Research on New Techniques and Development Trend of Software Testing

Zhang Hongchun

Hubei Polytechnic Institute, Xiaogan, Hubei, 432000

Abstract—**Some typical latest research results were summarized according to the analysis and research on the current situation of software testing both at home and abroad in this paper. On the basis of the current researches on software testing, the writer discussed the future research topic on software testing and proposed the assumption on software testing's subject, industry and technique development. It laid the foundation both for further boosting the development and formation of software testing discipline system and for effectively studying on the scientific development mechanism of software testing techniques and its industry.**

Keywords-software engineering; software quality; software testing; automated testing; research orientation.

CLC Number: TP311

Ⅰ.THE LATEST RESEARCH RESULTS AND HOT ISSUES ON SOFTWARE TESTING

With the research and development on component system testing, form-modeling and analysis techniques in embedded software and the software credibility, new results and hot issues on software testing keep emerging in recent years, part of which are introduced as follows.

A.The latest results of software testing.

a) Test-driven development (TDD)

The programming is guided by testing. Before coding, we should write the related test code at first, after that, program the test code, then test the develop code by testing program and the cycle continues, until the development is completed. The recent popular XP (Extreme Programming) mode strongly advocates this testing idea.

b) Iterative and incremental test.

It is evolved from the iterate model. After finishing the iteration, the system will incrementally integrate some new functions until the completion of the entire system function. It mainly focuses on the cumulative function in the regression test and each iterative cycle test is completed by two parts: incremental test on the current iterative product and the regression test on the completed function of former iterative cycle. This is one of IBM favorite test methods.

c)GUI automation test

GUIATF is an automated testing framework based on object-oriented capture technology for GUI. About the method of testing case generation, Fu Bo, from Beijing University of Aeronautics and Astronautics, and his mates put forward a kind of automatic generation method of test data based on ant colony algorithm [1]. By using bit coding, a model from input domain of the software under test to ant paths of the ant colony algorithm was established. This improved the diversity of ant paths and decreased the degrees of the precocity and stagnation. Jiang Zhongwei, from ZheJiang Sci-Tech University, applied the automatic generation idea of model-driven software code in MDA to the automatic generation of model-driven software testing cases and developed a framework which can automatically generate unit test case, thus improving the automation degree of the software testing [2].

d)Testability of component software

On the selection of the test case, Ma Liangli and his mates, from HuaZhong University of Science and Technology, applied a metadata selection method to select testing case [3]. It embedded the information and case to component in order to achieve the generation of testing case, and used the method of UML to show the relevant use case meta-model, testing case meta-model, the mapping between them, elements of the component metadata. Ma Zhen and his mates, from Xi'an University of Technology, proposed MTCGCBS, one of the GA, which introduced IGA and its advantages on the generation of component software testing case. In the mean time, they also raised an advanced method that is the "IIGA", which brought the parallel, migration, self-adoption and immune operator into traditional genetic algorithm, and also proofed its convergence [4]

e) Embedded Software Simulation Test

Sheng Yongqing and his mates, from TongJi University, can test various embedded software without significant modification by simulating the ARM embedded system on PC. YinYongfeng and his mates, from Beijing University of Aeronautics and Astronautics, designed a testing development environment for MVC-based embedded software, which not only ensure the successful development of ESTDE but improve the repeatability and adaptability of the system.

B.International and domestic research hotspots

a)The research about the usage of software testing techniques and methods directing to the features of software. Such as those researches about software testing techniques which aim at different types of software features like the real-time systems and embedded systems, etc.

b) The research about the software testing techniques directing to the new software development techniques, including the researches about the software testing techniques which aim at object-oriented technology, Internet structure, Java language and software component.

c) The research about testing automatic technique. It aims to improve the degree of automation in different steps of the testing and ease the burden of the workers, such as automatic generation of testing cases, automatic performance of regression tests, etc.

d)The research about testing tools and testing environment. Testing tools and testing environment should

be developed in line with the techniques and methods of software testing. Such as testing planning tool, testing designing tool, testing managing tool, static analysis tool, structure testing tool, regression testing tool, performance and load testing tool, and the improvement of effectiveness and interoperability of testing tools.

II THE CONCEPTION OF SOFTWARE TESTING DEVELOPMENT TREND

With the deepening of theoretical researches and industry practices, as well as the advancing of the software development techniques, the new theories, techniques, and industry of software testing will continue to develop. The ideas about the future of this field are as follows:

A. Establishing the testing conception directory of the field.

The testing conception directory lists the most possible testing conception for discovering the majority existing software faults. It can find out the main testing points of BUG and lay a foundation for writing testing cases. Like the case component, it is the sublimation or abstract for testing cases reusing.

a)The content of the conception directory.

For an inquiry function, the hypotheses of testing conception are as follows: 1) without preconditions, 2) an inquiry condition, 3) whether the fuzzy inquiry is supported, 4) whether the OR, ADN connections are available, 5) whether the results are exported, 6) Whether the results support sorting. This conception should be recorded, used and completed in the later testing and writing.

b) The structure of the conception directory.

Aiming at certain depth and breadth, a good testing conception should be easy to read (to skim), simple to search and flexible to use. You can easily find the things you want, and get the necessaries, ignore the unnecessaries.

B. Establishing a complete set of software testing automation system

A complete set of software testing automation system is a powerful integrated test environment, and as a part of software to be consummated and configured. It can automatically produce or restore the required stub code (drive, stub program and simulator) and generate the most appropriate test case, carry out and finally release the test report.

a)Automatic generation of test data. The ability of automatic generation should be strong, it can not only fit for the automatic generation of test data of white-box or black-box, but also for the automatic generation of asserted test data and regression test data.

b)Automatic generation of test oracle and expected result. Test oracle is a method to detect whether the system under test can run correctly or not. The expected result is used to judge the success or failure of the testing case; it is the deserved output of the program in accordance with the input.

c) Simplifying the regression testing case set. In order to reduce the cost of the regression testing, in the premise of quality assurance, the number of test cases which need to run under the regression test should be reduced. Hence, we should study how to simplify regression testing case set to meet the same test target and to save the cost for analyzing the test cases about which need to re-run. It is meaningful to reduce the cost of analyzing.

C.Establishing universal testing theories

Universal testing theories are to build reliable and comprehensive theories to support and enrich testing techniques, the requirements are as follows:

a)Clear testing hypothesis

It is necessary to clearly describe assumptions of different testing techniques, namely the testing hypothesis. Testing hypothesis can judge universal and visual testing activities. Many hypotheses of universal testing methods are homogeneity assumption (Assume that the software runs equally in each testing domain.) and regularity assumption.

b)Testing efficiency evaluation

The effectiveness of existing testing criterions and new testing criterions should be evaluated for the sake of establishing useful testing theories. In order to know the effectiveness of criterions in different defect classifications, the analytical, statistical and experimental data getting from additional studies should be provided to judge the effectiveness in revealing breakdowns.

c)Combination Testing

In the past, many researches concerned about the testing strategy techniques and tools in organizing and executing component aggregations. Component based development method and dynamic system combination method make the problem more and more evident. We need to know how to reuse the test results obtained in testing independent components (units or parts or subsystems), particularly what conclusions can be obtained from a system constituted by components, which additional test case still need to run in an integrated system.

D.Testing-based modeling study

From the perspective of a tester, testing-based modeling availably test the software through structuring a model rather than use a model to test, that is to say, the current model-based testing should be changed into testing-based modeling.

a)Testing-based modeling

Testing-based modeling can improve controllability and observability of the software. Unlike the model-based testing method, it can directly analysis the program execution results, collect information from the active or passive program execution, assemble some related attribute data and behavior.

b)Test oracle

Test oracle and test plan are closely related. We should define how to generate test case, give clear desired output to each test case, judge whether the software is qualified or unqualified through the observation of the test output. However, the precision and effectiveness of test oracle have severely impacted test costs and results, we should study on automated testing implementation method and integrate all available information.

E.Testing efficiency maximization research

The biggest obstacle for testing efficiency maximization is the increasing complexity of modern system. The complexity affects not only the system itself, but also its configuration environment.

a)Test overhead issues

Due to the high cost of regression testing, effective techniques should be adopted to reduce the quantity of regression testing, while the regression testing case should be classified and automatically executed. As another example, the operating data can be collected from the running program with few system overhead, thus extracting relevant information effectively. Furthermore, the collected data can be used to strengthen and improve the internal tests and maintenance activities.

b) Test costs issues

It is helpful for the testers to select the most appropriate test method through the combination of test costs with test process and techniques as well as the integration of economic value and test process. For carrying on the effective research, the direct and indirect costs of software testing techniques should be quantified; meanwhile, the estimated functional cost should be united with the efficiency ratio of existing testing technology.

c)Testers' skill issues

Whether the test is success or not depends on the testers' skills, determinations and motivations. Testers can know the basic conceptions, limitations and feasibilities of existing technologies via training. It is conducive to enhance the test maturity of the team by building technical training process.

F.Establishing scientific system for software testing education

Domestic universities and colleges should join hands with training centers. Besides the cultivation of the software testing technicians, universities and colleges should introduce testing practice simultaneously. Software workers' experience summaries are far from enough in terms of software testing theories and techniques, innovations of testing management and developments of testing tools, it can not rise to a height to guide the development of software testing industry, the due obligation should be undertaken by government departments, universities and scientific research institutions.

III.Conclusion

Software testing has become an integral part of the quality assurance of modern software, and software testing research is even the driven element of this part's development and application. In face of newer and higher demand of software testing, it is necessary to constantly summarize new achievements, study fresh hotspots and propose different ideas in order to promote the study on software testing system engineering, to facilitate the rapid development on software testing field and industry.

References

[1] Fu Bo, Automatic Generation Method of Test Data Based on Ant Colony Algorithm, Computer Engineering and Applications, 2007.43(12).

[2] Jiang Zhongwei, Zhang Yunhua, Xie Xuan'ang, The Generation of Software Testing Case Based on MDA, Computer Engineering and Applications, 2007.43(17).

[3] Ma Liangli, Lv Yansheng, Liu mengren, Research on Application of Metadata in Component Software Test Case, Mini-Micro Systems, 2006.12.

[4] Shan Jinhui, Jiang Ying, Sun Ping, the Research Development of Software Testing [J]. Journal of Peking University (Natural Science Edition), 2005, 41(1):134-145.

About the Author/Contact：

ZHANG Hong Chung，Male，Associate Professor，Master. Information Technology School, Hubei Polytechnic Institute, 432000,China，Direction of Research: Software Engineering.

E-Mail：zhang189589@163.com