

# PROFESSIONAL TESTER

SUBSCRIBE  
It's FREE  
for testers

Essential for software testers

| February 2011 | £ 4 / € 5 | v2.0 | number 7 |



## Managing manual testing

Including articles by:

**Bogdan Bereza-Jarociński** VictO **Mohamed Patel** Equiem **George Wilson** Original Software  
**Derk-Jan de Groot** Valori **David Yuill** HP **Erik van Veenendaal** **Ashwin Palaparthi** AppLabs  
**Marek Kucharski** Parasoft

# To maturity, and beyond

by Erik van Veenendaal

TMMi intends to help organizations achieve more effective, more efficient, continually improving testing. The first complete version was launched last month



**Erik van Veenendaal** describes the final two maturity levels which have been added

**The Test Maturity Model** Integration (“TMMi”) is a guideline and reference framework for test process improvement. Such a framework is often called a “model”, that is a generalized description of how an activity, in this case testing, should be done. TMMi can be used to complement Capability Maturity Model Integration (“CMMI”), the Carnegie Mellon Software Engineering Institute’s wider process improvement approach (see <http://sei.cmu.edu/cmmi>), or independently.

Applying TMMi to evaluate and improve an organization’s test process should increase test productivity and therefore product quality. In achieving this it benefits testers by promoting education, sufficient resourcing and tight integration of testing with development.

Like CMMI, TMMi defines maturity levels, process areas, improvement goals and practices. An organization that has not implemented TMMi is assumed to be at maturity level 1. Being at level 2, called “Managed”, requires the practices most testers would consider basic and essential to any test project: decision on approach, production of plans and application of techniques. I call it “the *project-oriented* level”.

The goals and practices required by level 3, “Defined”, invoke a test organization, professional testers (that is people whose main role is testing and who are trained to perform it) earlier and more strategic test planning, non-functional testing and

reviews. These practices are deployed across the organization, not just at the project level. I think of level 3 as the one where testing has become *institutionalized*: that is defined, managed and organized. To achieve that, testers are involved in development projects at or near their commencement.

Version 3.1 of TMMi, launched at EuroSTAR in December 2010, defines its top levels: 4 “Measured” and 5 “Optimization”.

#### **TMMi level 4: Measured**

This is the level where testing becomes *self-aware*. The *Test Measurement* process area requires that the technical, managerial and operational resources achieved to reach level 3 are used to put in place an organization-wide programme capable of measuring the effectiveness and productivity of testing to assess productivity and monitor improvement. Analysis of the measurements taken is used to support (i) taking of decisions based on fact and (ii) prediction of future test performance and cost.

Rather than being simply necessary to detect defects, testing at this level is *evaluation*: everything that is done to check the quality of all work products, throughout the software lifecycle. That quality is understood quantitatively, supporting the achievement of specified quality needs, attributes and metrics. Work products are evaluated against these quantitative criteria and management is informed and driven by that evaluation throughout the lifecycle. All of these practices are covered in the *Product Quality Evaluation* process area.

The *Advanced Peer Reviews* process area is introduced and builds on the review practices from level 3. Peer review

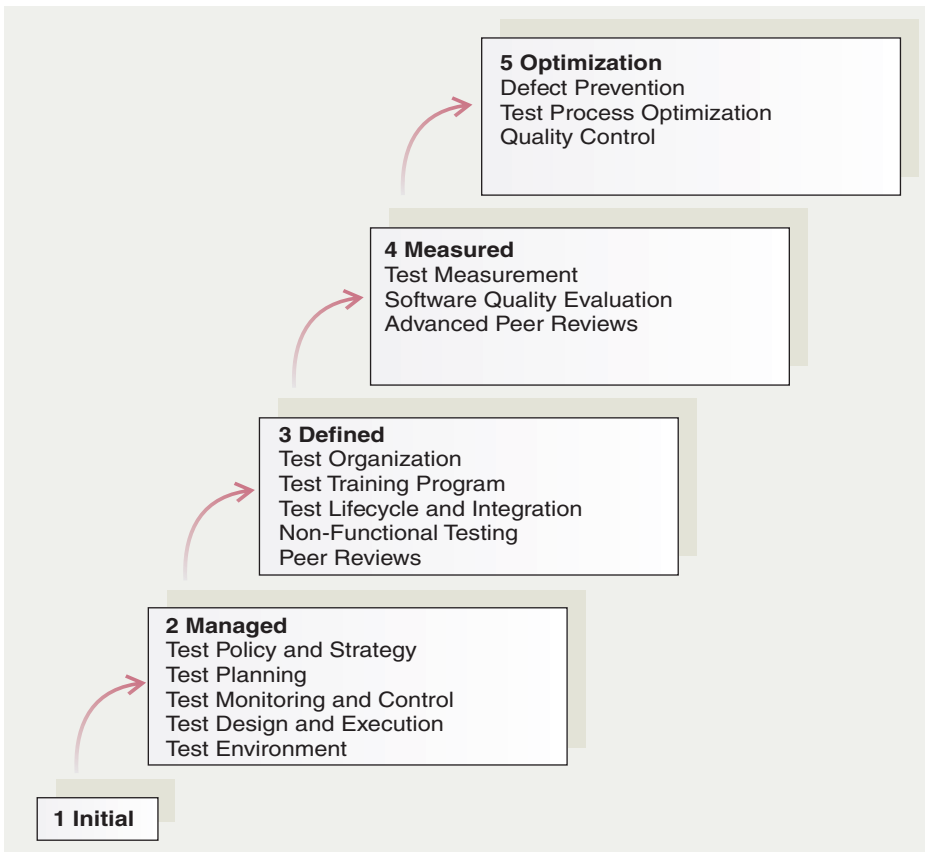


Figure 1: TMMi maturity levels and their process areas

becomes a practice to measure work product quality early in the life cycle. The findings and measurement results are the basis of the strategy, planning and implementation of dynamic testing of subsequent (work) products.

**TMMi Level 5: Optimization**

When the improvement goals at levels 2, 3 and 4 have been achieved, testing is defined completely and measured accurately, enabling its cost and effectiveness to be *controlled*. At level 5 the measurements become statistical and the control detailed enough to be used to fine-tune the process and achieve continuous further improvement: testing becomes self-optimizing.

Improvement is defined as that which helps to achieve the organization's business objectives. The basis for improvement is a quantitative understanding of the causes of variation inherent to the process; incremental and innovative change is applied to address those causes, increasing predictability. An optimizing process is also supported as

much as possible by automation and able to support technology transfer and test process component reuse.

To achieve such a process a permanent group, formed of appropriately skilled and trained people, is formally established. Some

organizations call this the *Test Process Group* or TPG: it relates to and grows from the test organization defined at TMMi level 3, but now takes on responsibility for practices introduced at level 5: establishing and applying a procedure to identify process enhancements, developing and maintaining a library of reusable process assets, and evaluating and selecting new test methods and tools.

Level 5 introduces a new process area, *Defect Prevention*. Defects are analyzed to identify their causes and action taken, comprising change to the test and/or other processes as necessary, to prevent the introduction of similar and related defects in future. By including these practices, at level 5 the objective of testing becomes to prevent defects.

This and the other process areas introduced at level 5, *Test Process Optimization* and *Quality Control*, are interdependent and cyclic: Defect Prevention assists product and process Quality Control, which contributes to Test Process Optimization, which in turn feeds into Defect Prevention and Quality Control. All three process areas are, in turn, supported by the continuing practices within the process areas established at the lower levels ■

---

*Erik van Veenendaal (<http://erikvanveenendaal.nl>) is a widely-recognized expert in software testing, an international testing consultant and trainer and the founder of Improve Quality Services BV (<http://improveqs.nl>). He is the lead author and developer of TMMi and vice chair of the TMMi Foundation. His new book with Jan Jaap Cannegieter, The Little TMMi: Objective-Driven Test Process Improvement, is reviewed on page 22*

---