for industrial settings. Thus, aspects such as maturity, evolution and stakeholders usually are unfeasible to measure. Again, this indicates that artifacts originating from student projects might be too trivial, resulting in little more empirical evidence than proofs-of-concept.

F. Measuring student/industrial artifacts (RQ5)

In this section, we present results in relation to RQ5, concerning the respondents' opinions about how differences between NL artifacts developed by students and industrial practitioners can be assessed. QQ7, filtered by QQ4, provides answers to this question.

A majority of the respondents of STUD and UNIV commented on the challenge of measuring differences between artifacts originating from industrial and student projects. Only four respondents explicitly mentioned suitable aspects to investigate. Two of them suggested looking for differences in quality, such as maintainability, extensibility and ambiguities. One respondent stated that the main differences are related to complexity (students use more trivial terminology). On the other hand, one academic respondent instead claimed that "In fact artifact written by students are undoubtedly the most verbose and better argued since their evaluation certainly depends on the quality of the documentation". Yet another respondent, a practitioner, answered that the differences are minor.

Notably, one respondent to QQ7 warned about trying to measure differences among artifacts, motivated by the great diversity in industry. According to the respondent, there is no such thing as an average artifact. "What is commonly called `requirements´ in industry can easily be a 1-page business plan or a 15-volumes requirements specification of the International Space Station", the respondent explained.

To summarize, the results achieved for QQ7 confirm our expectations that measuring the comparability is indeed a challenging task. Obviously, there is no simple measure to aim for. This is also supported by QQ5, the few validations of student artifacts that the respondents reported utilized only expert opinion or replications with industrial artifacts.

V. THREATS TO VALIDITY

This section provides a discussion of the threats to validity in relation to research design and data collection phases as well as in relation to results from the study. The discussion of the threats to validity is based on the classification proposed by Wohlin *et al.* [28], focusing on threats to construct, internal and external validity.

Construct validity is concerned with the relation between the observations during the study and the theories in which the research is grounded. The exact formulations of the questions in the used questionnaire are crucial in survey research as misunderstanding or misinterpreting the questions can happen. We alleviated this threat to construct validity by revising the questionnaire by an independent reviewer (except the authors of the paper who also revised the questions) who is a native English speaker and writer. To further minimize threats to construct validity, a pilot study was conducted on five senior researchers in software engineering. Still, the subjectivity of the data provided by our respondents can negatively influence the interpretability of the results. Due to a relatively low number of data points, the mono-operational bias threat to construct validity is not fully addressed. The anonymity of respondents was not guaranteed as the survey was sent via email; this leaves the evaluation apprehension threat unaddressed. Since the research is exploratory, the experimenter expectancies threat to construct validity is minimized. Finally, the literature survey conducted as the first step of the study helps to address the mono-method threat to construct validity, which however still requires further research to fully alleviate it.

Internal validity concerns confounding factors that can affect the causal relationship between the treatment and the outcome. By performing the review of the questionnaire questions, the instrumentation threat to internal validity was addressed. On the other hand, the selection bias can still threaten the internal validity as the respondents were not randomly selected. We have measured the time needed to answer the survey in the pilot study; therefore the maturation threat to internal validity is alleviated. Finally, the selection threat to internal validity should be mentioned here since respondents of the survey were volunteers who, according to Wohlin *et al.*, are not representative for the whole populations [28].

External validity concerns the ability to generalize the results of the study to industrial practice. We have selected a survey research method in order to target more potential respondents from various countries, companies and research groups and possibly generate more results [8]. Still, the received number of responses is low and thus not a strong basis for extensive generalizations of our findings. However, the external validity of the results achieved is acceptable when considering the exploratory nature of this study.

VI. DISCUSSION AND CONCLUDING REMARKS

We have conducted an exploratory survey of the comparability of artifacts used in IR-based traceability recovery experiments, originating from industrial and student projects. Our sample of authors of related publications confirms that artifacts developed by students are only partially comparable to industrial counterparts. Nevertheless, it commonly happens that student artifacts

used as input to experimental research are not validated with regards to their industrial representativeness.

Our results show that, typically, artifact sets are only rudimentarily described, despite the experimental framework proposed by Huffman Hayes and Dekhtyar in 2005. We found that a majority of authors of traceability recovery publications think that artifact sets are inadequately characterized. Interestingly, a majority of the authors explicitly suggested features of artifact sets they would prefer to see reported. Suggestions include general aspects such as contextual information during artifact development and artifact-centric measures. Also, domain-specific (link-related) aspects were proposed, specifically applicable to traceability recovery.

This survey, acting as an explanatory study, should be followed by an in-depth study validating the proposals made by the respondents and aim at making the proposals more operational. This in turn could lead to characterization schemes that help assess the generalizability of study results using student artifacts. The results could complement Huffman Hayes and Dekhtyars framework [12] or be used as an empirical foundation of a future revision. Moreover, studies similar to this one should be conducted for other application domains where student artifacts frequently are used as input to experimental software engineering, such as regression testing, cost estimation and model-driven development.

Clearly, researchers need to be careful when designing traceability recovery studies. Previous research has shown that using students as experimental subjects is reasonable [2, 10, 11, 16, 27]. However, according to our survey, the validity of using student artifacts is uncertain. Unfortunately, industrial artifacts are hard to get access to. Furthermore, even with access to industrial artifacts, researchers might not be permitted to show them to students. And even with that permission, students might lack the domain knowledge necessary to be able to work with them. Figure 7 summarizes general risks involved in different combinations of subjects and artifacts in traceability recovery studies. The most realistic option, conducting studies on practitioners working with industrial artifacts, is unfortunately often hard to accomplish with a large enough number of subjects. Instead, several previous studies used students solving tasks involving industrial artifacts [3, 14] or artifacts developed in student projects [6, 7, 20]. However, these two experimental setups introduce threats either related to construct validity or external validity. The last option, conducting studies with practitioners working with student artifacts, has not been attempted. We plan to further explore the possible combinations in future work.

Experimental setup	Practitioners as subjects	Students as subjects
Industrial artifacts	Too expensive	Students do not understand
Student artifacts	Not explored	Uncertain generalizability

Figure 7. Risks involved in different combinations of subjects and artifacts in traceability recovery studies.

ACKNOWLEDGMENT

Thanks go to the respondents of the survey. This work was funded by the Industrial Excellence Center EASE - Embedded Applications Software Engineering[2]. Special thanks go to David Callele for excellent language-related comments.

[2] http://ease.cs.lth.se

REFERENCES

[1] G. Antoniol, G. Canfora, G. Casazza, and A. De Lucia, “Information retrieval models for recovering traceability links between code and documentation,” In Proceedings of the International Conference on Software Maintenance, pp. 40-49, 2000.

[2] P. Berander, “Using students as subjects in requirements prioritization,” In Proceedings of the 2004 International Symposium on Empirical Software Engineering, pp. 167-176, 2004.

[3] M. Borg, and D. Pfahl, “Do better IR tools improve the accuracy of engineers' traceability recovery?,” In Proceedings of the International Workshop on Machine Learning Technologies in Software Engineering, pp. 27-34, 2011.

[4] C. L. Borgman, “From Gutenberg to the global information - infrastructure access to information in the networked world,” Chapter 4, MIT Press, 2003.

[5] A. De Lucia, F._Fasano, R._Oliveto, and G._Tortora, “ADAMS Re-Trace: A traceability recovery tool,” In Proceedings_of the European Conference on Software Maintenance and Reengineering, pp. 23-41, 2005.

[6] A. De Lucia, F. Fasano, R. Oliveto, and G. Tortora, ”Recovering traceability links in software artifact management systems using information retrieval methods,” ACM Transactions on Software Engineering Methodology, 16(4):13, 2007.

[7] A. De Lucia, R. Oliveto, and G. Tortora, “Assessing IR-based traceability recovery tools through controlled experiments,” Empirical Software Engineering, 14(1), pp. 57-92, February 2009.

[8] S. Easterbrook, J. Singer, M-A. Storey, D. Damian, “Selecting empirical methods for software engineering research,” Chapter 11 in Shull et al., 2008.

[9] L. Freund, and E. G. Toms, “Enterprise search behaviour of software engineers,” In Proceedings of the 29th Annual international ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 645-656, 2006.

[10] M. Höst, B. Regnell, and C. Wohlin. “Using students as subjects - A comparative study of students and professionals in lead-time impact assessment,” Empirical Software Engineering, 5(3), pp. 201-214, November 2000.

[11] M. Höst, C. Wohlin, and T. Thelin. “Experimental context classification: incentives and experience of subjects,” In Proceedings of the 27th International Conference on Software Engineering, pp. 470-478, 2005.

[12] J. Huffman Hayes and A. Dekhtyar, “A framework for comparing requirements tracing experiments,” International Journal of Software Engineering and Knowledge Engineering, 15(5), pp. 751-781, October 2005.

[13] J. Huffman Hayes, A. Dekhtyar, and S. K. Sundaram, "Advancing candidate link generation for requirements tracing: the study of methods," IEEE Transactions on Software Engineering, 32(1), pp. 4- 19, January 2006.

[14] J. Huffman Hayes, A. Dekhtyar, S. Sundaram, E. Holbrook, S. Vadlamudi, and A. April, “REquirements TRacing on target

(RETRO): improving software maintenance through traceability recovery," Innovations in Systems and Software Engineering, 3(3), pp. 193–202, January 2007.

[15] A. Jedlitschka, M. Ciolkowski, and D. Pfahl, "Reporting guidelines for controlled experiments in software engineering," Chapter 8 in Shull et al., 2008.

[16] B. A. Kitchenham, S. L. Pfleeger, L. M. Pickard, P. W. Jones, D. C. Hoaglin, K. El Emam, and J. Rosenberg, "Preliminary guidelines for empirical research in software engineering," IEEE Transactions on Software Engineering, 28(8), pp. 721- 734, August 2002.

[17] L. Kuzniarz, M. Staron and C. Wohlin, "Students as study subjects in software engineering experimentation," In Proceedings of the 3rd Conference on Software Engineering Research and Practice in Sweden, pp. 19-24, 2003.

[18] M. Lormans, H-G. Gross, A. van Deursen, R. van Solingen, and A. Stehouwer, "Monitoring requirements coverage using reconstructed views: an industrial case study," 13th Working Conference on Reverse Engineering, pp. 275-284, 2006.

[19] A. Marcus, and J. I. Maletic, "Recovering documentation-to-source-code traceability links using latent semantic indexing," In Proceedings of the 25th International Conference on Software Engineering, pp. 125-135, 2003.

[20] J. Natt och Dag, T. Thelin, and B. Regnell, "An experiment on linguistic tool support for consolidation of requirements from multiple sources in market-driven product development," Empirical Software Engineering, 11(2), pp. 303-329, June 2006.

[21] R. Oliveto, M. Gethers, D. Poshyvanyk, and A. De Lucia, "On the equivalence of information retrieval methods for automated traceability link recovery," In Proceedings of the 2010 IEEE 18th International Conference on Program Comprehension, pp. 68-71, 2010.

[22] C. Robson, "Real world research," Blackwell, Oxford, 2002.

[23] P. Runeson, and M. Höst, "Guidelines for conducting and reporting case study research in software engineering,". Empirical Software Engineering, 14(2), pp. 131-164, April 2009.

[24] F. Shull, J. Singer, and D. Sjøberg, "Guide to advanced empirical software engineering," Springer, London, 2008.

[25] J. Singer, S. E. Sim, and T. C. Lethbridge, "Software engineering data collection for field studies," Chapter 1 in Shull et al., 2008.

[26] G. Spanoudakis, A. d'Avila-Garces, and A. Zisman, "Revising rules to capture requirements traceability relations: A machine learning approach," In Proceedings of the 15th International Conference in Software Engineering and Knowledge Engineering, pp. 570–577, 2003.

[27] M. Svahnberg, A. Aurum, and C. Wohlin. "Using students as subjects - an empirical evaluation," In Proceedings of the Second ACM-IEEE International Symposium on Empirical Software Engineering and Measurement, pp. 288-290, 2008.

[28] C. Wohlin, P. Runeson, M. Höst, M. C. Ohlsson, B. Regnell, and A. Wesslen, "Experimentation in software engineering - An introduction," Kluwer Academic Publishers, 2000.

[29] Y. Zhang, R. Witte, J. Rilling, and V. Haarslev, "Ontological approach for the semantic recovery of traceability links between software artefacts," IET Software, 2(3), pp. 185 –203, June 2008.

APPENDIX

The table below shows the final versions of the questions used in the three types of questionnaires of our survey. Furthermore, the email sent to respondents clarified that the study excluded source code from the scope and only considered natural language software artifacts.

	Questionnaire	**Used in versions**
QQ1	Would you agree with the statement: "Software artifacts produced by students (used as input in traceability experiments) are representative of software artifacts produced in industry?" (Please select one number. 1 = totally disagree, 5 = totally agree) 1---2---3---4---5	STUD / UNIV / IND
QQ2	Typically, datasets containing software artifacts used as input to traceability experiments are characterized by size and number of correct traceability links. Do you consider this characterization as sufficient? Please explain why you hold this opinion. (Please select one number. 1 = totally insufficient 5 = fully sufficient) 1---2---3---4---5	STUD / UNIV / IND
QQ3	What would be a desirable characterization of software artifacts to enable comparison (for example between software artifacts developed by students and industrial practitioners)?	STUD / UNIV / IND
QQ4	In your experiment, you used software artifacts developed in the university project [NAME OF PROJECT]. Were the software artifacts developed by students?	UNIV
QQ5	Did you evaluate whether the software artifacts used in your study were representative of industrial artifacts? If you did, how did you perform this evaluation?	STUD / UNIV
QQ6	How representative were the software artifacts you used in your experiment of industrial software artifacts? What was the same? What was different?	STUD / UNIV
QQ7	How would you measure the difference between software artifacts developed by students and software artifacts developed by industrial practitioners?	STUD / UNIV