Towards a hybrid testing process unifying exploratory testing and scripted testing

Journal of Software: Evolution and Process

JOURNAL OF SOFTWARE: EVOLUTION AND PROCESS
J. Softw. Evol. and Proc. (2013)
Published online in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/smr.1621

Towards a hybrid testing process unifying exploratory testing and scripted testing

Syed Muhammad Ali Shah[1], Cigdem Gencel[2,*,†], Usman Sattar Alvi[3] and Kai Petersen[4]

[1]*Politecnico di Torino, Torino, Italy*
[2]*Free University of Bolzano-Bozen, Bolzano, Italy*
[3]*Seamless AB, Stockholm, Sweden*
[4]*Blekinge Institute of Technology, Karlskrona, Sweden*

SUMMARY

Given the current state of the art in research, practitioners are faced with the challenge of choosing scripted testing (ST) or exploratory testing (ET). This study aims at systematically incorporating strengths of ET and ST in a hybrid testing process to overcome the weaknesses of each. We utilized systematic review and practitioner interviews to identify strengths and weaknesses of ET and ST. Strengths of ET were mapped to weaknesses of ST and vice versa. Noblit and Hare's lines-of-argument method was used for data analysis. The results of the mapping were used as input to codesign a hybrid process with experienced practitioners. We found a clear need to create a hybrid process as follows: (i) both ST and ET provide strengths and weaknesses, and these depend on some particular conditions, which prevents preference of one approach to another; and (ii) the mapping showed that it is possible to address the weaknesses in one process by the strengths of the other in a hybrid form. With the input from literature and industry experts, a flexible and iterative hybrid process was designed. Practitioners can clearly benefit from using a hybrid process given the mapping of advantages and disadvantages. Copyright © 2013 John Wiley & Sons, Ltd.

Received 14 October 2012; Revised 24 July 2013; Accepted 31 July 2013

KEY WORDS: software process improvement; test process; prescriptive testing; scripted testing; test case based testing; exploratory testing; ad hoc testing; empirical study

1. INTRODUCTION

Software testing aims to verify whether software behaves as intended and identifies potential problems. A recent survey [37] indicates that testing is the main approach being used in industry to identify defects. Hence, there is a need to understand how to improve the efficiency and effectiveness of testing approaches. Two widely used testing processes in industry are scripted testing (ST) (also referred to as prescriptive or test case based testing in International Organization for Standardization/International Electrotechnical Commission [ISO/IEC] 29119 Software Testing Standard) and exploratory testing (ET) [24].

Scripted testing follows a prescriptive process, in which test cases are designed prior to test execution to structure and to guide the testing tasks. Many of the existing studies on ST have a focus on automated test case design, generation and prioritization, or testing technique selection [8, 12, 28, 45]. In a sense, ST is a plan-driven process for testing.

*Correspondence to: Cigdem Gencel, Free University of Bolzano-Bozen, 39100 Bolzano-Bozen, Bolzano, Italy.
†E-mail: cigdem.gencel@unibz.it

On the other hand, in ET, the tests are not defined in advance in an established test plan but are dynamically designed, executed, and modified [9]. Exploratory testing is also referred to as ad hoc testing [1] as it relies on the implicit and informal understanding of the testers. Because the literal meaning of ad hoc may correspond to sloppy and careless work, the term 'exploratory' was introduced by a group of experts instead of 'ad hoc' [6]. As the testers can freely explore an application by utilizing human intuition and experience [6, 40], and it is not explicit how they make this exploration, the tasks are performed manually rather than with an automation support.

Scripted testing and ET provide various benefits and weaknesses (Section 2). A few studies [12, 24, 25] mentioned that ET makes better use of testers' creativity and skills to discover the bugs that prescriptive testing may not uncover because of its mechanical nature. Agruss and Johnson [1] and Bach [6] claimed that software testing might benefit through using these approaches in combination. In general, there is a general interest in industry for a hybrid testing (HT) approach unifying the two approaches, which is, for example, visible in lively discussions in industry oriented blogs (see e.g., [43]).

In this study, our aim is to address the need for a systematic and repeatable investigation of such a hybrid process. To this end, we first explored the weaknesses and strengths of ST and ET by reviewing the literature and getting feedback from industry. Then, based on the signified findings by comparing the two approaches, we propose an HT process that unifies ET and ST in a way that some major weaknesses of ET and ST are minimized in a compromise form.

With these objectives, we formulated the research questions (RQs) for this study as follows:

- RQ1: What are the strengths of ST and ET?
- RQ2: What are the weaknesses of ST and ET?
- RQ3: What are the improvement opportunities for testing process by addressing some major weaknesses of ST and ET through unifying their processes in a hybrid form?

It is important to point out that this paper does not focus on individual testing techniques that can be used within the testing process. For example, common testing techniques in ST for black-box testing include, boundary value analysis [32], equivalence partitioning [32], and decision tables [39]. For ST, the commonly used white-box testing techniques include decision coverage [3], path coverage [3], multiple condition/decision coverage (MC/DC) [15], and data flow coverage [10]. One example of a technique in ET is smoke testing [25]. However, instead, our focus here is on the overall 'testing process' that fulfills the characteristics of ST and ET mentioned previously.

In order to answer our RQs, we used systematic literature review (SLR) ([31]) and interviews as the main research methods. Our research process is shown in Figure 1 and was inspired by the technology transfer model proposed by Gorschek *et al.* [18].

Our work starts off with the clear contrast between ET and ST. Consequently, companies could make conscious decisions on which process to choose based on evidence. This implies understanding the strengths and weaknesses of the approaches that are reported in the literature (see problems and issues in Figure 1).

Hence, the first phase (P1) of this exploratory research was investigating the strengths and weaknesses of ST and ET (P1 in Figure 1). Furthermore, we interviewed practitioners with extensive experience of ET and ST in order to identify their perspective on strengths and weaknesses and then compared the outcomes of the interviews to those of the literature review. Through interviews, we also could identify the connections between the strengths and weaknesses of ST and ET that later on helped in identifying the improvement opportunities for an HT process. The details of this step are given in Section 2.

After having identified strengths and weaknesses, we mapped the strengths of one process to the weaknesses of the other and vice versa (P2). Practitioners with extensive experience in both HT and ST were involved in this mapping. They also reviewed the final mapping to improve the reliability of the results. The outcome of P1 and P2 provided two major results that are helpful in working towards an HT: (i) clearly establishing the need for an HT; and (ii) knowing how the strengths and weaknesses of ET and ST relate to each others' help in (i) connecting them to the activities of the HT process to check whether weaknesses are addressed and strengths are supported; and in (ii) providing input to questions to be asked when evaluating an HT. The details of this step are given in Section 3.

DOI: 10.1002/smr

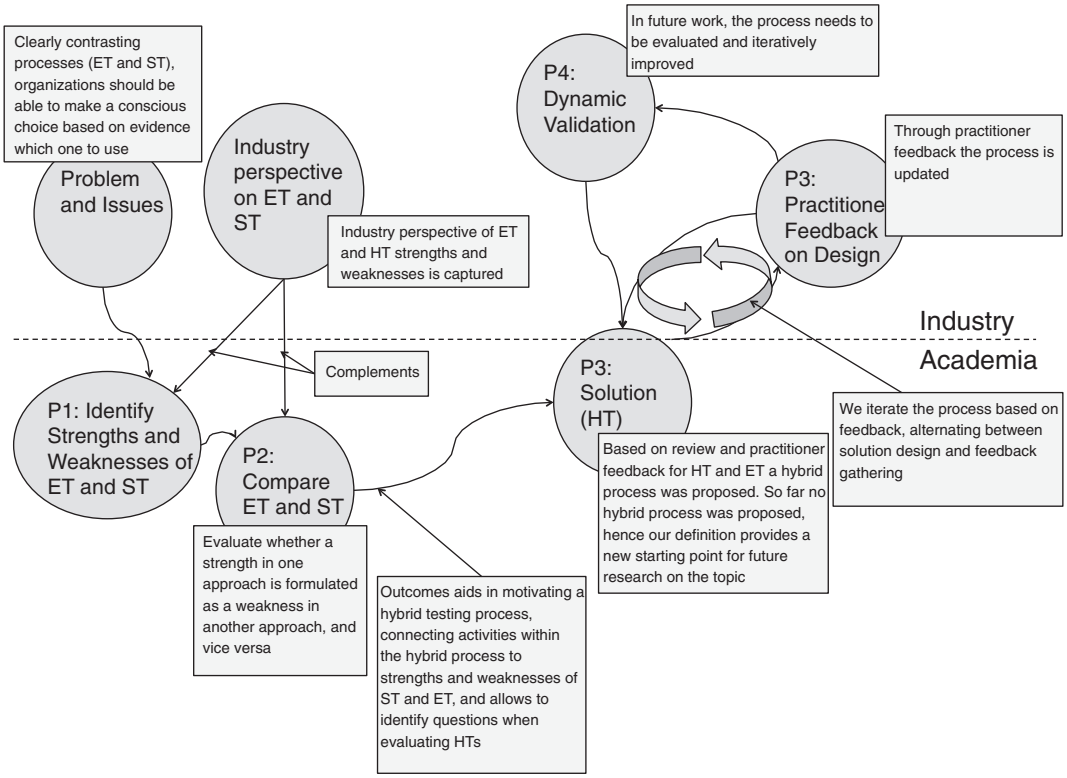

Figure 1. The exploratory research process.

With the input of the previous phase, we designed the HT process in the third phase (P3). We identified the process fragments and high-level structure of the process as suggested in [21]. The initial design was created by mapping the activities of ET and ST to the strengths and weaknesses identified. Having designed an initial version of the solution (HT process), we iteratively improved the design of the process with the practitioners' input (see solution [HT] and practitioner feedback on design). Codesigning the HT process with very experienced practitioners in both HT and ET improves the credibility of the solution proposed. The details of this step are given in Section 4.

As the outcome towards a practically applicable and useful HT process, we provide valuable directions based on making strengths and weaknesses between the two processes as well as how they relate to each other explicit. Furthermore, the HT process proposed was designed with practitioner input. In future work (dynamic validation in Figure 1), the process should be further evolved in controlled experiments, case studies, and action research.

Followed by our design steps presented in Sections 2, 3, and 4, we present the threats of validity to this study (Section 5) and provide the conclusion in Section 6. The conclusion provides answers to the RQs, implications for practitioners and researchers, as well as directions for future work.

2. PHASE 1: IDENTIFYING STRENGTHS AND WEAKNESSES OF EXPLORATORY TESTING AND SCRIPTED TESTING

In order to answer our RQs (RQ1: What are the strengths of ST and ET? and RQ2: What are the weaknesses of ST and ET?), we first performed an SLR (see [31] for guidelines of how to conduct systematic reviews) (Section 2.1). Then, we made semi-structured interviews with practitioners to investigate further the strengths and weaknesses of ST and ET in practice (Section 2.2). This provides the input for comparing the two processes (P2 in Section 3).

DOI: 10.1002/smr

2.1. Systematic literature review

Systematic literature review has several advantages over regular reviews where the research design of the literature is often not presented in sufficient detail. In particular, systematic reviews have the following advantages: (i) reduction of bias due to well-defined criteria for selecting studies; (ii) availability of guidelines of how to aggregate evidence from primary studies; (iii) rigor and documentation of design decisions make the review repeatable and extendable; and (iv) the documentation of every step of the review allows for replication (cf. [31, 35]).

In the succeeding texts, we present the details of the search, data extraction, and data synthesis processes of this SLR.

2.1.1. Search process. The basic steps we followed during the search process were as follows:

- Develop the review protocol.
- Perform the search.
- Review search results using the selection and quality assessment criteria.
- Select the primary studies and finalize the review.

In the succeeding texts, we first present the search strings, the selection criteria and procedure, the quality assessment checklist, and the data sources used for the search process. Then, we provide the results of the search and the selected primary studies. Finally, we discuss the data extraction and data synthesis processes, which led to the conclusions of the SLR.

Search strings: We formulated the keywords and the search strings according to our RQs. We used the synonyms and alternative terms for the keywords referring to linguistic dictionaries while limiting them within the context of software engineering. When deciding on the keywords, we also checked the general terminology used in the testing field (e.g., ISO/IEC 29119 and some key publications such as [24]) not to miss any important keyword. Furthermore, we asked an expert in the area to recap the design of the literature review as well as the list of included papers after the review to make sure that no important study is missed. We did not include keywords for specific testing techniques, as here, our focus was on the studies about test processes of ST and ET.To form the search strings, Boolean operators 'AND' and 'OR' were used to intersect or incorporate the search results for different keywords (Table I). In [31], it is proposed that pilot searches should be carried out in order to identify primary studies by using the defined search strings as defined in review protocol. The search strings were verified by conducting trail searches, and a preliminary search is carried out in order to identify the relevant literature by

Table I. Keywords: (A1 or A2 or A3 OR A4 or A5 or A6) and (B1 or B2 or B3 or B4 or B5 or B6 or B7 or B8].

ID	Keyword
A1	Exploratory testing
A2	ET
A3	Ad hoc testing
A4	Test case based testing
A5	TCBT
A6	Scripted testing
B1	Weakness
B2	Complexity
B3	Shortcoming
B4	Problem
B5	Issue
B6	Strength
B7	Efficiency
B8	Benefit

DOI: 10.1002/smr

the help of the Blekinge Institute of Technology (Sweden) (BTH) librarians.We chose the start year of the search from 2000 when ET was introduced (hence, we assumed that significant work should have been published afterwards) and the end date as January 2010. We made the search between February and May 2010.

Data sources: Search for the primary studies was carried out by using the following electronic resources: IEEE Xplorer, Association for Computing Machinery Digital Library, Engineering Village, Google Scholar, Institute for Scientific Information (ISI), Scopus, and Springer Link.'Zotero' reference management tool [47] was used to manage and keep the track of the primary studies.

Selection procedure and criteria: The selection of the primary studies included two consecutive steps. The inclusion and exclusion criteria were applied to titles and abstracts. After having identified the potentially relevant studies, the full text of the studies was read. In this step, further studies were excluded as it was not clear from the title and the abstract that they were irrelevant.

Our inclusion criteria when selecting the primary studies were the following:

- Studies provide full text and available for access.
- Studies peer-reviewed by other researchers (journal/conference/workshop papers and thesis).
- Studies published as a book or a book chapter.
- Technical reports (including work in progress) and research theses, for example, PhD (gray literature).
- Studies using the research methods: literature review, experiment, case study, field observation, survey, interviews, experience reports, and expert opinion.
- Studies that provide discussion on the strengths and/or weaknesses for ST and ET processes.

Our criteria to exclude the studies were the following:

- Studies not published in English language.
- Studies that were the duplicates of already included studies.
- Reports on blogs and private Web pages.
- Studies without any evaluation, comparative analysis, or relation to practical experience.

For the articles meeting our inclusion/exclusion criteria, we further applied the following quality assessment criteria:

- Research methodology: Is the research methodology mentioned and described (including research goal, data collection, analysis, etc.)?
- Results: Does the study report on the strengths and weaknesses of ST and ET processes based on a sound research process?
- Validity: Does the study discuss validity threats/limitations to the study?

Search Conduct. We performed the search using the data sources and the search strings. We review the search results and by manually going through the titles and abstracts applying the inclusion and exclusion criteria, which at the end left us with 100 studies for further review. After reading the full-text of the articles, 19 studies remained. The list of studies was cross-checked among the two reviewers, and the final list was agreed upon after discussion. We also consulted an external expert for reviewing the list of identified list of primary studies. He mentioned three more studies of relevance. We reviewed these studies and decided to include them in the primary studies list, which led to a final list of 21 studies to be input to the data extraction and analysis step. The selected primary studies are given in Table II. The primary studies included 10 conference papers, 3 journal papers, 4 books, 2 technical reports, 1 licentiate thesis, and 1 book chapter. Fifteen of the studies were published after 2004. In year 2009, five studies were published that shows an increasing trend in discussing either the strengths or weaknesses of ST and ET.

2.1.2. Data extraction. Two authors (Syed Muhammade Ali Shah and Usman Sattar Alvi) were the review team implementing the systematic review process. They designed the data extraction form (Table III) to obtain the required information from the primary studies in order to be able to answer RQ1 and RQ2. One of the other authors, who was not in the review team, reviewed the designed

DOI: 10.1002/smr

Table II. Included papers.

No.	Published Scripted testing/ Exploratory testing venue	Title	Method	Scripted testing		Exploratory testing	
				S	W	S	W
S1	Conference	Itkonen J, Mantyla M, Lassenius C, (2007) Defect Detection Efficiency: Test Case Based vs. Exploratory Testing. First Intern. Symposium on Empirical Software Engineering and Measurement. 20–21 September, Madrid. pp. 61–70.	Controlled experiment	√		√	
S2	Conference	Itkonen J, Mantyla M, Lassenius C, (2009) How do testers do it? An exploratory study on manual testing practices. 3rd Intern. Symposium on Empirical Software Engineering and Measurement. ESEM 2009. pp. 494–497	Field observation			√	√
S3	Technical report	Agruss C, Johnson B, (2000) Ad Hoc Software Testing, A perspective on exploration and improvisation, Florida Institute of Technology, USA, pp. 68–69.	Expert opinion			√	√
S4	Conference	Itkonen J, Rautiainen K, (2005) Exploratory testing: a multiple case study. Intern. Symposium on Empirical Software Engineering. 17–18 November, pp. 10.	Case study			√	√
S5	Journal	Ahonen J J., Junttila T, and Sakkinen M, (2004) Impacts of the Organizational Model on esting: Three Industrial Cases. Empirical Software Engineering. Springer, Netherlands, vol. 9, pp 275–296.	Case study	√	√		
S6	Conference	Andersson C, Runeson P, (2002) Verification and Validation in Industry: A Qualitative Survey on the State of Practice. Proc. of the Intern. Symposium on Empirical Software Engineering, IEEE Computer Society. 3–4 October, Washington, DC, pp. 37.	Survey		√		
S7	Thesis	Itkonen J, (2008) Do test cases really matter? An experiment comparing test case based and exploratory testing. Licentiate Thesis. Helsinki University of Technology, Finland.	Controlled experiment	√	√	√	√
S8	Book	Kaner, (1988) Testing Computer Software. TAB Professional & Reference Books.	Experience report			√	
S9	Book chapter	Bach J, (2004) Exploratory Testing. In: Smith J (ed) The Testing Practitioner, E. van Veenendaal, edn. UTN Publishers, Den Bosch, pp 253–265.	Experience report			√	
S10	Book	Kaner C, Bach J, Pettichord B, (2002) Lessons Learned in Software Testing, John Wiley & Sons, Inc, New York.	Controlled experiment			√	
S11	Conference	Shoaib L, Nadeem A, Akbar A, (2009) An empirical evaluation of the influence of human personality on exploratory software testing. IEEE 13th Intern. Conf. on Multitopic. 15 January, Islamabad, Pakistan. pp. 1–6.	Controlled experiment			√	

(Continues)

DOI: 10.1002/smr

Table II. (Continued)

No.	Published Scripted testing/ Exploratory testing venue	Title	Method	Scripted testing S	Scripted testing W	Exploratory testing S	Exploratory testing W
S12	Technical report	Bourque and Dupuis, (2004) Guide to the Software Engineering Body of Knowledge (SWEBOK), IEEE Computer Society, Los Alamitos, California.	Experience report		√		
S13	Book	Tinkham A, Kaner C, (2003) Learning Styles and Exploratory Testing. Portland. Oregon. USA.	Expert opinion			√	
S14	Book	Ryber T, (2007) Essential Software Test Design, Fearless Consulting.	Expert opinion	√			
S15	Conference	Fraser G, Gargantini A, (2009) Experiments on the test case length in specification based test case generation. ICSE Workshop on Automation of Software Test, 18–19 May, Vancouver, Canada, pp 18–26.	Controlled experiment	√			
S16	Conference	Grechanik M, Qing Xie, Chen Fu, (2009a) Maintaining and evolving GUI-directed test scripts. IEEE 31st Intern. Conf. on Software Engineering. 16–24 May, Vancouver, Canada, pp. 408–418.	Case study	√	√		
S17	Conference	Grechanik M, Qing Xie, Chen Fu, (2009b) Experimental assessment of manual versus tool-based maintenance of GUI-directed test scripts. IEEE Intern. Conf. on Software Maintenance. 20–26 September, Edmonton, Canada, pp. 9–18	Controlled experiment		√		
S18	Conference	Ng S, Murnane R T K, Grant D, Chen T, (2004) A preliminary survey on software testing practices in Australia. Australian Software Engineering Conference. 27 September Hawthorn, Australia, pp 116–125.	Survey	√	√		
S19	Journal	Yamaura, (1998) How to design practical test cases, Software, IEEE, vol.15, 1998, pp. 30–36.	Case study	√	√		
S20	Conference	Taipale O, Smolander K, Kalviainen H, (2006) Factors affecting software testing time schedule. Proc. of the Australian Software Engineering Conference. 18–21 April, Australia, pp.9.	Survey		√		
S21	Conference	Do H, Rothermel G, (2006) An empirical study of regression testing techniques incorporating context and lifetime factors and improved cost-benefit models. In: Proc. of the 14th ACM SIGSOFT Intern. Symp. On Foundations of Software Engineering. 5–11 November, New York, pp. 141–151.	Controlled experiment		√		
S22	Journal	Houdek F, Schwinn T, Ernst D, (2002) Defect detection for executable specifications - an	Controlled experiment			√	

(Continues)

DOI: 10.1002/smr

Table II. (Continued)

No.	Published Scripted testing/ Exploratory testing venue	Title	Method	Scripted testing S	Scripted testing W	Exploratory testing S	Exploratory testing W
		experiment. International Journal of Software Engineering and Knowledge Engineering, vol. 12, (6): pp. 637–655.					

ST, scripted testing; ET, exploratory testing; S, Strength; W, Weakness.

Table III. Data extraction form.

General information
Title of the article
Name of the author(s)
Date of publication
Venue of publication
Data source used to retrieve the research article
Specific information
Study environment: industry/academia/consultancy
Empirical methods used: experiment, case study, survey, field observation, interview, and literature review
Type of study participants: researchers, industry professionals, students
Relevant area of research study with details: ET, ST, weaknesses of ET, strengths of ET, strengths of ST, weaknesses of ST, and comparison of ST and ET

ST, scripted testing; ET, exploratory testing.

form to check relevancy of the data to be extracted and any missing information that needs to be captured. Then, the forms were slightly revised afterwards to include categories of relevant area of study that helped in uniformity of coding.

2.1.3. Data analysis and results. For data analysis and synthesis, we used Noblit and Hare's meta-ethnography method [34], which includes a set of techniques for synthesizing qualitative studies. In particular, we used the lines-of-argument synthesis strategy that involves building a general interpretation grounded in the findings of the primary studies [13]. It is essentially interpretive and seeks to reveal similarities and discrepancies among accounts of a particular phenomenon [7].

In lines-of-argument synthesis strategy, we first identified the 'first order constructs' and the 'second order constructs', and then we came up with the third order interpretations [11, 13]. The first order constructs refer to free codes from primary studies (i.e., each individual strength and weakness as stated in primary studies). From these free codes, we identified the 'second order constructs' that refer to descriptive themes in software engineering (e.g., less bogus defects and defect detection effectiveness). We then further interpreted these to develop third order (or synthetic) constructs. Thereby, four main categories were identified for the strengths and weaknesses: (i) testing quality; (ii) nature of the process (structuredness/flexibility); (iii) cost-effectiveness; and (iv) customer satisfaction. The two reviewers worked together during the analysis phase and made decisions for each construct after joint discussion. An example of how first, second, and third order constructs relate is shown in Figure 2.

The third order constructs and their links to second order constructs arising directly from the literature are presented in the following tables (Tables IV–VII).

We further made a quantitative analysis to provide some quantitative information regarding the percentage of studies with respect to specific types of strengths and weaknesses in addition to types of empirical methods used in those studies.

DOI: 10.1002/smr

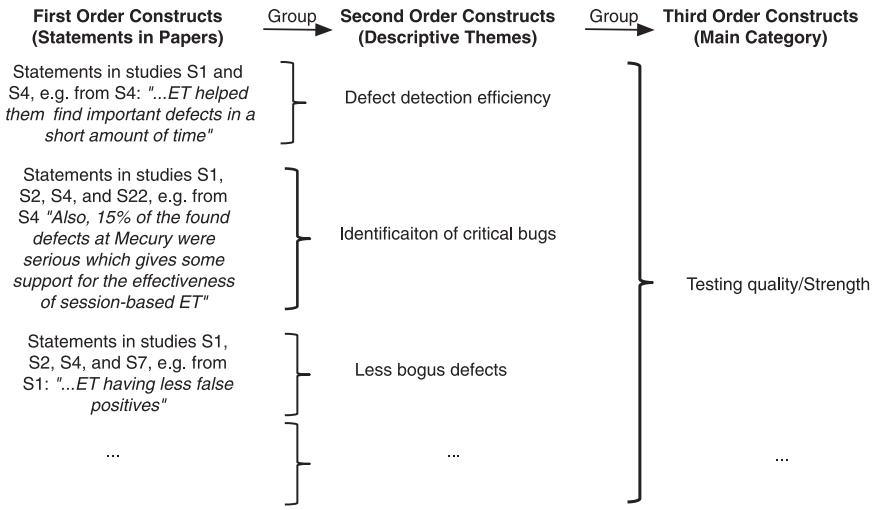

Figure 2. Lines-of-argument synthesis strategy analysis example.

The strengths of ET with respect to the main categories are shown in Table IV. In total, we identified 11 references that discuss the strengths of ET (Table II).

Analyzing the studies found for the strengths, we identified that 82% of the references (cf. S1 [24], S2 [25], S4 [26], S7 [42], S8 [30], S9 [6], S10 [29], S13 [44], and S22 [22]) highlight the strengths of ET related to testing quality (defect detection effectiveness/functionality coverage). The research methods used include controlled experiments, case studies, field observations, and personal experiences and opinions. Of the references, 36% (cf. S9 [6], S2 [25], S13 [44], and S22 [22]) identifies various strengths of ET related to cost-effectiveness by conducting controlled experiments, field observations, and personal experiences and opinions. Of the references, 36% (cf. S9 [6], S11 [40], S3 [1], and S4 [26]) states strengths related to the flexibility of ET in test analysis. The research methods used in these studies are case studies, controlled experiments, and personal experiences and opinions.

Table V shows the identified strengths of ST. We found eight references discussing the strengths of ST (Table II).

The research methods used in the identified studies for the strengths of ST include case studies, surveys, controlled experiments, and personal experiences and opinions. Of the references, 38% (cf. S1 [24], S7 [23], and S14 [36]) highlights the strengths related to testing quality (defect detection effectiveness/functionality coverage). Of the references, 75% (cf. S1 [24], S14 [36], S15 [16], S16 [20], S18 [33], and S19 [46]) mentions strengths of ST related to process flexibility. Of the references, 38% (cf. S7 [23], S14 [36], and S18 [33]) poses ST as good for customer satisfaction especially when there is a need to fulfill legal requirements.

Table VI shows the identified weaknesses of ET. We found four references that discuss the weaknesses of ET based on case studies, controlled experiments, field observations, and personal experiences and opinions (Table II). Among the identified four references, 75% states issues related to testing quality (cf. S2 [25], S3 [1], and S7 [23]). Of the cited references, 100% (cf. S2 [25], S3 [1], S4 [26], and S7 [23]) highlights various weaknesses particularly related to process flexibility. Moreover, some issues related to customer satisfaction are reported by 50% of references (cf. S3 and S4 [1, 26]).

Table VII presents the identified weaknesses of ST. In total, 10 references were identified for the weaknesses of ST (Table II). The research methods used in the identified studies are controlled experiments, surveys, personal experiences, and case studies. Of the references, 70% (cf. S12 [9], S7 [23], S16 [33], S19 [46], S5 [2], S21 [14], and S6 [4]) states that main problems reside in the quality of the design of the test cases. Of the references, 30% (cf. S7 [23], S18 [33], and S17 [19]) highlights issues related to cost-effectiveness. Of the references, 10% (cf. S7 [23]) mentions the issues related to process flexibility.

DOI: 10.1002/smr

Table IV. Strengths of ET.

Main category	Strengths of exploratory testing
Testing quality (defect detection effectiveness/functional coverage)	• Less bogus defects (reduced number of false-positives) (cf. [S1, S2, S4, S7]) • Identification of critical bugs in the system in shorter time (cf. [S1, S2, S4, S22]) • High defect detection efficiency (cf. [S1, S4]) • Investigation and isolation of defects becomes easier as tester directly observes system behavior (cf. [S4, S8, S9, S10, S13]) • Better regression testing (only if test steps are recorded and can later be replayed) (cf. [S1, S4, S8, S10])
Cost-effectiveness	• Rapid feedback on a new product or a feature as testing can be started immediately without extensive planning and coding of test suites (cf. [S9, S13]) • Quick learning of a new product by the tester who is exploring the system (cf. [S2, S9]) • Low reliance on comprehensive documentation as no documentation is needed, the experience of the tester guides the session (cf. [S9, S13]) • Easy maintenance as there is no need to maintain large test suites including a vast amount of test code (cf. [S9]) • More time allocation in actual testing of the product given that no comprehensive documentation/test code needs to be produced (cf. [S9, S22])
Nature of process (flexibility)	• Free exploration as the tester can freely explore the system (e.g., conduct unusual test scenarios) (cf. [S4, S9]) • Simultaneous learning and testing as the tester is exploring the system's functionality while testing (cf. [S4, S9]) • Improvising on scripted tests as scripted tests are not blindly followed, testers can improvise and explore freely (cf. [S9]) • Interpreting vague test instructions is possible in ET as the tester can complement with own experience (written automated test scripts based on oracles often require precise instructions) (cf. [S3]) • Diversification in testing as the freedom in writing tests leads to dissimilar results (cf. [S9]) • Utilization of testers' skills as the tester is not restricted by pre-defined rules of how to create test cases (cf. [S3, S11]) • Better product analysis as the product is explored from a usage perspective (cf. [S3]) • Improving existing tests as ET can be used to planning additions and improvements to already existing automated test suits (cf. [S4]) • Identifying missing tests that are overlooked by following a ST approach (additional tests can be found through ET) (cf. [S4]) • Cross-checking the work of another tester (ET should be used complementary to other test activities and can serve as a cross-check to ST test output) (cf. [S3, S9]) • Investigating a particular risk in order to plan a prescriptive test (cf. [S3])

ST, scripted testing; ET, exploratory testing.

2.2. *Interviews*

We conducted semi-structured interviews with practitioners in industry to further investigate the experiences and opinions of the domain experts for the weaknesses and strengths for ET and ST as a complementary to what we identified in the literature performing an SLR.

In the succeeding texts, we discuss the details of the data collection and the analysis phases of the systematic review.

2.2.1. Data collection. Four data collection instruments were designed by the two authors of this paper, who also performed the SLR (APPENDIX 8 Questionnaires 1–4). We first

DOI: 10.1002/smr

Table V. Strengths of scripted testing.

Main category	Strengths of scripted testing
Testing quality (defect detection effectiveness/functional coverage)	• Higher testing functionality coverage by making conscious/planned coverage decisions (cf. [S1,S7]) • Complex relationships of a function to be tested identified, cf. [S1,S7] • Most of the test conditions captured (e.g., all decisions are covered, all combinations of valid and invalid input samples of different valid and invalid classes) (cf. [S14]) • Test cases depict the overall picture of the perceived quality (cf. [S14])
Nature of process (structured/ guided)	• Oracles availability for the validation of the expected output against the actual value obtained from the test (cf. [S14, S19]) • Detailed information and guidance available for the tester for test execution (e.g., through testing techniques giving concrete guides of how to achieve specified coverage criteria) (cf. [S1, S18, S19]) • Resource independence in execution as tests can be run automatically when scripted (cf. [S15, S16]) • Repeatability of the same tests (e.g., for regression testing) (cf. [S1]) • Reusability of the test cases (cf. [S1]) • Better risk management (cf. [S14]) • Better analysis of the system specification from diverse angles as problems in the specification become visible when deriving tests from it (cf. [S15, S18, S19]) • Quality of the test cases can be validated (e.g., through test case reviews) (cf. [S14]) • Better tracking of progress (e.g., completed x% of the implemented test cases in the regression test suit) (cf. [S19]) • Early quality prediction based on test case metrics (cf. [S14, S19])
Customer satisfaction	• Required when legal and regulatory requirements are to be addressed (cf. [S7, S14]) • Better serves in acceptance testing (cf. [S14, S18]) • Better serves in release testing (cf. [S7, S14])

ST, scripted testing; ET, exploratory testing.

Table VI. Weaknesses of ET.

Main category	Weaknesses of exploratory testing
Testing quality (defect detection effectiveness/functional coverage)	• Hard to assess whether all new functionalities and features are tested (cf. [S2, S3]) • The quality of testing not known because of the dependency on the skills of the testers (cf. [S3]) • Unavailability of oracles (cf. [S7])
Nature of process (unstructured/ ad hoc)	• Difficulty in prioritizing and selecting the appropriate tests (cf. [S2]) • Difficulty in reevaluating the test (cf. [S7]) • Difficulty in monitoring and keeping track of the progress (cf. [S7, S4]) • Lack of effective risk management (cf. [S7]) • Repeatability of the tests is challenging because there is no documentation (cf. [S3]) • Investigating and isolating the actual cause of the problem taking longer time (cf. [S7])
Customer satisfaction	• Not suitable for acceptance, performance, and release testing (cf. [S3]) • Less accountability and audit ability (cf. [S3, S4])

ST, scripted testing; ET, exploratory testing.

DOI: 10.1002/smr

Table VII. Weaknesses of scripted testing.

Main category	Weaknesses of scripted testing
Testing quality (defect detection effectiveness/functional coverage)	• Defect detection effectiveness and functionality coverage rely on the quality of the test case design (cf. [S7]) • Dependency on testers' skills, experience, and domain knowledge for test case design (cf. [S7]) • Test cases being prone to human error (e.g., coding mistakes in written test cases) (cf. [S5, S12, S19]) • Quality of the test cases not known until their execution (cf. [S6, S19]) • The possibility of redesigning the test cases under time constraints to cause low quality design (cf. [S16, S20, S21]) • Not suitable for regression testing when test cases are not well maintained/updated (erosion of regression test suit) (cf. [S21])
Cost-effectiveness	• Exhaustive and protracted (cf. [S7]) • Designing and documenting require considerable effort (cf. [S18]) • Often overruns the assigned budget and time (cf. [S7, S18]) • Test cases not sufficient for the entire system life cycle (cf. [S18]) • Durability of the test cases not known (cf. [S7]) • Reusability and maintenance of test cases can be quite expensive (cf. [S17]) • Redesign or revision due to poor quality of the test cases increase the cost more (cf. [S17])
Nature of process (inflexibility)	• Prescriptive process does not give freedom to the testers (even in cases where the test cases quality is not good) (cf. [S7]) • The testers skills not utilized during test execution (cf. [S7]) • Difficulty in prioritizing the test cases (cf. [S7])

designed the questionnaires with open-ended questions based on the weaknesses and strengths of ET and ST as identified in the literature. In order to assure the quality of the instruments, first, another author of this paper cross-checked the questionnaire. Then, to check whether we need to add more relevant and follow up questions, we piloted the questionnaire with two industry practitioners having the knowledge on both ET and ST. Afterwards, we finalized the instruments.

We conducted interviews with five persons having worked as software testers, test managers, practitioners, or consultants. Our sampling of the interviewees was purposeful as we focused on practitioners with a very high level of experience in both types of processes (minimum 10 years), that is, ET and ST processes. In order to make this research more authentic and reliable, we selected interviewees who hold a senior position in reputable organizations. The experience adhered by such professionals was of great essence as they are also involved in interacting with stakeholders. By conducting interview of such people, it gave us broader insights of the problem domain from multiple perspectives. Given that a high requirement was put on the experience, the number of people to ask was limited, and it was a challenge to identify a high number of them. Hence, we focused on senior testers and also on people known in the testing domain with respect to their knowledge on ST and ET (two interviewees were, e.g., identified through keynotes they gave on the topic). The people interviewed fulfilled our criteria, but their number was limited given the previously mentioned requirements. Some diversity was achieved by interviewing people from different companies. The implications of the sampling strategy on the validity of the study are discussed in Section 5.

DOI: 10.1002/smr

Interviewee 1 has been working as a test manager in Logica AB (Sweden) for the last 2 years. In the past, he worked for a number of companies including Microsoft and UIQ Technologies. Interviewee 2 has been working as a consultant for Telenor AB (Sweden) for the last 2 years. Interviewee 3 is the owner of DevelopSense (Canada) and has been providing consultancy, training, coaching, and other services in software testing. Interviewee 4 has been working for Maquet Critical Care AB (Sweden) as a test manager for the last 6 years. Interviewee 5 is the founder of Satisfice Inc. (USA), which is dedicated to teaching and consulting in software testing and quality assurance. Most of his experience is with market-driven software companies such as Apple Computer and Borland.

Four of the interviews were conducted face-to-face and one online through Skype because of geographical distance. We presented the interviewees the aims of this research before the interviews. The duration of each interview was between 60 and 90 min. We took notes and recorded the interviews using a digital recorder. The data collected from the interviews were transcribed* in order to eliminate any irrelevant information.

2.2.2. Data analysis and results. The transcribed outputs of the interviews were qualitatively analyzed by applying the notice, collect, and think technique [38]. This is a nonlinear qualitative analysis model and consists of three phases: noticing, collecting, and thinking phases. These phases are iterative, recursive, and holographic in nature.

First, the two authors who also performed the SLR analyzed the interviews. Then, another author of this paper made an independent analysis. The results were cross-checked, and then after a discussion, the codes, the main categories, and the connections in between the main strengths and weaknesses were agreed upon solving very few disagreements also by consulting the interviewees.

In the noticing phase, all the relevant information highlighted by the interviewees regarding the strengths and weaknesses were noted using a heuristic coding approach. For example, during the noticing phase, for ET, we captured the following codes from the interviewees: 'less time', 'less documentation', 'more focused documentation', 'more time on actual testing', 'better resource utilization', and 'rapid feedback and quick learning of the product'. As for ST, we identified the codes as 'time consuming', 'exhaustive', 'too much documentation', taking time', 'less costly if test cases can be automatically generated', and 'time depends on the quality desired.'

Then, during the collecting phase, we sorted the weaknesses and strengths and categorized them under main categories based on the similarities and differences between them. Thereby, we identified 'cost-effectiveness' as a main category.

In the thinking phase, both the codes and the main categories were reexamined. Here, we observed that some of the strengths and weaknesses have connections. For example, one of the interviewees mentioned that even though ST takes more time because of too much documentation (hence, less cost-effective), ST was required especially in cases where there was a need to have documented proof of testing where legal and regulatory requirements were to be met. This was a good example showing why one approach should not replace the other, but rather a hybrid process, which optimizes the strengths and minimizes the weaknesses of both approaches, is required. Thereby, we used these insights for identifying improvement opportunities for an HT process as a complementary to what has been captured from the SLR.

In the succeeding texts, we summarize the results of the analysis for the strengths and weaknesses of ET and ST as experienced in industry. However, this time we preferred reporting the strengths and weaknesses in a narrative form instead of reporting them only independently as we did for the SLR (Table VIII shows the additional categories identified in comparison to SLR findings). This is because of that, through interviews, we also could capture the totality of philosophy as expressed by the interviewees for the strengths and weaknesses of ST and ET that might help in identifying the improvement opportunities for a hybrid process.

*Transcriptions can be found on http://www.bth.se/tek/aps/kps.nsf/pages/hybrid-testing-study

DOI: 10.1002/smr

Strengths and weaknesses of ET: The interviewees were of the opinion that unstructured and flexible process in ET could provide either strengths or weaknesses depending on the conditions.As for the strengths, they mentioned that a tester could freely explore different areas of the product and that ET was a process of simultaneous learning and testing. The interviewees had an agreement on the cost-effectiveness of ET because of less time spent on documentation (i.e., focused documentation for only logs, test notes, and videos after the execution), better resource utilization, rapid feedback, and quick learning of the product. Related to this, three of the interviewees mentioned that defect detection efficiency was likely to be high in ET as more time was spent on actual testing rather than on test design and comprehensive documentation.

Moreover, three interviewees were of the opinion that ET could achieve better regression testing and help in identifying most of the critical bugs. Three interviewees stated that ET was handy in investigating more risky parts of the software. Two interviewees claimed that customers were more satisfied as more bugs and also critical ones could be identified.All five interviewees highlighted one key strength of ET as a better utilization of the testers' skills. The reason was stated as testers to become more responsible, engaged, motivated, and creative, while they were given freedom. On the other hand, the interviewees also emphasized that this strength could also become a major weakness in some situations as the quality of testing became dependent on only testers' skills and the domain knowledge.According to three interviewees, the availability of an oracle becomes an issue when the application is too complex, the skills and the domain knowledge of the testers are insufficient, and if the time is running out, and functional specifications have not been updated. Moreover, they mentioned that the flexibility in the process caused significant difficulties in terms of managing, prioritizing, and tracking the tests. Four interviewees were of the opinion that managers and organizations were reluctant to implement ET because they thought they might lose control over testing. Two interviewees added that automation support was not possible for ET.All four interviewees agreed on the fact that using ET alone is not suitable in some cases, and it should be used as a complementary approach to prescriptive approaches. One of the interviewees stated that conducting only ET on complex application alone was not suitable and should be combined with other test approaches in order to ensure testing of critical functionality of complex and real time applications. One of the interviewees emphasized that ET was an approach and not a technique and, therefore, it was already being used with prescriptive techniques as ST. Two of the interviewees raised the need to have a more structured process for ET for better management. They also mentioned that ET could serve well in terms of testing quality if used together with a prescriptive approach such as ST.

Strengths and weaknesses of ST: Similar to ET, all interviewees stated that the structured and formal process in ST could provide either strengths or weaknesses depending on the conditions. As for one major strength, three of the interviewees mentioned that ST was required especially in cases where there was a need to have documented proof of testing where legal and regulatory requirements were to be met. Furthermore, one interviewee added that ST also served well for the acceptance testing.

All interviewees were of the opinion that ST provided better test guidance to testers on specifying desired outputs in test oracles and also could support testers in creative testing.

All interviewees mentioned that quality of testing (functionality coverage and defect detection efficiency) was depended on test case design quality. Moreover, two interviewees said that test case design quality was dependent on skills, experience, and domain knowledge of the designer, as well as on previously produced documents, such as software requirements specification or test plan. They stated that the test quality would be high if the design quality was high. Another benefit, pointed out by an interviewee, was early quality assurance with respect to requirements specifications. He stated that bugs could be found before testing starts when designing test cases from requirements specifications.

On the other hand, two of the interviewees stressed the fact that the quality of the test case design could not be known before testing. Three interviewees mentioned that a tester was not free to make decisions even if the test cases were not designed properly.

DOI: 10.1002/smr

Table VIII. Additional strengths and weaknesses identified from the interviews in comparison to literature.

Findings of systematic literature review and interview		Additional findings from the interview	
ET strengths	ET weaknesses	ET strengths	ET weaknesses
Better learning curve, better resource utilization (more responsible and engaged testers), less time consuming, better regression testing, handy in investigating risky parts of the code	More structured process required ET alone not suitable for testing complex applications; functionality coverage might be a problem; managing, prioritizing, and tracking the test process are difficult; quality depends on tester skills (e.g., oracle issues if testers are not skilled)	Focused documentation, difficulty in interpretation of the test results, difficulty in automation support	Reluctant managers fearing to lose control
ST strengths	ST weaknesses	ST strengths	ST weaknesses
Suitable for meeting regulatory requirements, learning of the product better from multiple scenarios, less costly if test cases are automatically generated, suitable for regression testing and acceptance tests	The designed test cases limits the possibilities and decisions of testers, time consuming and exhaustive process, difficult to follow each step of the test case, quality depends on the test case design quality (both functionality coverage and defect detection efficiency depend on the test case design quality), managing test cases is difficult and costly	Early quality assurance (test cases provide feedback before testing)	Test case design quality depends on the skills and the domain knowledge of the designer

ST, scripted testing; ET, exploratory testing.

DOI: 10.1002/smr

Three of the interviewees stated that most of the time, they experienced good functionality coverage in their companies when using ST. They added that this was because of documenting the test cases in correspondence with the requirement specification provided better functionality coverage. One stated that he experienced low defect detection efficiency. Two of the interviewees mentioned that finding defects by ST was difficult as it might be impossible to follow each and every step of the test case. About increasing testing quality, all interviewees were of the opinion that the quality of testing would increase if ET were used as a complementary approach to ST.

Low cost-effectiveness and difficulty in managing large number of test cases were stated as two major weaknesses of ST. All interviewees were of the opinion that designing, documenting, and executing test cases were too much time consuming and costly. One interviewee mentioned the need that the test cases should be updated continuously in the software development life cycle as the requirements change. Moreover, two interviewees added that the test cases required revision and/or redesign in cases of low quality design. These last two requirements bring more management overhead and thus cost.

2.3. Summary of the systematic literature review and interview results

We performed qualitative comparative analysis [17] to identify commonalities and diversities between the results obtained from the SLR and the industrial interviews.

The results of industrial interviews showed that most of the weaknesses and strengths identified from literature have also been experienced in industry (Table VIII). Therefore, we also distinguish findings reported both in the literature and by the interviewees from the new findings identified during the interviews. Furthermore, in the following paragraphs, we also discuss the new and more insights that we captured from the interviews providing a bigger picture with connections between the strengths and weaknesses in addition to what has been reported as individual strengths and weaknesses in the literature.

The weaknesses of ET were attributed to ET being an unstructured and ad hoc process (which causes difficulties in planning, managing, and tracking the testing process) or related to dependency of testing quality on the skills, experience, and domain knowledge of the testers.

For ST, many weaknesses were reported to be related to cost-effectiveness and dependency of testing quality on test case design quality. As for the strengths, many strengths for ET were reported as being related to cost-effectiveness, process flexibility, and testing quality; whereas for ST having a defined and repeatable process, testing quality, and being independent from the skills of testers during the test execution.

During the interviews, we identified some more aspects, which have not been reported in literature. For example, focused documentation was found to be a strength for ET. As for ST, another strength identified is early quality assurance. One of the interviewees stated that bugs could be found before testing starts when designing test cases from requirement specifications.

On the other hand, one weakness identified for ET is the reluctance of managers in organizations to implement ET because of having the fear to lose control over testing. Another weakness of ET is the difficulty in interpreting the test results because these are generated based on the testers' own experience and intuition. We also found that the interviewees do not believe that automation support is possible in ET.

Furthermore, from the interview results, we also could identify the conditions for when a strength of one approach could become a weakness and vice versa. For example, one significant conclusion is that quality of testing in ET and ST depends on some conditions. A few studies in literature reported ST to perform well for functionality coverage but poor for defect detection efficiency in comparison to ET.

However, the interviews revealed that quality of testing in ST depends on the test case design, which depends on the skills, experience, and domain knowledge of the *test designers* as well as the previous documents from which the product requirements are inherited. On the other hand, the quality of the testing in ET depends on skills, experience, and domain knowledge of the *testers who execute the tests*.

Therefore, when the testers lack some of these attributes, for example, domain knowledge and experience, it would be better to use either ST alone or ET as a support for ST. Or, if there is a doubt about the quality of previous documents (such as requirements specification) from which the test cases are to be derived, then ET might work better if the testers have domain knowledge and experience.

DOI: 10.1002/smr

Another significant conclusion from the interviews is that all interviewees emphasized using ET as a complementary approach to ST as they all believe that this would bring many benefits and help in overcoming major weaknesses. Hence, we identified the following improvement opportunities for designing an HT process:

- *Utilizing the skills and the domain knowledge of the testers during both design and test execution*. In ST, the quality of testing depends on the 'test case design', and the test case design quality depends on the test case designer skills, experience, and the domain knowledge as well as the previous documents from which the product requirements are inherited. In ET, the testing quality depends on the skills, experience, and domain knowledge of the testers who execute the tests. Therefore, there is a need to increase the utilization of all available test skills and expertise both in design and execution.
- *Defining a structured process with some level of flexibility*. This is required to enable better management and increased motivation of the testers by incorporating the creativity and skills of them as well as overcoming the risk of not being able to take an action when they encounter poor test case design. The defined process should also require more focused and less documentation in order to increase cost-effectiveness.

In the next section, we present the mapping of strengths of one approach to the weaknesses of the other to identify how to design the HT process by incorporating different aspects of ST and ET to overcome the weaknesses in the compromise form.

3. PHASE 2: MAPPING EXPLORATORY TESTING AND SCRIPTED TESTING IN RELATION TO STRENGTHS AND WEAKNESSES

A mapping process is a method of identifying problems and their solutions in a structured way. In this investigation, we used mapping process [27] as an important feature of research technique evaluation method, which helps to develop the mechanisms that support to find the solution of one testing approach weaknesses considering other approach strengths. For this, we list down one approach weaknesses against the other approach respective strengths.

Table IX shows the mapping of the identified strengths of ET as candidate solutions to the weaknesses of ST. Table X shows the mapping of the identified strengths of ST as candidate solutions to the weaknesses of ET. Observe that the benefits and weaknesses were previously categorized into testing quality, cost-effectiveness, nature of process, and customer satisfaction. The categories were used to match related benefits and strengths to each other. As an example, the ST issue of 'Prescriptive process does not give freedom to the testers' under the category of the nature of process is addressed in ET through 'free exploration'.

Overall, the intention is to leverage on the benefits listed on the right column of Tables IX and X by defining a structured prescriptive process, which at the same time gives flexibility to testers to conduct ET. In other words, by having both aspects in one compromise process would aid in overcoming some weaknesses of ST and ET, whereas the strengths of both processes are utilized.

In the following section, describing the P3 of this research, the hybrid process incorporating ST and ET is presented. We provide rationales on how the different activities map to the strengths and weaknesses identified earlier (P1 and P2).

4. PHASE 3: DESIGNING THE HYBRID TESTING PROCESS

As illustrated in Figure 1, we designed the process iteratively. Our design started out with creating an initial version of the process based on the results of P1 and P2. We start by presenting the design rationales for our initial process.

DOI: 10.1002/smr

Table IX. Mapping of the strengths of exploratory testing to the weaknesses of scripted testing.

Weaknesses of scripted testing	Strengths of exploratory testing as Candidate Solutions
Testing quality	Testing quality
• Defect detection effectiveness and functionality coverage rely on the quality of the test case design	• Less bogus defects (reduced number of false-positives)
• Test case design depends on the skill, experience, and domain knowledge of the testers	• Identification of critical bugs in the system in shorter time
• Test cases are prone to human mistakes	• High defect detection efficiency
• Quality of the test cases not known until their execution	• Investigation and isolation of defects become easier as tester directly observes system behavior
• Redesigning the test cases under time constraints may cause low quality design	• Better regression testing (only if test steps are recorded and can later be replayed)
• Not suitable for regression testing when test cases are not well maintained/ updated (erosion of regression test suit)	
Cost-effectiveness	Cost-effectiveness
• Exhaustive and protracted	• Rapid feedback on a new product or a feature
• Designing and documenting require considerable effort	• Quick learning of a new product by the tester who is exploring the system
• Often overruns the assigned budget and time	• Low reliance on comprehensive documentation
• Test cases are not sufficient for the entire system life cycle	• Easy maintenance as there is no need to maintain large test suites
• Durability of the test cases are not known	• More time allocation in actual testing of the product
• Reusability and maintenance of test cases can be quite expensive	• Focused documentation
• Difficulty in prioritizing the test cases	
• Redesign or revision due to poor quality of the test cases increase the cost more	
Process (inflexible)	Process (flexible)
• Prescriptive process does not give freedom to the testers	• Free exploration
• The testers skills not utilized during test execution	• Simultaneous learning and testing
• Difficulty in prioritizing the test cases	• Improvising on scripted tests as scripted tests are not blindly followed
	• Interpreting vague test instructions is possible in exploratory testing
	• Diversification in testing
	• Better utilization of the skills of testers
	• Better product analysis
	• Improving existing tests
	• Identifying missing tests that are overlooked by following a scripted testing approach
	• Cross-checking the work of another tester
	• Investigating a particular risk in order to plan a prescriptive test

4.1. Method engineering for initial hybrid testing process

Design goals: In order to identify the candidate solution, we take into consideration all the weaknesses and strengths of both approaches identified through SLR and from interviews. If one approach lack in providing some of the aspects in a candidate solution, it is taken from other approach and so forth. In other words, by having both aspects in one compromise process would aid in overcoming some

DOI: 10.1002/smr

Table X. Mapping of the strengths of scripted testing to the weaknesses of exploratory testing.

Weaknesses of exploratory testing	Strengths of scripted testing as Candidate Solutions
Testing quality	Testing quality
• Hard to assess whether all new functionalities and features are tested	• Higher testing adequacy by making conscious/planned coverage decisions (functionality coverage)
• The quality of testing not known because of the dependency on the skills of the testers	• Complex relationships of a function to be tested identified
• Unavailability of oracles	• Most of the test conditions captured (e.g., all decisions are covered, all combinations of valid and invalid input samples of different valid and invalid classes)
• Difficulty in interpreting the test results	• Test cases depict the overall picture of the perceived quality • Early quality assurance
Process (ad hoc and unstructured)	Process (structured and guided)
• Difficulty in prioritizing and selecting the appropriate tests	• Oracles availability for the validation of the expected output against the actual
• Difficulty in reevaluating the test	• Detailed information and guidance available for the tester for test execution
• Difficulty in monitoring and keeping track of the progress	• Resource independence in execution
• Lack of effective risk management	• Repeatability of the same tests
• Repeatability of the tests is challenging because there is no documentation	• Reusability of the test cases
• Investigating and isolating the actual cause of the problem taking longer time	• Better risk management
• Fear to lose control over testing	• Better analysis of the system specification from diverse angles
• Automation support not possible	• Quality of the test cases can be validated • Better tracking of progress • Early quality prediction based on test case metrics
Customer satisfaction	Customer satisfaction
• Not suitable for acceptance, performance, and release testing	• Required when legal and regulatory requirements are to be addressed
• Less accountability and audit ability	• Better serves in acceptance testing • Better serves in release testing

weaknesses of ST and ET, whereas the strengths of both processes are utilized. From the comparative analysis, we showed that weaknesses in one approach are potentially improved through strengths in the other process, refer to Section 3.

Process definition: We based the HT process on ISO/IEC 29119 (2009),[†] which is a software testing standard aiming to provide one definitive standard that captures vocabulary, processes, documentation, and techniques for the entire software testing lifecycle. The testing processes in this standard include organizational, management, and fundamental test processes.

When defining the HT process, we considered only the management and fundamental processes as given below. Organizational processes were not in the scope of the HT process definition, as these processes include definition of organizational test policy and test strategy that are outside of the main research focus of this paper.

- Management processes:
  - Test planning
  - Test monitoring and control
  - Test completion

[†]The ISO/IEC 29119 is a new upcoming standard, and currently, three parts are under development; part 1 (definitions and concepts), part 2 (test process), and part 3 (test documentation) were released for expert review. The working draft part 2 is used for this investigation. More information is available at http://www.softwaretestingstandard.org.

DOI: 10.1002/smr

- Fundamental processes:
  - Test design and implementation
  - Test environment setup
  - Test execution
  - Test incident reporting

In order to incorporate ET concepts into HT process definition, we used the session-based test management process defined by Bach [5]. The reason for choosing this process definition was that during our interviews, we identified that it is a well known approach in industry. In session-based test management, a test session is the basic testing work unit. This session is an uninterrupted block of reviewable and chartered test effort, that is, each session is associated with a test mission. Every test session is debriefed after execution. The debriefing occurs as soon as possible after the session. The test outcomes, issues, bugs, and related information are stored on the 'session sheets'.

As we previously reviewed the strengths and weaknesses with respect to testing quality, cost-effectiveness, structuredness of testing process, and customer satisfaction, we discuss how these four attributes were incorporated in the HT process design (also referred to as fragment selection in method engineering [21]). Hereafter, this reasoning has been taken into the collaborative design activity with the practitioners as presented in Section 4.2.

The bullets listed showed the initial idea of the process, in which it is tried on how to incorporate these four main attributes in the HT process. Hereafter, this is presented to the interviewees to obtain the feedback:

- *Testing quality*: Following Sections 2.1 and 2.2, we found out that testing quality (defect detection effectiveness and functionality coverage) depends on a couple of conditions for both ST and ET. For example, testing quality for ST depends on the test case design quality, which depends on the test designer skills. As for ET, the quality depends on the skills and the domain knowledge of the testers. Considering different quality aspects of each approach, in HT process, we need to adopt these aspects of both processes. For this, we unify the subsection's 'test design' and 'test execution' of both approaches in a formal manner. The idea is to achieve better coverage by defining requirement-based test cases (RBTC) [41] and test missions. For example, through the requirements, one can check whether all highly prioritized requirements have been tested. In order to achieve the defect detection effectiveness, we allow the testers to explore the product under testing freely and to utilize their intuitions and experience in identifying defects. In addition, HT also allows testers to execute the designed RBTC and test missions. Following the proposed HT process, our proposition is that the strengths of both the approaches are aligned, and the testing performed would be planned, and effective with the focus on complex function and having ability to identify critical defects.
- *Cost-effectiveness*: Following Sections 2.1 and 2.2, we found that ST is not a cost-effective approach where ET is cost-effective. Scripted testing highly relied on the test design phase where ET is meant to be simultaneous test design and execution. The HT process is meant to have cost-effectiveness by adopting both ST and ET attributes. For this, we tried to lessen the contribution of test design phase by introducing RBTC [41] and test missions in the HT process. The consideration of high-level test cases such as RBTC and test missions lessen the dependability on the formal test case design, which includes each aspect of conditions in the code, input data, and GUI under test. Thus, our design proposition is that the use of high-level test cases in the form of RBTC and test mission took less time in design, without much compromising on the benefits of the test design phase of ST. In complement to RBTC and test missions, we introduce a step of free exploration that could allow more time being spent on the actual testing task, rather than designing the test. Subsequently, the time saved in the test design phase should make the HT process more cost-effective in comparison to ST, and the introduction of free exploration may help to attain better quality in a form of defect detection efficiency (as is evident from our literature review).
- *Unstructured process*: Following the findings shown in Table VI, ET has no process structure, it is meant to be free exploration only, whereas ST has a structured process. This had negative consequences, such as difficulty to prioritize tests, reevaluating tests, monitoring progress, and so on. The attempt is to design HT in a way of not having a strict process but a semi-structured process

DOI: 10.1002/smr

that adopts strengths of both the approaches. Thus, considering the structure of ISO/IEC 29119 in conjunction of ET strength-free exploration, we aimed to provide HT a semi-structured process that would have a formal structure with free exploration being a part it. We also achieve this by allowing flexibility in work flows. The process is also designed so that practitioners are able to decide which activities are emphasized, depending on testing outcomes, type of systems, and type of tests. Further, the process is iterative in nature.

- *Customer satisfaction* From Sections 2.1 and 2.2, we observe that customers are very reluctant with ET, while they are satisfied with ST. The primary reason of customers not being satisfied with ET is the lack of a formal test design phase, on which they can evaluate their product, and which can be used for to document the fulfillment of contractual requirements. In the HT process definition, the attention is given to make such a process, which could satisfy the customers. Therefore, we include the definition of test design phase that could allow to overcome the reluctance of customers. This may help the HT process to be useful for legal requirements and acceptance/release testing. In addition, it also allows test managers to have control of their testing activities.

4.2. Collaborative design

We codesigned the HT process with the help of practitioner feedback. The practitioner feedback was collected by conducting semi-structured interviews with testing experts.

We conducted four face-to-face semi-structured interviews to receive feedback on the mapping of the strengths and weaknesses of ST and ET, and also for the proposed HT process. Here, we should mention that the development and refinement of the HT process was an iterative process considering the feedback of the interviewees.

Two of the interviewees are working for Logica AB (Sweden) as a test manager and a project manager. The other two interviewees work as test managers for Maquet Critical Care AB (Sweden) and Toolaware (Sweden). Three interviewees being involved in the collaborative design have also participated in the interviews.

A data collection instrument was designed to receive feedback and suggestions for the proposed HT process (APPENDIX A Questionnaire 5). To assure the quality of the instrument, all the questions were cross-checked by the authors of this paper. All the interviews were presented with the RQs before the interviews. A number of scenarios were shown in order to validate or grasp the improvement opportunities in the HT process. Approximate duration of each interview was between 30 and 45 min. The data were collected manually by taking notes and also by recording with the consents of the interviewees. The data collected were transcribed, and the irrelevant materials were omitted (i.e., the key points of the interview were separated from the general discussion).

The feedback given by the practitioners, as well as how it has been utilized in the process definition, is presented in the following:

Feedback of interviewee 1: Interviewee 1 suggested that the strengths and weaknesses of both test approaches were concise and detailed. Her concern was how in reality the strengths of each testing approach will work out on real projects and provide benefits. She added that the weaknesses of ET and ST were generic, and that in practice, there could be many ways to deal with such issues by other means. However, she affirmed providing a solution inferred from strengths of both test approaches and found attempting to resolve the weaknesses in this way as quite innovative. She also had some reservations on the debriefing session because she considered that managing the test team might even take more time because of having debriefing session. She recommended involving test leaders in HT process. *Reflection on feedback*: the debriefing session was not removed based on the feedback by the practitioner, the reason being that Interviewee 4 provided useful suggestions of how to utilize the debriefing session better. Overall, the practitioner agreed with the main idea of formation of HT process keeping the previously mentioned context as no further changes were suggested. We highlight that when executing the process, the suggestion of the practitioner should be followed to involve test leaders.

DOI: 10.1002/smr

Feedback of interviewee 2: Interviewee 2 said that mapping the strengths of both testing approaches to the weaknesses was a good way to compare both testing approaches. He mentioned that mapping was an ideal way of presenting the solution based on theoretical constructs, but practically, this mapping might not provide with 100% solution. He stated that it was a high-level presentation of strengths to weaknesses, but still all strengths of both test approach might have several weaknesses that may be associated with other indirect measures. He said that RBTC should only be used complementary, specifically where GUI testing was required, and test cases were hard to codify. *Reflection on feedback*: given our design, RBTC is complementary and can always be combined with free exploration, which indicates that our design addresses the practitioner's concerns. As the practitioner highlights, different emphasis might be given depending on the type of testing conducted (e.g., GUI testing).

Feedback of interviewee 3: Interviewee 3 highlighted that mapping strengths to weaknesses was an appropriate way of defining a compromise process based on ET and ST. He evaluated the mapping process and mentioned that the approach was quite elaborative. When we presented him with the initial process flow description, he added that he was not fond of flow boxes connected to each other telling him what to do, and he was of the opinion that the context should decide which box should be used in a specific situation. He also recommended the introduction of free exploration in order to learn about the application, that is, before, after, or during the execution of RBTC. He added that free exploration would provide an edge to the testers as they would be able to immediately look for any major abnormality in a very short span of time. *Reflection on feedback*: the flow boxes were retained for the purpose of presenting the process in this paper. It is important, however, to illustrate the flexibility of the flow through the process, which makes it semi-structured as pointed out earlier. Hence, formal descriptions (flow boxes or activity diagrams) might not be suited to represent the process to practitioners. Rather, a narrative form should be preferred. Free exploration has been emphasized in our process more based on this interviewee's feedback.

Feedback of interviewee 4: Interviewee 4 was of the opinion that there should be more flexibility in using any sort of test cases, not only RBTC. He also suggested that these RBTC should be made more generalized, and one should not limit to RBTC only.[‡] He said that it should be up to the testers or managers to decide upon what they need and require out of testing. And, he highlighted that performing ET at the beginning of testing life cycle could provide many benefits, and therefore, it should also be incorporated in the HT process. He pointed out that exit criteria should be explicitly discussed. He also recommended that upon the conclusion of every debriefing session, more test missions should be drafted based on the testers report and intuitions and that these newly devised test missions should become the input for further session executions. *Reflection on feedback*: The flexibility of the process is illustrated by showing different alternative paths through the process. Furthermore, the debriefing session is retained for the purpose specified by the interviewee.

After evaluating the mapping and the HT process, the HT process was refined based on the feedback received.

4.3. Defined hybrid testing process

Considering the design rationales, as well as the feedback by the practitioners, the brief descriptions of each subprocess in HT (Figure 3) are given in the following paragraphs:

- *Test planning*: The purpose of test planning in HT process is to plan, document, and communicate all the necessary and required information to all the stakeholders about what is going to happen regarding testing. HT test planning is inherited from the ST process. In order to have an improved planning process, the strengths of ET planning are also incorporated. These include specification of the scope and time, allocation of resources, risk planning for risk management, and mitigation.
- *Test mission design and implementation*: HT test design, introducing the RBTC [41] and test missions would help in enabling high functionality coverage and defect detection effectiveness in addition to cost-effectiveness through reducing the test bed size. The RBTC specify those test cases that are defined only from the requirement specification. The 'test mission' is a concrete instruction for testing and the problem being looked for.

DOI: 10.1002/smr

- *Test environment setup*: For HT, there is a freedom for the selection of test environment. Based on the test case design and implementation, the test environment in which the test will be executed is established and maintained.
- *Test execution*: Both RBTC and the test missions are executed, which were designed in test design phase. First, a tester has given the freedom to freely explore the application in order to learn and obtain knowledge about it. After that, RBTC and then the test missions are executed, and the execution artifacts are recorded. A session is a particular time slot assigned to a specific test

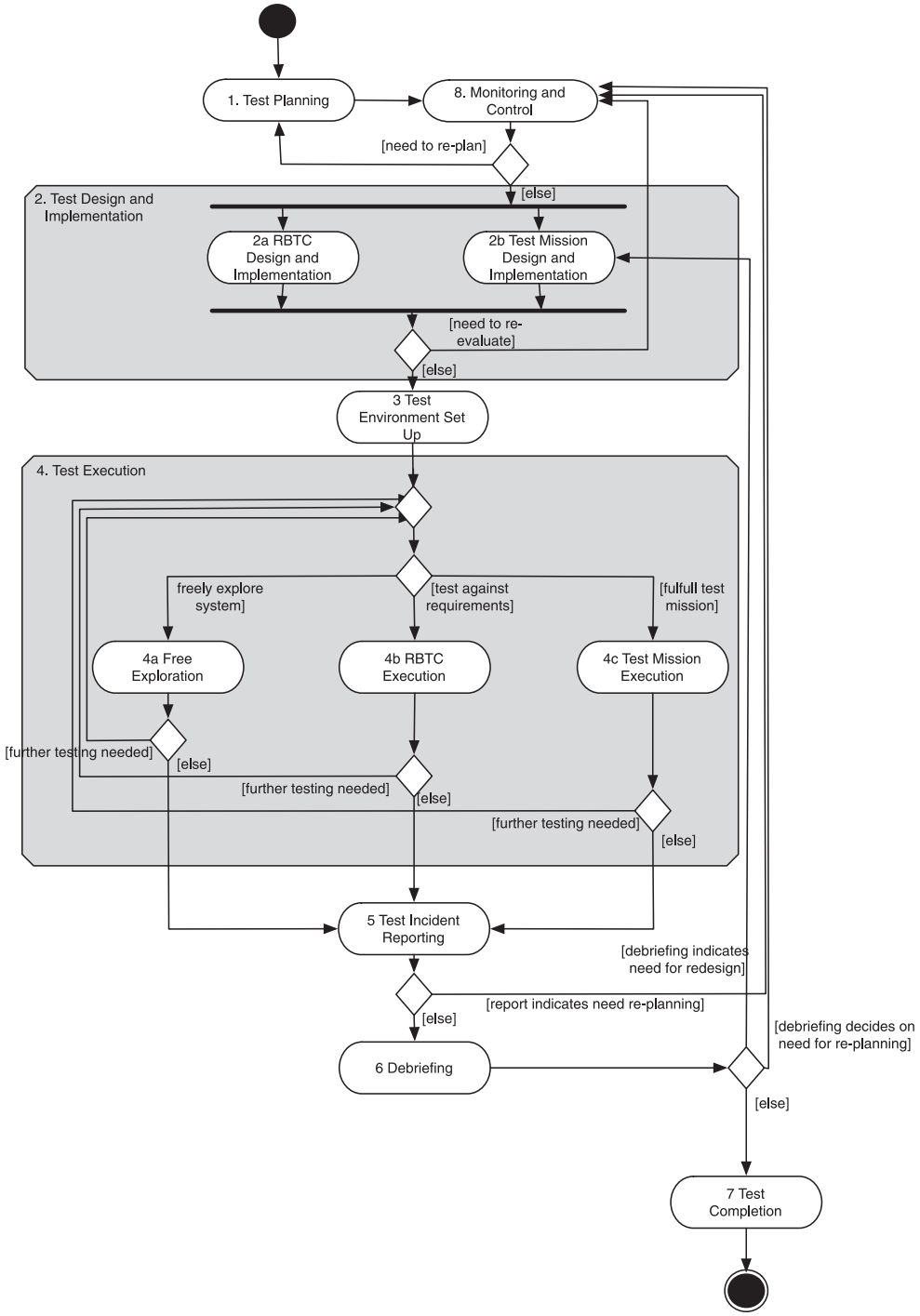

Figure 3. Process of hybrid testing.

DOI: 10.1002/smr

mission in which test mission has to be executed. A session time is an uninterrupted block of test time. A session time may last from 30 to 90 min.

- *Test incident reporting*: The purpose of test incident reporting in HT is to report the issues identified in the test execution to the relevant stakeholders in order to conduct further actions on the reported problems. The session sheet taken from the ET is used to report all incidents happened during the testing, and it has information about tested area, test notes, issues, faults, bugs, failures relevant information, or any other ambiguities related to the functionality. This provides focused documentation related to the testing with all relevant information.
- *Debriefing*: The purpose of debriefing session in HT is to obtain the input of a tester on the test mission, which was assigned to him, and to discuss about his observations. A debriefing session should also provide coaching to the tester regarding further test activities that needed to be performed. If required, a debriefing session can lead to the derivation of many test missions. After the completion of session, a debriefing session is set up between the tester and a test lead.
- *Test completion*: The purpose of test completion criteria is to make sure that the useful test assets such as test plans, test cases, and session sheets are made available, and all the results are documented, recorded, and communicated to the relevant stakeholders. Test completion criteria are met when an agreement has been reached that the testing being performed and managed is complete.
- *Test monitoring and control*: The purpose of HT monitoring and control is to ensure whether all the activities as specified in test plan are aligned with the actual execution of those activities. Hybrid testing monitoring and control provides assurance of whether or not the testing being performed is in line with the defined test plan. All the processes within the HT process, that is, tests design, test execution, test incident reporting, and test completion are being monitored and controlled.

The flow of the process is designed to be flexible and iterative (Figure 3). In the beginning of the process, test planning influences monitoring and control (e.g., which test targets should be monitored), while defining the targets test planning can be influenced and refined.

After having specified the plan and how to monitor and control, test design and implementation are conducted, and both RBTC design and test mission design are executed. With these activities completed, the outcome can be monitored and controlled, and eventually updates are made in the designs.

Thereafter, the test environment is set up. This is the prerequisite to conduct test execution. The test execution part is highly flexible. One can, for instance, start with an exploratory session, followed by test mission execution and RBTC. Another scenario is to only do free exploration. How much effort is spent and how many executions of the particular activities are conducted are not pre-specified and might vary with the testing context (e.g., type of testing performed or the type of system to be tested).

After having completed the test execution, test incidents are reported, and debriefing is conducted. At any point, one can return to the monitoring and control activity and, depending on the outcome, decide on how to continue in the process. That is, it is possible to continue at any point in the process after completing monitoring and control. We have not illustrated this in the Figure to sustain its readability.

5. THREATS TO VALIDITY

Because the HT process definition is based on the results of the SLR and interviews, the validity threats for each indirectly influence the validity of the proposed HT process. The internal and external validity threats for the SLR, the interviews, and the experiment are discussed in the following paragraphs.

5.1. Systematic literature review

For the SLR, one of the validity threats was associated with the possibility of missing any important publication. In order to eliminate this threat, when designing the search strings, we used the synonyms and alternative terms for the keywords referring to linguistic dictionaries while limiting them within the context of software engineering. When deciding on the keywords, we also checked the general terminology used in the testing field (e.g., testing standards such as ISO/IEC 29119 and key publications) not to miss any important keyword. The search strings were verified by conducting trail searches, and a preliminary search was carried out in order to identify the relevant literature by the help of the BTH

DOI: 10.1002/smr

(Sweden) librarians. Furthermore, we asked an expert in the area to review the design of the literature review as well as the list of included papers after the review to make sure that no important study is missed.

The quality of the data extraction form was checked by one of the other authors of this paper, who was not in the review team. The reviewer checked, in particular, the relevancy of the data to be extracted as well as whether any important information that needs to be captured is missing. Then, the forms were slightly revised after the pilot searches to include categories of relevant areas to study that helped in the uniformity of the coding.

To avoid selection bias during the selection process of the primary studies, two reviewers worked together to decide on the inclusion and exclusion of the studies. In addition to this, we also asked an external reviewer to check the final list of primary studies included in the SLR. As for the analysis phase, one threat could have been an individual bias when identifying the codes and the main categories for the strengths and weaknesses. In order to reduce this threat, a pair of reviewers worked together and identified the constructs after joint discussion.

5.2. Industrial interviews

For the interviews, the possibility of missing any important question in the questionnaires was one of the potential validity threats. In order to avoid this, we designed the questionnaires based on the findings of the SLR. Furthermore, we also included open-ended questions to identify additional strengths and weaknesses by letting the interviewees discuss their experiences.

Another threat could be the misinterpretation of the question and answers during the interviews. This threat was minimized by reviewing of the questionnaire. A number of senior software engineering students studying at BTH (Sweden) were asked to review the questions for ensuring the clarity of the meaning before conducting the actual interviews. A recording device was used to record the interviews, and the transcribed interviews were shared with the interviewees to avoid any misunderstanding.

Another threat was related to the fact that the data were gathered in the form of qualitative information during the interviews. A risk of misinterpretation of qualitative data exists because of the possibility of multiple interpretations. This risk was reduced by cross-checking the findings and also by getting feedback on our interpretations from the interviewees (member checking).

During the analysis phase, the two authors who also performed the SLR analyzed the interviews. To avoid researcher bias, another author of this paper made an independent analysis. The results were cross-checked, and then after a discussion, the codes, the main categories, and the connections in between the main strengths and weaknesses were agreed upon, solving very few disagreements also by consulting the interviewees.

There is also a threat to external validity because of a low number of interviewees. It was essential to involve practitioners with a vast amount of experience in ST and ET, as this provides the greatest potential to obtain additional experience-based insights complementing the results of the literature review. This constraint limited the number of persons we could involve in the research process. Overall, it was a trade-off between the levels of experience of practitioners versus the number of practitioners involved. It is important to highlight that for P1 and P2, both the literature review and the practitioners, complement each other. Having only one source would increase the risk of losing valuable information. Using source triangulation reduces the threat related to the number of responses. We required detailed and qualitative insights to design our HT process; therefore, we chose a qualitative data collection instrument (interview) over a sampling-based instrument (questionnaire). In S3, the HT process was codesigned with the practitioners, also involving both sources (practitioners and literature). The practitioners only had one contradiction in opinion of how to design the actual process (i.e., whether to have a debriefing session or not). Other suggestions were valuable complements to our suggested process (e.g., what to emphasize in GUI testing).

6. DISCUSSION AND CONCLUSION

The conclusion is divided into two parts. The first part summarizes the results, and the second part presents the implications for practitioners and researchers.

DOI: 10.1002/smr

6.1. Summary of findings

This study has mainly two contributions. First, the strengths and weaknesses of ST and ET were identified. Second, by bringing into light the improvement opportunities for a new testing process through unification of ST and ET in a compromise form, an HT process was defined in collaboration with practitioners.

What are the strengths of ST and ET?: The identified strengths and weaknesses were recognized under four main categories: (i) testing quality (defect detection effectiveness/functionality coverage); (ii) nature of the process (structure/flexibility); (iii) cost-effectiveness; and (iv) customer satisfaction.Major strength categories for ST were found to be related to the nature of the process, testing quality, and customer satisfaction. The structured and guided process of ST provides benefits such as repeatability of the tests, reusability of the test cases, early quality assurance, oracles availability for validating the testing quality, better risk management, independency from the testers' skills, and automation of the testing process. Moreover, good functionality coverage and increased customer satisfaction during product acceptance are two other identified strengths.As for ET, cost-effectiveness, the nature of the process, and testing quality were the main strength categories recognized. Exploratory testing was stated to be cost-effective because of less time being spent on documentation (i.e., focused documentation for only logs, test notes, and videos after the execution), better resource utilization, rapid feedback, and quick learning of the product. As for the testing quality, better defect detection effectiveness, better regression testing, and more critical bug detection were found to be the major strengths. Because the process of ET is flexible, the skills of testers are better utilized as they can freely explore the defects; and thus, the testers become more responsible, engaged, motivated, and creative, while they are performing the tests. Tables V and IV provide an explanation of the strengths.

What are the weaknesses of ST and ET?: For ST, major weaknesses were found to fall under testing quality, nature of the process, and cost-effectiveness categories. One of the major weaknesses was identified as the dependency of testing quality on the test case design, which depends on the skills, experience, and the domain knowledge of the designer as well as the previously produced documents. Testers, being not free to make decisions even if they see the problem about the test cases, were another weakness attributed to inflexibility of the test process. As for the cost-effectiveness, ST found to be time consuming and costly as it requires designing, documenting, executing, and managing large numbers of test cases, which should also be updated continuously in the software development life cycle as the requirements change. Moreover, the cost increases if test cases require revision and/or redesign in cases of low quality design.

On the other hand, major weaknesses of ET were identified as related to the nature of the process, testing quality, and customer satisfaction categories. The unstructured and ad hoc processes are found to cause difficulties in managing the testing process and risk, in prioritizing and selecting the appropriate tests, and in repeating the tests. Moreover, these also, in turn, create the fear of losing control over testing. As for testing quality, the dependency on the skills, experience, and domain knowledge of the testers are among the major weaknesses identified. These become more significant especially when the application to be tested is too complex. In addition, ET found to be not suitable for acceptance, performance, and release testing, which in turn lowers the accountability and hence customer satisfaction. Tables VI and VII and provide an explanation of the weaknesses.

What are the improvement opportunities for testing processes by addressing some major weaknesses of ST and ET through unifying their processes in a hybrid form?: The second contribution of this study is the identification of the improvement opportunities for the testing process through unification of ST and ET into a resultant HT approach. We defined the HT process considering ISO/IEC 29119, which is an upcoming software testing standard. The industrial evaluation of the proposed HT process was performed through interviews in industry. The practitioners stated that the HT process has merits to resolve some major issues of ST and ET test approaches and invited us to their companies for dynamically validating the HT process. The details of the identification of improvement opportunities through mapping ET and ST strengths and weaknesses to each other are provided in Section 3 and in Tables IX and X.

DOI: 10.1002/smr

Our study contributes to highlight the importance of experience. In order to further understand the merits of HT, we recommend to take the following actions. First, experiments have to be designed and the performance of testers with different experience levels for the different testing approaches has to be compared. Second, experience shall not be treated as a variable stating total experience in years. Instead, experience should be broken down in different kinds of experiences (e.g., programming, testing, and methodologies) relevant to testing to understand its impact on ET and HT processes. Furthermore, we plan to evaluate the hybrid testing process in further trials through action research.

6.2. *Implications for research and practice*

We discuss the implications for research and practice the findings from two perspectives, practitioners and researchers.

- *Practitioners*: Given the analysis of strengths and weaknesses of ST and ET (P1 and P2), a clear need has been established for hybrid processes. This leads to the proposition that practitioners can benefit from using a hybrid development process, hence, utilizing the strengths of both types of processes and addressing the weaknesses. The hybrid process presented in this paper is flexible baseline (indicated by different paths one can take through the process) of an HT process. The process has been codesigned with very experienced practitioners knowing both, ET and ST. This study makes their experience, as well as the experience reported in literature, accessible to other practitioners. Practitioners are now in need to adopt and refine the process in practice, as this is the prerequisite to extend and mature it. In particular, empirical evidence provided on the potential and usefulness of a hybrid process could speed up the technology transfer of HT processes. In particular, we found that there is an increasing trend of publications related to ST and ET studies discussing strengths and weaknesses, indicating that with evidence, the interest in adoption and evaluations increases.
- *Researchers*: We presented an approach that uses systematic review and practitioner input to design a new solution (HT process), the approach being based on the technology transfer model by Gorschek *et al*. [18]. Researchers might find the approach valuable in designing solutions combining evidence-based methods (here systematic review) and practitioner input in an exploratory way. The HT process needs further evaluation. Researchers hence should focus on conducting empirical studies with industry practitioners putting the process into action. In particular, researchers should evaluate the variances of the test process (e.g., testing with and without debriefing), how the activities and the flow through the process should differ for different types of testing (e.g., which activity in test execution is emphasized in terms of effort spent and number of executions depending on the type of testing, such as GUI testing versus unit testing), and what the longitudinal effects are of using an HT process. For these future activities, our research laid the foundations to continue such research.

6.3. *Future work*

In future work, we highlight the importance to evaluate the HT process first in controlled experiments and in industrial environments.

An experimental setup should focus on comparing ET, ST, as well as HT in relation to testing effectiveness (ability to identify critical defects) and efficiency (time needed for test design and execution).

Industrially focused studies need to focus on practitioners executing the process and learning how the process is tailored based on the context (e.g., different organizational test policies, types of system, and so forth). Earlier, we mentioned two types of tailoring, namely process structure (activities to be executed) and process flow (order of activities and relative effort spent on them).

DOI: 10.1002/smr

APPENDIX A: Interview guide

A.1. QUESTIONNAIRE 1: WEAKNESSES OF ET

- Q1. Have you come across with oracle issues while performing ET? (If yes, then how you cope with them)
- Q2. Have you experienced test coverage issues while performing ET? (If yes then how you cope with them)
- Q3. In what scenarios do you think ET is not preferred on other testing approaches?
- Q4. Do you think it is difficult to prioritize tests in ET?
- Q5. Have you experienced any issue related to planning and managing of testing tasks in ET?
- Q6. Do you consider tracking of tasks as an issue in ET?
- Q7. How you define the quality of testing keeping ET in context?
- Q8. Do you think ET is sufficient in determining the quality of testing? (Please specify)
- Q9. Have you experienced any problems while performing regression testing in ET? (Please specify)
- Q10 Do you think it is difficult to perform the test reviews?
- Q11. Have you come across with any test repeatability issue in ET and how you cope with it? (Please specify)
- Q12. Please list down problems related to ET, which you have experienced?
- Q13. How do you think that a free exploration affects the testing?
- Q14. Do you think managers are reluctant in formal implementation of ET approach for testing? (Please explain)

A.2. QUESTIONNAIRE 2: STRENGTHS OF ET

- Q1. What factors do you think, which can make ET more beneficial?
- Q2. Do you think that ET is more efficient in defect detection as compared to other test approaches? (Please specify the reasons)
- Q3. Do you think ET is an effective approach for investigation and isolation of defects?
- Q4. Do you think that ET is a cost effective approach? (Please specify)
- Q5. Do you find free exploration of application as an advantage?
- Q6. Do you think better regression testing can be achieved by utilizing ET? (Please specify how)
- Q7. Do you think ET is helpful in investigation of particular risky areas of software?
- Q8. Please list down the specific factors, which caused the need of introducing ET?
- Q9. What benefits you observed after introducing ET in your company?
- Q10. Do you think your customers are satisfied by the use of ET?
- Q11. Please list down perceived benefits of ET

A.3. QUESTIONNAIRE3: WEAKNESSES OF ST

- Q1. Do you think that designing of test cases is time consuming and an expensive activity? (If yes please specify)
- Q2. Have you come across with issues while revising existing test cases?
- Q3. In your experience have you found test case execution as an exhaustive process?
- Q4. Do you think ST is suitable for the regression testing?
- Q5. Do you think pre designed test cases are sufficient for the entire system life cycle?
- Q6. Do you think that test cases are durable (If not please specify)
- Q7. Do you think that redesigned test cases are sophisticated then old one (if yes please specify)
- Q8. Please list down problems related to ST that you have come across?

A.4. QUESTIONNAIRE 4: STRENGTHS OF ST

- Q1. Do you consider ST as an effective way of detecting and discovering faults?

DOI: 10.1002/smr

- Q2. Do you think ST is an effective way for formulizing and guiding the testing tasks?
- Q3. What benefits you think of, which can be achieved while designing and planning the tests before the actual test execution?
- Q4. Do you think test coverage is better achieved while using ST?
- Q5. Do you think that better tracking would be achieved through ST?
- Q6. Do you think ST improves the overall quality of testing?
- Q7. Do you think ST provides with better and reliable test results?
- Q8. Do you think that ST is beneficial where regulatory and legal requirements are to be fulfilled?
- Q9. Do you think that ST can predict reliability of software? (Please specify)
- Q10. Please list down benefit related to ST, which you have experienced?

A.5. QUESTIONNAIRE 5: VALIDATION OF THE MAPPING

- Q1. How you find mapping of weaknesses to strengths of both test approach?
- Q2: Do you think mapping process will lead to a better definition of a HT process?
- Q3: Do you consider that weaknesses are correctly mapped to the strengths?
- Q4: Do you think HT process based on these mappings have the tendency to solve the problems of both test approaches?
- Q5: What are your suggestions on the preliminary defined process of HT?
- Q6: Do you think that the preliminary defined HT process solves the weaknesses of both test approaches?
- Q7: How this HT process can be further enhanced in order to meet industrial needs?
- Q8: Do you think this HT process covers the weaknesses of both test approaches?
- Q9: Do you consider that this HT process will have some weaknesses associated with it?
- Q10: Please mention about the drawbacks associated with preliminary HT process definition?

ACKNOWLEDGEMENTS

We would like to thank all the participants in the study who provided valuable input in interviews. Furthermore, we thank the anonymous reviewers for valuable comments that helped in improving the paper. This work has been supported by ELLIIT, the strategic area for ICT research, funded by the Swedish Government.

REFERENCES

1. Agruss C, Johnson B. Ad hoc software testing, a perspective on exploration and improvisation. Technical report, Florida Institute of Technology, USA, April 2000.
2. Ahonen JJ, Junttila T, Sakkinen M. Impacts of the organizational model on testing: three industrial cases. *Empirical Software Engineering* 2004; **9**(4):275–296.
3. Ammann P, Offutt J. Introduction to Software Testing. Cambridge University Press: Cambridge, 2008.
4. Andersson C, Runeson P. Verification and validation in industry – a qualitative survey on the state of practice. In *International Symposium on Empirical Software Engineering (ISESE* 2002*)*, 2002; 37–47.
5. Bach J. Session-based test management. *Software Testing and Quality Engineering Magazine*, 2000; 2.
6. Bach J. Exploratory testing. In Veenendal EV (eds.). The Testing Practitioner. UTN Publishers, 2005.
7. Barnett-Page E, Thomas J Methods for the synthesis of qualitative research: a critical review. *BMC medical research methodology* 2009; **9**(1):59.
8. Bertolino A. Software testing research: achievements, challenges, dreams. In *Proceedings of the Workshop on the Future of Software Engineering (FOSE* 2007*)*, 2007; 85–103.
9. Bourque P, Dupuis R. Guide to the software engineering body of knowledge (swebok). Technical report, IEEE Computer Society, Los Alamitos, California, 2004.
10. Briand LC, Labiche Y, Lin Q. Improving the coverage criteria of uml state machines using data flow analysis. *Software Tests, Verification Reliability* 2010; **20**(3):177–207.
11. Britten N, Campbell R, Pope C, Donovan J, Morgan M, Pill R. Using meta ethnography to synthesise qualitative research: a worked example. *Journal of Health Services Research & Policy* 2002; **7**(4):209–215.
12. Copeland L. A practitioner's Guide to Software Test Design. Artech House: Boston, Mass., 2004.
13. Dixon-Woods M, Agarwal S, Jones D, Young B, Sutton A. Synthesising qualitative and quantitative evidence: a review of possible methods. *Journal of health services research & policy* 2005; **10**(1):45–53B.

DOI: 10.1002/smr

14. Do H, Rothermel G. An empirical study of regression testing techniques incorporating context and lifetime factors and improved cost-benefit models. In *Proceedings of the 14th ACM SIGSOFT International Symposium on Foundations of Software Engineering (FSE* 2006*)*, 2006; 141–151.
15. Dupuy A, Leveson N. An empirical evaluation of the mc/dc coverage criterion on the hete-2 satellite software. In *Proceedings of the Digital Aviation Systems Conference (DASC* 2000*)*, 2000.
16. Fraser G, Gargantini A. Experiments on the test case length in specification based test case generation. In *Proceedings of the 4th International Workshop on Automation of Software Test (AST* 2009*)*, 2009; 18–26.
17. Given LM. The Sage Encyclopedia of Qualitative Research Methods. SAGE: Los Angeles, 2008.
18. Gorschek T, Garre P, Larsson S, Wohlin C. A model for technology transfer in practice. *IEEE Software* 2006; **23**(6):88–95.
19. Grechanik M, Xie Q, Fu C. Experimental assessment of manual versus tool-based maintenance of gui-directed test scripts. In *Proceedings of the 25th IEEE International Conference on Software Maintenance (ICSM 2009)*, 2009; 9–18.
20. Grechanik M, Xie Q, Fu C. Maintaining and evolving gui-directed test scripts. In *Proceedings of the 31st International Conference on Software Engineering (ICSE 2009)*. IEEE Computer Society, 2009; 408–418.
21. Henderson-Sellers B, Gonzalez-Perez C, Ralyte J. Comparison of method chunks and method fragments for situational method engineering. In *th Australian Conference on Software Engineering (ASWEC* 2008*)*, 2008; 479–488.
22. Houdek F, Schwinn T, Ernst D. Defect detection for executable specifications – an experiment. *International Journal of Software Engineering and Knowledge Engineering* 2002; **12**(6):637–655.
23. Itkonen J. Do test cases really matter? an experiment comparing test case based and exploratory testing. PhD thesis, Helsinki University of Technology, Finland, 2008.
24. Itkonen J, Mäntylä M, Lassenius C. Defect detection efficiency: test case based vs. exploratory testing. In *Proceedings of the First International Symposium on Empirical Software Engineering and Measurement (ESEM* 2007*)*, 2007; 61–70.
25. Itkonen J, Mäntylä M, Lassenius C. How do testers do it? an exploratory study on manual testing practices. In *Proceedings of the Third International Symposium on Empirical Software Engineering and Measurement (ESEM* 2009*)*, 2009; 494–497.
26. Itkonen J, Rautiainen K. Exploratory testing: a multiple case study. In *International Symposium on Empirical Software Engineering (ISESE* 2005*)*, 2005; 84–93.
27. Jupp V. The Sage Dictionary of Social Research Methods. Sage publications Limited, 2006.
28. Juzgado NJ, Moreno AM, Vegas S. Reviewing 25 years of testing technique experiments. *Empirical Software Engineering* 2004; **9**(1-2):7–44.
29. Kaner C, Bach J, Pettichord B. Lessons Learned in Software Testing: A Context-Driven Approach. Wiley: New York, 2002.
30. Kaner C, Falk J, Nguyen HQ. Testing Computer Software. Van Nostrand Reinhold: New York, 2. ed. edition, 1993.
31. Kitchenham B, Charters S. Guidelines for performing systematic literature reviews in software engineering. Technical Report EBSE-2007-01, Software Engineering Group, School of Computer Science and Mathematics, Keele University, July 2007.
32. Murnane T, Reed K, Hall R. On the learnability of two representations of equivalence partitioning and boundary value analysis. In *Proceedings of the 18th Australian Software Engineering Conference (ASWEC* 2007*)*, 2007; 274–283.
33. Ng SP, Murnane T, Reed K, Grant D, Chen TY. A preliminary survey on software testing practices in Australia. In *Proceedings of the 15th Australian Software Engineering Conference (ASWEC 2004)*, 2004; 116–127.
34. Noblit GW, Hare RD. Meta-Ethnography : Synthesizing Qualitative Studies. Sage Publications: Beverly Hills, Calif., 1987.
35. Petersen K. Measuring and predicting software productivity: a systematic map and review. *Information & Software Technology* 2011; **53**(4):317–343.
36. Ryber T; Meddings P. Essential Software Test Design. Fearless Consulting: Stockholm, 2007.
37. Scanniello G, Fasano F, Lucia AD, Tortora G. Does software error/defect identification matter in the Italian industry? *IET Software* 2013; **7**(2):76–84.
38. Seidel JV. Qualitative data analysis. Technical Report www.qualisresearch.com, Qualis Research, Colorado Springs, Colorado, 1998.
39. Sharma M. S. C. B. Automatic generation of test suites from decision table – theory and implementation. In *Proceedings of the Fifth International Conference on Software Engineering Advances (ICSEA 2010)*, 2010; 459–464.
40. Shoaib L, Nadeem A, Akbar A. An empirical evaluation of the influence of human personality on exploratory software testing. In *Proceedings of the IEEE 13th International Multitopic Conference (INMIC* 2009*)*, 2009; 1–6.
41. Tahat LH, Bader A, Vaysburg B, Korel B. Requirement-based automated black-box test generation. In *Proceedings of the 25th International Computer Software and Applications Conference (COMPSAC 2001)*, 2001; 489–495.
42. Taipale O, Kälviäinen H, Smolander K. Factors affecting software testing time schedule. In *Proceedings of the 17th Australian Software Engineering Conference (ASWEC 2006)*, 2006; 283–291.
43. Thomson A. How to choose between exploratory and scripted testing, last accessed May 2013, url: http://www.stickyminds.com/sitewide.asp?function=edetail&objecttype=art&objectid = 6271.
44. Tinkham A, Kaner C. Lessons Learned in Software Testing: A Context-Driven Approach. Wiley: New York, 2002.
45. Vegas S, Juristo N, Basili VR. Identifying Relevant Information for Testing Technique Selection: An Instantiated Characterization Scheme. Springer: Heidelberg, Germany, 2003.
46. Yamaura T. How to design practical test cases. *IEEE Software* 1998; **15**(6):30–36.
47. Zotero. A reference management tool, project of Roy Rosenzweig Center for History and New Media, funded by the Andrew W. Mellon Foundation, the Institute of Museum and Library Services, and the Alfred P. Sloan Foundation. available at: http://www.zotero.org/.

DOI: 10.1002/smr

AUTHORS' BIOGRAPHIES:

Syed Muhammad Ali Shah is a PhD candidate at Politecnico di Torino in Italy. His research focuses on the analysis of the software testing techniques/approaches and analysis of defect data to calculate the reliability, characterizing the impact of software factors on quality. He received his MS degree in Software Engineering from Blekinge Institute of Technology (BTH), Sweden. He has research interests in software testing, software reliability, software process improvement, and empirical software engineering.

Cigdem Gencel is a senior researcher at the Faculty of Computer Science of the Free University of Bolzano, Italy. She received her PhD from the Middle East Technical University in Turkey, in 2005. Her research focus is providing novel solutions to real industrial challenges, particularly in the areas of software size and effort estimating, software measurement, software project management, requirements elicitation methods, and process improvement. She is a member of the International Advisory Council of COSMIC and a member of the COSMIC Measurement Practices Committee.

Usman Sattar Alvi holds an MS degree in Software Engineering from BTH, Sweden. He is currently working as a software quality assurance engineer in Seamless AB (www.seamless.se). His job responsibilities include validation and verification of software products and processes. Furthermore, his research interests include software testing, process engineering, software metrics, software requirements engineering, and software architecture.

Kai Petersen is a senior researcher at BTH, Sweden. He received his PhD from BTH in 2010. His research focuses on software processes, software metrics, lean and agile software development, quality assurance, and software security in close collaboration with industry partners. Kai has authored over 30 articles in international journals and conferences and has been the industry chair of REFSQ 2013.

DOI: 10.1002/smr