Survey on software testing practices

Article *in* IET Software · June 2012

DOI: 10.1049/iet-sen.2011.0066

3 authors, including:

Jihyun Lee
Chonbuk National University
61 PUBLICATIONS **246** CITATIONS

SEE PROFILE

Sungwon Kang
Korea Advanced Institute of Science and Technology
135 PUBLICATIONS **856** CITATIONS

SEE PROFILE

A Survey on Software Testing Practices

Jihyun Lee, Sungwon Kang, and Danhyung Lee

Computer Science Department, Korea Advanced Institute of Science and Technology,
291 Daehak-ro, Yuseong-gu, Daejeon, Korea

{jhyunlee, sungwon.kang, danlee}@kaist.ac.kr

Abstract

To develop high-quality software, it is essential to use software testing methods and tools (STMTs) effectively and efficiently. We conducted a survey with a wide variety of companies and experts that are involved in software testing in order to identify the current practices and opportunities for improvement of STMTs. The survey results revealed five important findings regarding the current practices of STMTs and opportunities for improvement: low usage rate of STMTs, difficulties due to a lack of STMTs, use of testing tools in a limited manner, demand for interoperability support between methods and tools of software development and testing, and need for guidance to evaluate STMTs or to describe the capabilities of STMTs. These findings and other related results from the survey will be useful for improving STMTs practices and developing software testing tools.

1. Introduction

Software testing methods are techniques, procedures, patterns, or templates used to conduct software testing tasks effectively and efficiently. Examples include metric-based approaches for test estimation, black-box or white-box techniques for test design, static testing techniques, and so on. Software testing tools are software products that provide automated or semi-automated support for software testing methods and processes, such as tools for dynamic analysis, coverage analysis, and test design. Software testing tools are developed to support particular software testing methods [1] and automate the methods fully or partially.

The current software testing practices are far from satisfactory, as indicated in various studies on software testing practices [2-5], software testing research status and their challenges [6, 9], and reviews on testing methodologies and techniques, as well as testing tools [7-8]. They argue that there are still needs for sophisticated tools [2, 6] and seamless integration between build and test [4], and there are gaps between testing research and industry practices [3]. They provide opportunities for improvement of test practices from process [7], testing techniques [8], and tools for supporting them [8, 10, 13]. Especially, Ng et al. [10] reports that the portions of testing methodologies used, testing activities performed, and automated testing activities with the barriers to adoption each of them in Australia. There are researches about which testing strategies and selection methods are used [11] and about which methods and tools are popular [12]. And Kasurinen et al. [13] observes the practices in software test automation and identified factors that affect software test automation.

The previous surveys study the usage status of the existing software testing methods and tools (STMTs) in practice and the barriers to adoption of STMTs; however, relatively little is known about the current status of the practices in STMTs in term of capabilities, limitations, improvements, and needs. It is necessary to identify what

capabilities of STMTs should be enhanced in order to ensure the efficiency and effectiveness of testing. Therefore, we conducted a survey for organizations and experts involved in software testing to identify the current practices and opportunities for improvements in STMTs. The survey results are supposed to be used to determine the necessity of the international standard for the capabilities of STMTs. The focus of the survey is not on specific methods or tools are used in the industry but on software testing process and activities that are supported by the methods and tools. The respondents were asked about which software testing process and activities are well supported and which are not, so that the survey aims to investigate as-is practices and perceived weaknesses of STMTs, and needs for the capabilities of STMTs.

The paper is structured as follows: Section 2 presents the structure of the survey and the population that took part in the survey. Section 3 presents the analysis results of the survey. Threats to validity are discussed in Section 4, and finally, Section 5 presents survey findings and conclusions.

2. Survey design and execution

The questionnaire[1] we developed includes over 50 questions and is broken down into three sections: general information and testing environment of the respondent company, current practices and perceived weaknesses of STMTs, and needs for additional capabilities of STMTs. The first section of the questionnaire asks general information about the company or the experts that participate in the survey. The second section asks about the software testing practices of the participated company. The questions in this section consist of the portion of testing budget, difficulties in software testing, and level of test automation. The last section poses questions on the current practices of STMTs in terms of test processes and their activities, which are selected based on ISO/IEC 29119 [14]. The selected test activities are more fine-grained than those used in other previous surveys [10, 13]. For example, our survey intended to find status of which activities of test design process are supported by systematic STMTs so as to find which capabilities of STMTs that support test design process should be improved to enhance their usage.

We also developed a questionnaire for software testing experts. Its basic structure is the same as the questionnaire for company. As for the rating scale, except for STMTs usage related questions, the questionnaire uses a five level rating scale [15], which either ranges from "strongly agree" to "strongly disagree" including “Don’t know” or ranges from “not relevant” to “very relevant”.

The survey was distributed to the companies whose CIO or CTO names could be found in the Fortune 1,000 companies website. The resulting population was 73, of which the number of actual responses was 14. And ISO/IEC JTC1/SC7 members of STMT Study Group also invited the companies with high market share in their countries, of which the selected population and the actual response outcome were both 19. Altogether 33 companies participated in the survey for companies with the response rate of 36%, which was calculated using AAPOR [16]. To reduce the deviation, we discarded the survey responses of the organizations which target the domestic market. As the result, 9 respondents were discarded and a total of 24 responses were analyzed.

Background questions for companies show that the major business sectors for the respondents are software

[1] http://salab.kaist.ac.kr/STMTs

(33%), finance (15%), electronics (15%), communications (7%), transportation (4%), and others (26%). Other business sectors include retail, packaging, insurance, biotechnology, military, and automotive. All respondent companies develop software for business operation or as part of their final products. Occupations of most of the survey respondents were CTO, IT director, project manager, or QA engineer.

In the survey for experts there were 20 participants out of 50 invited respondents who are ISO/IEC JTC1/SC7 members and their colleagues. 59% of the respondents work in the area of education, 24% work as consultants, and 17% work as executives or project managers. 83% of the respondents have more than 10 years of software engineering experience and 47% of them have more than 10 years of software testing experience.

3. Analysis of the survey responses

This section summarizes the responses to the survey questions. Section 3.4 describes the differences in software testing between the organizations with CMM/CMMI level 3 or above and those with level 2 or below. Section 3.5 presents the differences in software testing between the organizations whose products are mission critical and others. In Section 3.6 the differences in views on STMTs between experts and industries are analyzed.

3.1 Testing environments

In the survey respondents were asked to describe their testing environments. First, with respect to budget, respondents answered that on average they spend 24% of the total budget on software testing. This result is consistent with that of Ng et al. [10] (10% to 29% of the initial budget) and that of Kasurinen et al. [13] (24% of the total effort) but less than Juristo et al. [3] (more than 50%). However, approximately 50% of respondents answered that they do not know the budget spent on each activity of software testing. In addition, to determine the obstacles in software testing, respondents were asked to select the level of relevancy, with possible answers ranging from 1 ("not relevant") to 5 ("very relevant") for each of the four obstacles: i.e. lack of test process, lack of testing techniques (methods), lack of tool support, and low skill level. The majority of the respondents answered that the difficulties in software testing are due to a lack of STMTs support and testers' lack of skill. In contrast, the score of test processes was lower than among other obstacles. This is the same as the result in Ng et al. [10] and Grindal et al. [11].

Second, respondents were asked about the current status of test automation. Approximately 74% of the practitioners reported that the software test automation rate is below 50%, and only 16% of them reported that it exceeds 50%. The result is lower than 68.2% that was reported by Ng et al. [10]. The majority of the practitioners did not automate their test estimation, test planning, or test design processes among other processes. The average of test automation for each of the software test processes is shown in Figure 1.

3.2 Usage of STMTs

Motivating factors behind this question are to identify the capabilities of STMTs that are being broadly used and must be developed or improved in order to find out how STMTs can help software testing. Respondents were asked

for which software testing activities they are using STMTs. They were asked to select one or more activities among given activities for each testing process and to add their uses of STMTs in cases where they are not listed. The responses are summarized in Table 1. The first column presents processes, and every process has rows of activities and a usage rate for each activity. In particular, activities and usage rates of test design and execution processes are presented by test levels; i.e. unit test, integration test, system test, or acceptance test. Tool usage was significantly lower than the usage of testing methods. In particular, the usage of STMTs in defect management and reporting processes was higher than that of other processes. Respondents answered that they are satisfied with the current support of STMTs but that the current support is limited and much research is needed on related capabilities of STMTs. Respondents recommended in particular implementation of capabilities such as interoperability between software development tools and software testing tools, interoperability of management tools and testing tools (e.g. coordinating between quality assurance and testing results), better automated test case generation, and better seamless management of test cases in execution. The following subsections describe the usage of test activities by each test process.

3.2.1 Test estimation

Among respondents 14% answered that they do not use any methods for software testing. We also discovered that companies use test estimation methods for estimating test efforts, test schedule, and number of test cases while rarely using STMTs for estimating the cost involved. In the case of tools usage, rates of test effort and cost estimation were relatively low. They answered that their test estimation is performed largely manually based on their own know-how or personal knowledge without use of standardized guidance. Concerning the use of tools, as shown in the test estimation row of Table 1, 57% of respondents answered that they do not use any tools for test estimation. Approximately half of the users of methods are using tools.

Table 1. Usage of STMTs

Test Estimation		***Activities***		Estimating test effort	Estimating costs involved	Estimating test schedule	Estimating number of test cases			Do not use
		Usage Rates	Methods	57%	38%	57%	48%			14%
			Tools	24%	19%	33%	33%			57%
Test Planning		***Activities***		Determining test items	Identifying testing-related risks	Determining a test schedule	Determining testing approaches	Defining test deliverables	Defining testing environment	Do not use
		Usage Rates	Methods	57%	38%	62%	52%	57%	43%	14%
			Tools	19%	0%	5%	19%	10%	5%	52%
Test Design		***Activities***		Specifying test design	Specifying test case	Specifying test procedure	Generating test cases	Generating test scripts	Documenting test environment requirements	Do not use
Test Levels	Unit Test	***Usage Rates***	Methods	33%	38%	24%	48%	29%	10%	29%
			Tools	10%	24%	14%	33%	19%	5%	48%
	Integration Test		Methods	52%	71%	48%	57%	43%	29%	10%
			Tools	19%	24%	19%	19%	29%	10%	43%
	System Test		Methods	48%	71%	57%	71%	57%	43%	10%
			Tools	33%	33%	38%	43%	48%	14%	19%
	Acceptance Test		Methods	24%	38%	38%	38%	33%	29%	33%
			Tools	10%	14%	10%	19%	14%	10%	48%
Test Execution		***Activities***		Executing tests	Recording test log	Recording test incident report log				Do not use
Test Levels	Unit Test	***Usage Rates***	Methods	43%	29%	33%				38%
			Tools	33%	33%	38%				29%
	Integration Test		Methods	57%	52%	67%				14%
			Tools	48%	43%	52%				19%
	System Test		Methods	62%	62%	71%				24%
			Tools	57%	57%	62%				14%
	Acceptance Test		Methods	43%	33%	48%				33%
			Tools	38%	33%	43%				38%
Defect management		***Activities***		Recording defects	Tracking defect status	Preventing defects	Resolving defect	Reporting defect resolution results	Analyzing and reporting defect information	Do not use
		Usage Rates	Methods	67%	67%	33%	52%	62%	52%	14%
			Tools	67%	76%	24%	43%	57%	52%	10%
Test reporting		***Activities***		Writing test summary report	Documenting anomaly report	Controlling test	Tracking test progress	Managing testware		Do not use
		Usage Rates	Methods	57%	52%	48%	57%	62%		24%
			Tools	52%	38%	48%	62%	57%		24%

3.2.2 Test planning

About half of respondents (52%) answered that they do not use any tools for test planning while about 50% of respondents answered that they use methods for doing most of their test planning activities. The test planning row of Table 1 shows that the rate of tool usage is remarkably low compared to that of method usage. Tools are rarely used for risk management, scheduling, and environment definition in software testing. It is not easy to automate planning processes because of the nature of planning. However, test plans govern the whole progress of software testing, so the capabilities to support activities of test planning process should be defined.

3.2.3 Test design

As shown in the test design row of Table 1, when respondents were asked to choose activities of each test level

that are performed by using STMTs, the responses indicated that average 30% of practitioners use methods for unit test, which is the lowest among different test levels, and average 58% answered that they use methods for system test, which is the highest. More than 50% of respondents answered that they use methods for test case specification and generation but the percentage of tool usage was less than half of the method usage. The status of tool usage rate in test cases/scripts generation is similar to the results reported in Ng et al. (30%) [10].

3.2.4 Test execution

The questions for test execution were asked in the same manner as with those regarding test design. The usage rates are slightly higher than those of test design. The results is lower than that reported by Ng et al. (79.5%) [10]. However, the test execution rows of Table 1 indicate that the gap between the usage rates of STMTs is narrower than that of test design. We found that testing methods for test execution were widely applied to activities of integration and system test while the rates of tool usage were relatively low. In particular, the majority of respondents answered that they do not use STMTs in test design or test execution of unit tests.

3.2.5 Defect management

Table 1 shows that a majority of practitioners use STMTs for defect management. Only about 10% of respondents answered they do not use any STMTs for defect management. Table 1 also indicates that the majority of practitioners use STMTs for defect recoding, defect tracking, and defect reporting. STMTs usage for defect prevention was considerably lower than the other defect related categories.

3.2.6 Test reporting

As shown in the rows of test reporting in Table 1, about 50% of practitioners use STMTs for test reporting. The average usage rate of STMTs for recording, tracking, and reporting defects is 65%. It is similar to what Ng et al. reports (61%) in [10], which did not decompose activities for survey. The tool usage rate for documentation is lower than others. However, the variation in the rates of STMTs usage is small compared with other software testing processes. The majority of respondents answered that they are strongly satisfied with or are satisfied with STMTs that they are currently using.

3.3 Opportunities for improvement

From the responses to the survey we identified numerous opportunities for improving STMTs. Responses were requested on three aspects: test activities, test levels, and integration and coordination between software development and testing.

3.3.1 Test activities

The survey questions were asked with respect to test activities of each test process. The answers indicate which test activities are being practiced in conjunction with the utilization of STMTs better than others. As shown in Fig. 2,

which aggregates the results of Table 1, most activities of test planning, test estimation, and test design are performed manually. Tool usage was significantly low in test design activities. There is a substantial difference in the usage rate of methods and tools between test planning activities and test design activities. However, in the case of defect management and test reporting, the usage was concentrated on progress and status tracking activities. Respondents wanted better support of STMTs in defect prevention, root cause analysis, predictive reporting, defect pattern definition, etc. Figure 2(a) is a summary of the usage of STMTs including a comparison for each test process. It shows that the percentage of tool usage is lower than that of method usage.

3.3.2 Test levels

The survey results from each test level reveal that STMTs for unit tests and acceptance tests are poorly used (Figure 2(b)). As mentioned in Grindal et al. [11] the majority of tests are run at the system level. Unit test and acceptance test rows for test design and execution in Table 1 show in detail that the usage of STMTs in all activities is lower than at the other levels. As for the support of methods for test design, it is necessary to enhance capabilities in terms of documentation of test environments, specifications of test procedures, and generation of test scripts. In particular, tool support for test design at unit and acceptance test levels should be enhanced.

3.3.3 Integration and coordination with software development

The survey results indicate that demand for interoperability between development tools and methods and STMTs is high. In order to achieve interoperability between software development and testing, interoperability requirements in terms of activity or task-levels should be specified in advance, and thereafter necessary capabilities of STMTs for interoperating with tools and methods for software development should be derived. The respondents answered that the integration and coordination of management tools and testing tools is essential (average value of responses was 4.4, where possible answers ranged from 1 (not important) to 5 (very important)) and need be improved.

3.4 Software testing practices in mature organizations

From a comparison of the responses from mature organizations with CMM/CMMI Level 3 or above (19 companies), and less mature organizations with CMM/CMMI Level 2 or below (5 companies), the rate of STMTs usage in mature organizations was higher than in less mature organizations. The former highly use methods for test estimation and test planning (Figure 3(a)).

At each test level, mature organizations use methods in unit tests and integration tests with rates of 37% and 61%, respectively, while less mature organizations use methods for unit tests and integration tests with rates of 18% and 33%, respectively. As for usage of software testing tools (Figure 3(b)), the gap in test execution between the two groups of organizations was wide. Mature organizations use tools with a rate of 31% while only 12% of less mature organization use tools for unit tests. In the case of integration tests, 40% of mature organizations use tools while only 15% of less mature organizations uses them. The differences in tool usage rates for the other test levels were similar.

In summary, mature organizations use STMTs more extensively than less mature organizations. Mature organizations perform preparation processes such as test estimation and test planning by utilizing methods, whereas the difference in tool usage between them was smaller. In the case of test execution, defect management, and test reporting, the difference in methods usage rates between mature and less mature organizations was similar. In contrast, the gap in the rate of tool usage for test execution was significantly wide.

3.5 Software testing practices in mission critical product

We were interested in assessing the differences in usage of STMTs between respondents who develop mission critical products and those who do not. In the case of methods, more than 50% of mission critical organizations use methods for software testing (Figure 4(a)). The overall rate of tools usage in mission critical organization is higher than that of non-mission critical organizations, but the differences are not large except for with respect to defect management (Figure 4(b)).

3.6 Analysis of responses from experts and industry leaders

We asked experts what software testing activities should be supported by STMTs in order to determine what level of support is desired from their perspectives. We attempted to identify the gaps between the good/best practices of industry and the experts' expectations in usages of STMTs. Experts expected industries to utilize methods for test planning and defect management more than their current levels, but as shown in Figure 5(a) the gap is narrower than that for tools. Experts answered that test planning and reporting tools should be used more (Figure 5(b)), as 22% vs. 9% for test planning and 51% vs. 32% for testing reporting. Experts' expectations on using STMTs for test design specification are higher than the actual practice (31% vs. 18% for methods and 54% vs. 39% for tools). Among various defect management activities, defect prevention shows a gap between experts' expectations and industries' practice (40% vs. 33% in methods and 35% vs. 24% in tools). Tools usage of test reporting is higher than the expectation of experts (51% vs. 32%).

Among the experts, 88% answered that guidance to understanding the relationship among software testing methods, tools, and activities is needed (the average score was 4.4). This result is consistent with the answers of industry leaders (the average score was 4.2). In addition, experts were asked to prioritize software testing topics so as to determine important activities that should be supported by STMTs for efficient and effective software testing. 'Test cases and scenarios design' was the highest priority activity, followed by 'exit criteria for testing', 'test methods and tools', and 'test management' in order of decreasing priority.

4. Threats to validity

The following are the factors that could affect the validity of the survey:

- *Granularity of survey questions*: Lest the number of questions designed based on software testing activities be too large, we did not decompose testing activities into tasks. Thus, although a respondent answers that she or

he uses STMTs for performing a testing activity it might not be all tasks of the activity.

- *Assumption on maturity of the survey population*: We selected companies that have achieved a prominent position in the world market and assumed that the testing practices of such companies would be better than others. To alleviate this assumption we analyzed testing practices of companies with CMM/CMMI Maturity Level 3 or above and companies with Level 2 or below separately as Level 3 guarantees that an organization has standard processes, tools, and methods for testing [17]. The method usage for test estimation and test planning and the tool usage for test execution in companies with CMM/CMMI Maturity Level 3 or above were quite higher than those in companies with Level 2 or below.
- *Representativeness*: The sizes of respondent companies are large (10,000 or more employees), so there are several organization units in a company. Thus, the responses from a department may not represent the practices of the whole company. We tried to mitigate this threat by asking for participation of CIOs/CTOs and asking them to get help from test engineers to give balanced and technically accurate answers.

5. Survey finding and conclusions

The survey was conducted to investigate the state of software testing practices in terms of STMTs. In addition, we attempted to discover the long-term vision of STMTs for supporting organizations involved in software development so that they can manage and improve their software testing practices effectively and efficiently. The findings from the survey can be summarized as follows:

First, the usage of STMTs is low. Unlike other surveys [10, 13], our survey questions were designed at a more detail level without mentioning names of specific STMTs in order to maximize opportunities for improvement. Tool usage was found to be considerably lower (below 30% except for defect management) than method usage (approximately 50%). This means that the STMTs currently in use are not mature enough, and as such their usages are limited. This makes it important to integrate processes, methods, and tools by defining the capabilities of STMTs so that they can complement each other's role, which will enhance STMTs usage and eventually contribute to an effective and efficient software testing.

Second, the respondents are experiencing difficulties in software testing due to a lack of STMTs. This is closely related to the result of Ng et al.[10], which mentioned that a major barrier of method usage was a lack of support tools and a barrier to tool usage was difficulty of using them. This indicates that the gap between the capabilities of the currently used STMTs and the industry needs is substantial, or the capabilities of STMTs are not widely known. To the users, the core capabilities of STMTs can guide evaluation and selection of suitable STMTs, and to the tool suppliers they can provide the minimum set of tool capabilities for supporting methods or processes.

Third, software testing tools are being used in a limited way. Even in defect management and reporting, where tools are rather broadly used, tool usage was limited to activities such as recording defects, documenting test results, or tracking the status of defects. Tools are rarely used in defect prevention due to its high complexity and difficulty. Though defect prediction is a popular research area on which much research has been conducted [18], there are practical problems that fault data are not properly manipulated in most projects and that the cost to collect and maintain fault data is too high. The survey results reveal that STMTs for supporting defect prevention, predictive

reporting, and defect pattern discovery must be improved so as to cope with such problems.

Fourth, the demand for support for interoperability between methods and tools of software development and testing is high. Although STMTs are closely tied to software development, the capabilities currently supported by STMTs are not adequate. Guidance describing capabilities for supporting interoperability should be provided so as to prepare testing activities in support of related software development activities and ensure traceability and integrity between artifacts of software development and testing. Before implementation-level or system-level interoperability of testing tools can be achieved, it is essential to specify the requirements for the basic methods-level and capability-level interoperabilities. In addition, respondents also pointed out the isolation problem between the different STMTs, which was reviewed by Elberzhager et al. [19] from the perspective of methods, especially between tools from different vendors and between tools supporting different software testing activities.

Finally, the respondents answered that guidance to evaluate STMTs or to describe the capabilities of STMTs for supporting testing activities is necessary; approximately 70% of the companies and 83% of the experts surveyed agreed. More than 50% of the respondents answered that they perform their software testing based on individuals' know-how or personal knowledge without standardized guidance. Testing activities that can be supported by STMTs need be identified and criteria to compare and evaluate STMTs need to be set. In addition, relationships between STMTs and testing activities should be made explicit.

These findings and related data from the survey can be used by those trying to improve the practice of STMTs for efficient and effective software testing. The findings would also be interesting to companies that develop software testing tools, as the results encompass the capabilities of STMTs that are widely used as well as those that are not currently supported but are much needed.

Acknowledgements

Many thanks are due to the participants for taking their precious time to respond to the survey. ISO/IEC JTC1/SC7 working group chairs and members and SGTMST members who helped collect questionnaires for the survey are also gratefully acknowledged.

References

1 SWEBOK: ‘Guide to the Software Engineering body of Knowledge: 2004 Version’. *IEEE Computer Society*, 2004

2 Glass, R. L., Collard, R., Bertolino, A., Bach, J., and Kaner, C.: ‘Software Testing and Industry Needs’. *IEEE Software*, July/Aug, 2006, pp. 55-57

3 Juristo, N., Moreno, A. M., and Strigel, W.: ‘Software Testing Practices in Industry’. *IEEE Software*, July/Aug, 2006, pp. 19-21

4 Bertolino, A.: ‘Software Testing Research and Practice’. In Proc. 10th Int. Workshop Abstract State Machines, LNCS 2589, 2003, pp. 1-21

5 Chillarge, R.: ‘Software Testing Best Practices’. *IBM Research – Technical Report RC 21457*, April, 1999

6 Bertolino, A.: ‘Software Testing Research: Achievements, Challenges, Dreams’. In Proc Future of Software Engineering (FOSE’07), 2007, pp. 85-103

7 Taipale, O. and Smolander, K.: ‘Improving Software Testing by Observing Practice’. In Proce. Int. Symposium on Empirical Software Engineering (ISESE’06), 2006, pp. 262-271

8 Juristo, N., Moreno, A. M., and Vegas, S.: ‘Reviewing 25 Years of Testing Technique Experiments’. *Empirical Software Engineering*, 9(1-2), 2004, pp. 7-44

9 Whittaker, J. A.: ‘What is Software Testing? And Why is it So Hard?’. *IEEE Software*, Jan./Feb. 2000, pp. 70-79

10 Ng, S.P., Murnane, T., Reed, K., Grant, D., and Chen, T.Y.: ‘A Preliminary Survey on Software Testing Practices in Australia’. In Proc. Conf. Australian Software Engineering, 2004, pp. 116-125

11 Grindal, M., Offutt, J., and Mellin, J.: ‘On the Testing Maturity of Software Producing Organizations’. In Proc. of the Testing: Academic & Industrial Conference – Practice and Research Techniques, 2006, pp. 171-180

12 Garousi, V. and Varma, T.: ‘A Replicated Survey of Software Testing Practices in the Canadian Province of Alberta: What has changed from 2004 to 2009?’. The Journal of Systems and Software, 83, 2010, pp. 2251-2262

13 Kasurinen, J., Taipale, O., and Smolander, K.: ‘Software Test Automation in Practice: Empirical Observations, Advances in Software Engineering’. Vol. 2010

14 ISO/IEC 29119: ‘Software and Systems Engineering – Software Testing – Part 2: Test Process’. Draft CD, 2009

15 Lissitz, R.W. and Green, S.B.: ‘Effects of the Number of Scale Points on Reliability: A Monte Carlo Approach’. *J. Applied Psychology*, 60(1), 1975, pp. 10–13

16 AAPOR, The American Association for Public Opinion Research (AAPOR), ‘*Standard Definitions: Final Dispositions of Case Codes and Outcome Rates for Surveys*’. 7th edn, http://www.aapor.org, Accessed Nov. 2011

17 Chrissis, M, Konrad, M., and Shrum, S.: ‘CMMI Guidelines for Process Integration and Product Improvement’ (Addison-Wesley, 2nd edn. 2006)

18 Catal, C.: ‘Software Fault Prediction: A Literature Review and Current Trends’. Expert Systems with Applications, 38, 2011, pp. 4626-2636

19 Elberzhager, F., Münch, and J., Nha, V.: ‘A Systematic Mapping Study on the Combination of Static and Dynamic Quality Assurance Techniques’. Information and Software Technology, Available online, 2011

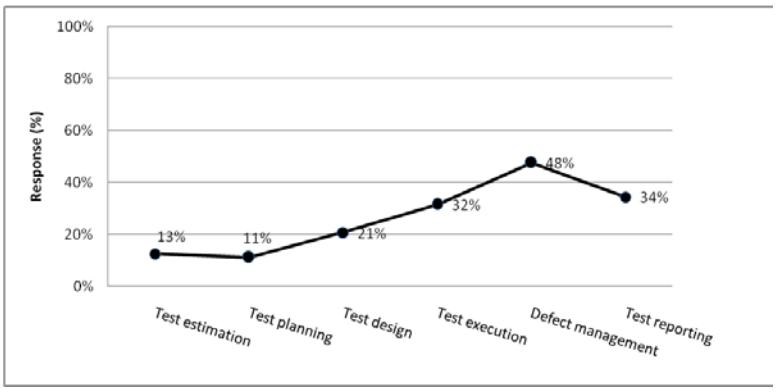
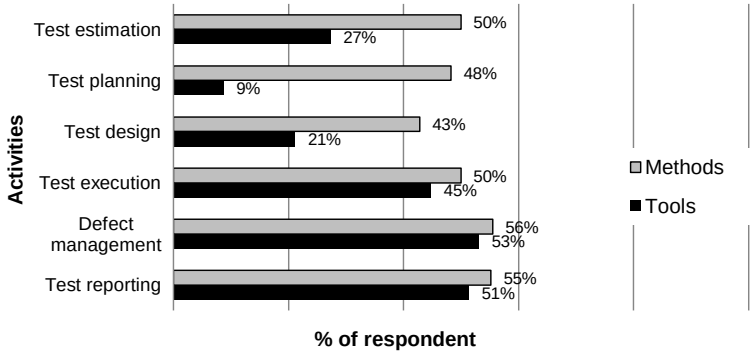
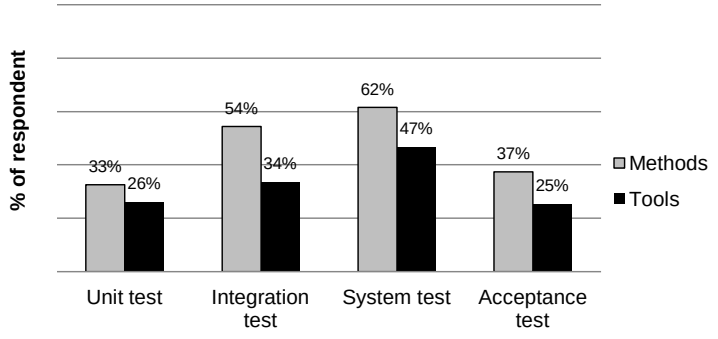


Fig. 1. Level of test automation

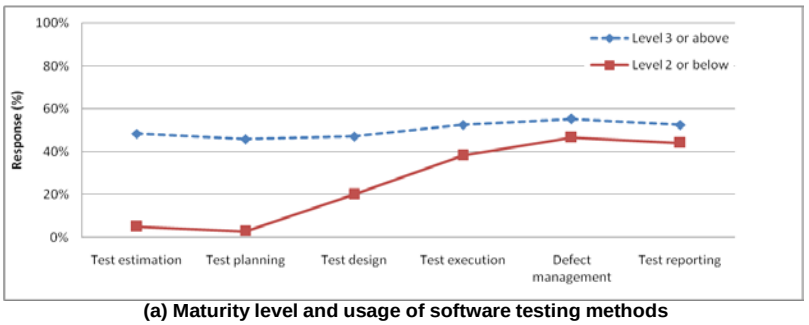


(a) Usage of STMTs in test processes

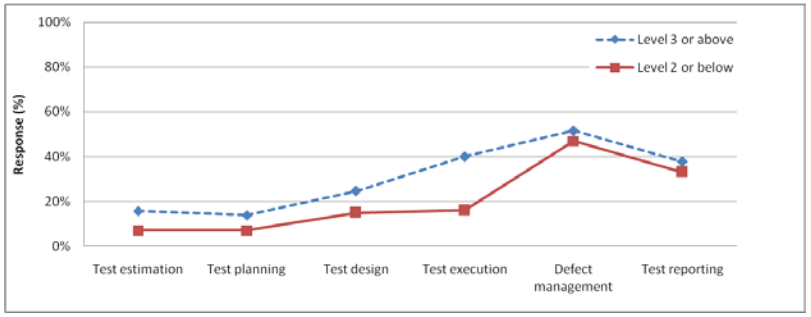


(b) Average usage rate of STMTs at test levels

Fig. 2. Usage of STMTs

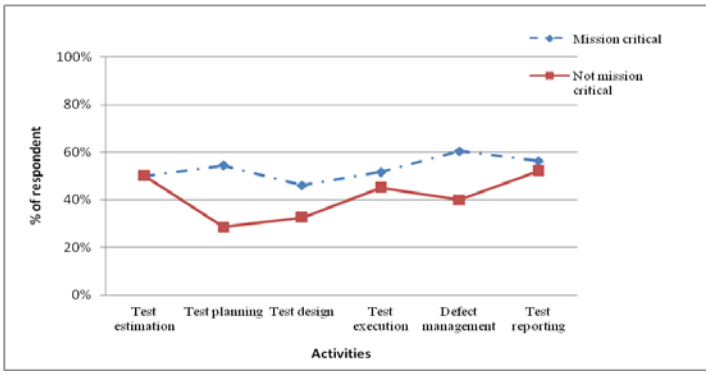


(a) Maturity level and usage of software testing methods

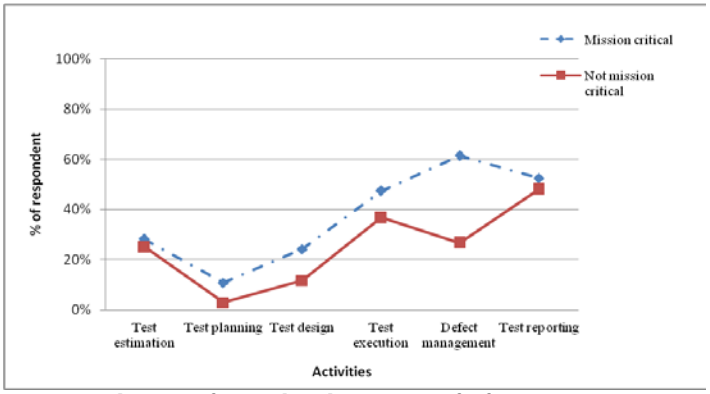


(b) Maturity level and usage of software testing tools

Fig. 3. Usage of STMTs in mature organizations

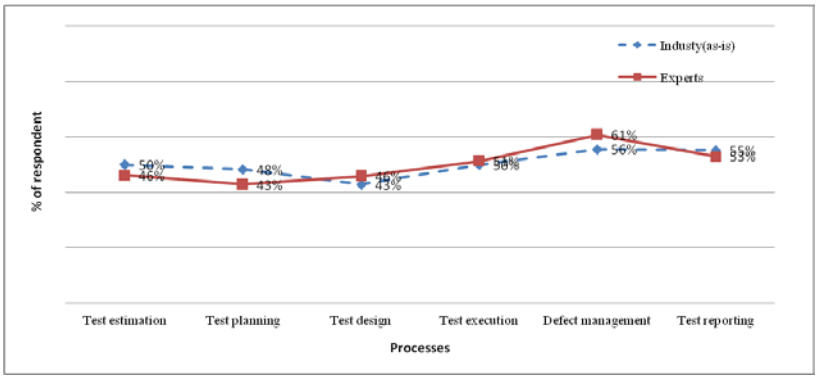


(a) Usage of test methods and importance of software testing

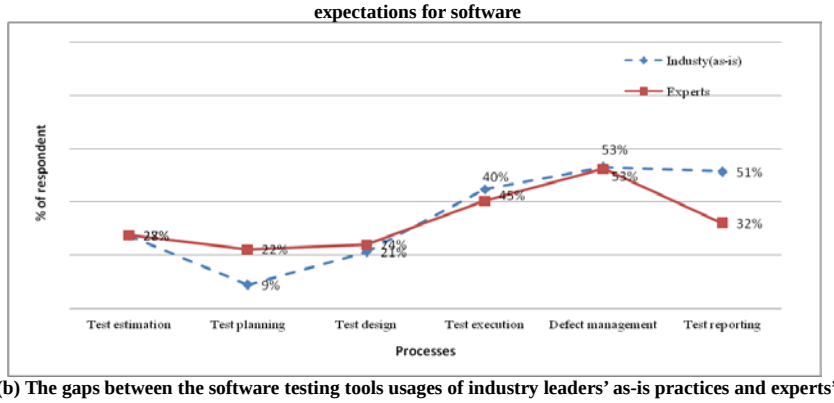


(b) Usage of test tools and importance of software testing

Fig. 4. Usage of STMTs in mission critical businesses



(a) The gaps between the testing methods usages of industry leaders' as-is practices and experts' expectations for software



(b) The gaps between the software testing tools usages of industry leaders' as-is practices and experts' expectations

Fig. 5. Responses from experts and industry leaders