Whitepaper
Collection

Volume

3

Improvement Of Test Processes Using TPI®

An Overview

Published by Senior Consulting
Strategical, Conceptual and Technical Consulting

Alexander Gola
Gröbenzeller Strasse 37
82178 Puchheim, Germany
phone +49 89 84050934
mobile +49 172 5473831
e-mail alex@alexander-gola.de
web www.alexander-gola.de

The information contained in this document represents the current view of the author of the date of publication. The author cannot guarantee the accuracy of any information presented after the date of publication.
This document is for informational purposes only. The author makes no warranties, express or implied, as to the information in this document. Without limiting the rights under copyright, no part of this document may be reproduced, stored in or introduced into a retrieval system or transmitted in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) or for any purpose, without the express written permission of the author.

Abstract

Testing as a phase located between programming and implementation is often considered as an expensive and uncontrollable process. Testing takes too much time, costs a lot more than planned and offers insufficient conception of the quality of the test process. Therefore, the quality of the information system and the risks for the business can be difficult to determine.

This paper describes the content and structure of the TPI®-model created by Sogeti Nederland B.V., which gives practical guidelines for assessing the maturity level of testing in an organization. The paper includes a general description of the model, which deals with how to implement and how to consolidate the improvements, explanations of the scope, characteristics and key areas.

Modification History

Version	Date	Author	Description of Changes
0.1	10.09.2003	Alexander Gola	Initial Release
1.0	19.10.2003	Alexander Gola	First Release
1.1	21.02.2004	Alexander Gola	New Layout, Figure and Table List, CMM process changes
1.2	21.07.2004	Alexander Gola	Enhancements
1.3	03.01.2005	Alexander Gola	Layout changes, Enhancements
1.4	12.01.2005	Alexander Gola	Changes, Enhancements
1.5	18.01.2005	Alexander Gola	Enhancements, Chapter “Trademarks”
1.6	27.07.2005	Alexander Gola	Changes
1.7	18.10.2005	Alexander Gola	Publishing informations
1.8	26.01.2007	Alexander Gola	Changed company address

Table Of Contents

List Of Figures

List Of Tables

1 Test Process Improvement Model

During recent years software process improvement has become an important part of the IT industry. Software environments are dominated by business process reengineering, client/server, object methods, internet/intranet, virtual offices and teams, new database technologies, downsizing, reorganization and most of all, a fierce focus on competitiveness and return on investment. Most of the rules that governed projects only a few years ago cannot be effectively applied in existing software development environments. Despite these facts that testing often accounts for 30–40 % of the total project costs, only limited attention is given to testing.

The Test Process Improvement Model (TPI®) offers a framework to determine the weak and strong areas of a testing process in an organization. The model contains a maturity model to assess a process, a test maturity matrix, checklists and improvement suggestions and activities.

1.1 History

The TPI® model has been developed by Tim Koomen and Martin Pol in 1997 based on the practical knowledge and experiences of test process development. One of the main reasons for developing the model was that testing is to be considered as an important, but difficult and uncontrolled process. The model could be a guideline to make improvement of the testing process easier.

1.2 Philosophy and Concepts

The testing activity in an information system development can be defined as follows:

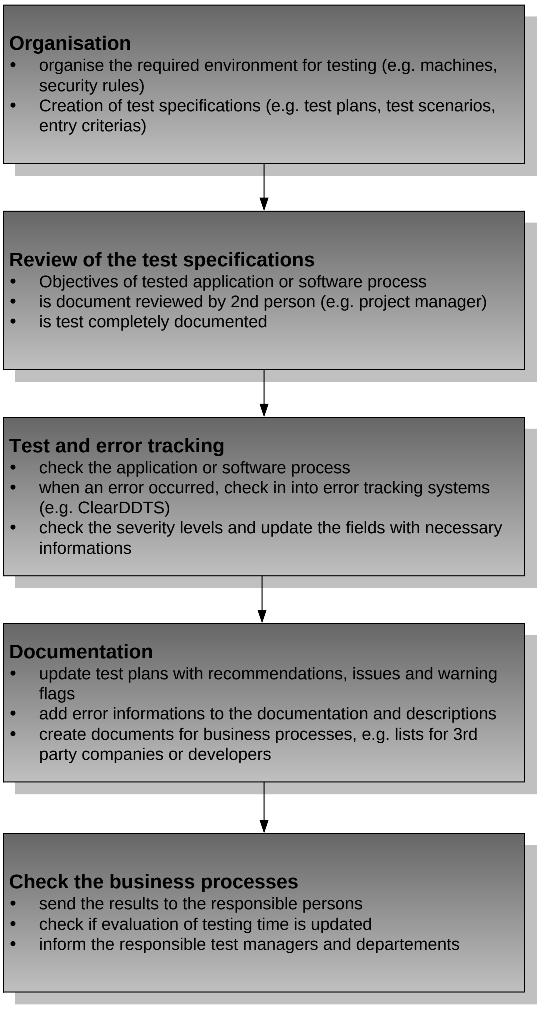
Testing is a process of planning, preparing, executing and analyzing, aimed at establishing the characteristics of an information system and demonstrating the difference between the actual status and the required status.

Test planning and preparation activities emphasize the fact that testing should not be regarded as a process that can be started when the object to be tested is delivered. A test process requires accurate planning and preparation phases before any measurement actions can be implemented.

Testing reduces the level of uncertainty about the quality of a system. The level of testing effort depends on the risks involved in bringing this system in to operation and on the decision of how much time and money is to be spent on reducing the level of uncertainty.

1.2.1 Organization Life-Cycle

The actual situation of a test organization is described in the following figure.



Organisation
- organise the required environment for testing (e.g. machines, security rules)
- Creation of test specifications (e.g. test plans, test scenarios, entry criterias)

Review of the test specifications
- Objectives of tested application or software process
- is document reviewed by 2nd person (e.g. project manager)
- is test completely documented

Test and error tracking
- check the application or software process
- when an error occurred, check in into error tracking systems (e.g. ClearDDTS)
- check the severity levels and update the fields with necessary informations

Documentation
- update test plans with recommendations, issues and warning flags
- add error informations to the documentation and descriptions
- create documents for business processes, e.g. lists for 3rd party companies or developers

Check the business processes
- send the results to the responsible persons
- check if evaluation of testing time is updated
- inform the responsible test managers and departements

Figure 1 Organization Life-Cycle

1.2.2 Software Life-Cycle

The actual situation of a software life-cycle is described in the following figure.

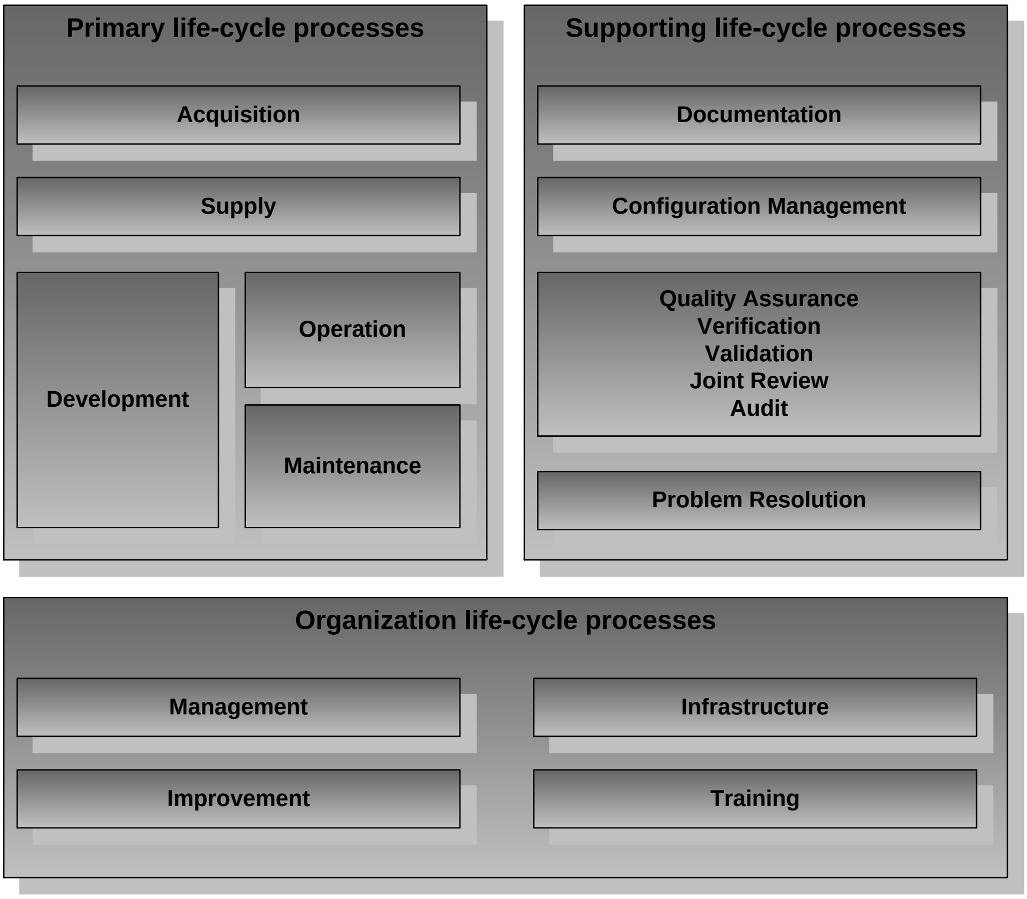


Figure 2 Software Life-Cycle

1.3 Improvement Processes

As an answer to uncoordinated situations in many organizations, different test communities has created its complementary improvement models, such as the Test Process Improvement Model (TPI®), the Capability Maturity Model® (CMM) or SPICE (ISO 15504). These different models offers insight in the “maturity” of the test process within an organization and helps to define gradual and controllable improvement steps.

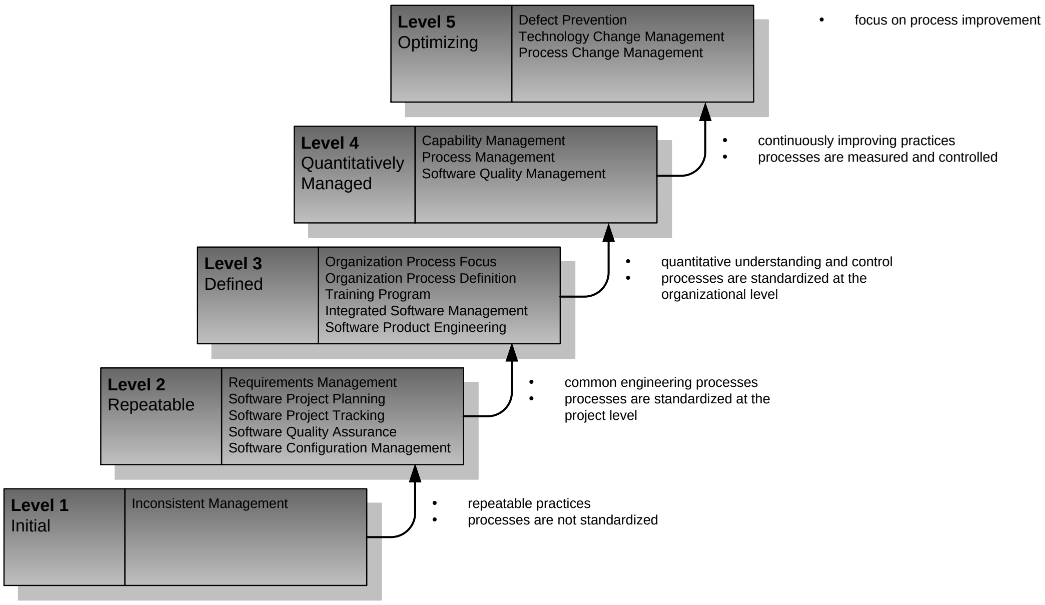

Level 5
Optimizing
Defect Prevention
Technology Change Management
Process Change Management
• focus on process improvement
Level 4
Quantitatively Managed
Capability Management
Process Management
Software Quality Management
• continuously improving practices
• processes are measured and controlled
Level 3
Defined
Organization Process Focus
Organization Process Definition
Training Program
Integrated Software Management
Software Product Engineering
• quantitative understanding and control
• processes are standardized at the organizational level
Level 2
Repeatable
Requirements Management
Software Project Planning
Software Project Tracking
Software Quality Assurance
Software Configuration Management
• common engineering processes
• processes are standardized at the project level
Level 1
Initial
Inconsistent Management
• repeatable practices
• processes are not standardized

Figure 3 Capability Maturity Model®

In the figure you can see, that the main target is to shift the software testing ad hoc to systematic, integrated software testing. It details a process that produces measurable results that can be incrementally improved. Testing is handled as an integral part of the development process (resulting in testware), using the same systems engineering discipline as software development.

2 Scope of Test Process Improvement

The scope of test process improvement usually comprises high-level tests like system and acceptance tests. To improve more test processes, attention must also be given to verification activities and low-level tests like unit and integration tests.
To do this, a company must provide a essential framework for successful test management. It focuses on two key areas: the development and management of a successful testing organization and team and the development of an effective test strategy.
The TPI® model based on these key areas and defines 20 key areas. Separate key areas are included in order to give due attention to these processes as well.

The model is visualized as follows.

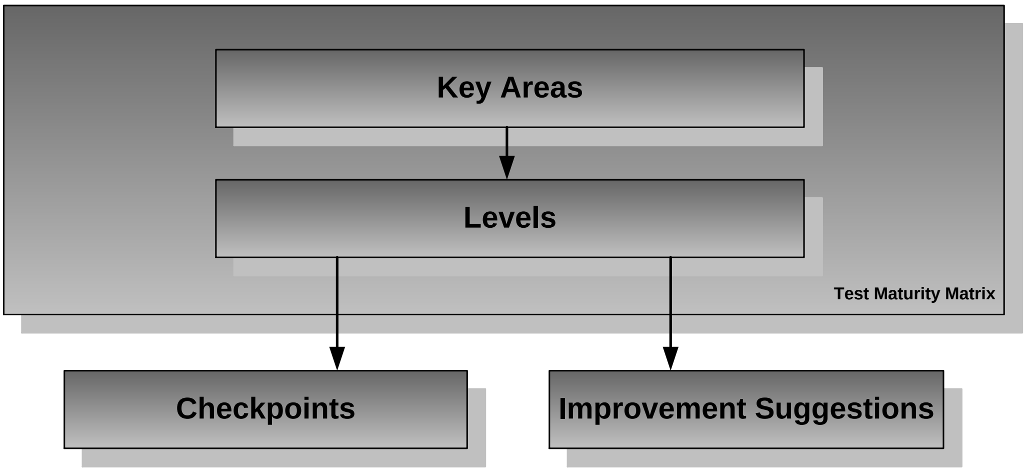


Figure 4 TPI® Model

2.1 Key Areas

By looking at the different aspects under a structured test process, 20 key areas can be recognized for the TPI® model and has from 1 to up 4 levels. These key areas cover the total test process. Most of the areas are related to the improvement of system and acceptation tests. For more mature testing processes, there are also areas related to white-box testing types.

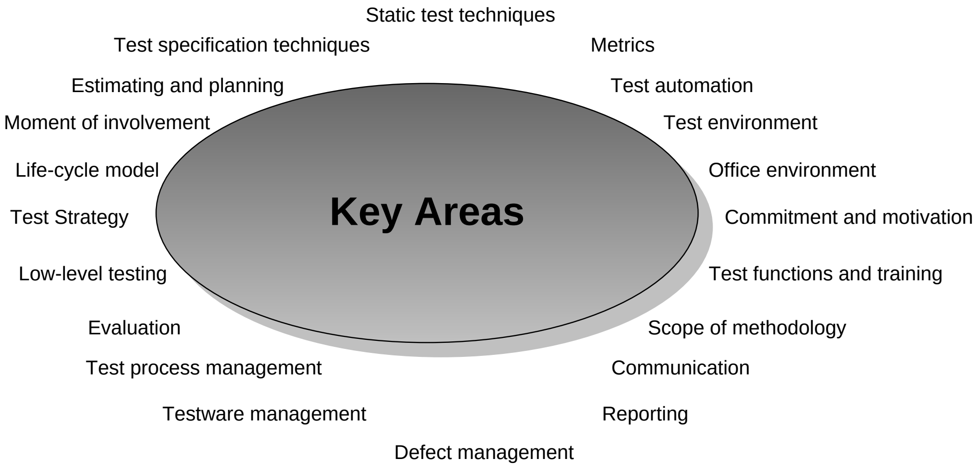


Figure 5 Key Areas of the TPI® Model

For a description overview see the spreadsheet "TPI Key Areas".

2.2 Levels

In order to enable insight in the state of the key areas, the TPI® model supplies them with ascending levels (generally from A to D). On the average, there are three levels for each key area. It is obvious that not each key area will be addressed equally thoroughly: each test process has its strengths and weaknesses.

Each higher level is better than its prior level in terms of time (faster), money (cheaper) and/or quality (better). By using levels we can unambiguously assess the current situation of the test process. It also increases the ability to advice targets for stepwise improvement.

Each level consists of certain requirements for the key area. These checkpoints of a certain level also comprise the requirements of lower levels: a test process at level B fulfils the requirements of both level A and B. If a test process does not satisfy the requirements for level A, it is considered to be at the lowest and, consequently, undefined level for that particular key area.

Levels Key Areas	A	B	C	D
Test Strategy	Strategy for single high-level test	Combined strategy for high-level tests	Combined strategy for high-level tests plus low-level tests or evaluation	Combined strategy for all test and evaluation levels
Life-Cycle Model	Planning, Specification, Execution	Planning, Preparation, Specification, Execution, Completion		
Moment of Involvement	Completion of test basis	Start of test basis	Start of requirements definition	Project initiation
Estimating and Planning	Substantiated estimating and planning	Statistically substantiated estimating and planning		
Test Specification Techniques	Informal techniques	Formal techniques		
Static Test Techniques	Inspection of test basis	Checklists		
Metrics	Project metrics (product)	Project metrics (process)	System metrics	Organization metrics (>1 system)
Test Automation	Use of tools	Managed test automation	Optimal test automation	
Test Environment	Managed and controlled environment	Testing in most suitable environment	Environment on call	
Office Environment	Adequate and timely office environment			
Commitment and Motivation	Assignment of budget and time	Testing integrated in project	Test-engineering	

		organization		
Test Functions and Training	Test manager and testers	Formal methodical, technical and functional support, management	Formal internal Quality Assurance	
Scope of Methodology	Project specific	Organization generic	Organization optimizing	
Communication	Internal communication	Project communication (defects, change control)	Communication within the organization about the quality of the test processes	
Reporting	Defects	Progress (status of tests and products), activities (costs and time, milestones), defects with priorities	Risks and recommendations, substantiated with metrics	Recommendations have a Software Process Improvement character
Defect Management	Internal defect management	Extended defect management with flexible reporting facilities	Project defect management	
Testware Management	Internal testware management	External management of test basis and test object	Reusable testware	Traceability system requirements to test cases
Test Process Management	Planning and execution	Planning, execution, monitoring, and adjusting	Monitoring and adjusting within organization	
Evaluation	Evaluation techniques	Evaluation strategy		
Low-level Testing	Low-level test life-cycle: planning, specification and execution	White-box techniques	Low-level test strategy	

Table 1 List of Levels

For a description overview example see the spreadsheet “TPI Levels”.

2.3 Checkpoints

In order to determine the requirements of certain levels, the checkpoints are used. The requirements are defined in the form of questions that need to be answered positively in order to reach certain level. Based on the checkpoints a test process can be assessed and for each key area the proper level can be established. As each next level of a key area is considered an improvement, this means that the checkpoints are cumulative: in order to classify for level B the test process needs to answer positively to the checkpoints both of level B and of level A.

2.4 Test Maturity Matrix

The main purpose of the matrix is to show the strong and weak sides of the current test process and to support prioritizing actions for improvement. A filled in matrix offers all participants a clear view of the current situation of the test process. Furthermore, the matrix helps in defining and selecting proposals for improvement.

All levels and key areas are related to each other in a maturity matrix. This has been done as a good way to express the internal priorities and dependencies between levels and key areas. The vertical axis of the matrix indicates key areas, the horizontal axis shows scales of maturity. In the matrix each level is related to a certain scale of test maturity. These 13 scales can be divided into three categories:

- Controlled: Scales 1 to 5 are mainly for the control of the test process. The test process is carried out in phases according to a strategy defined in advance. Test specification techniques are used for testing, and defects are recorded and reported. The testware and test environment are well controlled and the test staff are sufficiently trained.
- Efficient: The levels in scales 6 to 10 aim more on the efficiency of the test process. The efficiency can be achieved e.g. by automating the test process, by better integration between the mutual test process and with the other parties within the system development.
- Optimizing: An efficient test process in the current situation may not be an efficient one in the future. The levels in scales 11 to 13 are for increasing optimization of the test process and they focus more on ensuring that continuous improvement of the test process will be part of the working methods of the organization.

After determining the levels for each key area, attention should be directed as to which improvement steps to take. This is because not all key areas and levels are equally important.

Scale Key Area	0	1	2	3	4	5	6	7	8	9	10	11	12	13
Test Strategy		A					B				C		D	
Life-Cycle Model		A			B									
Moment of Involvement			A				B				C		D	
Estimating and Planning				A							B			
Test Specification Techniques		A		B										
Static Test Techniques					A		B							
Metrics						A			B		C			D
Test Automation				A				B			C			
Test Environment				A				B						C
Office Environment				A										
Commitment and Motivation		A				B						C		
Test Functions and Training				A			B			C				
Scope of Methodology					A						B			C
Communication			A		B							C		
Reporting		A			B		C					D		
Defect Management		A				B		C						
Testware Management			A			B				C				D
Test Process Management		A		B								C		
Evaluation							A			B				
Low-level Testing					A		B		C					

Table 2 Test Maturity Matrix

The matrix works from left to right, so low mature key areas are improved first. As a consequence of the dependencies between levels and key areas, practice has taught us that real “outlyers” (e.g. key areas with high scales of maturity, whereas surrounding key areas have medium or low scales) give little return on investment. Without violating the model, deviation is permitted, but sound reasons should exist for it.

For a matrix overview example see the spreadsheet “TPI Test Maturity Matrix”.

3 Test Areas

Test areas are based on checkpoints for each level.

3.1 Test Strategy

The test strategy has to be focused on detecting the most important defects as early and as cheaply as possible. The test strategy defines which requirements and (quality) risks are covered by what tests. The better each test level defines its own strategy and the more the different test level strategies are adjusted to each other, the higher the quality of the overall test strategy.

3.2 Life-Cycle Model

Within the test process a number of phases can be defined, such as planning, preparation, specification, execution and completion. In each phase several activities are performed. For each activity the following aspects should be defined: purpose, input, process, output, dependencies, applicable techniques and tools, required facilities, documentation, etc. The importance of using a life-cycle model is an improved predictability and controllability of the test process, because the different activities can be planned and monitored in mutual cohesion.

3.3 Moment of Involvement

Although the actual execution of the test normally begins after the realization of the software, the test process must and can start much earlier. An earlier involvement of testing in the system development path helps to find defects as soon and easy as possible and perhaps even to prevent errors. A better adjustment between the different tests can be done and the time that testing is on the critical path of the project can be kept as short as possible.

3.4 Estimating and Planning

Test planning and estimating indicate which activities have to be carried out when, and the necessary resources (people). Good estimating and planning are very important, because they are the basis of, for example, allocating resources for a certain time frame.

3.5 Test Specification Techniques

The definition of a test specification technique is "a standardized way of deriving test cases from source information". Applying these techniques gives insight into the quality and depth of the tests and increases the reusability of the test.

3.6 Static Test Techniques

Not everything can and should be tested dynamically, that is, by running programs. Inspection of products without running programs, or the evaluation of measures which must lead to a certain quality level, is called static tests. Checklists are very useful for this.

3.7 Metrics

Metrics are quantified observations of the characteristics of a product or process. For the test process, metrics of the progress of the process and the quality of the tested system are very important. They are used to control the test process, to substantiate the test advice and also to make it possible to compare systems or processes. Why has one system far fewer failures in operation than another system, or why is one test process faster and more thorough than another? Specifically for improving the test process, metrics are important by evaluating consequences of certain improvement actions, by comparing data before and after performing the action.

3.8 Test Automation

Automation within the test process can take place in many ways and has in general one or more of the following aims:

- fewer hours needed,
- shorter lead time,
- more test depth,
- increased test flexibility,
- more and/or faster insight in test process status,
- better motivation of the testers.

3.9 Test Environment

The test execution takes place in a so-called test environment. This environment mainly comprises the following components:

- hardware
- software
- means of communication
- facilities for building and using databases and files
- procedures

The environment should be composed and set up in such a way that by means of the test results it can be optimally determined to what extent the test object meets the requirements. The environment has a large influence on the quality, lead time, and cost of the test process. Important aspects of the environment are responsibilities, management, on-time and sufficient availability, representativeness and flexibility.

3.10 Office Environment

The test staff needs rooms, desks, chairs, PCs, word-processing facilities, printers, telephones, and so on. A good and timely organization of the office environment has a positive influence on the motivation of the test staff, on communication in- and outside the team, and on the efficiency of the work.

3.11 Commitment and Motivation

The commitment and the motivation of the persons involved in testing are important prerequisites for a smoothly running test process. The persons involved are not only the testers, but also, for example, the project management and the line management personnel. The latter are mainly important in the sense of creating good conditions. The test process thus receives enough time, money, and resources (quantitatively and qualitatively) to perform a good test, in which cooperation and good communication with the rest of the project results in a total process with optimum efficiency.

3.12 Test Functions and Training

In a test process the correct composition of a test team is very important. A mix of different disciplines, functions, knowledge, and skills is required. Besides specific test expertise, knowledge of the subject matter, knowledge of the organization and general IT knowledge is required. Social skills are also important. For acquiring this mix, training etc. is required.

3.13 Scope of Methodology

For each test process in the organization a certain methodology or working method is used, comprising activities, procedures, regulations, techniques etc. When these methodologies are different each time or when the methodology is so generic that many parts have to be drawn up again each time, it has a negative effect on the test process efficiency. The aim is that the organization uses a methodology which is sufficiently generic to be applicable in every situation, but which contains enough detail so that it is not necessary to rethink the same items again each time.

3.14 Communication

In a test process, communication with the people involved must take place in several ways, within the test team as well as with parties such as the developer, the user, the customer, etc. These communication forms are important for a smoothly running test process, not only to create good conditions and to optimize the test strategy, but also to communicate about the progress and the quality.

3.15 Reporting

Testing is not so much "defect detection" as about giving insight in the quality level of the product. Reporting should be aimed at giving well-founded advice to the customer concerning the product and even the system development process.

3.16 Defect Management

Although managing defects is in fact a project matter and not specifically of the testers, the testers are mainly involved in it. Good management should be able to track the life-cycle of a defect and also to support the analysis of quality trends in the detected defects. Such analysis is used, for example, to give well-founded quality advice.

3.17 Testware Management

The products of testing should be maintainable and reusable and so they must be managed. Besides the products of the testing themselves, such as test plans, specifications, databases and files, it is important that the products of previous processes such as functional design and realization are managed well, because the test process can be disrupted if the wrong program versions, etc. are delivered. If testers make demands upon version management of these products, a positive influence is exerted and the testability of the product is increased.

3.18 Test Process Management

For managing each process and activity, the four steps from the Deming circle are essential: plan, do, check and act. Process management is of vital importance for the realization of an optimal test in an often turbulent test process.

3.19 Evaluation

Evaluation means inspecting intermediate products such as the requirements and the functional design. The importance of evaluation is that the defects are found at a much earlier stage in the development process than with testing. This makes the rework costs much lower. Also, evaluation can be set up more easily because there is no need to run programs or to set up an environment etc.

3.20 Low-level Testing

The low-level tests are almost exclusively carried out by the developers. Well-known low-level tests are the unit test and the integration test. Just as evaluation, the tests find defects at an earlier stage of the system development path than the high-level tests. Low-level testing is efficient, because it requires little communication and because often the finder is both the error producer as well as the one who corrects the defect.

For a complete level, area and checkpoint overview for each test area see the spreadsheet “TPI Test Areas”.

4 Process of Change

The process of test improvement is similar to any other improvement process.

The figure below shows the various activities within this process.

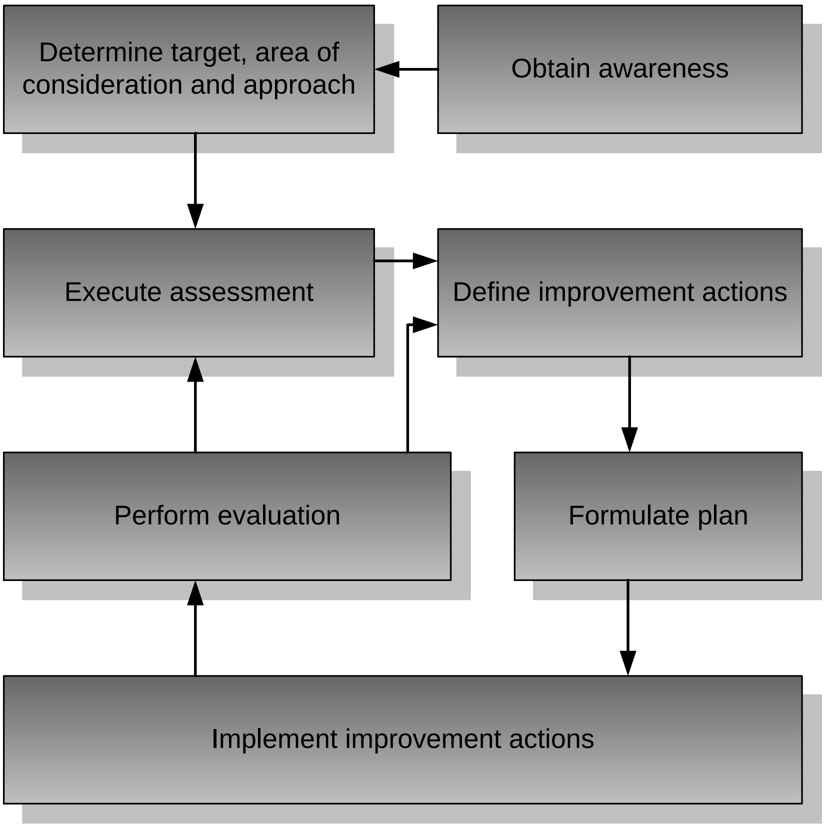

Determine target, area of consideration and approach
Obtain awareness
Execute assessment
Define improvement actions
Perform evaluation
Formulate plan
Implement improvement actions

Figure 6 Process of Change

4.1 Obtain Awareness

The first activity of a test improvement process is to create awareness for the necessity to improve the process. Generally speaking, a number of problems concerning testing is the reason for improving the test process. There is a need to solve these problems and an improvement of the test process is regarded as the solution. This awareness also implies that the parties mutually agree on the outlines and give their commitment to the change process.
Commitment should not only be acquired at the beginning of the change process, but be retained throughout the project. This requires a continuous effort.

4.2 Determine Target, Area of Consideration and Approach

It must be determine what the improvement targets are and what the area of consideration is. Should testing be faster, cheaper or better ? Which test processes are subjects for improvement, how much time is available for the improvement and how much effort is it allowed to cost ?

4.3 Execute Assessment

In the assessment activity, an evaluation is given of the current situation. The use of the TPI® model is an important part of the assessment, because it offers a frame of reference to list the strong and weak points of the test process. Based on interviews and documentation, the levels per key area of the TPI® model are examined by using checkpoints, and it is determined which checkpoints were met, which were not met, or only partially. The Test Maturity Matrix is used here to give the complete status overview of the test process. This will show the strengths and weaknesses of the test process in the form of levels assigned key areas and their relative position in the matrix.

4.4 Define Improvement Actions

The improvement actions are determined based on the improvement targets and the result of the assessment. These actions are determined in such a way that gradual and step by step improvement is possible.
The TPI® model helps to set up these improvement actions. The levels of the key areas and the Test Maturity Matrix give several possibilities to define gradual improvement steps. Depending on the targets, the scope, the available time and the assessment results, it can be decided to carry out improvements for one or more key areas. For each selected key area it can be decided to go to the next level or, in special cases, even to a higher level. Besides this, the TPI® model offers a large number of improvement suggestions which help to achieve higher levels.

4.5 Formulate Plan

A detailed plan is drawn up to implement all or a part of the short term improvement actions. In this plan the aims are recorded and it is indicated which improvements have to be implemented at what time to realize these aims. The plan deals with activities concerning the content of the test process improvement as well as general activities needed to steer the change process in the right direction.

4.6 Implement Improvement Actions

The plan is executed. Because during this activity the consequences of the change process have the largest impact, much attention should be spent on communication. Opposition, which no doubt is present, must be brought to the surface and be discussed openly. It has to be measured to what extent actions have been executed and have been successful. A means for this is the so-called "self assessment", in which the TPI® model is applied in order to quickly determine the progress. Here, the persons involved inspect their own test processes using the TPI® model.
Another vital part of this phase is consolidation. It should be prevented that the implemented improvement actions have a once-only character.

4.7 Perform Evaluation

To what extent did the implemented actions yield the intended result? In this phase the aim is to see to what extent the actions were implemented successfully as well as to evaluate to what extent the initial targets were met. A decision about the continuation of the change process is made based on these observations.

5 Improvement Suggestions

Improvement actions can be defined in terms of desired higher levels. For reaching a higher level the checkpoints render much assistance. Beside these, the model has other means of support for test process improvement: The improvement suggestions, which are different kinds of hints and ideas that help to achieve a certain level of test maturity. Unlike the use of checkpoints, the use of improvement suggestions is not obligatory. Each level is supplied with several suggestions.
The target of these suggestions is to give the knowledge and experiences to reach a desired level in the organization and to define specific improvement steps.

In the first step it is recommended to analyze your business processes. Use the following figure to collect information about the actual process situation.

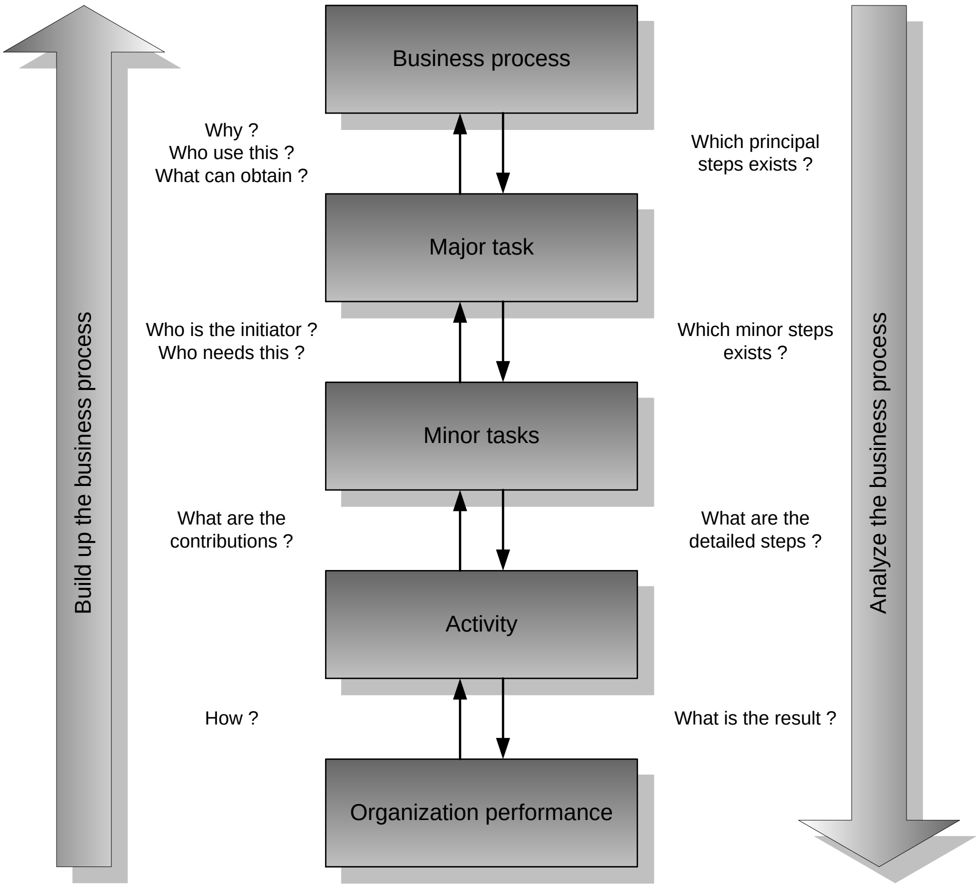

Build up the business process
Business process
Why ?
Who use this ?
What can obtain ?
Which principal steps exists ?
Major task
Who is the initiator ?
Who needs this ?
Which minor steps exists ?
Minor tasks
What are the contributions ?
What are the detailed steps ?
Activity
How ?
What is the result ?
Organization performance
Analyze the business process

Figure 7 Business Processes

After this, use the TPI® tracking tool (see chapter 6) to gather informations in depth and to find solutions.

Examples to improve the test strategy

1.A.1 Testing strategy for single tests	**Checkpoint:** A motivated consideration of the product risks takes place, for which knowledge about the system and knowledge about the usage and the management of the system is essential. **Solution:** the following tasks are required: • if only one technique is available, make simple variations, which give either more or less test depth • define a re-test approach which makes a balance between total or partial testing • divide the system in testable subsystems and set priorities • divide the system in testable quality attributes and set priorities • to shorten the critical path testing time make an incremental testing

Table 3 Testing Strategy for Single Tests

1.B.1 Combined testing strategy for black-box tests	**Checkpoint:** The various black-box test types, often the system test, the acceptance test and the production acceptance test, are adjusted to each other regarding test strategy (risks, quality attributes, scope, and planning). **Solution:** the following tasks are required: • define a position for an independent test coordinator • create a master plan • use of (formal) design techniques • Independent check on completeness and accuracy of testware

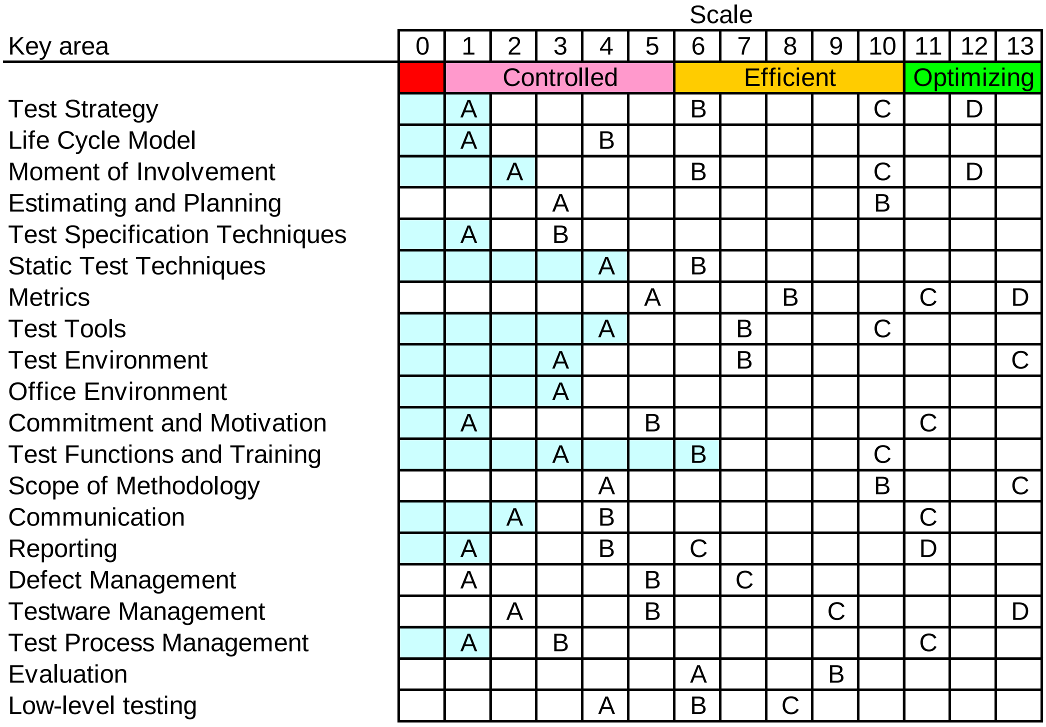
Table 4 Testing Strategy for Black-Box Tests

After these steps and in synchronization with organization goals a fulfillment of the following criteria's are possible:

- Cost and benefit ratio
- Time and resource consumption
- Easy and ongoing actions first
- Degree of acceptance
- Reduce high risks
- Usage and effectiveness of deliverables and products in fulfilling their requirements
- Balance between ratio and feeling

6 TPI® Tracking Utility

The described utility is a tool to measure the maturity of an organization test process and to track progress in test process improvement. To use the utility for evaluation meetings, see the spreadsheet "TPI Tracking Tool".
In the following figure a final evaluation result is shown:



Scale

Key area	0	1	2	3	4	5	6	7	8	9	10	11	12	13
	🟥	Controlled					Efficient					Optimizing		
Test Strategy		A					B				C		D	
Life Cycle Model		A			B									
Moment of Involvement			A				B				C		D	
Estimating and Planning				A							B			
Test Specification Techniques		A		B										
Static Test Techniques					A		B							
Metrics						A			B			C		D
Test Tools					A			B			C			
Test Environment				A				B						C
Office Environment				A										
Commitment and Motivation		A				B						C		
Test Functions and Training				A			B				C			
Scope of Methodology					A						B			C
Communication			A		B							C		
Reporting		A			B		C					D		
Defect Management		A				B		C						
Testware Management			A			B				C				D
Test Process Management		A		B								C		
Evaluation							A			B				
Low-level testing					A		B		C					

Figure 8 TPI® Tracking Utility

It is recommended, that at least the first use of the utility be guided by an external TPI® consultant. To allow for quick feedback loops it is recommended to conduct the evaluation every 3 or 4 months.

Before the meeting, the moderator

- selects a number of participants (typically 4-6). Especially test engineers should participate but it might be interesting to include also other development roles (project leader, developer, architect, integrator, quality assurance, configuration management) and interfacing roles (engineers from other disciplines, product managers, marketers). Ensure that these participants can represent different projects.
- sends invitations to the participants

During the meeting

- If participants are not familiar with TPI® the moderator should give a short introduction.
- Agree on TPI® terms that need interpretation (e.g. test depth, high-level and low-level test, test manager, test basis)
- The moderator should explain the scoring. The following scores are possible:
  - 0-1: this practice is not required and is (almost) never done
  - 2-3: this practice is sometimes required or is sometimes done
  - 4-5: this practice is required but not always done, or the practice is regularly performed although is not required or checked
  - 6-7: this practice is normally required and usually done
  - 8-9: this practice is required, is done and it is checked (the practice is institutionalized)
  - 10: this practice is institutionalized and is a world class example
  - ?: if the participant doesn't know the answer
  - na: if the practice is not applicable

  Remember that the questions are applicable in almost all cases. So, the moderator should check if "n/a" scores are acceptable. Finally, lots of "?" often point to a areas where more information is needed by the engineers.
- The questionnaires are handed out to the participants
- Participants fill in a questionnaire and hand it over to the moderator as soon as it is completed
- The moderator enters the results in the evaluation spreadsheet
- Large deviations in opinions are discussed to ensure that all participants have the same understanding about the statement with that large deviation. A participant may request to change his or her score after such an explanation. It is important that the moderator has in depth knowledge of TPI® and is able to interpret it correctly for the given department.

After the meeting the moderator

- generate the evaluation report
- adds recommendations for process improvement based on the evaluation report
- distributes the evaluation report and recommendations

When all participants are very knowledgeable in TPI® the moderator might send out the questionnaires together with the invitation. The participants complete the questionnaires and send them back to the moderator. The moderator fills in the evaluation spreadsheet. During the meeting only the large deviations are discussed possibly resulting in a few changes in the scores.

Links

Sogeti provides information technology services to businesses and public sector organizations. The Sogeti Group has operations throughout the U.S. and Europe, and is a wholly owned subsidiary of the international Cap Gemini SA organization, one of the worlds leading providers of consulting, technology and outsourcing services. Sogeti Group companies operate in a decentralized and entrepreneurial way, while at the same time joining forces as an informal international network with a presence in U.S. locations and throughout Europe.

Homepage: http://www.sogeti.com

References

Books:

Test Process Improvement: A practical step-by-step guide to structured testing, Tim Koomen and Martin Pol, Addison-Wesley (1999), ISBN 0-201-59624-5

Whitepapers:

Quality Abbreviations and Acronyms, Certitude GmbH, Munich, December 2004

TPI – a model for Test Process Improvement, Jari Andersin, University of Helsinki, October 2004

A Comparison of TMM and other Test Process Improvement Models, Ron Swinkels, November 2000

Articles:

Trademarks and other Acknowledgements

“CMM” and “Capability Maturity Model” are registered in the US Patent and Trademark Office.

“CMM Integration”, “CMMI” and “SCAMPI” are service marks of Carnegie Mellon University, Pittsburgh USA.

“TPI” is a registered trademark of Sogeti USA LLC.