

On the Role of Communication, Documentation and Experience during Testing - An Exploratory Study

Timea Illes-Seifert, Barbara Paech
University of Heidelberg,
Institute of Computer Science, Im Neuenheimer Feld 326
Germany-69120 Heidelberg
{illes-seifert, paech}@informatik.uni-heidelberg.de

Abstract: Nowadays, the quality of software is becoming more and more a competitive factor. As complete testing is impossible, testers have to make decisions, e.g. to choose which parts of the software have to be tested in which way. For this purpose, testers need a lot of information, such as input documentation which serves as a basis for the derivation of test cases or information on the project status which serves as a basis for planning the testing process. Thus, testers rely on up-to-date and complete information in order to make sound decisions. Consequently, the quality of the testing process depends on the quality of the information sources available for the testers. This paper presents the results of an exploratory study conducted during the SIKOSA research project with expert testers of our industry partners in order to identify the most valuable sources of information during testing. Particularly, we conducted interviews in order to investigate which documents are often used by testers, as well as the role of communication and experience. Our results show that defect reports are very valuable. User manuals and problem reports are equally important, because they represent real usage of the software and serve testers as an input for realistic test cases. In addition, our results show the influence of an independent testing team on test process characteristics.

1. Introduction

Spectacular software failures like the crash of the Ariane 5 rocket [Do97], but also software failures which occur daily in business software show that testing activities are essential in order to detect defects before release. However, complete testing is impossible, and as a consequence, testers have to make a lot of decisions during the testing process in order to constrain the set of potentially infinite number of test cases to a set which can possibly detect the most critical defects. Thus, testers make decisions, e.g. on the test design technique to be used in order to derive test cases or on the test data which serve as input for test cases. In order to conduct all these decisions thoroughly, testers need complete and up-to-date information, e.g. about requirements, project status, etc. The main assumption of our work is that *the better the information and the information flow between testers and other project members (e.g. requirements engineers or project manager) is, the better will be the quality of the decisions made during testing*. The knowledge of testers' information needs allows to provide testers with the right information at the right time and to define the best way of providing it (e.g. documented, verbal). Based on this knowledge, test process improvements can be designed and implemented.

In this paper, we present the results of an exploratory study, performed during the SIKOSA research project with expert testers, the aim of which is to analyze information flow within the testing processes. Particularly, we analyze which documents are frequently used and which roles are consulted when making decisions during testing. In addition, we investigate the role of experience needed to make sound decisions. The results of the study serve as a basis for recommendations regarding the optimization of a testing process.

The remainder of this paper is organized as follows. Section 2 introduces a conceptual decision framework, containing decisions to be made during the testing process. We used this framework as the basis for our study during data collection and analysis. In Section 3, related work is presented. Section 4 describes the design of the study. We conduct expert interviews for exploring information flow patterns in testing. Section 5 presents the findings whereas Section 6 deals with the threats to validity of our study. Finally, Section 7 concludes the paper.

2. Decisions within the Testing Process

This section introduces some basic concepts and the decision framework which served as a basis for the exploratory study. In previous research work, we analyzed the testing process [IHPR05], [BIL07a], [BIL07b] and developed a decision framework, which identifies the decisions to be made during the testing process and assigns them to decision levels.

Test planning and control (TP&C). Since testing is a complex process, thorough planning and monitoring is needed. Consequently, during TP&C activities testers decide on schedules, resources, and efforts estimated for testing activities, as well as on risks (which threaten the successful completion of the testing).

Test strategy definition (TSD). The main goal of TSD is to define which parts of the software should be tested in which way (e.g. how intensively or with which test design techniques) in order to find the most critical defects fast. Correspondingly, testers decide on test end criteria, defining conditions which have to be fulfilled to finish testing activities. In addition, decisions on the test design techniques used to develop test cases (to find the most critical defects) have to be made and a test model can be selected. A test model, e.g. a state model facilitates the derivation of test cases. Closely related to the selected model, the decision on the representation for the model, as well as coverage criteria, e.g. transition coverage or state coverage in case of a state model can also be decided during TSD. In the case of automation, testers also have to decide on the degree of automation.

Test analysis and design (TA&D). During test analysis and design activities, testers decide on test cases including test steps, test data and test sequences. In addition, testers review the documentation to be used as input for testing activities and decide on their quality, e.g. testers make decisions about the testability of the requirements specification document. If the requirements specification does not fulfil the required quality from the testers' point of view some rework is needed.

Test execution (TE). During test execution, decisions on the evaluation of the executed test cases (called test runs) have to be made. Consequently, testers have to decide, whether a test run revealed a defect or not.

Test cycle evaluation (TCE). During test cycle evaluation, the results of the test runs have to be analyzed. Thus, testers check if the test end criteria have been fulfilled and decide whether testing activities can be finished or not.

3. Related Work

Similar work, analysing information gathering strategies of maintainers is described in [S202] and in [TL01]. Most related work focuses on the description of the test process, e.g. the fundamental test process presented in [SLS06] addresses phases and activities to be passed through when testing a software system. The IEEE standard for software test documentation [IEEE98] specifies all artefacts to be created during the testing process, e.g. the test plan; the information flow, as well as the information sources needed are not part of the standard. Another group of related work represents test process improvement models like TPI (Test Process Improvement) [KP02] or test maturity assessment models, e.g. TMM (Testing Maturity Model) [BSC96]. The focus of these models is not the information flow within the testing process, but the steps for its improvement, respectively on criteria to assess the maturity of the organizational testing process. None of the presented references contains empirical studies. The work which is most related to our work is [Da05]. The authors present guidelines for requirements engineering practices which facilitate testing. In contrast to the work in [Da05] which addresses requirements engineering processes and artefacts, this study has a larger focus including other information sources of the software development project. In addition, we analyze communication, as well as the role of experience during testing. To our knowledge, this is the first study exploring the information flow during the testing process in detail. Empirical studies are essential in understanding the nature of information processes. This is also the case with the testing process. By this study, previously formulated advices in literature which are not supported by empirical studies could be confirmed. For example, the outstanding role of the requirements specification and of previously found defects for the testing process could be confirmed. This study, however, also allows insights which have not been yet considered in literature, e.g. the role of the user for testers. Knowing that information from customer is so valuable for testers, processes can be adapted appropriately in order to facilitate the information flow from customers to testers.

4. Study Design

In this chapter, we present the research questions; we introduce the characteristics of the participants of the exploratory study and provide an overview of the data collection and data analysis methods used to gather and to investigate the data.

4.1. Research Questions

Subsequently, we will outline the research questions addressed in this study and the rationales behind. The respective questions and rationales are listed in Table 1.

Table 1: Research Questions

Questions	Rationale
Q1: Which documents are frequently used by testers when making which testing decisions?	The main assumption of this research question is that documents are an important information source for all participants of the software engineering process, including testers. To know which documents are frequently used by testers is important, because quality assurance activities concerning information sources often consulted by testers can be intensified purposefully.
Q2: What role does communication play as an information source?	The main assumption of this question is that documentation is never completely sufficient as input to the testing process, so that details have to be clarified in face-to-face discussions. And even if documentation was complete, up-to-date and unambiguous, communication is often preferred to reading documents.
Q3: What is the role of experience in testing?	This is an important question to be analysed, because it is important to know to what extent and for which decisions testers rely on their experience instead of documentation. Knowing this enables to decide: Which activities are suited for test automation? Which decisions are suited to be executed by novice testers?

4.2. Participants

The main criterion for the selection of the participants was their experience in the testing area. As a consequence, all participants out of five organizations had at least three years of experience, and most of the participants had five to ten years of experience. Three participants had even more than ten years of experience. Table 2 summarizes the characteristics of the participants. Organisation A and E develop standard software, whereas the other organisations develop individual software. Only organisation C develops software for in-house use. The testers in organisation A work on the same project, whereas the testers in the organisations C and D work on different projects.

Table 2: Participants' Characteristics

Experience (in years)	Role(s)	Main Tasks	Organisation
>10	Test designer	Test planning Test case design Manual test execution	D
>10	Test designer	Test planning Manual test execution	D
>10	Test manager	Establishment of a standard testing process including supporting tools	B
>10	Tester, Test manager	Test planning Manual test execution	E
10	Test manager, Quality engineer	Test planning Test case design Monitoring system operation	D
10	Test manager, test designer	Test management and control Test case prioritization Human resources management and motivation	A
5	Test manager	Product development, Manual test execution and protocol Coordination of testing activities Product roll-out (= deployment in the productive environment)	C
5	Test designer	Supports test manager in planning activities Test case design Manual test execution and protocol	A
5	Test designer	Test case design Execution of test cases Fault localisation Regression testing	D
3	Test automation engineer	Manual test execution and protocol Test automation: implementation of the test automation framework	A
3	Test manager	Test planning Manual test execution	C

4.3. Study Process

The study was performed as a qualitative study. Qualitative studies use data in form of text, images and sound drawn from observations, interviews and documentary evidence, and analyse it using methods that do not rely on precise measurement to yield their conclusions [Se99]. We used this research method because it helps to gain more experienced with the analysed phenomenon. In our case, our goal was to get a deep understanding of the information flow during the testing processes.

The study was conducted in the form of seven face-to-face interviews and one telephone interview. Three interviewees completed the questionnaire “offline”. The interviews were semi-structured, based on a questionnaire sent in advance to the participants. The interviews took three hours on average.

The questionnaire itself consists of three parts. The first part contains questions regarding the testers’ experience and role in the organization, as well as questions on the organizational testing process. Particularly, the interviewees were asked about the testing decisions to be made during the testing process in their particular organisation. The second part of the questionnaire addresses communication and documentation sources during testing, whereas the third part contains questions regarding the role of experience within particular activities. In the second part of the questionnaire, the interviewees got a list of documents that could theoretically be used during testing decisions, e.g. requirements specification or design specification. Then we asked the interviewees which documents are needed when making particular decisions. The interviewees were also asked to indicate documents not contained in the list, as well as a “wish list” containing documents currently not available to them. Similarly, we asked the interviewees which specific roles are consulted when making particular decisions. In the third part of the questionnaire, we asked the interviewees to rate the experience needed to make particular decisions. The questionnaire can be found in [IP08].

Data Collection. In the data collection phase, field notes taken during the interviews were coded and stored in a study data base. Coding [Se99] is a procedure which transforms qualitative data into quantitative data by assigning values to qualitative statements. This allows the combination of qualitative and quantitative methods for data analysis. During the coding, interviewees were contacted when ambiguities in the data occurred.

To assure the validity of our results, we used multiple information sources for evidence as recommended in [Yi03]. Thus, beside interviews, document analyses have been performed. We analyzed test case specification templates and test case specifications, test protocols and test process descriptions, as well as input documentation, e.g. requirements in the organization, the testers belong to. Furthermore, we got insight into other information sources like discussion forums. Another aspect considered to assure validity was the representativeness of the interviewees with regard to their qualification, experience and testing tasks. All interviewees are experienced testers, four of them with more than ten years of testing experience.

Data Analysis. For the data analysis, we used different qualitative and quantitative analysis methods. Quantitative methods were used in order to determine patterns and tendencies in the data, e.g. by counting which role is consulted most of all during the testing process. Qualitative methods were used to search for an explanation for these particular tendencies. Thus, we performed cross-case analysis [Se99] and partitioned the data into different categories by using different criteria, e.g. we partitioned the data depending on the testing group’s organization as an independent team or not.

5. Findings

In this section, the analysis of the results of the study is presented. First, we detail the test process characteristics, including the roles and decisions mentioned by the interviewees. Then, we discuss the documentation, communication and experience characteristics. Finally, we present the problems during the testing process as mentioned by the interviewees.

5.1. Test Process Characteristics

Test planning and control. With the exception of risk analysis, all decisions to be made during TP&C (as described in section 2) are equally often mentioned to be performed. 9 out of 11 interviewees mention that a particular decision is made during the testing process in their organization. Only about half of the interviewees (6 cases of evidence) cited that a risk analysis is performed when deciding on relevant risks influencing the testing project. Only 4 of the interviewees report that all TP&C related decisions are in the testing team's field of responsibility. Three interviewees even indicate that all TP&C related decisions are performed by persons not belonging to the testing team, mostly by the project manager. In all other cases, TP&C related decisions are partially made by the testing team.

Test strategy definition is a task not well established within the testing processes we analyzed. Only few decisions are indicated to be made, where the definition of test end criteria is a decision mentioned most by the interviewees (9) followed by the selection of the test design technique (5). All other activities are rarely cited.

Test analysis and test design. Decisions on test steps, on test data and on test sequences are indicated to be made by nearly all interviewees, whereas the assessment of the testability, as well as the assessment of the quality of the input documentation is indicated only by about half of the interviewees. These decisions are mostly made by the testing team. Within organizations not having an independent testing team, these decisions are performed by developers (where the "tester" is not the developer of that particular part of the software). Since most organizations do not automate tests, the realization of test cases and consequently all decisions related to test automation are confirmed only by a small part of the interviewees.

Test execution and test cycle evaluation. All interviewees report to make decisions concerning the success or failure of particular test runs. The decision on the test run evaluation is mostly made by testers, in some cases by the whole testing team. The evaluation of a test cycle is only performed by fewer than half of the interviewees.

To sum up, it is not surprising that decisions indicated to be made by almost all interviewees concern test decisions in the narrow sense (test case definition and test execution). However, TP&C, as well as TA&D related decisions are each indicated to be performed on average by 9 out of 11 interviewees. Decisions concerning the TCE, as well as the TSD are made on average by fewer than half of the interviewees. Figure 1 shows the test process characteristics as mentioned by the participants.

5.2. Documentation Characteristics

TP&C related decisions, particularly decisions on effort and schedule, require the most documentation, followed by TA&D decisions, especially decisions on test data and test steps, as well as on the definition of test sequences. The interviewees report a high need of documentation during TCE, especially the requirements and design specification. Decisions during TSD and TE require little documentation.

The role of the requirements specification. The requirements specification is by far the most important document for testers (46% of all decisions need the requirements specification as input, see also Figure 2.). During TP&C, the requirements specification is especially used for decisions concerning effort estimation and scheduling, whereas during TA&D the requirements specification is especially used to decide on test cases (including test steps, test data and test sequences). In addition, the requirements specification is also used during TE in two contexts. First, when testers are pressed for time, they report to use the requirements specification as test specification. In this case, decisions on the test design are made concurrently to the test execution. Second, in case of a failure or of an unexpected behaviour, testers consult the requirements specification in order to analyze if it is a real failure. All testers emphasize the importance of the requirements specification to be up-to-date and complete.

Learning from defects. Previously found defects are a very valuable information source for testers, whereas both defects found by the test team, as well as defects reported by customers are almost equally important (25%, respectively 24% of all decisions require customer problem report respectively bug reports as input, see also Figure 2). Testers report that previously found defects are good indicators for defects in the software because of following reasons:

- (1) Many defects persist across different versions. Two categories of persisting defects are reported by testers: *permanent defects*, which occur across all versions and *“jumping” defects*, which regularly “jump” over a constant number of versions.
- (2) The correction of a defect introduces more defects.

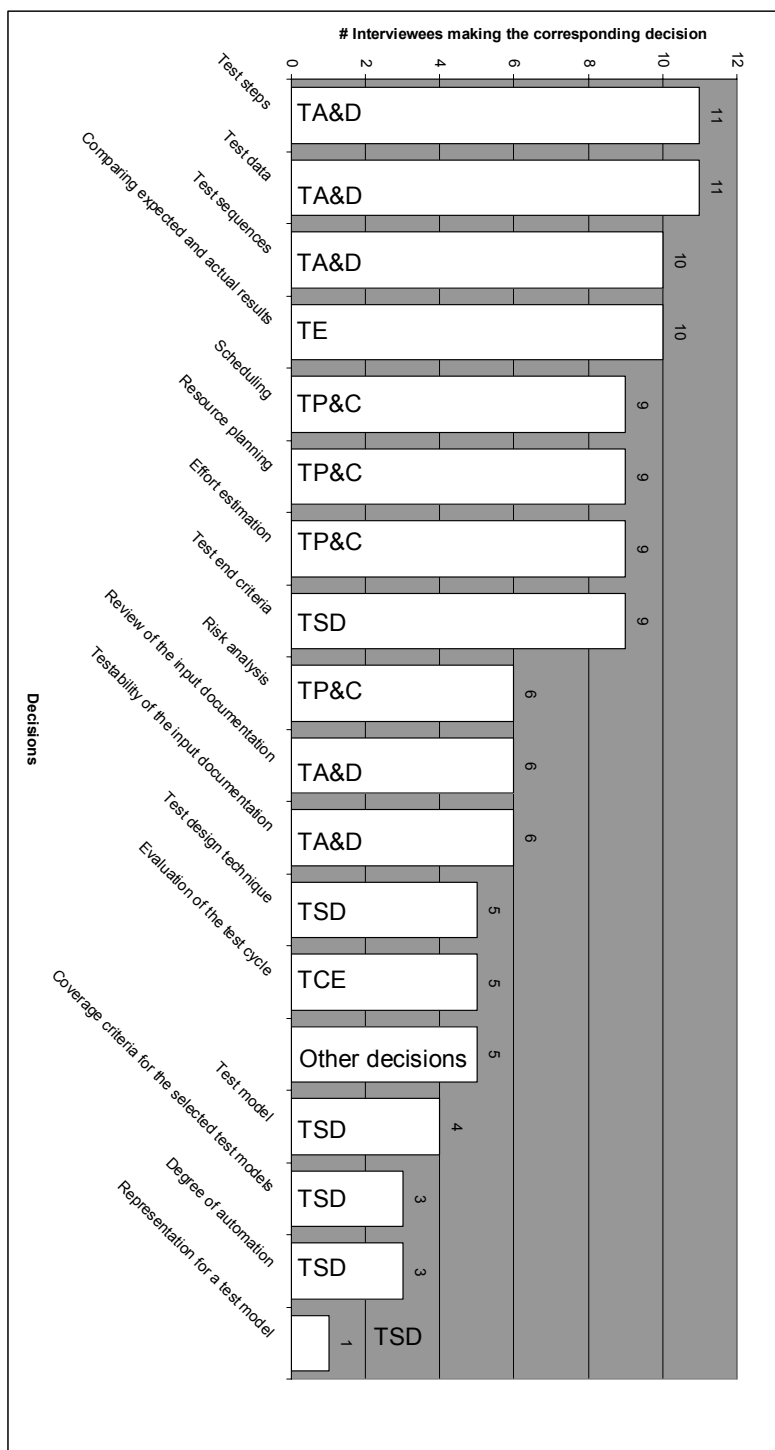


Figure 1: Test Process Characteristics

Knowing potential defects, testers can decide on the test effort to be spent to test particular areas of the software. Defects also serve as input for TA&D. On the one hand, testers select test cases to be re-executed if they revealed a defect. On the other hand, testers develop new test cases on the basis of known defects using different strategies, which we refer to as *intensifying*, *expansion* and *transferring*.

- (1) *Intensifying*: Testers investigate the functionality more intensively and usually vary the test data, for example, or the preconditions of the test case.
- (2) *Expansion*: Testers search for functionality used by the functionality which revealed the defect or using this functionality.
- (3) *Transferring*: Testers search for similar functionality (which could contain the same defect).

Figure 2 illustrates the documents needed as input during testing as mentioned by the interviewees.

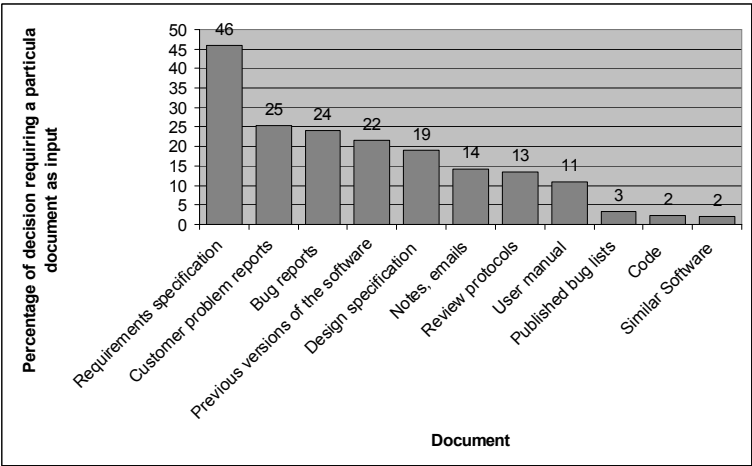


Figure 2: Documentation needs during testing

The role of the user within the testing process. Even though only few of the testers are in direct contact with users, the users play an important role during testing. Using documentation produced for and by users, testers can develop more realistic and more relevant test cases. Thus, testers bridge the gap to the customer by using customer problem reports and user manuals in order to develop realistic test scenarios and in order to define test environments and configurations close to the real productive environments. Consequently, this documentation is very valuable when deciding on test data and test steps. One interviewee also mentioned to use the user manual to get familiar with the software system.

Wish lists. When asked for information sources, which are not available but which were useful for testing, 4 interviewees emphasize that up-to-date and complete requirements are crucial and more important than other documented information sources. Interviewees cite following reasons why requirements specifications are usually not up-to-date and complete: time pressure at creation time, as well as the fact that requirements engineers are not aware of the tester’s information needs.

5.3. Communication Characteristics

During the testing process, most communication occurs among the requirements engineer and the project manager followed by the developer. Testers have direct contact with the customer only when the customer is “in-house”. Apart from this, there is no direct communication between testers and customers, in spite of their request for this type of communication. Figure 3 shows the percentage of decisions in which a role is involved.

Most communication is reported to take place during TA&D, where the main communication partners mentioned by the interviewees are requirements engineers and project managers. However, during TE, there is also a great need for communication. The main contact persons are requirements engineers and developers mostly in the presence of a failure. Communication during TP&C occurs mostly with the project manager. However, little communication takes place during TSD and TCE.

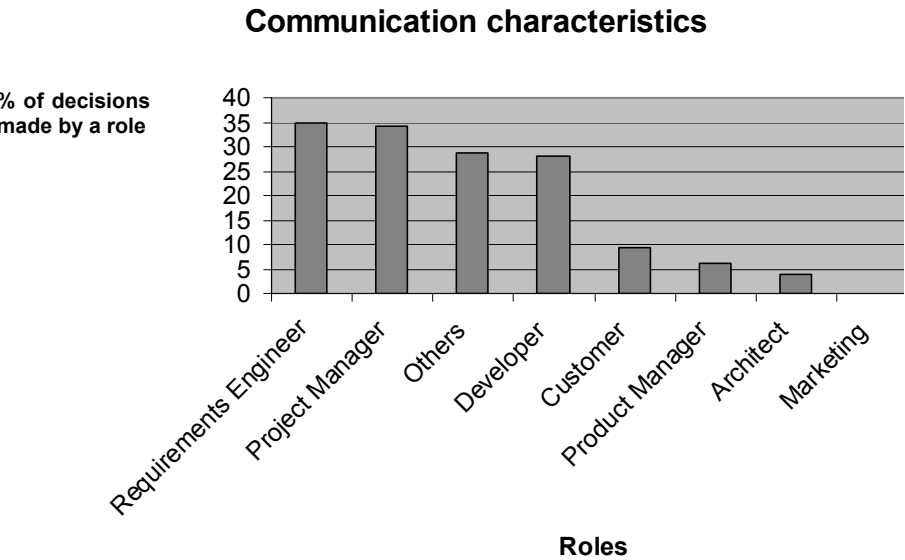


Figure 3: Communication Characteristics

5.4. Experience Characteristics

The interviewees report that most experience is required during TP&C and during TCE. In addition, a lot of experience is required for TA&D. In contrast, little experience is required for decisions related to TSD and to TE.

Among the decisions made during the testing process, the definition of test data is stated to be the decision which requires the system specific experience at most. All interviewees indicate that this decision requires very much experience. In addition, this is the only decision which solely requires system specific experience. Effort estimation and risk analysis, as well as the evaluation of the test cycle are also indicated by the interviewees to require high system specific experience. In general, almost all decisions require more system specific than general experience. Managerial activities, e.g. scheduling, resource planning and effort estimation require the most general experience. As expected, test case execution and evaluation require the least system specific and general experience.

5.5 Problems

In the following, the main problems as mentioned by the interviewees are presented.

Poor quality of the documents used as input, especially poor quality of the requirements specification is one of the major issues during testing. We asked the interviewees to indicate the most severe problems occurring during testing. One of the most frequently mentioned problem concerns the quality of the input documents, particularly the lack of quality of the requirements specification. Only two participants do not indicate poor requirements (e.g. incomplete, unambiguous) as one of the most difficult problems during testing. Three participants especially require more detailed descriptions, particularly concerning pre- and post conditions of a requirement, as well as dependencies between requirements and between the software and its environment (including the software and hardware environment). One of the main reasons for the poor quality of the requirements from the testers' point of view is the lack of involvement in the review process. Only half of the interviewees report that testers are involved in the review process. In one special case, the requirements specification is not reviewed at all.

Testing decisions require system specific experience. Almost all decisions require more system specific than general experience. In addition, testers indicate to rely on their own experience, rather than on experience made by others, as they do not frequently consult published defect lists.

Testers rely on their own experience more than on test design techniques when deciding on test data and test steps. Testers rely more on their own experience than on test design techniques which generate a high amount of test cases and prefer an exploratory-oriented approach. In addition, in the case of time pressure, testers deviate from systematic approaches and reduce the set of test cases according to their own experience. For example, the testers apply equivalence partitioning results in 6 equivalence classes and rate 4 of them as unrealistic and with low potential to detect defects. In this case, they decide to specify and execute only these two test cases which they appraise to be well suited to reveal a defect.

High documentation and communication needs during test execution suggest incomplete descriptions of the expected outcome in test case specifications. Reasons for this are either quality deficiencies in the documentation which served as input for decisions on test cases or shortage of time when testers decided on test cases, leading to incomplete descriptions of the expected outcome.

The results of a test cycle can not be objectively assessed. Surprisingly, testers point out the role of experience in the evaluation a test cycle. One would expect that the evaluation of the test results requires “only” a decision on the efficiency of the test strategy, i.e. “Have the test design techniques been applied and have the test end criteria been met?” But since the test strategy definition is not well established in testing processes, the decisions related to TSD have to be taken later, namely during the test evaluation. In addition, one participant criticizes the lack of a systematic learning process across test cycles.

6. Threats to Validity

One threat to validity of our study is the fact that the results may be specific to the particular interviewees. We addressed this problem by selecting very experienced testers for the interviews. Another threat is the ability to generalize the results due to the fact that we selected a small population. We addressed this problem by using techniques which assure validity of qualitative studies [Se99], [Yi03]: 1) Diversification: Diversity with respect to the focus of the activities performed by the interviewees was a key criterion when selecting the participants of the study. 2) Methodological triangulation: We used different methods to analyse the data (quantitative and qualitative techniques, as described in Section 3.3). 3) Explanatory triangulation: by trying out several explanations for all results in Section 4. For example, the result, that the requirements specification document is a key information source for testers can be confirmed by several facts. First, asked for main problems in the testing process, almost all interviewees indicate the poor quality of the requirements specification. In addition, asked for required input for different decisions, the interviewees indicate the requirements specification as an important input for almost all decisions. Finally, asked for a “wish list”, the testers indicate that the quality of the requirements specification is more important than other sources of information. Based on these three facts, we conclude that the requirements specification is an important information source for

testers. Nevertheless, organisations with a higher degree of test automation or which use more formal models (e.g. embedded area) may show different results.

7. Conclusions and Future Work

In this research work, we presented the results of an exploratory study performed during the SIKOSA project with experienced testers with the aim to analyze the information flow during testing as the starting point for test process improvements. This work served as basis for the definition of the PAT3-Approach [IP06] as part of the whole SIKOSA methodology. The PAT3 Approach captures testing **experience and knowledge** in form of patterns. PAT3 defines five pattern categories (process patterns, automation patterns, transformation patterns, testability patterns, traceability patterns) which improve the interface between requirements engineering and testing.

The main results of our study regarding the research questions formulated in Section 3.1 can be summarized as follows:

The requirements specification is, not surprisingly, the document used most frequently during testing (Question 1). This document is used as input for all decisions to be made. However, there is another information source which is almost equally valuable: previously found defects. In addition, the requirements engineer and the project manager are roles consulted most frequently by testers (Question 2). Surprisingly, testers mention a high communication overhead during test execution. This fact is an indicator for poor quality of the requirements specification, confirmed as a major problem during testing by almost all interviewees. Experience plays an important role, and the definition of test data requires by far the most experience (Question 3). Moreover, decisions related to TP&C and TCE require much experience. At first glance, the latter is unexpected, but since most organizations do not define a test strategy, evaluation is not easy in the absence of operational goals. As expected, test execution requires little experience and is consequently well suited to be automated.

This exploratory study gives first indications for hypotheses we aim to verify in subsequent empirical studies. Based on our results, we can formulate the following hypotheses. **H1:** Previously found defects are good indicators and predictors for future defects. **H2:** Embedding the user into the testing process (particularly into the prioritization and reviewing the test cases) increases the efficiency of the testing process. **H3:** Testing decisions and activities require more system specific than general experience. **H4:** Combined approaches integrating experience into traditional test design techniques lead to better test cases.

Acknowledgements

We would like to thank all the interviewees for their cooperation and help by providing information and insight into documents.

8. References

- [BIP07a] Borner, L.; Illes-Seifert, T.; Paech, B.: Entscheidungen im Testprozess, Software Engineering Konferenz (SE 2007), Hamburg, March 27 - 30, 2007.
- [BIP07b] Borner, L.; Illes-Seifert, T.; Paech, B.: The Testing Process - A Decision Based Approach, In: in: Proceedings of the The Second International Conference on Software Engineering Advances (ICSEA 2007), IEEE Computer Society, Cap Esterel, France, 25.-31. August, 2007. Cap Esterel, French Riviera, France, August 25 - 31, 2007
- [BSC96] Burnstein, I.; Suwannasart, T.; C. R Carlson: Developing a Testing Maturity Model for Software Test Process Evaluation and Improvement, Proceedings of the IEEE International Test Conference on Test and Design Validity, 1996.
- [Da05] Dahlstedt, A.: Guidelines Regarding Requirements Engineering Practices in order to Facilitate System Testing, the Proceeding of the 11th International Workshop on Requirements Engineering: Foundation for Software Quality, Porto, Portugal, 13-14 June 2005.
- [Do97] Dowson, M.: The Ariane 5 software failure. SIGSOFT Softw. Eng. Notes, ACM Press, Mar. 1997, 22, 2, pp. 84.
- [IEEE98] IEEE Std. 829-1998, IEEE standard for software test documentation, Software Engineering Technical Committee of the IEEE Computer Society, USA, 1998.
- [IHPR05] Illes, T.; Herrmann, A.; Paech, B; Rückert, J.: Criteria for Software Testing Tool Evaluation. A Task Oriented View. Proceedings of the 3rd World Congress for Software Quality, 2005
- [IP08] Illes, T.; Paech, B.: On the Role of Communication, Documentation and Experience during Testing - Results of an Interview Study, SWEHD-TR-2008-02, <http://www-swe.informatik.uni-heidelberg.de/research/publications/reports.htm>, 2008.
- [IP06] Illes, T.; Paech, B.: From "V" to "U" or: How Can We Bridge the V-Gap Between Requirements and Test?, Software & Systems Quality Conferences 2006, on May 10th 2006 in Düsseldorf.
- [ISTQB05] International Software Testing Qualifications Board, ISTQB Standard Glossary of Terms used in Software Testing V1.1, September 2005.
- [KP02] Koomen, T.; Pol, M.: Test Process Improvement: A step-by-step guide to structured testing. Addison-Wesley, 1999.
- [KP03] Kitchenham, B; Pfleeger, S. L.: Principles of survey research part 6: data analysis, SIGSOFT Softw. Eng. Notes 28, 2, Mar. 2003, pp 24-27.
- [MP02] Mosley, D. J.; Posey, D. J.: Just Enough Software Test Automation, Prentice Hall, July 2002.
- [Se99] Seaman, C.B.: Qualitative Methods in Empirical Studies of Software Engineering, IEEE Transactions on Software Engineering, 25(4), July/August 1999, pp. 557-572.
- [Se02] Seaman, C.B.: The Information Gathering Strategies of Software Maintainers, In Proceedings of the international Conference on Software Maintenance ICSM. IEEE Computer Society, Washington, DC.
- [SLS06] Spillner, A.; Linz, T.; Schaefer, H.: Software Testing Foundations - A Study Guide for the Certified Tester Exam - Foundation Level - ISTQB compliant, Dpunkt Verlag, 2006.
- [TL01] Tjortjjs, C.; Layzell, P.: Expert Maintainers' Strategies and Needs when Understanding Software: A Case Study Approach. In Proceedings of the Eighth Asia-Pacific on Software Engineering Conference (APSEC). IEEE Computer Society, Washington, DC, 2001.
- [Yi03] R.K. Yin, "Case Study Research, Design and Methods", SAGE Publications, USA, 2003.