

Survey 2011: »Software Test in Practice«

Peter Haberl, Andreas Spillner, Karin Vosseberg, Mario Winter

German
Testing Board

www.german-testing-board.info

Translation funded by
German Testing Board e.V.

Survey 2011: »Software Test in Practice«

Performed in cooperation with

University of Applied Sciences Bremen
University of Applied Sciences Bremerhaven
University of Applied Sciences Köln
ANECON Software Design & Beratung G.m.b.H.
German Testing Board e.V.
Swiss Testing Board

Supported by

ASQF e.V., Austrian Testing Board, dpunkt Verlag GmbH,
Softwareforen Leipzig GmbH, GI e.V. Working Group „Testing,
Analyse and Verification“



dpunkt.verlag

Initial Situation

In recent years, there have been many new trends in the areas of quality assurance and testing, e.g. test-driven development, exploratory testing, model-based testing or also agile methodologies. These trends offer both opportunities and challenges in testing and quality assurance. Which of the new trends have already found their way into day-to-day testing? Has anything changed in recent years? What does the day-to-day testing in organizations look like today? These and further questions were the basis for the Survey 2011 “Software Test in Practice” [Sur11].

In May 2011, an anonymous online survey on the subject of “Software Test in Practice” was conducted in the German-speaking area in cooperation of Bremen, Bremerhaven and Cologne Universities of Applied Sciences, ANECON Software Design & Beratung G.m.b.H., the German Testing Board e.V. (GTB) and the Swiss Testing Board (STB). Some of the questions were taken from a survey from the year 1997 (“Prüf- und Testprozesse in der Softwareentwicklung”, [Mül98]), in order to identify changes and new trends in the area of quality assurance in software development.

The good level of participation reaffirms yet again the topicality and importance of software quality assurance. Different groups of people (developers, testers and managers) were asked about various key aspects; in doing so, the responses allow for different role-specific views on software testing. Also gratifying was the very wide mix of organization sizes and industries, with a slight emphasis on the automotive, telecommunication and banking areas. Both employees from small and medium-sized companies and from large organizations took part in the survey, whereby the participation from medium-sized enterprises (101-1,000 employees) slightly outnumbers the others. The area of research has a relatively low representation (approx. 10 %), which in itself is an indicator for the practical relevance of the study’s results.

The Survey

The survey was conducted online via the open-source tool Ilias [Ilias11] at the Bremen University of Applied Sciences in the form of three role-specific questionnaires. Apart from establishing the changes in relation to the survey conducted in 1997, the objective of the survey was, firstly, to determine the status quo of quality and test activities in practice, and, secondly, to analyze the need for action for research, training and consultation in testing. The following assumptions are to be verified or falsified on the basis of the survey:

1. Has quality awareness increased and is quality assurance integrated into the software development process at an early stage?
2. What influence do agile methodologies have on testing and quality activities?
3. How are testers perceived, and is the value they add to quality assurance recognized?
4. Is test automation in acceptance test, system test and unit test promoted systematically and methodically?
5. Is repetitive testing and simple test execution outsourced to (onshore, nearshore or offshore) companies?

An evaluation of all questions (in German) is available via the Internet under: <http://www.softwaretest-umfrage.de>

Statistics Regarding the Survey Participants

In total, 1,623 persons started the survey and more than half completed the comprehensive questionnaire from start to finish. The contents were broken down into 6-7 blocks of questions with up to 100 questions all in all (depending on role). The high number of responses allows for sound conclusions to be drawn. Very gratifying was the high number of participations from the three addressed groups of people (developers, testers and managers) as well as the great commitment shown by the participants in answering the comprehensive catalogue of questions.

The overwhelming majority of survey participants came from Germany (77%), followed by Switzerland (13%) and Austria (10%). In order to make well-informed statements about the practice of testing, it was necessary to attract as many industries and experienced participants as possible. Both objectives were fully achieved. More than 15 industries were represented: from the public sector to light and heavy industries to the service and financial sectors.

Also as far as the size of the participating companies is concerned, a representative sample can be determined. Almost a third of the participants work in companies with 101-1,000 employees, but also small companies (with 11-100 employees) are very well represented, making up a quarter of the total participants. Large and very large organizations participated in the survey with 8% and with 13% respectively.

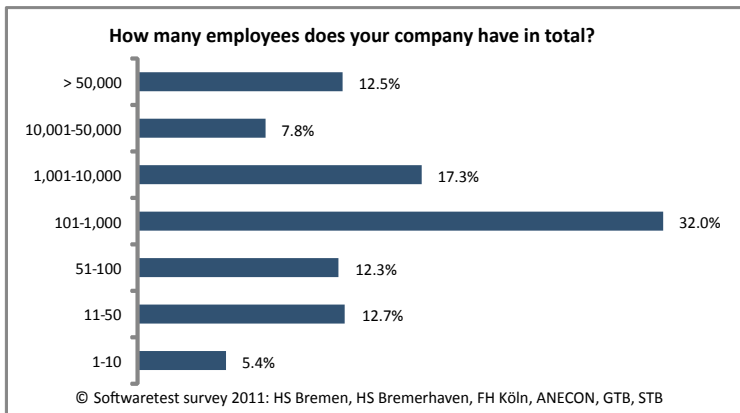


Fig. 1 Company size

With regard to their academic qualification, 70% of survey participants stated that they have successfully completed a diploma study or a bachelor's or master's degree, and more than 8% even hold a doctorate. The professional experience represented in the survey was impressive: A quarter of participants stated that they have 5-10 years of experience in the IT industry, 40% have 10-20 years and a fifth have even more than 20 years of experience. As

expected, the domain is predominantly male; only 14% of the persons that responded were women.

Structure of the Survey

The survey is structured according to specific roles. The following three groups of roles in organizations were compiled for the different questionnaires:

- Project leaders, quality assurance representatives, test managers and testers (group of testers)
- Business analysts, developers, employees from business operations & support, and other employees (group of developers)
- Executive and middle management (group of management)

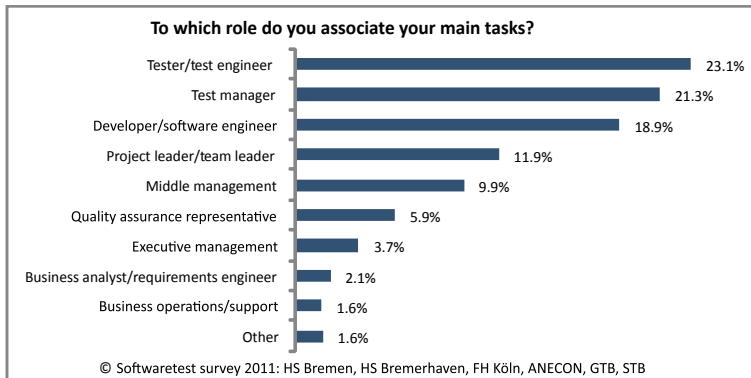


Fig. 2 Roles in the organization

Some questions were put to all three groups, whilst others related to particular roles so that specific aspects could be enquired. Response to questions was optional, which means that it was possible to continue with the next question without answering a question. Equally distributed across all groups, 50% of participants fully completed the very comprehensive questionnaire. Considering the available distribution of individual roles in the or-

ganization, the desired target group of practitioners with different perspectives on quality assurance (QA) could be achieved (see fig. 2) and a sound data basis for the evaluation established. In order to prevent a distortion of the results through different interpretations of technical terms, the terms used were in most cases linked to the definitions in the ISTQB® (International Software Testing Qualifications Board) glossary [ISTQB10].

Risk Management

Almost 70% of participants stated that they perform risk management in their projects. Most frequently mentioned was that risks were assessed several times either as required (38%) or at previously planned important project events (28%). Only 16% of participants specified that risk assessment is performed regularly every month. For weekly intervals the value was only 7%. This allows us to conclude that risks are submitted for new assessment through project events (such as steering committee meetings, phase changes, etc.), or at the request of the project manager or line manager.

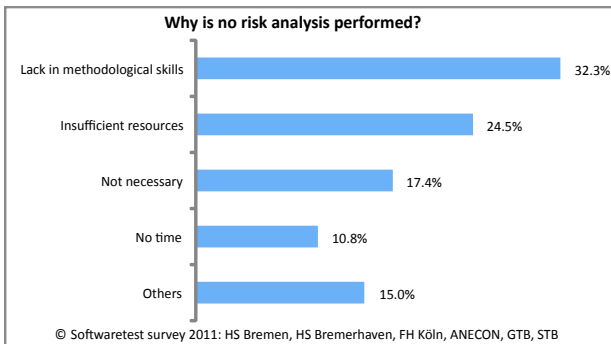


Fig. 3 Reasons against performing a risk analysis

If participants that do not perform risk management are asked for the reason, the most frequent responses are lack in methodological skills and insufficient resources (see fig. 3). Here organizations should invest more into training.

Software Development Methodologies

Since software development has changed increasingly into an engineering discipline, which today is often performed industrially and divided up into distributed teams, the methods and frameworks used have also developed further. The need for efficient and cost-effective software production has also reached software testing. But how is software testing methodically implemented in organizations?

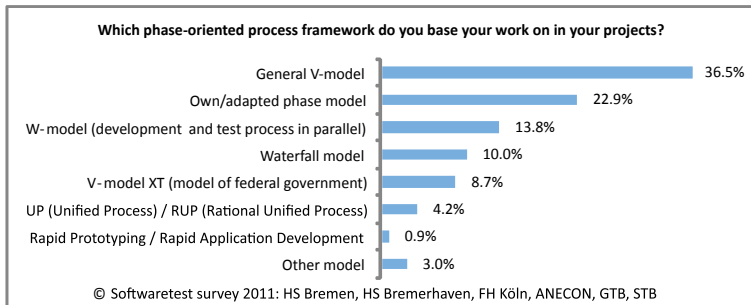


Fig. 4 Traditional software development methodologies

The majority of participants (54%) develop according to a phase-oriented framework. More than a third of these are oriented to the general V-model; UP and RUP do not play a big role (see fig. 4). As can be expected, the V-model XT is more frequently used in local authorities, arms production and defense (approx. 12%). In contrast, its usage is almost non-existent (2%) in research.

The results have shown that in environments where phase-oriented frameworks are used, performing quality assurance measures in the early phases has increased in comparison with the 1997 survey, but are still concentrated on the late phases of software development (see fig. 5). On closer inspection, however, this pleasing trend is not yet sufficient, since quality assurance is still not used comprehensively enough in the initial study and concept phase.

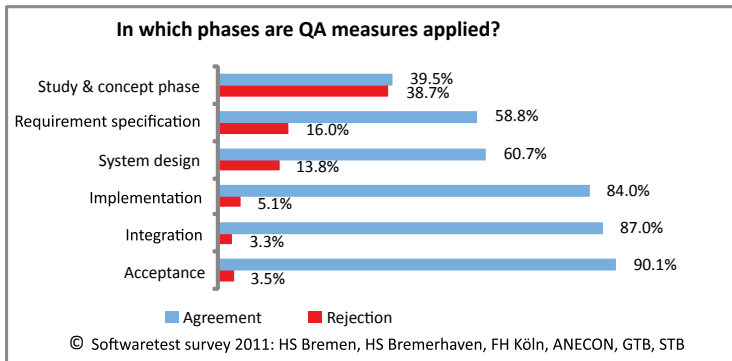


Fig. 5 Quality assurance in phase-oriented methodologies

More than a third of the interviewees agreed that they use QA measures already in the study phase; back in 1997 this was only the case for a quarter. However, the share of those that do not tend to use quality assurance in this phase (39%) is roughly as large as those that do. For the requirement specification phase also, almost 60 % (compared to 45% in 1997) state that they use QA measures. From the system design phase onwards, quality assurance is used across all industries. If you compare the automotive, banking and aerospace industries, however, there are differences: In the aerospace industry, QA measures are much more frequently used in the early phases compared to other industries. In the automotive industry quality assurance is mostly used slightly later, but still earlier than in banking, which is in the middle of the field. All in all, this result is no surprise, and a trend toward the early use of QA measures is apparent.

Test and Quality Assurance in Agile Projects

Agile software development methodologies already represent one quarter of those in the survey, with Scrum evidently the clear favorite with 57%. Kanban and XP only take up a small share, but a quarter of users stated that they apply their own agile frameworks (see fig. 6). Whilst just over half of sur-

vey participants still work with a sequential phase model, the surprising fact is the high share of 17% that do not use any explicit methodologies at all.

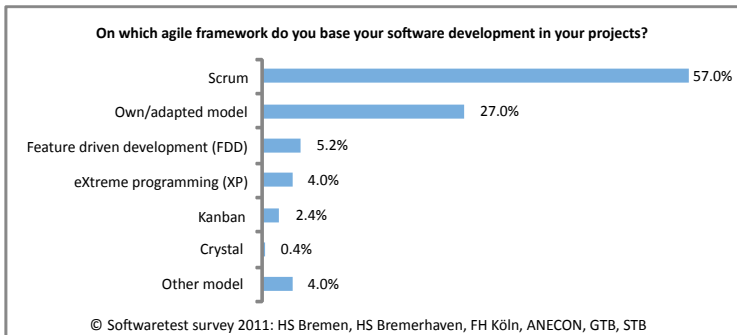


Fig. 6 Agile software development methodologies

Differences Regarding Industries and Company Sizes

It was hardly surprising that the defense and arms production industries use agile projects the least, whereas almost a third of local authorities apply agile. Most agile projects are found in the media, consumer goods and entertainment industries (35-38%).

With regard to company sizes, there are significant differences. Whilst around 40% of small to very small companies (1-100 employees) apply agile methodologies, around 30% of medium-sized enterprises still rely on agility. In large to very large organizations (>1,001 employees), the share of agile projects is even lower with between 17 and 19% (see fig. 7).

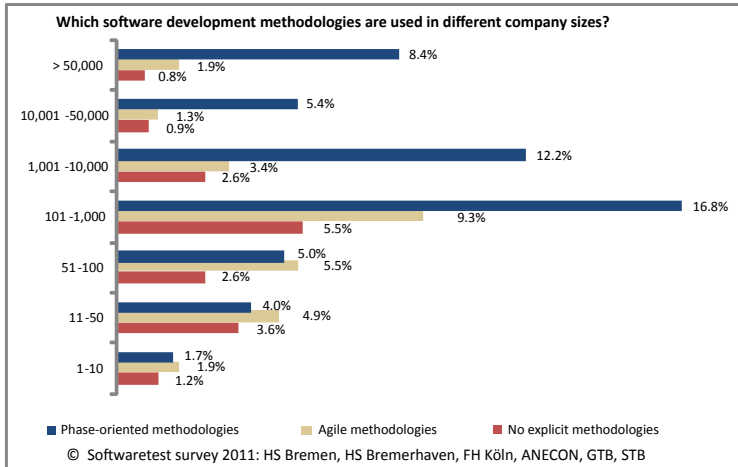


Fig. 7 Size of company and software development methodologies

The question whether there is an independent QA group or QA department in the organization was answered negatively by 37% of agile participants, but by the users of phase-oriented methodologies with only 28%. This response, however, must also be seen in relation to the company size, since agile frameworks are particularly popular in small companies that usually do not have their own separate QA department.

Involvement of the Customer in Agile Projects

The involvement of quality assurance into agile projects has not yet been conclusively determined. However, there is some indication that the pure software development tasks are performed according to agile guidelines, but that QA has not yet been geared up for agile methodologies.

This gives rise to the impression that agility has not yet taken on properly. One of the benefits of agile methods is the closeness to the customer. It could therefore have been expected that in agile projects the employees from other business departments are better involved into the QA measures

than in traditional projects. As a matter of fact, the business departments in conventional phase-oriented projects are, for instance, involved in test case design with almost 50%, compared with only 33% in agile projects. In agile projects, reviews are conducted by the business departments in only 57% of the responses given, whereas in phase-oriented projects this at least applies to 72%. This appears to be due to the fact, that there is not yet a consistent perception of the Product Owner role.

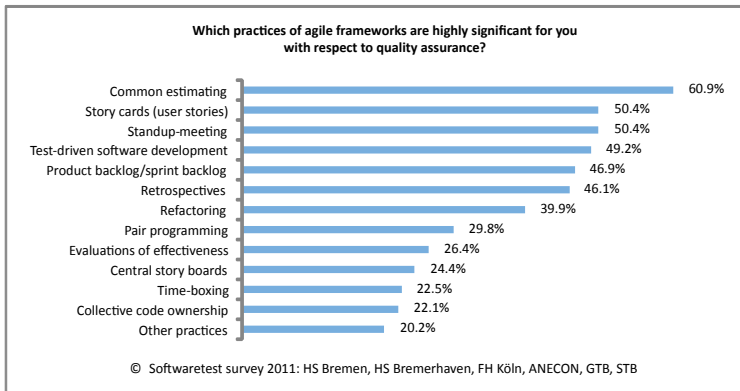


Fig. 8 Practices of agile methodologies

The question, which practices of agile frameworks are highly significant with regard to QA, provided some surprising responses (see fig. 8). Only approximately half of the survey participants that use agile practices stated that test-driven development, stand-up meetings, retrospectives or user stories are significant practices for QA.

The results are unforeseen in that a stronger inclusion of the advantages of agile frameworks, e.g., the nearness to the business departments or customer, or the recognition of good practices as QA measures, had been expected.

In the agile projects, 77% of survey participants stated that unit tests were part of each iteration. Only slightly below this is the pervasion of each

iteration with integration tests. Here too, higher approval ratings - if following the pure agile doctrine - had been anticipated.

When taking a closer look at a tool usage, differences in the respective methodologies became apparent. Whilst tool-supported test execution is approximately at the same level, there is a significant difference in the usage of tools for test case specification: In agile projects this lies (with 38%) significantly below that of typical phase-oriented projects with 53%.

17% of survey participants stated that they do not follow a defined process for executing their test activities, since they develop in an agile manner and determine the test activities rather spontaneously. To this end, there is some room for improvement for the agile projects.

Responsibility for Quality Assurance

When asking survey participants who is responsible for QA in the projects, developers are mentioned much more frequently in the agile projects (with 50%) as opposed to in the phase-oriented projects, with only 30%. The difference also becomes very apparent when looking at the roles of project manager, test manager and QA representative. In agile projects, all these roles are less important in respect of QA than in phase-oriented projects. This does not come as a surprise, since in agile projects developers are allocated a higher degree of responsibility regarding their work products.

It did, however, come as a surprise that in agile projects testers are much more frequently named as the group of people responsible for QA (48%) as opposed to in phase-oriented projects (36%). The thesis that agile projects make the role of the explicit tester superfluous can therefore not be confirmed.

Agility and Quality

If quality is considered from the viewpoint of defects in production, there is hardly any difference between phase-oriented and agile methodologies (see fig. 9). Both approaches equally claim that deliveries are achieved with-

out severe defects or with only few defects. Too many severe defects come in for both methodologies at less than 3%.

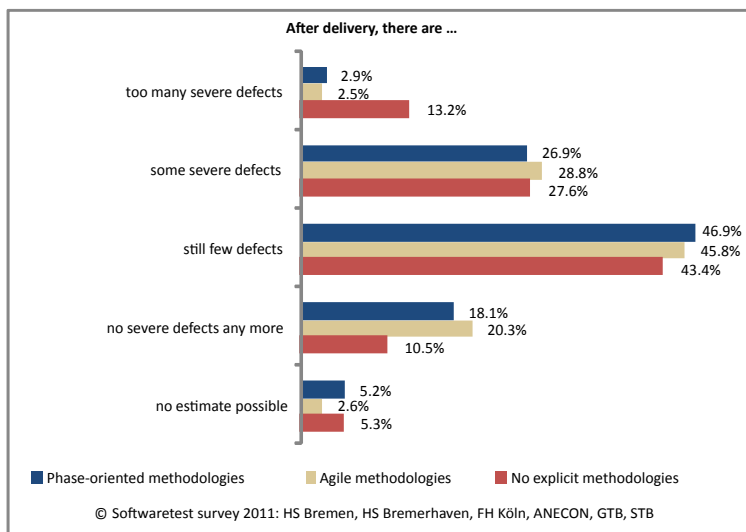


Fig. 9 Estimate of defects occurring after delivery

However, wide differences are revealed when comparing the responses of those that do not use either a phase-oriented or an agile framework. Here the deliveries containing too many severe defects are with 13% significantly higher. In addition, these projects achieve a lower share of deliveries without any severe defects (only 10%) in contrast to nearly 20% for the agile and phase-oriented projects.

It is therefore noted that neither agile nor phase-oriented methodologies have an advantage with regard to the quality of the deliveries, but without using an explicit methodologies, the quality is significantly worse.

Organization

Whereas in 1997 the question inquiring about having an own quality assurance organizational unit in the company was only affirmed by 31% of survey participants, the current survey shows, with 66%, a much more positive result (see fig. 10).

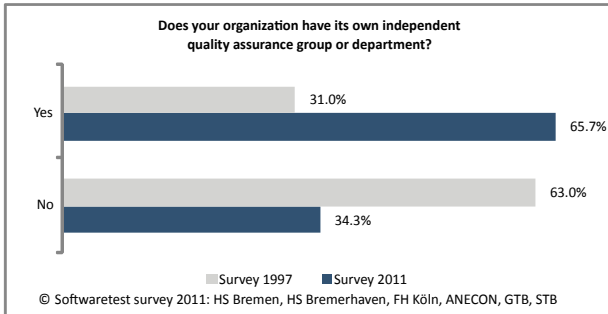


Fig. 10 Independent quality assurance department (comparison 1997/2011)

For 44% of the survey participants, this group is allocated to quality management, for 21% to software development, and for 12% to the business departments. When asked about the percentage share of persons in quality assurance compared to all persons employed in software development, differences to 1997 become evident.

One quarter of the interviewees stated that 2-5% of the employees in software development are assigned quality assurance tasks (1997: 19%). Whilst back in 1997 almost a quarter of the survey participants stated that 30-50% of employees in the software development departments work in the area of QA, today only 8.2% are still of this opinion. In total, the share has slightly increased, whereby the main focus is at 5-30% (see fig. 11).

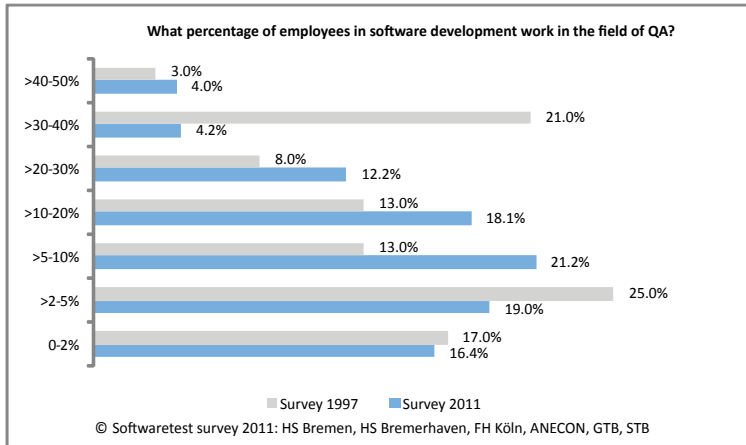


Fig. 11 Employees working in QA (comparison 1997/2011)

The 1997 survey also enquired about who performs the tests at the test levels. At that time, it was mainly developers that executed the tests, and trained testers played a secondary role. What is the situation today?

Today 86% of the survey participants still see the developers working in test execution, but with a major leap from 26 to 77% trained testers have caught up considerably (see fig. 12). Changes can also be observed in the area of test case design. For 78% of survey participants (1997: 35%) test cases are designed by testers, for 56% (1997: 60%) of participants this work is carried out by developers, and for 40% (1997: 54%) by employees from the business departments. External consultants are only employed for this task by 7% (1997: 10%) of the organizations. This clearly demonstrates that systematic test case design through trained testers qualified for the job has increased.

A clear trend toward outsourcing could not be affirmed, with only 15% of organizations employing external service providers for test execution. When asked who is responsible for quality assurance in their projects, as little as 6% of interviewees stated that they contract this task out to external companies. Here a significantly higher share had been expected.

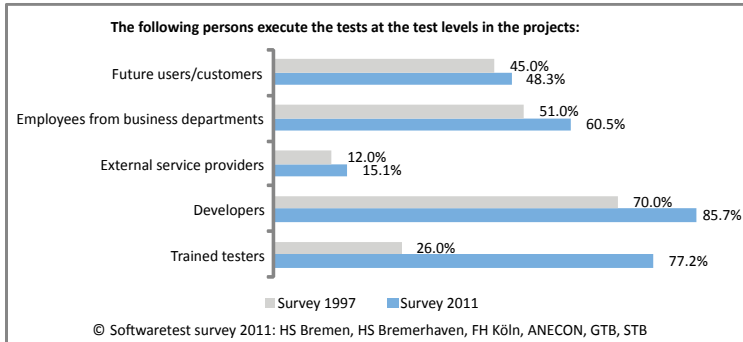


Fig. 12 Test execution (comparison 1997/2011)

Test Process and Quality Assurance Measures

With the ISTQB® test process, a fundamental and very well described test process is available, which consists of test planning and control, test analysis and design, test implementation and execution, evaluation and reporting, and test closure activities. However, is this test process also used in practice? 40% of interviewees stated that they use their own test process for performing the test activities. The ISTQB® test process is applied by 14%, and the test process prescribed by the software development framework is used by 17%. It is surprising and alarming at the same time that around 30% do not follow any defined process. An audit of their test process is performed by approx. 40% of participants. In this connection 4% of the interviewees use TMMi®, 17% TPI®/TPI Next® and 25% prefer another reference model. All in all, the result is gratifying when considering that almost half of the organizations that have a test process also audit it.

Test Case Design

Systematic test case design and test data definition are widespread. More than half of the participants of the group of testers stated that a review and analysis of the test basis and the test object for testability are performed as part of test preparation. More than 80% stated that test cases are designed, and 60% each stated that test data definition and test data generation are carried out.

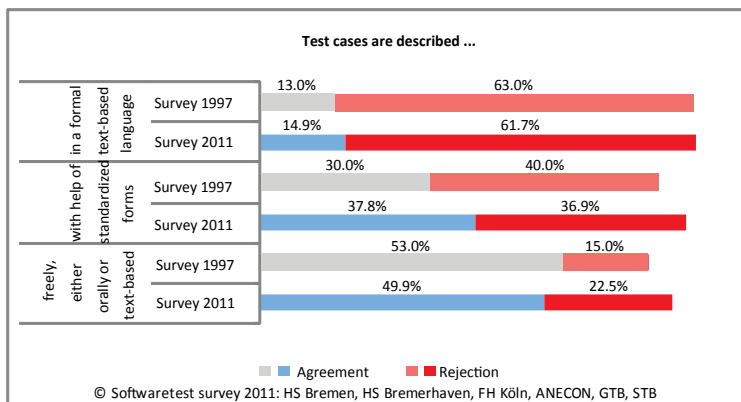


Fig. 13 Test case design (comparison 1997/2011)

Test cases are still described freely in verbal or text-based forms in just under 50% of all cases (see fig. 13). Even today, formal languages are only used by 15% of interviewees, showing that these approaches have not become established in practice. Almost 50% of participants perform a review of their test cases.

The expected test results are available prior to test execution in 72% of all cases. This survey result is a positive surprise. However, slightly less than 60% are actually made available for later tests. How are actual and expected results compared? Approximately 50% of participants compare the results manually. Apart from that, 22% stated that the comparison is normally au-

tomated and performed with the help of tools. However, 36% reject the automated comparison of test results.

The effectiveness of test cases is rated high, since a majority of defects are found during testing. 10% of participants rate the effectiveness of their test cases as very high (almost all defects are found). In comparison, twice as many rate test case effectiveness as medium (some defects are found) and only 4% are of the opinion that effectiveness is bad because the test cases find too few defects.

Test Data

Data protection is a matter that organizations must take very seriously these days. It is therefore quite surprising that 15% of survey participants state that they still test with original data from production (see fig. 14). 23% use an (anonymized) copy of the production data, and 30% of the interviewees comprehensively document the test data. In addition, more than half (58%) of the survey participants stated that they do not explicitly distinguish between test case generation and the generation of associated test data.

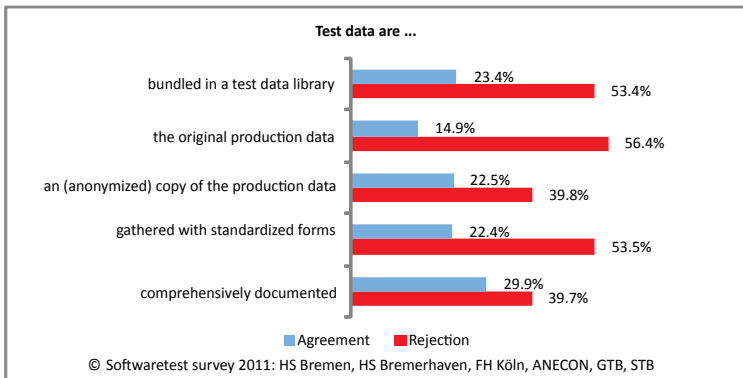


Fig. 14 Test data management

Test Environment

A pleasant result revealed by the survey is that more than 80% use a separate test system. Nevertheless, more than half use the development system, and 22% still use the production system for testing (see fig. 15).

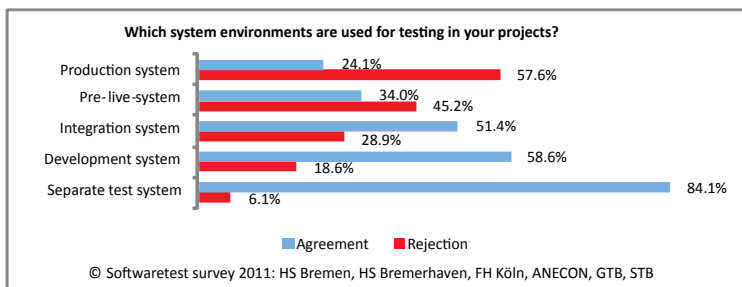


Fig. 15 Test environment

When it comes to the administration of the test environments, different groups of persons are frequently in charge at the same time (see fig. 16). In this context, testers from the project team, central infrastructure and developers from the project team are each mentioned by ca. 50% of the survey participants. In this connection, organizations presumably distinguish between the administration of test data, administration of test objects and administration of basic components (hardware, operating system, database, web server, etc.).

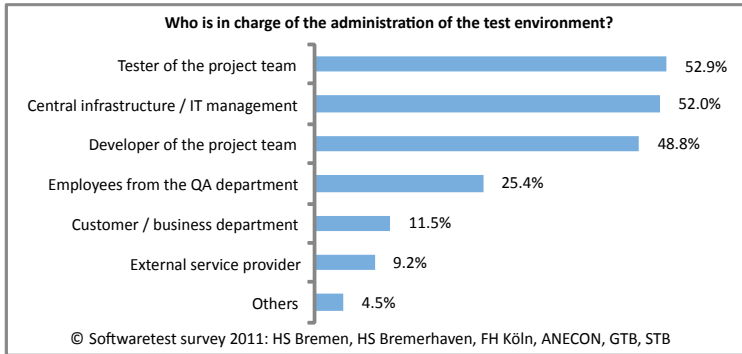


Fig. 16 Administration of test environment

Test Completion

Metrics such as requirements coverage (75%), test case execution rate (60%) or code coverage (25%) are amongst the measures used by the survey participants for test control. A systematic pattern can be detected in these figures, and the criteria are in 85% of all cases mostly or always adhered to.

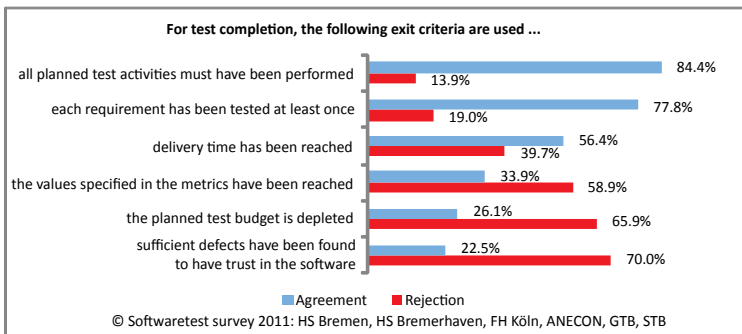


Fig. 17 Test exit criteria

Although a good test process maturity can be concluded from this, it is nonetheless surprising that more than half of the survey participants close their test activities when the delivery time has been reached, and one quarter when the test budget is exhausted (see fig. 17). This is an indication that testing continues to be planned as a „buffer“ in the project that will be „sacrificed“ if there are delays from preceding project phases.

Rules of Quality Assurance

Quality assurance measures are determined for 40% of survey participants through regulations/guidelines (see fig. 18). For 31% the measures are the projects' own responsibility; no explicit regulations whatsoever apply only for 7%.

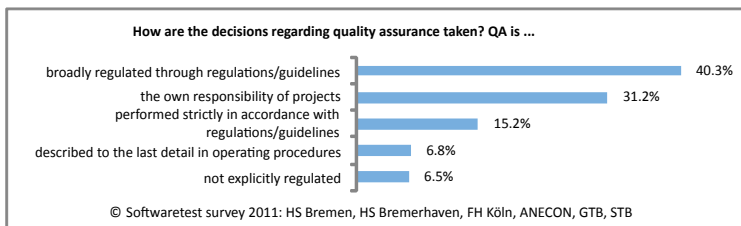


Fig. 18 Rules of quality assurance

The tasks of quality assurance are recognized and there is sufficient budget made available. On average 20% of the total budget and 20% of the total time available are used for quality assurance and are estimated using this fixed rate in advance. In this respect, there are hardly any differences compared to 1997. Almost 60% stated that the budget is estimated and planned together with other software development activities in a total package. Only 7% of survey participants do not plan any explicit budget for quality assurance (see fig. 19).

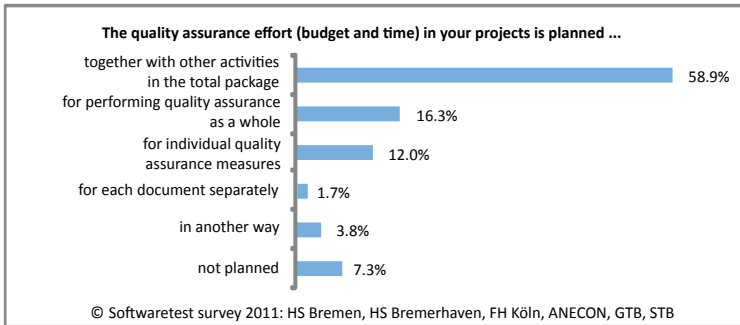


Fig. 19 Effort for QA

40% of survey participants stated that the planned budget and the planned time for quality assurance are on average largely adhered to. However, there were also 18% that stated that the budget is exceeded by more than 50% (with almost a tenth even exceeding by 50-100%).

In the cost estimate the system test comes out top (with 81%) as an important element, followed by acceptance and integration test. Reviews are judged as not important in the cost estimate by 23% and load/performance tests by 17%.

Defect Management

There was high agreement in the responses to the question how the defects found are managed. Generally, each defect found is allocated to a priority class and then corrected in accordance with the defect priority. Over 70% allocate defects to priority categories. In the 1997 survey, this had only been the case for 55%, but even back then the defects were corrected (for 81%) in accordance with the defect priority (see fig. 20).

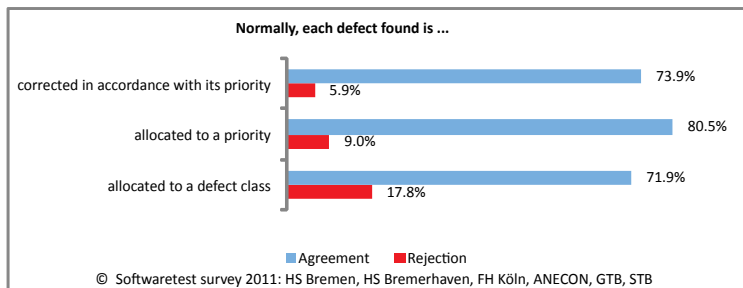


Fig. 20 Defect management

Re-testing for defect resolution is performed by 81% of survey participants. In this connection, 70% of interviewees simply repeat the failed test cases that revealed the defects, 43% also run additional test cases from those relating to the defect, and 23% execute all test cases (full-scale regression test).

Special Test Types

If survey participants are asked which special test types are used in the projects, regression tests top the bill (with 80%), followed by smoke tests (60%). Whilst load and performance tests still play a big role (just under half of the responses), the survey revealed that security and penetration testing was only stated by 26% of interviewees (see fig. 21). This is all the more astonishing in the light of frequent reports about data abuse and data theft; more safety awareness would certainly be appropriate.

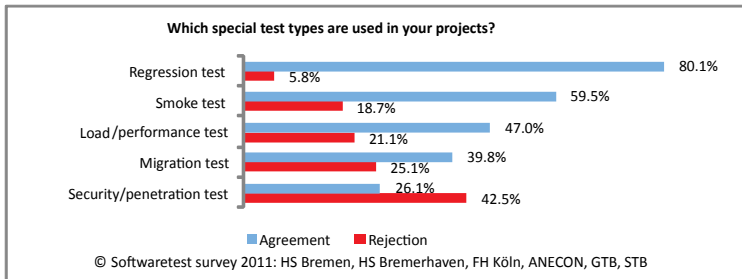


Fig. 21 Different test types

Test Tools and Automation

Test tools have in recent years been substantially developed, and have gained a great deal of importance through trends in areas such as application life cycle management and agile development. New players have come into the market, and the established tool suppliers find themselves challenged by high-performance open source tools. But how are the tools actually used in practice?

The study has shown that test tools are most frequently (89%) used for executing test cases. 66% of survey participants use tools for comparison of the achieved and expected results of tests. Not quite half of the interviewees use tools to support test case specification, and the same goes for setting up an initial state of test objects prior to test execution. Test tools are used by 37% for test data generation, and by 32% for test case generation.

Automation of Test Levels

The results clearly show that test automation is focused on unit testing. Whilst in the unit test more than half of the tests are automated by 70% or more (26% of interviewees even declared to have automated 100% of the unit test), the figures for the integration test show that 30% of participants still automate 70% or more. The acceptance test is the least automated test

level, and more than 40% of survey participants perform it manually (see fig. 22).

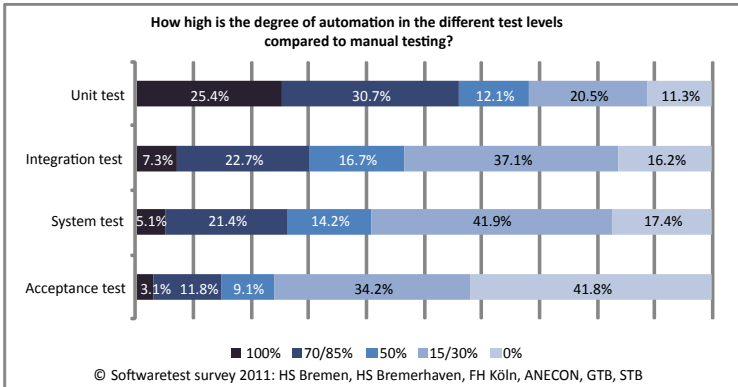


Fig. 22 Degree of test automation

What did come as a surprise was that also in the agile projects, complete automation of the unit tests only occurs in 43% of all cases. It would have been reasonable to expect that close to 100% of the unit tests are automated.

Use of Test Tools

When it comes to the use of test tools, there is hardly any difference between industry sectors. However, it was shown that developers use test tools significantly less (51%) than is the case for testers (70%) (see fig. 23). In agile projects, the percentage of developers using test tools is higher (71%). Nevertheless, a higher result was expected.

The study has shown that developers as a potential user group are still not targeted sufficiently by test tool producers.

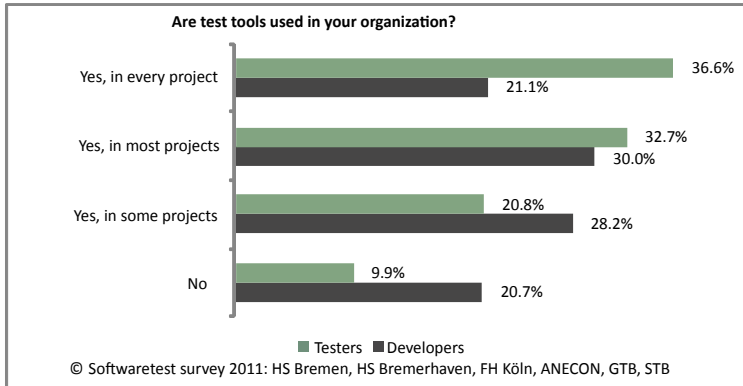


Fig. 23 Use of test tools

Reviews and Test Design Techniques

Amongst the static test design techniques, informal reviews is leading. Almost 70% of the interviewees stated that they use informal reviews. Technical reviews of documents/software code are performed by 57% of the participants. Walkthroughs and peer reviews are used by more than 40% of interviewees. These figures are as expected.

Tool support is used in particular for checking coding standards and programming rules. The generation of code metrics, static analysis of the software architecture, and control flow analyses are performed with tool support in around 20% of all cases. However, almost 40% of the survey participants do not use any tools for static analysis (see fig. 24).

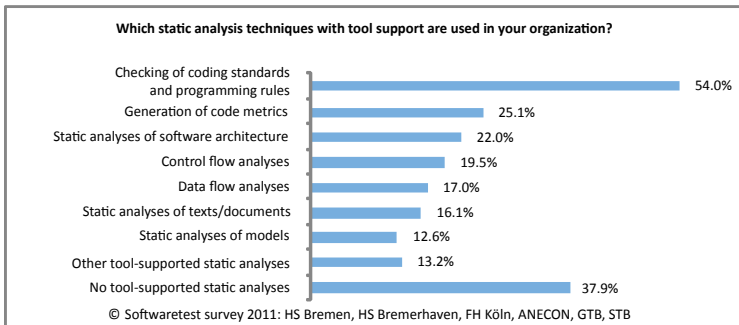


Fig. 24 Tool support of static analysis

More detailed questions regarding testing methodologies were addressed exclusively to the group of testers who were believed to have the appropriate know-how. The survey showed that in their dynamic testing activities this group uses principally specification-based test design techniques (Black-Box) (92%) and experience-based test design techniques (82%). Approximately half of the survey participants use structure-based test design techniques (White-Box). Just under a third perform dynamic analysis supported by tools.

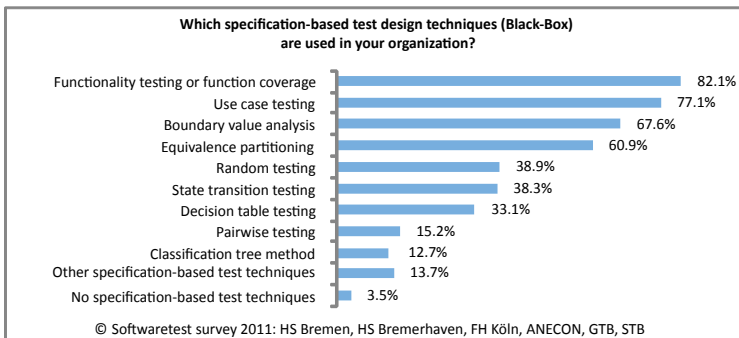


Fig. 25 Specification-based test design techniques

When it comes to the black-box test design techniques, 82% rely on functionality testing or function coverage, followed by use case testing, boundary value analysis and equivalence partitioning. State transition testing is only used by 38%, and decision table testing only by 33% (see fig. 25).

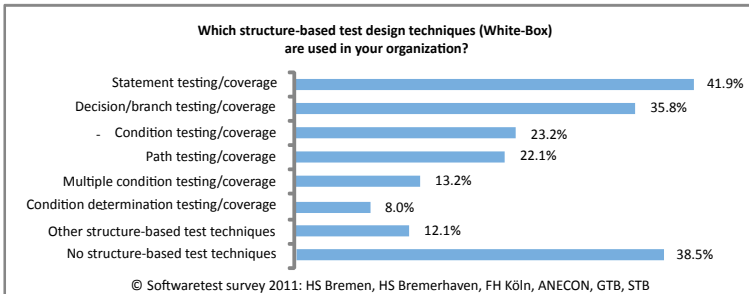


Fig. 26 Structure-based test design techniques

Amongst white-box test design techniques, statement testing / statement coverage are performed most frequently (42%). However, the survey showed that no structure-based test design techniques whatsoever are used by almost as many of the participants (see fig. 26).

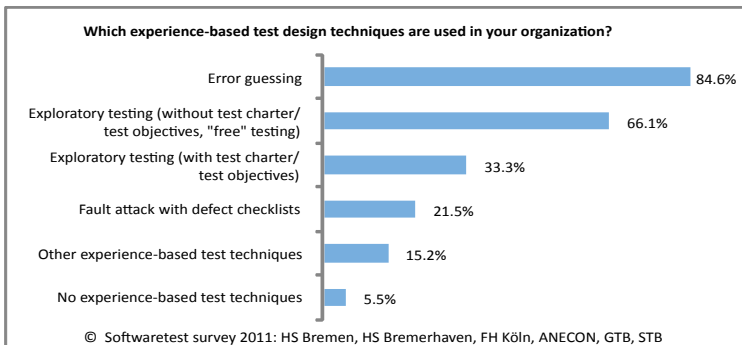


Fig. 27 Experience-based test design techniques

If experience-based test design techniques are used in the organization, it is error guessing that is used most frequently in 85% of cases. Exploratory testing („free“ testing without test charter/test objectives) is mentioned by two thirds of the interviewees, followed by exploratory testing with test charter/test objectives used by one third (see fig. 27). It is, however, interesting that the results of exploratory testing are not used for designing test cases. Whilst 18% agree that they design their test cases through an exploratory technique, 43 % tend to reject this.

Testing from the Developers' Perspective

Compared to the 1997 survey, the share of employees trained in testing has fortunately risen. Testers themselves are very much aware of their significance, and they are accepted by developers. 77% of the group of developers stated that tests are executed by trained testers. This result is expected and is what had been hoped for.

Nevertheless the different groups of testers and developers each perceive their own group as the one bearing most of the quality assurance burden. This becomes distinctly apparent in the responses to the question of who is responsible for QA in the projects. Whilst the group of testers sees the responsibility to be primarily with the project managers, QA representatives and test managers, the group of developers state their own role to be by far (65%) the one responsible for QA. Test managers and QA representatives are mentioned by less than 30%.

Standards, Communities and Certificates in Software Testing

When asked about qualification measures in the area of QA, more than a third of developers respond that they do not receive any support in this field (see fig. 28). Developers get significantly less support through organization-wide or individual training and qualification programs in quality assurance than testers. There is cause for concern that developers very often perform tests, but do not receive adequate further training. Especially with regard to the agile frameworks and the increasing requirements regarding security

vulnerabilities and stability, organizations should start to re-think the situation and also train developers in the fundamental test processes and test tools.

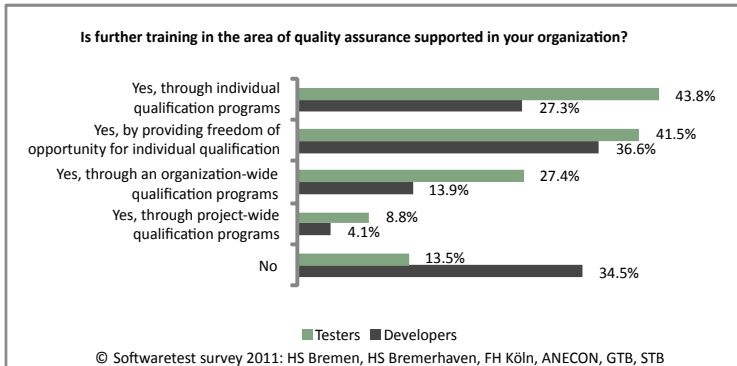


Fig. 28 Qualification program in organizations

The ISTQB® training scheme is known not just amongst testers, but also in management circles, by 70%. In the group of developers, however, this figure only stands at just under 40% (see fig. 29). Testers are accepting the training and qualification program very well: 67% of testers that know the ISTQB® training scheme are also certified at Foundation Level, and 90% of them find this training helpful for their job.

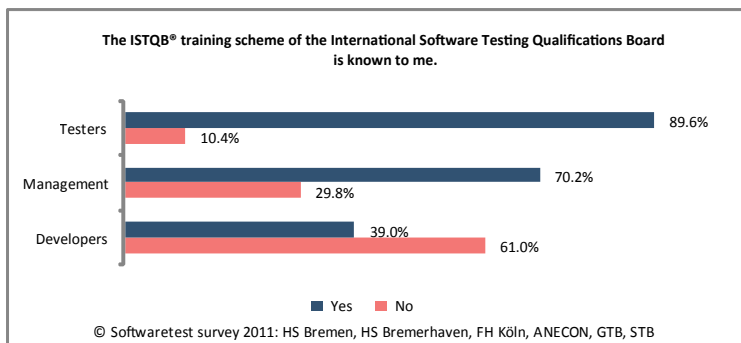


Fig. 29 ISTQB® training scheme

For personal further training, the management group uses primarily events and conferences. It therefore comes as no surprise that the events of working groups and regional groups in the test community are first and foremost attended by interviewees from the management group, followed by the group of testers. On the whole, however, this offering is only taken up by a few.

Conclusion

The primary objective of quality assurance is seen to be in the increased capability. Cost reduction only comes in second place. Even though it might appear that cost reduction often seems to be in the foreground, the result of the current survey seems to confirm the result of the 1997 survey. Here, too, increased capability was named in first place, followed by the reduction of costs. What was remarkable, however, is the fact that these statements are to a high degree consistent between both the management and the tester groups. This clearly shows that the objectives of testers are very much consistent with their organization's objectives.

More than 40% of organizations that have a test process also have it audited. This very clearly demonstrates that software test and quality assurance in software development are becoming more professional. In addition, metrics such as „requirements coverage“, „test case execution rate“, or „defect detection rate“ are more frequently cited as test exit criteria. On the other hand, however, 56% of the survey participants still state that testing is completed when the delivery time has been reached, which makes testing into a sort of „buffer or residual activity“. The test exit criteria are always or almost always kept in 86% of all cases, which shows that a systematic approach is recognized and also followed.

On the whole, it can be confirmed that quality awareness has increased, and that quality assurance is integrated into the software development process at an early stage.

After delivery, severe defects were still found in the system for a third of the interviewees. With regard to defects occurring in production, there is no discernible difference between agile and phase-oriented frameworks. Both of them are, however, significantly more effective than using no explicit framework at all.

Whilst the question enquiring about an independent quality assurance department in the organization was affirmed only by 31% of survey participants in 1997, the 66% in the current survey is a markedly more positive result. The budget for QA and testing is currently around 20%. This is estimated as a fixed rate and it is mostly adhered to.

It is pleasing that testers work very systematically, and that it goes without saying that this group works methodically in test levels.

In management, the added value of testing is recognized. However, in contrast to the testers, management is more optimistic when it comes to the quality assurance budget being adhered to and to the effectiveness of risk management. A positive result that should also be noted is the broad agreement between testers and management regarding the assessment of these aspects. Compared with 1997, testers are increasingly perceived as important and on an equal footing with developers. The value added by quality assurance is recognized.

Developers are involved in quality measures to the same degree as testers. If you look at the extent of training, there are clear differences. Developers have fewer opportunities for qualification in the area of testing. As far as the use of tools and the application of test processes are concerned, they should get better support.

A positive outcome is that the use of dedicated environments for the tests is very widespread (> 80%). It is surprising, however, that test automation is still little used in acceptance, system and integration testing. Test automation is focused on the unit tests, although even this test level is still far from being completely automated. Acceptance and system tests are only being substantially automated by a small group of survey participants.

An extensive trend toward shifting test activities to external service providers cannot be identified. Consultants and testing service providers are only to a minor degree entrusted with the execution of tests.

The ISTQB® training scheme is widespread, and the testers are highly satisfied with the benefits of the ISTQB® certification. However, working groups are rarely attended, and there is room for improvement regarding the communication within the test community.

Literature and links

[Sur11]

Umfrage 2011 – Softwaretest in der Praxis, see:

<http://www.softwaretest-umfrage.de/>

[Mül98]

U. Müller, T. Wiegmann, O. Avci, »State of the Practice« der Prüf- und Testprozesse in der Softwareentwicklung, Ergebnisse einer empirischen Untersuchung bei deutschen Softwareunternehmen, in: W. Mellis, G. Herzwurm, D. Stelzer (Hrsg.), Studien zur Systementwicklung, Band 16, 1998, see:

<http://systementwicklung-archiv.bibliothek.informatik.uni-koeln.de/28>

[Ilias11]

Open Source e-Learning, see:

<http://www.ilias.de>

[ISTQB10]

ISTQB®/GTB Standard Glossary of Terms Used in Software Testing. Version 2.1. ISTQB®/GTB e.V. 2010. As download from the German Testing Board e.V., see: <http://www.german-testing-board.info/de/downloads.shtm>

How to read the graphics

In graphics without legend the following color semantics were applied:

- ■ All participants were asked
- ■ Only part of the participants were asked

In accordance with the Likert scaling, responses such as always or mostly were interpreted as being in agreement, rarely or never as rejections and partly as neutral. In addition, we did not distinguish in the presentation between multiple-choice and single-choice questions. This is obvious from the context.

Authors

Peter Haberl

is as managing director at the ANECON Software Design & Beratung G.m.b.H. responsible for the services and customer relations in Germany.

Prof. Dr. Andreas Spillner

is Professor for Software Engineering and Quality Assurance at the Bremen University of Applied Science. He was founding member and now member of honorary of the German Testing Board e.V. He was founding Chairman of the GI Working Group „Test, Analyze and Verification“. Since 2007 he is GI Fellow.

Prof. Dr. Karin Vosseberg

is Professor for System Integration and Quality Assurance at the Bremerhaven University of Applied Science.

Prof. Dr. Mario Winter

is Professor at the Cologne University of Applied Science with focus on Software Development and Project Management. He is founding member of the German Testing Board e.V. and he was Chairman of the GI Working Group „Test, Analyze and Verification“.

This document is a translation of *Umfrage 2011: Softwaretest in der Praxis. 1. überarb. Auflage, dpunkt.verlag, 2012*. Translation was done by Elke Bath