

Regression testing for large-scale embedded software development – Exploring the state of practice

Nasir Mehmood Minhas, Kai Petersen, Jürgen Börstler, Krzysztof Wnuk

SERL Sweden, Blekinge Institute of Technology, Karlskrona, Sweden
{nasir.mehmood.minhas, kai.petersen, jurgen.borster, krzysztof.wnuk}@bth.se

Abstract

Context. A majority of the regression testing techniques proposed by the research have not been adopted in industry. To increase adoption rates, we need to better understand the practitioners' perspectives on regression testing.

Objective. This study aims at exploring the regression testing state of practice in the large-scale embedded software development. The study has two objectives, 1) to highlight the potential challenges in practice, and 2) to identify the industry-relevant research areas regarding regression testing.

Method. We conducted a qualitative study in two large-scale embedded software development companies, where we carried out semi-structured interviews with representatives from five software testing teams. We did conduct the detailed review of the process documentation of the companies to complement/validate the findings of the interviews.

Results. Mostly, the practitioners run regression testing with a selected scope, the selection of scope depends upon the size, complexity, and location of the change. Test cases are prioritized on the basis of risk and critical functionality. The practitioners rely on their knowledge and experience for the decision making regarding selection and prioritization of test cases. The companies are using both automated and manual regression testing, and mainly they rely on in-house developed tools for test automation. The challenges identified in the companies are: time to test, information management, test suite maintenance, lack of communication, test selection/prioritization, lack of assessment, etc. The proposed improvements are in line with the identified challenges. Regression testing goals identified in this study are customer satisfaction, critical defect detection, confidence, effectiveness, efficiency, and controlled slip through of faults.

Conclusions. Considering the current state of practice and identified challenges we conclude that there is a need to reconsider the regression test strategy in the

companies. Researchers need to analyze the industry perspective while proposing new regression testing techniques. The industry-academia collaboration projects would be a good platform in this regard.

Keywords:

Regression testing, practices, challenges, goals, multi-case study

1. Introduction

Testing is an essential aspect of any software development project. The companies are facing various testing challenges, especially for large and complex products [1, 2]. The most challenging testing activity in large-scale systems development is regression testing, which can consume up to 80% of the total testing cost [3, 4]. Regression testing (RT) is a retest activity to ensure that system modifications do not affect other parts of the system and that the unchanged parts of the system are still working as they did before a change.

RT is a challenge for the software industry, especially for large-scale embedded software development companies in the context of systems with continuous integration and delivery. Considering the recent systematic literature review studies conducted on the topic of regression testing [5–8] we can conclude that it is a well-researched area and a large number of techniques have been proposed in the literature. Despite extensive research on RT, research results are not finding their way into practice. There are several reasons, like differences in terminology, availability of research results and a lack of empirical evaluation of RT techniques in real industrial environments [4, 9].

Other factors that affect the transition of research to the industry are, communication-gap between practitioners and researchers, consideration of testing challenges at the different level of abstraction, and differences in the vocabulary and context regarding the concepts of testing [10]. It is essential that researchers must consider real-world situations. Without a focus on real industrial needs and practices, a majority of new techniques proposed in the literature will not fit with existing practices. Testing is not only a technical challenge, but it is a socio-technical issue [1].

In this paper, we have investigated the current state of regression testing practice in two large-scale embedded software development companies. The contributions of this study are as follows:

- Regression testing definition: presenting the perspective of practitioners.
- Regression testing practices: the purpose was to understand that how practitioners undertake the regression testing activity in the companies. Along with the regression testing practices, we identified selection and prioritization criteria, and the challenges regarding the regression testing practices.
- Improvement suggestions to overcome the identified challenges.

- Regression testing goals and success criteria.

The remainder of this paper is structured as follows. Section 2 discusses the related work. Section 3 describes our research methodology. Threats to validity are discussed in Section 4. Results of the study are presented and discussed in Section 5. A summary, organized along our research questions, as well as conclusions can be found in Section 6.

2. Related Work

There is a large body of research on software testing ([11, 12]), but most of this research is focused on methods and tools. There is only little work on the perspectives of testing practitioners in industry on regression testing. We found only 13 articles [4, 5, 13–23] related to our work, which we discuss further below. Out of these 13 studies, eight deal specifically with regression testing, whereas five have a broader focus. Other than regression testing specific studies, we included those papers in the related work, where authors are discussing any aspect of regression testing.

With respect to the main focus of the related works, we organized them into four groups: (1) general testing practices, (2) testing methods and tools, (3) automated testing, and (4) regression testing.

General Testing Practices. Two studies, both of them surveys, investigated general testing practices ([13, 14]). Dias-Neto et al. [13] conducted a survey-based study to identify the software testing practices in South America. The study was carried out with the practitioners from Brazil and Uruguay. The authors highlight the testing practices that the practitioners consider as essential and beneficial. For instance, the practitioners think testing documentation as useful for current and future testing activities. The practitioners acknowledge the importance of test management and error reporting tools. The essential testing types are system and regression testing. The authors also highlight some weaknesses of the companies. For example, they indicate that the companies do not measure their testing activity and the tendency of using test automation tools is not encouraging.

Kassab et al. [14] conducted a web-based survey to explore how software practitioners are using testing. The focus of their research was to study the overall testing practices. Authors indicate that the use of black box testing techniques is more common as compared to white box testing. Regression testing is one of the testing levels getting more attention of the companies. There is a trend in the companies to outsource the regression testing activity. Among the surveyed companies majority of telecommunication and gaming companies prefer the outsourcing of regression testing, and they are satisfied with this practice. The authors highlight requirement coverage as the most used metric in the companies, followed by the test execution rate. Test stopping criteria for the majority of the companies is the deadline of the project.

Testing Methods and Tools. Ng et al. [23] investigated software testing practices in ICT companies in Australia. Their focus was on testing methodologies/techniques, tools, metrics, and standards. The authors highlighted that the training share for testing staff is low, according to the results presented in the study, universities and training colleges offer only 10.7% of the total training courses for testing staff. Regarding regression testing, the authors report that 69.2% of the organizations are using regression testing for all of the applications they are developing. Regarding the frequency of regression testing, 53.3% of the organizations repeat regression testing for every new version of the product and 28.9 % reported the use of regression testing after every change.

Automated Testing. Two studies focused on the state of practice in testing automation ([18, 21]). Rafi et al. [18] highlighted the benefits and challenges of automated testing using a systematic literature review followed by a practitioner survey. They found only few studies discussing the benefits and limitations of automated testing. The authors conclude that automation is beneficial in an environment where excessive regression testing is performed and it helps in improving test coverage. The key limitations are initial setup costs, training, and availability of reliable testing tools. Furthermore, a majority of testers believe that automated testing cannot replace manual testing.

Kasurinen et al. [21] studied the current state of practice and required improvements in software test automation in an interview-based empirical study. The authors suggest that most of the practitioners are satisfied with their current test policy and testing practices, and they are not thinking of any change in it. Regarding automation, the authors reveal that only 26% of the test cases were automated, and the majority of these test cases are related to unit and regression testing. The automation of regression testing is a common practice among the companies, and regression testing was the most practiced testing type in the sample organizations.

Regression Testing. Eight studies focused specifically on regression testing aspects ([4, 5, 15–17, 19, 20, 22]). Brahneborg et al. [15] extracted the challenges corresponding to the existing methods of regression testing from a set of empirical studies. The authors classified the challenges into two categories, 1) method related challenges, 2) Organization related challenges. Among method related challenges they highlighted, handling failures, performance measurement, handling fault distribution, scalability of techniques, and tool support. Whereas regarding organization related challenges, the authors describes existence of a structured test suite (test suite maintenance), information availability, knowledge and skills of testers and developers, and management support.

Minhas et al. [16] conducted a focus group based study to investigate the views of industry practitioners and software engineering researchers concerning regression testing. The authors explore the perceptions of both communities about the scope of regression testing. They also identify the regression testing success goals. The authors listed confidence, high precision, and fault-slippage as the essential goals of regression testing. They conclude that the perception of

the practitioners and researchers about regression testing is alike, and there are similarities in views concerning regression testing goals. Finally, the authors indicate the need for a measurement of regression testing task to enable the testers for measuring their success. The goals identified in the study are given in Table 2.

Parsons et al. [17] conducted case-studies and an online survey to investigate the adoption of regression testing strategies in agile development. The authors analyzed different contextual factors that can have an impact on regression testing strategy. The focus of their research was to identify the organizational maturity regarding regression testing and operational aspects of regression testing in the surveyed organizations. The authors found that the maturity of the organization is the primary factor for successful regression testing. The authors conclude that organizations can get potential benefits of investments in regression testing. The authors highlighted the need for investigations in the areas of change and risk management regarding regression testing.

Yoo and Harman [5] surveyed the literature on regression test case minimization, selection, and prioritization. They specified the state of the art, trends and issues concerning these areas of regression testing. The authors conclude that the trend to evaluate regression testing techniques is getting a significant increase in the research. However, the majority of empirical studies are carried out with systems under test of less than 10,000 lines of code and test suite sizes of less than 1,000 test cases. They also found that almost 60% of the empirical studies on regression testing are using programs from the software infrastructure repository (SIR¹) ([24]). It indicates that evaluation of regression testing techniques in real industrial context is limited. The authors argued that there is a potential risk of over-fitting the research on regression testing techniques to the programs that are readily available. They suggested that for the future research on regression testing, researchers should opt for alternative data sources and focus should be on the transfer of technology to industry.

Juergens et al. [19] carried out an industry case study to highlight the challenges concerning regression test selection techniques when applied in manual system tests. They suggest that the testing effort could exceed the expected limit when applying selective regression testing techniques on manual testing. The authors also think that under-specification of manual tests can reduce the conclusiveness of results. Finally, they suggest strategies to improve the manual regression test selection.

Engström and Runeson [20] investigated regression testing practices and challenges and pointed out that a majority of the challenges are related to testability issues and good practices are related to test automation. The authors note that most findings of their study are related to testing in general and are not specific to regression testing. However, there was a consensus among the practitioners regarding the concept and importance of regression testing. Some key findings of this study are provided in Table 2.

¹<http://sir.unl.edu/portal/index.php>

Harrold and Orso [4] analyzed the state of research and practice in regression testing. The authors conducted the review of research on regression testing and informal survey with the researchers and practitioners. They concluded that hardly a few methods and tools proposed in the research are in use of industry. The authors identified various issues that are hindering the transition of proposed techniques to practice. They also highlighted the technical/conceptual issues, needed to be addressed by the research. Issues identified by Harrold and Orso are listed in Table 2.

Lin [22] conducted a literature-based survey on the regression testing research and recent practices. Their goal was to identify the gaps in regression testing research and its industrial practices. The author suggests that there are some gaps and efforts should be made towards the implementation of regression testing research in industry. Addressing these gaps should be the future direction in research.

Summary. A summary of related work is presented in Table 1. Of the 11 studies discussed above, only 6 focused on aspects of regression testing. Overall, the related work shows that there is a gap between research and practice. New techniques should have to be evaluated in the real industrial context. The researchers should work on technology transfer of the regression testing techniques.

In order to build a mapping between the existing literature and findings of our study, we extracted related information from some of the studies presented in the related work. Considering the scope and method the studies that could be regarded as closely relevant to our work are [4, 20]. Both studies are of exploratory nature and purpose to investigate the current state of practice concerning RT. We extracted information related to practices, selection and prioritization criteria and challenges from these studies. We did extract some information regarding the selection and prioritization criteria from the literature survey by Yoo and Harman [5]. For the regression testing challenges, we also utilized the study by Brahneborg et al. [15]. Regarding RT goals we extracted the information from [16, 25]. We have listed the key findings of these studies in Table 2. It is essential to specify that the literature findings are not exhaustive because finding practices, challenges, etc. from the literature was not the goal of this study.

3. Methodology

We conducted a multi-case study in two large companies to investigate the current state of practice regarding regression testing. Data were mainly collected through interviews, and the review of process documentation complemented the findings. Interviews provide an opportunity for direct interaction with the respondents and resolving issues with interpretations during the interview sessions. We choose to conduct semi-structured interviews, since it allows improvisation and exploration of the studied objects [26].

Table 1: Summary of related work. The first column indicates the subsection in Section 2 (GTP: General Testing Practices, TMT: Testing Methods and Tools, AT: Automated Testing, RT: Regression Testing).

	Ref.	Methods	Year	RT ¹	Focus of the study
GTP	[13]	Survey	2017	No	Characterizing the testing practices in South America.
	[14]	Survey	2017	No	Overall testing practices.
TMT	[23]	Survey	2004	No	Current state of testing practices, testing methodologies/techniques, testing tools, metrics, and standards in Australian ICT companies.
AT	[18]	SLR and Survey	2012	No	Benefits and challenges of automated testing.
	[21]	Interviews	2010	No	Current state and required improvement in test automation.
RT	[15]	Literature study	2017	Yes	Identification of regression testing challenges.
	[16]	Focus Group	2017	Yes	Regression Testing goals, information needs and metrics.
	[17]	Case study and survey	2014	Yes	Regression testing strategies and the factors that influence the adoption.
	[5]	Literature Survey	2012	Yes	A detailed analysis of trends and issues in regression testing concerning minimization, selection and prioritization.
	[19]	Case study	2011	Yes	Regression test selection challenges when applied to system manual tests.
	[20]	Focus group and survey	2010	Yes	Regression testing practices and challenges.
	[4]	Literature review and survey	2008	Yes	An analysis of the state of the research and the state of the practice.
	[22]	Literature survey	2007	Yes	Identification of gaps between the regression testing research and practice.

¹Whether the work focuses on regression testing aspects.

3.1. Research Questions

- RQ1 How is regression testing perceived by the practitioners in industry?
RQ2 What is the current state of regression testing practice in industry?
RQ2.1 How are the practitioners selecting and prioritizing test cases?
RQ2.2 What are the key challenges for the practitioners regarding regression testing?
RQ3 What are the possible improvements regarding regression testing practices suggested by the practitioners?
RQ4 How do the practitioners evaluate their success in regression testing?

3.2. Case Companies

We conducted this study from the platform of EASE² (An industry-academia collaboration project). The companies participated in this study are Sony Mobile Communications AB and Axis Communications AB, from hereafter we will

²EASE- the Industrial Excellence Centre for Embedded Applications Software Engineering <http://ease.cs.lth.se/about/>

Table 2: Literature findings on RT state of practice

Aspect	ID	Description	Ref
RT Practices	LPr1.	In industry test suite maintenance is largely manual.	[4]
	LPr2.	For RT, many organizations rerun all test cases (retest all).	[4, 20]
	LPr3.	A most common approach is running core set of test cases.	[4, 20]
	LPr4.	In large number of organization, test case are selected randomly on the basis of experience of testers.	[4]
	LPr5.	Mostly organizations use in-house build techniques and tools	[4]
	LPr6.	Some practitioners prefer to run as many as possible.	[20]
	LPr7.	Start RT as early as possible.	[20]
	LPr8.	Run RT before each release.	[20]
	LPr9.	Complete re-test for critical parts.	[20]
	LPr10.	Focus is on functional test cases	[20]
	LPr11.	Selection of test cases depends on the situation.	[20]
	LPr12.	The amount and frequency of RT depends upon the various factors.	[20]
Prioritization	LPc1.	Change.	[4, 5, 20]
	LPc2.	Cost.	[4]
	LPc3.	Running time.	[4]
	LPc4.	Criticality.	[4]
	LPc5.	Complexity of the test cases.	[4]
Selection	LSc1.	Change.	[4, 5, 20]
	LSc2.	Historical test data on test case effectiveness.	[4, 5]
	LSc3.	Timing data on the last time a test case was run.	[4]
	LSc4.	Traceability between requirements to test cases.	[4, 5]
	LSc5.	Situation based selection.	[20]
	LSc6.	Areas affected by the changes.	[20]
Challenges	LC1.	Identification of obsolete test cases.	[4]
	LC2.	Selection of relevant test cases.	[4, 20]
	LC3.	Test case prioritization.	[4]
	LC4.	Test suite augmentation.	[4]
	LC5.	Removing redundant test cases.	[4, 20]
	LC6.	Creating effective test cases.	[4, 20]
	LC7.	Manual testing (expensive and time consuming).	[4]
	LC8.	information maintenance.	[4, 15]
	LC9.	Test suite maintenance.	[4, 15, 20]
	LC10.	Test suite assessment.	[4]
	LC11.	Time to RT.	[20]
	LC12.	Balance between manual and automated RT.	[20]
	LC13.	Execution of automated RT.	[20]
	LC14.	Time to analyze results.	[20]
	LC15.	Management support.	[15]
Goals	LG1	Increasing rate of fault detection (effectiveness)	[16, 25]
	LG2	Increasing coverage of test suite (coverage)	[16, 25]
	LG3	Increasing confidence regarding the system reliability (confidence)	[16, 25]
	LG4	Identifying high risk (critical) faults	[25]
	LG5	Identifying change specific faults (effected areas)	[25]
	LG6	The customer should not find fault in the product (fault slip-page to the customer)	[16]
	LG7	Finishing RT in limited time and low cost (efficiency)	[16]
	LG8	Running most effective test cases (inclusiveness)	[16]
	LG9	Excluding non effective test cases (precision) [16]	[16]

use Sony and Axis to refer the case companies. Both Sony and Axis are large-scale embedded software development companies, and both are the collaborators in the EASE project. Sony is a multinational telecommunication company and producing smartphones and other smart products. The core part of the software

is used in different versions and variants of the devices. Axis is manufacturing network cameras and other surveillance devices, and they claim to be the inventors of the first network camera. Regarding the software for the products, the core platform is similar for all products. Both the companies require a concise effort regarding regression testing of their respective products. Reason being that both companies are producing their respective products for many years and they have several releases of these products. The software update is a regular activity for both companies.

Table 3: Overview of interviewees. Column *Perspective* refers to the area/branch the interviewee is working in.

	PID	Team ¹	Current Role	Exp ²
Company A	P1	SWSE	Test Architect & Team Lead	10
	P2	SWSE	Manager Verification	9
	P3	SWSE, FCS	Verification Engineer	13
	P4	FCS	Test Architect	10
	P5	FCS	Verification Engineer	10
	P6	FCS	Product Owner	11
Company B	P7	FWD	Tech Lead	6
	P8	FWD	Engineering Manager	8
	P9	FWR	Program Manager	2
	P10	PTS	Technical Coordinator	7
	P11	PTS	Test Area Maintainer	6

¹ SWSE (Software Security & Enterprise), FCS (Flash Core Security), FWD (Firmware Development), FWR (Firmware Release & Upgrade), PTS (Platform Storage).

² Experience in number of years the practitioner is working with the current company.

From Sony, the practitioners from two teams (Software Security & Enterprise (SWSE) and Flash Core Security (FCS)) participated in the study. These teams are working on two different parts of a product, and the software for both parts is developed within the respective teams. SWSE team is working on the enterprise part of the product, and performs testing for software security and enterprise at the system level. Testing for software security is automated, whereas testing for the enterprise part is manual. For test automation a third-party tool is used. FCS team is working on the boot & loader part and performs testing mainly at the component level. Testing for the loader is automated, whereas testing for the boot part is manual. The team uses an in-house developed tool for automated testing and currently working on complete test automation and the adoption of an open source tool.

From Axis, practitioners from three teams (Firmware Development (FWD), Firmware Release & Upgrade (FWR), and Platform Storage (PTS)) participated in the study. FWD team is responsible for the new product development and new feature additions to products. They perform testing at unit, integration, and system level. Mainly regression testing is automated except the testing of new features.

FWR team works on upgrades and new releases. Their primary responsibility is pre-release (Beta testing), performance, and stability testing. Performance

and stability testing is automated, but complemented by manual exploratory testing. Regression testing is regarded as input to release. PTS team is responsible for the storage part of the platform and performs testing at the unit and system level. For regression testing, the policy of the team is to run all test suits during nightly trials. The automation framework is common for the whole platform.

3.3. Data Collection

The data was mainly collected through interviews, and it was complemented by the process documentation provided by the companies.

3.3.1. Selection of Participants

Before the actual study, with the help of project contact persons, we conducted two separate workshops at both companies. The purpose was to present our research objectives to the managers and potential participants. We did highlight the required experience and roles of the potential participants. These workshops enabled us to understand the overall hierarchy of the companies, their team structures, and the working procedures. The selection of the participants was based on convenience sampling [27]. Convenience sampling is a non-probability non-random sampling method. We refer our selection as convenience sampling because we selected those participants who were fulfilling our selection criteria of role and experience, and who were willing to participate. A summary of the participants is presented in Table 3.

3.3.2. Interview Design

Based on the research questions, we prepared an interview guide consisting of open-ended questions. We did not restrict ourselves to the pre-defined questions, we added/improvised the interview questions during the sessions. The first author developed the interview guide, while the second author helped during the revisions, and an independent researcher reviewed the interview guide. The interview guide consists of seven sections: introduction, background, current practices, selection and prioritization, challenges and improvements, and success criteria. The complete interview guide is presented in Appendix A.

3.3.3. Interview Execution

In total, we conducted eleven interviews (five at Axis, and six at Sony) with the representatives of five teams. The first and second author participated in the interviews at Axis, whereas the first and fourth author participated in the interviews at Sony. The first author guided all the eleven interviews, while the second and fourth authors assisted in their respective sessions. Observer triangulation was utilized during all interviews. Besides the person guiding an additional researcher took the interview notes.

Each interview took about one hour and was audio-recorded with the consent of the participants. During the interview, we complemented our documentation with mind-maps and free text notes. The template of mind-map used for this

study is presented in Figure 1. The template consists of six nodes, five nodes represent the main topics of our study (i.e., current practices, selection and prioritization criteria, challenges, improvements, and evaluation), whereas one node of the mid-map shows the background information of the participant.

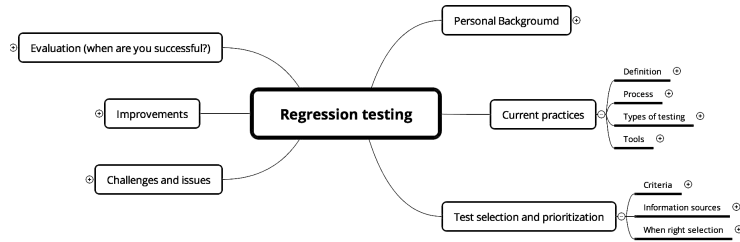


Figure 1: Mind-map used for data management and classification.

3.3.4. Process Documentation

The primary source of information for our study was the interviews, to complement our findings we requested both companies to provide us their process documentation under NDA agreement. The documentation was mainly related to test cases, test management tools, test strategy/plan, internal presentations, and training materials. We have undergone the review of these documents in detail and extracted the relevant information. The extracted information complemented some of the practices defined by the practitioners (reuse of existing test cases, when to create new test cases, at which level regression testing is performed in the companies, distribution of manual and automated work, and etc.), selection / prioritization criteria (criteria to select the scope, risks, effected areas, change related measures, and etc.), organizational goals related to testing, and the success criteria (i.e., when to stop regression testing?). The analysis of process documentation helped us to understand the testing procedures adopted at the companies, and to validate the findings of the interviews further.

3.4. Interpreting, Analyzing and Validating Interview Scripts

To minimize misunderstands and misinterpretations, we documented the interviews by three different means: structured mind-maps, free text notes, and audio recordings. The mid-map was the master document for the recording of interview results. Based on the research questions, we already structured the mind-maps according to the main themes of our study (See Figure 1). For the analysis of interview notes and audio transcribed transcripts we followed the steps defined by Cruzes and Dybå [28]. We also took inspiration from the method followed by the Petersen and Wohlin in their study [29].

Step 1: Transcribing the Interview Recordings. The first step was to finalize the interview scripts. In this step we transcribed the audio recordings, the first author transcribed all the audio records and the second author randomly verified

these transcripts. We did use the free text notes and process documentation as a separate source of raw data.

Step 2: Clustering of Categories. The second step was to cluster the raw data into meaningful groups. We used color coding to separate the concepts in the scripts and identified the similarities and differences in the themes of different scripts. For example, the statements were grouped according to the categories, “current practices”, “selection and prioritization criteria”, “challenges”, etc. As an example, we have listed some statements of interviewees in the in Table 4. We clustered the raw data separately for each participating team.

Table 4: Analysis procedure adopted for step2 and step 3.

SNo	Original Statement	Category	Subcategory	Restructured Statement
1	<ul style="list-style-type: none"> i) Working very close with our development team, Developers cannot merge any thing with out the approval of testers ii) Decision making approach: Before heavy documents and test plans. Scoped out a lot of overhead, now it is within the project, in collaboration with the developers and with the project manager and QA (decision within the development team). 	Practice	Process	Developers and testers collaborate while deciding the scope for the testing of changes and new features.
2	<ul style="list-style-type: none"> i) If we don't find issues with regression testing, we try exploratory to find something ii) We do some exploratory testing to find new issues. 	Practice	Testing type	Use of exploratory testing as an alternative to RT.
3.	<ul style="list-style-type: none"> i) Usually when we are closer to the freeze of the code deadline we try to make a little bigger scope. ii) We run full scope at the of development, when developers cannot make changes. iii) If there is a change in the core of the product we run the full scope. 	Practice	Process	Regression testing is performed with full/bigger scope at code-freeze, before release, or in case of changes in the core of the system.
4.	We tag our test cases like sanity, scope, mandatory, tagged test cases are suppose to be really important test cases.	Practice	Prioritization	Use of priority tags for the test cases with respect to the relevant importance.

Step 3: Clustering of Subcategories. In the third step, we assigned the labels (themes) to the statements already clustered in step 2. In this step, beside the labeling process, we also did restructure the statements where it was necessary. Table 4 presents the labels along with the restructured statements.

Step 4: Mapping categories and subcategories to mind-maps. In this step we mapped the results generated from free text notes and audio generated transcripts to the mind-maps. Resultantly we did the necessary updates in the min-maps.

Step 5: Generation of Results. From the final copy of mind-maps we generated Tables (5, 6, 7, 8, 9, 10, 11, & 12) of results according to the research questions presented in the section 3.1.

Step 6: Validation of Results. In qualitative studies, the researcher’s bias can influence the interpretation of the results. It can be avoided by validating the interpretations from the sources.

As a first validation step, we compared the results with the information extracted from the process documentation. Although the process documentation did not cover the complete results, however, at some extent it provided us confidence about the correctness of the results that we had collected through interviews.

To validate our interpretation of results, we conducted two workshops with the participants of our study and the representatives of the companies at EASE. We used validation sheets to record the issues regarding the interpretation of the results. Feedback of the study participants and companies representatives was positive, they identified a few minor revisions (mainly related to the RT practices), which we adjusted accordingly. Lately, we provided the final results to the companies representatives for the final verification, their response was encouraging, and they did not raise any issue in the final results.

4. Threats to Validity

This study is exploratory and based on the experience and perceptions of industry practitioners. The data was collected through semi-structured face-to-face interviews. We asked open-ended questions to capture viewpoints without any restriction. The purpose was to avoid researcher bias and get insight into current industry practices. Furthermore to triangulate the data sources, along with the interviews we also utilized the process documentation provided by the companies. In our discussion of threats to validity, we follow the guidelines by Runeson and Höst [26].

Construct Validity. This aspect of validity is regarding the underlying operational measures, concepts, and terms of the study. In our case, the selection of the participants regarding their competence and relatedness to the study area was a potential threat. Similarly, missing any critical aspect from the interview design could also be a threat. To mitigate these, we conducted pre-study workshops, the goal of these workshops was to explain the purpose of the study to the focal persons in the companies and resultantly to select appropriate participants for the study (Ref: Section 3.3.1). Regarding the interview design, we have carefully designed and reviewed the interview guide (See Section 3.3.2 and Appendix A).

Internal Validity. This aspect of validity threat is essential if causal relations are examined. Generally, we can state that studying causal relationships was not in the scope of this study. Therefore we did not consider this aspect of validity threat.

External Validity. This aspect of the validity refers to the generalization of findings. Participants of this study were representing five different teams of two large companies. The findings of this study represent the communication-based companies working on embedded software systems. To ensure the applicability in a larger population, we have linked our finding with the existing literature. Further analytical generalization of results is possible, to support this we have presented a detail discussion on the cases under study in Section 3.

Reliability. To ensure the reliability of the results, we assured the correctness of the collected data and interpretation of the results. Sections 3.3 and 3.4 present the detail in this regard. For each round of interview two authors participated. To triangulate what was said in the interviews we reviewed the process documentation of the participating companies. We did ensure the results triangulation, by involving multiple authors in the results interpretations, and we did validate the results in two workshops with the participants of the study.

5. Results and Discussion

This section presents the results regarding the practitioners' perspectives on regression testing, test case selection and prioritization criteria, challenges, improvements, and success criteria. The subsections are organized according to the research questions presented in Section 3.

5.1. The practitioners' Perceptions of Regression Testing (RQ1)

The purpose of our first research question was to elicit what participants think about regression testing. Our first interview question was “*What is regression testing for you?*”. Instead of standard definitions of regression testing [30, 31], our participants provided practical descriptions of regression testing with close relations to the perspectives (teams) they were representing.

From Table 5, we can see that the practitioners covered three aspect while defining regression testing: 1) The timing or frequency of regression testing, 2) the scope of regression testing, and 3) the overall goal of regression testing. Regarding timing, the participants define the need for regression testing after changes or fixes, before release, whereas some participants suggest running regression testing continuously. About the scope of RT, the practitioners are flexible, some of the participants describe running RT with smaller scope, whereas some prefer to adopt a “re-test all” policy. Finally, the participants are agree on the goal of RT, (i.e. to get confidence about the integrity of the system after changes, fixes or before the release). Difference in the initiation timing of RT and scope does not means that teams / companies are perceiving RT differently. In fact this represent that on which development level a team is employing the RT and how often they are making changes or adding new features to the releases.

We thus synthesized the views of all participants from both the companies and finalized the following definition:

Table 5: The practitioners’ definitions of regression testing, as response to interview question “What is regression testing for you?”.

CID ¹	PID ²	RT Definition
SWSE	P1.	To verify that introduced changes/fixes have not changed the behavior of the system in a negative way.
	P2.	Small sanity check on the changing part of system , try to figure out nothing has broken.
	P3.	Run selected test cases continuously , to check If something has broken or if everything has broken.
FCS	P4.	For every release , along with the testing of the new features, it is much important to make sure that old functionality is intact.
	P5.	To make it sure that everything still works, need to run a regression test for every new fix from developers. It could differ concerning what are the changes.
	P6.	It is a validation that primary use cases are working, regression testing is a hygiene thing to make sure that any basic functionalists have not broken. It is not expected to find much during regression testing.
FWD	P7.	To test what happened with other products, did changes destroy other products?
	P8.	Regression testing is a constant qualifier , and it is a process to test over and over.
FWR	P9.	To make sure that there is no regression, no degrades. Regression testing is input to release .
PTS	P10.	Regression testing is to verify that the functionality that was working previously, still works correctly or something has broken that was not expected while making changes .
	P11.	To verify that during changes or adding new features the overall quality of the functionality, performance of the database has not decreased.

¹ Company/team ID according to Table 3.

² Participant ID according to Table 3.

Regression testing is a repetitive activity which is applied to ensure that changes/fixes/upgrades did not affect the behavior of the system/product/product-line negatively and nothing was broken or destroyed in the functionality of the system/product/product line.

5.2. Regression Testing Practices (RQ2)

The objective of second research question was to understand how the practitioners conduct regression testing in their projects. Table 6 summarize regression testing practices identified in our study.

From Table 6, it is evident that the practitioners do collaborate with each other while making any decision regarding RT. The companies are using a mix of automated and manual RT, at Sony the practitioners specified that almost 50% of their testing activity is automated, whereas participants from Axis claimed that majority of the RT is automated. For test automation the companies are using in-house build test tools, except one team who is currently using a third party tool. At Axis all three teams are using the same test automation and management tool. The scope of RT depends on change/fix and time-line of the project. For changes, testers prefer to run the smaller scope (selected test cases),

Table 6: Regression testing practices.

ID	Description of practice	SWSE	FCS	FWD	FWR	PTS	CPL ¹
Pr1.	<i>Collaboration</i> : Developers and testers collaborate while deciding the scope for the testing of changes and new features.	✓	✓	✓	✓	✓	
Pr2.	<i>Execution frequency</i> : The frequency of executing tests depends upon the type/importance of the functionality.	✓					
Pr3.	<i>Execution frequency</i> : Testing as much as possible with selected test cases near the completion of the project.					✓	LPR6
Pr4.	<i>Reuse of test cases</i> : Using existing test cases for the testing of changes.	✓			✓		
Pr5.	<i>Adding new test cases</i> : New test cases are added for new features and issue leakage.	✓	✓				
Pr6.	<i>Running full scope</i> : Regression testing is performed with full/bigger scope at code freeze, before release, or in case of changes in the core of the system.	✓	✓		✓	✓	LPr2
Pr8.	<i>Nightly testing</i> : RT is executed with full scope in nightly runs.			✓		✓	LPr2
Pr7.	<i>Day time testing</i> : Select a scope that should run (fast) in a day time and cover bigger part.	✓		✓		✓	
Pr9.	<i>Weekly round or RT</i> : Run a weekly round of RT with a smaller scope.	✓					
Pr10.	<i>Scope Selection</i> : Mostly run RT with selected scope because of time constraint.		✓				
Pr11.	<i>Running fix suite</i> : Using fix set of test cases (that cover the core/basic functionality) for RT.	✓					LPr3
Pr12.	<i>RT Automation</i> : The companies are using both manual and automated RT.	✓	✓	✓			
Pr13.	<i>Testing tools</i> : For test automation using third party tools.	✓					
Pr14.	<i>Testing tools</i> : For test automation using in-house developed tool.		✓	✓	✓	✓	LPR5
Pr15.	<i>RT suite size</i> : Regression test suites are fairly large.	✓	✓				
Pr16.	<i>Exploratory testing</i> : Use of exploratory testing to complement RT.	✓	✓		✓	✓	
Pr17.	<i>Test strategy</i> : For every new project a detailed test strategy / plan is developed.	✓	✓	✓		✓	
Pr18.	<i>Priority tags</i> : Using tags with test cases to determine the priority.		✓				
Pr20.	<i>Traceability labels</i> : Use of test case labels to link with respective modules.					✓	
Pr19.	<i>Tickets</i> : Use of tickets for highlighting the issues, it helps to identify the type and nature of the error.				✓		
Pr21.	<i>When to introduce RT</i> : Sometime early start of regression testing is preferred to catch defects at early stages.	✓				✓	LPr7
P22.	<i>RT level</i> : Applying RT at System level.	✓		✓		✓	
P23.	<i>RT level</i> : Applying RT at Component level.		✓				
Pr24.	<i>RT goal</i> : The goal of regression testing is to have confidence that product is in good shape.			✓	✓		

CPL: Corresponding practice(s) in literature.

in case of complex changes (e.g., in the core) they try to opt for a wider scope. Near the completion of project (code freeze or near the release) the practitioners

favor the execution of RT with full scope (re-test all). The practitioners at Axis also highlighted that during day time they run selected test cases and during nightly testing they prefer to run full scope. The participants from SWSE (Sony) highlighted that they run weekly rounds of RT with selected scope, whereas FCS (Sony) adopt the selection of scope because of time constraints.

The practitioners reuse of existing test cases as regression test suite, they do augment new test cases in the existing test suites in case of adding of new functionality or any issue leakage. The practitioners label the test cases with “test tags” and/or “test case labels” according to the respective importance and/or modules. The use of labels conceptually differs in both the companies, at Sony labels are used to highlight the relative priority of the test case, and at Axis labels are used to link the test cases with respective modules. Regarding test strategy or test planning, two teams SWSE (Sony) and PTS (Axis) highlighted this aspect, both teams claimed it a company wide activity. At Sony, a high level test strategy exists that serves the basis for the design of detailed test strategy. For every new project test architect design the detailed strategy, it includes the details about the modules to test, scope of testing, test cases, and etc. At (Axis), there exists an organization wide test plan. Preparing and updating the project specific test plan and test scripts is the responsibility of the QA team. Some of the practices identified in our study are already defined in the existing literature (See table 2), we have created the mapping of literature identified practices in the last column of Table 6. Considering the results of a survey-based study conducted by Dias-Neto et al. [13] in which authors surveyed the overall software testing practices we can conclude that practices identified in our study are purely regression testing related. As opposed to the survey conducted by Engström and Runeson [20] where authors specified that the practices identified in their survey are general testing practices.

5.2.1. Test Case Selection and Prioritization (RQ2.1)

An important aspect of regression testing practices is test case selection and prioritization, which we investigated in RQ2.1. A summary of selection and prioritization criteria along with the information sources used for decision making can be found in Tables 7 and 8, respectively, we did map the information sources with the selection/prioritization criteria in Table 7 under the column heading “UIS”. The last column “CCrL” in Table 7 lists the selection/prioritization criteria available in the literature (Literature findings are listed in Table 2).

The primary criteria for test selection and prioritization for regression test suite is the ‘change’. The practitioners assess the size, complexity, and location (area) of change. The participants from all the teams highlighted this criterion. Feedback from developers, and experience and knowledge of the practitioner about the system are the information sources used to assess the change. Experience and knowledge of system and feedback from developers are the primary sources of information for majority selection/prioritization criteria. For instance, scope assessment, fix regression test suite, critical functionality, and coverage are the criteria where the practitioners are using their experience to access these criteria.

Table 7: Test selection and prioritization criteria.

ID ¹	Criteria	SWSE	FCS	FWD	FWR	PTS	UIS ²	CCrL ³
SPCr1.	Change (size, complexity, and location).	✓	✓	✓		✓	IS1, IS2, IS7	LPc1, LPc5
PCr2.	Risk.	✓			✓	✓	IS2, IS4	
PCr3.	Critical functionality.	✓	✓				IS1, IS2, IS3, IS7	LPc4
PCr4.	Defect priorities.				✓		IS6	
SCr5.	Situation based scope assessment.		✓				IS2, IS7	LSc5
SCr6.	Coverage(feature/module).			✓		✓	IS2, IS5	
SCr7.	Affected areas.			✓		✓	IS1, IS2	LSc6
SCr8.	Deadlines.	✓	✓		✓		IS8	
SCr9.	Least recently used test cases.	✓					IS2, IS5	
SCr9.	Most frequently failed test cases.			✓			IS2, IS5	

¹ SPCr: Selection/Prioritization Criteria, PCr: Prioritization Criteria, SCr: Selection Criteria

² UIS: Utilized Information Sources.

³ CCrL: Corresponding Criteria in Literature

Table 8: Information sources utilized for test selection and prioritization.

ID	Information sources	SWSE	FCS	FWD	FWR	PTS
IS1.	Developers' feedback.	✓	✓	✓	✓	
IS2.	Experience and Knowledge of system.	✓	✓	✓		✓
IS3.	Issue ranking.	✓				
IS4.	Team meetings.	✓		✓		✓
IS5.	Test history.	✓				
IS6.	Test tags.		✓			
IS7.	Feature to test traceability.			✓		✓
IS8.	Project managers.	✓	✓		✓	

Other relevant criteria that are used for the test selection or prioritization are risk, deadlines, and critical functionality. Three out of five teams are using these criteria. Measurement of risk is based on change (area and complexity of change), the practitioners are sometimes using issue ranking to prioritize the risk. Knowledge of the practitioner about risk areas is another source of information for risk measurement. Deadlines are central concerning the scope selection. If the project is on schedule and deadlines are relaxed then testers prefer to run full scope (complete test suite) for regression testing. If deadlines are tight (often, it is the case), then the practitioners opt for an adaptive (selected) scope. In this case, they do prioritize the test cases on the basis of the critical (essential) functionality. The selected scope consists of the test cases that cover the critical functionality and the primary use cases (basic function-

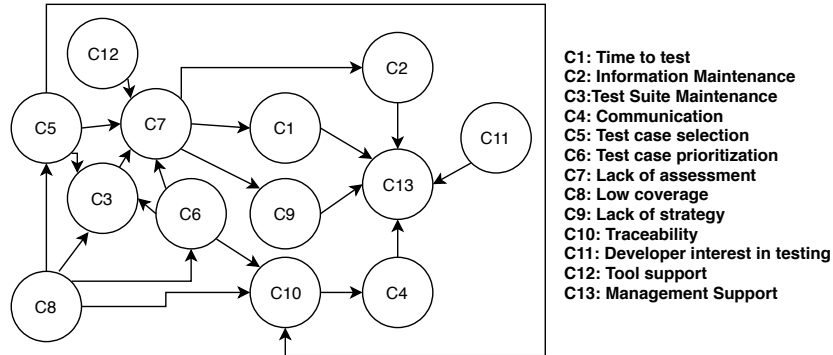


Figure 2: Relationship between RT challenges.

ality). The companies are using a predefined test suite that covers the primary use cases (basic functionality). They update this suite (add new test case) in case developers add new functionality (features) or if they think that something (test case) is missing.

Some selection criteria are team specific. For instance, SWSE team select the test cases that have not been executed recently (least recently used test cases), they do choose the test cases concerning the date of use. The testers at FWD team are using most recently failed test cases and most frequently failed test cases during the selection process.

5.2.2. Key Challenges in Regression Testing (RQ2.2)

Table 9 summarizes the regression testing challenges reported by the practitioners.

The identified challenges can be classified into two categories, 1) management related challenges, and 2) technical challenges. Management related challenges include “C1: Time to test”, “C2: Information management”, “C4: Communication”, “C9: Lack of strategy”, “C11: Developers interest in testing”, and “C13: Management support”. Whereas technical challenges are “C3: Obsolete test cases (Test suite maintenance)”, “C5: Test case selection”, “C6: Test case prioritization”, “C7: Evaluating and improving (Lack of assessment)”, “C8: Low coverage”, “C10: Traceability”, and “C12: Tool support”. Figure 2 presents the relationship between the identified challenges, from the figure we can see that the central challenge is C13 (i.e., Management support) and it could cause all other related problems. Time to test (C1), information management (C2), and lack of strategy (C9) are the root cause of technical challenges. Among the technical challenges C7 (i.e., Lack of assessment) is the basis of other technical challenges including C3, C5, C6, C8, & C12). Similarly, C10 (i.e., traceability) can effect the C5 (selection) and C6 (prioritization). Finally C3, C5, C6, & C10 cause the low coverage (C8).

In the last column of Table 9 we mapped the challenges identified from the

Table 9: Regression testing challenges.

ID	Challenge	SWSEFCS	FWD	FWR	PTS	CCL ¹
C1	<i>Time to test:</i> Too much testing in a short time. Lack of time is the primary hindrance for the assessments and innovation of RT.	✓	✓	✓	✓	LC11
C2	<i>Information management:</i> Because of poor information maintenance and lack of documentation it is hard to extract the required information.	✓	✓	✓	✓	LC8
C3	<i>Obsolete test cases:</i> There are tests which are not failing for a long time, a big challenge as it effect the efficiency and effectiveness.	✓	✓	✓	✓	LC9
C4	<i>Communication:</i> Lack of information about the changes and upcoming plans from the other teams.	✓	✓	✓	✓	
C5	<i>Test case selection:</i> Instead of the dynamic test scope focus is on fixed test suites. Adding value to the scope by selecting relevant test cases is a challenge.	✓	✓	✓	✓	
C6	<i>Test case prioritization:</i> Inappropriate assumptions about the priorities and what to select.		✓	✓	✓	LC3
C7	<i>Evaluating and improving:</i> Lessons learned on a project are often broad in nature, not possible to reflect on the success. Time to test is one the factors that limits the option of evaluations and improvements.	✓	✓	✓	✓	LC10
C8	<i>Low coverage:</i> Detection of new bugs is a challenge this is because of, running the same test cases for a long time that is causing the low test coverage.	✓	✓	✓		
C9	<i>Lack of strategy:</i> Focus is on project success instead of organizational success, no long-term strategy.		✓	✓		
C10	<i>Traceability:</i> Finding trace-links between tests and other artifacts is a challenge.	✓		✓		
C11	<i>Developers interest in testing:</i> Developers' least interest in quality, delivering their code without testing.	✓				
C12	<i>Tool support:</i> The testers have to go through excessive manual work in RT because of the unavailability of good verification tools.	✓				LC13
C13	<i>Management support:</i> Lack of understanding about the importance of verification.		✓			LC15

¹ CCL: Corresponding challenges in literature

literature (See Table 2) with the challenges identified in our study. Seven challenges (C1, C2, C3, C6, C7, C12, & C13) identified in our study are similar to the challenges identified in the studies [4, 15, 20]. Interesting aspect is that first study by Engström and Runeson [20] was conducted in 2010 and the other study by Harrold and orso [4] was carried out in 2008. Despite the voluminous research on regression testing [5], it is evident that the challenges are not fully addressed after 10 years. It is an indicator that either published research is not fulfilling the industrial needs or the industrial stakeholders are unable

to exploit the intended benefits from the available research. Along with the identification of challenges with the help of practitioners, it is also important to work on improvements in a close collaboration with the practitioners.

5.3. Suggested Improvements for Regression Testing (RQ3)

In RQ3, we investigated improvements that the practitioners seek in their testing environment. Table 10 summarize the possible improvements we identified with the help of the practitioners.

A majority of identified improvements correspond to the challenges presented in Table 9. Last column (CC) of Table 10 list the challenges that could be addressed by the respective improvements. There is often a one-to-one mapping between the challenges and the suggested improvements. For instance, C2 (information management) is identified as a challenge in both the companies, the practitioners even who did not recognize it as challenge agree to build and maintain a standard information repository (I-1). Similarly, C3 (test suite maintenance) is highlighted as a challenge. To address the issue, the practitioners suggest to work on the elimination of irrelevant and obsolete test cases (I-2: Test suite minimization) and updating test suites by adding new relevant test cases (I-3: Test suite augmentation). Working on test suite minimization will also be helpful for reduction of testing time, that ultimately will be helpful to cope with the challenge of time to test (C1). Another important suggestion that can be helpful for the reduction in time to test is to introduction of parallelized testing (I-12). Overall test optimization (I-14) will also improve the situation.

Availability of good verification tools (C12) is an issue of concern for the companies, because of this fact testers have to do a lot of manual work in RT. In this regard, it is suggested to identify and adopt the verification tools (I-4) appropriate for the companies environment. To cope with the challenge of communication (C4), the practitioners think that there is a need to work for the improved collaboration (I-5). Although, the practitioners (developers and testers) do have a certain level of communication, but there is no formal mechanism for the communication. Specifically, with reference to information sharing regarding changes and change plan. In the presence of well maintained information repository this issue should be minimized. Test case selection and test case prioritization are of central importance in regression testing, especially for the large-scale development. Because right selection with appropriate priority order can improve the coverage and defect detection rate. The companies lack in both areas (C5: Test case selection and C6: Test case prioritization). From the identified improvement (I-7, I-8) provides the basis for the improvement in selection and prioritization methods. Exploratory testing (I-6) could be an immediate solution choice for the practitioners.

5.4. Goals and Criteria for Successful Regression Testing (RQ4)

In response to our question “How do you evaluate the success in regression testing?” the majority of the participants responded that we don’t use any metrics to measure the success, and we do the subjective measurement. Subsequently, we did change our question, “If you have to evaluate the success,

Table 10: Identified Improvements.

ID	Improvements	SWSE	FCS	FWD	FWR	PTS	CC ¹
I-1.	<i>Information repository:</i> Need to build and maintain a standard information repository, to make information extraction easier and faster.	✓	✓	✓	✓		C2
I-2.	<i>Removing irrelevant test cases:</i> Need to identify and remove irrelevant/obsolete test cases, to reduce the test cost and improve the efficiency and effectiveness.	✓		✓	✓	✓	C1, C3
I-3.	<i>Test suite augmentation:</i> Need to update test suites by adding new relevant test cases, to improve the coverage and effectiveness.	✓					C3
I-4.	<i>Good verification tools:</i> Need to identify and add good verification tools, to reduce the dependence on manual testing.	✓	✓				C12
I-5.	<i>Improved collaboration:</i> There is a need to regulate the collaboration mechanism between developers and tester, to improve the information sharing (What developers are changing? and what testers have to test?)	✓	✓	✓		✓	C4
I-6.	<i>More exploratory testing:</i> To find new bugs, need to do more exploratory testing.	✓	✓				C8
I-7.	<i>Evaluate:</i> To measure success and improve RT, need to collect test execution statistics and evaluate.		✓	✓		✓	C7
I-8.	<i>Improved test strategy:</i> Need to improve the testing strategy and innovate testing methods.		✓				C9
I-9.	<i>Test History:</i> Need to maintain test execution history dynamically, so that it could be reused to innovate testing.		✓			✓	C5, C6
I-10.	<i>Early start of regression testing:</i> To ensure the stability, need to introduce early start of regression testing.				✓		C1
I-11.	<i>Test case classification:</i> To classify test cases according to severity, need to introduce a standard template for error reporting, that should allow linking tests to errors.				✓		C6, C10
I-12.	<i>Parallelized testing:</i> To cope with the time constraints, need to introduce the concept of parallelized testing.					✓	C1
I-13.	<i>Selective RT:</i> Need to improve the selection mechanism of test cases, selection should be without any compromise.					✓	C5
I-14.	<i>Test optimization:</i> To shorten the release cycle, need to work on overall optimization of RT.				✓		C1, C3, C5, C6

¹ CC: Corresponding challenge(s).

which would be your success goals?” Tables 11 and 12 summarize goals and criteria regarding the success of regression testing. The success criteria refer to the conditions that are essential to achieving the regression testing goals.

We identified six regression testing goals along with the seven success criteria. From Table 11 it is evident that the goals “Critical defect detection”,

Table 11: Regression testing Success goals.

ID	Goal	SWSE	FCS	FWD	FWR	PTS	CSC ¹	CGL ²
SG1.	<i>Customer satisfaction</i> : The released product is working and the customer is not complaining.	✓	✓				SC6	
SG2.	<i>Critical defect detection</i> : RT is regarded as successful if it can find the critical bugs (no critical bugs should be delivered to the customer).	✓	✓	✓	✓	✓	SC1, SC2, SC3, SC7	LG4
SG3.	<i>Confidence</i> : The tester should be confident about the achieved quality.	✓	✓		✓	✓	All SCs	LG3
SG4.	<i>Effectiveness</i> : In term of fault detection, the goal is to find as many bugs as it is possible.	✓	✓	✓	✓	✓	SC1, SC2, SC3, SC5, SC7	LG1
SG5.	<i>Controlled fault slip-through</i> : How many issues have slipped to the customer is important, it provide a measure to success. The goal is to keep fault-slip through as low as possible.	✓	✓	✓			SC1, SC2	LG6
SG6.	<i>Efficiency</i> :Running the planned scope for RT in a limited time.	✓	✓	✓	✓	✓	SC1, SC2, SC3	LG7

¹ CSC: Corresponding Success Criteria.

² CGL: Corresponding goals in the literature

Table 12: Regression testing Success criteria.

ID	Criteria	SWSE	FCS	FWD	FWR	PTS
SC1.	<i>Right selection</i> : Based on requirements selecting and defining the right test cases. Selection of appropriate test cases is the key for successful RT.	✓	✓	✓		
SC2.	<i>Knowledge of changes</i> : For successful regression testing, QA should be well aware of changes in the system.	✓		✓		
SC3.	<i>Early start of RT</i> :The success criteria in RT is to start of RT as early as possible.			✓		
SC4.	<i>Coverage</i> : Coverage is one the criteria to evaluate the success.	✓	✓	✓		
SC5.	<i>Tester's Experience</i> : Knowledge and experience of tester is the subjective measure of confidence in RT.	✓	✓			
SC6.	<i>Customer's feedback</i> : The customer feedback is the measure of confidence in the software testing.	✓	✓			
SC7.	<i>Quality of test cases</i> : Carefully designed test cases can guarantee the finding issues and good coverage.	✓	✓			✓

“Confidence”, “Effectiveness”, and “Efficiency” are highlighted by the majority of the participating teams. Regarding success criteria, mainly there were defined by three teams, one team defined only one success criterion, and one

team did not discuss any success criteria. In Table 11, we mapped the success criteria with the respective goals, we did map all criteria with the goal “confidence”. Infect confidence is a subjective term, and we cannot measure the confidence, a tester could be confident about his testing results based on the experience and knowledge. However, to have confidence in regression testing, testers have to achieve other success goals. We identified RT goals from the previous studies [16, 25], these goals are listed in Table 2. The goals identified from litterateur are mapped to the goals identified in this study (see last column in Table 11)

Despite the fact mentioned earlier regarding the initial response of practitioners on the assessment of success, the results are encouraging. The practitioners are aware of the importance of success, and they know how to measure the success. Presently, they are not making any objective assessments because of environment support, availability of resources, and an appropriate mechanism for the assessment.

6. Summary and Conclusions

We have conducted an interview-based study in two large communication companies and investigated various aspects of regression testing in practice. In the following, we summarize our findings regarding each research question.

The definition of RT that emerged from the practitioners (**RQ1**) (see Section 5.1) is in line with the standard definition presented in ISO, IEEE, system and software engineering vocabulary [31]. The goal of regression testing is to get confidence that the system changes have not introduced unwanted system behavior rather than to find errors. The scope of regression testing depends on the timing in the projects (e.g. small fix or a major release) and risk analysis for incoming changes. This study confirms our previous viewpoints [16] and Engström et al. [20].

Looking at the regression testing practices (**RQ2**), our respondents are using manual and automated regression testing, there is a lot of manual work that practitioners have to do. For test automation mainly the companies are relying on in-house developed tools. Inability to find a third party tool is an interesting findings that points towards specific testing requirements or processes that can not be captured by a general regression testing tool support. Another possible explanation is that our respondents prefer to change the tools rather than wait for tool vendors to provide the necessary adaptations. This allows for faster and better alignment between the testing process, testing strategy and the tool support. Test strategy definition seems to be an ad hoc practice among our respondent which confirms the need for in-house and flexible tooling.

The testers and developers collaborate in the decision making regarding various aspects of RT. Only SWSE and FWR respondents confirmed to reuse test cases. This may explain who exploratory testing is used as a replacement for regression testing when they fail to find issues with fixed regression suites. Greater test reuse seems to be hindered by a lack of traceability labels (mentioned by one group) or tickets (also mentioned by one group), see Table 6 for details.

The scope and time of change drive regression testing activities. Our practitioners mostly run a static regression testing scope that cover the basic/core functionality of the respective systems. Size, position and complexity of change drive test case selection, supported by domain knowledge and experience and dialog with developers. When pressed by short deadlines, our practitioners limit the scope of regression testing. The selection/prioritization criteria identified in our study is closely related to the selection/prioritization criteria available in the related studies [4, 5, 20].

Looking at the 12 identified challenges (RQ2.2), information management, test suite maintenance, communication, test case selection, and test case prioritization are common to both the companies. We identified fourteen challenges from the related studies [4, 20]. There are six challenges identified in our study are also available in the related work (See Table 9). Interestingly, communication is frequently mentioned by traceability is mentioned only by one group. This may mean that our respondents underestimate the role of traceability mechanisms in enriching and supporting communication between the roles involved in regression testing. Similarly, traceability could be supporting test suite maintenance mentioned by almost all respondents involved. We believe that more research should be directed towards understanding the relations between the challenges and how the solutions can mitigate them.

We identified two improvement categories for regression testing practices (**RQ3**): 1) Improvements related to the management affairs, 2) improvements related to technical interventions, these improvements are related to test case selection, test case prioritization, test suite minimization, test suite augmentation, and assessment of regression testing. Yoo and Harman [5] presented a literature survey of 159 papers on test case selection, minimization and prioritization techniques, it includes a large number of techniques proposed in these three areas. Despite the lot of work in these areas, the practitioners still think that there is a need for improvement. Which indicates either the techniques proposed in the literature are not fulfilling the requirements of industry or the practitioners are not aware of these techniques. Surprisingly, our respondent pointed good verification tools as a necessary improvement despite developing in-house and heavily tailored solutions themselves. Another interesting aspect is the identification of irrelevant or obsolete test cases that appears to be a similar challenge that the identification of obsolete software requirements surveyed by Wnuk et al. where over 80% of the respondents confirmed negative influence of obsolescence on their processes [32].

The success goals identified in our study are, customer satisfaction, critical defect detection, confidence, effectiveness, controlled fault slip-through, and efficiency (**RQ4**). Our study also reveals some preconditions which can guarantee the success of RT. For instance, right selection of test cases, and knowing the changes are the conditions that are essential for the success of RT. Regarding goals findings of this study complement the findings of previous studies [16, 25]. Similarity in the success goals identified in two different studies indicate that there is an awareness and urge in the industry regarding evaluating the success of regression testing.

Acknowledgments

The authors would like to thank the practitioners from Sony Mobile Communications AB and Axis Communications AB, who participated in all sessions of the study. We are also thankful to all facilitators who helped us in organizing the study. The funding agency Vinnova, as well as Sony Mobile Communications, Axis and Softhouse support the work through the EASE Industrial Excellence Center (reference number 2015-03235).

Appendix A. Interview Guide

Introduction

Personal introduction. Interviewers tell the participant about themselves, their background and training, and interest in the area of software testing.

Study goal. The lead interviewer explains the study goal to the participant.

Goal: The goal of the study is to know the state of regression testing practice in the large-scale embedded software development. The purpose is to know that how companies are managing their test systems, specially with reference to regression testing. The following points are the focus of the interview.

- Current Practices ,
- Test Selection and prioritization,
- Challenges and issues,
- Improvements, and
- Evaluation of success goals.

Benefit: This study will provide the basis for improving the different aspects of regression testing considering the different views of people within the organization. We believe your opinion is valuable. This investigation gives you (interviewee) a chance to contribute to the improvement of the regression testing environment.

Interview process. Interviewer describes the overall process, that how the interview will take place.

◦ *Interview duration:* The interview will be completed in about an hour time.

◦ *Interview questions:* there may be some questions that the interviewee perceives as stupid, silly, or difficult to answer. It is possible that an appropriate question for one person may not be suitable for the other.

◦ *Counter questions:* The interviewee may feel free to ask counter questions for the clarification of an interview question and can disagree with any statement of the interviewer.

◦ *Answers:* We believe that in an interview, we can not rate any answer as right or wrong. The interviewee need not worry about in this regard, and we

expect he/she will answer the questions on the basis of knowledge and experience.

Respondent Background

The purpose of this section is to know about the participant's professional background, current role and responsibilities.

Question 1: Could you please briefly describe on your professional background?

- Your qualification,
- Overall Experience,
- Time in the current company.

Question 2: How you will define your expertise?

- Software Engineering,
- Software Development,
- Software testing.

Question 3: Please specify about your current job.

- Your current team,
- Your role in the team.

Question 4: Can you please brief us about your current project(s).

Interview Part to explore the RT state of practice

This is the central part of this interview, and we are interested to know about the current practice, selection and prioritization criteria, challenges, and improvements regarding regression testing. In the final part of the interview, we will discuss the regression testing success goals. We will start by asking our questions regarding current practices. Please feel free to add detail at any point of the interview that you think we missed to ask or you forget to describe.

Defining regression testing. The purpose is not to get the academic definition of regression testing. The interviewers are interested to know the perception of the practitioner.

Question 1: What is regression testing for you?

Questions regarding current practices. The aim is to know how practitioner's team is performing regression testing.

Question 1: Can you please give us a walk-through of overall process and highlight where and how often regression testing is performed?

Question 2: Have you been involved in decision making regarding (When to do regression testing? Which test cases should be executed? How to select a subset of candidate test cases?)?

Question 3: In your team regression testing is manual or automated?

Question 4: For automation which tools are in use of your company / team?

Question 5: Decision are taken by the individuals or by the QA team? (Who are the people involved in decision making?)

Selection and prioritization. Although, selection and prioritization are regarded as the part of practice. Considering the importance of selection and prioritization, we are asking focused questions.

Question 1: Do you use some predefined criteria for the selection and / or prioritization of test cases?

Question 2: Which information you use while making decisions for selection and/or prioritization?

Question 3: Do you or your team maintain the required information? Is this information readily available?

Question 4: When have you made the right selection of test cases?

Question 5: How do you evaluate / know whether the selection was right?

Challenges and Improvements. Challenges are the obstacles that can hinder the smooth and successful execution of the operations. Like other working environments, practitioners working in software development organizations are facing different issues. Our interest is to know those issues which are recurring and require attention.

Question 1: What are the challenges for testers regarding regression testing?

Question 2: Do you have any mitigation strategies to overcome these challenges?

Question 3: What are the challenges, you think need to pay more attention?

Question 4: Considering the underlying challenges, can you identify the areas of improvement?

Success Criteria. To determine the success of any activity, we measure it with the predefined goals, that is, if the goals have met or not.

Question 1: What is your criteria of success of regression testing? Do you measure the success?

Question 2: At your company / team do you define success goals?

Question 3: For a successful regression testing what are the goals that should be achieved?

Question 4: How will you determine that the desired goals have been achieved?

Closing Questions. We mentioned earlier that the goal of this research is to identify potential problems and come up with suggested solutions. Your opinion counts!

Question 1: In your opinion which is the most important area that should have to be the focus of this research?

Question 2: Do you want to share some more information which you think is important to consider, that we may have missed?

References

- [1] J. Rooksby, M. Rouncefield, I. Sommerville, Testing in the wild: The social and organisational dimensions of real world practice, Computer Supported Cooperative Work (CSCW) 18 (5-6) (2009) 559.

- [2] N. B. Ali, K. Petersen, M. V. Mäntylä, Testing highly complex system of systems: an industrial case study, in: Empirical Software Engineering and Measurement (ESEM), 2012 ACM-IEEE International Symposium on, IEEE, 2012, pp. 211–220.
- [3] E. Engström, P. Runeson, G. Wikstrand, An empirical evaluation of regression testing based on fix-cache recommendations, in: Proceedings of the Third International Conference on Software Testing, Verification and Validation (ICST), 2010, IEEE, 2010, pp. 75–78.
- [4] M. J. Harrold, A. Orso, Retesting software during development and maintenance, in: Proceedings of Frontiers of Software Maintenance FoSM, IEEE, 2008, pp. 99–108.
- [5] S. Yoo, M. Harman, Regression testing minimization, selection and prioritization: a survey, *Software Testing, Verification and Reliability* 22 (2) (2012) 67–120.
- [6] R. Kazmi, D. N. A. Jawawi, R. Mohamad, I. Ghani, Effective regression test case selection: A systematic literature review, *ACM Comput. Surv.* 50 (2) (2017) 29:1–29:32.
- [7] M. Khatibsyarbini, M. A. Isa, D. N. Jawawi, R. Tumeng, Test case prioritization approaches in regression testing: A systematic literature review, *Information and Software Technology*.
- [8] S. U. R. Khan, S. P. Lee, N. Javaid, W. Abdul, A systematic review on test suite reduction: Approaches, experiment’s quality evaluation, and guidelines, *IEEE Access* 6 (2018) 11816–11841.
- [9] G. M. Kapfhammer, Empirically evaluating regression testing techniques: Challenges, solutions, and a potential way forward, in: Proceedings of the Fourth International Conference on Software Testing, Verification and Validation Workshops (ICSTW), 2011, pp. 99–102.
- [10] E. Engström, K. Petersen, N. B. Ali, E. Bjarnason, SERP-test: a taxonomy for supporting industry-academia communication, *Software Quality Journal* 25 (4) (2017) 1269–1305.
- [11] A. Bertolino, Software testing research: Achievements, challenges, dreams, in: Proceedings of the Workshop on the Future of Software Engineering (FOSE07, 2007, pp. 85–103.
- [12] A. Orso, G. Rothermel, Software testing: a research travelogue (2000–2014), in: Proceedings of the Workshop on Future of Software Engineering (FOSE14), 2014, pp. 117–132.
- [13] A. C. Dias-Neto, S. Matalonga, M. Solari, G. Robiolo, G. H. Travassos, Toward the characterization of software testing practices in south america: looking at brazil and uruguay, *Software Quality Journal* 25 (4) (2017) 1145–1183.

- [14] M. Kassab, J. F. DeFranco, P. A. Laplante, Software testing: The state of the practice, *IEEE Software* 34 (5) (2017) 46–52.
- [15] D. Brahneborg, W. Afzal, A. Čaušević, A pragmatic perspective on regression testing challenges, in: *Software Quality, Reliability and Security Companion (QRS-C)*, 2017 IEEE International Conference on, IEEE, 2017, pp. 618–619.
- [16] N. M. Minhas, K. Petersen, N. B. Ali, K. Wnuk, Regression testing goals-view of practitioners and researchers, in: *24th Asia-Pacific Software Engineering Conference Workshops (APSECW)*, 2017, IEEE, 2017, pp. 25–31.
- [17] D. Parsons, T. Susnjak, M. Lange, Influences on regression testing strategies in agile software development environments, *Software Quality Journal* 22 (4) (2014) 717–739.
- [18] D. M. Rafi, K. R. K. Moses, K. Petersen, M. V. Mäntylä, Benefits and limitations of automated software testing: Systematic literature review and practitioner survey, in: *Proceedings of the 7th International Workshop on Automation of Software Test*, IEEE Press, 2012, pp. 36–42.
- [19] E. Juergens, B. Hummel, F. Deissenboeck, M. Feilkas, C. Schlogel, A. Wubbeke, Regression test selection of manual system tests in practice, in: *Proceedings of the 15th European Conference on Software Maintenance and Reengineering (CSMR)*, 2011, pp. 309–312.
- [20] E. Engström, P. Runeson, A qualitative survey of regression testing practices, in: *Proceedings of the International Conference on Product Focused Software Process Improvement*, Springer, 2010, pp. 3–16.
- [21] J. Kasurinen, O. Taipale, K. Smolander, Software test automation in practice: empirical observations, *Advances in Software Engineering* 2010.
- [22] X. Lin, Regression testing in research and practice, *Computer Science and Engineering Department University of Nebraska, Lincoln* (2007) 1–402.
- [23] S. Ng, T. Murnane, K. Reed, D. Grant, T. Chen, A preliminary survey on software testing practices in australia, in: *Software Engineering Conference, 2004. Proceedings. 2004 Australian*, IEEE, 2004, pp. 116–125.
- [24] H. Do, S. Elbaum, G. Rothermel, Supporting controlled experimentation with testing techniques: An infrastructure and its potential impact, *Empirical Software Engineering* 10 (4) (2005) 405–435.
- [25] S. Jafarin, D. Nandi, S. Mahmood, Test case prioritization based on fault dependency, *International Journal of Modern Education and Computer Science* 8 (4) (2016) 33.
- [26] P. Runeson, M. Höst, Guidelines for conducting and reporting case study research in software engineering, *Empirical software engineering* 14 (2) (2009) 131.

- [27] B. Kitchenham, S. L. Pfleeger, Principles of survey research: part 5: populations and samples, ACM SIGSOFT Software Engineering Notes 27 (5) (2002) 17–20.
- [28] D. S. Cruzes, T. Dyba, Recommended steps for thematic synthesis in software engineering, in: Empirical Software Engineering and Measurement (ESEM), 2011 International Symposium on, IEEE, 2011, pp. 275–284.
- [29] K. Petersen, C. Wohlin, A comparison of issues and advantages in agile and incremental development between state of the art and an industrial case, Journal of systems and software 82 (9) (2009) 1479–1490.
- [30] I. S. C. Committee, et al., Ieee standard glossary of software engineering terminology (ieee std 610.12-1990). los alamos, CA: IEEE Computer Society.
- [31] ISO/IEC, ISO/IEC/IEEE24765:2010: Systems and software engineering–vocabulary (2010).
- [32] K. Wnuk, T. Gorschek, S. Zahda, Obsolete software requirements, Information and Software Technology 55 (6) (2013) 921 – 940. doi:<https://doi.org/10.1016/j.infsof.2012.12.001>.
URL <http://www.sciencedirect.com/science/article/pii/S0950584912002364>