International Journal of Information Technology and Knowledge Management
January June 2009, Volume 2, No. 1, pp. 167-171

EFFECTIVENESS OF TESTING PROCESSES FOR MODERN SOFTWARE DEVELOPMENT

Rajender Bathla[*], Shallu Miglani[**], Vivek[***] & Sukhdev Singh[****]

A process is a significant concept in modern software development. Only when the activities are organized in process descriptions, can these be communicated, followed, observed, and improved. The basis for understanding what testing is, is therefore the understanding of the testing process. This paper presents the general concept of a process, and expands on this to present a suggestion for a generic testing process. For each of the activities in this generic process this paper presents suggestions for detailed activities, meant to serve as inspiration for further work on a truly generic testing process.

1. Introduction

This is an attempt to look at the subject of testing and modern soft- ware development methodologies in the context of critical systems. The demands of testing will be necessarily high and so methodologies can be viewed in the way that they contribute towards 'good' testing. It is then suggested **as** to how testing theory my be used to drive methodology development.

It is interesting to consider why software developers employ the soft- ware development methods that they do. There might be a variety of reasons for this: 'It's the one we used last time', 'It's the one that we're trained in', 'The customer wants us to', and so on. Very infrequently is the answer heard, 'Because it provides a sound basis for testing our implementation'. It is then even more interesting to **ask** developers how they use the software development methodologies to test the implementations. Is it a process which arises naturally out of the methodology in which the developers have confidence or is it an ad. hoc. Approach which has to wrestle with the documentation a methodology produces to extract the necessary information?[4] In order to examine software development methodologies it is useful to examine the way in which they aid 'good' testing. A framework in which to do this is the testing theory proposed by Goodenough and Gerhart [10] and developed by Weyuker and Ostrand [11]. The theory defines an ideal test which, if found and successfully executed, constitutes a proof of correctness of the program. Finding an ideal test is not a trivial task. However, the theory is useful because it may be used to evaulate the extent to which methodologies go to support the production of an ideal test. Everything we do, from cooking a meal to producing the most complicated software products, follows a process. A process is a series of activities performed to fulfill a purpose and produce a tangible output based on a given input. The process view on software development is gaining more and more interest. Process models are defined to assist organizations in process improvement; that is, in making their work more structured and efficient.

2. The Concept of a Process

A process is a series of activities to be performed to fulfill a specific purpose. Based on an input and following a number of activities a tangible output is produced.

It is important to remember that the tangible output (for example a specification) is not the goal itself. The goal is to perform the activities, to think, to discus, to try things out, to make decisions, to document, and what ever else is needed. The tangible output is only the way of communicating how the purpose of the process has been fulfilled.

Processes can be described and hence monitored and improved.

Based on [1], one of the earliest books about process improvement, a process description must always as a minimum include:

- A definition of the input;
- A list of activities – the procedure;
- A description of the output.

In the basic description of a process the purpose is implicitly described in the list of activities. Since then a lot of work has been done on understanding and describing the concept

[*] M.Tech (IT) Student of M. M. University Mullana, Haryana (India). *E-mail: dr.bathla@gmail.com*

[**] M.Tech (CSE) Final Year Student of JMIT, Radaur, Haryana (India). *E-mail: shallu.miglani@gmail.com.*

[***] Deptt. of Computer Science & Engg., HCTM, Kaithal, Haryana (India).

[****] Deptt. of Information Technology, HCTM, Kaithal, Haryana (India). *E-mail: sukhdev_kuk@rediffmail.com*

of a process. In Europe and Canada work has started on an ISO standard for process definition, so far [2] has been issued. These guidelines characterize the following attributes of process description:

- Title;
- Purpose;
- Outcomes
- Activities
- Tasks

In the United States of America CCMI© (Capability Maturity Model Integration) is being updated and it is gaining awareness and deployment around the globe. In [3] it is specified that a process consists of set of activities that may have the following attributes:

- Process roles; Applicable standards;
- Applicable procedures, methods, tools, and resources; Process-performance objectives;
- Entry criteria; , Inputs;
- Product and process measures to be collected and used;
- Verification points;

This seems to be a more comprehensive and not least, more useful process description than the one suggested in [2]. I'm personally surprised that [2] does not mention input. In any case a process description must be operational. It is not supposed to fill pages and pages. It should ideally fit on a single page, maybe even a web page, with references to more detailed descriptions of methods, techniques, and templates.

2.1 Processes Depend on Each Other

Processes may be assembled in process hierarchies or process architectures. In this case, the input to a process must be the output from one ore more proceeding process except perhaps for the very first, where the infamous napkin with the original idea is the input.

Likewise, the output from a process must be the input to one or more other processes. Processes hence do not include information about what their proceeding and/or subsequent processes may be. The only connection between processes is the information they produce and use. The dependencies between processes can be depicted in a process model, where it is shown how Outputs from processes serve as inputs to other processes.

2.2. An Overall Generic Test Process

Testing can be described as a process. An overall generic test process could be described like this, using the most basic description and only listing input and output in terms of work product:

The purpose of the test process is to provide

Information to assure the quality of the product, decisions, and the processes for a testing assignment.

Software testing at our manuf facility is at the end of the typical testing process

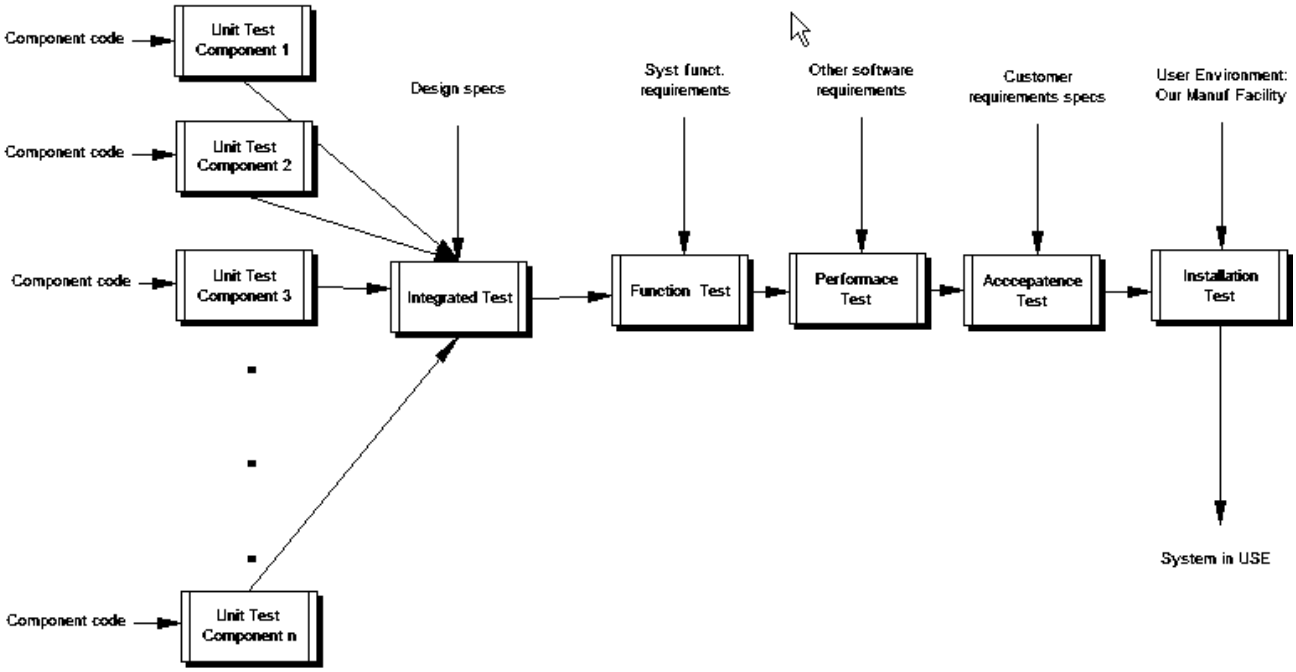

Figure 1: Testing Process Model

The inputs on which this process is based are:

- test strategy, Project Plan, Master test plan, information about how the testing is progressing. The activities are:
- initial test planning, monitoring, control, and re-planning
- test design and development, test execution
- test evaluation and reporting, test closure activities. The outputs being produced are
- specific test plan, test specification, test environment specification. Actual test environment including test data
- test logs, progress report, test summary report, test experience report

2.3 Iterative Generic Test Process

There are a few points that need to be understood for the generic testing process:

The test activities need not be performed in strict sequential order. Test monitoring, control, and re- planning are constant activities in the sense that they are not just done once in the beginning of the test assignment. Monitoring of the process should be done on an ongoing basis, and controlling and re-planning activities performed when the need arises. A model is not a scientific truth; when using a model, even a very well defined model, we should be open for necessary tailoring to specific situations [2].

The generic test process is iterative - not a simple straight forward process. It must be foreseen that we'll have to perform the activities more than once in an iterative way before the exit criteria have been fulfilled. The iterations to be foreseen in the test process are shown in Figure 1. My personal experience, supported by the experience of other consultants, shows that in most cases 3 iterations must be carried out as a minimum before the test process can be completed. The first activity from which an iteration may occur [8] is the test execution. This is where we detect the failures, when the actual result is different from the expected [5]. The resulting iterations may be:

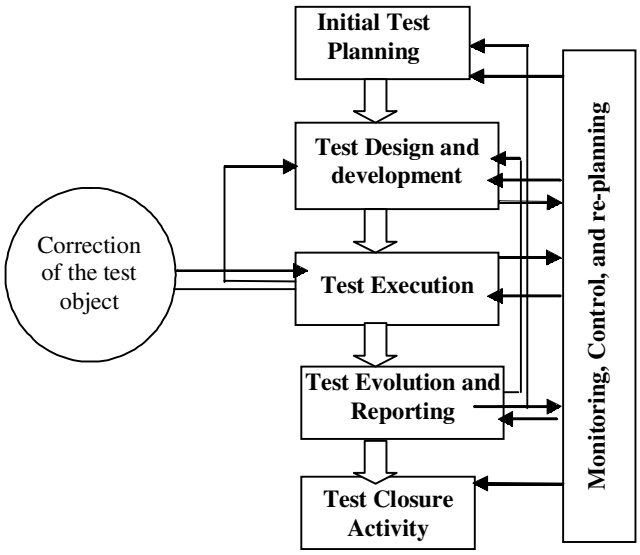

Figure 2. Iterative Generic Test Process

Example: A calculation does not give the expected result, and it appears that the algorithm for the calculation has been coded wrongly. When the defect has been corrected we must confirmation test the software using the test procedure that encountered the failure in the first place. Well probably also perform some regression testing.

(2) The defect is in the test procedure Example: A calculation does not give the expected result, but here it appears that the test case was wrong. The defect must be corrected and the new test case(s) must be executed. This iteration usually goes back to the design and development activity. The second activity from which an iteration may occur is the evaluation of the exit criteria. This is where we find out if the exit criteria are not met. The resulting iteration in this situation may be:

(3) More test cases must be specified to increase coverage, and these must then be executed. Example: In the evaluation it turns out that the decision coverage for a component is only 87%. One more test case is designed and when this is executed the coverage reaches 96%. The exit criteria are relaxed or strengthened in the test plan.

Example: The coverage is found to be too small because of an error handling routine that is very hard to reach. The required coverage for the component is relaxed to 85%.

3. ACTIVITIES IN TESTING SUB-PROCESSES

Each of the activities in the generic testing process may be expanded to a (sub-)process in its own right. For each of these, the process can be described in the same way as the overall testing process, at least including a definition of the input, a list of activities, and a description of the output. Other information may of course also be provided[8]. In the following, only purpose and lists of suggested activities for each of the six testing sub- processes are provided. The lists are a result of a workshop held at Test Management Summit in London, January 2008. The lists are by no means correct or complete, but may be used as inspiration for further work on defining a generic testing process.

3.1 Initial Test Planning

The purpose of the initial test planning process is to verify the mission of the testing, to define the decisions to transform

the test strategy into an operational plan for the performance of the actual testing task at hand[7].

The activities may include:

- verification of the mission of the test assignment definition of the objectives of the testing, including coverage targets
- production of work-break down structure, identification of deliveries
- estimation of activities, production of budget
- specification of resources, including training needs
- identification of environmental needs
- coordination with stakeholders
- risk management, documentation of decisions (in test plan), motivation of staff, communication with stakeholders

3.2 Test Monitoring, Control, and Re-Planning

The short story of monitoring and control is that if it looks as if we are going to end up in the all too familiar situation illustrated here, we have to take precautions.

The purpose of the monitoring, control, and re-planning process is to stay in control and make necessary corrections to the plan when it no longer reflects the reality as the testing activities are performed and the test assignment progresses.

The activities may include:

- collection of measurements, analysis of measurements, continuous risk management.
- re-estimation ,implementation of corrective actions as appropriate
- re-delegation of tasks, motivation of staff, teaching staff
- participate in process improvement activities
- configuration management of test planning documentation

3.3. Test Design and Development

The purpose of the test design and development process is to design and write tests that provide the largest possible coverage to meet the coverage demands in the test plan. Furthermore it includes the detailed definition and establishment of the test environment. The work in this process is highly iterative.

The activities may include:

- reading/studying test basis to understand it, definition of structure of test specification
- identification of test conditions
- selection of test technique(s) according to the approach specified in the plan
- identification of defects in the test basis (e.g., requirements)
- reporting of identified defects, development of detailed test cases, identification of test data.
- registration of test coverage, feedback to test monitoring
- mentoring of junior testers, evaluation of tools, establishment of test environment
- extraction and possibly masking and/or creation of test data

3.4 Test Execution

The purpose of the test execution is to perform the physical test in the correct environment, and register the execution and the results.

The activities may include:

- reception of test object, exploratory testing
- performance of smoke test (readiness)
- production of test execution sequence (detailed planning / procedures)
- execution of tests, analyzing observations
- logging of test execution course
- reporting of incidents, including those found in the test ware, confirmation testing
- specification of regression testing need, regression testing
- communication to stakeholders, maintain test environment
- resetting test environment to original state (especially test data)
- mentoring/training/teaching junior testers
- mentoring/training/teaching non-testers (users and others participating in test execution)
- feedback to test monitoring, configuration management of test ware

3.5 Test Evaluation and Reporting

The purpose of the test evaluation and reporting progress is to ensure that the test objective, including the coverage target has been achieved and communicate the results of the test assignment in such ways that they are understandable and useful for the stakeholders [2].

The activity may include :

- checking results against objectives (e.g., achieved coverage and outstanding defects)
- feedback to monitoring, production of reports to stakeholders

3.6 Test Closure

The purpose of the test closure process is to clean up after the test assignment and secure valuable physical assets and information in the organization.

The activities may include:

- clearing out unnecessary test environments, including test data
- hand-over of test ware as applicable, performing retrospection meeting
- identification of process improvement candidates, communication to stakeholders

4. CONCLUSION

The establishment of a generic test process is essential to the future of testing. Only when the performers of a profession know the basic activities and hence what is expected of them, can they work at learning their trade and constantly improve their skills. Only when organizations define the ways they want their employees to work, are they able to monitor what is going on and make corrective actions and continuous improvement of the processes. It is however not a simple task. The understanding of what a process description consists of is still young and incomplete. Since this forms the basic for the work, an agreement must first be reached as to how to make the description of a generic test process. The people involved in testing come from many different backgrounds, many are still more or less autodidact, and they have many different experiences in their baggage. Testing is performed in organizations spanning from rocket science to toys and from the most sophisticated technology to the most sophisticated banking and administration. The test activities are described in many places, like: national and international standards, branch specific guidelines and regulations, syllabi, software development models, books, paper and so forth. There is a lot of work ahead; this makes it even more important to get going.

References

[1] Watts S. Humphrey, *Managing the Software Process*, Addison-Wesley, (1989).

[2] ISO/IEC DTR 24774 - *System and Software Engineering Life Cycle Management Guidelines for Process Description*, (2006).

[3] Mary Beth Chrissis, Mike Konrad, and Sandy Shrum, CMMI © Second Edition – *Guidelines for Process Integration and Product Improvement*, Addison-Wesley, (2007).

[4] Burnstein, Ilene, Practical Software Testing. Springer Verlag New York Inc., USA, (2003).

[5] Beizer, B. Software Testing Techniques. Second Edition,Van Nostrand Reinhold Company Limited, (1990).

[6] Gill, G. and C. Kemerer. "Cyclomatic Complexity Density and Software Maintenance Productivity". *IEEE Transactions on Software Engineering*, (December 1991).

[7] Gerald, D. Everett *et al*. Software Testing: Testing Across the Entire Software Development Life Cycle. John Wiley & Sons, Inc. Publication, New Jersey, (2007).

[8] T. Chow, "Testing Software Design Modeled by of the *IEEE International Conference on Automated Software Engineering*, Montreal, Quebec.

[9] Mosley, Daniel. *Just Enough Software Test Automation* (2002).

[10] Good Enough, J. B. and Gerhart, S. L. 'Toward a Theory of Test Data Selection' *IEEE TSE* , SE-1**,** (June 1975).

[11] Weyuker, E. J. and Ostrand, T. J. 'Remarks on the Theory of Test Data Selection' Digest for the Worlcshop on Software Testing and Test Documentation, Ft. Lauderdale, Fla. *IEEE* (1978).