# Empirical Software Engineering

# Report 4:

# Analyzing and Comparing Survey Studies

Group 12

191250123 孙浩峰(Presenter)

191250219 邹英龙

191250133 陶泽华

191250025 丁笑宇

# hw4

# 1.Study Summary

## 1.1 Overview

We collected totally 39 papers which utilize the methodology of survey, published in 2020 and including 6 from ESEM, 26 from EMSE and 7 from EASE. After our further study,  we discovered some former misclassification. We've  made corresponding adjustment taking our discovery into consideration, for instance, 9 papers from EASE were included originally but now 7 remains.

表1：ESEM

| ID | CITATION |
| --- | --- |
| 1 | Jannik Fischbach, Henning Femmer, Daniel Mendez, Davide Fucci, and Andreas Vogelsang. 2020. What Makes Agile Test Artifacts Useful? An Activity–Based Quality Model from a Practitioners' Perspective. In Proceedings of the 14th ACM / IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM) (ESEM '20). Association for Computing Machinery, New York, NY, USA, Article 41, 1—10. DOI:https://doi.org/10.1145/3382494.3421462 |
| 2 | Alex Serban, Koen van der Blom, Holger Hoos, and Joost Visser. 2020. Adoption and Effects of Software Engineering Best Practices in Machine Learning. In Proceedings of the 14th ACM / IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM) (ESEM '20). Association for Computing Machinery, New York, NY, USA, Article 3, 1—12. DOI:https://doi.org/10.1145/3382494.3410681 |
| 3 | Cecilia Apa, Martin Solari, Diego Vallespir, and Guilherme Horta Travassos. 2020. A Taste of the Software Industry Perception of Technical Debt and its Management in Uruguay: A survey in software industry. In Proceedings of the 14th ACM / IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM) (ESEM '20). Association for Computing Machinery, New York, NY, USA, Article 42, 1—9. DOI:https://doi.org/10.1145/3382494.3421463 |
| 4 | Edna Dias Canedo, Rodrigo Bonifácio, Márcio Vinicius Okimoto, Alexander Serebrenik, Gustavo Pinto, and Eduardo Monteiro. 2020. Work Practices and Perceptions from Women Core Developers in OSS Communities. In Proceedings of the 14th ACM / IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM) (ESEM '20). Association for Computing Machinery, New York, NY, USA, Article 26, 1—11. DOI:https://doi.org/10.1145/3382494.3410682 |
| 5 | Alex Serban, Koen van der Blom, Holger Hoos, and Joost Visser. 2020. Adoption and Effects of Software Engineering Best Practices in Machine Learning. In Proceedings of the 14th ACM / IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM) (ESEM '20). Association for Computing Machinery, New York, NY, USA, Article 3, 1—12. https://doi.org/10.1145/3382494.3410681 |
| 6 | Héctor Cadavid, Vasilios Andrikopoulos, Paris Avgeriou, and John Klein. 2020. A Survey on the Interplay between Software Engineering and Systems Engineering during SoS Architecting. In Proceedings of the 14th ACM / IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM) (ESEM '20). Association for Computing Machinery, New York, NY, USA, Article 2, 1—11. https://doi.org/10.1145/3382494.3410671 |

表2：EMSE

| ID | CITATION |
| --- | --- |

| | |
|---|---|
| 1 | Xu, B., An, L., Thung, F. *et al.* Why reinventing the wheels? An empirical study on library reuse and re–implementation. *Empir Software Eng* **25,** 755—789 (2020). https://doi.org/10.1007/s10664–019–09771–0 |
| 2 | Kotti, Z., Kravvaritis, K., Dritsa, K. *et al.* Standing on shoulders or feet? An extended study on the usage of the MSR data papers. *Empir Software Eng* **25,** 3288—3322 (2020). https://doi.org/10.1007/s10664–020–09834–7 |
| 3 | Gleirscher, M., Marmsoler, D. Formal methods in dependable systems engineering: a survey of professionals from Europe and North America. *Empir Software Eng* **25,** 4473—4546 (2020). https://doi.org/10.1007/s10664–020–09836–5 |
| 4 | Li, P.L., Ko, A.J. & Begel, A. What distinguishes great software engineers?. *Empir Software Eng* **25,** 322—352 (2020). https://doi.org/10.1007/s10664–019–09773–y |
| 5 | Rahman, A., Farhana, E. & Williams, L. *The 'as code' activities*: development anti–patterns for infrastructure as code. *Empir Software Eng* **25,** 3430—3467 (2020). https://doi.org/10.1007/s10664–020–09841–8 |
| 6 | Yates, R., Power, N. & Buckley, J. Characterizing the transfer of program comprehension in onboarding: an information–push perspective. *Empir Software Eng* **25,** 940—995 (2020). https://doi.org/10.1007/s10664–019–09741–6 |
| 7 | Abdalkareem, R., Oda, V., Mujahid, S. *et al.* On the impact of using trivial packages: an empirical case study on *npm* and *PyPI*. *Empir Software Eng* **25,** 1168—1204 (2020). https://doi.org/10.1007/s10664–019–09792–9 |
| 8 | Morales, R., Khomh, F. & Antoniol, G. RePOR: Mimicking humans on refactoring tasks. Are we there yet?. *Empir Software Eng* **25,** 2960—2996 (2020). https://doi.org/10.1007/s10664–020–09826–7 |
| 9 | Rios, N., Spínola, R.O., Mendonça, M. *et al.* The practitioners' point of view on the concept of technical debt and its causes and consequences: a design for a global family of industrial surveys and its first results from Brazil. *Empir Software Eng* **25,** 3216—3287 (2020). https://doi.org/10.1007/s10664–020–09832–9 |
| 10 | Zampetti, F., Vassallo, C., Panichella, S. *et al.* An empirical characterization of bad practices in continuous integration. *Empir Software Eng* **25,** 1095—1135 (2020). https://doi.org/10.1007/s10664–019–09785–8 |
| 11 | Salza, P., Palomba, F., Di Nucci, D. *et al.* Third–party libraries in mobile apps. *Empir Software Eng* **25,** 2341—2377 (2020). https://doi.org/10.1007/s10664–019–09754–1 |
| 12 | Amreen, S., Mockus, A., Zaretzki, R. *et al.* ALFAA: Active Learning Fingerprint based Anti–Aliasing for correcting developer identity errors in version control systems. *Empir Software Eng* **25,** 1136—1167 (2020). https://doi.org/10.1007/s10664–019–09786–7 |

| | |
|---|---|
| 13 | Vassallo, C., Panichella, S., Palomba, F. *et al.* How developers engage with static analysis tools in different contexts. *Empir Software Eng* **25,** 1419—1457 (2020). https://doi.org/10.1007/s10664–019–09750–5 |
| 14 | Ralph, P., Baltes, S., Adisaputri, G. *et al.* Pandemic programming. *Empir Software Eng* **25,** 4927—4961 (2020). https://doi.org/10.1007/s10664–020–09875–y |
| 15 | Panichella, S., Zaugg, N. An Empirical Investigation of Relevant Changes and Automation Needs in Modern Code Review. *Empir Software Eng* **25,** 4833—4872 (2020). https://doi.org/10.1007/s10664–020–09870–3 |
| 16 | Dey, T., Mockus, A. Deriving a usage–independent software quality metric. *Empir Software Eng* **25,** 1596—1641 (2020). https://doi.org/10.1007/s10664–019–09791–w |
| 17 | Guo, H., Kafalı, Ö., Jeukeng, AL. *et al.* ÇORBA: crowdsourcing to obtain requirements from regulations and breaches. *Empir Software Eng* **25,** 532—561 (2020). https://doi.org/10.1007/s10664–019–09753–2 |
| 18 | Brindescu, C., Ahmed, I., Jensen, C. *et al.* An empirical investigation into merge conflicts and their effect on software quality. *Empir Software Eng* **25,** 562—590 (2020). https://doi.org/10.1007/s10664–019–09735–4 |
| 19 | Biørn–Hansen, A., Rieger, C., Grønli, TM. *et al.* An empirical investigation of performance overhead in cross–platform mobile development frameworks. *Empir Software Eng* **25,** 2997—3040 (2020). https://doi.org/10.1007/s10664–020–09827–6 |
| 20 | Iung, A., Carbonell, J., Marchezan, L. *et al.* Systematic mapping study on domain–specific language development tools. *Empir Software Eng* **25,** 4205—4249 (2020). https://doi.org/10.1007/s10664–020–09872–1 |
| 21 | Jiarpakdee, J., Tantithamthavorn, C. & Treude, C. The impact of automated feature selection techniques on the interpretation of defect models. *Empir Software Eng* **25,** 3590—3638 (2020). https://doi.org/10.1007/s10664–020–09848–1 |
| 22 | Rousseau, G., Di Cosmo, R. & Zacchiroli, S. Software provenance tracking at the scale of public source code. *Empir Software Eng* **25,** 2930—2959 (2020). https://doi.org/10.1007/s10664–020–09828–5 |
| 23 | Bangash, A.A., Sahar, H., Hindle, A. *et al.* On the time–based conclusion stability of cross–project defect prediction models. *Empir Software Eng* **25,** 5047—5083 (2020). https://doi.org/10.1007/s10664–020–09878–9 |
| 24 | Arya, D.M., Guo, J.L.C. & Robillard, M.P. Information correspondence between types of documentation for APIs. *Empir Software Eng* **25,** 4069—4096 (2020). https://doi.org/10.1007/s10664–020–09857–0 |
| 25 | Ranganath, VP., Mitra, J. Are free Android app security analysis tools effective in detecting known vulnerabilities?. *Empir Software Eng* **25,** 178—219 (2020). https://doi.org/10.1007/s10664–019–09749–y |

| | |
|---|---|
| 26 | Yao, K., Li, H., Shang, W. *et al.* A study of the performance of general compressors on log files. *Empir Software Eng* **25,** 3043—3085 (2020). https://doi.org/10.1007/s10664-020-09822-x |

表3：EASE

| ID | CITATION |
|---|---|
| 1 | Orges Cico, Anh Nguyen Duc, and Letizia Jaccheri. 2020. An Empirical Investigation on Software Practices in Growth Phase Startups. In Proceedings of the Evaluation and Assessment in Software Engineering (EASE '20). Association for Computing Machinery, New York, NY, USA, 282—287. https://doi.org/10.1145/3383219.3383249 |
| 2 | Nitish Patkar, Mohammad Ghafari, Oscar Nierstrasz, and Sofija Hotomski. 2020. Caveats in Eliciting Mobile App Requirements. In Proceedings of the Evaluation and Assessment in Software Engineering (EASE '20). Association for Computing Machinery, New York, NY, USA, 180—189. https://doi.org/10.1145/3383219.3383238 |
| 3 | Francisco Dalton, Márcio Ribeiro, Gustavo Pinto, Leo Fernandes, Rohit Gheyi, and Baldoino Fonseca. 2020. Is Exceptional Behavior Testing an Exception? An Empirical Assessment Using Java Automated Tests. In Proceedings of the Evaluation and Assessment in Software Engineering (EASE '20). Association for Computing Machinery, New York, NY, USA, 170—179. https://doi.org/10.1145/3383219.3383237 |
| 4 | Philipp Tschannen and Ali Ahmed. 2020. On the Evaluation of the Security Usability of Bitcoin's APIs. In Proceedings of the Evaluation and Assessment in Software Engineering (EASE '20). Association for Computing Machinery, New York, NY, USA, 405—412. https://doi.org/10.1145/3383219.3383277 |
| 5 | Rolando P. Reyes, Oscar Dieste, Efraín R. Fonseca C., and Natalia Juristo. 2020. Publication Bias: A Detailed Analysis of Experiments Published in ESEM. In Proceedings of the Evaluation and Assessment in Software Engineering (EASE '20). Association for Computing Machinery, New York, NY, USA, 130—139. https://doi.org/10.1145/3383219.3383233 |
| 6 | Sávio Freire, Nicolli Rios, Boris Gutierrez, Darío Torres, Manoel Mendonça, Clemente Izurieta, Carolyn Seaman, and Rodrigo O. Spínola. 2020. Surveying Software Practitioners on Technical Debt Payment Practices and Reasons for not Paying off Debt Items. In Proceedings of the Evaluation and Assessment in Software Engineering (EASE '20). Association for Computing Machinery, New York, NY, USA, 210—219. https://doi.org/10.1145/3383219.3383241 |
| 7 | Elias Brattli Sørensen, Edvard Kristoffer Karlsen, and Jingyue Li. 2020. What Norwegian Developers Want and Need From Security-Directed Program Analysis Tools: A Survey. In Proceedings of the Evaluation and Assessment in Software Engineering (EASE '20). Association for Computing Machinery, New York, NY, USA, 505—511. https://doi.org/10.1145/3383219.3383293 |

## 1.2 Our target papers

We choose 5 papers for further analysis and comparison.

### 1.2.1 Adoption and Effects of Software Engineering Best Practices in Machine Learning(ESEM)

BackGround: The increasing reliance on applications with machine learning (ML) components calls for mature engineering techniques that ensure these are built in a robust and future–proof manner.

Aim: The researchers aim to empirically determine the state of the art in how teams develop, deploy and maintain software with ML components.

Method: The researchers mined both academic and grey literature and identified 29 engineering best practices for ML applications. They conducted a survey among 313 practitioners to determine the degree of adoption for these practices and to validate their perceived effects. They also tested correlations and investigated linear and non–linear relationships between practices and their perceived effect using various statistical models.

Result: The researchers' findings indicate, for example, that larger teams tend to adopt more practices, and that traditional software engineering practices tend to have lower adoption than ML specific practices. Also, the statistical models can accurately predict perceived effects such as agility and traceability, from the degree of adoption for specific sets of practices. Combining practice adoption rates with practice importance, researchers identify practices that are important but have low adoption, as well as practices that are widely adopted but are less important for the effects they studied.

Conclusion: Their survey and the analysis of responses received provide a quantitative basis for assessment and step–wise improvement of practice adoption by ML teams.

### 1.2.2 A Survey on the Interplay between Software Engineering and Systems Engineering during SoS Architecting(ESEM)

BackGround: The Systems Engineering and Software Engineering disciplines are highly intertwined in most modern Systems of Systems (SoS), and particularly so in industries such as defense, transportation, energy and health care. However, the combination of these disciplines during the architecting of SoS seems to be especially challenging; the literature suggests that major integration and operational issues are often linked to ambiguities and gaps between system–level and software–level architectures.

Aim: This paper aims to empirically investigate: 1) the state of practice on the interplay between these two disciplines in the architecting process of systems with SoS characteristics; 2) the problems perceived due to this

interplay during said architecting process; and 3) the problems arising due to the particular characteristics of SoS systems.

Method: Researchers conducted a questionnaire–based online survey among practitioners from industries in the aforementioned domains. The survey combined multiple–choice and open–ended questions, and the data collected from the 60 respondents were analyzed using quantitative and qualitative methods.

Results: 1) Although mostly the software architecting process is governed by system–level requirements, the way requirements specified by systems engineers and the lack of domain–knowledge of software engineers often lead to misinter–pretations at software level. 2) Unclear and/or incomplete specifications could be a common cause of technical debt in SoS projects, which is caused by insufficient interface definitions. 3) While not directly related to the interplay of the two disciplines, the survey also indicates that low–level hardware components are often not considered when modeling or simulating the system.

Conclusion: The survey indicates the need for tighter collaboration between the two disciplines, structured around concrete guidelines and practices for reconciling their differences. A number of open issues identified by this study require further investigation.

### 1.2.3 Formal methods in dependable systems engineering: a survey of professionals from Europe and North America（EMSE）

BackGround: Formal methods (FMs) have been around for a while, still being unclear how to leverage their benefits, overcome their challenges, and set new directions for their improvement towards a more successful transfer into practice.

Aim: We study the use of formal methods in mission–critical software domains, examining industrial and academic views.

Method: We perform a cross–sectional on–line survey.

Results: Our results indicate an increased intent to apply FMs in industry, suggesting a positively perceived usefulness. But the results also indicate a negatively perceived ease of use. Scalability, skills, and education seem to be among the key challenges to support this intent.

Conclusion: We present the largest study of this kind so far (N = 216), and our observations provide valuable insights, highlighting directions for future theoretical and empirical research of formal methods. Our findings are strongly coherent with earlier observations by Austin and Graeme.

### 1.2.4 A Taste of the Software Industry Perception of Technical Debt and its Management in Uruguay: A survey in software industry

Background: Technical debt (TD) has been an important focus of attention in recent years by the scientific community and the software industry. TD is a concept for expressing the lack of internal software quality that directly affects its capacity to evolve. Some studies have focused on the TD industry perspective.

Aims: To characterize how the software industry professionals in Uruguay understand, perceive, and adopt technical debt management (TDM) activities.

Method: To replicate a Brazilian survey with the Uruguayan software industry and compare their findings.

Results: From 259 respondents, many indicated any awareness of the TD concept due to the faced difficult to realize how to associate such a concept with actual software issues. Therefore, it is possible to observe a considerable variability in the importance of TDM among the respondents. However, a small part of the respondents declares to carry out TDM activities in their organizations. A list of software technologies declared as used by practitioners was produced and can be useful to support TDM activities.

Conclusions: The TD concept and its management are not common yet in Uruguay. There are indications of TD unawareness and difficulties in the conduction of some TDM activities considered as very important by the practitioners. There is a need for more effort aiming to disseminate the TD knowledge and to provide software technologies to support the adoption of TDM in Uruguay. It is likely other software engineering communities face similar issues. Therefore, further investigations in these communities can be of interest.

## 1.2.5 The practitioners' point of view on the concept of technical debt and its causes and consequences: a design for a global family of industrial surveys and its first results from Brazil（EMSE）

Background: Studying the causes of technical debt (TD) could aid in TD prevention, thus easing the job of TD management. On the other hand, better understanding of the effects of TD could also aid in TD management by facilitating more informed decisions about incurring and paying off debt.

Aims: Create a deeper understanding, and confirming existing evidence, of the causes and effects of TD by collecting new evidence from real–world TD examples.

Method: InsighTD is a globally distributed family of industrial surveys on the causes and effects of TD. It is designed to run as a large–scale study based on continuous and independent replications in different countries. The survey instrument asks practitioners to describe in detail a real example of TD from their experience. We present in this paper the design of InsighTD, which has the primary goal of replication at a large–scale, with the results of the study in Brazil as a small part of the larger puzzle.

Results: The first iteration of the InsighTD survey, carried out in Brazil, yielded 107 responses. We identified a total of 78 causes and 66 effects, which confirm and also extend the current knowledge on causes and effects of TD. Then, we organized the identified set of causes and effects in probabilistic cause–effect diagrams. The proposed

diagrams highlight the causes that can most contribute to the occurrence of TD as well as the most common effects that occur as a result of debt.

Conclusions: We intend to reduce the problem of isolated TD investigations that are not yet representative and build a continuous and generalizable empirical basis for understanding practical problems and challenges of TD.

## 1.3 Reasons

1. Questionnaire–based research should be used as the main research method. Therefore, it is convenient for more fine–grained classification and research.
2. The papers should have high integrity of key elements. We focus on whether the paper includes instrument evaluation, because this step is an important guarantee for the quality of the questionnaire.
3. High quality and convincing conclusions should be put forward, and the corresponding survey results should be used as data support.

## 1.4 Problems and gained experiences

### 1.4.1 Problems

1. There was no detailed description of the subject details, which greatly hindered the information collection.
2. We can't discover clear response rate in papers which is supposed to given.
3. Some papers take pilot study as a key measure to ensure construction validity while the other articles don't do so. We question the appropriateness of this approach

### 1.4.2 Solutions

1. If question list not given, we should read the relevant parts of the paper carefully and look for descriptions and information that suggest details, purpose of the questions and possible responses for instance.
2. Total number of respondents may be given. If not, we should summarize the all the information, mentioned in the paper, about specific situation of questionnaire recovery.
3. After in–depth study, we found out that only a few papers consider pilot study as a key measure for construction validity. Thus we consider it an unnecessary method which can be omitted.

# 2.Study Analysis and Comparison

## 2.1 Study Analysis

| CHARACTERISTICS | **Adoption and Effects of Software Engineering Best Practices in Machine Learning(ESEM)** |
|---|---|
| Type | cross–sectional and questionaire–based |
| Objective | The survey was designed to measure the adoption of practices and also to assess the effects of adopting specific sets of practices. |
| Format | self–administrated questionnaire(web based)<br>logic order of questions |
| Questions | 5 closed questions for preliminaries + 31 closed questions for section 1 + 9 open questions for section 2 |
| Type of Answer | yes/no answers (not sure) + textual answers + agreement scales on Likert Scale |
| Instrument Evaluation | Pilot studies: Before distributing the survey, five practitioners with diverse backgrounds were interviewed in order to check if any information from the survey was redundant or whether important practices were missing. |
| Response Rate | Totally 350 valid responses, 313 complete responses in particular. As the total of respondents remains unknown, accurate response rate can't be calculated. |
| Sampling | Snowball strategy + Convenience sampling |
| Length | Although the questionaire contains 45 questions in total, the questions themselves are uncomplicated that the average completion time is 7 minutes. Thus, the length of instrument can be considered appropriate to some extent. |
| Study | quantitative analysis + descriptive statistics |

| Validities | Adoption and Effects of Software Engineering Best Practices in Machine Learning(ESEM) |
|---|---|
| Internal Validity | The data extracted from literature may be subject to bias. To limit this bias, several authors with different backgrounds have been involved in the extraction process. In the future, the researchers intend to further test completeness and soundness of catalogue of practices through participant validation interviews. |
| External Validity | The survey answers may be subject to bias. Although the adoption rates for respondents in Europe do not present striking differences when compared to those in South America or Asia, Europe remains over–represented. This bias can be removed by gathering more data. |
| Criterion Validity | The measurements used to investigate the relationship between groups of practices and their intended effects may be subject to bias. Rather than measurements of actual effects, we used the perceived effects as evaluated by the survey respondents. |

| CHARACTERISTICS | **A Survey on the Interplay between Software Engineering and Systems Engineering during SoS Architecting(ESEM)** |
|---|---|
| Type | cross–sectional and questionaire–based |
| Objective | The goal of this study is to identify the problems related to the interplay between SE and SWE disciplines during SoS architecting in practice. |
| Format | self–administrated questionnaire(web based +email based)<br>logic order of questions |
| Questions | closed questions + open questions |
| Type of Answer | numeric values + textual answers + response categories(multiple–choice) |
| Instrument Evaluation | The researchers piloted two consecutive versions of the survey, each with a different set of respondents, for a total of 7 respondents. The pilot survey respondents provided key feedback on the wording and the consistency of the questions. |
| Response Rate | A total of 76 responses were collected. 16 responses were excluded, among which 15 responses were incomplete and 1 response was from the target population. |
| Sampling | Convenience sampling + Snowball strategy |
| Length | The questionaire contains totally 24 questions |
| Study | quantitative analysis + qualitative analysis + descriptive statistics |

| Validities | A Survey on the Interplay between Software Engineering and Systems Engineering during SoS Architecting(ESEM) |
|---|---|
| External Validity | The non–probabilistic sampling design used for data collection is a potential threat for the external validity of the study.To mitigate this threat, the survey was distributed not only through the personal networks of the authors, but also through organizations and social media platforms. |
| Cosntruct Validity | Construct validity refers to the degree to which the operational measures, in this case the online survey, reflect what the researchers have in mind and are consistent with the research questions. To improve the validity of the study in this respect, researchers piloted two consecutive versions of the survey. Besides, the authors have iteratively refined the study design. |

| CHARACTERISTICS | **Formal methods in dependable systems engineering: a survey of professionals from Europe and North America（EMSE）** |
|---|---|
| Type | a cross–sectional on–line survey |
| Objective | The survey studies the use of formal methods in mission–critical software domains, examining industrial and academic views. |
| Format | self–administrated questionnaire(web based) |
| Questions | half–open + open questions |
| Type of Answer | yes/no answers + textual answers + frequency scales |
| Instrument Evaluation | Pilot studies: Before distributing the survey, we derived answer options from the literature, our own experience with FMs, SE research training, discussions with other SE researchers and colleagues from industry, pilot responses, and coding of open answers. |
| Response Rate | Given a recent estimate of worldwide 23 million SE practitioners and assuming that at least 1% are mission–critical SE practitioners, our population might comprise at least 230K persons, possibly around 38K in the US and 61K in Europe.Footnote6 We received N = 216 responses resulting in an estimated response rate between 1 and 2% and a population coverage of at most 0.1% globally and 0.2% in the US and in Europe. |
| Sampling | snowball sampling |
| Length | The questionaire contains totally 13 questions. |
| Study | qualitative analysis + descriptive statistics |

| Validities | Formal methods in dependable systems engineering: a survey of professionals from Europe and North America（EMSE） |
|---|---|
| Face and Content Validity | We derived answer options from the literature, our own experience with FMs, SE research training, discussions with other SE researchers and colleagues from industry, pilot responses, and coding of open answers. Particularly, the classification of FM methods and the list of obstacles or challenges were derived from our own training, literature knowledge prior to this study, and experience as well as from occasional personal discussions with SE experts from academia and industry. Most questions are half–open, allowing respondents to go beyond given answer options. We treat degree and relative frequency as 3–level Likert–type scales. |
| Construct Validity | Bias by omitted scale values/Respondents are encouraged to provide open answers to all questions, helping us to check scale completeness. Between 8% and 40% of the respondents made use of the text field “Other.” Our systematic map confirms that we have not listed unknown challenges in QR13. We identified three additional challenges via open answers and the literature. We believe to have achieved good criterion validity through questions and scales for distinguishing important sub–groups of our population. |
| Internal Validity | Google Forms includes data points only if all mandatory questions are answered and the submit button is pressed. We also performed a consistency check of MC questions and corrected 5 data points where “I do/have not…” was combined with other checked options. |
| External Validity | Lack of FM knowledge / 11 to 18% of our respondents did not know specific challenges. For RQ1, dnk–data points have no influence because the findings of RQ1 directly describe and interpret the status quo of UFMp. For test purposes, we included dnk–data points in the analyses of RQ2 and RQ3, with no relevant influence. |
| Reliability | Why would a repetition of the procedure in Section 4 with different samples from the same population lead to the same results? Internal consistency / All 7 items for the concept “obstacle to c show good internal consistency for our sample with a Cronbach $\alpha = 0.84$, the PEOU–part of C7 consisting of 5 items shows an $\alpha = 0.79$ The other concepts are not measured with multiple items. ◦ |

| CHARACTERISTICS | A Taste of the Software Industry Perception of Technical Debt and its Management in Uruguay: A survey in software industry(ESEM) |
|---|---|
| Type | Questionnaire–based survey |
| Objective | This study's objective is to understand and characterize how the software professionals in Uruguay understand, perceive, and manage TD, as well as the level of adoption of TDM technologies, using the engaged practitioners as proxies. |
| Format | self–administrated questionnaire(web based) |
| Questions | half–open + open questions |
| Type of Answer | response categories + textual answers + frequency scales |
| Instrument Evaluation | A pilot trial was performed in January 2019 after reviewing all the materials by the Brazilian and Uruguayan researchers. It involved four software engineering researchers from IS.uy Program with a strong relation with software industry. |
| Response Rate | The level of participation in the survey can be considered high. We assume that it was due to the effort invested in the dissemination of the study. The strong cooperation between industry and academy in Uruguay and the multiple personal contacts in the industry that the researchers have helped to strengthen this response rate. |
| Sampling | convenient sampling |
| Length | The questionaire contains totally 14 questions. |
| Study | qualitative analysis |

| Validities | A Taste of the Software Industry Perception of Technical Debt and its Management in Uruguay: A survey in software industry(ESEM) |
|---|---|
| External Validity | Regarding the generalization of the results that affect the external validity, we had a relatively high response rate, according to the Uruguayan population (3.5 million), when compared with the original study and with other related works. The participants' characterization presented in section 4.1 revealed a high level of diversity concerning the participants' role, the size, type, and activity area of the participants' organization. |
| Internal & Construct Validity | Regarding the participants' bias and the instrumentation that affect the internal and the construct validity; the participants might have misunderstood some terms and concepts presented in the questionnaire, based on their different experiences and knowledge. |
| Conclusion Validity | To mitigate the research bias in the data analysis that affects the conclusion validity, we minimized the amount of open questions in the questionnaire to avoid the subjective data interpretation. |

| CHARACTERISTICS | **The practitioners' point of view on the concept of technical debt and its causes and consequences: a design for a global family of industrial surveys and its first results from Brazil（EMSE）** |
|---|---|
| Type | Questionnaire–based survey |
| Objective | The study's objective is to create a deeper understanding, and confirm existing evidence, of the causes and effects of TD by collecting new evidence from real–world TD examples. |
| Format | self–administrated questionnaires(web–based) |
| Questions | closed and open questions |
| Type of Answer | closed questions + open questions |
| Instrument Evaluation | The basic coding and analysis of causes and effects into categories has been, and is expected to continue to be, done similarly in all InsighTD replications. A standard, basic, well–defined set of analysis procedures helps to reduce the possibility of bias, and to ensure consistency and comparability of results. However, replication teams are free to develop and employ other analysis strategies to investigate other questions and issues as they arise. |
| Response Rate | The survey in Brazil was online from December 7th, 2017, until January 7th, 2018. In total, we sent the survey invitation to about 513 professionals and 112 of them completed the full questionnaire. This represents an approximate response rate of 22%. |
| Sampling | convenient sampling |
| Length | The questionaire contains totally 14 questions. |
| Study | qualitative analysis |

| Validities | **The practitioners' point of view on the concept of technical debt and its causes and consequences: a design for a global family of industrial surveys and its first results from Brazil（EMSE）** |
|---|---|
| Internal Validity | Maturation and instrumentation are two threats to internal validity that could affect this study. Instrumentation is the effect caused by the artifacts used for the study execution, in our case, the questionnaire. If it is badly designed, the study results are affected negatively. To deal with this threat, the questionnaire was designed in a way that we have only direct questions and, thus, requiring as little interpretation as possible, avoiding a misunderstanding that would lead to meaningless answers. |
| Construct Validity | In this study, some social threats to construct validity can arise. These threats are concerned with issues related to behavior of the participants and the experimenters. Overall, participants may, based on the fact that they are part of a study, act differently than they do otherwise. To help prevent hypothesis guessing and evaluation apprehension, in the invitation e–mail, we clearly explain the purpose of the study and ask the participants to answer questions based on their own experience. |
| Conclusion Validity | This aspect is concerned with to what extent the data and the analysis are independent of the specific researchers. In InsighTD, the major threat to conclusion validity arises from the coding process as coding is essentially a creative task. To mitigate this threat, we first conducted a pilot phase in the analysis. After agreeing on the first resulting codes, the coding process was performed individually by two researchers. Then, they discussed the results until they reached consensus. |
| External Validity | Threats to external validity are conditions that limit our ability to generalize the results. We reduce this threat by achieving a diversity of participants who answered the survey. We also cannot say that our results represent the other countries in the InsighTD project. On the contrary, we expect partially different results in different countries. Thus, we need to follow our design of a family of surveys, conducting continuous replications in different countries and synthesizing the results to reach a more reliable and empirically founded result. |

## 2.2 Study Comparison

1. All of the five studies obtain survey data by conducting web–based questionnaires. Besides, sampling methods utilized by researchers are quite similar, all of them are non–probabilistic sampling designs ranging from

snowball sampling to convenient sampling.

2. Question types varies. The questionnaires designed in the three selected studies contain open questions and closed questions, while the others contains open questions and half–open questions only.
3. All of the selected studies carried out instrument evaluation. It seems pilot trail (or pilot study) a popular instrument evaluation method that nearly all the selected studys use this approach.
4. From the perspective of the number of topics, most of the studies has proper length of qurstionaire, which range from ten to twenty questions.
5. Type of listed validities varies among selected studies. Some of them covers all types of validities while some of the validities are missed in the other studies.
6. Study1.2.1 and Study1.2.2 both adopt combination of open questions and closed questions. In Study1.2.1, closed questions account for up to 80%. While in Study1.2.2, The proportion of open questions and closed questions is much closer compared to Paper1. Besides, most of the closed questions are multiple–choice ones.
7. Study1.2.1 and Study1.2.2 both utilize snowball strategy along with convenience sampling like all the other studies. Study1.2.1 oriented to the members of teams using ML components with diverse backgrounds and divide them into five categories according to the team level. Howerver, Study1.2.2 only invited experts with sufficient background knowledge and develop experience.
8. Quantitative analysis and descriptive statistics are both applied in Study1.2.1 and Study1.2.2. Qualitative analysis are not carried out for questionnaire survey in Study1.2.1, because it has been completed in previous work. However, Study1.2.2 carry out qualitative analysis for questionnaire survey.