National Research Journal of Information Technology & Information Science
Volume-1, Issue-1, Year-2014 PP: 1-8

A Short Note on Test Maturity Model Integrated Technique in Project Assessment

Vishwa Nath Maurya*
Head, Department of Applied Mathematics & Statistics, School of Science & Technology
The University of Fiji, Fiji Islands

Avadhesh Kumar Maurya**
Head, Department of Electronics and Communication Engineering
Lucknow Institute of Technology, G.B. Technical University, Lucknow, India

Dibyanshu Singh, Anurag Priyadarshi, Gaurav Anand, and Nida Ateeq
Department of Computer Science & Engineering, Sapthagiri College of Engineering, Bangalore
Visvesvaraya Technological University, Belgaum, India

Abstract

Present paper demonstrates an application aspect of maturity model integrated technique for project assessment in multinational software companies. In this paper, test maturity model integration (TMMi) criterion has been used to improve testing process in software organizations. Many organizations find value in benchmarking their progress in test process improvement for not only internal purposes but also for external customers and suppliers. The TMMi provides an excellent reference model to be used during such assessments. Assessment teams use TMMi to guide their identification and prioritization of findings. These findings along with the guidance of TMMi practices are used to plan improvements for the organization. This application helps in evaluating projects under various companies using TMMi Levels and Standards and hence, generating reports in form of graphs showing the areas that need to have improvement.

Keywords: Project assessment; capability maturity model; TMMi; CMMi; test process improvement; windows; MySQL; PHP; Perl or Python; hypertext preprocessor; hypertext markup language; extended markup language; cascading style sheets; quality assurance.

1. Introduction

For the past decade, the software industry has invested substantial effort to improve the quality of its products. This has been a difficult job, since the size and complexity of software increases rapidly while customers and users are becoming more and more demanding. Despite encouraging results with various quality improvement approaches, the software industry is still far from zero defects. To improve product quality, the software industry has often focused on improving its development processes. A guideline that has been widely used to improve the development processes is the Capability Maturity Model. The Capability Maturity Model (CMM) and its' successor the Capability Maturity Model Integration (CMMI) are often regarded as the industry standard for software process improvement. Despite the fact that testing often accounts for at least 30-40% of the total project costs, only limited attention is given to testing in the various software process improvement models such as the CMM and the CMMI. As an answer, the testing community has created its own improvement models. The TMMi is a detailed model for test process improvement and is positioned as being complementary to the CMMI.

2. Background

The TMMi framework has been developed by the TMMi Foundation as a guideline and reference framework for test process improvement and is positioned as a complementary model to the CMMI addressing those issues important to test managers, test engineers and software quality professionals. Testing as defined in the TMMi is applied in its broadest sense to encompass all software product quality-related activities. Testing is the process consisting of all lifecycle activities, static and dynamic, concerned with planning, preparation and evaluation of software products and related work products to determine that they satisfy specified requirements, to demonstrate that they are fit for purpose and to detect defects. Just like the CMMI staged representation, the TMMi also uses the concept of maturity levels for process evaluation and improvement. Several noteworthy researchers (e.g. [1], [2], [3]….[23]) and references therein confined their attention in different versions of the project assessment and module testing. Furthermore process areas, goals and practices are identified. Applying the TMMi maturity criteria will improve the test process and have a positive impact on product quality, test engineering productivity, and cycle-time effort. The TMMi has been developed to support organizations with evaluating and improving their test process. Within the TMMi, testing evolves from a chaotic, ill-defined process with a lack of resources, tools and well-educated testers to a mature and controlled process that has defect prevention as its main objective. Practical experiences are positive and show that TMMi supports the process of establishing a more effective and efficient test process. Testing becomes a profession and a fully integrated part of the development process. As stated the focus of testing changes from defect detection to defect prevention.

3. Sources and Test Levels

The development of the TMMi has used the TMM3 framework as developed by the Illinois Institute of Technology. In addition to the TMM, it was largely guided by the work done on the Capability Maturity Model Integration (CMMI), a process improvement model that has widespread support in the IT industry. The CMMI has both a staged and continuous representation. Within the staged representation the CMMI architecture prescribes the stages that an organization must proceed through in an orderly fashion to improve its development process. Within the continuous representation there is no fixed set of levels or stages to proceed through. An organization applying the continuous representation can select areas for improvement from many different categories. The TMMi has been developed as a staged model. The staged model uses predefined sets of process areas to define an improvement path for an organization. This improvement path is described by a model component called a maturity level. A maturity level is a well-defined evolutionary plateau towards achieving improved organizational processes.

Whereas some models for test process improvement focus mainly on higher test levels, e.g., Test Process Improvement (TPI), or address only one aspect of structured testing e.g., the test organization, the TMMi addresses all test levels (including static testing) and aspects of structured testing. With respect to dynamic testing, both lower test level (e.g., component test, integration test) and higher test levels (e.g., system test, acceptance test) are within the scope of the TMMi. Studying the model more in detail one will learn that the model addresses all four cornerstones for structured testing (lifecycle, techniques, infrastructure and organization).

4. Description of the TMMi and CMMi

It is also important to note that TMMi is positioned as a complementary model to the CMMi. In many cases a given TMMi level needs specific support from process areas at its corresponding CMMI level or from lower CMMI levels. In exceptional cases there is even a relationship to higher CMMi levels. Process areas and practices that are elaborated within the CMMI are mostly not repeated within TMMi; they are only referenced. For example the process area configuration management, which is also applicable to test (work) products / test ware, is not elaborated upon in detail within the TMMi; the practices from CMMi are referenced and implicitly re-used. Many organizations find value in benchmarking their progress in test process improvement for both

internal purposes and for external customers and suppliers. Test process assessments focus on identifying improvement opportunities and understanding the organization's position relative to the selected model or standard. The TMMi provides an excellent reference model to be used during such assessments. Assessment teams use TMMi to guide their identification and prioritization of findings. These findings along with the guidance of TMMi practices are used to plan improvements for the organization. For more details, we refer [21], [22] & [23].

5. Overview of the Present Work

TMMi has a staged architecture for process improvement as shown in Figure 1.1. It contains stages or levels through which an organization passes as its testing process evolves from one that is ad hoc and unmanaged, to one that is managed, defined, measured, and optimized. Achieving each stage ensures that an adequate improvement has been laid as a foundation for the next stage. The internal structure of the TMMi is rich in testing practices that can be learned and applied in a systematic way to support a quality testing process that improves in incremental steps. There are five levels in the TMMi that prescribe a maturity hierarchy and an evolutionary path to test process improvement. Each level has a set of process areas that an organization needs to implement on to achieve maturity at that level. Experience has shown that organizations do their best when they focus their test process improvement efforts on a manageable number of process areas at a time, and that those areas require increasing sophistication as the organization improves. Because each maturity level forms a necessary foundation for the next level, trying to skip a maturity level is usually counter-productive. At the same time, you must recognize that test process improvement efforts should focus on the needs of the organization in the context of its business environment and the process areas at higher maturity levels may address the current needs of an organization or project.

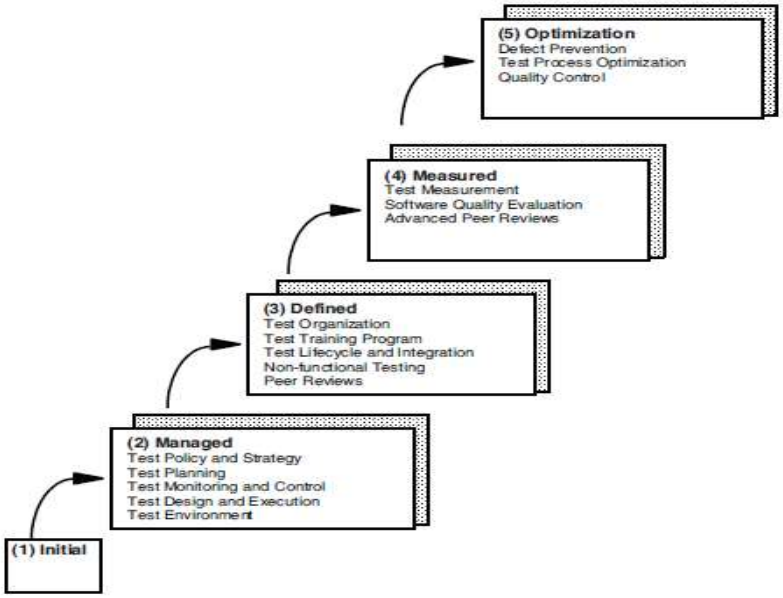

Figure 1.1: TMMi maturity levels and process areas

5.1 Level 1: Initial

At TMMi level 1, testing is a chaotic, undefined process and is often considered a part of debugging. The organization usually does not provide a stable environment to support the processes. Success in these organizations depends on the competence and heroics of the people in the organization and not the use of proven processes. Tests are developed in an ad hoc way after coding is completed. Testing and debugging are interleaved to get the bugs out of the system. The objective of testing at this level is to show that the software runs without major failures. Products are released without adequate visibility regarding quality and risks. In the field, the product often does not fulfill its needs, is not stable, and/or is too slow. Within testing there is a lack of resources, tools and well-educated staff. At TMMi level 1 there are no defined process areas. Maturity level 1 organizations are characterized by a tendency to over commit, abandonment of processes in a time of crises, and an inability to repeat their successes. In addition products tend not to be released on time, budgets are overrun and delivered quality is not according to expectations.

5.2 Level 2: Managed

At TMMi level 2, testing becomes a managed process and is clearly separated from debugging. The process discipline reflected by maturity level 2 helps to ensure that existing practices are retained during times of stress. However, testing is still perceived by many stakeholders as being a project phase that follows coding. In the context of improving the test process, a company-wide or program-wide test strategy is established. Test plans are also developed. Within the test plan a test approach is defined, whereby the approach is based on the result of a product risk assessment. Risk management techniques are used to identify the product risks based on documented requirements. The test plan defines what testing is required, when, how and by whom. Commitments are established with stakeholders and revised as needed. Testing is monitored and controlled to ensure it is going according to plan and actions can be taken if deviations occur. The status of the work products and the delivery of testing services are visible to management. Test design techniques are applied for deriving and selecting test cases from specifications. However, testing may still start relatively late in the development lifecycle, e.g., during the design or even during the coding phase. In TMMI level 2 testing are multi-leveled: there are component, integration, and system and acceptance test levels. For each identified test level there are specific testing objectives defined in the organization-wide or program-wide test strategy. The processes of testing and debugging are differentiated. The main objective of testing in a TMMi level 2 organizations is to verify that the product satisfies the specified requirements. Many quality problems at this TMMi level occur because testing occurs late in the development lifecycle. Defects are propagated from the requirements and design into code. There are no formal review programs as yet to address this important issue. Post code, execution-based testing is still considered by many stakeholders the primary testing activity. The process areas at TMMi level 2 are:

- Test Policy and Strategy
- Test Planning
- Test Monitoring and Control
- Test Design and Execution
- Test Environment

5.3 Level 3: Defined

At TMMi level 3, testing is no longer confined to a phase that follows coding. It is fully integrated into the development lifecycle and the associated milestones. Test planning is done at an early project stage, e.g., during the requirements phase, and is documented in a master test plan. The development of a master test plan builds on the test planning skills and commitments acquired at TMMi level 2. The organization's set of standard test processes, which is the basis for maturity level 3, is established and improved over time. A test organization and a specific test training program exist, and testing is perceived as being a profession. Test process improvement is fully institutionalized as part of the test organization's accepted practices. Organizations at level 3 understand

the importance of reviews in quality control; a formal review program is implemented although not yet fully linked to the dynamic testing process. Reviews take place across the lifecycle. Test professionals are involved in reviews of requirements specifications. Whereby the test designs at TMMi level 2 focuses mainly on functionality testing test designs and test techniques are expanded at level 3 to include non-functional testing, e.g., usability and/or reliability, depending on the business objectives. A critical distinction between TMMi maturity level 2 and 3 is the scope of the standards, process descriptions, and procedures. At maturity level 2 these may be quite different in each specific instance, e.g., on a particular project. At maturity levels 3 these are tailored from the organization's set of standard processes to suit a particular project or organizational unit and therefore are more consistent except for the differences allowed by the tailoring guidelines. Another critical distinction is that at maturity level 3, processes are typically described more rigorously than at maturity level 2. As a consequence at maturity level 3, the organization must revisit the maturity level 2 process areas. The process areas at TMMi level 3 are:

- Test Organization
- Test Training Program
- Test Lifecycle and Integration
- Non-functional Testing
- Peer Reviews

5.4 Level 4: Measured

Achieving the goals of TMMi level 2 and 3 has the benefits of putting into place a technical, managerial, and staffing infrastructure capable of thorough testing and providing support for test process improvement. With this infrastructure in place, testing can become a measured process to encourage further growth and accomplishment. In TMMi level 4 organizations, testing is a thoroughly defined, well-founded and measurable process. Testing is perceived as evaluation; it consists of all lifecycle activities concerned with checking products and related work products. An organization-wide test measurement program will be put into place that can be used to evaluate the quality of the testing process, to assess productivity, and to monitor improvements. Measures are incorporated into the organization's measurement repository to support fact-based decision making. A test measurement program also supports predictions relating to test performance and cost. With respect to product quality, the presence of a measurement program allows an organization to implement a product quality evaluation process by defining quality needs, quality attributes and quality metrics. (Work) products are evaluated using quantitative criteria for quality attributes such as reliability, usability and maintainability. Product quality is understood in quantitative terms and is managed to the defined objectives throughout the lifecycle. Reviews and inspections are considered to be part of the test process and are used to measure product quality early in the lifecycle and to formally control quality gates. Peer reviews as a defect detection technique is transformed into a product quality measurement technique in line with the process area Product Quality Evaluation. TMMi level 4 also covers establishing a coordinated test approach between peer reviews (static testing) and dynamic testing and the usage of peer reviews results and data to optimize the test approach with both aiming at making testing more effective and more efficient. Peer reviews are now fully integrated with the dynamic testing process, e.g. part of the test strategy, test plan and test approach. The process areas at TMMi level 4 are:

- Test Measurement
- Product Quality Evaluation
- Advanced Peer Reviews

5.5 Level 5: Optimization

The achievement of all previous test improvement goals at levels 1 through 4 of TMMi has created an organizational infrastructure for testing that supports a completely defined and measured process. At TMMi

maturity level 5, an organization is capable of continually improving its processes based on a quantitative understanding of statistically controlled processes. Improving test process performance is carried out through incremental and innovative process and technological improvements. The testing methods and techniques are optimized and there is a continuous focus on fine tuning and process improvement. An optimized test process, as defined by the TMMi is one that is:

- managed, defined, measured, efficient and effective
- statistically controlled and predictable
- focused on defect prevention
- supported by automation as much is deemed an effective use of resources
- able to support technology transfer from the industry to the organization
- able to support re-use of test assets
- focused on process change to achieve continuous improvement.

To support the continuous improvement of the test process infrastructure, and to identify, plan and implement test improvements, a permanent test process improvement group is formally established and is staffed by members who have received specialized training to increase the level of their skills and knowledge required for the success of the group. In many organizations this group is called a Test Process Group. Support for a Test Process Group formally begins at TMMi level 3 when the test organization is introduced. At TMMi level 4 and 5, the responsibilities grow as more high level practices are introduced, e.g., identifying At TMMi level 5, the Test Process Optimization process area introduces mechanisms to fine-tune and continuously improve testing. There is an established procedure to identify process enhancements as well as to select and evaluate new testing technologies. Tools support the test process as much as is effective during test design, test execution, regression testing, test case management, defect collection and analysis, etc. Process and test ware reuse across the organization is also common practice and is supported by a test (process) asset library. The three TMMi level 5 process areas, Defect Prevention, Quality Control and Test Process Optimization all provide support for continuous process improvement. In fact, the three process areas are highly interrelated. TMMi level 5, testing is a process with the objective of preventing defects. The process areas at TMMi level 5 are:

- Defect Prevention
- Quality Control
- Components of the TMMi

6. Maturity Levels and Organizational Test Process

A maturity level within the TMMi can be regarded as a degree of organizational test process quality. It is defined as an evolutionary plateau of test process improvement. Each level progressively develops an important part of the organization's test processes. There are five maturity levels within the TMMi. Each maturity level tells what to implement in order to achieve the given level. The higher the maturity level the organization achieves, the more mature the test process of the organization is. To reach a particular maturity level, an organization must satisfy all of the appropriate goals (both specific and generic) of the process areas at the specific level and also those at earlier maturity levels. Note that all organizations possess a minimum of TMMi level 1, as this level does not contain any goals that must be satisfied.

As stated with the exception of level 1, each maturity level consists of several process areas that indicate where an organization should focus to improve its test process. Process areas identify the issues that must be addressed to achieve a maturity level. Each process area identifies a cluster of test related activities. When the practices are all performed a significant improvement in activities related to that area will be made. In the TMMi, only those process areas that are considered to be key determinants of test process capability are identified. All process

areas of the maturity level and the lower maturity levels must be satisfied to consider a maturity level to be achieved [23].

7. Organizational Problem Statement and Objectives

To take steps that are measurable and that score the existing maturity using TMMi foundation standards, Benchmark the test maturity of any organization against the industry standards, and help organizations meet the TMMi standards. More and more organizations are trying to improve their software development processes. The reference improvement model that is most often used is the Capability Maturity Model Integration (CMMI). In this model process areas related to testing, verification and validation, are described. But the level of detail is limited, especially from the viewpoint of the test professional. To fill in this gap the Test Maturity Model (TMM) has been developed, which has recently been succeeded by the Test Maturity Model Integration (TMMi). This application helps the organization to meet the requirements needed to be able to achieve TMMi standards. It evaluates the projects under given companies, i.e. which TMMi level a project belongs to and reports are generated stating whether a certain project needs improvement. If so, then the areas a project can improve upon are specified.

8. Conclusive Observations and Future Enhancement

The TMMi maturity criteria for enhancement of testing process in software organizations have been demonstrated. Through the present paper it is observed that the TMMi criteria is significantly useful to improve the testing process and has a positive impact on product quality, test engineering productivity, and cycle-time effort. In addition to this, we remark here that the TMMi criteria proposed in this paper is quite applicable to support software organizations for evaluating and improving test process. Its application helps in evaluating the given projects in a company against TMMi standards and finding areas that still need to be improved upon. The reports generated in form of graphs helps in understanding and evaluating a project in its testing phase.

- Features to grade companies against the TMMi standards and giving it a TMMi level based on the project's assessment can be added.
- Having more than one assessor and more than one employee for a project.
- Integrating the project on a cloud so that it can be available to various organizations and companies.

Acknowledgements

The present paper is an integral part of the project work entitled "Project Assessment Using Test Maturity Model Integration (TMMi)"; submitted in partial fulfillment for the award of Bachelor of Engineering in Computer Science and Engineering of Visvesvaraya Technological University, Belgaum, India during the academic year 2012-13.The last four authors of this paper would like to express their immense gratitude to their Supervisor Prof. (Dr.) Vishwa Nath Maurya, Head, Department of Applied Mathematics & Statistics, School of Science & Technology, The University of Fiji, Fiji & Ex Professor & founding Director, Vision Institute of Technology, Aligarh (G.B. Technical University, Lucknow, India) and Co-Guides Mrs. Poornima G.J., Assistant Professor of Department of Computer Science & Engineering, Sapthagiri College of Engineering, Bangalore and A/Prof. (Er.) Avadhesh Kumar Maurya, Assistant Professor & Head, Department of Electronics & Communication Engineering, Lucknow Institute of Technology (G.B. Technical University, Lucknow) for their valuable guidance and sincere advice, and regular assistance throughout the project work. The last four authors are also grateful to Dr. Puttamadappa C., Principal, Sapthagiri College of Engineering Bangalore, Dr. C.M. Prashanth, Professor & Head, Department of Computer Science and Engineering, Sapthagiri College of Engineering, Bangalore for his help and inspiration during the tenure of the B.Tech. Program.

REFERENCES

1. Fewster, Mark and Graham, Dorothy, Software test automation effective use of test execution tools. Harlow : Addison-Wesley, 0-201-33140-3, 1999.
2. Kaner, Cem, Pitfalls and strategies in automated testing. s.l. : IEEE, 1997.
3. Karlsson, Even-André, Andersson, Lars-Göran and Leion, Per, Daily build and feature development in large distributed projects. s.l. : ACM, 1-58113-206-9/00/06, 2000.
4. Koomen, Tim and Pol, Martin, Test process improvement: A step-by-step guide to structured testing. s.l. : Addison Wesley, 1999.
5. Koroorian, Saam and Kajko-Mattsson, Mira, A tale of two daily build projects.s.l. :IEEE, 978-0-7695-3372-8, 2008.
6. Koskela, Lasse, Introduction to code coverage Javaranch, 2004. [Online] http://www.javaranch.com/journal/2004/01/IntroToCodeCoverage.html.
7. Maurya A.K., Maurya V.N. and Ahmad A., Remote system accessing and network security using efficient experimental techniques, Journal of Engineering and Technology Research, Scientia Research Library, Georgia, Vol.1, No.1, 2013.
8. Maurya V.N. and Bathla R.K., Design and Analytical Study of Module Testing, Ph.D. Thesis, Department of Computer Science & Engineering, CMJ University, Shillong, Meghalaya, India, 2012.
9. Maurya V.N., Bathla R.K, Maurya A.K. and Arora D.K., A dynamic innovative scenario of automated regression testing using software testing tool, International Journal of Information Technology and Operations Management, Academic and Scientific Publishing, New York, USA, Vol. 1, No.1, pp. 1-10, 2013. ISSN: 2328-8582
10. Maurya V.N., Bathla R.K., Maurya A.K., Arora D.K. and Gautam R.A., An alternate efficient sorting algorithm applicable for classification of versatile data, International Journal of Mathematical Modeling and Applied Computing, Academic & Scientific Publishing, New York, USA, Vol.1, No. 1, pp. 1-10, 2013.
11. Maurya V.N., Bathla Rajender Kumar, Maurya Avadhesh Kumar & Arora Diwinder Kaur, New fuzzy logic model for effort estimation in software module development, Journal of Engineering and Technology Research, Scientia Research Library, Georgia, Vol. 2, Issue 1, pp. 10-16, 2014, ISSN: 2348-0424, USA CODEN JETRB4
12. Maurya V.N., Maurya A.K., Singh Dibyanshu, Priyadarshi Anurag, Anand Gaurav, and Ateeq Nida, Test maturity model integrated technique for enhancement of software testing process in organizations, Journal of Engineering and Technology Research, Scientia Research Library, Georgia, Vol. 1, Issue 1, pp. 42-56, 2013, ISSN: 2348-0424
13. Persson Christer and Yilmaztürk Nur, Establishment of automated regression testing at ABB: Industrial experience report on 'Avoiding the Pitfalls'. s.l. : IEEE, 2004.
14. Pol Martin, Teunissen Ruud and Veenendaal Erik Van, TMAP, Software testing, 2002. 0-201-74571-2
15. Rajender Kumar, Maurya V.N. and Maurya A.K., A cost- benefit model for evaluating regression testing technique, International Journal of Software Engineering & Computing, Serials Publications, New Delhi, India, Vol.4, No. 2, pp. 84-89, 2012. ISSN: 2229-7413
16. Ramler, Rudolf and Wolfmaier, Klaus, Economic perspective in test automation: Balancing automated and manual testing with opportunity cost. Hagenberg : ACM, 2006.
17. Runeson Per, A survey of unit testing practices.s.l. : IEEE, 2006.
18. Sebesta Robert W., Programming the World Wide Web, Pearson Education, 4th Edition, 2008.
19. Stefan Berner, Roland Weber and Rudolf Keller, Observations and lesson learned from automated testing. s.l. : ACM, 2005.
20. Veenendaal Erik Van, Test maturity model integration (TMMI) Version 2.0. s.l.: TMMi Foundation, 2009.
21. Veenendaal Van Erik , TMMi Release 1.0, TMMi Foundation, 2012.
22. Veenendaal Van Erik, Jan Jaap Cannegieter, Test maturity model integration (TMMi) results of the first TMMi benchmark-where are we today, Eurostar software testing, 2012.
23. Westin Fredrik et al., Lesson learned from attempts to implement daily Build. s.l.:IEEE, 2007. 1534-5351/07.