UNIVERSITY OF GOTHENBURG

Evaluation of Test Process Improvement approaches

An industrial case study

Master of Science Thesis in the Programme Software Engineering

SNEHAL ALONE

KERSTIN GLOCKSIEN

University of Gothenburg

Chalmers University of Technology

Department of Computer Science and Engineering

Göteborg, Sweden, 2013

Evaluation of Test Process Improvement approaches
An industrial case study

SNEHAL ALONE
KERSTIN GLOCKSIEN

Examiner: CHRISTIAN BERGER

University of Gothenburg
Chalmers University of Technology
Department of Computer Science and Engineering
SE-412 96 Göteborg
Sweden
Telephone + 46 (0)31-772 1000

Department of Computer Science and Engineering
Göteborg, Sweden 2013

ACKNOWLEDGEMENTS

This thesis was conducted during spring 2013 within the Software Engineering Master's Programme at Chalmers University of Technology and University of Gothenburg, Sweden. The thesis was carried out as a case study and a project at Volvo IT in Gothenburg.
First, we would like to thank Johan Uneman at Volvo IT for initiating this project and giving us the opportunity to conduct the thesis at the company. We would further like to give a special thanks to Anna Thorén and Fredrik Ahlborg who have served as our supervisors at Volvo IT.
Further, we want to thank our supervisor at Gothenburg University, Richard Torkar, whose support, inspiration, and knowledge have been very valuable throughout this thesis work.
As a part of the study, we have conducted numerous interviews at Volvo IT. We would therefore like to thank all Volvo IT employees who we have met during this project and who have shown a great willingness to help and assist in answering all of our questions.

Abstract

Context: Test Process Improvement (TPI) approaches are frameworks or models that guide software development organizations to investigate, assess and improve their software testing process.

Objectives: We extend existing work in the area of Test Process Improvement by identifying available approaches and by evaluating them in regards to their characteristics. Furthermore, two selected approaches are evaluated with respect to their content and assessment results.

Methods: In the first part of this study we use a systematic literature review to identify the existing TPI approaches which are then used in the second part of the study. The second part of the study is an industrial case study in which two TPI approaches are applied in an industrial setting.

Results: We contribute in providing (1) a complete, in our opinion, list of 16 existing TPI approaches and their characteristics, (2) a detailed comparison of the content and the results of the two applied approaches (TPI Next and TMMi) and (3) experience in applying them in industry. As a result of this research we found that the content as well as the assessment results of the two approaches are similar to a great extent.

Conclusions: Numerous Test Process Improvement approaches are available, but not all are generally applicable for industry. One major difference between available approaches is their model representation. Even though, the applied approaches generally show strong similarities, differences in the assessment results are noticeable due to their different model representations.

Contents

List of Tables

List of Figures

1 Introduction

This section gives an introduction to the thesis, starting with a background to the topic. Further, the purpose and aim of the study, the research questions and the delimitations of the study are described. The section finishes by giving an outline of the remaining sections in the report.

1.1 Background

Software testing is an ingrained part in the software development process. It is an important activity to support software quality. Major parts of the resources in a software development project are consumed for testing purpose. Studies show that testing constitutes more than 50% of the overall costs of software development [Harrold, 2000]; and with the increasing complexity of software the proportion of testing costs will still rise unless more effective ways of testing are found. One main focus of investigation in industry, for reducing cycle time and development costs, and at the same time increasing software quality are their testing processes [Collofello et al., 1996]. However, state of practice in testing is sometimes ignored or unknown in software development organizations, and testing is done in an *ad hoc* way [Bertolino, 2007] without designated testing roles being defined.

In the past, several Test Process Improvement (TPI) approaches have been developed to help organizations in assessing and improving their testing processes. Nevertheless, to successfully be able to improve testing processes of a specific organization an appropriate approach has to be found which suits their specific needs and the methodologies used in that company. Obviously, the expectations of the companies differ depending on, e.g., internal goals, maturity awareness and process knowledge.

In conclusion, there is a need in providing an overview of available TPI approaches and their specific characteristics in order to assist organizations in selecting the approach most suitable for them.

1.2 Purpose and aim of the study

The main objective of this study is to evaluate existing TPI approaches. This evaluation is split into two parts. First, a general evaluation is applied to all approaches found by a systematic literature review. Second, a more specific and detailed evaluation is performed on a subset of the approaches in a case study.

The first part starts by finding a sufficient set of TPI approaches available in literature. Then these approaches are evaluated by a set of criteria. Besides providing information about the identified TPI approaches useful for further research, this evaluation constitutes the basis for the selection of appropriate approaches to by applied in the setting of the organization under study in the second part of the project, i.e., the case study.

The second part starts with a pre-selection of applicable approaches based on the results of the first evaluation (inclusion and exclusion criteria). A presentation of the pre-selected approaches to the organization results in two approaches which are decided to be applied in parallel in the organization. The selected approaches are examined and evaluated in more detail regarding their specific content, and finally, after application of both approaches to the organization their results are compared.

1.3 Research questions

With respect to the first part of the study, the identification and general evaluation of existing TPI approaches, the following research questions are answered:

RQ1 Which different TPI approaches can be found in literature?

The intention of this question is to identify research papers dealing with approaches developed to improve software testing processes.

RQ2 What are the specific characteristics of these Test Process Improvement approaches?

The motivation for this question is to identify the unique characteristics which differentiate the approaches from each other. The answers to this question provide the information needed to answer RQ 2.1 and 2.2.

RQ2.1 Which approaches are generally applicable in industry?

Based on the characteristics identified by RQ2 the general applicability of the approaches is investigated by this question. The applicability might be limited by a limitation of the approach to a specific domain, insufficient information, or not completed development, and thus these approaches are not generally applicable in industry.

In the second part of the study, the case study, the following research questions are then answered:

RQ2.2 Which approaches are valuable for test process improvements in the company under study?

Also based on the characteristics identified by RQ2 the approaches appropriate for application in the case organization are selected by organization representatives based on their expert opinion and expectations.

RQ3 How well can the content of the selected approaches be mapped to each other?

To be able to compare the assessment results of the TPI approaches applied in the case organization the similarities and differences with respect to the content of the selected approaches need to be identified. Besides being important input for RQ4, and thus affects the case study, the answers to this question provide significant information in regards to a general evaluation of the applied TPI approaches.

RQ4 How do the results of the selected approaches differ after applying them?

The intention of this question is to investigate to which extent the approaches provide similar assessment results and in which aspects their results differ.

1.4 Delimitations

As this thesis concentrates on the identification and evaluation of TPI approaches and their assessment results, the actual application process of the approaches plays a subordinate role. Therefore, the application of the approaches is limited to the assessment of the test process. Furthermore, the specific assessment processes of the applied approaches have been adapted to the needs of this study as described in later sections.

1.5 Outline of the study

This report is divided into nine sections.

Section 2 - Related work - provides an overview of literature reviews and case studies conducted in areas related or similar to Test Process Improvement.

Section 3 - Study design and method selection - describes the overall design of the study based on the Technology Transfer Model and gives reasons for the selection of the two mainly applied research methods, systematic literature review and case study.

Section 4 - Execution of the systematic literature review - describes in detail the steps executed in the systematic literature review.

Section 5 - Execution of the case study - describes the case study design, the case and the steps executed in the case study. In addition it provides some information about the two TPI approaches applied in the case study.

Section 6 - Results - provides the answers to the research questions.

Section 7 - Discussion - discusses the results from the systematic literature review and the case study.

Section 8 - Threats to validity - describes the threats to validity of this study and how they have been addressed.

Section 9 - Conclusion - concludes the study.

2 Related work

This section focuses on literature reviews and case studies conducted in areas related or similar to the area of Test Process Improvement. Details on the specific TPI approaches, on the other hand, are provided by the systematic literature review and summarized as the results of RQ1 and RQ2 (see Section 6.1 and 6.2).

The specific research area of Test Process Improvement appears to be insufficiently studied. Even though a sufficient number of test process improvement approaches have been developed in the past, no systematic literature reviews identifying the available approaches, and no independently conducted case studies applying the TPI approaches are known to us. Available studies about the development of new Test Process Improvement approaches sometimes include case studies, experiments or surveys as validation of the approach, as in [Jacobs and Trienekens, 2002], [Heiskanen et al., 2012], [Jung, 2009], [Taipale and Smolander, 2006], [Kasurinen et al., 2011a] and [Karlström et al., 2005]. Comparisons and evaluations of test process improvement approaches are reported in [Swinkels, 2000] and [Farooq and Dumke, 2008]. However, the research described in these papers is not based on a systematic literature review used as input to further validation.

More literature, with respect to literature reviews and case studies, is available in the related area of Software Process Improvement (SPI). Test Process Improvement is strongly related to SPI since the development of TPI approaches arose from the inadequate consideration of testing processes in SPI approaches. Thus, TPI approaches have distinctly been influenced by existing SPI approaches, e.g., CMM/CMMi or SPICE. Available literature reviews in the area of SPI focus on the state of art in SPI [Zil-e-Huma et al., 2012], SPI applied in small and medium enterprises, both, in general [Pino et al., 2008], in a specific domain like web development [Sulayman and Mendes, 2011], and as evaluation strategies and measurements used to assess the impact of different SPI initiatives [Unterkalmsteiner et al., 2012].

Furthermore, several case studies have been conducted with respect to CMM. The focus in these case studies is especially on requirements needed to be fulfilled to reach specific maturity levels of CMM and the actual action of process improvement. The longitudinal study by Fitzgerald and O'Kane [1999] reports how a company achieved each of the CMM maturity levels up to level 4 in a time period of four years. The case studies presented in [Dangle et al., 2005] and [Ilyas and Malik, 2003] focus on the process changes needed to evolve from CMM level 2 to 3 and to adapt a company's existing processes to the processes proposed by CMM level 2. Experiences in actually performing the CMM assessment with regards to a specific process are reported in [Kiiskila, 1998].

Comparisons of multiple SPI approaches are given in [Varkoi and Makinen, 1998] and [Wang et al., 1999]. In the case study presented in [Varkoi and Makinen, 1998] CMM and SPICE assessments are applied in two related software development units. The structures of both models are analyzed and a mapping between both models is performed for a specific process area. Finally, the assessed SPICE process capabilities and CMM maturity levels are compared. A comparison of the assessment results, the robustness and the average time needed for the assessment of the SPI methodologies SPICE, CMM, BOOTSTRAP, ISO 9000, and SPRM is given in [Wang et al., 1999].

3 Study design and method selection

In this section we present the overall study design and the research methods selected for the study. First, the Technology Transfer Model, a model which realizes industry involvement in academic studies, is introduced and the steps needed to be performed in our study are presented based on this model. Second, the main research methodologies used in this study are presented, i.e., a systematic literature review and a case study. Brief introductions to the methods are given, the reasons for selecting them are discussed, and alternative methods and the reasons for disregarding them are presented.

3.1 Overall design

As stated in the introduction the objective of this study is to support industry in finding appropriate TPI approaches that fulfill the specific needs of an organization. For this purpose, a close cooperation between academia and industry within the execution of this study is preferable.

The Technology Transfer Model introduced by [Gorschek et al., 2006] provides a concept to realize this cooperation. The model consists of the following seven consecutive steps, which are performed, both, in industry and academia.

- Step 1 - Problems in industry are identified.
- Step 2 - Problems are studied in academia and a problem statement is formulated.
- Step 3 - Candidate solutions are formulated by academia in close cooperation with industry.
- Step 4 - Candidate solutions are validated in academia, e.g., in a lab experiment.
- Step 5 - Candidate solutions are statically validated in industry, e.g., through interviews and seminars.
- Step 6 - Candidate solutions are dynamically validated in industry, e.g., in pilot projects and small controlled tests.
- Step 7 - Solutions are released and fully implemented in industry.

Academia and industry equally benefit from the use of this concept. Researchers receive the opportunity to study industry relevant issues and validate their research results in a real setting. Practitioners, on the other hand, receive first-hand knowledge about new technology which helps them in optimizing their processes.

To adopt the above benefits in our study we based the overall study design on the Technology Transfer Model. However, the individual steps have been slightly adapted to fit the specific needs of our industry partner.

A graphical overview of our study design based on the Technology Transfer Model is shown in Figure 1 and the steps are described as follows:

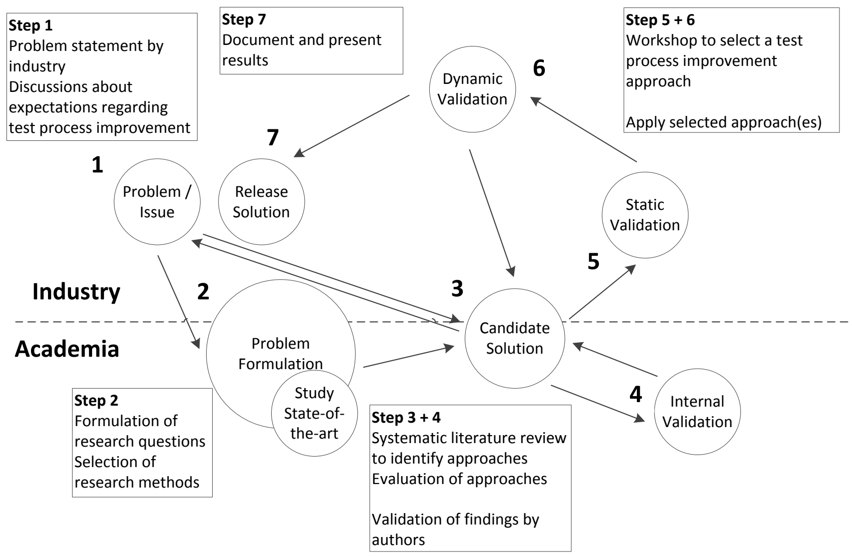

Figure 1: Technology Transfer Model (originally published in [Gorschek et al., 2006]).

Step 1 - Problem/issue A problem statement given by industry and discussions with company representatives about expectations and needs identify the problem as the unavailability of sufficient knowledge about the practiced testing process and a potential for process improvements.

Step 2 - Problem formulation A preliminary study of the problem indicates the availability of Test Process Improvement approaches providing frameworks and models to assess the current state of a testing process and to identify improvement suggestions. Based on this knowledge and the industry needs the research questions (See Section 1.3) are formulated and appropriate research methodologies to successfully address the different research questions are decided upon.

Step 3 - Candidate solution A systematic literature review is conducted to identify available TPI approaches (RQ1).

The characteristics of the approaches are identified by data extraction from the primary studies (RQ2) and an exclusion process based on the extracted data provides a selection of generally applicable TPI approaches (RQ2.1).

Step 4 - Internal validation The findings from RQs 1, 2, and 2.1 are partly validated by a number of authors of the primary studies identified by the systematic literature review. They are asked to verify the list of available test process improvement approaches for completeness and the applied exclusion process for correctness.

Step 5 - Static validation The pre-selected generally applicable test process improvement approaches are presented in industry. The $100 method, a cumulative voting method [Rinkevics and Torkar, 2013], is used to select the approaches to be applied in the organization (RQ2.2)

Step 6 - Dynamic validation The selected TPI approaches are applied in the organization. To assess the testing process, interviews are conducted and the data is analyzed based on the instructions given by the approaches. Afterwards the assessment results are compared (RQ4) based on a prior mapping of the content of the approaches (RQ3).

Step 7 - Release solution The results of the study are collected, documented and presented in academia and industry.

Based on this overall design we decided to conduct the study by using two research methods, a systematic literature review and a case study. The reasons for the selection of these two methods are given in the successor of this section. The systematic literature review covers Steps 3 and 4 of the model, candidate solutions and their characteristics are identified by the systematic review and the results are internally validated. Steps 5 and 6 of the model, the static and dynamic validation, are explicitly covered by the case study.

3.2 Systematic literature review

The first part of the study aims in identifying a comprehensive set of available TPI approaches. We decided to conduct a systematic literature review to achieve this goal.

A systematic literature review provides a mechanism for evaluating, identifying and interpreting "all available research relevant to a particular research question, topic, area or phenomenon of interest" [Kitchenham and Charters, 2007]. It summarizes the existing evidence concerning a technology.

Only thoroughly conducted systematic literature reviews are of scientific value. By following the guidelines for performing a systematic literature review in software engineering by [Kitchenham and Charters, 2007] we ensured that the research was unbiased and repeatable. Especially, the detailed definition and documentation of the search strategy and the study selection process supports the repeatability of the study. Furthermore, it provides the possibility of an assessment of the completeness of the study.

Alternatives to a systematic literature review are systematic mapping studies and tertiary reviews.

A systematic mapping identifies what evidence is available on a topic. It focuses on frequencies and trends in publications regarding the specific topic and by that identifies areas in which more systematic reviews and more primary studies should be conducted in future. Furthermore, it can also help to identify forums in which specific research topics are discussed. Since our focus was on identifying a sufficient set of available Test Process Improvement approaches regardless of the frequency of related primary studies and the time of publication for example, a systematic mapping was disregarded as a suitable research method.

A tertiary review is a systematic review of systematic reviews. It is less resource intensive than a systematic review or a systematic mapping. But, since we are not aware of any systematic reviews regarding TPI approaches, conducting a tertiary review was out of question.

Another alternative to identify the TPI approaches is a survey, a means for collecting data from a population or a representative sample of it. But due to the fact that a survey result is dependent on the response rate and can be biased by the respondent's knowledge and opinion, this method has been neglected.

3.3 Case study

The second part of the study was a case study. Indirectly, the use of the Technology Transfer Model for the study design already implied the use of a case study. To answer RQs 2.2 and 4 the direct involvement of industry was inevitable. The answers to these questions are significantly influenced by individuals and their processes. In these kinds of multidisciplinary areas case studies are often conducted. Furthermore, this part of the study had an observational character which further indicated the applicability of a case study. The assessment instructions given by the TPI approaches predetermined the use of interviews which are often used for data collection in case studies.

A case study provides a means to study a contemporary phenomenon in its natural context. Even though case studies are often criticized for being of less value, hard to generalize from and being biased by researchers, this criticism can be prevented by the use of proper research methodology practices. In order to alleviate the above criticism, we decided to conduct the case study following Runeson and Höst's [2009] guidelines for conducting and reporting case study research in software engineering.

However, there are alternatives to the case study approach. For example, action research is also conducted in a natural setting. But compared to case studies, in action research the researcher is directly involved in the process of improvement or change intended by the research. The process of research itself influences the outcome of the study. Since RQs 2.2 and 4 only have observational character and do not require actual process changes within the case organization initiated by the researcher, action research was disregarded as an appropriate research method.

A second alternative would be experiments. In experiments particular phenomena are studied in an isolated and controlled setting. They are not used to study phenomena in a natural setting, where the activities and processes are impacted by unpredictable factors. Therefore, an experiment could not have been used to address our research questions, which clearly implied the involvement of industry.

Finally, a survey could have been a possible alternative instead of using interviews as a part of the case study. But since the TPI approaches used interviews as the default means of data collection, we neglected surveys as a viable approach.

4 Execution of the systematic literature review

In this section the execution of the systematic literature review is described in detail.

4.1 Systematic literature review

We followed the guidelines for conducting a systematic literature review as proposed by [Kitchenham and Charters, 2007].

4.1.1 Review questions

Research Question 1 (Which different TPI approaches can be found in literature?), RQ2 (What are the specific characteristics of these TPI approaches?) and RQ2.1 (Which approaches are generally applicable in industry?) are explicitly addressed by the systematic review.

4.1.2 Data sources and search strategy

The search strategy was based on the following steps:

(i) identification of the first search term: 'Software Testing Process',

(ii) identification of further search terms from the titles of papers found with the first search term,

(iii) identification of further search terms from papers already known related to the research question,

(iv) identification of alternate words and synonyms for terms used in the titles of found papers and

(v) use of quotation for the complete search string to search for exact words.

This search strategy was developed after conducting a pilot search, which is described at the end of this section.

We used the following search terms:

Software Testing Process, Software Test Process, Testing Process Improvement, Test Process Improvement, Test Maturity Model, Testing Maturity Model, Testing Process Model, Test Process Model, Software Testing Standard, Software Testing Optimization, Test Improvement Model, Testing Improvement Model

The search was divided into three phases (see Figure 2). Each search phase was followed by a detailed study selection phase (see Section 4.1.3 and Figure 3).

Phase 1 In the first phase, we searched electronic databases. The search was not limited in terms of the publication year. Within the following electronic databases we searched in title, abstract and keywords:

- ACM Digital Library,

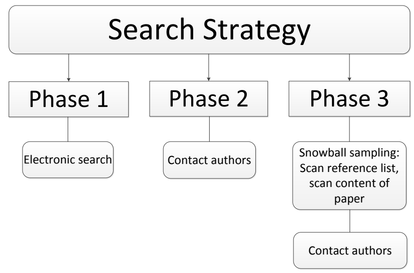

Figure 2: Phases of the search strategy.

- IEEE Xplore Digital Library,
- ScienceDirect and
- Springer Link.

In Springer Link a limitation to search only in title, abstract and keywords was not possible, therefore we searched in full-text.

Phase 2 After selecting the first data set we performed the second phase of the search to have a more representative set of candidate studies. In the second phase, we contacted the authors of 22 papers found in the electronic search of the first phase which had been selected as candidate studies to ask them for further suggest papers regarding their research topic. The contact was established using the email addresses mentioned in the candidate study or by email addresses found on the internet.

A total of 34 authors were contacted. For two authors no email address was available. Out of these 34 sent emails, 11 were undeliverable due to expired email addresses. We got a reponse from eight authors, out of which four provided relevant information. A summary of our inquiries and the reponses can be found in Appendix A.

Phase 3 In the third phase, snowball sampling [Goodman, 1961] was conducted. The two researchers performed different means of searches. Researcher A scanned the reference list of all the primary studies to identify further papers. Researcher B scanned the content of the primary studies to identify referenced papers within the text that deal with TPI approaches. The different searches complemented each other since the titles of some papers in the reference lists did not always clearly indicate that the paper is dealing with TPI approaches. Whereas for these references the relevance regarding the TPI research area was clearly indicated in the context of the text. The number of found papers by snowball sampling are shown in Table 1.

Additionally, the third phase was completed by contacting the authors of the candidate studies identified by snowball sampling that dealt with previously unknown TPI approaches.

Table 1: Numeric results of snowball sampling.

Original Reference	**Researcher A**	**Researcher B**	**Total after duplicate removal**
[Ryu et al., 2008]	3	3	3
[Taipale and Smolander, 2006]	1	1	1
[Farooq et al., 2008b]	5	5	6
[Jung, 2009]	10	10	10
[Rana and Ahmad, 2005]	0	0	0
[Saldaña Ramos et al., 2012]	9	6	9
[Burnstein et al., 1996]	2	1	2
[Xu-Xiang and Wen-Ning, 2010]	0	0	0
[Tayamanon et al., 2011]	3	3	3
[Jacobs and Trienekens, 2002]	6	6	6
[Kasoju et al., 2013]	1	0	1
[Kasurinen et al., 2011a]	8	6	8
[Heiskanen et al., 2012]	9	9	9
[Farooq and Dumke, 2008]	8	8	8
[Rasking, 2011]	3	2	3
[Reid, 2012]	0	0	0

A total of 16 papers were scanned, and the authors of three papers were contacted by email; in the end a total of five authors were contacted this way. Out of these five sent emails, four were not deliverable due to expired email addresses. One author replied but did not provide us with further research papers.

Pilot search The ultimate search strategy was decided after conducting a pilot search using the search string "Software AND Testing AND Process AND Improvement" in all databases mentioned above. The search was restricted to title, abstract and keywords in journals and modified individually to suite the respective style of each database. The detailed search queries along with our findings are shown in Appendix B.

The search results of the pilot search were not satisfactory. We could not proceed with this search and had to modify the search strategy to the one described earlier in this section because of the following reasons:

- The Springer Link database does not provide the same restriction options as the other databases, so that the search resulted in a very large first data set (more than 24,000 papers).
  - It does not have the option to restrict the search only to title, abstract and keywords. It is only possible to have either a full text search or to restrict the search to the title only.
  - The restriction options regarding publication type distinctly differ from the other databases. A restriction to journals is not given, instead the search can be restricted to articles or chapters. But journal publications can be included, both, in articles and chapters.
  - A restriction of the search to specific journals is possible, as also in the other databases. But that was no option since the search should be kept as open as possible.

- The search in ACM, ScienceDirect and IEEE Xplore resulted in a reasonable first data set but only four of these candidate studies were identified as relevant to the research topic in the first exclusion phase of reading title and abstract.

4.1.3 Study selection

For selecting the primary studies in this systematic review, the following inclusion criteria was applied, i.e., we included studies for which any of these questions were answered with ‘yes’:

- Does the paper talk about TPI approaches?
- Does the paper contain a case study in regards to TPI?
- Does the paper contain a comparison between TPI approaches?
- Does the paper talk about an assessment done in any company with respect to TPI?

Additionally, the following exclusion criteria was applied, i.e., we excluded papers that:

- only relate to Software Process Improvement in general, not TPI in particular and,
- describe general software testing models.

Phase 1 of the search strategy, the electronic database search, resulted in a total of 404 papers. After eliminating duplicates found by more than one electronic database, the number of papers resulted in 396 (see Table 2). A complete list of all references including the results of the study selection process can be found in Table 15 in Appendix C.

The exclusion was done in several steps. Every step of the exclusion process was first performed by two researches independently.

Title and abstract exclusion First, two researchers independently conducted an inclusion and exclusion process by reading title and abstract resulting in one of the three possible remarks for each paper - ‘yes’ or ‘maybe’ for further investigation in the next study selection step, and ‘no’ for exclusion due to irrelevance to the research question. In this first step, the researchers agreed to exclude 320 papers by title and abstract.

To be able to measure the reliability of the inclusion and exclusion process the inter-rater agreement was calculated using Cohen’s kappa coefficient [Cohen et al., 1960]. The coefficient indicates the degree of agreement between two judges that exceeds the expected agreement by chance. The higher the value the more reliable are the results of the judgement and it can be expected that the judgement is reasonably based on knowledge and not made by chance.

The number of observed agreement was 354 (89.39% of the observations). The number of agreements expected by chance was 301.3 (76.08% of the observations).

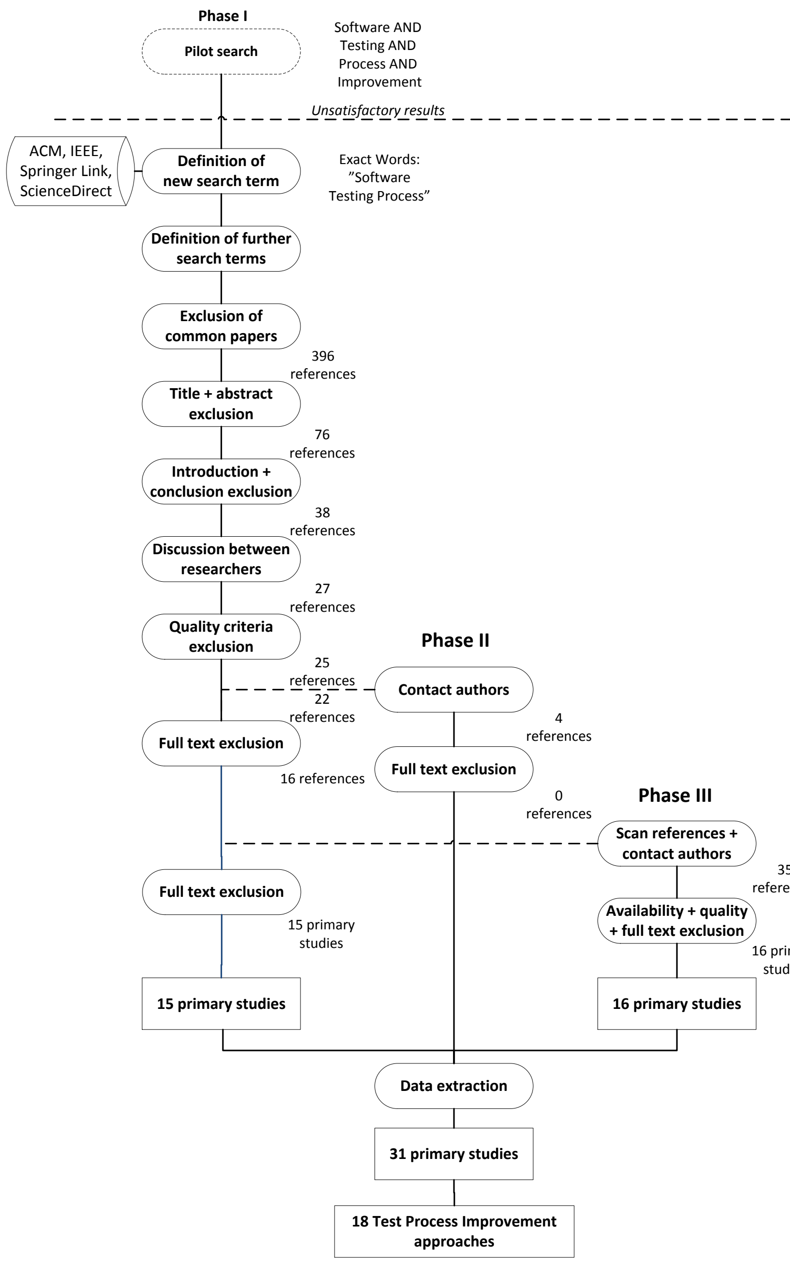

Figure 3: Study selection process.

Table 2: Numeric results of electronic search.

Search term	ACM	ScienceDirect	IEEE	Springer Link
Software Testing Process	42	10	81	131
Software Test Process	21	1	28	132
Testing Process Improvement	2	1	5	39
Test Process Improvement	13	1	9	40
Testing Maturity Model	4	0	7	17
Test Maturity Model	5	0	1	17
Software Test Optimization	1	0	0	1
Test Process Model	5	0	12	32
Testing Process Model	3	0	7	32
Test Improvement Model	2	0	0	6
Testing Improvement Model	0	0	0	6
Software Testing Standard	3	0	1	8
Total per database (before duplicate removal)	101	13	151	461
Total per database (after duplicate removal)	74	12	129	187
Total (before duplicate removal)	404			
Total (after duplicate removal)	396			

The agreement between the two researchers quantified by Cohen's kappa results in 0.557. The strength of this agreement is considered to be 'moderate'. It is significantly higher than the expected agreement by chance and therefore a reliable judgement. For calculation of the agreement the remarks 'yes' and 'maybe' have both been considered as 'yes', so that the agreement was only calculated for two categories - 'yes/maybe' and 'no'.

Introduction and conclusion exclusion Next, the remaining 76 papers were further investigated. The researchers applied the detailed inclusion and exclusion criteria to introduction and conclusion following the same process with three possible remarks for each paper like in the previous step.

The researchers agreed to exclude 38 papers and to include 16 papers. For 22 of the papers a discussion about inclusion or exclusion was needed due to the present disagreement between the two researchers. Unlike in the previous step, the remarks 'yes' and 'maybe' occurring together for one paper have been considered as disagreement and resulted in the need for discussion.

The number of observed agreements was 57 (75.00% of the observations). The number of agreements expected by chance was 35.9 (47.23% of the observations). The agreement between the two researchers quantified by Cohen's kappa results in 0.526. The strength of this agreement is 'moderate' and therefore considered as a reliable judgement. The agreement was calculated for three categories - 'yes', 'maybe' and 'no'.

Within the discussion further 11 papers were then excluded by consensus of the two researchers. This application of detailed exclusion criteria resulted in a number of 27 references.

Quality criteria exclusion Third, two papers were excluded by the application of the quality criteria described in Section 4.1.4.

Table 3: Results of study selection process.

Year	Reference	Primary Study	Exclusion criteria
2008	[Ryu et al., 2008]	yes	
2006	[Taipale and Smolander, 2006]	yes	
2009	[Meng, 2009]	no	Alleged Plagiarism
2009	[Xin-ke and Xiao-Hui, 2009]	no	Alleged Plagiarism
2008	[Farooq et al., 2008b]	yes	
2010	[Cruz et al., 2010]	no	Software testing process as part of a reference model for software industry
2009	[Jung, 2009]	yes	
1997	[Heiser, 1997]	no	General information about software testing
2005	[Rana and Ahmad, 2005]	yes	
2012	[Saldaña Ramos et al., 2012]	no	Necessary skills for software testers
1996	[Burnstein et al., 1996]	yes	
2010	[Kasurinen, 2010]	no	Practicality and applicability of software testing standards
2011	[Kasurinen et al., 2011b]	no	Ability of organizations to change the testing process
2001	[Kikuchi and Kikuno, 2001]	no	Introduction of a new tool to an organization
2000	[He et al., 2000]	no	Software testing metrics
2008	[Oh et al., 2008]	no	Implementation strategy for test process improvements
2010	[Xu-Xiang and Wen-Ning, 2010]	yes	
2011	[Tayamanon et al., 2011]	yes	
2002	[Jacobs and Trienekens, 2002]	yes	
2013	[Kasoju et al., 2013]	yes	
2011	[Kasurinen et al., 2011a]	yes	
2012	[Heiskanen et al., 2012]	yes	
2006	[Bueno et al., 2006]	no	CMMi-based evaluation of a software testing model
2008	[Farooq et al., 2008c]	no	Evaluation of testing processes in service-oriented architecture
2008	[Farooq and Dumke, 2008]	yes	
2011	[Rasking, 2011]	yes	
2012	[Reid, 2012]	yes	

Exclusion in the context of contacting authors After applying the quality criteria, Phase 2 of the search strategy - contacting authors - was started in parallel to Phase 1. During preparation for Phase 2 further three papers were excluded by consensus due to the irrelevance to the research topic.

Full text exclusion At the end of Phase 1, the full-text of the remaining 22 papers was read and a further 6 papers were excluded by consensus. The remaining 16 papers identified as relevant to the topic were further considered as basis for conducting Phase 3 - snowball sampling. Finally, we agreed to exclude one more paper based on re-reading the full-text.

The detailed exclusion process of Phase 1 of the search strategy resulted in 15 primary studies (See Table 3).

Phase 2 of the search strategy, emailing the authors, resulted in four additional papers suggested by them, but these were later excluded when applying the exclusion criteria. In Phase 3 of the search strategy, 35 references found by snowball sampling were further investigated. Out of these 35 candidate studies, 12 papers were not freely available and 5 were excluded by reading the full-text. A further three papers were excluded based on the criteria specified for quality assessment (See Section 4.1.4).

Table 4 summarizes the results of the study selection in Phase 3.

In conclusion, the 15 primary studies found by the electronic database search were complemented by 16 primary studies found by snowball sampling. This, we believe indicates the importance of conducting snowball sampling in systematic reviews.

4.1.4 Study quality assessment

As the purpose of the systematic review was to find TPI approaches mentioned in literature we applied few quality criteria.

The first quality assessment was already done within the search in the electronic databases by excluding papers that:

- Were not written in English.
- Were not available as full-text.

During the first study selection, we excluded two papers that seemed to indicate plagiarisms [Meng, 2009] and [Xin-ke and Xiao-Hui, 2009]. Major parts of the papers were completely identical, including title and abstract. Only the content of some smaller sections, parts of the structure and particularly the name of the TPI approaches differed a bit. Since both papers were published in the same year, it was not obvious which one was the original. We contacted the authors of both papers and informed them about this, and that we had to exclude their papers from our research due to this reason.

During the snowball sampling phase three papers were excluded due to quality issues. One paper was excluded because it was not written in English and two references were excluded because they were not peer reviewed papers.

4.1.5 Data extraction

The data extraction was divided into two phases, identifying TPI approaches described by the primary studies (RQ1) and extracting detailed information about the approaches as basis for the evaluation of the approaches (RQ2). During the first phase, the name and, if available, the abbreviation of the TPI approach presented in the investigated paper was extracted.

For the second phase a data extraction form was prepared. For each TPI approach identified in the first phase of data extraction, the following information was extracted: 'Based on/influenced by', 'Domain', 'Developed by', 'Status of development', 'Completeness of information', 'Assessment model', 'Assessment procedure', 'Assessment instrument', 'Improvement suggestions', 'Process reference model', 'Maturity structure', 'Model representation', 'Character of approach', 'Structure/components', 'Addressing', 'Process areas'.

The extracted characteristics of the approaches can be explained as follows:

Based on/influenced by Earlier developed models or frameworks that function as basis or that have influenced the development of this approach.

Domain A specific domain which this approach is addressing. If empty, a specific domain is either not mentioned or it is explicitly said that the approach is universally applicable.

Developed by An institute, foundation or cooperation that developed the approach. If empty, the approach was developed by a single researcher or a smaller group of researchers, and an institute, foundation or cooperation was not explicitly mentioned.

Status of development There are two dimensions of the status of development possible: 'under development' or 'completed'. Status at the time of publication of the primary study or if available, according to the information obtained by contacting the author(s). If the research was validated by case studies, surveys or experiments, this is also mentioned.

Completeness of information There are three dimensions regarding the completeness of the information possible: 'concept', 'brief description' or 'detailed description'. Papers assessed as 'concept' only present the idea of how the approach could look like. Normally, approaches that are assessed as 'under development' are only presented as concepts in the respective study. For approaches with 'detailed descriptions' all the information is available to apply the approach. Detailed information about the assessment process, the components and the structure of the approach is available. 'Brief descriptions' provide more information than concepts but not all elements of the approach are described in detail.

Assessment model It is checked if the approach provides an assessment model. An assessment model provides a framework for the result of the assessment. The assessment results are maturity levels, that determine the state of practice of the assessed organization.

Assessment procedure It is checked if the approach provides instructions how to perform the assessment.

Assessment instrument It is checked if the approach provides an instrument, e.g., a questionnaire, which is used for the assessment.

Improvement suggestions It is checked if the approach provides information about processes that need improvement to be able to move to a higher maturity level.

Process reference model It is checked if the approach provides a reference model that represents the ideal process which the organizations should be aiming for to reach the highest level of maturity.

Maturity structure It is checked if the approach uses maturity levels to assess an organization's test process. If yes, the maturity levels are listed.

Model representation There are two possible types of model representations, 'continuous' or 'staged'. In an approach with a continuous model, each process area is classified by a number of maturity levels, so that the maturity level of each process area can be assessed and improved separately. In an approach with a staged model a maturity level is composed by a set of specific process areas which are linked only to this specific maturity level. To evolve to the next higher maturity level all requirements of all process areas of this maturity level and the preceding maturity levels have to be satisfied.

Character of approach There are two dimensions, 'qualitative' or 'quantitative'. Qualitative approaches investigate the test process based on qualitative data, e.g., through interviews with employees. Quantitative approaches use quantitative data like metrics for the assessment of the test process.

Structure/components Describes the structure of the approach and its components.

Addressing If the approach is addressing specific roles in the organization under assessment, these are listed here.

Process areas Lists the aspects of the testing process that are investigated by the approach.

4.1.6 Evaluation criteria

In order to examine the TPI approaches found regarding their applicability for the case study the following exclusion criteria was determined:

- Has the development of the approach been completed?
- Is broad information about the approach available? (The information is more than a description of the concept.)
- Is there an assessment instrument (e.g., a questionnaire) available for this approach?
- Is the approach not specific to a domain?

TPI approaches for which one or more of these questions were answered with 'no', were excluded for the successional part of the study.

4.1.7 Validation of results

The findings of the systematic literature review were validated by the feedback from a set of authors of the selected primary studies. We contacted the authors by email which had replied during Phase 2 of the systematic review as well as the authors of the studies identified by the snowball sampling. A total of seven authors were contacted. Three authors replied and gave feedback to our evaluation. With one author we conducted an interview in which he answered our validation questions.

Table 4: Snowball sampling - results of study selection process. (Free indicates availability.)

Year	Reference	Free	Content	Exclusion criteria Other	Primary Study
1993	[Chaar et al., 1993]	yes	Technique for assessing the effectiveness of inspections and test activities		no
2003	[Burnstein, 2003]	yes			yes
2007	[Farooq et al., 2007]	no			no
2008	[Farooq et al., 2008a]	no			no
2007	[Farooq and Dumke, 2007]	yes			yes
2010	[IEE, 2010]	no			no
1997	[Ericson et al., 1997]	yes			yes
2010	[TMMi Foundation, 2010]	no			no
2005	[Karlström et al., 2005]	yes			yes
1995	[Kit and Finzi, 1995]	no			no
1999	[Burnstein et al., 1999]	yes			yes
1999	[Koomen and Pol, 1999]	yes			yes
2004	[TPI, 2004]	yes			yes
2006	[Koomen et al., 2006]	no			no
2002	[Pol, 2002]	yes	Software testing methodology		no
2013	[v. Ewijk et al., 2013]	yes			yes
2011	[TMM]	yes		Website	no
2008	[van Veenendal, 2008]	yes			yes
2005	[Koomen, 2005]	no			no
2005	[Miller et al., 2005]	yes		Presentation slides	no
2007	[Kim and Jang, 2007]	no			no
2000	[Jacobs et al., 2000]	yes			yes
2000	[Swinkels, 2000]	yes			yes
2008	[Sanz et al., 2008]	no		Not in English	no
2010	[Mala et al., 2010]	yes	Technique for test case generation		no
2012	[Steiner et al., 2012]	yes			yes
1996	[Suwannasart, 1996]	yes			yes
1988	[Gelperin and Hetzel, 1988]	yes	Evolution of software testing		no
2006	[Taipale et al., 2006]	no			no
1998	[Homyen, 1998]	yes			yes
2002	[Ham, 2002]	Other year	Contribution of software process improvement models to test process improvement		no
2001	[Ham and Veenendaal, 2001]	no			no
2006	[Kulkarni, 2006]	yes			yes
2002	[Koomen, 2002]	yes			yes
2004	[Koomen, 2004]	no			no

We provided them with a list of all TPI approaches that we found in the systematic literature review and asked them if they think this list is complete. Furthermore, we presented them our exclusion criteria that have been decisive for the pre-selection of applicable approaches for the case study and the particular inclusion/exclusion result for the particular approach presented by the contacted author. Individually, the authors were asked if they agree to the evaluation of their approach.

A summary of the inquiries and responses is given in Appendix D.

5 Execution of the case study

This section describes the case under investigation along with the case study protocol, the case description and the unit of analysis.

5.1 Case study design

The case study was designed and conducted by following Runeson and Höst's [2009] guidelines for conducting and reporting case study research in software engineering.

A good case study design and planning contributes significantly to its success. This study, as it was actually conducted, is characterized by the design presented in Table 5.

Even though a case study is planned beforehand, the design process of a case study is flexible which means that it allows for and even encourages changes of objectives and research questions within the actual execution of the study. In the present study, the possibility for changes during study iterations was further promoted by our overall research design based on the Technology Transfer Model. The study is composed by a number of sequential steps which are dependent on each other. The outcome of the preceding steps influences the research question addressed by the next step.

Especially after performing Step 5 of the Technology Transfer Model (the static validation) we had to change the objective and the following research questions of the study. In the beginning, it was planned to only select one TPI approach that would be applied in the case organization. However, since the selection process in industry resulted in two approaches, we decided to apply both approaches and compare their content and results (more of this is covered later in this section).

5.2 Case description

The organization under study is a part of Volvo IT which is a subsidiary of the Volvo Group, a large Swedish automotive organization. The team develops and maintains information systems within the product development (PD) and purchasing (PU) area for an external customer.

Both areas, PD and PU, consist of several different information systems and applications developed in a number of different programming languages. Systems in the PD area are handling product data needed for product development in automotive industry. PU systems manage, for example, suppliers information. In total, around 45 employees are working in the case organization, of which around 20 are located in Gothenburg (Sweden) and 25 in Bangalore (India).

Apart from line management the following roles could be found within the organization: Maintenance manager, project manager, coordinator, system analyst, business analyst and developer. Smaller teams composed by system and/or business analysts and developers are responsible for one or several of the systems/applications in either the PD or PU area. The developers are mainly located in India.

Testing is not seen as a major activity of the development or maintenance process. Within the team, there are no designated testing roles. Even though a corporate test policy is available for Volvo IT, it is unknown to which extent these guidelines are followed by the team. The processes are rather set in accordance to the requirements

Table 5: Case study design.

Study characteristics		
Objective	Exploratory	Identify TPI approaches valuable for the case organization, apply them and compare their content and their assessment results
Case	Holistic	Investigating the testing process and the team members involved in testing as a whole
Data collection	Qualitative	Collecting data through interviews, observation and documents
Triangulation	Data triangulation	Several interviews with team members with different roles, working in different areas and locations
	Methodological triangulation	Interviews, observations and document analysis.

of the external customer. Moreover, it is perceived that each team member follows her own testing process.

However, there is a general consensus that the quality of development deliverables is good. This notion is mainly based on the lack of frequent or serious complaints from customer side.

The testing policy is provided by a globally operating department of Volvo IT, called ADT (Application Development and Technology). The department is responsible for establishing standard processes in different areas of software development. Furthermore, they offer the service of testing process assessment.

The study is conducted as a holistic case study, the team members involved in testing and their testing process are studied as a whole.

During the whole study key personnel, in the following sections called as 'organization representatives', supported us in all decision making processes, e.g., interviewee selection. The 'organization representatives' are representing different levels of authority within the organization. They are composed of the line manager of the organization, the maintenance manager of each area, and one system/business analyst of each area.

5.3 Selection of TPI approaches

The systematic literature review resulted in a list of available TPI approaches. The use of several exclusion criteria checking the general applicability of the approaches in industry resulted in a narrowed down set of approaches possibly applicable in the case study. The selection of the actual approaches to be applied in the case organization was done during a workshop.

5.3.1 Workshop

The participants of the workshop were composed by the 'organization representatives' and two persons from outside the organization, who had shown interest in participating. Both of the external participants were members of the ADT team within Volvo IT (mentioned in Section 5.2). They worked in the area of testing in general and TPI in particular and had a keen interest in our study.

The workshop consisted of two steps: A presentation held by the two researchers followed by a cumulative voting. Both steps are described in more detail next.

Presentation The presentation started with an introduction to the research process and the objective of conducting the workshop. Then the results of the systematic literature review as well as the exclusion criteria used for the pre-selection of applicable TPI approaches were presented. Finally, the pre-selected approaches were explained in detail. The information provided for each approach was based on the following classifying parameters:

Developed by Which company, organization, research group or individual researcher developed this approach?

Based on Which approach/methodology is the approach based on? For example, it might be based on CMMi.

Model representation Which type of model representation is used in the approach? Continuous or staged?

Key elements What are the key elements of the approach? For example, checkpoints or specific goals and practices.

Process areas Which areas are investigated by the approach? For example, test strategy, stakeholder commitment or test policy.

Assessment procedure What is the assessment procedure of the approach? For example, interviews with open-ended questions.

Thereafter, detailed content-wise examples of the investigated process areas were provided.

During the presentation of the characteristics of the pre-selected approaches and the content-wise examples particular attention was given on emphasizing the differences between the approaches without rating these as advantages or disadvantages. The approaches were presented in a objective way without emphasizing any specific approach to prevent biased decisions.

After the presentation printed material about each of the presented approaches was handed out to all participants and an open discussion about the approaches was held. The discussion phase was mainly used to answer questions regarding the presentation. The workshop finally ended with a voting to decide which approach(es) should be applied in the organization under study.

Cumulative voting The decision which TPI approach was to be applied in the case organization was done by using the $100 method.

The $100 method is a cumulative voting method to make a selection between several alternative options. Each participant of the voting is provided with a virtual $100 to distribute between the options. The participants can distribute any amount between

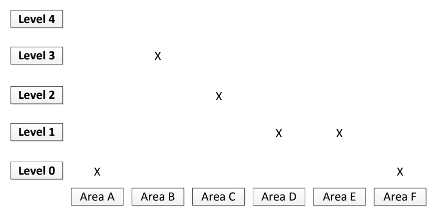

Figure 4: Continuous model representation.

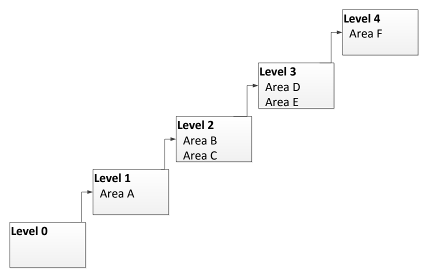

Figure 5: Staged model representation.

0 and 100 on any of the options. The only demand is that each participant has to distribute $100 in total at the end of the voting. The higher an amount spent on an option the more priority this option has. The option with the highest result will be selected.

All participants of the workshop except for the researchers had a vote with equal weighting. Each participant's vote consisted of $100 which could be distributed arbitrarily between the presented approaches with any amount between 0 and 100.

The selection process of the TPI approaches to be applied in the case organization resulted in two approaches, TPI NEXT and TMMi. We decided to use both approaches simultaneously and compare their results afterwards.

5.4 General information about TPI®NEXT and TMMi®

Here the concepts and especially the specific terminologies of both approaches are introduced to provide better understandability for the remainder of this section. Moreover, further information regarding the approaches is given in the presentation of the results of RQ 2 in Section 6.2.

5.4.1 Staged vs. continuous model representation

One significant difference between TMMi and TPI NEXT is their type of model representation. TMMi represents a staged model, whereas TPI NEXT represents a continuous model.

In a continuous model the main focus lies on the process areas (see Figure 4). Each process area is classified by a number of maturity levels, so that the maturity level of each process area can be assessed separately. This allows to focus on specific process areas during the assessment.

In a staged model the main focus lies on the maturity levels (see Figure 5). A maturity level is composed by a set of specific process areas which are linked only to this specific maturity level. To evolve to the next higher maturity level all requirements of all process areas of this maturity level and the preceding maturity levels have to be satisfied. The ideas for both types of model representation are taken from CMMi, which provides a staged as well as a continuous representation.

5.4.2 TPI®NEXT

The TPI NEXT model consists of seven elements: Key areas, maturity levels, checkpoints, improvement suggestions, enablers and clusters. The final element is then the results of an assessment, which are represented in the test maturity matrix.

Key areas The test process is distributed amongst key areas, which focus on different aspects of the test process. TPI NEXT has 16 key areas. Each key area may have different levels of maturity and the combination of the key areas defines the maturity of the test process as a whole. However, for each key area the maturity is measured individually.

Maturity levels The TPI NEXT model has four maturity levels: Initial, controlled, efficient and optimizing. A higher maturity level can only be reached if the preceding maturity level is fulfilled.

Checkpoints Checkpoints are statements regarding the test process. The question whether these stated requirements are satisfied by the investigated test process have to be answered with simple ‘yes’ or ‘no’ replies. A checkpoint always relates to a specific key area and a specific maturity level of the respective key area. A key area is at a certain maturity level when all its checkpoints are satisfied.

Clusters The model enables a stepwise growth from initial to optimizing levels. Each step is indicated by clusters of checkpoints. A cluster is a group of checkpoints from multiple key areas that function as one improvement step. A cluster is used for the purpose of increasing the maturity of the test process. Each cluster is identified by an alphabetic character that represents its position in the improvement path.

Enablers The test process and software development lifecycle model go hand in hand. The enablers help to understand how both processes can benefit from exchanging each others best practices and working closely together.

Improvement suggestions TPI NEXT provides recommendations in terms of improvement suggestions and helps the organization to obtain higher maturity by suggesting them how the checkpoints can be met. The improvement suggestions are practice-based, adaptable and optional to consider.

Test maturity matrix After conducting a test process assessment the analysis result is shown diagrammatically in a test maturity matrix. This matrix provides an overall picture of the current situation of the test process by highlighting the fulfilled checkpoints of all key areas.

Furthermore, the test maturity matrix provides an insight by showing a comparison between its current situation and what level should be achieved in the future to obtain higher maturity.

5.4.3 TMMi®

TMMi consists of five maturity levels: Initial, managed, defined, measured and optimization. Each maturity level is composed by a set of process areas. The process areas are specific aspects of the testing process that are deemed to be significant for the particular level of maturity. These process areas have to be fully considered in an organization's testing process in order to reach the respective level of maturity.

In more depth, process areas are further composed by three groups of components: Required, expected and informative components.

Required components Required components consist of specific and generic goals which must be achieved to fulfill the requirements of the specific process area and the associated maturity level. Specific goals are specific to a process area, whereas generic goals are generally defined statements recurring in all process areas.

Expected components Specific and generic goals are further described by specific and generic practices which belong to the group of expected components. These practices or acceptable alternatives are typically in place to achieve the goals.

Informative components Informative elements can be sub-practices, example work products, notes, examples or references. They serve as further information about specific and generic practices.

5.5 Mapping between TPI®NEXT and TMMi®

In order to compare the results of an assessment it is important to compare the approaches first to see if the two approaches are similar or if they differ to a great extent.

Table 6: Keywords extracted from TPI NEXT

Key area	Keywords
Stakeholder commitment	stakeholder, resource, commitment, product risk, test process
Degree of involvement	involvement, involved, lessons learned
Test strategy	test strategy, test level
Test organization	test organization, test policy
Communication	communication, test team
Reporting	report, product risk, lifecycle, test process
Test process management	test plan, evaluation
Estimation and planning	effort, estimation, test plan, dependency, techniques
Metrics	metrics
Defect management	defect, management, monitor, future
Testware mangement	management, test process, testware, documents
Methodology practice	methodology, test process, test methods, feedback, template
Tester professionalism	tester professionalism, training, test tasks, performance
Test case design	test case, test design, test basis
Test tools	test tool
Test environment	test environment, test environment requirement

Therefore, a mapping between the two selected approaches, TPI NEXT and TMMi, was done before the actual performance of the assessment.

The mapping of TPI NEXT and TMMi consisted of checking similarities or differences between the key areas of TPI NEXT and the process areas of TMMi. To obtain triangulation, this mapping was first performed by the two researchers individually.

Both researchers followed the same process, but they examined the approaches from different perspectives. Researcher A mapped the content of TPI NEXT to TMMi, while Researcher B mapped the content of TMMi to TPI NEXT. The mapping is illustrated in Figure 6 and is described as follows:

- Identification of keywords

  Keywords that represent the process areas of TMMi with its specific goals and respectively the key areas TPI NEXT with its checkpoints were identified. The keywords extracted from TMMi are shown in Table 7 and the keywords extracted from TPI NEXT are shown in Table 6.

Table 7: Keywords extracted from TMMi

Goal	Subgoal	Keyword
Test policy and strategy		test policy, test strategy
	Establish a test policy	test policy
	Establish a test strategy	test strategy
	Establish test performance indicators	performance indicator, performance, indicator
Test planning		test planning
	Perform a product risk sssessment	product risk assessment, risk
	Establish a test spproach	test approach
	Establish test estimates	test estimates, estimate, estimating
	Develop a test plan	test plan
	Obtain commitment to the test plan	commitment, test plan

Continued on next page

Table 7 – continued from previous page

Goal	Subgoal	Keyword
Test monitoring and control		test monitoring, test control, monitoring, control, monitor
	Monitor test progress against plan	progress
	Monitor product quality against plan and expectations	quality
	Manage corrective action to closure	corrective action, closure
Test design and execution		test design, test execution, design, execution
	Perform test analysis and design using test design techniques	test analysis, analysis, test design technique, test design
	Perform test implementation	test implementation, implementation, implement
	Perform test execution	test execution, execution
	Manage test incidents to closure	test incident, incident, closure
Test environment		test environment
	Develop test environment requirements	test environment requirement, test environment, requirement
	Perform test environment implementation	test environment implementation, implementation, implement
	Manage and control test environments	test environment
Test organization		test organization
	Establish a test organization	test organization
	Establish test functions for test specialists	test function, test specialist, specialist, function
	Establish test career paths	test career path, career path, career
	Determine, plan and implement test process improvements	test process improvement, improvement, improve
	Deploy the organizational test process and incorporate lessons learned	organizational test process, test process, incorporate lessons learned, lessons learned
Test training program		test training program, training
	Establish an organizational test training capability	training
	Provide test training	training
Test lifecycle and integration		test lifecycle, test integration, lifecycle, integration, integrate
	Establish organizational test process sssets	organizational test process asset, test process asset, test process, asset
	Integrate the test lifecycle models with the development models	test lifecycle model, development model, model
	Establish a master test plan	master test plan, master
Non-functional testing		non-functional, non functional
	Perform a non-functional product risk asssessment	product risk assessment, risk
	Establish a non-functional test approach	test approach
	Perform non-functional test analysis and design	test analysis, analysis, test design, design
	Perform non-functional test implementation	test implementation, implementation, implement
	Perform non-functional test execution	test execution, execution
Peer reviews		peer review, review

Continued on next page

Table 7 – continued from previous page

Goal	Subgoal	Keyword
	Establish a peer review approach	peer review, review
	Perform peer reviews	peer review, review
Test measurement		test measurement, measurement
	Align test measurement and analysis activities	test measurement, measurement, analysis
	Provide test measurement results	test measurement result, measurement result, measurement, result
Product quality evaluation		product quality evaluation, quality, evaluation
	Project goals for product quality and their priorities are established	project goal, product quality, priority, priorities, goal, quality
	Actual progress toward achieving the project's product quality goals is quantified and managed	progress, product quality goal, product quality, quality, goal
Advanced reviews		review
	Coordinate the peer review approach with the dynamic test approach	peer review approach, peer review, review, dynamic test approach, test approach, dynamic
	Measure product quality early in the lifecycle by means of peer reviews	product quality, quality, lifecycle, peer review, review
	Adjust the test approach based on review results early in the lifecycle	test approach, review result, review, lifecycle
Defect prevention		defect prevention, defect, prevention
	Determine common causes of defects	cause, defect
	Prioritize and define actions to systematically eliminate root causes of defects	root cause
Quality control		quality control, quality
	Establish a statistically controlled test process	statistic, test process
	Testing is performed using statistical methods	statistic
Test process optimization		test process optimization, test process, optimization, optimize
	Select test process improvements	test process improvement, test process, improvement, improve
	New testing technologies are evaluated to determine their impact on the testing process	test process, testing process
	Deploy test improvements	improvement, improve
	Establish re-use of high quality test process assets	test process, re-use, test process asset

- Search for keywords

  The key words identified in one approach were searched in the other approach. Hits were documented in a matrix which showed at which location the key words were found in the searched approach.

  For better search results, the data basis for the search was extended to specific goals besides process areas in TMMi and checkpoints besides key areas in TPI

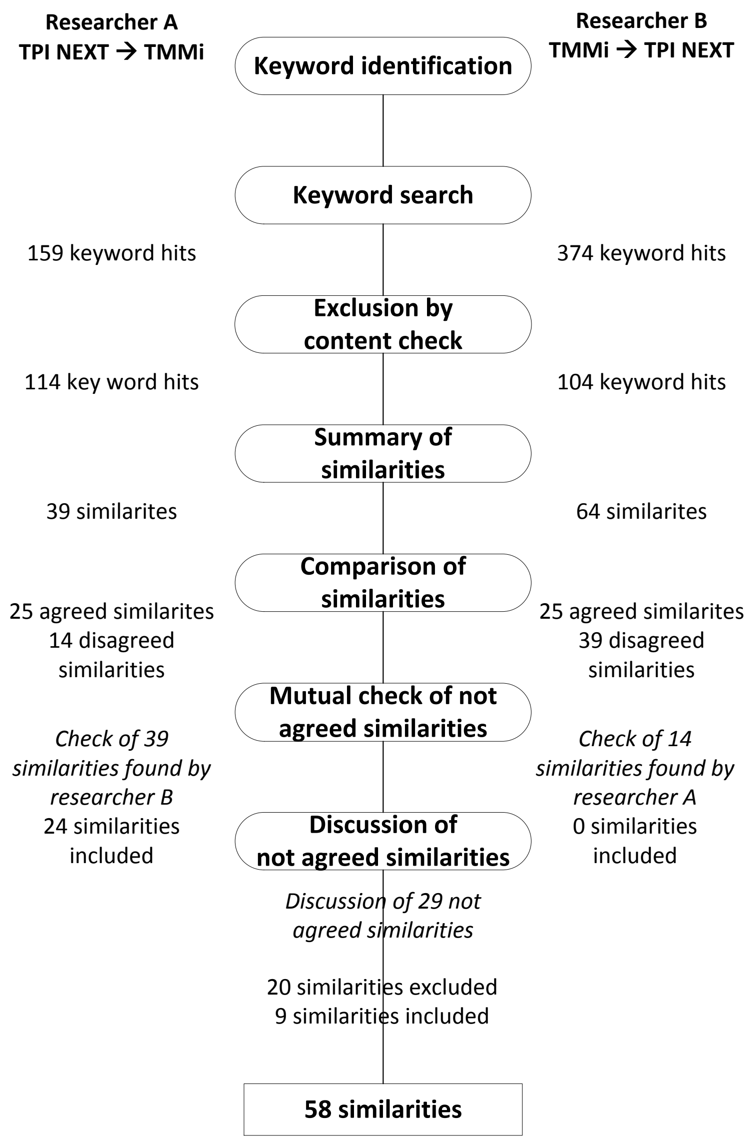

Figure 6: Mapping between TPI NEXT and TMMi.

NEXT. The search of keywords from TPI NEXT in TMMi by Researcher A resulted in 159 hits, and the search of keywords from TMMi in TPI NEXT by Researcher B resulted in 374 hits.

- Inclusion and exclusion of hits based on their context

  The contents of the process areas (TMMi) and key areas (TPI NEXT) that contained the identical keywords were checked upon whether they convey the same meaning and appear in the same context in both approaches.

  Researcher A excluded 45 keyword hits in which the keywords were not used in the same context in both approaches. Researcher B excluded 270 keyword hits.

- Summary of individually found similarities between TPI NEXT and TMMi

  The extended data basis for the keyword search was now narrowed down to process areas and key areas only. Keyword hits from lower levels were transferred to the corresponding higher levels. The results were summarized to 39 similarities found by Researcher A and 64 similarities found by Researcher B.

- Comparison of individually found similarities

  The mapping results of both researchers were compared. In total, 25 of the found similarities between TPI NEXT and TMMi had been found by both researchers, while 14 similarities had been only found by Researcher A and 39 had been only found by Researcher B.

- Mutual check of not agreed similarities

  All similarities only identified by one researcher were checked by the other researcher. Researcher A checked the 39 similarities that were only identified by Researcher B, and Researcher B checked the 14 similarities that were only identified by Researcher A. In this step Researcher A agreed to include 24 similarities found by Researcher B. Researcher B did not include any similarities in this step.

- Final discussion of not agreed similarities

  The remaining 29 similarities found by only one researcher were now discussed by both researchers. Both researchers presented their arguments for exclusion or inclusion of these similarities between TPI NEXT and TMMi. In the discussion, the researchers agreed to exclude 20 and to include 9 alleged similarities.

Finally, a total of 58 similarities between TPI NEXT and TMMi were identified.

5.6 Test process assessment using TPI®NEXT and TMMi®

Besides an independent comparison of the content of TPI NEXT and TMMi, the mapping between the two approaches builds a solid basis for the comparison of the results from the application of the approaches. The assessment of the case organization's testing process using TMMi will result in one test maturity level valuing the process as a whole. The assessment result of TPI NEXT will be a matrix stating the maturity level for each key area separately.

For the application of TPI NEXT and TMMi the instructions given in [v. Ewijk et al., 2013] and [van Veenendal, 2008] were followed. However, both approaches demand for an assessment execution by experienced personal. TPI NEXT either proposes to perform the assessment by an individual who is familiar with the test processes or the BDTPI (Business Driven Test Process Improvement) model or recommends the use of an external expert. In TMMi the requirements in regards to the assessor are even stricter. The TMMi Assessment Method Application Requirements (TAMAR) state that a formal assessment can only be performed by an accredited Lead Assessor. The necessary accreditation can only be gained by the TMMi Foundation.

Assessments without an accredited assessor can only be performed as informal assessments. Formal and informal assessments mainly differ in the presentation of the assessment result. Only formal TMMi assessments allow the statement of the maturity level of the assessed organization. Informal assessments result in a report describing the state of practice in the assessed organization. Due to the absence of an accredited assessor we could base our assessment only on the instructions of an informal assessment. Nevertheless, since the objective of the application of the approaches was to compare their assessment results, we adapted the procedures proposed by the approaches with respect to this aim.

The assessment process of both approaches is generally similar, collection of data through interviews, data analysis and documentation of results.

The use of the two different approaches in this study was split between the researchers. Researcher A was responsible for the TPI NEXT assessment and Researcher B was responsible for the TMMi assessment. However, due to time limits and for the convenience of all participants, we decided to have joint interviews for both approaches.

5.6.1 Data collection

The data needed for the test process assessments was mainly collected through interviews. Additionally, testing documents and processes that were identified during the interviews as relevant for the assessment, were studied and observed. The reason for collecting data from several sources was to include triangulation in our study to make our conclusion stronger and to eliminate chances of only one interpretation.

Interviewee selection The participants were selected with the help of the 'organization representatives'. The selection of a team member as an interviewee was based on her involvement in testing activities. Furthermore, the selected interviewees should present a representative sample of the population. Therefore, both areas, PD and PU, and both development sites, Gothenburg and Bangalore, were covered as well as all roles related to testing activities.

Two members of the group of 'organization representatives' were also selected as interviewees. Besides their professional knowledge of the case organization's testing process they were selected because their interviews served as pilot studies.

Table 8: Interviewee description.

Interviewee no.	Role	Experience in organization [years]	Location	Area
1	System analyst	2	Gothenburg	PU
2	System analyst	2.5	Gothenburg	PD
3	System analyst	24	Gothenburg	PU
4	System analyst	10	Gothenburg	PD
5	Project manager	2	Gothenburg	PU/PD
6	Business analyst	22	Gothenburg	PD
7	Application developer	2	Bangalore	PD
8	Application developer	2.5	Bangalore	PU
9	Application developer	1.2	Bangalore	PU
10	Application developer	2.5	Bangalore	PU
11	Application developer	5	Bangalore	PU
12	Application developer	2	Bangalore	PD

An anonymized list of all interviewees stating their roles, working area and their current location is specified in Table 8.

Interview design The interview questions were designed with respect to the aim of having joint interviews for both approaches. Due to this objective we decided to have semi-structured interviews with mainly open ended questions. This strategy aimed in getting maximum information from one question. With general phrased, open ended questions we aimed in combining the overall content of all key areas of TPI NEXT and all process areas of TMMi in one common questionnaire. Furthermore, available questionnaires from TPI approaches served as input to the process of interview question development [Taipale and Smolander, 2006] [Kasoju et al., 2013]. The feedback from the interviewees of the two pilot interviews was additionally used to reframe and rephrase the questions after conducting these first two interviews. The semi-structured interview approach allowed us to adjust the course of the interview, the set of asked questions and their level of detail according to the interviewees role and her knowledge.

Basically, the interviews were structured by the following themes:

Introduction A short introduction to research topic and process was given.

Warm-up questions Questions regarding the interviewee's age, educational background, years of experience in the case organization and in IT in general were covered in this theme.

Overview of work tasks Questions regarding the interviewee's usual work tasks and her involvement in testing served as important information for the further course of the interview.

Questions specific to testing This was the major section in which we tried to cover all process areas. Therefore most of the questions were specific to testing such as regression testing, test environment, testing with respect to product risks, test plan, test cases, testing tools, defects and training on testing.

Statistical questions about the interview These questions were ask to get the opinion concerning interview design, questions, duration and the general feeling about the interview.

The complete set of pre-designed questions is given in Appendix E.

Execution of the interview Prior to the interview phase emails were sent to all interviewees briefly describing the purpose and process of the study and the purpose and relevance of the interviews. Except for the two pilot interviews, the duration of the interviews was set to a maximum of 60 minutes. All interviews were recorded in an audio format and, additionally, notes were taken. The interviews were conducted in person with the participants in Gothenburg (Sweden) while telephone interviews were conducted with the interviewees in Bangalore (India).

As basis for the data analysis, the contents of all interviews were briefly transcribed after the interview phase. The individual transcript of each interview was sent to the respective interviewee with the request to check the content for its correctness. Occurring changes were considered later in the data analysis phase.

Observation Observation helps to understand processes better by seeing the actual performance of the process. For some of the processes or system feature the researchers made observation to get in-depth understanding about their processes by sitting next to the interviewees when they were actually executing any tests or related process.

Document analysis Process documents such as test policy, software test description, test cases, test plans, testing reports and all other documents related to testing were studied to gain a deep understanding of the organization processes and standards which in turn helped in understanding and analyzing the interview data.

5.6.2 Data analysis

The data collection phase was followed by data analysis. Since the main focus lay on assessment of state of practice with respect to test process and not the identification and implementation of improvements the instructions regarding improvement suggestions were neglected during data analysis. Especially, the process of TPI NEXT was affected by this decision.

The main element of the assessment with TPI NEXT is the verification of the checkpoints provided by the model. Based on the interview data, the documents studied and the processes observed Researcher A checked the fulfillment of the checkpoints for each key area. Fulfilled checkpoints were marked with 'yes' and not fulfilled checkpoints were marked with 'no'. The results were documented in a spreadsheet provided on the TPI NEXT website. The spreadsheet automatically produces the TPI NEXT Test Maturity Matrix which highlights the fulfilled checkpoints in the respective maturity level of each key area [1]. Due to the limitation to the assessment of the state of practice the consideration of clusters was disregarded.

[1] `http://www.tmap.net/en/tpi-NEXT/downloads`

In a formal assessment of TMMi the result is based on the degree of fulfillment of specific and generic goals. TMMi provides a rating scale which specifies the degree of fulfillment in detail. In an informal assessment, as described by the TMMi Foundation, this procedure is not proposed. However, since we needed to build a basis on which we could compare the results of the TPI NEXT assessment and the TMMi assessment with respect to our mapping of the two approaches we adapted the assessment procedure to this purpose. Based on the interview data Researcher B checked the fulfillment of the specific and generic goals associated to the process areas of maturity Level 2. The fulfillment for each specific and generic practice was classified by the following rating: 'fully fulfilled', 'partly fulfilled' or 'not fulfilled'.

If the organization's testing process is performed exactly like the practices proposed by the TMMi model or by an acceptable alternative this practice is marked as 'fully fulfilled'. If only particular steps of the proposed practices are in place in the testing process of the organization this practice is marked as 'partly fulfilled'. If a TMMi practice is not followed at all this practice is marked as 'not fulfilled'. Due to the staged character of the TMMi model an assessment of a higher level is not needed if the goals of the preceding level are not fully fulfilled. Therefore only the process areas and goals of TMMi Level 2 were investigated.

Both, the assessment procedure of TPI NEXT and the informal assessment of TMMi do not require the assessor to provide particularly strong or multiple evidences for her decision if a checkpoint or a goal is fulfilled or not. Hence, the decision relies on the assessor's interpretation of the interview data with respect to the compliance with the model provided by the respective approach. Both researchers agreed upon an analysis procedure in which a checkpoint or a goal was stated as fulfilled if an indication of the fulfillment was given by at least one interviewee.

6 Results

In this section the results of the study are presented. The section is structured by the research questions answered by this study.

6.1 Research question 1

To answer RQ 1 (Which different TPI approaches can be found in literature?), a systematic literature review, as described in Section 4.1, was conducted. The TPI approaches found by the literature review are presented in Table 9.

In total, 18 approaches have been identified. They are described in more detail in Section 6.2.

The studies [Farooq and Dumke, 2008],[Farooq and Dumke, 2007] and [Swinkels, 2000] have been identified as primary studies related to the research question since they are discussing TPI approaches. However, these studies are not listed in Table 9 because they are not explicitly presenting one specific approach but rather comparing several approaches. Most of the approaches presented in these papers are already included in the approaches listed in Table 9. Nevertheless, some approaches mentioned in these papers have not been further considered in this study due to lack of available information or due to generalizability issues (since they have only been developed for a specific company).

6.2 Research question 2

To answer RQ 2 (What are the specific characteristics of these Test Process Improvement approaches?), the data about the approaches presented in Section 4.1.5 was extracted from the primary studies found by the systematic review. The characteristics of each approach are presented in Appendix F.

Based on their main concept the approaches can be grouped into four groups:

- TMM and related approaches.
- TPI and related approaches.
- Standards and related approaches.
- Individual approaches.

In the following paragraphs the identified approaches are described. Brief background information and the most important characteristics are pointed out for each approach.

Table 9: Found approaches.

Ref	Approach	Abbreviation
[Ryu et al., 2008]	Ministry of National Defense-Testing Maturity Model	MND-TMM
[Taipale and Smolander, 2006]	Observing Practice	
[Farooq et al., 2008b]	Meta-Measurement approach	
[Jung, 2009]	Embedded Test Process Improvement Model	Emb-TPI
[Rana and Ahmad, 2005]	Testing Maturity Model	TMM
[Burnstein et al., 1996]	Testing Maturity Model	TMM
[Xu-Xiang and Wen-Ning, 2010]	Plan-Do-Check-Action (PDCA)-based software testing improvement framework	
[Tayamanon et al., 2011]	Testing Maturity Model	TMM
[Jacobs and Trienekens, 2002]	Metrics Based Verification and Validation Maturity Model	MB-VV-MM
[Kasoju et al., 2013]	Evidence-based Software Engineering	
[Kasurinen et al., 2011a]	Self-Assessment framework for ISO/IEC 29119 based on TIM	
[Heiskanen et al., 2012]	Test Process Improvement Model for Automated Test Generation	ATG add-on for TPI
[Rasking, 2011]	Test Maturity Model integration	TMMi®
[Reid, 2012]	Software Testing Standard ISO/IEC 29119, ISO/IEC 33603	
[Burnstein, 2003]	Testing Maturity Model	TMM
[Farooq and Dumke, 2007]	Several	
[Ericson et al., 1997]	Test Improvement Model	TIM
[Karlström et al., 2005]	Minimal test practice framework	MTPF
[Burnstein et al., 1999]	Testing Maturity Model	TMM
[Koomen and Pol, 1999]	Test Process Improvement	TPI
[TPI, 2004]	TPI®Automotive	TPI®Automotive
[v. Ewijk et al., 2013]	TPI®NEXT	TPI®NEXT
[van Veenendal, 2008]	Test Maturity Model integration	TMMi®
[Jacobs et al., 2000]	Testing Maturity Model	TMM
[Steiner et al., 2012]	Test SPICE	
[Suwannasart, 1996]	Testing Maturity Model	TMM
[Homyen, 1998]	Testing Maturity Model	TMM
[Koomen, 2002]	Test Process Improvement	TPI

TMM and related approaches

TMM - Testing Maturity Model The Testing Maturity Model was developed by a research group at the Illinois Institute of Technology in the late 1990s. Its purpose is to assess and improve testing processes in software development organizations. Furthermore it can be used as a model to represent the ideal incrementally growing testing process. Especially assessments from inside the company are possible. Amongst other sources, the development of TMM was influenced by CMM. The need for the development of the model emerged since existing evaluation frameworks did not address testing in a sufficient way. Especially the structure of TMM is inspired by CMM. It represents a staged model and consists of the following components:

- Five maturity levels: Initial, Phase Definition, Integration, Management and Measurement, Optimization/Defect Prevention and Quality Control,
- maturity goals (MG), maturity subgoals (MSG) and activities and tasks with responsibilities (ATR) and,
- an assessment model (TMM-AM).

TMMi®- Test maturity model integration TMMi is generally known as the successor of TMM. It was developed by the TMMi Foundation. The TMMi Foundation, a non-profit organization, was founded in 2005 by a group of leading test and quality practitioners. Their aim was to develop a testing model which covers the experience and best practices of a broad group of experts and would find acceptance in industry. Besides TMM as development basis, TMMi was influenced by CMMi. TMMi consists of:

- Five maturity levels: Initial, Managed, Defined, Measured, Optimization,
- process areas in each maturity level,
- required components: Specific and generic goals,
- expected components: Specific and generic practices and,
- informative components: Sub-practices, example work products, notes, examples or references.

The TMMi maturity levels have been inspired by the TMM maturity structure but further developed according to industry needs. Whereas the introduction of required, expected and informative components was established due to the influence of CMMi. Most generic goals and practices were even adopted from CMMi.

MND-TMM - Ministry of National Defense-Testing Maturity Model MND-TMM was developed to address the specific needs of weapon software system development. It combines the concepts of several approaches. It was influenced by TMM and TMMi and uses the continuous representation of CMMi. Furthermore, an OWL ontology is used to describe the elements of the model. Most elements of MND-TMM have been adopted from TMMi like specific and generic goals.

The model consists of ten process areas which are summarized in four categories - Military, Process, Infrastructure and Techniques. Each process area has five maturity levels. Due to the use of a continuous model the maturity of each process area can be assessed individually.

MB-VV-MM - Metrics based verification and validation maturity model The MB-VV-MM is a quantitative framework to improve validation and verification processes. Metrics are used to select process improvements and to track and control the implementation of improvement actions. The approach was based on TMM and enhanced by additions to specially support the validation and verification process. Similar to TMM, it consists of five maturity levels.

TIM - Test Improvement Model The Test Improvement Model serves as a guidebook for improvements of the test process and focuses explicitly on cost-effectiveness and risk management. Its intention is to identify the current state of practice with strong and weak elements and to make suggestions how to further strengthen the strong elements and to improve the weak elements. It was inspired by SEIs Capability Maturity Model and Gelperin's Testability Maturity Model.

TIM belongs to the group of continuous models and it is seen as the first step of the PDCA method, the planning phase. The model consists of five key areas. Each key area has five levels of maturity: Initial, baselining, cost-effectiveness, risk-lowering and optimizing, which are each represented by one overall goal and several subgoals.

TPI and related approaches

TPI - Test Process Improvement The Test Process Improvement model was developed in a Dutch company called IQUIP in the late 1990s. The model is based on the test approach TMap. It helps analyzing the current situation and identifying strengths and weaknesses of an organization's test process.

TPI is a continuous approach. It consists of 20 key area which represent different points of view on the test process. Each key area can have up to four levels of maturity. Checkpoints are used to determine the maturity level of each key area. They are requirements that have to be met for a test process to be classified in a specific level of maturity.

A Test Maturity Matrix provides an overview of the testing maturity of the assessed organization by highlighting the satisfied checkpoints and maturity levels per key area.

TPI®NEXT TPI NEXT is the successor of TPI, developed by the Dutch company Sogeti (a corporate merger of IQUIP and other companies). Compared to the original TPI approach the number of key areas in TPI NEXT has been reduced to 16 and additional elements— enablers and clusters—have been introduced to the model to more efficiently address industry needs in Test Process Improvement.

TPI Automotive A further approach developed by the Dutch company Sogeti is TPI Automotive. It follows the same principles as TPI but was specifically adapted to the needs of software testing in automotive industry.

ATG add-on for TPI - Test Process Improvement Model for Automated Test Generation This approach represents an add-on for the existing TPI to address the aspects of automated test generation in Test Process Improvement, especially the use of formal methods. The add-on extends TPI by:

- new maturity levels in the key areas of 'Static Test Techniques' and 'Test Specification Techniques',
- new key areas 'Modeling approach', 'Use of models', 'Test confidence', Technological and methodological knowledge' and
- new checkpoints.

Emb-TPI - Embedded Test Process Improvement Model Embedded TPI focuses on improving the testing process for embedded software by especially considering hardware issues of testing. The model consists of the following elements:

- capability model,
- maturity model,
- test evaluation checklist,
- evaluation & improvement procedure and,
- enhanced test evaluation model.

Standards and related approaches

Test SPICE The intention of developing Test SPICE was to provide a process reference model (PRM) and process assessment model (PAM) specific for test process assessment in conformance with the requirements of ISO/IEC 15504 II. Using ISO/IEC 15504 V as a starting point and reusing its structure, the Test SPICE model was developed by:

- identically transferring processes from ISO/IEC 15504 V to Test SPICE,
- replacing original processes from ISO/IEC 15504 V with specific test processes,

- renaming processes of ISO/IEC 15504 V and,
- inserting new specific test processes to Test SPICE.

Software Testing Standard ISO/IEC 29119, ISO/IEC 33063 ISO/IEC 29119 is a testing standard. The need for this standard was identified due to the traditionally poor coverage of testing in standards. Available standards with respect to testing cover only small, particular parts of testing, not the overall testing process.

ISO/IEC 29119 is divided into four parts: concepts and terminology, test process, test documentation and test techniques. By working in accordance to the process proposed in the standard a specific product quality can be guaranteed. In addition, ISO/IEC 33063, the process assessment standard related to the testing standard, provides a means to assess the compliance of a testing process to ISO/IEC 29119.

Self-Assessment framework for ISO/IEC 29119 based on TIM The goal of this approach is to provide an assessment framework that checks the compliance of an organization's test process with the standard ISO/IEC 29919. Therefore, the concept of the Test Improvement Model (TIM) with its maturity levels has been combined with the propositions of the standard. The model is divided into three levels: Organizational, project and execution level. Similar to TIM this approach has five maturity levels: Initial, baseline, cost-effectiveness, risk-lowering and optimizing, and also follows the continuous approach which means that the key areas are assessed separately.

Individual approaches

Meta-Measurement approach This approach focuses on the specification and evaluation of quality aspects of the test process. It is based on the concept of Evaluation Theory [Ares Casal et al., 1998] and it has been adapted to address the test process sufficiently. It consists of the following steps:

- Target(Software Test Processes).
- Evaluation Criteria (Quality Attributes).
- Reference Standard (Process Measurement Profiles).
- Assessment Techniques (Test Process Measurements).
- Synthesis Techniques (Quality Matrix, Quality Indexes).
- Evaluation Process.

Plan-Do-Check-Action (PDCA)-based software testing improvement framework The PDCA-based software testing improvement framework was developed to specifically address test processes provided as services by third party testing centers. The concept of this approach is based on the hypothesis that knowledge management plays an important role in process improvements. The framework is divided into the following phases:

- Build a learning organization through knowledge management.
- Plan the adaptive testing processes.
- Plan implementation and data analysis.
- Continuous improvement.

Evidence-Based Software Engineering In this individual approach, improvements for the test process are identified by the use of evidence-based software engineering. First, challenges in the testing process of an organization are identified by interviews. Then, solutions to these challenges are searched by a systematic literature review. Finally, an improved test process is presented by value-stream mapping.

Observing Practice In this approach the test process is studied by conducting detailed interviews with varying roles involved in testing in several interview rounds. The data gained by the interviews is analyzed by the use of grounded theory. Problems and at the same time possible solutions are identified by the analysis.

MTPF - Minimal test practice framework MTPF is a light-weight approach which addresses smaller organizations. Its goal is to increase acceptance of proposed improvements by the involvement of the entire organization. The framework addresses five categories which correspond to areas in testing. The introduction of process improvement is leveled in three phases which are adapted to the size of the organization.

6.2.1 Research question 2.1

To answer RQ 2.1 (Which approaches are generally applicable in industry?), the exclusion criteria specified in Section 4.1.6 was applied on the 18 TPI approaches identified by the systematic review.

This exclusion procedure led to a set of six approaches generally applicable due to their completed development, the sufficient information provided about the approach, an available assessment instrument and its independency from a specific domain.

These six approaches are TMM, TMMi, TPI, TPI NEXT, Test SPICE and Observing Practice. With regard to the four groups of approaches identified in the previous section, these six approaches represent each group.

Even though TPI NEXT is the successor of TPI, and the concept of TMMi is based on TMM and TMMi is often also seen as the successor of TMM, these approaches are still considered separately in this study.

6.2.2 Research question 2.2

To answer RQ 2.2 (Which approaches are valuable for test process improvements for the company under study?), a workshop was conducted in the case organization in which the approaches identified as generally applicable were presented and two approaches were selected by a cumulative voting. The results of the voting are presented in Table 10.

Table 10: Results from applying cumulative voting.

	TPI	TPI®NEXT	TMM	TMMi®	TestSPICE	Observing Practice
Participant 1	0	40	20	39	0	1
Participant 2	0	50	0	50	0	0
Participant 3	0	60	0	40	0	0
Participant 4	0	50	0	50	0	0
Participant 5	0	0	0	100	0	0
Participant 6	0	100	0	0	0	0
Participant 7	0	100	0	0	0	0
Total	0	400	20	279	0	1

An objective first look at the results shows that TPI NEXT received the highest scores with 400 points and TMMi got the second highest scores with 279 points. Clearly behind are the scores for the third and fourth ranking. On the third rank is TMM with 20 points and Observing Practice reached the fourth rank with only 1 point. TPI and TestSPICE did not get any votes.

Considering the knowledge and experience in the field of Test Process Improvement of two of the participants, the interpretation of the results requires a different perspective. Unlike the other participants of the workshop Participants 6 and 7 already had detailed knowledge about TPI prior to the workshop. One of them even has significant experience in performing test process assessments using TPI.

If the votes of Participants 6 and 7 were disregarded, TMMi would have received the highest scores with 279 points, compared to TPI NEXT with 200, TPI with 20 and Observing Practice with 1 point. Due to the fact that in both perspectives TPI NEXT and TMMi clearly obtained the highest rankings we decided to apply both approaches in the case study.

6.3 Research question 3

To answer RQ 3 (How well can the content of the selected approaches be mapped to each other?), a mapping based on keywords extracted from both approaches respectively and the search of these keywords in the other approach was conducted.

TMMi Specific Goal	TPI Next Key area	Stakeholder commitment			Degree of involvement			Test strategy			Test organization			Communication			Reporting			Test process management			Estimating and planning		
		C	E	O	C	E	O	C	E	O	C	E	O	C	E	O	C	E	O	C	E	O	C	E	O
2	Test policy and strategy	X						X	X	X		X													
2	Test Planning	X				X		X	X								X	X		X	X		X	X	X
2	Test Monitoring and control		X											X				X		X					
2	Test Design and Execution																								
2	Test Environment																								
3	Test Organization						X				X	X	X									X			
3	Test Training Program																								
3	Test Lifecycle and Integration										X								X			X			
3	Non-functional Testing																								
3	Peer Reviews																								
4	Test Measurement																X		X						
4	Product Quality Evaluation		X											X											
4	Advanced Reviews																		X						
5	Defect Prevention					X																			
5	Quality Control																								
5	Test Process Optimization												X						X			X			

X Similarity between TMMi and TPI NEXT

Figure 7: Mapping between TPI NEXT and TMMi—Results Part 1.

TMMi Specific Goal \ TPI Next Key area		Metrics			Defect management			Testware management			Methodology practice			Tester professionalism			Test case design			Test tools			Test environment		
		C	E	O	C	E	O	C	E	O	C	E	O	C	E	O	C	E	O	C	E	O	C	E	O
2	Test policy and strategy																								
2	Test Planning	x																							
2	Test Monitoring and control				x																		x		
2	Test Design and Execution																x	x							
2	Test Environment																						x	x	
3	Test Organization												x	x	x	x									
3	Test Training Program													x											
3	Test Lifecycle and Integration										x														
3	Non-functional Testing																	x							
3	Peer Reviews																								
4	Test Measurement		x																						
4	Product Quality Evaluation		x																						
4	Advanced Reviews																								
5	Defect Prevention						x												x						
5	Quality Control																								
5	Test Process Optimization												x						x			x			

x Similarity between TMMi and TPI NEXT

Figure 8: Mapping between TPI NEXT and TMMi—Results Part 2.

The similarities between the process areas of both approaches are presented in Figures 7 and 8 as the results of the mapping.

For the interpretation of the results it is crucial to take into consideration the different model representations of TPI NEXT and TMMi. TPI NEXT is a continuous approach. Each key area can be assessed individually by all maturity levels. On the other hand, TMMi is a staged approach. The process areas are linked to the maturity level.

Most of the aspects covered by lower levels of maturity in the key areas of TPI NEXT can by found in the process areas of Maturity Level 2 of TMMi. Exceptions are the key areas 'Testware management', 'Methodology practice', 'Tester professionalism' and 'Test tools'. None of the aspects of these key areas are covered in Maturity Level 2 of TMMi. However, lower maturity aspects of the key areas 'Methodology practice' and 'Tester professionalism' are covered by Maturity Level 3 of TMMi.

The aspects of TPI NEXT's 'Testware management' key area are not covered by TMMi at all. And likewise, the process area 'Quality Control' of TMMi is not addressed by TPI NEXT at all.

On the contrary, even though aspects of all maturity levels of the TPI NEXT key areas 'Test strategy', 'Test organization', 'Reporting', 'Test process management', 'Estimating and planning', 'Tester professionalism' and 'Test case design' are covered by process areas of TMMi, the maturity levels of these TPI NEXT key areas do not exactly correspond to the respective maturity levels in TMMi. While the aspects of all maturity levels of TPI NEXT's key area 'Test strategy' correspond to TMMi's process areas 'Test policy and strategy' and 'Test planning' in Maturity Level 2 and the aspects of all maturity levels of the key area 'Estimating and planning' in TPI NEXT correspond to 'Test planning' also in Maturity Level 2 of TMMi, the aspects of TPI NEXT's 'Tester professionalism' are reflected by the process areas 'Test organization' and 'Test training program' in Maturity Level 3 of TMMi. Furthermore, the aspects of the key areas 'Test organization', 'Reporting', 'Test process management' and 'Test case design' are corresponding to process areas of different maturity levels of TMMi.

However, most aspects addressed by process areas in higher maturity levels of TMMi (Levels 4 and 5) are accordingly addressed by the highest maturity level (optimizing) in the key areas of TPI NEXT. And likewise, most aspects addressed by process areas in lower maturity levels of TMMi (Levels 2 and 3) are addressed by lower maturity levels (controlled and effective) in the key areas of TPI NEXT.

6.4 Research question 4

To answer RQ 4 (How do the results of the selected approaches differ after applying them?), both approaches, TPI NEXT and TMMi, were used in parallel to assess the case organization's test process.

Both assessments came to the result that the maturity of the case organization's test process is still in the initial level.

Since this result is not very meaningful yet, a closer look is taken to the detailed results of both assessments and these results are compared with regards to the earlier performed mapping of the approaches against each other.

TMMi Specific Goal		TPI Next Key area	Stakeholder commitment			Degree of involvement			Test strategy			Test organization			Communication			Reporting			Test process management			Estimating and planning		
			C	E	O	C	E	O	C	E	O	C	E	O	C	E	O	C	E	O	C	E	O	C	E	O
			PF	NF	NF	PF	PF	PF	PF	PF	NF	NF	NF	NF	PF	NF	NF	PF	NF	NF	PF	NF	NF	PF	NF	NF
2	Test policy and strategy	PF	X						X	X	X		X													
2	Test Planning	PF	X				X		X	X								X	X		X	X		X	X	X
2	Test Monitoring and control	PF		X											X				X		X					
2	Test Design and Execution	PF																								
2	Test Environment	PF																								
3	Test Organization	NA						X				X	X	X									X			
3	Test Training Program	NA																								
3	Test Lifecycle and Integration	NA										X								X			X			
3	Non-functional Testing	NA																								
3	Peer Reviews	NA																								
4	Test Measurement	NA																X		X						
4	Product Quality Evaluation	NA		X											X											
4	Advanced Reviews	NA																		X						
5	Defect Prevention	NA					X																			
5	Quality Control	NA																								
5	Test Process Optimization	NA												X						X			X			

X Similarity between TMMi and TPI NEXT

Similarities in assessment results

Differences in assessment results

Figure 9: Mapping between TPI NEXT and TMMi. Comparison of assessment results Part 1.

TMMi Specific Goal		TPI Next Key area	Metrics			Defect management			Testware management			Methodology practice			Tester professionalism			Test case design			Test tools			Test environment		
			C	E	O	C	E	O	C	E	O	C	E	O	C	E	O	C	E	O	C	E	O	C	E	O
			NF	NF	NF	FF	PF	NF	NF	PF	NF	NF	NF	NF	NF	NF	NF	PF	NF	NF	PF	PF	PF	PF	NF	PF
2	Test policy and strategy	PF																								
2	Test Planning	PF	X																							
2	Test Monitoring and control	PF				X																		X		
2	Test Design and Execution	PF																X	X							
2	Test Environment	PF																						X	X	
3	Test Organization	NA												X	X	X	X									
3	Test Training Program	NA													X											
3	Test Lifecycle and Integration	NA										X														
3	Non-functional Testing	NA																	X							
3	Peer Reviews	NA																								
4	Test Measurement	NA		X																						
4	Product Quality Evaluation	NA		X																						
4	Advanced Reviews	NA																								
5	Defect Prevention	NA						X												X						
5	Quality Control	NA																								
5	Test Process Optimization	NA												X						X			X			

X Similarity between TMMi and TPI NEXT — Similarities in assessment results (dark grey) — Differences in assessment results (light grey)

Figure 10: Mapping between TPI NEXT and TMMi. Comparison of assessment results Part 2.

The matrices in Figures 9 and 10 illustrate the assessment results of both the TMMi and the TPI NEXT assessment in combination with the mapping results. The fulfillment degree of the process areas in TMMi and the key areas separated by maturity level in TPI NEXT respectively is indicated by three levels, 'FF' (fully fulfilled), 'PF' (partly fulfilled) and 'NF' (not fulfilled). To achieve a rating of 'fully fulfilled', in TMMi, all specific goals of the respective process area, and in TPI NEXT, all checkpoints of the respective key area, have to be fulfilled. TMMi process areas that have not been investigated in the case organization are marked with 'NA' (not applicable).

The staged model representation of TMMi demands the assessment to begin with the investigation of process areas belonging to Maturity Level 2. Only if all process areas of Level 2 are fulfilled the assessment proceeds with the investigation of process areas belonging to Maturity Level 3. Due to the low level of maturity present in the case organization the assessment was therefore limited to the process areas of Maturity Level 2 only.

In the contrary, the continuous approach of TPI NEXT allows for an assessment of all key areas. Therefore, some aspects of the case organization that have been investigated by TPI NEXT and assessed as partly fulfilled, have not been investigated by TMMi, namely some aspects of the TPI NEXT key areas 'Degree of involvement', 'Communication', 'Reporting' 'Test tools'.

The TMMi assessment resulted in all five process areas of Maturity Level 2 being assessed as 'partly fulfilled'. In general, the outcome of the TPI NEXT assessment shows a similar result. However, there are some key areas of TPI NEXT in which similarities with TMMi process areas of Level 2 had been identified by the mapping, but which have been assessed as 'not fulfilled' in the TPI NEXT assessment compared to the 'partly fulfilled' rating in TMMi. These are the efficient level of 'Stakeholder commitment', the optimizing level of 'Test strategy', the efficient level of 'Test organization', the efficient level of 'Reporting', the efficient level of 'Test process management', the efficient and optimizing level of 'Estimating and planning', the controlled level of 'Metrics', the efficient level of 'Test case design' and the efficient level of 'Test environment'.

The TPI NEXT assessment resulted in one key area being fully fulfilled, namely the controlled level of 'Defect management'. The mapping between TMMi and TPI NEXT had shown that the process area in TMMi dealing with similar aspects to this key area was 'Test monitoring and control'. Since the process area belongs to Maturity Level 2 it has also been investigated in the TMMi assessment but it was only assessed as partly fulfilled.

For some specific maturity levels of TPI NEXT, key areas that have been assessed as partly fulfilled for the case organization, the mapping between the two approaches had not identified similarities with TMMi process areas. These are the controlled level of 'Degree of involvement', the efficient level of 'Defect management', the efficient level of 'Testware management', the controlled and efficient level of 'Test tools', and the optimizing level of 'Test environment'.

7 Discussion

In this section the results presented in the previous section are discussed.

The systematic review and the mapping between TMMi and TPI NEXT performed within the case study provide a major general contribution to the body of knowledge with respect to TPI approaches.

Confirmed by researchers working in the area of Test Process Improvement, our systematic review provided a reasonably complete set of approaches. For some of the 18 identified approaches some of the researchers contacted as validation of the review results stated appreciative that they have not been aware of these approaches before. However, many of the research papers about these approaches do not provide sufficient information, or the approaches do not include assessment instruments which makes the approaches difficult to be applied in industry. Many of the identified approaches have even only been developed as concepts. Another limitation to the general applicability of the approaches is their specialization to a specific domain.

Primarily, based on the origin of the approach and the testing model which builds the framework for the assessment, we divided the approaches into four groups.

The first group consists of TMM and approaches that are based on TMM or that have been influenced in their development by TMM. Since TMM itself has been significantly influenced by CMM, one approach, TIM, has been included in this group that has not explicitly been influenced by TMM rather than by CMM. So it could also be considerable to characterize the approaches in this group as influenced by CMM.

In contrast, the formation of the second group, is clearly less ambiguous. It consists exclusively of TPI and TPI-based approaches. The third group represents standards and approaches related to these standards. Here, the classification to this group was again more ambiguous. One approach, the self-assessment framework for ISO/IEC 29119 based on TIM, has been included in this group since the testing model for this approach is provided by the standard ISO/IEC 29119. Viewed from another perspective, this approach could have been also included in the first group since the assessment process is based on TIM. However, the assessment process was not the *primary* criteria of our classification. Finally, the fourth group includes all other approaches that do not have a testing model. They present individual assessments which are not built on a predefined framework.

An alternative classification of the approaches could have been done by their model representation which would result in three groups, approaches without a model, approaches with a continuous model representation and approaches with a staged model representation. In this kind of classification the individual approaches would have been grouped as approaches without a model and the TPI approaches would have been allocated to the group of approaches with continuous model representation. The remaining approaches, however, would have been split between the two groups of continuous or staged model representations. Especially, in the TMM-related approaches, both, continuous and staged model representations are used. This, in turn, emphasizes the influence of CMM to these approaches, since CMM provides both, a continuous and a staged representation.

One further classification would have been conceivable; the distinction between qualitative and quantitative approaches. But surprisingly, only one approach was identified using quantitative data for the assessment. All the other assessments are done based on qualitative data gained from interviews or surveys. Even though the analysis of qualitative data provides much more space for individual interpretations and less defined structure, it appears to us that the analysis of qualitative is preferred over quantitative data analysis. This tendency to qualitative studies with respect to testing processes corresponds to statements given by interviewees during the interview phase of the assessment. It was claimed that the testing process followed is dependent, e.g., on the current situation, the workload or the tester's experience in an area. This individuality of the process makes an unambiguous interpretation of metrics more difficult and therefore the use of qualitative approaches *more* reasonable.

With respect to the selected TPI approaches to be applied in the case organization, it is clearly reflected, in our opinion, that trust in the provided methodologies plays an important role in industry. Only few of the approaches identified by the systematic review have been known to our industry partner before this study took place. The best known approaches in industry are TMMi and TPI/TPI NEXT, which have actually selected by the case organization. It could be argued that, these are the most commercially promoted ones, therefore the best known in industry. But on the other hand many of the approaches further developed in academia are also based on these approaches. Often a well known approach is further developed for a specific domain. In our opinion, this shows that industry and academia agree to a great extent about the requirements for TPI approaches.

Moreover, industry is to a great extent familiar with the well established concepts CMM/CMMi assessments and asks for similar certificates with respect to testing. A formal assessment performed by a lead assessor accredited by the TMMi Foundation provides this kind of certificate. Therefore, industry trusts in approaches influenced by CMMi. We suppose that the awareness of CMM/CMMi in the case organization and the influence of CMMi on TMMi influenced the voting of at least one participant of the static validation process.

From different points of view, it was also interesting that, firstly, only approaches using a testing reference model have been selected for application in the case organization and, secondly, approaches with different model representations.

Neglecting the individual approach without a reference model could be justified, in our opinion, by the better guidance for the assessment and trust in the best practices provided by approaches with testing reference models.

The selection of one approach with a continuous model representation and one with a staged representation is especially interesting with respect to the performed mapping between the two approaches and comparison of their results. The advantages and disadvantages of these two different representations are often discussed. It is claimed that the continuous approaches, like TPI NEXT, offer more room for improvements in practice. The ability to focus on individually chosen aspects of the test process provides the assessor the freedom to adapt the TPI to the specific needs of the organization; industry seems to deem that as a very valuable characteristic of a TPI approach.

In staged approaches, like TMMi, it seems to be very difficult to fulfill the requirements to achieve the next higher level. Since all aspects of a maturity level have to be fulfilled as a whole, a clear majority of the assessed organizations is still at low levels. An official survey performed by the TMMi Foundation on the organizations assessed by a formal TMMi assessment states that 11% of the assessed organizations are at initial level and 89% are at Level 2 [2].

Therefore, the low TMMi assessment result of the case organization in this study is not surprising. But, on the hand, it might have been expected that the TPI NEXT assessment would have led to a better result. However, due to the results of the mapping between TMMi and TPI NEXT, these similar assessment results are absolutely reasonable.

Despite their different model representations, the mapping between the approaches showed that they principally resemble to a great extent. Apart from smaller differences, they investigate the same aspects of the testing process and they basically categorize specific requirements to the process in the similar level's maturity. On this basis, it is very likely that they come to the same assessment result.

Nevertheless, the mapping and the detailed comparison of the assessment results, indicated that the requirements of the maturity levels in TMMi are much stricter and more difficult to reach than in TPI NEXT. The comparison results showed that some aspects of the testing process covered by lower maturity levels in TPI NEXT and identified as partly fulfilled in the case organization are allocated to much higher maturity levels in TMMi which have not even been investigated due to the non-fulfillment of Maturity Level 2. And furthermore, the mapping showed that some aspects allocated to Maturity Level 2 in TMMi are spread over all three maturity levels of TPI NEXT. Even an achievement of the highest maturity level in TPI NEXT, in regards to these aspects, would still not lead to an achievement of a higher maturity level in TMMi. Moreover, our experience in performing the assessments with both approaches showed that the definitions given for the checkpoints in TPI NEXT are more superficial and provide a lot of freedom for individual interpretations. Whereas, especially, the generic and specific practices, together with the work examples in TMMi give very detailed descriptions of the testing process, which provides a good guidance in conducting the assessment. However, for the successful application of both approaches, extended knowledge in software testing is essential.

[2] `http://www.tmmi.org/pdf/TMMISurvey2012.pdf`

8 Threats to validity

This section describes the threats to validity of this study and how they have been addressed. An analysis of validity threats enhances the correctness of a research study by early identifying factors possibly affecting the results and allowing actions to be taken to mitigate or minimize the threats to the finding. The structure of this section is given by the four categories of validity threats considered with respect to this study - threats to construct validity, internal validity, external validity and conclusion validity. The validity threats related to these four categories and possibly affecting this study are introduced with reference to [Wohlin et al., 2012]. For each kind of validity threat it is further discussed how this threat was particularly addressed in the design and performance of this study.

8.1 Construct validity

Construct validity considers how well the results to a concept or theory used behind the study can be generalized. It can be further categorized into design and social type. Design threats cover issues concerning design of the study while social threats are related to the behavior of the subject and the study.

Mono operational bias Mono operational bias refers to the use of only one single independent variable, case subject or treatment in a research study.

With respect to data collection, this threat was delimited by using methodological triangulation. Data collection was done through interviews, observations and document analysis.

Mono method bias Mono method bias means that single measures are used or single observations are done.

This threat was mitigated by the two researchers executing major parts of the research independently and cross checking the results. Furthermore, different methods were used by the two researchers with respect to snowball sampling and the mapping between the approaches. Conclusions were drawn by comparing the results.

Evaluation apprehension Evaluation apprehension is a social threat about a human's tendency to present herself in a better way when being evaluated.

To mitigate this, the interviewees were assured that the data collected in the interviews would be anonymous which helped them to provide honest and realistic information.

8.2 Internal validity

Internal validity refers to the act of establishing a casual relationship between the treatment and the outcome. These threats are further identified as single group threats because the study is applied to a single group.

Maturity Maturity considers the factors that can affect the reaction of the subject differently (negatively or positively) as time passes. Negative affect being that the subject gets tired or bored during the interview.

To mitigate this threat the duration of the interviews was planned not to exceed one hour.

Selection Selection is the natural variation in human performance and how their behavior is affecting the result.

There are two threats we identified in this regards, first, selection of interviewees who represent the whole population. This threat was minimized by asking the 'organization representatives' for help regarding interviewee selection, since they had very good insight in the organization. The second threat was identified while conducting the workshop for static validation where two external participants tried to put more emphasis on one TPI approach, namely TPI NEXT, due to their experience with this approach. This threat was later mitigated by selecting two approaches for application in the case organization.

Publication bias This threat to validity is related to systematic reviews where publication bias arises due to unpublished or grey literature which is not made available.

To minimize this threat we contacted the authors of the primary studies through email and asked for unpublished literature with respect to TPI approaches.

8.3 External validity

Threats to external validity are conditions that limit the ability to generalize the results of a study to industrial practice.

Interaction of selection and treatment This threat occurs if a wrong subject is selected from the study population and the result cannot be generalized.

This threat was minimized by selecting different interviewees from different areas (PU and PD), roles and locations (Gothenburg and Bangalore).

External validity can also be related to the the finding of a systematic review. We say that our exclusion and inclusion criteria and the quality assessments were precisely good to get a good representative set of primary studies.

8.4 Conclusion validity

Conclusion validity is concerned with issues affecting the ability to draw correct conclusion about relations between the treatment and the outcome of an experiment.

Reliability of the treatment implementation This threat refers to the implementation of a treatment to the subject. Treatment in our study would be the basis presented in applying the inclusion and exclusion criteria to select papers as primary studies in the systematic review.

To mitigate this threat we defined explicit inclusion and exclusion criteria and provided detailed documentation of the study selection process to be able to prove how the primary studies had been identified.

Random heterogeneity of subject Heterogeneity of subjects means that the subjects of a study belong to a varied group with respect specified characteristics. While homogeneity of subjects means that the subjects belong to the same group based on these characteristics. In our case, the interviewees belong to different areas within the organization, have different roles, and are located in different countries. Thus it can be concluded that we have a chance of facing random heterogeneity.

Threats to the conclusion validity regarding questionnaires used in the interviews were mitigated by conducting pilot interviews with two 'organization representatives'. Furthermore, the interview audio recordings were briefly transcribed and the transcipts were sent back to the respective interviewee to conform if the data had been interpreted correctly.

9 Conclusions

This study was divided into two parts. In the first part, we conducted a systematic literature review to identify available Test Process Improvement (TPI) approaches. A total of 18 approaches have been found. To our knowledge this is the first systematic literature review conducted in this area. Based on information extracted from the identified primary studies with respect to completeness of development, availability of information and assessment instruments, and domain limitations of the approaches, six generally applicable TPI approaches have been identified - TMM, TMMi, TPI, TPI NEXT and Observing practice. These six approaches mainly differ in regards to the use of testing process reference models, and their model representation.

In the second part of this study, we conducted a case study in which, first, two approaches to be applied in the case organization were selected, and second, two parallel assessments of the organization's testing process were performed using these approaches. The approaches used in this case study were TMMi and TPI NEXT. A major distinction between these two approaches is their model representation, TMMi has a staged model and TPI NEXT uses a continuous model. Based on an initially performed mapping between TMMi and TPI NEXT the assessment results were compared. With both approaches the testing process of the case organization was assessed to be in the 'initial' level. Based on the mapping between the approaches and the comparison of their detailed assessment results, we found out that both approaches are very similar. Mostly, they cover the same aspects of the testing process and categorize these aspects to similar levels of maturity. However, a closer look shows that the detailed assessment results differ, particularly caused by the different model representations of the two approaches.

The generalizable results of the systematic review and the mapping between the two TPI approaches provide, on the one hand, a good basis for further research in this area. There is a need to conduct further case studies comparing assessment results to strengthen the findings and to perform similar mappings between further approaches to extend the knowledge. On the other hand, these results essentially support industry in selecting an approach to improve software testing processes.

References

TMMi Foundation website. URL `www.tmmifoundation.org/`.

TPI Automotive, Version 1.01, 2004. URL `http://www.tpiautomotive.de/Docs/TPI%20automotive%20version%201.01.pdf`.

Software and systems engineering - Software testing - Part 2: Test process, 2010.

J.M. Ares Casal, O. Dieste Tubio, R. Garcia Vazquez, M. Lopez Fernandez, and S. Rodriguez Yanez. Formalising the software evaluation process. In *Computer Science, 1998. SCCC'98. XVIII International Conference of the Chilean Society of*, pages 15–24. IEEE, 1998.

Antonia Bertolino. Software testing research: Achievements, challenges, dreams. In *Future of Software Engineering, 2007. FOSE'07*, pages 85–103. IEEE, 2007.

P.M.S. Bueno, A.N. Crespo, and M. Jino. Analysis of an artifact oriented test process model and of testing aspects of CMMI. In *Product-Focused Software Process Improvement*, volume 4034 of *Lecture Notes in Computer Science*, pages 263–277. Springer Berlin Heidelberg, 2006. ISBN 978-3-540-34682-1. doi: 10.1007/11767718_23. URL `http://dx.doi.org/10.1007/11767718_23`.

I. Burnstein. *Practical Software Testing: A Process-oriented Approach*. Springer Inc., New York, NY, USA, 2003.

I. Burnstein, T. Suwanassart, and R. Carlson. Developing a testing maturity model for software test process evaluation and improvement. In *Test Conference, 1996. Proceedings., International*, pages 581–589, 1996. doi: 10.1109/TEST.1996.557106.

I. Burnstein, T. Suwannasart A. Homymen, G. Saxena, and R. Grom. A testing maturity model for software test process assessment and improvement. 1999.

J.K. Chaar, M.J. Halliday, I.S. Bhandari, and R. Chillarege. In-process evaluation for software inspection and test. *Software Engineering, IEEE Transactions on*, 19(11): 1055–1070, 1993. ISSN 0098-5589. doi: 10.1109/32.256853.

Jacob Cohen et al. A coefficient of agreement for nominal scales. *Educational and psychological measurement*, 20(1):37–46, 1960.

J.S. Collofello, Z. Yang, J.D. Tvedt, D. Merrill, and I. Rus. Modeling software testing processes. In *Computers and Communications, 1996., Conference Proceedings of the 1996 IEEE Fifteenth Annual International Phoenix Conference on*, pages 289–293. IEEE, 1996.

P. Cruz, R. Villarroel, F. Mancilla, and M. Visconti. A software testing process for the reference model of Competisoft. In *Chilean Computer Science Society (SCCC), 2010 XXIX International Conference of the*, pages 51–59, 2010. doi: 10.1109/SCCC.2010.39.

K.C. Dangle, P. Larsen, M. Shaw, and M.V. Zelkowitz. Software process improvement in small organizations: a case study. *Software, IEEE*, 22(6):68–75, 2005. ISSN 0740-7459. doi: 10.1109/MS.2005.162.

T. Ericson, A. Subotic, and S. Ursing. TIM - A test improvement model. *Software Testing, Verification and Reliability*, 7(4):229–246, 1997. ISSN 1099-1689. doi: 10.1002/(SICI)1099-1689(199712)7:4⟨229::AID-STVR149⟩3.0.CO;2-M. URL `http://dx.doi.org/10.1002/(SICI)1099-1689(199712)7:4<229::AID-STVR149>3.0.CO;2-M`.

A. Farooq and R.R. Dumke. Evaluation approaches in software testing.

A. Farooq and R.R. Dumke. Research directions in verification & validation process improvement. *SIGSOFT Softw. Eng. Notes*, 32(4), July 2007. ISSN 0163-5948. doi: 10.1145/1281421.1281425. URL `http://doi.acm.org.proxy.lib.chalmers.se/10.1145/1281421.1281425`.

A. Farooq and R.R. Dumke. Developing and applying a consolidated evaluation framework to analyze test process improvement approaches. In *Software Process and Product Measurement*, volume 4895 of *Lecture Notes in Computer Science*, pages 114–128. Springer Berlin Heidelberg, 2008. ISBN 978-3-540-85552-1. doi: 10.1007/978-3-540-85553-8_10. URL `http://dx.doi.org/10.1007/978-3-540-85553-8_10`.

A. Farooq, H. Hegewald, and R.R. Dumke. A critical analysis of the testing maturity model. *Metrics News, Journal of GI-Interest Group on Software Metrics, 12(1):35-40*, 2007.

A. Farooq, R.R. Dumke, A. Schmietendorf, and A. Hegewald. A classification scheme for test process metrics. *In SEETEST 2008: South East European Software Testing Conference, Heidelberg, Germany*, July 2008a.

A. Farooq, K. Georgieva, and R.R. Dumke. A meta-measurement approach for software test processes. In *Multitopic Conference, 2008. INMIC 2008. IEEE International*, pages 333–338, 2008b. doi: 10.1109/INMIC.2008.4777759.

A. Farooq, K. Georgieva, and R.R. Dumke. Challenges in evaluating SOA test processes. In *Software Process and Product Measurement*, volume 5338 of *Lecture Notes in Computer Science*, pages 107–113. Springer Berlin Heidelberg, 2008c. ISBN 978-3-540-89402-5. doi: 10.1007/978-3-540-89403-2_10. URL `http://dx.doi.org/10.1007/978-3-540-89403-2_10`.

B. Fitzgerald and T. O'Kane. A longitudinal study of software process improvement. *Software, IEEE*, 16(3):37–45, 1999. ISSN 0740-7459. doi: 10.1109/52.765785.

D. Gelperin and B. Hetzel. The growth of software testing. *Commun. ACM*, 31(6):687–695, June 1988. ISSN 0001-0782. doi: 10.1145/62959.62965. URL `http://doi.acm.org.proxy.lib.chalmers.se/10.1145/62959.62965`.

L. A. Goodman. Snowball sampling. *The Annals of Mathematical Statistics*, 32(1), 1961.

T. Gorschek, P. Garre, S. Larsson, and C. Wohlin. A model for technology transfer in practice. *IEEE Software*, 23(6):88–95, 2006.

M. Ham. Testing in software process improvement. *MB-TMM project report*, pages 12–5, 2002.

M. Ham and E. v. Veenendaal. State of the art literature. part 1, mb-tmm project report 12-1-1-fp-im. 2001.

M. J. Harrold. Testing: a roadmap. In *Proceedings of the Conference on The Future of Software Engineering*, ICSE '00, pages 61–72, New York, NY, USA, 2000. ACM. ISBN 1-58113-253-0. doi: 10.1145/336512.336532. URL `http://doi.acm.org/10.1145/336512.336532`.

Y. He, H. Hecht, and R.A. Paul. Measuring and assessing software test processes using test data. In *High Assurance Systems Engineering, 2000, Fifth IEEE International Symposim on. HASE 2000*, pages 259–264, 2000. doi: 10.1109/HASE.2000.895470.

J.E. Heiser. An overview of software testing. In *AUTOTESTCON, 97. 1997 IEEE Autotestcon Proceedings*, pages 204–211, 1997. doi: 10.1109/AUTEST.1997.633613.

H. Heiskanen, M. Maunumaa, and M. Katara. A test process improvement model for automated test generation. In *Product-Focused Software Process Improvement*, volume 7343 of *Lecture Notes in Computer Science*, pages 17–31. Springer Berlin Heidelberg, 2012. ISBN 978-3-642-31062-1. doi: 10.1007/978-3-642-31063-8_3. URL `http://dx.doi.org/10.1007/978-3-642-31063-8_3`.

A. Homyen. An assessment model to determine test process maturity, Ph.D. thesis. 1998.

F. Ilyas and R. Malik. Adhering to cmm level 2 in medium size software organizations in pakistan. In *Multi Topic Conference, 2003. INMIC 2003. 7th International*, pages 434–439, 2003. doi: 10.1109/INMIC.2003.1416766.

J. Jacobs, J. v. Moll, and T. Stokes. The process of test process improvement. *XOOTIC Magazine*, 8(2):23–29, 2000.

J.C. Jacobs and J.J.M. Trienekens. Towards a metrics based verification and validation maturity model. In *Software Technology and Engineering Practice, 2002. STEP 2002. Proceedings. 10th International Workshop on*, pages 123–128, 2002. doi: 10.1109/STEP.2002.1267622.

E. Jung. A test process improvement model for embedded software developments. In *Quality Software, 2009. QSIC '09. 9th International Conference on*, pages 432–437, 2009. doi: 10.1109/QSIC.2009.64.

D. Karlström, P. Runeson, and S. Nordén. A minimal test practice framework for emerging software organizations. *Software Testing, Verification and Reliability*, 15 (3):145–166, 2005. ISSN 1099-1689. doi: 10.1002/stvr.317. URL `http://dx.doi.org/10.1002/stvr.317`.

A. Kasoju, K. Petersen, and M.V. Mäntylä. Analyzing an automotive testing process with evidence-based software engineering. *Information and Software Technology*, 55(7):1237 – 1259, 2013. ISSN 0950-5849. doi: http://dx.doi.org/10.1016/j.infsof.2013.01.005. URL `http://www.sciencedirect.com/science/article/pii/S0950584913000165`.

J. Kasurinen. Elaborating software test processes and strategies. In *Software Testing, Verification and Validation (ICST), 2010 Third International Conference on*, pages 355–358, 2010. doi: 10.1109/ICST.2010.25.

J. Kasurinen, P. Runeson, L. Riungu, and K. Smolander. A self-assessment framework for finding improvement objectives with ISO/IEC 29119 test standard. In *Systems, Software and Service Process Improvement*, volume 172 of *Communications in Computer and Information Science*, pages 25–36. Springer Berlin Heidelberg, 2011a. ISBN 978-3-642-22205-4. doi: 10.1007/978-3-642-22206-1_3. URL `http://dx.doi.org/10.1007/978-3-642-22206-1_3`.

J. Kasurinen, O. Taipale, and K. Smolander. How test organizations adopt new testing practices and methods? In *Software Testing, Verification and Validation Workshops (ICSTW), 2011 IEEE Fourth International Conference on*, pages 553–558, 2011b. doi: 10.1109/ICSTW.2011.63.

J. Kiiskila. Practical aspects on the assessment of a review process. In *Euromicro Conference, 1998. Proceedings. 24th*, volume 2, pages 867–870 vol.2, 1998. doi: 10.1109/EURMIC.1998.708114.

N. Kikuchi and T. Kikuno. Improving the testing process by program static analysis. In *Software Engineering Conference, 2001. APSEC 2001. Eighth Asia-Pacific*, pages 195–201, 2001. doi: 10.1109/APSEC.2001.991477.

E. Kim and Y. Jang. A test improvement model for embedded software testing. In *Proceedings of the 11th IASTED International Conference on Software Engineering and Applications*, pages 79–84. ACTA Press, 2007.

E. Kit and S. Finzi. *Software testing in the real world: Improving the process*. ACM Press/Addison-Wesley Publishing Co., 1995.

B.A. Kitchenham and S. Charters. Guidelines for performing systematic literature reviews in software engineering. 2007.

T. Koomen. Worldwide survey on test process improvement. technical report. 2002.

T. Koomen. Worldwide survey on test process improvement. technical report. 2004.

T Koomen. Stepwise improvement of the test process using TPI. In *ICS Test Conference*, 2005.

T. Koomen and M. Pol. *Test process improvement: A practical step-by-step guide to structured testing*. Addison-Wesley, 1999.

T. Koomen, L. v.d. Aalst, B. Broekman, and M. Vroon. *TMap Next, for result-driven testing*. UTN Publishers, 2006.

S. Kulkarni. Test process maturity models–yesterday, today and tomorrow. In *Proceedings of the 6th Annual International Software Testing Conference, Delhi, India*, 2006.

T.O.A. Lehtinen, M.V. Mäntylä, and J. Vanhanen. Development and evaluation of a lightweight root cause analysis method (arca method)–field studies at four software companies. *Information and Software Technology*, 53(10):1045–1061, 2011.

D.J. Mala, V. Mohan, and M. Kamalapriya. Automated software test optimisation framework - an artificial bee colony optimisation-based approach. *Software, IET*, 4 (5):334–348, 2010. ISSN 1751-8806. doi: 10.1049/iet-sen.2009.0079.

C.-X. Meng. A goal-driven measurement model for software testing process. In *Information Technology and Applications, 2009. IFITA '09. International Forum on*, volume 1, pages 656–659, 2009. doi: 10.1109/IFITA.2009.565.

M. Miller, A. Goslin, and ICS Test-UK. Tmm-a case study. In *Software Quality Conferences incorporating ICSTEST-UK, London, UK*, 2005.

H. Oh, B. Choi, H. Han, and W.E. Wong. Optimizing test process action plans by blending testing maturity model and design of experiments. In *Quality Software, 2008. QSIC '08. The Eighth International Conference on*, pages 57–66, 2008. doi: 10.1109/QSIC.2008.19.

F. Pettersson, M. Ivarsson, T. Gorschek, and P. Öhman. A practitioners guide to light weight software process assessment and improvement planning. *Journal of Systems and Software*, 81(6):972–995, 2008.

F.J. Pino, F. Garca, and M. Piattini. Software process improvement in small and medium software enterprises: a systematic review. *Software Quality Journal*, 16 (2):237–261, 2008. ISSN 0963-9314. doi: 10.1007/s11219-007-9038-z. URL `http://dx.doi.org/10.1007/s11219-007-9038-z`.

M. Pol. *Software testing: A guide to the TMap approach*. Addison-Wesley, 2002.

K.K.R. Rana and S.S.U. Ahmad. Bringing maturity to test. *Information Professional*, 2(2):30–33, 2005. ISSN 1743-694X.

M. Rasking. Experiences developing TMMi® as a public model. In *Software Process Improvement and Capability Determination*, volume 155 of *Communications in Computer and Information Science*, pages 190–193. Springer Berlin Heidelberg, 2011. ISBN 978-3-642-21232-1. doi: 10.1007/978-3-642-21233-8_18. URL `http://dx.doi.org/10.1007/978-3-642-21233-8_18`.

S. Reid. The new software testing standard. In *Achieving Systems Safety*, pages 237–255. Springer London, 2012. ISBN 978-1-4471-2493-1. doi: 10.1007/978-1-4471-2494-8_17. URL http://dx.doi.org/10.1007/978-1-4471-2494-8_17.

K. Rinkevics and R. Torkar. Equality in cumulative voting: A systematic review with an improvement proposal. *Information & Software Technology*, 55(2):267–287, 2013.

P. Runeson and M. Höst. Guidelines for conducting and reporting case study research in software engineering. *Empirical Software Engineering*, 14(2):131–164, 2009.

H. Ryu, D.-K. Ryu, and J. Baik. A strategic test process improvement approach using an ontological description for MND-TMM. In *Computer and Information Science, 2008. ICIS 08. Seventh IEEE/ACIS International Conference on*, pages 561–566, 2008. doi: 10.1109/ICIS.2008.78.

J. Saldaña Ramos, A. Sanz-Esteban, J. García-Guzmán, and A. Amescua. Design of a competence model for testing teams. *Software, IET*, 6(5):405–415, 2012. ISSN 1751-8806. doi: 10.1049/iet-sen.2011.0182.

A. Sanz, J. Saldaña, J. García, and D. Gaitero. Testpai: A proposal for a testing process area integrated with CMMI. *European Systems and Software Process Improvement and Innovation (EuroSPI 2008), Dublin, Ireland*, 2008.

M. Steiner, M. Blaschke, M. Philipp, and T. Schweigert. Make test process assessment similar to software process assessment: The Test SPICE approach. *Journal of Software: Evolution and Process*, 24(5):471–480, 2012.

M. Sulayman and E. Mendes. An extended systematic review of software process improvement in small and medium web companies. In *Evaluation Assessment in Software Engineering (EASE 2011), 15th Annual Conference on*, pages 134–143, 2011. doi: 10.1049/ic.2011.0017.

T. Suwannasart. Towards the development of a testing maturity model, unpublished Ph.D.thesis. March 1996.

R. Swinkels. A comparison of TMM and other test process improvement models. *MB-TMM Project Report*, pages 12–4, 2000.

O. Taipale and K. Smolander. Improving software testing by observing practice. In *Proceedings of the 2006 ACM/IEEE international symposium on Empirical software engineering*, ISESE '06, pages 262–271, New York, NY, USA, 2006. ACM. ISBN 1-59593-218-6. doi: 10.1145/1159733.1159773. URL http://doi.acm.org.proxy.lib.chalmers.se/10.1145/1159733.1159773.

O. Taipale, K. , and H. Kälviäinen. Cost reduction and quality improvement in software testing. *In Software Quality Management Conference*, 2006.

T. Tayamanon, T. Suwannasart, N. Wongchingchai, and A. Methawachananont. TMM appraisal assistant tool. In *Systems Engineering (ICSEng), 2011 21st International Conference on*, pages 329–333, 2011. doi: 10.1109/ICSEng.2011.65.

TMMi Foundation. Test maturity model integration(TMMi), version 1, 2010. URL `http://www.tmmi.org/pdf/TMMi.Framework.pdf`.

M. Unterkalmsteiner, T. Gorschek, A.K.M.M. Islam, K.C. Chow, R.B. Permadi, and R. Feldt. Evaluation and measurement of software process improvement x02014;a systematic literature review. *Software Engineering, IEEE Transactions on*, 38(2): 398–424, 2012. ISSN 0098-5589. doi: 10.1109/TSE.2011.26.

A. v. Ewijk, B. Linker, M. van Oosterwijk, and B. Visser. *TPI® NEXT-Business Driven Test Process Improvement*. SogetiBooks, 2013.

E. van Veenendal. Test maturity model integration(TMMi), version 1.1. 2008.

T.K. Varkoi and T.K. Makinen. Case study of cmm and spice comparison in software process assessment. In *Engineering and Technology Management, 1998. Pioneering New Technologies: Management Issues and Challenges in the Third Millennium. IEMC '98 Proceedings. International Conference on*, pages 477–482, 1998. doi: 10.1109/IEMC.1998.727808.

Y. Wang, A. Dorling, H. Wickberg, and G. King. Experience in comparative process assessment with multi-process-models. In *EUROMICRO Conference, 1999. Proceedings. 25th*, volume 2, pages 268–273 vol.2, 1999. doi: 10.1109/EURMIC.1999.794790.

C. Wohlin, P. Runeson, M. Höst, M.C. Ohlsson, B. Regnell, and A. Wesson. *Experimentation in software engineering*. Springer Publishing Company, Incorporated, 2012.

L. Xin-ke and Y. Xiao-Hui. A goal-driven measurement model for software testing process. In *Software Engineering, 2009. WCSE '09. WRI World Congress on*, volume 4, pages 8–12, 2009. doi: 10.1109/WCSE.2009.27.

L. Xu-Xiang and Z. Wen-Ning. The PDCA-based software testing improvement framework. In *Apperceiving Computing and Intelligence Analysis (ICACIA), 2010 International Conference on*, pages 490–494, 2010. doi: 10.1109/ICACIA.2010.5709948.

Zil-e-Huma, M. Bano, and N. Ikram. Software process improvement: A systematic literature review. In *Multitopic Conference (INMIC), 2012 15th International*, pages 459–464, 2012. doi: 10.1109/INMIC.2012.6511481.

A Summarized inquiries and responses from contacting authors in systematic literature review

The authors of 21 papers were contacted. The email inquiries were structured as follows:

- Introduction as master students of Software Engineering at Gothenburg University.
- Introduction to the thesis topic.
- Stating that paper written by the contact person has been identified as primary study to our thesis topic.
- Asking for further studies conducted in the presented research area.
- If applicable, referring to the development status mentioned in the paper.

We got the following responses:

- One of the authors of [Farooq et al., 2008b], [Farooq et al., 2008c] and [Farooq and Dumke, 2008] provided us with the technical report [Farooq and Dumke].
- One of the authors of [Kasoju et al., 2013] suggested to look into general Software Process Improvement frameworks, lean software development methods and light-weight process improvement approaches related to testing, like presented in [Pettersson et al., 2008] and [Lehtinen et al., 2011].
- One of the authors of [Taipale and Smolander, 2006] provided us with access to data and publications of the project MASTO.
- The author of [Rasking, 2011] offered to conduct an interview.
- The authors of [Burnstein, 2003], [Taipale and Smolander, 2006], [Kasurinen et al., 2011a], [Ryu et al., 2008], [Heiskanen et al., 2012] kindly responded to our email but could not provide us with further research papers.

B Pilot search - Search queries

Table 11: Pilot Search ACM

Search Query	Publication Type	Hits	Relevant
((((((Abstract:Test AND Abstract:Process AND Abstract:Improvement AND Abstract:Software))OR((Title:Test AND Title:Process AND Title:Improvement AND Title:Software))))) AND (PublishedAs:proceeding))	Proceeding	52	0
((((((Abstract:Test AND Abstract:Process AND Abstract:Improvement AND Abstract:Software))OR((Title:Test AND Title:Process AND Title:Improvement AND Title:Software))))) AND (PublishedAs:journal))	Journal	0	0

Table 12: Pilot Search IEEE

Search Query	Content Type	Topic	Hits	Relevant
((((Test) AND Process) AND Improvement) AND Software)	Journals and Magazines	Computing and Processing (Hardware/Software)	285	3

Table 13: Pilot Search ScienceDirect

Search Query	Include	Subject	Hits	Relevant
TITLE-ABSTR-KEY(Test Process Improvement Software)	Journals	Computer Science	39	1

Table 14: Pilot Search Springer Link

Search Query	Content Type	Discipline	Sub-discipline	Hits	Relevant
Test AND process AND improvement AND software	Article	Computer Science	SWE	4767	not checked
Test AND process AND improvement AND software	Chapter	Computer Science	SWE	20613	not checked

C Table of all references found in systematic literature review

Table 15: Systematic literature review - All references

		Title and abstract exclusion			Introduction and conclusion exclusion									
Database	Title	Researcher A	Researcher B	Excluded	Researcher A	Researcher B	Excluded	Included	Discussion needed	Excluded in discussion	Excluded by quality criteria	Excluded in Phase 2	Excluded by full-text	Primary study
Springer Link	The New Software Testing Standard	maybe	maybe		yes	yes		X						X
Springer Link	Experiences Developing TMMi as a Public Model	yes	yes		yes	yes		X						X
Springer Link	Developing and Applying a Consolidated Evaluation Framework to Analyze Test Process Improvement Approaches	yes	yes		yes	yes		X						X
Springer Link	A Test Process Improvement Model for Automated Test Generation	yes	maybe		yes	yes		X						X
Springer Link	A Self-assessment Framework for Finding Improvement Objectives with ISO/IEC 29119 Test Standard	yes	yes		yes	yes		X						X
ScienceDirect	Analyzing an Automotive Testing Process with Evidence-Based Software Engineering	maybe	yes		yes	yes		X						X
IEEE, Springer Link	Towards a metrics based verification and validation maturity model	yes	yes		yes	yes		X						X
IEEE	TMM Appraisal Assistant Tool	yes	yes		yes	yes		X						X
IEEE	The PDCA-based software testing improvement framework	no	yes		no	yes			X					X

Continued on next page

Table 15 – continued from previous page

Database	Title	Title and abstract exclusion: Researcher A	Researcher B	Excluded	Introduction and conclusion exclusion: Researcher A	Researcher B	Excluded	Included	Discussion needed	Excluded in discussion	Excluded by quality criteria	Excluded in Phase 2	Excluded by full-text	Primary study
IEEE	Developing a Testing Maturity Model for software test process evaluation and improvement	yes	yes		yes	yes		X						X
IEEE	Bringing maturity to test	yes	yes		yes	yes		X						X
IEEE	A Test Process Improvement Model for Embedded Software Developments	yes	maybe		no	yes			X					X
IEEE	A meta-measurement approach for software test processes	yes	yes		no	yes			X					X
ACM	Improving software testing by observing practice	yes	yes		yes	yes		X						X
ACM, IEEE	A Strategic Test Process Improvement Approach Using an Ontological Description for MND-TMM	yes	yes		no	yes			X					X
IEEE	Design of a competence model for testing teams	no	maybe		no	yes			X				X	
Springer Link	Challenges in Evaluating SOA Test Processes	maybe	no		no	yes			X				X	
Springer Link	Analysis of an Artifact Oriented Test Process Model and of Testing Aspects of CMMI	yes	no		yes	yes		X					X	
IEEE	Measuring and assessing software test processes using test data	maybe	yes		yes	yes		X					X	

Continued on next page

Table 15 – continued from previous page

Database	Title	Title and abstract exclusion: Researcher A	Researcher B	Excluded	Introduction and conclusion exclusion: Researcher A	Researcher B	Excluded	Included	Discussion needed	Excluded in discussion	Excluded by quality criteria	Excluded in Phase 2	Excluded by full-text	Primary study
IEEE	Improving the testing process by program static analysis	yes	yes		yes	maybe			X				X	
IEEE	Elaborating Software Test Processes and Strategies	yes	yes		yes	yes		X					X	
IEEE	A Software Testing Process for the Reference Model of Competisoft	maybe	yes		no	yes			X				X	
IEEE	Optimizing Test Process Action Plans by Blending Testing Maturity Model and Design of Experiments	yes	yes		yes	yes		X				X		
IEEE	How Test Organizations Adopt New Testing Practices and Methods?	no	maybe		yes	yes		X				X		
IEEE	An overview of software testing	yes	maybe		no	yes			X			X		
IEEE	A Goal-Driven Measurement Model for Software Testing Process	maybe	yes		no	yes			X		X			
IEEE	A Goal-Driven Measurement Model for Software Testing Process	yes	yes		no	yes			X		X			
Springer Link	Towards Software Testing Process Improvement from Requirements	no	yes		yes	no			X	X				
Springer Link	Test process assessments and improvement	yes	yes		maybe	maybe			X	X				

Continued on next page

Table 15 – continued from previous page

Database	Title	Title and abstract exclusion: Researcher A	Researcher B	Excluded	Introduction and conclusion exclusion: Researcher A	Researcher B	Excluded	Included	Discussion needed	Excluded in discussion	Excluded by quality criteria	Excluded in Phase 2	Excluded by full-text	Primary study
Springer Link	Software Reuse and the Test Development Process: A Combined Approach	maybe	no		yes	no			X	X				
Springer Link	Past, Present and Future of Process Improvement in Ireland An Industry View	yes	yes		no	yes			X	X				
Springer Link	Experiences from Informal Test Process Assessments in Ireland Top 10 Findings	maybe	maybe		no	yes			X	X				
IEEE	The Definition of a Testing Process to Small-Sized Companies: The Brazilian Scenario	no	maybe		maybe	maybe			X	X				
IEEE	The application and research of spaceflight tracking and controlling software testing process new model	no	yes		maybe	maybe			X	X				
IEEE	Reflective Architecture Based Software Testing Management Model	maybe	maybe		yes	maybe			X	X				
IEEE	Modeling software testing processes	no	maybe		yes	no			X	X				
IEEE	A Embedded Software Testing Process Model	maybe	no		no	maybe			X	X				
ACM	A survey on software test maturity in Korean defense industry	no	yes		yes	maybe			X	X				

Continued on next page

Table 15 – continued from previous page

		Title and abstract exclusion			Introduction and conclusion exclusion									
Database	Title	Researcher A	Researcher B	Excluded	Researcher A	Researcher B	Excluded	Included	Discussion needed	Excluded in discussion	Excluded by quality criteria	Excluded in Phase 2	Excluded by full-text	Primary study
Springer Link	Towards Incorporation of Software Security Testing Framework in Software Development	maybe	no		no	no	X							
Springer Link	Towards a Capability Model for the Software Release Planning Process Based on a Multiple Industrial Case Study	maybe	no		no	no	X							
Springer Link	The Role of Different Approaches in Inspection Process Improvement	maybe	no		no	no	X							
Springer Link	The Design of Focus Area Maturity Models	no	maybe		no	no	X							
Springer Link	The Comparative Study for ENHPP Software Reliability Growth Model Based on Mixture Coverage Function	maybe	no		no	no	X							
Springer Link	Summaries of PIE Reports	no	maybe		no	no	X							
Springer Link	Summaries of PIE Reports	no	maybe		no	no	X							
Springer Link	Software Test Capability Improvement Method	yes	yes		no	no	X							
Springer Link	Software Reliability Modeling	maybe	no		no	no	X							

Continued on next page

Table 15 – continued from previous page

Database	Title	Title and abstract exclusion: Researcher A	Title and abstract exclusion: Researcher B	Title and abstract exclusion: Excluded	Introduction and conclusion exclusion: Researcher A	Introduction and conclusion exclusion: Researcher B	Introduction and conclusion exclusion: Excluded	Included	Discussion needed	Excluded in discussion	Excluded by quality criteria	Excluded in Phase 2	Excluded by full-text	Primary study
Springer Link	Software Analysis and Model Checking	maybe	no		no	no	X							
Springer Link	Perspectives	no	maybe		no	no	X							
Springer Link	Investigating the effect of variations in the test development process: a case from a safety-critical system	maybe	no		no	no	X							
Springer Link	Integrating Joint Reviews with Automotive SPICE Assessments Results	maybe	no		no	no	X							
Springer Link	Impacts of the Organizational Model on Testing: Three Industrial Cases	maybe	no		no	no	X							
Springer Link	Hybrid Modeling of Test-and-Fix Processes in Incremental Development	maybe	no		no	no	X							
Springer Link	Experience Reports	no	maybe		no	no	X							
Springer Link	Bridge the Gap between Software Test Process and Business Value: A Case Study	yes	maybe		no	no	X							
IEEE	VOCAL: a framework for test identification and deployment	yes	maybe		no	no	X							

Continued on next page

Table 15 – continued from previous page

Database	Title	Title and abstract exclusion: Researcher A	Researcher B	Excluded	Introduction and conclusion exclusion: Researcher A	Researcher B	Excluded	Included	Discussion needed	Excluded in discussion	Excluded by quality criteria	Excluded in Phase 2	Excluded by full-text	Primary study
IEEE	Validating and improving test-case effectiveness	maybe	no		no	no	X							
IEEE	Towards Automation in Software Test Life Cycle Based on Multi-Agent	maybe	no		no	no	X							
IEEE	Testing IT: an off-the-shelf software testing process	maybe	no		no	no	X							
IEEE	Research on the measurement framework ISTPM based on MPM	yes	yes		no	no	X							
IEEE	Research and Implementation of Knowledge Management Methods in Software Testing Process	maybe	no		no	no	X							
IEEE	Research and Establishment of Quality Cost Oriented Software Testing Model	no	maybe		no	no	X							
IEEE	Outlining Developers' Testing Process Model	yes	yes		no	no	X							
IEEE	Modeling Software Testing Process Using HTCPN	yes	no		no	no	X							
IEEE	Investigation of Knowledge Management Methods in Software Testing Process	no	maybe		no	no	X							

Continued on next page

Table 15 – continued from previous page

Database	Title	Title and abstract exclusion: Researcher A	Researcher B	Excluded	Introduction and conclusion exclusion: Researcher A	Researcher B	Excluded	Included	Discussion needed	Excluded in discussion	Excluded by quality criteria	Excluded in Phase 2	Excluded by full-text	Primary study
IEEE	Improvement of software testing process of enterprise project	yes	yes		no	no	X							
IEEE	An optimal release policy for software testing process	maybe	no		no	no	X							
IEEE	A Markov Decision Approach to Optimize Testing Profile in Software Testing	yes	yes		no	no	X							
IEEE	A lean metric acquisition and presentation environment for the assessment of a test process improvement experiment	no	maybe		no	no	X							
IEEE	A comparison of software-testing methodologies	maybe	no		no	no	X							
ACM	Monitoring the software test process using statistical process control: a logarithmic approach	no	maybe		no	no	X							
ACM, IEEE	Modeling and controlling the software test process	maybe	maybe		no	no	X							
ACM	Experiences of applying SPC techniques to software development processes	no	maybe		no	no	X							
ACM	Effective test metrics for test strategy evolution	no	maybe		no	no	X							

Continued on next page

Table 15 – continued from previous page

Database	Title	Title and abstract exclusion: Researcher A	Researcher B	Excluded	Introduction and conclusion exclusion: Researcher A	Researcher B	Excluded	Included	Discussion needed	Excluded in discussion	Excluded by quality criteria	Excluded in Phase 2	Excluded by full-text	Primary study
ACM	Chasing rainbows: improving software testing in the real world	no	yes		no	no	X							
ACM	A learning strategy for software testing optimization based on dynamic programming	no	maybe		no	no	X							
Springer Link	Who tested my software? Testing as an organizationally cross-cutting activity	no	no	X										
Springer Link	What Is a Test Case? Revisiting the Software Test Case Concept	no	no	X										
Springer Link	Verification & Validation in Medi SPICE	no	no	X										
Springer Link	Web Engineering: Beyond CS, IS and SE Evolutionary and Non-Engineering Perspectives	no	no	X										
Springer Link	Web Application Testing	no	no	X										
Springer Link	Using Software Inspection as a Catalyst for SPI in a Small Company	no	no	X										
Springer Link	Using Process Simulation to Assess the Test Design Effort Reduction of a Model-Based Testing Approach	no	no	X										

Continued on next page

Table 15 – continued from previous page

Database	Title	Title and abstract exclusion			Introduction and conclusion exclusion									
		Researcher A	Researcher B	Excluded	Researcher A	Researcher B	Excluded	Included	Discussion needed	Excluded in discussion	Excluded by quality criteria	Excluded in Phase 2	Excluded by full-text	Primary study
Springer Link	Using formal methods for ensuring quality requirements of systems	no	no	X										
Springer Link	Use of Non-IT Testers in Software Development	no	no	X										
Springer Link	Usability Engineering Methods Plethora	no	no	X										
Springer Link	Theory and practice of software factories	no	no	X										
Springer Link	Theoretical Feasibility of Conditional Invariant Detection	no	no	X										
Springer Link	The Ten Best Practices for Test Case Prioritization	no	no	X										
Springer Link	The Security Process Model of Embedded Systems	no	no	X										
Springer Link	The Role of Design Components in Test Plan Generation	no	no	X										
Springer Link	The process modeling cookbook orientation, description and experience	no	no	X										
Springer Link	The OOSPICE Assessment Component: Customizing Software Process Assessment to CBD	no	no	X										

Continued on next page

Table 15 – continued from previous page

		Title and abstract exclusion			Introduction and conclusion exclusion									
Database	Title	Researcher A	Researcher B	Excluded	Researcher A	Researcher B	Excluded	Included	Discussion needed	Excluded in discussion	Excluded by quality criteria	Excluded in Phase 2	Excluded by full-text	Primary study
Springer Link	The marginal value of increased testing: An empirical analysis using four code coverage measures	no	no	X										
Springer Link	The Dynamic Architecture Maturity Matrix: Instrument Analysis and Refinement	no	no	X										
Springer Link	The Control of IT Risk by Means of Risk-Based Testing	no	no	X										
Springer Link	The ability of directed tests to predict software quality	no	no	X										
Springer Link	Testing Safety Critical Ada Code Using Non Real Time Testing	no	no	X										
Springer Link	Testing of Service-Oriented Architectures A Practical Approach	no	no	X										
Springer Link	Testing in Multi-Agent Systems	no	no	X										
Springer Link	Testing Coverage and Removal Models	no	no	X										
Springer Link	Testing Concurrent Systems: A Formal Approach	no	no	X										
Springer Link	Test Management Traceability Model to Support Software Testing Documentation	no	no	X										

Continued on next page

Table 15 – continued from previous page

		Title and abstract exclusion			Introduction and conclusion exclusion									
Database	Title	Researcher A	Researcher B	Excluded	Researcher A	Researcher B	Excluded	Included	Discussion needed	Excluded in discussion	Excluded by quality criteria	Excluded in Phase 2	Excluded by full-text	Primary study
Springer Link	Test criteria and coverage measures for software integration testing	no	no	X										
Springer Link	Test based security certifications	no	no	X										
Springer Link	Technology Review: Adapting Financial Measures: Making a Business Case for Software Process Improvement*	no	no	X										
Springer Link	Systematic testing and formal verification to validate reactive programs	no	no	X										
Springer Link	Supplement of Security-Related Parts of ISO/IEC TR 15504	no	no	X										
Springer Link	Stufen der Durchführung von IT-Projekten	no	no	X										
Springer Link	Student modeling to support multiple instructional approaches	no	no	X										
Springer Link	Stretching the paradigm for software reliability assurance	no	no	X										
Springer Link	Specification of Service Level Agreements: Problems, Principles and Practices	no	no	X										

Continued on next page

Table 15 – continued from previous page

		Title and abstract exclusion			Introduction and conclusion exclusion									
Database	Title	Researcher A	Researcher B	Excluded	Researcher A	Researcher B	Excluded	Included	Discussion needed	Excluded in discussion	Excluded by quality criteria	Excluded in Phase 2	Excluded by full-text	Primary study
Springer Link	Software Testing's Place in Development	no	no	X										
Springer Link	Software Testing Process Automation Based on UTP A Case Study	no	no	X										
Springer Link	Software Testing Method Considering the Importance of Factor Combinations in Pair-Wise Testing	no	no	X										
Springer Link	Software testing for dependability assessment	no	no	X										
Springer Link	Software Testing	no	no	X										
Springer Link	Software Process Model Blueprints	no	no	X										
Springer Link	Software Process Measurement Based on Six Sigma	no	no	X										
Springer Link	Software Process and Testing Models	no	no	X										
Springer Link	Software Fault Prediction with Object-Oriented Metrics Based Artificial Immune Recognition System	no	no	X										

Continued on next page

Table 15 – continued from previous page

		Title and abstract exclusion			Introduction and conclusion exclusion									
Database	Title	Researcher A	Researcher B	Excluded	Researcher A	Researcher B	Excluded	Included	Discussion needed	Excluded in discussion	Excluded by quality criteria	Excluded in Phase 2	Excluded by full-text	Primary study
Springer Link	Software Fault Localization Using Elastic Net: A New Statistical Approach	no	no	X										
Springer Link	Software Evolution and the Code Fault Introduction Process	no	no	X										
Springer Link	Software Engineering Processes and Tools	no	no	X										
Springer Link	Software Engineering	no	no	X										
Springer Link	Software Dependability Metrics and Analysis Based on AADL Error Model	no	no	X										
Springer Link	Software Defect Estimations Under Imperfect Debugging	no	no	X										
Springer Link	Simple Metrics for Improving Software Process Performance and Capability: A Case Study	no	no	X										
Springer Link	Setting a Framework for Trusted Component Trading	no	no	X										
Springer Link	Separation of Concerns in Teaching Software Engineering	no	no	X										
Springer Link	Separating sequence overlap for automated test sequence generation	no	no	X										

Continued on next page

Table 15 – continued from previous page

		Title and abstract exclusion			Introduction and conclusion exclusion									
Database	Title	Researcher A	Researcher B	Excluded	Researcher A	Researcher B	Excluded	Included	Discussion needed	Excluded in discussion	Excluded by quality criteria	Excluded in Phase 2	Excluded by full-text	Primary study
Springer Link	Search Based Software Engineering: Techniques, Taxonomy, Tutorial	no	no	X										
Springer Link	SAS: A Tool for the GQM+Strategies Grid Derivation Process	no	no	X										
Springer Link	Safety-Critical Standards for Verification and Validation	no	no	X										
Springer Link	Runtime Verification in Context: Can Optimizing Error Detection Improve Fault Diagnosis?	no	no	X										
Springer Link	Quality Impact of Introducing Component-Level Test Automation and Test-Driven Development	no	no	X										
Springer Link	Proposing an ISO/IEC 15504-2 Compliant Method for Process Capability/Maturity Models Customization	no	no	X										
Springer Link	Process Improvement in Requirements Management: A Method Engineering Approach	no	no	X										
Springer Link	Problems and Prospects in Quantifying Software Maintainability	no	no	X										

Continued on next page

Table 15 – continued from previous page

Database	Title	Title and abstract exclusion: Researcher A	Researcher B	Excluded	Introduction and conclusion exclusion: Researcher A	Researcher B	Excluded	Included	Discussion needed	Excluded in discussion	Excluded by quality criteria	Excluded in Phase 2	Excluded by full-text	Primary study
Springer Link	Prioritizing JUnit Test Cases: An Empirical Assessment and Cost-Benefits Analysis	no	no	X										
Springer Link	Overview of Software Processes and Software Evolution	no	no	X										
Springer Link	Optimized Software Process for Fault Handling in Global Software Development	no	no	X										
Springer Link	Neural Network Based Software Reliability Prediction with the Feed of Testing Process Knowledge	no	no	X										
Springer Link	Modeling and Analysis of Grid Service Reliability Considering Fault Recovery	no	no	X										
Springer Link	Model Checking in Practice: Analysis of Generic Bootloader Using SPIN	no	no	X										
Springer Link	Life-Cycle E-commerce Testing with OO-TTCN-3	no	no	X										
Springer Link	Leveraging People-Related Maturity Issues for Achieving Higher Maturity and Capability Levels	no	no	X										

Continued on next page

Table 15 – continued from previous page

		Title and abstract exclusion			Introduction and conclusion exclusion									
Database	Title	Researcher A	Researcher B	Excluded	Researcher A	Researcher B	Excluded	Included	Discussion needed	Excluded in discussion	Excluded by quality criteria	Excluded in Phase 2	Excluded by full-text	Primary study
Springer Link	Legal Sufficiency of Testing Processes	no	no	X										
Springer Link	Jawi Generator Software Using ARM Board Under Linux	no	no	X										
Springer Link	JadexCloud - An Infrastructure for Enterprise Cloud Applications	no	no	X										
Springer Link	Introduction	no	no	X										
Springer Link	Introduction	no	no	X										
Springer Link	Introduction	no	no	X										
Springer Link	Introducing the Data Role in Models for Database Assessment	no	no	X										
Springer Link	Introducing and Developing Professional Standards in the Information Systems Curriculum	no	no	X										
Springer Link	Integrating Manual and Automatic Risk Assessment for Risk-Based Testing	no	no	X										
Springer Link	Integrating Knowledge Management and Quality Management	no	no	X										

Continued on next page

Table 15 – continued from previous page

		Title and abstract exclusion			Introduction and conclusion exclusion									
Database	Title	Researcher A	Researcher B	Excluded	Researcher A	Researcher B	Excluded	Included	Discussion needed	Excluded in discussion	Excluded by quality criteria	Excluded in Phase 2	Excluded by full-text	Primary study
Springer Link	Increasing testing productivity and software quality: A comparison of software testing methodologies within NASA	no	no	X										
Springer Link	Incorporating Software Testing as a Discipline in Curriculum of Computing Courses	no	no	X										
Springer Link	Improving Verification & Validation in the Medical Device Domain	no	no	X										
Springer Link	Improving the Testing and Testability of Software Product Lines	no	no	X										
Springer Link	Improving the ROI of Software Quality Assurance Activities: An Empirical Study	no	no	X										
Springer Link	Improving software tests using Z Specifications	no	no	X										
Springer Link	Imperfect-debugging Models	no	no	X										
Springer Link	Impact of Corporate and Organic Growth on Software Development	no	no	X										

Continued on next page

Table 15 – continued from previous page

Database	Title	Title and abstract exclusion: Researcher A	Title and abstract exclusion: Researcher B	Title and abstract exclusion: Excluded	Introduction and conclusion exclusion: Researcher A	Introduction and conclusion exclusion: Researcher B	Excluded	Included	Discussion needed	Excluded in discussion	Excluded by quality criteria	Excluded in Phase 2	Excluded by full-text	Primary study
Springer Link	How to get mature global virtual teams: a framework to improve team process management in distributed software teams	no	no	X										
Springer Link	High Integrity Ada	no	no	X										
Springer Link	Ginga-J - An Open Java-Based Application Environment for Interactive Digital Television Services	no	no	X										
Springer Link	Functional, Control and Data Flow, and Mutation Testing: Theory and Practice	no	no	X										
Springer Link	From Process Programming to Process Engineering	no	no	X										
Springer Link	Foundational Issues: Probability and Reliability	no	no	X										
Springer Link	Formalization and assessment of regulatory requirements for safety-critical software	no	no	X										
Springer Link	Finding and Ranking Research Directions for Software Testing	no	no	X										

Continued on next page

Table 15 – continued from previous page

Database	Title	Title and abstract exclusion: Researcher A	Researcher B	Excluded	Introduction and conclusion exclusion: Researcher A	Researcher B	Excluded	Included	Discussion needed	Excluded in discussion	Excluded by quality criteria	Excluded in Phase 2	Excluded by full-text	Primary study
Springer Link	Factors with Negative Influence on Software Testing Practice in Spain: A Survey	no	no	X										
Springer Link	Exploratory Study of Scientific Visualization Techniques for Program Visualization	no	no	X										
Springer Link	Experiences and Challenges of Introducing Risk-Based Testing in an Industrial Project	no	no	X										
Springer Link	Evolving a Test Oracle in Black-Box Testing	no	no	X										
Springer Link	Evaluation of software testing metrics for NASA's Mission Control Center	no	no	X										
Springer Link	Evaluation of features to support safety and quality in general practice clinical software	no	no	X										
Springer Link	Estimating test effectiveness with dynamic complexity measurement	no	no	X										
Springer Link	Error Propagation Monitoring on Windows Mobile-Based Devices	no	no	X										
Springer Link	Entropy based software processes improvement	no	no	X										

Continued on next page

Table 15 – continued from previous page

		Title and abstract exclusion			Introduction and conclusion exclusion									
Database	Title	Researcher A	Researcher B	Excluded	Researcher A	Researcher B	Excluded	Included	Discussion needed	Excluded in discussion	Excluded by quality criteria	Excluded in Phase 2	Excluded by full-text	Primary study
Springer Link	Effective Bug Hunting with Spin and Modex	no	no	X										
Springer Link	Effective Black-Box Testing with Genetic Algorithms	no	no	X										
Springer Link	Editorial	no	no	X										
Springer Link	Domain Testing	no	no	X										
Springer Link	Development Site Security Process of ISO/IEC TR 15504	no	no	X										
Springer Link	Developing control and integration software for flexible manufacturing systems	no	no	X										
Springer Link	Developing a customizable process modeling environment: Lessons learnt and future prospects	no	no	X										
Springer Link	Determining the Improvement Potential of a Software Development Organization Through Fault Analysis: A Method and a Case Study	no	no	X										
Springer Link	Determining the Effect of Tangible Business Process Modeling	no	no	X										

Continued on next page

Table 15 – continued from previous page

Database	Title	Title and abstract exclusion: Researcher A	Researcher B	Excluded	Introduction and conclusion exclusion: Researcher A	Researcher B	Excluded	Included	Discussion needed	Excluded in discussion	Excluded by quality criteria	Excluded in Phase 2	Excluded by full-text	Primary study
Springer Link	Designing and Conducting an Empirical Study on Test Management Automation	no	no	X										
Springer Link	Design by Selection: A Reuse-Based Approach for Business Process Modeling	no	no	X										
Springer Link	Deriving a Valid Process Simulation from Real World Experiences	no	no	X										
Springer Link	Dependability and Trust in Organizational and Domestic Computer Systems	no	no	X										
Springer Link	Defining a Requirements Process Improvement Model	no	no	X										
Springer Link	Dealing with Scalability in an Event-Based Infrastructure to Support Global Software Development	no	no	X										
Springer Link	Data Mining for Software Testing	no	no	X										
Springer Link	Conclusion	no	no	X										
Springer Link	Commitment Nets in Software Process Improvement	no	no	X										

Continued on next page

Table 15 – continued from previous page

		Title and abstract exclusion			Introduction and conclusion exclusion									
Database	Title	Researcher A	Researcher B	Excluded	Researcher A	Researcher B	Excluded	Included	Discussion needed	Excluded in discussion	Excluded by quality criteria	Excluded in Phase 2	Excluded by full-text	Primary study
Springer Link	Capture-Recapture Methods	no	no	X										
Springer Link	Building Test Cases through Model Driven Engineering	no	no	X										
Springer Link	Building Reusable Testing Assets for a Software Product Line	no	no	X										
Springer Link	Building Reusable Testing Assets for a Software Product Line	no	no	X										
Springer Link	Automating unit and integration testing with partial oracles	no	no	X										
Springer Link	Automated testing as an aid to systems integration	no	no	X										
Springer Link	Automated Test Reduction Using an Info-Fuzzy Network	no	no	X										
Springer Link	Automated Software Testing with .NET	no	no	X										
Springer Link	Assessing the Dependability of Embedded Software Systems Using the Dynamic Flowgraph Methodology	no	no	X										
Springer Link	Artificial neural networks as multi-networks automated test oracle	no	no	X										
Springer Link	An Open Framework for Managed Regression Testing	no	no	X										

Continued on next page

Table 15 – continued from previous page

Database	Title	Title and abstract exclusion: Researcher A	Researcher B	Excluded	Introduction and conclusion exclusion: Researcher A	Researcher B	Excluded	Included	Discussion needed	Excluded in discussion	Excluded by quality criteria	Excluded in Phase 2	Excluded by full-text	Primary study
Springer Link	An ISO/IEC 15504 Security Extension	no	no	X										
Springer Link	An Investigation on the Occurrence of Service Requests in Commercial Software Applications	no	no	X										
Springer Link	An investigation on the feasibility of cross-project defect prediction	no	no	X										
Springer Link	An Introduction to Visual Studio Team Test	no	no	X										
Springer Link	An adaptive and trustworthy software testing framework on the grid	no	no	X										
Springer Link	Adapting the Test Process for Web Applications Strategies from Practice	no	no	X										
Springer Link	Ada for High Integrity	no	no	X										
Springer Link	A Tool to Create Process-Agents for OEC-SPM from Historical Project Data	no	no	X										
Springer Link	A Software Tool to Support the Integrated Management of Software Projects in Mature SMEs	no	no	X										

Continued on next page

Table 15 – continued from previous page

Database	Title	Title and abstract exclusion: Researcher A	Researcher B	Excluded	Introduction and conclusion exclusion: Researcher A	Researcher B	Excluded	Included	Discussion needed	Excluded in discussion	Excluded by quality criteria	Excluded in Phase 2	Excluded by full-text	Primary study
Springer Link	A set of complexity metrics for guiding the software test process	no	no	X										
Springer Link	A Review of Prominent Work on Agile Processes Software Process Improvement and Process Tailoring Practices	no	no	X										
Springer Link	A Refined Non-parametric Algorithm for Sequential Software Reliability Estimation	no	no	X										
Springer Link	A Quantitative Analysis into the Economics of Correcting Software Bugs	no	no	X										
Springer Link	A Novel BCC Algorithm for Function Optimization	no	no	X										
Springer Link	A Meta-model Framework for Software Process Modeling	no	no	X										
Springer Link	A Maturity Framework for the Enterprise Modeling and Engineering Process	no	no	X										
Springer Link	A Goal-Oriented Software Testing Methodology	no	no	X										
Springer Link	A Formal Technique for Reducing Software Testing Time Complexity	no	no	X										

Continued on next page

Table 15 – continued from previous page

		Title and abstract exclusion			Introduction and conclusion exclusion									
Database	Title	Researcher A	Researcher B	Excluded	Researcher A	Researcher B	Excluded	Included	Discussion needed	Excluded in discussion	Excluded by quality criteria	Excluded in Phase 2	Excluded by full-text	Primary study
Springer Link	A Fault Prediction Model with Limited Fault Data to Improve Test Process	no	no	X										
Springer Link	A Coverage Relationship Model for Test Case Selection and Ranking for Multi-version Software	no	no	X										
Springer Link	A binomial software reliability model based on coverage of structural testing criteria	no	no	X										
ScienceDirect	Transfer learning for cross-company software defect prediction	no	no	X										
ScienceDirect	Software testing processes as a linear dynamic system	no	no	X										
ScienceDirect	Software Organizations and Test Process Development	no	no	X										
ScienceDirect	Performance analysis of software reliability growth models with testing-effort and change-point	no	no	X										
ScienceDirect	Optimal test sequence generation using firefly algorithm	no	no	X										
ScienceDirect	Optimal and adaptive testing for software reliability assessment	no	no	X										

Continued on next page

Table 15 – continued from previous page

Database	Title	Title and abstract exclusion: Researcher A	Researcher B	Excluded	Introduction and conclusion exclusion: Researcher A	Researcher B	Excluded	Included	Discussion needed	Excluded in discussion	Excluded by quality criteria	Excluded in Phase 2	Excluded by full-text	Primary study
ScienceDirect	Mathematical modeling of software reliability testing with imperfect debugging	no	no	X										
ScienceDirect	Applying machine learning to software fault-proneness prediction	no	no	X										
ScienceDirect	An investigation of artificial neural networks based prediction systems in software project management	no	no	X										
ScienceDirect	An architectural model for software testing lesson learned systems	no	no	X										
ScienceDirect	A case study in branch automation	no	no	X										
IEEE	Workflow-Based Process Management of Network Management Testing	no	no	X										
IEEE	What do the software reliability growth model parameters represent?	no	no	X										
IEEE	Using simulation for assessing the real impact of test coverage on defect coverage	no	no	X										
IEEE	Using sensitivity analysis to validate a state variable model of the software test process	no	no	X										

Continued on next page

Table 15 – continued from previous page

Database	Title	Title and abstract exclusion: Researcher A	Researcher B	Excluded	Introduction and conclusion exclusion: Researcher A	Researcher B	Excluded	Included	Discussion needed	Excluded in discussion	Excluded by quality criteria	Excluded in Phase 2	Excluded by full-text	Primary study
IEEE	Using grounded theory to understand testing engineers' soft skills of third-party software testing centers	no	no	X										
IEEE	URL-driven automated testing	no	no	X										
IEEE	The Sleuth approach to aerospace software testing	no	no	X										
IEEE	Testing requirements for mobile applications	no	no	X										
IEEE	Testing process model and classification of test methods for embedded software of electrical products	no	no	X										
IEEE	Testing of the rapidly developed prototypes	no	no	X										
IEEE	Testing applications using domain based testing and Sleuth	no	no	X										
IEEE	Test case generation and reduction by automated input-output analysis	no	no	X										
IEEE	Tailoring test process by using the component-based development paradigm and the XML technology	no	no	X										

Continued on next page

Table 15 – continued from previous page

		Title and abstract exclusion			Introduction and conclusion exclusion									
Database	Title	Researcher A	Researcher B	Excluded	Researcher A	Researcher B	Excluded	Included	Discussion needed	Excluded in discussion	Excluded by quality criteria	Excluded in Phase 2	Excluded by full-text	Primary study
IEEE	Supporting the software testing process through specification animation	no	no	X										
IEEE	Study of task profile oriented embedded software test aiming to improve reliability	no	no	X										
IEEE	Statistical software debugging: From bug predictors to the main causes of failure	no	no	X										
IEEE	Software test process control: status and future directions	no	no	X										
IEEE	Software reliability model with bathtub-shaped fault detection rate	no	no	X										
IEEE	Software release control using defect based quality estimation	no	no	X										
IEEE	Sensitivity analysis of a state variable model of the Software Test Process	no	no	X										
IEEE	Selection, Evaluation and Generation of Test Cases in an Industrial Setting: A Process and a Tool	no	no	X										
IEEE	Secondary Power Supply Universal Test System Based on PXI Bus	no	no	X										

Continued on next page

Table 15 – continued from previous page

		Title and abstract exclusion			Introduction and conclusion exclusion									
Database	Title	Researcher A	Researcher B	Excluded	Researcher A	Researcher B	Excluded	Included	Discussion needed	Excluded in discussion	Excluded by quality criteria	Excluded in Phase 2	Excluded by full-text	Primary study
IEEE	Search-based Prediction of Fault-slip-through in Large Software Projects	no	no	X										
IEEE	Research on a Behavior-Based Software Test Process Model	no	no	X										
IEEE	Research of Software Defect Prediction Model Based on Gray Theory	no	no	X										
IEEE	Reliability Growth Modeling for Software Fault Detection Using Particle Swarm Optimization	no	no	X										
IEEE	Regression Testing Process Improvement for Specification Evolution of Real-World Protocol Software	no	no	X										
IEEE	Re-engineering the test development process	no	no	X										
IEEE	Real-time operating systems tutorial	no	no	X										
IEEE	PSO based test coverage analysis for event driven software	no	no	X										
IEEE	Programs That Test Themselves	no	no	X										

Continued on next page

Table 15 – continued from previous page

Database	Title	Title and abstract exclusion: Researcher A	Researcher B	Excluded	Introduction and conclusion exclusion: Researcher A	Researcher B	Excluded	Included	Discussion needed	Excluded in discussion	Excluded by quality criteria	Excluded in Phase 2	Excluded by full-text	Primary study
IEEE	Proceedings The Eighth International Symposium on Software Reliability Engineering	no	no	X										
IEEE	Proceedings of the 29th Annual International Computer Software and Applications Conference (COMPSAC 2005)	no	no	X										
IEEE	Performance evaluation model for test process	no	no	X										
IEEE	Optimal Software Testing Case Design Based on Self-Learning Control Algorithm	no	no	X										
IEEE	On the Trend of Remaining Software Defect Estimation	no	no	X										
IEEE	On the determination of an appropriate time for ending the software testing process	no	no	X										
IEEE	Modeling and Simulating the Quality of Sequential Iterative Development Processes	no	no	X										
IEEE	Modeling and control of the incremental software test process	no	no	X										

Continued on next page

Table 15 – continued from previous page

Database	Title	Title and abstract exclusion: Researcher A	Researcher B	Excluded	Introduction and conclusion exclusion: Researcher A	Researcher B	Excluded	Included	Discussion needed	Excluded in discussion	Excluded by quality criteria	Excluded in Phase 2	Excluded by full-text	Primary study
IEEE	Model-Based Time Estimate of Software Test Processes	no	no	X										
IEEE	Implementing Software Test Management Using SpiraTeam Tool	no	no	X										
IEEE	IDATG: an open tool for automated testing of interactive software	no	no	X										
IEEE	Genetic algorithm based test data generator	no	no	X										
IEEE	Fire control radar automatic testing modernization - better, faster, sustainable	no	no	X										
IEEE	Feedback control of the software test process through measurements of software reliability	no	no	X										
IEEE	Fault exposure ratio estimation and applications	no	no	X										
IEEE	Extended TTCN in software testing	no	no	X										
IEEE	Experience of applying statistical control techniques to the function test phase of a large telecommunications system	no	no	X										

Continued on next page

Table 15 – continued from previous page

		Title and abstract exclusion			Introduction and conclusion exclusion									
Database	Title	Researcher A	Researcher B	Excluded	Researcher A	Researcher B	Excluded	Included	Discussion needed	Excluded in discussion	Excluded by quality criteria	Excluded in Phase 2	Excluded by full-text	Primary study
IEEE	Event-Based GUI Testing and Reliability Assessment Techniques – An Experimental Insight and Preliminary Results	no	no	X										
IEEE	Evaluation of software testing process based on Bayesian networks	no	no	X										
IEEE	Engineering process for an online testing process of control software in production systems	no	no	X										
IEEE	Developing a customized software engineering testing for Shared Banking Services (SBS) System	no	no	X										
IEEE	Confidence-based reliability and statistical coverage estimation	no	no	X										
IEEE	Composing a framework to automate testing of operational Web-based software	no	no	X										
IEEE	Category Model of Process of Repeated Software Testing	no	no	X										
IEEE	Better testing through oracle selection: (NIER track)	no	no	X										

Continued on next page

Table 15 – continued from previous page

Database	Title	Title and abstract exclusion: Researcher A	Researcher B	Excluded	Introduction and conclusion exclusion: Researcher A	Researcher B	Excluded	Included	Discussion needed	Excluded in discussion	Excluded by quality criteria	Excluded in Phase 2	Excluded by full-text	Primary study
IEEE	Best Practices for the Formal Software Testing Process: A Menu of Testing Tasks [Book Review]	no	no	X										
IEEE	Automation testing process modeling method of SOA-based isomerous software	no	no	X										
IEEE	Arranging software test cases through an optimization method	no	no	X										
IEEE	Application of missing data approaches in software testing research	no	no	X										
IEEE	An overview of software cybernetics	no	no	X										
IEEE	An Industrial Survey on Contemporary Aspects of Software Testing	no	no	X										
IEEE	An effective equivalence partitioning method to design the test case of the WEB application	no	no	X										
IEEE	An automated testing methodology based on self-checking software	no	no	X										
IEEE	An Approach Using RUP Test Discipline Process for Shared Banking Services (SBS) System	no	no	X										

Continued on next page

Table 15 – continued from previous page

		Title and abstract exclusion			Introduction and conclusion exclusion									
Database	Title	Researcher A	Researcher B	Excluded	Researcher A	Researcher B	Excluded	Included	Discussion needed	Excluded in discussion	Excluded by quality criteria	Excluded in Phase 2	Excluded by full-text	Primary study
IEEE	Advanced models for software reliability prediction	no	no	X										
IEEE	Adaptive Software Testing in the Context of an Improved Controlled Markov Chain Model	no	no	X										
IEEE	A Transformation-Based Approach to Generating Scenario-Oriented Test Cases from UML Activity Diagrams for Concurrent Applications	no	no	X										
IEEE	A tool based approach for automation of GUI applications	no	no	X										
IEEE	A three-dimensional visualization tool for software fault analysis of a distributed system	no	no	X										
IEEE	A state model for the Software Test Process with automated parameter identification	no	no	X										
IEEE	A Scenario-Centric Approach for the Definition of the Formal Test Specifications of Reactive Systems	no	no	X										
IEEE	A quantitative Learning Model for Software Test Process	no	no	X										

Continued on next page

Table 15 – continued from previous page

Database	Title	Title and abstract exclusion: Researcher A	Researcher B	Excluded	Introduction and conclusion exclusion: Researcher A	Researcher B	Excluded	Included	Discussion needed	Excluded in discussion	Excluded by quality criteria	Excluded in Phase 2	Excluded by full-text	Primary study
IEEE	A proposed framework for full automation of software testing process	no	no	X										
IEEE	A proposal of a process model to create a test factory	no	no	X										
IEEE	A novel test process modeling method for automatic test	no	no	X										
IEEE	A new role of graphical simulation: Software testing	no	no	X										
IEEE	A Model of Third-Party Integration Testing Process for Foundation Software Platform	no	no	X										
IEEE	A model of knowledge management system in managing knowledge of software testing environment	no	no	X										
IEEE	A framework for Web applications testing	no	no	X										
IEEE	A Framework for the Vamp;V Capability Assessment Focused on the Safety-Criticality	no	no	X										
IEEE	A formal model of the software test process	no	no	X										

Continued on next page

Table 15 – continued from previous page

		Title and abstract exclusion			Introduction and conclusion exclusion									
Database	Title	Researcher A	Researcher B	Excluded	Researcher A	Researcher B	Excluded	Included	Discussion needed	Excluded in discussion	Excluded by quality criteria	Excluded in Phase 2	Excluded by full-text	Primary study
IEEE	A Family of Software Reliability Growth Models	no	no	X										
IEEE	A criticism on the capture-and-recapture method for software reliability assurance	no	no	X										
IEEE	A contract-checking test method for CTCS Hardware-in-the-Loop Simulation system	no	no	X										
IEEE	A Conceptual Framework to Integrate Fault Prediction Sub-Process for Software Product Lines	no	no	X										
IEEE	A Conceptual Framework for Open Source Software Test Process	no	no	X										
IEEE	A Comparative Evaluation of Unit Testing Techniques on a Mobile Platform	no	no	X										
ACM	Virtual framework for testing the reliability of system software on embedded systems	no	no	X										
ACM	White-box testing for database-driven applications: a requirements analysis	no	no	X										

Continued on next page

Table 15 – continued from previous page

		Title and abstract exclusion			Introduction and conclusion exclusion									
Database	Title	Researcher A	Researcher B	Excluded	Researcher A	Researcher B	Excluded	Included	Discussion needed	Excluded in discussion	Excluded by quality criteria	Excluded in Phase 2	Excluded by full-text	Primary study
ACM	Using the OPT improvement approach in the SQL/DS development environment	no	no	X										
ACM	Using simulation to support implementation of flexible manufacturing Cell	no	no	X										
ACM	Using implied scenarios in security testing	no	no	X										
ACM	Using fault slippage measurement for monitoring software process quality during development	no	no	X										
ACM	Using an SQL coverage measurement for testing database applications	no	no	X										
ACM	Towards economical software release recommendations	no	no	X										
ACM	Towards certifying the testing process of Open-Source Software: New challenges or old methodologies?	no	no	X										
ACM	Towards a documentation maturity model	no	no	X										

Continued on next page

Table 15 – continued from previous page

		Title and abstract exclusion			Introduction and conclusion exclusion									
Database	Title	Researcher A	Researcher B	Excluded	Researcher A	Researcher B	Excluded	Included	Discussion needed	Excluded in discussion	Excluded by quality criteria	Excluded in Phase 2	Excluded by full-text	Primary study
ACM	The influence of multiple artifacts on the effectiveness of software testing	no	no	X										
ACM	The data mining approach to automated software testing	no	no	X										
ACM	Test prioritization for pairwise interaction coverage	no	no	X										
ACM	Test case selection and prioritization: risk-based or design-based?	no	no	X										
ACM	Surveying model based testing approaches characterization attributes	no	no	X										
ACM	Supporting the selection of model-based testing approaches for software projects	no	no	X										
ACM	Specification-based regression test selection with risk analysis	no	no	X										
ACM	Simulating patient flow through an Emergency Department using process-driven discrete event simulation	no	no	X										
ACM	Selection and execution of user level test cases for energy cost evaluation of smartphones	no	no	X										

Continued on next page

Table 15 – continued from previous page

Database	Title	Title and abstract exclusion: Researcher A	Researcher B	Excluded	Introduction and conclusion exclusion: Researcher A	Researcher B	Excluded	Included	Discussion needed	Excluded in discussion	Excluded by quality criteria	Excluded in Phase 2	Excluded by full-text	Primary study
ACM	Security policy testing via automated code generation	no	no	X										
ACM, IEEE	Search-enhanced testing (NIER) track	no	no	X										
ACM	Requirements traceability in automated test generation: application to smart card software validation	no	no	X										
ACM	Requirements elicitation for an intelligent software test environment for the physically challenged	no	no	X										
ACM, IEEE	Reliability Analysis of Component Software Based on Testing Data Transformation	no	no	X										
ACM	Perspectives on automated testing of aspect-oriented programs	no	no	X										
ACM	Optimal and adaptive testing with cost constraints	no	no	X										
ACM	Neural nets method for estimation of the software retesting necessity	no	no	X										
ACM	Model driven transformation between design models to system test models using UML: a survey	no	no	X										

Continued on next page

Table 15 – continued from previous page

		Title and abstract exclusion			Introduction and conclusion exclusion									
Database	Title	Researcher A	Researcher B	Excluded	Researcher A	Researcher B	Excluded	Included	Discussion needed	Excluded in discussion	Excluded by quality criteria	Excluded in Phase 2	Excluded by full-text	Primary study
ACM	MobileTest: A Tool Supporting Automatic Black Box Test for Software on Smart Mobile Devices	no	no	X										
ACM	Mitigating program security vulnerabilities: Approaches and challenges	no	no	X										
ACM	Making testing product lines more efficient by improving the testability of product line architectures	no	no	X										
ACM	Lessons learnt from the analysis of large-scale corporate databases	no	no	X										
ACM	Knowledge management and software testing	no	no	X										
ACM	Key challenges in software internationalisation	no	no	X										
ACM	JSXM: a tool for automated test generation	no	no	X										
ACM	Investigating test-and-fix processes of incremental development using hybrid process simulation	no	no	X										
ACM	Intelligent user interfaces for correspondence domains (panel session): moving IUIs off the desktop	no	no	X										

Continued on next page

Table 15 – continued from previous page

Database	Title	Title and abstract exclusion: Researcher A	Researcher B	Excluded	Introduction and conclusion exclusion: Researcher A	Researcher B	Excluded	Included	Discussion needed	Excluded in discussion	Excluded by quality criteria	Excluded in Phase 2	Excluded by full-text	Primary study
ACM	Improving the handsets network test process via DMAIC concepts	no	no	X										
ACM	Harmonizing the test support for object-oriented legacy systems using state-of-the-art test tools	no	no	X										
ACM	GridUnit: software testing on the grid	no	no	X										
ACM	Generation of improved test cases from UML state diagram using genetic algorithm	no	no	X										
ACM	Exploiting software architecture to support requirements satisfaction testing	no	no	X										
ACM	Evaluation of model-based testing techniques selection approaches: An external replication	no	no	X										
ACM	Emergency department simulation and determination of optimal attending physician staffing schedules	no	no	X										
ACM	Early estimation of defect density using an in-process Haskell metrics model	no	no	X										

Continued on next page

Table 15 – continued from previous page

Database	Title	Title and abstract exclusion: Researcher A	Researcher B	Excluded	Introduction and conclusion exclusion: Researcher A	Researcher B	Excluded	Included	Discussion needed	Excluded in discussion	Excluded by quality criteria	Excluded in Phase 2	Excluded by full-text	Primary study
ACM, Springer Link	Development system security process of ISO/IEC TR 15504 and security considerations for software process improvement	no	no	X										
ACM	Developing a synchronous web seminar application for online learning	no	no	X										
ACM	Deriving models of software fault-proneness	no	no	X										
ACM	Demand-driven structural testing with dynamic instrumentation	no	no	X										
ACM	Data space testing	no	no	X										
ACM	Choices, choices: comparing between CHOC'LATE and the classification-tree methodology	no	no	X										
ACM	Case-based software reliability assessment by fault injection unified procedures	no	no	X										
ACM	Automating software tests with partial oracles in integrated environments	no	no	X										

Continued on next page

Table 15 – continued from previous page

		Title and abstract exclusion			Introduction and conclusion exclusion									
Database	Title	Researcher A	Researcher B	Excluded	Researcher A	Researcher B	Excluded	Included	Discussion needed	Excluded in discussion	Excluded by quality criteria	Excluded in Phase 2	Excluded by full-text	Primary study
ACM	An empirical study of regression testing techniques incorporating context and lifetime factors and improved cost-benefit models	no	no	X										
ACM	An empirical comparison between direct and indirect test result checking approaches	no	no	X										
ACM	Adaptive testing of software components	no	no	X										
ACM	Adapting model-based testing techniques to DEVS models validation	no	no	X										
ACM	Adapting and adjusting test process reflecting characteristics of embedded software and industrial properties based on referential models	no	no	X										
ACM	Active learning for automatic classification of software behavior	no	no	X										
ACM	A study on agility and testing processes in software organizations	no	no	X										
ACM	A software test documentation standard	no	no	X										
ACM	A software frame for modeling and runtime control of adaptive testing	no	no	X										

Continued on next page

Table 15 – continued from previous page

		Title and abstract exclusion			Introduction and conclusion exclusion									
Database	Title	Researcher A	Researcher B	Excluded	Researcher A	Researcher B	Excluded	Included	Discussion needed	Excluded in discussion	Excluded by quality criteria	Excluded in Phase 2	Excluded by full-text	Primary study
ACM	A model-based approach to ongoing product evaluation	no	no	X										
ACM	A framework and tool support for the systematic testing of model-based specifications	no	no	X										
ACM	A case study in system integration using the Build approach	no	no	X										

D Summarized inquiries and responses from internal validation

We provided the authors that had been selected for the internal validation with a list of all TPI approaches that had been found in the systematic literature review and asked them if they think this list was complete. Furthermore, we presented them our exclusion criteria that have been decisive for the pre-selection of applicable approaches for the case study and the particular result for the approach presented in the specific author's study. Individually, the authors were asked if they agree to the evaluation of their approach.

The authors of the studies identified by the snowball sampling were also asked for additional papers about their research topic.

One of the authors of [Taipale and Smolander, 2006] validated our list of approaches as "good" and stated that it even contains approaches unknown to him.

The application of our evaluation criteria had first resulted in the exclusion of the TestSPICE approach due to the absence of an assessment instrument and the availability of only a concept. The information given by one of the authors of [Steiner et al., 2012] let us revoke this exclusion. He pointed out that sufficient information on TestSPICE is available on the TestSPICE website [3], and that an process assessment model is available. Therefore, TestSPICE has been included as an generally applicable approach for the further research.

One of the authors of [Heiskanen et al., 2012] agreed to our evaluation regarding the ATG add-on for TPI. He stated that an spreadsheet had been used for the assessment which had not been published.

In a short interview about TMMi conducted with the author of [Rasking, 2011] he agreed to our evaluation results for TMMi and also confirmed the results of our systematic review as "very complete".

[3] `www.testspice.info`

E Interview questions

Warm-up questions

- How old are you?
- What is your educational background?
- How many years of working experience do you have within this organization? And in IT in general?

Overview of work tasks

- What is your role in the organization?
- Which systems/applications are you working with?
- Could you please give us an overview of your usual work tasks?

Questions specific to testing

- How is your work related to testing?
- When you think of the testing you are doing, do you follow a specific testing process?
- Do you follow a specific method?
- How are regressiontests and retests done?
- Who is involved in the test processes, inside or outside of your team?
- Do you assign testing tasks to specific persons?
- In which activities is the customer involved?
- Could you please define your stakeholders?
- How is the stakeholder involved in the overall project? And at what time?
- How do you plan for your testing, what are the activities involved in planning, like resource management, etc.?
- Do you have a test plan? What does the test plan include, for example test assignments, test scope, roles or responsibilities?
- Who is involved in planning? Is the customer also involved?
- What are the things you consider when you plan your testing?
- Are you monitoring the testing activities?

- Do you analysis the product risks and do you have a test strategy related to the product risks?
- Could you please explain the differences of your test levels?
- How do you design test cases?
- Do you use specific test design techniques?
- How is the relation between requirements and test cases?
- How do you document the test cases? Do you follow any specific template? Please provide us with an example document.
- Do you have any tools to support the testing?
- Is everyone on the same level of knowledge about testing tools within your team?
- How do you handle communication about the project progress amongst your team? How is the communication with the customer done?
- How do you report the testing? Please provide us with the document.
- Do you have any metrics to estimate or monitor the test process? How do you record them?
- What is the process when you find a defect?
- Do you have a defect document? Please provide us with a template how you report them.
- Do you think every one follows the same process and uses the same resources?
- How does the test environment look like? Who is responsible? How is it maintained?
- Since you dont have any specific role as tester how did you gain knowledge about testing? Do you undergo any training?

Statistical questions about the interview

- How do you feel about the duration of the interview?
- Was it difficult to answer the questions?
- We used open ended questions. Would you have preferred ‘yes’ and ‘no’ questions?

F Characteristics of approaches

Table 16: Characteristics of TMM.

Characteristics	
Approach	TMM - Testing Maturity Model
Reference	[Rana and Ahmad, 2005], [Burnstein et al., 1996], [Tayamanon et al., 2011], [Burnstein et al., 1999], [Homyen, 1998], [Burnstein, 2003], [Jacobs et al., 2000], [Suwannasart, 1996]
Based on/influenced by	CMM
	Gelperin and Hetzel's evolutionary testing model
	Industrial testing practices studies
	Beizer's progressive phases of a tester's mental model
	Thayer's management model
Domain	-
Developed by	Illinois Institute of Technology, USA
Status of development	Complete
	Validated in an experiment
Completeness of information	Detailed description
	Additional information: team selection and training
Assessment model	Yes
Assessment procedure	Available
Assessment instrument	Available
	Questionnaire
	Mainly yes/no questions + open questions
	Individual interviews after first round of pre-defined questions
Improvement suggestions	Available
	Recommendation of testing tools and test-related metrics
Process reference model	No
Maturity structure	Yes
	1: Initial
	2: Phase-Definition
	3: Integration
	4: Management and Measurement
	5: Optimizing/Defect prevention and quality control
Model representation	Staged
Character of approach	Qualitative
Structure/components	Maturity levels
	Maturity goals (MG)
	Maturity subgoals (MSG)
	Activities, tasks, and responsibilities (ATR)
	Metrics
	Tool recommendations
	Critical views (managers, developers, usersclients)
Addressing	Test managers
	Test groups
	Software quality assurance staff
Process areas	Testing and debugging goals and policies
	Test planning process
	Testing techniques and methods
	Test organization
	Technical training program
	Software life cycle
	Controlling and monitoring
	Review
	Test measurement program
	Software quality evaluation
	Defect prevention
	Quality control
	Test process optimization

Table 17: Characteristics of TMMi®

Characteristics	
Approach	TMMi®- Test Maturity Model integration
Reference	[Rasking, 2011], [van Veenendal, 2008]
Based on/influenced by	CMMi (staged representation), TMM
Domain	-
Developed by	TMMi Foundation
Status of development	Complete
Completeness of information	Detailed description
Assessment model	Yes
Assessment procedure	Available
Assessment instrument	Not available
Improvement suggestions	Available
Process reference model	No
Maturity structure	Yes 1: Initial 2: Managed 3: Defined 4: Measured 5: Optimization
Model representation	Staged
Character of approach	Qualitative
Structure/components	Maturity levels Process areas Specific goals Specific practices Generic goals Generic practices
Addressing	Test managers Test engineers Software quality professionals
Process areas	Test policy and strategy Test planning Test monitoring and control Test design and execution Test environment Test organization Test training program Test lifecycle and integration Non-functional testing Peer reviews Test measurement Product quality evaluation Advanced reviews Defect prevention Quality control Test process optimization

Table 18: Characteristics of MND-TMM

Characteristics	
Approach	MND-TMM - Ministry of National Defense-Testing Maturity Model
Reference	[Ryu et al., 2008]
Based on/influenced by	TMM
Domain	Defense - military weapon systems
Developed by	Partially supported by Defense Acquisition Program Administration and Agency for Defense Development
Status of development	Under development
Completeness of information	Concept
Assessment model	Yes
Assessment procedure	Not available
Assessment instrument	Not available
Improvement suggestions	Not available
Process reference model	No
Maturity structure	Yes 5 levels
Model representation	Staged + continuous Similar to the continuous approach of CMMi
Character of approach	Qualitative
Structure/components	Maturity levels Categories Test process areas (TPAs) Specific goals Specific practices Sub practices Generic goals Common features
Addressing	-
Process areas	Military: Software quality evaluation Process: Test strategy Test planning Test process management Infrastructure: Test organization Test environment Testware management Techniques: Testing techniques Test specification Fault management

Table 19: Characteristics of MB-VV-MM

Characteristics	
Approach	MB-VV-MM - Metrics Based Verification and Validation Maturity Model
Reference	[Jacobs and Trienekens, 2002]
Based on/influenced by	TMM
Domain	-
Developed by	Consortium of industrial companies (defense and civil systems, telecommunication and satellites, consumer and professional electronics), consultancy and service agencies (software quality, testing, and related vocational training) and an academic institute (Frits Philips Institute, University of Technology - Eindhoven), Netherlands
Status of development	Under development Validated in various experiments
Completeness of information	Concept
Assessment model	Yes
Assessment procedure	Not available
Assessment instrument	Not available
Improvement suggestions	Not available
Process reference model	No
Maturity structure	Yes 1: Initial 2: Repeatable 3: Defined 4: Managed and aligned 5: Optimizing
Model representation	Staged Planned to address continuous aspects
Character of approach	Quantitative/qualitative
Structure/components	Maturity levels Process areas Process goals Metrics Generic practices
Addressing	-
Process areas	V&V Environment V&V Design methodology V&V Monitor and control V&V Project planning V&V Policy and goals Peer reviews V&V Lifecycle embedding Training and program Organization embedding Qualitative process measurement Quality measurement and evaluation Organizational alignment Process optimization Quality management Defect prevention

Table 20: Characteristics of TIM

Characteristics	
Approach	TIM - Test Improvement Model
Reference	[Ericson et al., 1997]
Based on/influenced by	CMM
	TMM - Testability Maturity Model
Domain	-
Developed by	-
Status of development	Complete
Completeness of information	Brief description
Assessment model	Yes
Assessment procedure	Not available
Assessment instrument	Not available
	No use of yes/no-questions
Improvement suggestions	Not available
Process reference model	No
Maturity structure	Yes
	Initial
	Baselining
	Cost-effectiveness
	Risk-lowering
	Optimizing
Model representation	Unknown
Character of approach	Qualitative
Structure/components	Key areas
	Maturity levels
	Overall goal for the level
	Subgoals
	Activities
	Checkpoints
Addressing	-
Process areas	Organization
	Planning and tracking
	Test cases
	Testware
	Reviews

Table 21: Characteristics of TPI

Characteristics	
Approach	TPI - Test Process Improvement
Reference	[Koomen and Pol, 1999], [Koomen, 2002]
Based on/influenced by	SPICE
	Tmap
Domain	-
Developed by	Sogeti
Status of development	Complete
Completeness of information	Detailed description
Assessment model	Yes
Assessment procedure	Available
Assessment instrument	Available
	Checkpoints
Improvement suggestions	Available
Process reference model	Yes
Maturity structure	Yes
	Controlled
	Efficient
	Optimized
Model representation	Continuous
Character of approach	Qualitative
Structure/components	Key areas (20)
	Maturity levels
	Checkpoints (300)
	Test maturity matrix
	Improvement suggestions
	Dependencies between different levels of the key areas
Addressing	-
Process areas	Test strategy
	Life-cycle model
	Moment of involvement
	Estimation and planning
	Test specification techniques
	Static test techniques
	Metrics
	Test tools
	Test environment
	Office environment
	Commitment and motivation
	Test functions and training
	Scope of methodology
	Communication
	Reporting
	Defect management
	Testware management
	Test process management
	Evaluation
	Low-level testing

Table 22: Characteristics of TPI®NEXT

Characteristics	
Approach	TPI®NEXT
Reference	[v. Ewijk et al., 2013]
Based on/influenced by	Tmap NEXT
Domain	-
Developed by	Sogeti
Status of development	Complete
Completeness of information	Detailed description
Assessment model	Yes
Assessment procedure	Available
Assessment instrument	Available
Improvement suggestions	Available
Process reference model	Yes
Maturity structure	Yes
Model representation	Continuous
Character of approach	Qualitative
Structure/components	Key areas (16)
	Maturity levels
	Checkpoints (157)
	Clusters
	Enablers
	Test maturity matrix
	Improvement suggestions
	Dependencies between different levels of the key areas
Addressing	-
Process areas	Stakeholder commitment
	Degree of involvement
	Test strategy
	Test organization
	Communication
	Reporting
	Test process management
	Estimating and planning
	Metrics
	Defect management
	Testware management
	Methodology practice
	Tester professionalism
	Test case design
	Test tools
	Test environment

Table 23: Characteristics of TPI®Automotive

Characteristics	
Approach	TPI®Automotive
Reference	[TPI, 2004]
Based on/influenced by	Tmap
	TPI
Domain	Automotive
Developed by	Sogeti
	German automotive industry
Status of development	Complete
Completeness of information	Detailed description
Assessment model	Yes
Assessment procedure	Available
Assessment instrument	Available
	Checkpoints
Improvement suggestions	Available
Process reference model	Yes
Maturity structure	Yes
	Maximum 4 levels (individual for each key area)
Model representation	Continuous
Character of approach	Qualitative
Structure/components	Key areas (21)
	Maturity levels
	Checkpoints
	Test maturity matrix
	Improvement suggestions
	Dependencies between different levels of the key areas
Addressing	-
Process areas	Test strategy
	Life-cycle model
	Moment of involvement
	Estimation and planning
	Test design techniques
	Static test techniques
	Metrics
	Test automation
	Test environment
	Office and laboratory environment
	Commitment and motivation
	Test functions and training
	Scope of methodology
	Communication
	Reporting
	Defect management
	Testware management
	Test process management
	Evaluation
	Low-level testing
	Integration testing

Table 24: Characteristics of ATG add-on for TPI

Characteristics	
Approach	ATG add-on for TPI - Test Process Improvement Model for Automated Test Generation
Reference	[Heiskanen et al., 2012]
Based on/influenced by	TPI
Domain	Automated testing
Developed by	-
Status of development	Complete Validated in a case study
Completeness of information	Brief description
Assessment model	Yes
Assessment procedure	Not available
Assessment instrument	Not available Checkpoints
Improvement suggestions	Not available
Process reference model	No
Maturity structure	Yes Maximum 4 levels (individual for each key area)
Model representation	Continuous
Character of approach	Qualitative
Structure/components	Key areas Maturity levels Checkpoints Test maturity matrix Improvement suggestions Dependencies between different levels of the key areas
Addressing	-
Process areas	Test strategy Life-cycle model Moment of involvement Estimation and planning Test specification techniques Static test techniques Metrics Test tools Test environment Office environment Commitment and motivation Test functions and training Scope of methodology Communication Reporting Defect management Testware management Test process management Evaluation Low-level testing Modeling approach Use of models Test confidence Technological and methodological knowledge

Table 25: Characteristics of Emb-TPI

Characteristics	
Approach	Emb-TPI - Embedded Test Process Improvement Model
Reference	[Jung, 2009]
Based on/influenced by	TPI
Domain	Embedded software
Developed by	-
Status of development	Complete
	Validated in a case study and a survey
Completeness of information	Brief description
Assessment model	Yes
Assessment procedure	Not available
Assessment instrument	Not available
Improvement suggestions	Not available
Process reference model	No
Maturity structure	Yes
Model representation	Continuous
Character of approach	Qualitative
Structure/components	Key areas
	Maturity levels
	Checkpoints
	Test maturity matrix
	Improvement suggestions
	Dependencies between different levels of the key areas
Addressing	-
Process areas	18 key areas with 6 categories:
	Test process
	Test technique
	Test automation
	Test quality
	Test organization
	Test infrastructure

Table 26: Characteristics of Test SPICE

Characteristics	
Approach	Test SPICE
Reference	[Steiner et al., 2012]
Based on/influenced by	ISO 15504 part 5
Domain	-
Developed by	SQS Group
Status of development	Complete
Completeness of information	Detailed description
Assessment model	Yes
Assessment procedure	Not available
Assessment instrument	Not available
Improvement suggestions	Not available
Process reference model	Yes
Maturity structure	No
Model representation	-
Character of approach	Qualitative
Structure/components	Process categories Process groups Processes
Addressing	-
Process areas	Process categories and groups: Primary life cycle processes Test service acquisition Test service supply Test environment operation Testing Supporting life cycle processes Test process support Organizational life cycle processes Management Resource and infrastructure Process improvement for test Regression and reuse engineering

Table 27: Characteristics of Software Testing Standard ISO/IEC 29119 /ISO 33063

Characteristics	
Approach	Software Testing Standard ISO/IEC 29119 /ISO 33063
Reference	[Reid, 2012]
Based on/influenced by	-
Domain	-
Developed by	ISO/IEC
Status of development	Under development
Completeness of information	Brief description
Assessment model	Yes
Assessment procedure	Not available
Assessment instrument	Not available
Improvement suggestions	Not available
Process reference model	Yes
Maturity structure	No
Model representation	-
Character of approach	Qualitative
Structure/components	Process descriptions Test documentation Test techniques
Addressing	-
Process areas	Test policy Organizational test strategy Test plan Test status report Test completion report Test design specification Test case specification Test procedure specification Test data requirements Test environment requirements Test data readiness report Test environment readiness report Test execution log Incident report

Table 28: Characteristics of Self-Assessment framework for ISO/IEC 29119 based on TIM

Characteristics	
Approach	Self-Assessment framework for ISO/IEC 29119 based on TIM
Reference	[Kasurinen et al., 2011a]
Based on/influenced by	ISO/IEC 29119 TIM
Domain	-
Developed by	Supported by the ESPA-project
Status of development	Complete Validated in pilot study with pre-existing data (four different case organizations)
Completeness of information	Detailed description
Assessment model	Yes
Assessment procedure	Available
Assessment instrument	Available Open questions
Improvement suggestions	Not available (only individual examples from the case study)
Process reference model	Yes
Maturity structure	Yes 0: Initial 1: Baseline 2: Cost-effectiveness 3: Risk-lowering 4: Optimization
Model representation	Continuous
Character of approach	Qualitative
Structure/components	Processes Maturity levels
Addressing	Software designer Software architect Manager Test manager Project leader Tester
Process areas	Organizational test process (OTP) Test management process (TMP) Test planning process (TPP) Test monitoring and control process (TMCP) Test completion process (TCP) Static test process (STP) Dynamic test process (DTP)

Table 29: Characteristics of Meta-Measurement approach

Characteristics	
Approach	Meta-Measurement approach
Reference	[Farooq et al., 2008b]
Based on/influenced by	Evaluation Theory
Domain	-
Developed by	-
Status of development	Under development
Completeness of information	Concept
Assessment model	Yes
Assessment procedure	Not available
Assessment instrument	Not available
Improvement suggestions	Not available
Process reference model	No
Maturity structure	No
Model representation	-
Character of approach	Quantitative
Structure/components	Target Evaluation criteria Reference standard Assessment techniques Synthesis techniques Evaluation process
Addressing	-
Process areas	Activities Product (document, test cases, etc.) Resource (software, hardware, personnel) Roles

Table 30: Characteristics of PDCA-based software testing improvement framework

Characteristics	
Approach	PDCA-based software testing improvement framework
Reference	[Xu-Xiang and Wen-Ning, 2010]
Based on/influenced by	PDCA
Domain	Third party testing center
Developed by	-
Status of development	Complete (thesis work)
Completeness of information	Brief description
Assessment model	No
Assessment procedure	Not available
Assessment instrument	Not available
Improvement suggestions	Not available
Process reference model	No
Maturity structure	No
Model representation	-
Character of approach	Unknown
Structure/components	Test improvement framework divided into phases: Plan Do Check Action
Addressing	-
Process areas	-

Table 31: Characteristics of Evidence-based Software Engineering

Characteristics	
Approach	Evidence-based Software Engineering
Reference	[Kasoju et al., 2013]
Based on/influenced by	Evidence-based Software Engineering
Domain	Automotive software (applied in this domain, but not necessarily limited to it)
Developed by	-
Status of development	Complete
Completeness of information	Brief description
Assessment model	No
Assessment procedure	Not available
Assessment instrument	Not available
Improvement suggestions	Not available (only individual examples from the case study)
Process reference model	No
Maturity structure	No
Model representation	-
Character of approach	Qualitative
Structure/components	Multi-staged evidence-based software engineering research process Case study with interviews to identify strengths and weaknesses of the testing process Domain specific literature review/mapping to find solutions to identified problems Value stream mapping identify process wastes, show locations of improvements
Addressing	-
Process areas	-

Table 32: Characteristics of Observing Practice

Characteristics	
Approach	Observing Practice
Reference	[Taipale and Smolander, 2006]
Based on/influenced by	-
Domain	Software products and applications of an advanced technical level, mission critical, real-time-environments (applied in this domain, but not necessarily limited to it)
Developed by	Supported by the ANTI-project
Status of development	Complete Factors affecting testing know-how and organizations have not been addressed yet Validated in a case study with 4 organizational units
Completeness of information	Detailed description
Assessment model	No
Assessment procedure	Not available
Assessment instrument	Available structured and semi-structured questions, 4 theme-based interview rounds
Improvement suggestions	Not available (only individual examples from the case study)
Process reference model	No
Maturity structure	No
Model representation	-
Character of approach	Qualitative
Structure/components	Interviews Grounded theory to analyze data Classify data into categories Illustrate interdependencies of the categories with cause-effect graphs Process improvement propositions
Addressing	Managers of development Managers of testing Testers System analyst
Process areas	Factors affecting testing, for example: Involvement of testing in the development process Management of the complexity of testing Risk-based testing Communication and interaction between development and testing Use and testing of software components Adjusting testing according to the business orientation of an organizations unit Factors affecting testing know-how and organization Categories derived from data analysis: Involvement of testing in the development process Testing schedules Communication and interaction between development and testing Planning of testing Use of software components Complexity of testing

Table 33: Characteristics of MTPF

Characteristics	
Approach	MTPF - Minimal test practice framework
Reference	[Karlström et al., 2005]
Based on/influenced by	-
Domain	-
Developed by	-
Status of development	Complete
	Validated in a case study and a survey
Completeness of information	Brief description
Assessment model	No
Assessment procedure	Not available
Assessment instrument	Not available
Improvement suggestions	Not available
Process reference model	No
Maturity structure	No
Model representation	-
Character of approach	Qualitative
Structure/components	3 phases depending on the size of the organizational unit
	Introduction phase consisting of 5 steps: prepare, introduce, review, perform, evaluate
Addressing	-
Process areas	Problem and experience reporting
	Roles and organization issues
	Verification and validation
	Test administration
	Test planning