

PROFESSIONAL TESTER

SUBSCRIBE
It's FREE
for testers

Essential for software testers

February 2012 | £4 / €5 | v2.0 | number 13 |

THIS ISSUE OF
PROFESSIONAL TESTER
IS SPONSORED BY



development testing

Including articles by:

Chris Adlard
Coverity

Les Hatton
Kingston University and
Oakwood Computing
Associates

**Harry M. Sneed and
Manfred Baumgartner**
ANECON

Boguslaw Czwartkowski
Parasoft

Geoff Quentin

CTP7

by Geoff Quentin

Part seven – key points and messages of the Consolidated Testing Process



The conclusion of Professional Tester's exclusive publication of **Geoff Quentin's** formal, rigorous, standards-based method for testing any new software-based system

For everyone's sake – and I do mean everyone – testing must no longer be considered an optional extra. Putting an end to this dreadful mistake is the purpose of the CTP. It aims to achieve it by showing how a mature test process can be consolidated into a mature development process.

Process maturity is a critical issue for me. As a trainer I have spent several decades repeatedly introducing, examining and justifying what I understand about testing then working with many project teams trying to apply those ideas to their varied projects. That has left me in no doubt that to succeed the test process must be as mature as the development process and vice versa. Much has been made of “test process improvement” but that is pointless without a formal development process including integrated test processes. ISO/IEC 12207 is a very mature and scalable development process with close alignment to many other standards. This was my reason for choosing it to be the starting point of the CTP. It does not describe development alone: it goes on to provide processes for support and

maintenance. More work is needed to analyse these and consolidate testing into them too.

Terminology

“Consolidated” refers to a closely-bound combination of a robust development process with testing ideas to create a testing process which is manageable by the use of checkpoints. The V model is only a model, not a process, and is too often cited, explained and even implemented using poorly-defined terms that mean different things to different people. *The CTP is the V model* but uses defined terms from established standards.

Weak terminology has always dogged testing and continues to do so. The list of useless words I hate includes “smoke test” (a joke describing what electronics hobbyists do to their projects) and “white box” (you can't see into it any more than you can a black one). The CTP is an attempt to move away from the ambiguous and towards the defined. The key message is that all project information is open, available and intelligible to all involved. It is especially critical that test results such as coverage and issue reports are clearly defined, measured, graded and balanced against other project data such as risks and benefits. Nothing should be obscured, whether deliberately or inadvertently, for example due to incompetent use of words. Testers must use only formal and defined terminology.

Requirements

Both development and testing, at all levels, absolutely require detailed lists of attributes to be tested, expressed in such way that they can be assigned a binary (yes or no) result. This can be complicated for non-functional or content-related requirements so these must be analysed and expanded until it becomes simple. For example “the system must be very responsive” breaks down into testable attributes such as “when the user has chosen a valid file 10MB in size to be processed, no more than 5 seconds must elapse between the user pressing the return key and the result being displayed”. The failure described by Hans Schaefer in the December 2011 issue of PT (see <http://professionaltester.com/files/PT-issue12.pdf>, page 20), where a help file displayed text in the wrong language, should

have been avoided by specifying and testing something like “after the user has selected German mode, all words displayed must be in the list <http://busintranet/resources/dictionary/german.pdf> until either (i) the user selects a different language mode; (ii) the user logs out; or (iii) the user's session times out”. Vague statements that can give rise to difficult non-functional requirements can often be converted to functional ones: for example “the system must be appropriately secure” may be proposed to mean “all data must be 128-bit AES encrypted before it is stored or transmitted by the system”.

Standards and reviews

Understanding the purpose and proper conduct of acceptance testing eventually leads to the logical conclusion that the most important sentence in a specification of requirements document is “The product must pass all acceptance tests devised by the acceptance test team”. This places great responsibility upon that team: to test all its tests and to ensure that the tests represent what is truly required and acceptable. In order to rise to this challenge testers must:

- know and apply all the standards that relate to software testing
- draw on existing standards as appropriate and enhance the standards where ambiguity exists
- enhance and develop further standards within the framework
- ensure that all testing work is done within an agreed framework and to agreed standards established at project start
- ensure that testing activities and materials are reviewed thoroughly at each project milestone
- have a good understanding of all development processes appropriate within the framework.

The key points of the CTP support this endeavour.

Traceability

The attributes to be tested need to be applicable to the level where they occur. For example the maximum response time for acceptance expressed above may give rise to file system retrieval and screen refresh

The Consolidated Testing Process

speeds at unit test, interface throughput speeds at integration test and performance-under-load profiles at system test. But all must form part of and be easily traceable to the requirement for acceptance and the business objective it supports. A programmer required to demonstrate that a component meets given criteria must be able to check that each criterion exists for a purpose. It is the unique, and much discussed, nature of software that every fragment of source code contributes to the product. All code must have a known purpose and be checked to see it achieves it. Then tests must be devised which can demonstrate that check has been done.

Business and project leaders

The challenge for people whose view is too far removed from detailed code is to avoid specifying subjective, untestable or otherwise useless attributes. Those in senior positions are too often allowed to concentrate on what is easy, for example performance and capacity,

and neglect more difficult areas such as ease of use and security. They should consider more frequently a possible future scenario in which they are required to explain from a witness box how these attributes were handled at all stages of their project. The CTP aims to encourage and provide a basis for visibility that will allow them to do so and to show that all testing done was cost-effective, appropriate and accountable. It can allow everyone involved to be confident at all times that what is being developed will be acceptable.

Testing in all disciplines

We are often told that the number of testers is growing fast. For those of us who believe in

testing that seems at first sight a good thing, and of course it's partly attributable to the evolution and spread of technology. But how many of those testers are needed for the wrong reasons: to shore up badly-run projects or replace proper process with thankless hard work hunting for defects whose introduction should not have been allowed? Following the CTP properly requires that every analyst, architect, designer and programmer must have testing as part of his or her job specification. In the future perhaps people will become professional testers first then move into one of these roles, rather than moving in the opposite direction or being seen as an optional addition to them ■

Geoff Quentin was chairman of the British Computer Society Special Interest Group in Software Testing at its foundation in 1989, is author of The Tester's Handbook and many seminal training courses, won the European Testing Excellence Award in 2006 and founded, with his wife Caroline, this magazine. The previous parts of this series are in the March, May, July and November 2010 and June and October 2011 issues. All are available free in the archive at <http://professionaltester.com>

Did you know that AddQ work with

agile education
business value
consultants test
test automation test management
test management efficiency
quality assurance
requirements analysis
test framework
business development



Telefon 08-501 108 90. Fax 08-501 108 91
info@addq.se www.addq.se

Verifysoft
TECHNOLOGY



Testing Solutions for Productivity and Quality

Conformiq Tool Suite

CONFORMIQ
Automated Test Design™

Benefit from Automated Test Design™ by reducing time and costs for your black-box tests.

Static Code Analysis

Testwell

- ✓ Testwell CMT++ Complexity Measures Tool for C/C++
- ✓ Testwell CMTJava Complexity Measures Tool for Java

Code Coverage Analyser

Testwell

- ✓ Testwell CTC++ for C, C++, Java and C#
- ✓ All compilers, all embedded targets, all coverage levels

www.verifysoft.com

