Mapping software testing practice with software testing research—SERP-test taxonomy

Engström, Emelie; Peterson, Kai

Published in:
International conference on Software Testing, Verification and Validation Workshops

2015

Mapping software testing practice with software testing research - SERP-test taxonomy

Emelie Engström*, Kai Petersen†
* Dept. of Computer Science, Lund University, Sweden
emelie.engstrom@cs.lth.se
† School of Computing, Blekinge Institute of Technology, Sweden
kai.petersen@bth.se

***Abstract*—There is a gap between software testing research and practice. One reason is the discrepancy between how testing research is reported and how testing challenges are perceived in industry. We propose the SERP-test taxonomy to structure information on testing interventions and practical testing challenges from a common perspective and thus bridge the communication gap. To develop the taxonomy we follow a systematic incremental approach. The SERP-test taxonomy may be used by both researchers and practitioners to classify and search for testing challenges or interventions. The SERP-test taxonomy also supports comparison of testing interventions by providing an instrument for assessing the distance between them and thus identify relevant points of comparisons.**

I. Introduction

Despite extensive research on testing in the research community, gained knowledge have not had much effect on industry practices. We believe that one aggravating factor for the discrepancy between theory and practice within this field is that researchers and practitioners approach testing challenges very differently: at different abstraction levels, from different perspectives and with different objectives.

As a response to this challenge we propose the SERP-test taxonomy which primarily has a problem focus on software testing. The aim is to support communication and evaluation of testing research from a pragmatic perspective.

Many classifications of testing research have been proposed, either explicitly e.g. unit testing classifications [1] or implicitly through systematic reviews [2]. However, it is a widely recognized challenge that practitioners and researchers use different terminologies, making it challenging to map practical challenges to research solutions [3]. Thus we believe it is necessary to structure knowledge from other points of views than what has traditionally been made. In most cases existing classification schemes have a solution focus, i.e. they classify testing based on characteristics of the interventions rather than e.g. the objective of applying them.

To construct SERP-test we followed a systematic approach inspired by previous efforts of taxonomy construction in software engineering (cf. [4], [5], [6]).

2015 IEEE Eighth International Conference on Software Testing, Verification and Validation Workshops (ICSTW)
10th Testing: Academic and Industrial Conference - Practice and Research Techniques (TAIC PART)
978-1-4799-1885-0/15/$31.00 ©2015 IEEE

The goal of our taxonomy construction is threefold: (1) Support communication between researchers and practitioners by enabling common classification of interventions and practical challenges; (2) Increase accessibility of research results by supporting the search for matches between interventions and practical challenges in software testing; (3) Support study design and comparison of interventions from a pragmatic perspective by providing a tool for identifying relevant points of comparison.

To ensure our resulting taxonomy will serve its purpose and be useful for both practitioners and researchers both perspectives have been considered throughout the development process. We combined a top-down and bottom-up approach. While our initial definition of the taxonomy was based on some commonly accepted high-level classifications from literature we have been restrictive in adding detailed classifications merely based on literature. Instead we expect the taxonomy to evolve incrementally as it is used.

II. SERP-test taxonomy

A. Interventions and Challenges

Our starting point was to find the common denominators for classifying software testing interventions and challenges. In line with taxonomies from other domains (e.g. medicine [7]) we define an intervention as *"an act performed (e.g. use of a technique or a process change) to adapt testing to a specific context, to solve a test issue, to diagnose testing or to improve testing"*. An intervention is constrained by its application context and has one or more objectives or known effects within a certain scope of the testing. Similarly, challenges in software testing can be described in terms of objectives constrained by the context and delimited in scope.

First we analyzed existing taxonomies, standards and classifications of software testing which are used in academia and in industry (e.g. [4], [8]). There are several proposals in literature on how to classify interventions. However they are neither orthogonal nor necessarily useful for the purpose of identifying relevant evaluation points from a problem perspective. We investigated classifications of research on combinatorial testing, model based testing, search-based testing and unit testing. Combinatorial testing techniques are here classified based on how they model the SUT, which combination strategies are used to generate test suites and how test cases are

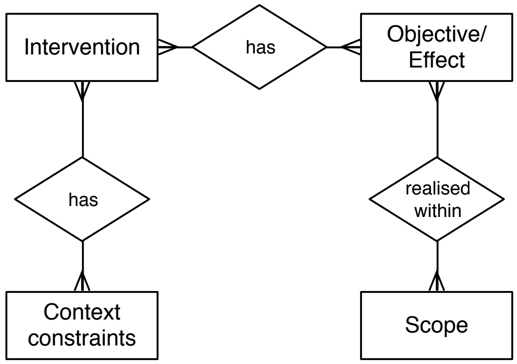


Fig. 1. SERP taxonomy structure

prioritized. Model based techniques are classified according to which information the test models represent and describe. The literature review on search based testing [2] classified research not on characteristics of the techniques but on how techniques had been empirically evaluated (i.e. objective and context). Unit testing research is typically classified based on source of information (e.g. code, specifications or testers intuition).

The main difference between the academic and the industrial taxonomies is the primary focus. While academia tend to focus on characteristics of the intervention industrial standards categorize the area from a process perspective. The terminology of the V-model, i.e. test levels occurs in both types of taxonomies but from different perspectives. Greatest agreement between the academic and the pragmatic perspective is in the terminology describing the scope of a testing intervention and it was easy to identify an understandable high level classification scheme for this facet. There is also agreement regarding the lack of classifications of relevant context constraints.

B. Structure

At this stage of taxonomy development we propose hierarchical classification schemes according to four facets: intervention, context constraints, objectives/effects and scope (cf. Engström and Petersen [9] for definitions). We have not detailed the intervention facet further at this stage. In addition to the review of literature described above we also interviewed test experts from both industry and academia and so far this is an open question. Instead our aim is to search for related groups of interventions based on similarities in classification according to the other three facets.

The elements of the taxonomy structure are linked to each other according to Figure 1 (cf. Engström and Petersen [9] for further details and examples). This means that one may enter the taxonomy from any of those entry points to search for matches between testing interventions and challenges.

C. Effect/Objective of intervention

To classify a practical challenge, the practitioner has to think about the objective targeted when addressing the challenge. For example, low defect detection ability of the current approach may be observed, and the goal is hence to improve testing with respect to effectiveness. It is relevant to classify both empirically shown effects and non-evaluated objectives of testing interventions.

Figure 2 shows the scheme for classifying testing objectives according to SERP-test. The objective for applying a testing intervention may be to improve testing (e.g. increase transparency or efficiency), diagnose testing, adapt testing to a new or changed testing context (e.g. a new process or a new type of software) or to solve an unsolved testing problem.

Note that the term *new* in *new context* and *new test problem* is relative. If, for example, a proposed intervention is suggested to solve a specific problem such as performance testing rather than improving existing performance testing it should be classified as having a problem solving objective even though performance testing is not new to everyone. Similarly if the proposed intervention deals with challenges due to circumstances in the application context, e.g. performance testing in software product lines, it should be classified as having an adapting objective if it is not accompanied with comparisons to other interventions at that point of comparison. From a practitioners' point of view this approach supports problem description based on their current state of testing.

D. Scope of intervention

The scope of a testing intervention describes the extent of the effect, intended or measured, on the test process. Correspondingly the scope of the challenge describes which part of the testing process is primarily affected by the challenge or desired improvement.

Figure 3 show the available categories in SERP-test. At the highest level of abstraction the test process is divided into test planning, test design, test execution, and test analysis.

Test planning refers to decisions on the testing and how to achieve the testing goal. This category is further divided into decisions on coverage criteria and test strategy. A test

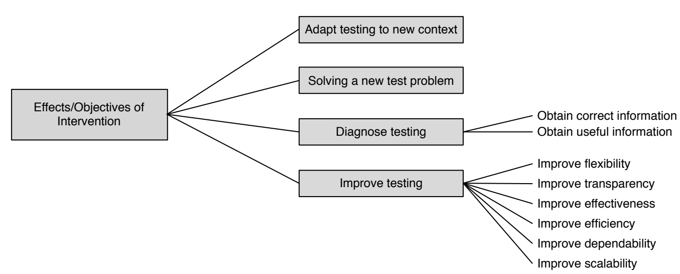


Fig. 2. Effect/Objective classification tree of SERP-test taxonomy

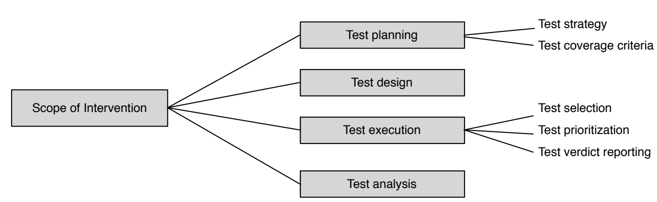


Fig. 3. Scope classification tree of SERP-test taxonomy

strategy is a description of which techniques to use to achieve the testing goal or the test coverage criteria. The test coverage criteria is in turn a specification of the requirements on a test set. This category is important in the pragmatic taxonomies, ISTQB [8] and ISO/IEC 29119 [10], but not as common in the academic classifications.

The remaining three subcategories matches well to both academic and pragmatic taxonomies. Test design refers to the activity of deriving and creating test cases and test sets. Test execution includes scripting, execution and verdict reporting of a test case and a test set. Test analysis refers to the evaluation and analysis of the test outcome with respect to some criteria.

E. Context of application

The context facet regards constraints in the testing context. When classifying interventions the primary factors preventing applicability should be considered while from a problem perspective the primary factors that delimits the selection of possible interventions should be considered. Figure 4 show the available categories.

Interventions and generalization of empirical evaluations of interventions is limited by many factors in the context. Just like for the characteristics of interventions literature suggests taxonomies for classification of context in software engineering [11]. Key is to identify the most relevant in each case and thus we only added few high-level categories here.

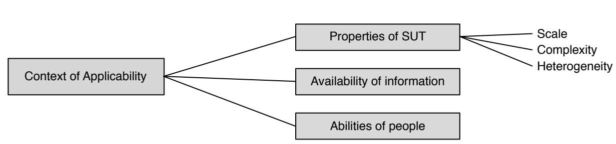


Fig. 4. Contex classification tree of SERP-test taxonomy

III. USE CASES EXAMPLES

The goal of the taxonomy is to provide means to bridge the gap between research and practice in software testing by improving communication. One way of doing this is to *a) support classification of research* from a problem perspective and vice versa to *b) support the search for research* through guidance in problem description. Another type of support relates to the *c) comparison and evaluation of interventions* which may be useful both for searching solutions to problems, designing relevant studies as well as for comparing interventions in for example *d) systematic reviews*.

a) Classifying research: In order to compare research results we need to be able to classify it. For example, in an empirical design study we explored the use of heat maps of historical test data for test scoping support [12]; the classification of the solution being evaluated is shown in Figure 5. A perquisite for visualization is access to the information to visualize. Thus, we classify a limitation of the proposed intervention, documented and accessible test execution information. Furthermore we observed direct effects within the scope of test planning (improved flexibility, improved transparency and improved effectiveness) and indirect effects on test design (improved effectiveness) and consequently classified effects within both these scopes. The ability to describe a testing intervention is a prerequisite for search and comparison.

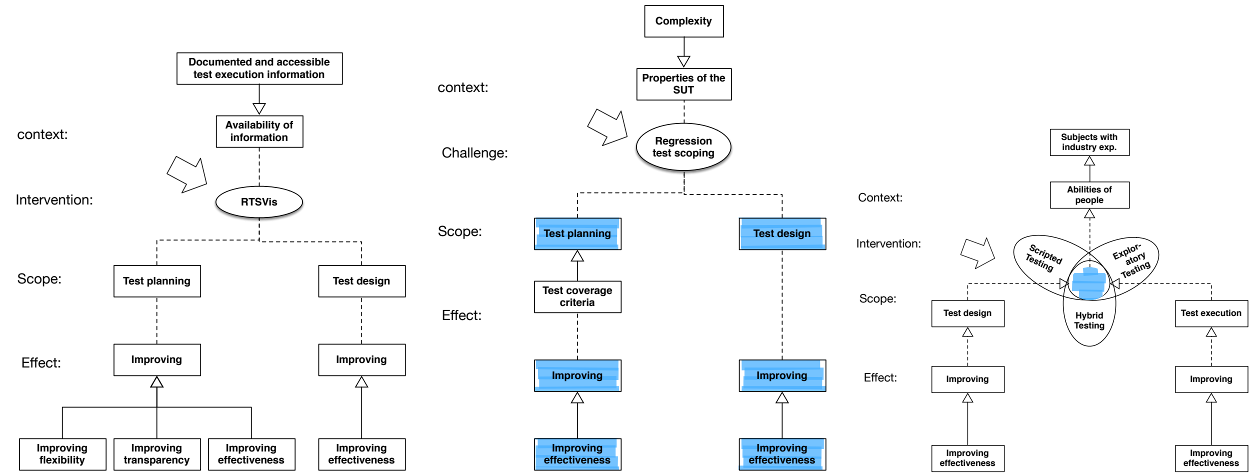
b) Search for research: To exemplify the search for research we use survey input from one practitioner with more than 20 years of experience in software testing. He chose to focus on the challenge in determining the most effective level of regression testing, which he considers problematic both in test planning (i.e. determining the amount of test) and test design (i.e. creating the specific tests). The challenge is to improve effectiveness of the regression testing. Furthermore he believes that the complexity of the SUT delimits the selection of applicable interventions. Figure 5(b) shows how his challenge description maps to the example classification in Figure 5(a) (see highlighted boxes). To search for research he enters the taxonomy from the scope perspective searching for test interventions with the test objective to improve effectiveness of test planning. He finds a match at the highest level of abstraction of the scope classification in both cases. Then he filters on interventions with the objective to improve effectiveness and still got a match. Finally he filters on testing interventions that is not limited in terms of system complexity. That is, by first classifying ones own challenges the number of solutions to look at in further detail are reduced. Now the practitioner only has to look at solutions for test planning and test design that are focused on improving effectiveness.

c) Comparing interventions: The motivation for comparing may stem both from the practitioners wish to find a solution to a practical problem or from the researchers wish to position his or her own work in relation to the state of the art. In the first case a search for research is first done as described to find interventions relevant for the challenge and the context. The taxonomy is then utilized in a similar manner for both cases to support identification of relevant points of comparison. We let a comparative study on test strategies [13] illustrate how the taxonomy can be used for comparing interventions. The entry point here is the interventions just like in the first example. First the match between the techniques is examined to find the least common denominator in each of the classification facets, see Figure 5(c). All three techniques affect effectiveness of both test design and test execution. How and to what extent depend on the skill level of the testers.

d) Systematic reviews: The taxonomy may also be used to guide all stages of a systematic review. First, defining relevant research questions for a systematic review may be guided by classification of challenges and thus encourage taking a problem perspective on the systematic review. In the search for relevant literature the populated taxonomy may form a starting point. The guidelines for comparing interventions may be applied in the synthesis of research findings.

IV. CONCLUSION

The objective of our taxonomy construction at this stage was to define a common starting point for taxonomy evolution which could capture both a problem perspective and a



(a) Example classification of Regression Test Scope Visualization (RTSVis) [12]

(b) Example search based on challenge classification guided by the SERP-test online survey

(c) Example comparison based on challenge classification guided by the SERP-test online survey

Fig. 5. Usage scenarios for the taxonomy

solution perspective with respect to software testing. Thus the goal was not to specify the details of the taxonomy but to identify common denominators between the two perspectives to provide a basis for further classifications. A taxonomy is never complete, rather it should be expandable. Our taxonomy has been designed with this in mind, and the initial structure has been evaluated as being sound and beneficial from the perspectives of researchers and practitioners.

It is essential to popularize the taxonomy and making it available to the test community (practitioners and researchers) through scientific publication, as this is a requirement to: (a) achieving an agreed on view on the terminology used in the taxonomy; and (b) receive further input to extend the taxonomy with new categories and data sets to reflect on its usage. The taxonomy can only unfold its full potential through widespread awareness and a high number of contributions in the form of challenges and solutions.

To succeed with the project we need to motivate researchers and practitioners to classify their interventions and challenges. We believe researchers will benefit as they find practical challenges for their solutions, providing a check of relevance of their research. Furthermore, opportunities for collaboration will be identified. Practitioners on the other hand may receive rapid feedback of potential solutions when describing and classifying their challenges. For both researchers and practitioners the activity of systematically describing their challenges and interventions (guided by the other party's perspective) may drive their work in fruitful directions.

References

[1] S. Vegas, N. J. Juzgado, and V. R. Basili, "Maturing software engineering knowledge through classifications: A case study on unit testing techniques," *IEEE Trans. Software Eng.*, vol. 35, no. 4, pp. 551–565, 2009.

[2] S. Ali, L. C. Briand, H. Hemmati, and R. K. Panesar-Walawege, "A systematic review of the application and empirical investigation of search-based test case generation," *IEEE Trans. Software Eng.*, vol. 36, no. 6, pp. 742–762, 2010.

[3] C. Reinisch, "Academia-industry collaboration: Ways to a strong and beneficial partnership," in *International Workshop on Long-term Industrial Collaboration on Software Engineering*, 2014, pp. 1–2.

[4] M. Unterkalmsteiner, R. Feldt, and T. Gorschek, "A taxonomy for requirements engineering and software test alignment," *ACM Trans. Softw. Eng. Methodol.*, vol. 23, no. 2, p. 16, 2014.

[5] V. Stricker, A. Heuer, J. M. Zaha, K. Pohl, and S. D. Panfilis, "Agreeing upon soa terminology - lessons learned," in *Future Internet Assembly*, 2009, pp. 345–354.

[6] D. Smite, C. Wohlin, Z. Galvina, and R. Prikladnicki, "An empirically based terminology and taxonomy for global software engineering," *Empirical Software Engineering*, vol. 19, no. 1, pp. 105–153, 2014.

[7] D. Heather, *Asperger Personalities, Anxiety and Hypnosis*. Lulu. com, 2012.

[8] ISTQB, "Standard glossary of terms used in software testing."

[9] K. Petersen and E. Engström, "Finding relevant research solutions for practical problems - the serp taxonomy architecture," in *International Workshop on Long-term Industrial Collaboration on Software Engineering (WISE 2014)*, 2014.

[10] ISO/IEC, "The international standard for software testing - concepts & definitions."

[11] K. Petersen and C. Wohlin, "Context in industrial software engineering research," in *Proceedings of the 3rd International Symposium on Empirical Software Engineering and Measurement (ESEM '09)*, pp. 401–404. [Online]. Available: http://dx.doi.org/10.1109/ESEM.2009.5316010

[12] E. Engström, M. Mäntylä, M. Borg, and P. Runeson, "Supporting regression test scoping with visual analytics."

[13] S. M. A. Shah, U. S. Alvi, C. Gencel, and K. Petersen, "Comparing a hybrid testing process with scripted and exploratory testing: A pre-experimental study with practitioners."