

Professional TESTER

Number 17
January 2004
ISSN 1742-8742

THE MAGAZINE FOR TESTING PROFESSIONALS

£4

SOFTWARE TESTING WEEK

London
22-26 March 2004

Database
tester

Developer

Automation
specialist

Senior
tester

web
tester

All testers

Advanced
tester

**Substantial,
authoritative,
applicable
training from
the UK's leader**

QBIT
LIMITED

25,000+ testers trained

Having problems solving the software testing puzzle?



Whether it's Weblogic or Websphere, Tibco or MQ, Oracle or DB2, **nmqa** have the skills and experience to solve your testing problems.

For the most flexible, innovative and high quality software testing solutions, think:



For more information on this and other services, please visit www.nmqa.com

11-15 betterton street
covent garden
london
WC2H 9BP

tel: 0207 470 8818

email: info@nmqa.com

Soft skills or just wooly thinking?

Skills, training and certification was first suggested as a theme for Professional Tester over two years ago and has been under consideration ever since, but we seem to have got the timing about right. The re-emergence of strong demand for testing skills and opportunity for those who have them (as demonstrated by the unusually large number of recruitment advertisements in this issue) is the ideal situation in which to discuss the acquisition of those skills. As our contributors have shown, new resources to support learning about testing, and testing itself, are becoming available all the time. The industry, and this magazine, are preparing for an exciting 2004. But there is no room for complacency; testing is still beset by serious difficulties. In fact it is difficult to argue that it has solved *any* of the problems which have plagued it since its inception.

The rate of development of IT, and the acceleration of that rate, has become a cliché but is no less real for that. Testing has failed to keep up. Compare modern computer-based systems with those of 20 years ago: the many ways in which they have improved are obvious, but is *quality* one of them? And if not, who is to blame? Bean-counting project managers who don't take testing seriously? Irresponsible developers who are hostile to constructive input? Semi-literate business analysts who cannot write testable requirements? All of these, of course: but we must accept that testers have failed both to convince these groups to change, and to find effective ways to work around their inadequacies. Why?

In my opinion testers – especially those with a high profile in the industry such as authors, conference speakers, trainers and so on – tend to waste too much time and energy on irrelevant opinions. Because almost all software represents or models something about the world, virtually everything in the world comes into it somewhere; but this does not mean testers or developers must become subject-matter experts for every project on which they work. Similarly, because software is created by people, people's behaviour and interactions affects that creation. But this does not mean that testers should aspire to be experts on these matters; they usually lack both the aptitude and specialized education to do so. It's rather pretentious for testers to hold forth on human psychology, sociology, interface heuristics, coding standards, management (other than practical test management) etc.

Being a professional means concentrating on the job and ignoring personal distractions; and to the professional, the desire to do a good job is sufficient motivation to make the necessary effort. The poor quality of the software in use almost everywhere shows that testing is, taken as a whole, unsuccessful. This situation is unlikely to improve until testers, and especially thinkers on testing, achieve a much clearer understanding of what is their own business and focus their undoubted abilities exclusively on that.

Edward Bishop
Editor

Professional TESTER

Number 17 • January 2004

News

Conferences 4

Developments 5

Skills, training and certification

Skillware: a buyer's guide 6

Bogdan Bereza-Jarociński isn't sarcastic. Oh no, not at all

Getting our own house in order 8

Daryl Enfield is mad as hell and he's not going to take it anymore

Bug hunting is not a career 18

The way forward for testers according to QBIT's philosophy

Examining testing 23

Victoria Reinthal on the past, present and future of ISEB and its testing qualifications

Help yourself 24

Alan Richardson's self-education regime

International relations 26

The internationalization of testing certification as seen by Tilo Linz, Pete Bingham and Andy Redwood

Articles and features

Sound structure cuts XPosure 10

Karen Espley reports on a major migration testing project

Automatic for the people 14

Erik van Veenendaal's analysis of a new test tools survey

The testing project and its infrastructure . 16

The risks of weak infrastructure management. By Felix Redmill

The enemy within 20

Sarah Saltzman explains the dangers of dormant code

Letters 22

The bottom line 30

Nick Blamey on making the ROI case for automated web testing

The next generation 30

Mike Holcombe explains how testing is taught at university

The theme for the next issue, in April 2004, will be **User interface testing, including usability testing**. This is a vital, yet neglected, topic; the business impact of software users experiencing difficulties can be exactly the same as that of critical functional failure. Ideas and opinions in letter form are, as always, invited. If you would like to contribute an article please download our guidelines from www.professionaltester.com and let us know what you have in mind before starting to write.

Professional Tester is published quarterly by Test Publishing Ltd, 73 West Road, Shoeburyness, Essex SS3 9DT. Tel 01702 294491; Fax 01702 299040. Email editor@professionaltester.com. Views expressed by contributors are not necessarily those of the editor or proprietors. ©Test Publishing Ltd 2004. All rights reserved. No part of this publication may be reproduced in any form without written

News: conferences

Our favourite speakers at EuroSTAR 2003 in December. Report by Edward Bishop

The last big conference of the year featured keynotes from many famous and respected testers, as well as dozens of tutorials and presentations spread over the five days. **Erik Boelen** of Gitek was one of several speakers on 'agile' methods, but stood out due to his honest and realistic explanation of both the advantages and disadvantages he perceives. His company's decision to 'go agile' on a project was made when it became obvious that a usable specification of requirements for an e-government web application was not forthcoming. This documentation was replaced



by frequent, informal meetings between small groups of representatives of the customer and the analysis (or "functional") team, and between the analysis, development and test teams. Boelen feels that the agile-extreme-iterative-exploratory (the distinction between these terms seems to be becoming less clear) approach is suitable for projects with short development times, an imposed lack of fully-documented requirements which are likely to undergo a lot of change, where it is possible for business analysis, development and test teams can be located close together physically, and where the functionality of the product is quite limited ("I wouldn't be happy to use it on a really big application, eg a banking system").

Our lead article writer in this issue, **Bogdan Bereza-Jarociński**, made some excellent points about an often misunderstood subject: security testing. He disagrees with IEEE 12207 and IEEE 830 which consider security a non-functional requirement; according to Bereza-Jarociński for most if not all applications security is a function. He emphasised the risks to security caused by bad usability as well as the better-known ones related to heavy load of web applications, and the importance of working from testable, verified security requirements rather than the too-common and useless approach of 'trying to hack into it' late in development.

The 'bill of rights for testers' proposed by **Tom Gilb** reiterated the things testers have been complaining about for years - lack of requirements, not enough time, late involvement etc - but in these days of pragmatism and compromise it was good to hear an authoritative figure reminding us of how things *should* be. His

thoughts on requirements were particularly interesting: he described an experiment where three experienced test managers individually reviewed one page of a requirements document looking for ambiguity and clarity defects classified as *major*, *minor*, or *design* (ie design specs mixed up in the requirements, which Gilb despises). The number of defects found by the managers differed widely and between them they still missed 120 major



defects on the page. By extrapolating this to an entire document, dismayingly figures emerged for the time taken to achieve 'clean' requirements and the possible loss to the project if poor requirements are allowed forward to the test design phase. Gilb proposed an agreed standard of less than one defect per page, and that testers should have the right to simply reject anything with more than this as substandard.

Compendium Developments' **Alan Richardson** of believes one can become a better tester, and "bring the joy back" into testing, by taking part in voluntary beta testing. This enables the tester to work on products which appeal to their personal interest, and whose developers will be grateful for rather than hostile to input. Almost all expert developers write code for their own interest in their spare time; in an inspiring and entertaining talk Richardson elucidated the benefits testers can gain by copying this habit and how to get started.

Rather than just trying to read the CMMI like most of us, **Uwe Hehn** of Method Park Software has been using a computer to search it for phrases relevant to software testing, and applying a mathematical algorithm based on the number of times testing themes and practises are mentioned per page in each process area and at each level. The products of his work are a table which 'rates' each process area for usefulness, and a guide to the highlights for testers. Among the many interesting conclusions Hehn has drawn are that the CMMI contains more of use to



test management than specific testing practices, that it emphasises greatly verification and validation issues, and that there is no straightforward mapping of the model onto existing testing models. The jury is still out on the usefulness of CMMI but anyone wishing to evaluate it for possible use in their projects should refer to Hehn's research first - it will save the new reader of the CMMI a vast amount of work.

On the final day we were delighted to see again PT's very favourite conference speaker **Harry Sneed**, currently of Software Data Services. Sneed specialises in designing accurate and efficient methods of analysing code to produce rigorous and irrefutable proof that the programmers who wrote it are incompetent. Needless to say this does not make him popular and he regaled us with another of his stories of being run out of town by angry developers who refused to work with him on the premises. This was followed by a new version of his lament for testers. "You're the modern equivalent of Roman galley slaves or cotton pickers, forced to do something that should be fully automated and one day will be. In the meantime being



allowed to come to conferences like this helps to maintain the correct slavish attitude." He went on, again as usual, to treat us to a breathtaking display of technical skill. This time

this was an explanation of a method by which test cases are derived from natural language analysis of specifications: in effect, *automated static analysis not of code but of requirements*. This is achieved by a program which parses the text looking for predicate clauses, ie phrases describing logical conditions, thus extracts test cases, and (after a little manual intervention needed because of badly-written requirements) produces WinRunner scripts automatically - each instantly traceable to the clause in the spec that generated it. The details are explained fully, and clearly, in an accompanying paper (he was one of very few speakers to do this rather than simply printing out PowerPoint slides).

At the end of the conference it was announced that next year's will take place concurrently with a new conference, *EuroSP3*, which stands for 'software process, software people, software projects'. Delegates will be able to attend whatever presentations they wish from both conferences. Full details will appear in Professional Tester when available. **PT**

News: developments

New shareware and trialware • Scapa enters load arena • improvements to Intasoft, Parasoft, Compuware, Seapine tools • what's on in 2004 • shameless plug for irrelevant book

And all who train in her Cresta will officially unveil its newly refurbished and equipped Enterprise Testing Education Centre (ETEC) and eCafé training facility in central London at a champagne and canapés launch party on Thursday 26th February. This is an ideal opportunity to view the accommodation and hardware and find out about the training events and activities which will be taking place there. Please email Cresta for an invitation.

info@cresta.net • +44 (0)207 448 4688

Feedback is a gift WinFeedback is a new Windows scripting extension for testing, monitoring and automation purposes, including response timing, up-timing, functional testing, stress testing, health monitoring and task automation. It is shareware and a company licence costs just US\$65.

www.winfeedback.com

Try saying this with your dentures in Scapa Technologies has launched StressTestExpress, a new HTTP load and stress testing tool designed for use with IBM WebSphere Studio.

www.scapatech.com • +44 (0)131 652 3939

Plus ça change... Intasoft's change and configuration management system, AllChange, has been integrated with Araxis Merge, the visual file and folder comparison/merging application. The combination provides a fast baseline comparison facility, allowing users to determine exactly what has changed between software releases.

www.intasoft.net • +44 (0)1392 447780

...plus c'est la meme chose Compuware's File-AID/CS test data management and preparation tool is now at version 3.0. It runs on Windows, UNIX and Linux and now allows the use and transformation (using a new GUI) of multiple data formats including Oracle, DB2 UDB, Microsoft SQL Server, Sybase, XML, Access, Excel, VSAM, IMS, DB2 UDB for z/OS, QSAM and ASCII, and supports stored procedures, triggers and XML.

www.compuware.co.uk • +44 (0)1753 444 444

Track maintenance Seapine Software has released version 6.1 TestTrack Pro, its defect manager, and version 2.1 of Surround SCM 2.1, its configuration manager. The upgraded versions include productivity enhancements, additional security controls, and enhanced integration with each other.

www.seapine.co.uk • +44 (0)1344 297613

Caveat for emptors soon CEA, France's public-sector R&D establishment, has developed Caveat, a new tool for testing safety-critical software, the result of more than ten years of research at the French Commission for Atomic Energy. Caveat is based on proof and static analysis techniques; as well as finding defects, it gives detail on the cases which cause the defects to occur. By then interacting with its user, Caveat is able to provide mathematical proof that an error will not occur under defined conditions. A commercial version is expected soon from TNI-Valiosys.

www.tni-valiosys.com • +33 (0)298 05 27 44

Don't repeat yourself 23rd-29th February is International RSI (repetitive strain injury) Awareness Week.

www.rsi.org.uk • 0800 018 5012

Les MISRA ables Parasoft has extended CodeWizard, its C/C++ static analyzer, and C++Test, its unit testing framework, to automatically enforce the full set of 120 Motor Industry Software Reliability Association (MISRA) coding standards.

www.parasoft.com • +33 (0)164 892600

Oo aar Testing co-ordinator at Mellon European Fund Services Graeme Brooke's first book will be released shortly. The subject is not testing but football (of a sort) and the story is described as "The Ipswich Town fan's version of Fever Pitch". Graeme was inspired to put pen to paper after attending his 1,000th Ipswich game and wrote the book during 2003 at the same time as studying for his ISEB Practitioner Certificate. Profits will go to Cancer Research UK.

"Brooke's side - Adventures of a Tractor Boy" from Heritage House Publishers
www.heritage-house.co.uk

+44 (0)1255 870959

NASty business VeriTest has announced that it will be operating Imation's new storage networking lab. The facility will enable deploying organizations and device manufacturers to perform interoperability testing, performance validation, proof-of-concept, compatibility testing and disaster recovery planning on network data storage solutions. VeriTest will also offer clients independent, vendor-neutral guidance to clients, who are able to conduct testing in the lab or, using remote access capabilities, from their own locations.

www.lionbridge.com • +44 (0)1189 497076

Freeloader Empirix is offering a week's free trial of its FarSight web site monitoring service. All configuration can be done online, monitoring will begin immediately, and users will receive a daily email containing the information produced.

www.empirix.com • +44 (0)1344 668080

Imagone ImagoQA has been acquired by Microgen, the information management software house and consultancy. Operations will be integrated and Imago's High Holborn offices are to close.

www.microgen.co.uk • +44 (0)1753 847100

Events diary 2004

Twelfth Annual Safety-Critical Systems Symposium: 17th-19th February in Birmingham.

csr@newcastle.ac.uk • +44 (0) 191 221 2222

Fourth Annual International Software Testing Conference in India: 18th-21st February in New Delhi, Bangalore and Pune.
www.qaiindia.com/Conferences/sw_tst_2004

International Conference on Practical Software Testing Techniques: 22nd-26th March in Washington DC and 27th September-1st October in Minneapolis
www.psqtconference.com • +1 763 546 0072

TestExpo: 25th March in London. Admission free.
www.testexpo.co.uk • +44 (0)1702 290558

ICSTEST: 21st-23rd April in Düsseldorf.
www.icstest.com • +49 (0)2203 915480

STAREAST: 17th-21st May in Orlando, Florida. **STARWEST:** 15th-19th November in Anaheim, California.
www.sqe.com

newQuality2004: the first testing conference for the new EU member countries. 2nd-4th June in Warsaw.
www.bbj.com.pl/newQuality2004_eng.html

AsiaSTAR: 6th-9th September in Canberra, Australia. **EuroSTAR:** 29th November-3rd December in Cologne.
www.testingconferences.com
+353 (0)91 514470

ICSTEST-UK: 6th-7th October in London.
www.icstest.com • +44 (0)1483 733121

Please send press releases, news etc to press@professionaltester.com



Skillware: a buyer's guide

Testing specialist and trainer Bogdan Bereza-Jarociński has a healthily cynical view of testing training

“Training in testing reduces long term costs. It is not uncommon for less than 1% of an organisation's training budget to be dedicated to testing ... Formal training in test management, test design, static and dynamic techniques, to name but a few of many testing activities, will put systematic and structured method into ... tests. If testing is an average 40% of project budget, why is training investment so low?”

– Andy Redwood (*Golden Rules for Quality Testing*, Professional Tester March 2002)

Framework

“Who saves money on test training knows the price of everything, but the value of nothing”, said Oscar Wilde (well, not really, but he might have if he had lived today).

Andy Redwood's statement above provides a frightening message. Obviously, never in the history of human technical endeavour have so many tested so much for so little! Understaffed, underpaid, under-budgeted and underdogs, testers have nevertheless killed many dragons, traced many monsters to their lairs and often saved the world... or at least the day.

So you are a tester. You have known all along, or discovered only recently, that testing requires specialised skills. You have some of them already, but you want and need more. Somewhere in your mailbox, buried under heaps of spam, there is an advertisement of some test training course. Its title has caught your eye: *Survival Kit for Test Managers on the Verge of Nervous Collapse* (3 days, £1,200).

A good course title is like an inviting entrance. An entrance to what: a shortcut on the way to skills and knowledge or a maze that will only delay you?

Well, both. If you take the right path, turn in the right direction at the right moment, you will surely arrive long before others, who stumble and fall on their arduous journey through the wilderness of hands-on experience. On the other hand, signposts may be misleading and many comfortable paths may tempt you away from your primary goal,

which should be better testing based on more comprehensive testing skill.

Practical experience cannot replace training, but training cannot replace practical experience, either. You need them both to become an expert tester (or test manager, or QA specialist, or whatever you choose to call yourself in order to avoid being tester-labelled).

Five golden rules for finding test training

1. Decide what you need: requirements procurement

Decide what your short-term and long-term career goals are. What skill do you need to achieve them? This applies for individuals; for companies, just remove the word 'career' from the sentence above.

2. Decide how to get what you need: requirements breakdown

To get or enhance some skills, practical experience is the best teacher. For others, self-study may suffice, but beware - it may be easier to retain your motivation during a few days on a training course than during long-time, late-night reading. Finally, there are skills that are best obtained by attending appropriate training courses.

3. What is appropriate: design

Various test-training courses abound. To choose the best, you must first of all differentiate between your short-term and long-term goals.

To satisfy your short-term goals, you would be sorely tempted to find a cookbook course. Immediate satisfaction, obvious return on investment - very promising. Therefore, many training providers offer that kind of training - after all, they are there to supply what customers want!

So, what if you have difficulties implementing bug-tracking system in your current project? Probably, some provider somewhere offers exactly what you think that you need, like *Implementing a Bug Tracking System in Middle Sized IT-projects* (2 days, £750).

Wait a minute - this training course may be what you want at the moment, but it is emphatically not what you really need. To solve this particular immediate problem, you have more suitable options available: Internet search, talking with your colleagues, discussion forums, visiting bug tracking tools vendors' sites.

The purpose of training is to provide long-lasting, generic knowledge, which is flexible enough to be (i) used in many different situations, (ii) adopted to various needs, and (iii) developed on your own when new challenges arise.

For example, narrow “bug tracking knowledge” is rather useless in the long term without related knowledge about change and configuration management. Useful test training should cover a sufficiently broad area.

Otherwise, after attending your *Implementing a Bug Tracking System in Middle Sized IT-projects* course you will soon have to continue with *Performance Requirement Change Management for Web Applications* (2 days, £750). After that, perhaps *Interface Testing for Embedded Systems* (3 days, £1,200), or why not *What Every Test Manager Must Know about State-Transition GUI Testing* (1 day, £350)?

The artificial examples above are exaggerations, but some test-training vendors seem to have already gone too far in this direction! In their efforts to please customers, they offer training with “verifiably” high short-term ROI. My advice to you as customer is: think in terms of long-term benefits, not immediate profit.

This is what training is for: providing you with the tools of your (ie a tester's) trade, not with quick solutions to specific, transient needs!

4. Training forms: implementation

Learning new skills fast is often boring. Skilled teachers can make this experience somewhat less painful with the help of jokes, anecdotes, exercises, practical examples, discussions and demos, but such methods are not free. A training course full of pleasing didactic diversions either takes more time or

contains less material. Besides, it typically presents information in less structured, more chaotic way.

It is up to you - the student - to decide which form suits you better. Beware of flashy "hands-on" training. Sometimes, it gives you great dexterity in your fingers, but much less in your head. On the other hand, a course with an impressive syllabus but delivered with a dull, colourless voice and slides packed with small text may simply fail to deposit its contents safely in your head.

Personally, I am rather cynical about so-called "workshops". They are manna from heaven for overworked test training instructors, but are they really good for students? A few slides with "theoretical background", then a question, then half an hour group-work, five minutes' presentations of groups' results and voilà! - the instructor goes home with great course evaluation forms. Participants are happy, too, having talked with understanding, sympathetic peers for so many hours. The question is, has anyone actually learned anything new?

Test training should not be like an Alcoholics Anonymous meeting, where participants mainly vent their feelings of inadequacy, confess their failings, complain about the unsympathetic world and finally go home feeling relieved and happier!

Teachers can make or break almost any course material. Generally, there are two extremes: different, but equally hopeless.

The first extreme is superb practitioners, grizzled super-consultants who "have seen it all" and can approach any problem with superior confidence in their inexhaustible experience. Their approach to teaching is giving heaps of "tips and tricks", but hardly any systematic knowledge. Trivia like structured lectures, legible notes and appropriate pacing of their talking are typically below their dignity. Commonly found among teachers provided by companies whose main business is consulting, and who find test training may be useful for marketing purposes - but hardly a goal in itself.

The second extreme are car-seller types. They have little practical experience outside their immediate expertise area. They have often received very intensive training in this particular tool or method which they are supposed to teach, so they can deliver their mantras with breakneck speed. However, any deeper questions typically put them off balance, so they pretend not to understand them or wave them off as irrelevant.

Avoid both types and look for teachers with lots of practical experience, appropriate knowledge about teaching, and enthusiasm for their task deeper than pretending to be gurus or

achieving a marketing goal. Ask others, do not be afraid to ask for references, asking who teachers are etc. You are the customer, remember!

5. Testing Test Training (3T)

Feeling slightly empty-headed and elated, like primary school pupils on Friday afternoon, you head away from the training centre back to your ordinary work. Course binder becomes dustware, enthusiasm quickly wanes.

It need not be so! After all, no project is complete without going through the right-hand side of the V Model, so why should test training be any different?

Component testing test training: read through your course binder again. Write the most important points down (at this moment, not much may be left from that wonderful, interactive workshop you enjoyed so much!). Prepare your own summary, make a short presentation for your colleagues. Only then your knowledge becomes really "internalised" and available when needed.

Integration and system testing test training: try to employ in practice some (but not all in one go!) of the new things that you have learned. You will probably encounter unexpected difficulties. Hopefully, the teacher gave you his or her email address, urging you to contact him/her (and hoping that you will not). Use it now! PT

Wanted: Automated Testers with Mercury Toolset Experience

About Cresta

Cresta tests and implements mission-critical systems minimising risk, cost and time. We successfully deliver test solutions to our market leading clients across numerous industries including finance, telecommunications, pharmaceutical and petro-chemical industries. Cresta pride ourselves in the quality and calibre of our consultants whose credentials speak for themselves. We run a continuous education program so that our consultants are kept up-to-date with the latest industry best practice, technologies and toolsets.

The role

We currently have a number of permanent and/or contract opportunities to join our team. As a Cresta Consultant you will be responsible for working at various client sites, executing testing of software systems in accordance with the Test Strategy as determined between the Client and Cresta.

The Candidate

Successful candidates will:

- Have 2+ years Mercury Toolset experience in any of: Winrunner, TestDirector, QuickTestPro or LoadRunner.
- Be personable, enthusiastic and diligent
- Enjoy travel throughout the UK
- Currently have a Mercury CPS/CPC or the desire to study towards one
- Be client facing with an ability to add value to the client and Cresta to each and every assignment

Please send your covering letter and CV to kristy.mcshane@cresta.net with reference PT0023/AT/MP. No agencies.



Getting our own house in order

Daryl Elfield, Managing Consultant at Starbase, believes it's no use complaining that testing is not given the importance it deserves when it's largely conducted by hard-working amateurs who happen to fit a personality profile

Testing has come a long way in the last few years. As a consultant I am spending less time convincing project managers of the need for testing, and a lot more time discussing whether the testing they are doing is sufficient. This is a huge step forward and means that testing is finally being taken more seriously. However, we still have a long way to go and one of the places we can start is cleaning up our own act. It's time to ask: are we actually any good at what we do?

The problem we have as an industry is that for years now we have "got by" with minimal budgets, tight deadlines, and insufficient resources which have all led to a survival mentality – we accept what we have because we doubt it can or will get better. We accept any resources because it's better to have some testers, even if they are developers, than none at all! Unfortunately not only has this compromised the quality of our work, but it has played to the misconception held by most of the IT world – that pretty much anyone can do testing, because it's not that hard!

We have also been able to "get away" with using testers who have little or no experience. When your job consists of either pointing out flaws in other people's work or explaining all the reasons why something isn't ready to be tested or isn't ready to go live, it is rare for the quality microscope to be turned on your own work.

This is never a tenable situation long-term. Eventually project managers will or have asked: "what am I getting for my money?" and "why are you spending so long testing?" It is no longer acceptable to blame everyone else. We need to make sure we have our own house in order.

So let's start at the beginning and ask:

Where do testers come from?

Most project managers and developers when faced with that question would probably say: "the seventh circle of hell" but it's a serious

question with serious consequences. And my answer, to paraphrase The Bard, is as follows: "Some are born testers, some become testers, and some have testing thrust upon them".

Some are born testers

My belief is that one of the many things preventing testing from being taken seriously is the common industry view that testing is a "knack" – you either have it, or you don't. Under this theory, training or experience is irrelevant. Although it's true there are naturally gifted testers out there, they are unfortunately not representative. As professionals, we should not encourage this point of view, since it undermines the skills and experience that, in some cases, we have spent years acquiring.

"We are dragging ourselves down by not insisting on training. It's time to say 'I'm mad as hell and I'm not going to take it anymore'"

Some become testers

How many times have you heard that a tester is just a failed developer? Other than being insulting on the face of it, it implies not only that testing is an inferior skill to development, but also that development skills are easily transferable. Unfortunately "failed developers" are not the only people beating a path into testing. We also find support people looking for a new challenge and business users who fancy their technical skills.

Is this really the best way to become a tester? How do we know whether someone

new to testing is any good? More importantly, how do we ensure they are sufficiently trained to call themselves testers? In reality, most people can become testers, but it shouldn't happen through osmosis. I'll return to this later.

Some have testing thrust upon them

How many of you know of testing teams made up entirely of developers or users? "We don't have a test team, the developers test it" is a common theme as are countless variations on it. Who are these unfortunate people forced to do a job they either don't want to, or don't know how to do? And how good a job do we think they're doing? People forced to do testing will do bad testing. People forced to do testing with no training will not even know that the testing they're doing is bad!

Given this background perhaps the original question ought to be "where should testers come from?"

What makes a good tester? There are countless articles and books on this question, but they typically give an answer by defining the personality of a tester, rather than the actual skills required. Yes, a tester must be patient, have good communication skills, honesty etc. but you can have all these qualities and be a terrible tester. Why? Because at the heart of it, you don't know what you're doing!

What does "writing test scripts" mean? What does "defect logging" actually involve? Are these just things you can pick up or can you be trained? Would it make any difference if you were?

How is that such a crucial IT skill apparently requires no training? How can a company insist on "quality" products, when the very people tasked with ensuring it often have the least training? These days a developer without a computer science degree is increasingly rare, and there are any number of certifications in the various programming languages. But companies will let almost

anyone become a tester, regardless of skills or experience.

We perpetuate this situation by not insisting on training for ourselves and the members of our team. A foundation course in software testing should be a minimum requirement for obtaining a job in testing, not something to be picked up later. Would an IT manager hire a Java developer with no Java experience or training? Would a project manager let the support team "have a go" at writing C++ for his next project? It's laughable, but it's something we accept as testers every day.

Testing is not just an art. It's true that some people have a "knack" for it, just as some people have a gift for programming. But many more people can be trained in the basic skills required to be a reliable tester – requirements analysis, test scripting, test execution and defect logging. If your team do not have these skills, shouldn't this be raised as a major project risk?

So what makes a good tester?

We are all familiar with defending testing as a skill, and much of the need to be officially trained in testing is to convince a sceptical industry that what we know is worth knowing. But beyond that, how do we know we're actually any good at what we do unless we have a standard to measure it against? How do you

distinguish a good tester from a bad tester?

Let's go back to the beginning - where do we learn to be testers? Unfortunately our skills as testers are largely determined by the first job we take - are we surrounded by good practice, or is it a "suck it and see" operation? Hundreds of us became testers during Y2K and the dot.com boom and although we learnt to test efficiently, we never learnt the basics. I have conducted countless interviews where even fundamental questions like "what is regression testing?" or "have you ever performed document validation?" are met with blank stares.

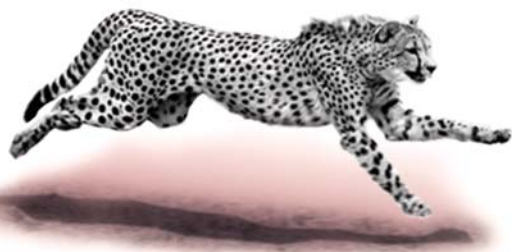
Any actual testing skills are picked up ad-hoc, on the job. How many of us have been reduced to training our teams ourselves because the company we work for won't pick up the bill for training testers? And how good is that training if we ourselves have never been trained?

So here's what it comes down to. On a personal level, we need to get trained to be any good at our jobs. A tester with no training is just performing user acceptance testing - useful, but that's what the customer is for! If you hand me a technical specification I will not, for example, be able to conduct system testing unless I know a) what it is and b) what it consists of.

But beyond this we need to get trained so that the industry as a whole gives us the respect and credit we deserve. It is no use saying: "no-one thinks testing is important" when it is largely conducted by hard-working amateurs backed by no recognisable qualifications.

So approach your test career like you would any other IT career. Start with the basics - get the ISEB Foundation Certificate in Software Testing, even if you've been testing for years. You'll be amazed just how much you have forgotten. If you're specialising in a particular skill (eg load or automated testing), get the relevant training or certification either from the tool vendor or a qualified third party. And don't stop there: there are specialist courses in web testing, in risk analysis, LoadRunner protocols – the list is endless. This is your career, and no one will look after it as well as you will.

We are dragging ourselves down as an industry by not insisting on either training for ourselves or for our teams. Isn't it time we all said, in the immortal words of Howard Beale in the film *Network*, "I'm mad as hell and I'm not going to take it anymore?" PT



Automated Testing. Are you maximising the benefits?

StarBase's comprehensive suite of solutions support Mercury Interactive's ever expanding range of Automated Testing and Application Management tools. Our Consulting services include:

Load/Performance Testing

"The Combination of StarBase testing expertise and CIMA application design knowledge assured a successful project"
Guy Gaskins, Chartered Institute of Management Accountants

Functional Regression Testing

"Within Functional Testing, StarBase have proved themselves to be much more than a service provider, they have evolved with us to be a valuable extension of our IT department".
Matt Cowdrey, AON

StarBase's unique Advanced LoadRunner Training

"Very enjoyable course with useful methodologies. We will be adopting a number of the suggested working practices"
Richard Green, HSBC Bank

To maximise the benefits from your Testing requirements call 0208 905 1120 or visit www.starbase.co.uk



StarBase are a Mercury Interactive top tier business partner



Sound structure cuts XPosure

Karen Espley of Cresta explains how a recent major migration project was carried out to timescale and within budget

Scientists in GlaxoSmithKline's R&D Discovery division rely on robust IT systems and cutting edge applications in their quest to unravel the secrets of health and disease. The IT team, Discovery IT, had until recently to operate in an unusually complex and disparate Windows NT based user environment.

As part of a global project that will be rolled out to all of GSK's 110,000 employees, it became necessary to plan for the migration of Discovery IT's 7,000 users to *GSK Connect*, a unified desktop environment based on Windows XP. GSK Connect will enable Discovery's scientists to install and configure scientific applications directly on their desktops with less support.

With over 700 separate applications supporting the R&D effort, and others in constant development, Discovery IT was under extreme pressure to complete the migration test programme. The challenge was compounded by the lack of a structured, enterprise wide application testing methodology.

The supplier

GSK engaged Cresta to assist with making the migration occur within the project timescale and to an agreed budget by creating a test strategy using a structured methodology which provided processes, tools and techniques to manage and execute the complete testing life-cycle of the project effectively. Such a methodology requires and ensures that a rigorous process is followed and that the process is thoroughly documented, and encourages frequent delivery of results in planned test cycles, thereby giving early feedback to management and development (see figure 1). These factors were key to the success of the GSK Connect project.

In addition, Cresta recommended the use of Mercury Interactive's specialist testing tool *TestDirector* to improve the speed and effectiveness of application deployment. The main reason for this choice was TestDirector's unique global management facilities. The tight integration of its four modules provided

smooth flow of information between the testing phases, and its web interface enabled excellent communication among physically distributed testing teams - a key priority for Discovery IT as they sought to create an efficient global application testing process.

The challenge

Cresta's challenge was to ensure the GSK Connect application testing programme would:

- mitigate risk
- be delivered within the timescale
- lead to a real transfer of knowledge to staff at Discovery
- be delivered within budget.

Discovery needed to ensure that the existing 700 applications would be compatible with the new GSK Connect desktop and that the respective business units would be able to perform their normal scientific activities following the deployment of GSK Connect.

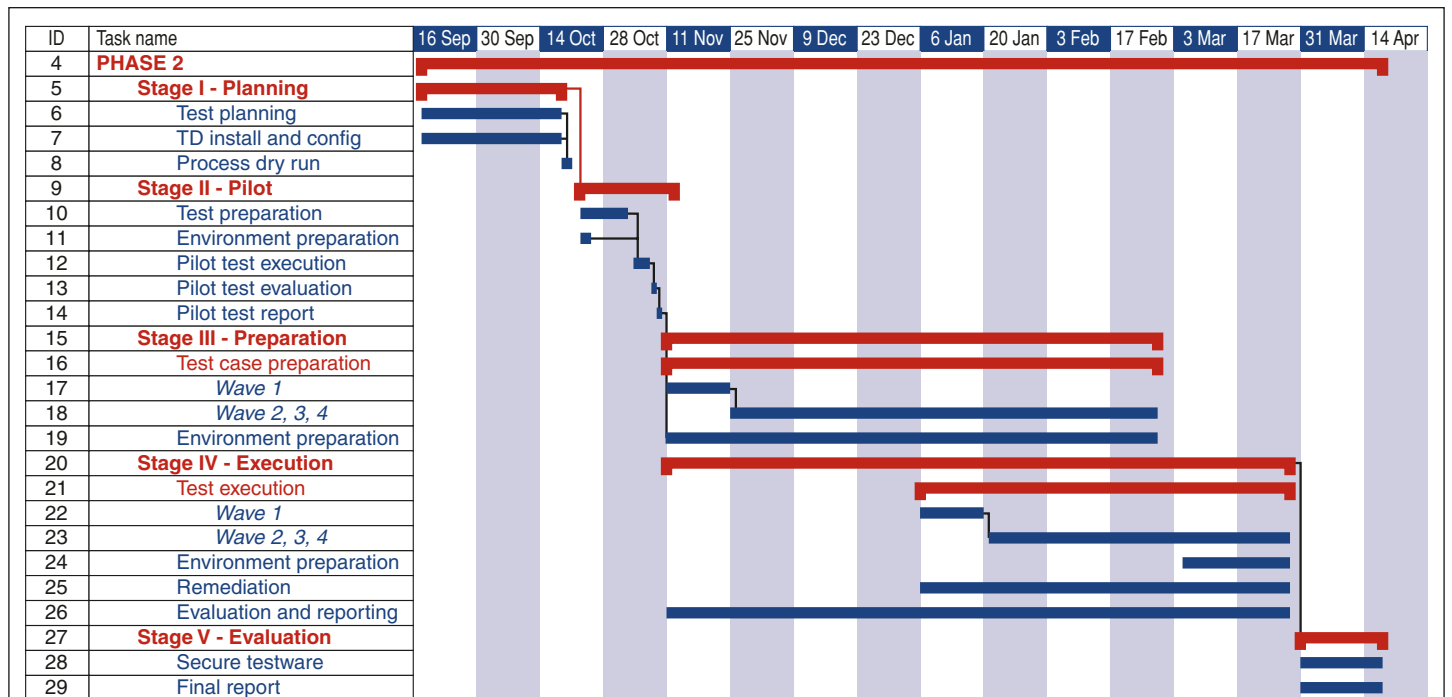


Figure 1. Sample project plan highlighting the iterative testing and frequent delivery of results



The cresta testing training eCafé, where accidents should happen

Inadequate software testing costs business literally billions of pounds a year, and unforeseen accidents in system implementation can undermine your credibility to the point that you might not recover.

Cresta's new Enterprise Testing Education Centre (ETEC) offers a new approach to testing education that allows you to test applications to destruction in a safe environment without making a mess of your business.

Through simulations, courses, training and consultancy, Cresta provides a single source of contact for all testing solutions to allow organisations and individuals to improve their understanding of testing in major IT environments. Based at our eCafé in the City of London and backed by Mercury Interactive, the global market leader in business technology optimisation (BTO), Cresta, through the ETEC, helps you protect your investment in SAP, Siebel, Oracle Apps and PeopleSoft.

The ETEC is open 12/7 and offers both supervised learning and instructor-led training, not only during the working week but also in the evenings and at weekends. All our staff are professionally qualified and field-experienced, so we can help you across the whole range of testing disciplines – from exploring specific scenarios and applications to understanding and implementing best practice.

For testing professionals, we offer a career development path leading to industry-recognised qualifications.

So drop in for coffee. We'd love to show you how to spill milk to maximum effect.

For more details please call:
Mike Stevenson on 020 7448 4688,
email: training@cresta-group.com or
visit: www.cresta.net/testing_training.php



www.cresta-group.com



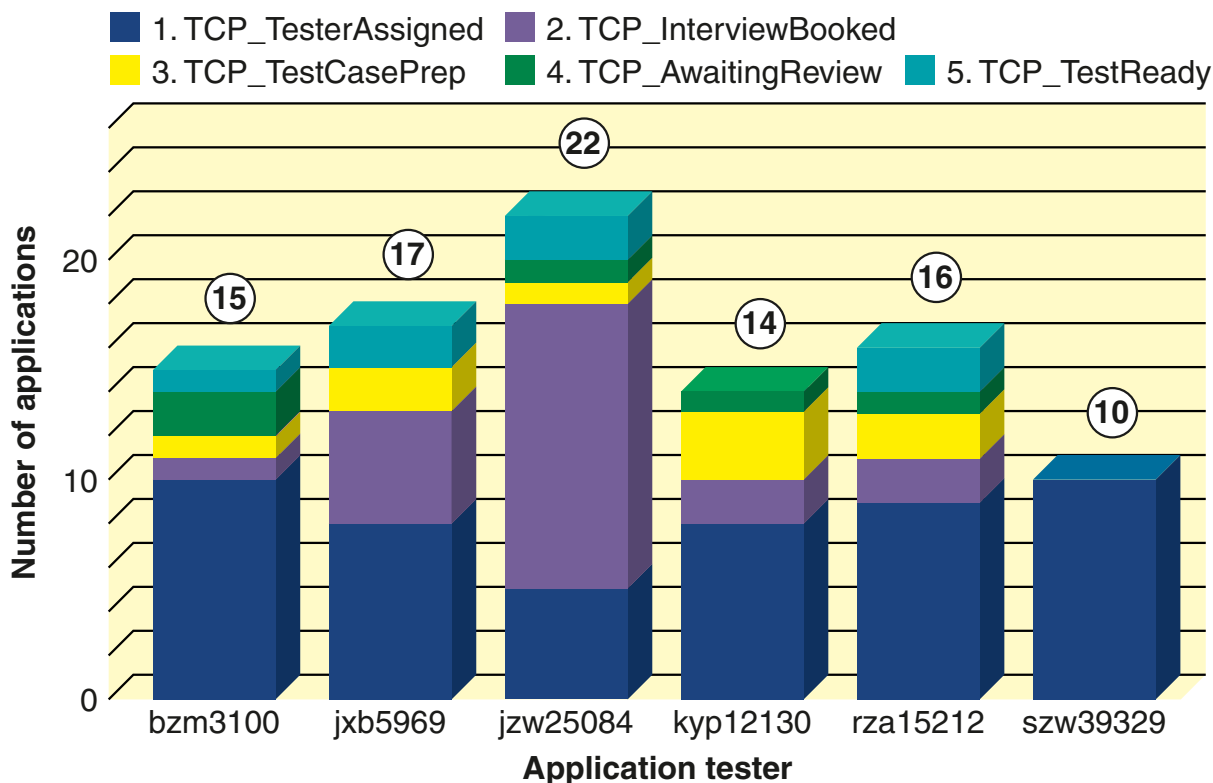


Figure 2. Metrics showing the progress of testing by application and tester

Several risks were inherent in this:

- although Microsoft has stated that the risk of migrating from Windows NT to Windows XP is not as great as from other versions of Windows, there are nevertheless potential violations of operating integrity and security that can occur at the operating system kernel level, which are specific to Windows XP and which could have impacted the functionality and stability of the GSK Connect desktop
- Many of the applications within GSK have dependencies on each other and need to co-exist – output from one application is input to another. The incompatibility of just one application could therefore have rendered a whole suite or range of business processes unworkable
- Support and security access could have changed
- disaster recovery processes and effectiveness may have been impacted by the new desktop build.

The approach

The following activities were undertaken:

- definition of the applications purposes and descriptions of the user base
- identification and testing of all applications' primary functions (any function that is so important that in the estimation of a normal user its inoperability would render the product unfit for purpose)

- performance of focussed instability testing on a set of identified functions
- performance of a set of generic tests checking basic Windows functionality common to most applications (copy/paste, file system, printing etc)
- using of Windows XP AppVerifier tool for testing a wide variety of known compatibility issues.

Each application was defined and documented and then test cases were written and created in a document that described briefly tests that exercised all the primary functions.

The project was delivered in two phases:

- 1 200 applications were centrally resourced and executed by Discovery IT
- 2 500 unsupported scientific applications were delegated to nominated application owners with the business.

Phase two was defined as a means of avoiding the expensive knowledge transfer that would otherwise be required to test the applications centrally. It also ensured the scientists were challenged to show commitment to the future use of the applications they had installed.

The use of metrics was a key component to ensure the testing was on target (see figure 2).

The outcome

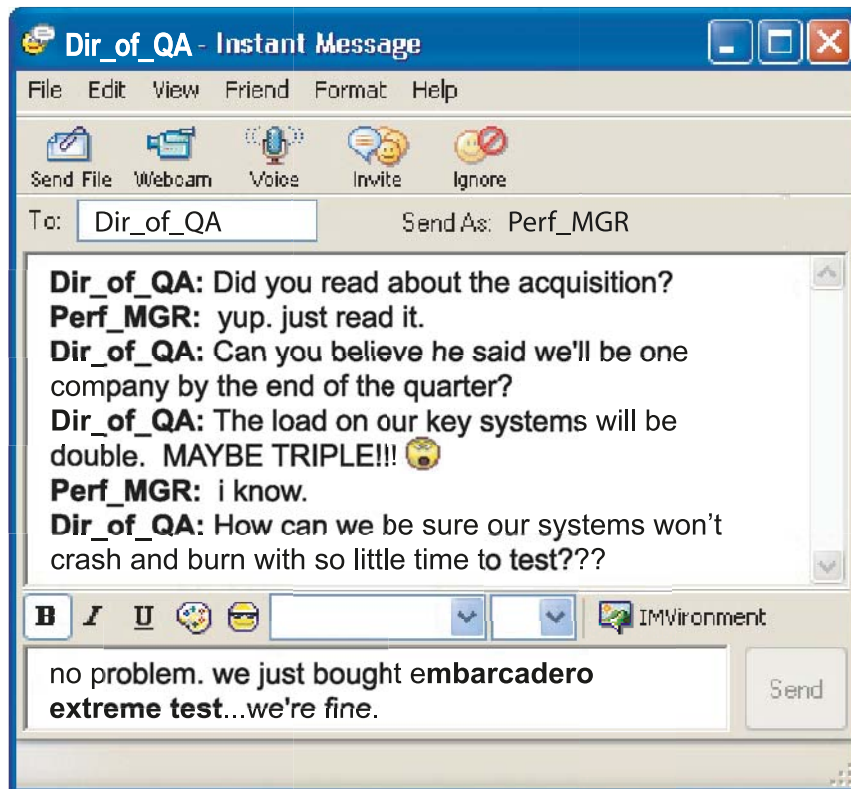
The use of a structured methodology and supporting resources enabled Discovery IT to deliver the project in a timely manner with minimal disruption of normal software develop-

ment and IT support services. However that is not the end of the story; as a result of the project Discovery IT now has:

- consolidation of applications and retirement of applications no longer in use, minimising support requirements and licensing costs
- a customised implementation of TestDirector that supports the business and reduces management burden through locally developed computer-based training and remote training using NetMeeting, customised user guides and quick reference cards, dedicated metrics and management reporting
- comprehensive documentation of the functionality of applications and re-usable test scripts
- a re-usable test process for future application compatibility testing projects
- comprehensive central metrics, management reporting and a complete audit trail
- an up to date application inventory.

These benefits provide a solid basis for the efficient planning and delivery of major test programmes in the future, and provide the platform from which Discovery IT can develop its testing capability, including the use of other tools.

With Embarcadero Extreme Test, the impossible becomes *possible*.



Extreme Test allows QA teams to use a goals-based approach to create realistic user scenarios that produce more accurate results in a fraction of the time. For more information or to evaluate Embarcadero Extreme Test, please visit www.embarcadero.com/proftester.asp or contact **Chris Birks** on **+44 1628 684443** (chris.birks@embarcadero.co.uk)



Automatic for the people

Erik van Veenendaal explains the results of Improve Quality Services' latest survey on testing tools and how they are used

The quality of test tools has certainly matured during the past number of years. Their scope, diversity and level of application have increased. But how far have we come? What tools are used most and what is the level of satisfaction?

In order to find out, we at Improve Quality Services have carried out a survey in over 400 organisations world-wide ranging from IT companies with less than 200 employees to large multinationals. The results can be used to benchmark your current situation, and to identify areas where tools could be used beneficially. To make interpretation more meaningful, a distinction has been made between the market areas of technical applications (eg industry, embedded software and telecommunications) and information systems (eg banking, insurance and government).

This paper should also raise the interest of tool providers since it shows what testers are looking for and what tools still often end up as shelfware.

At a time when time-to-market is more critical than ever before, and applying the latest development methods and tools has shortened the time it takes to develop new systems, it is clearer than ever that testing is on the critical path of software development and that having an effective test process is necessary to ensure that

deadlines are met. In this situation tools are needed to provide the necessary support. After all we're living in an IT society and we're testing software. In recent years tools have grown to maturity and can, if implemented correctly, provide support in raising the efficiency, quality and control of the test process.

Implementation

Test tools may be classified according to the activities they support. The main support currently offered by test tools is intended for test management and test execution. In Table 1 data is shown regarding the implementation of the various tools – the percentage of companies actually using a certain tool type, either off-the-shelf or self-made. No less than 72% of the companies that participated in the survey indicated that they had at least one test tool.

The table shows immediately a general tendency in tool uptake; in the area of technical applications substantially more tools are available and applied than in the area of information systems. This is true for almost every type of tool. There seems to be a large uptake of test management tools. The implementation ratio of defect management tools is still unbelievably low compared to the number of offerings

available in the market. There are many defect management tools available, in all price ranges and for varying levels of maturity. To me it is disconcerting that around 50% of the test organisations are still doing defect manage-

	Control	Quality	Efficiency
Record and playback	*	*	**
Defect management	**	*	*
Coverage	**	**	*
Configuration management	***	*	
Static analysis	**	***	

Table 3: People's favourites

	Technical applications	Information systems	Overall
Test management	45%	30%	35%
Defect management	59%	44%	51%
Configuration management	50%	25%	37%
Configuration management of testware	37%	10%	23%
Static analysis	22%	4%	12%
Test design	18%	13%	15%
Coverage	17%	2%	9%
Performance	35%	15%	25%
Record and playback	29%	30%	29%

Table 1: Test tools implementation ratio

	Control	Quality	Efficiency
Test management and control	5.8	5.1	5.4
Test preparation	5.4	5.7	5.5
Test execution	5.9	6.5	6.6

Table 2: Test tool satisfaction level

approximately the same is record (capture) and playback. Although it is often perceived as the most popular test tool, only one out of three organisations seems to have a record and playback tool. This means that most scripted testing is still done manually and full regression testing is almost impossible.

User Satisfaction

Having a tool is one thing, but how satisfied are we regarding our test tools; do we perceive many or even any benefits? In the tool survey we inquired regarding the level of satisfaction, overall and then per type of tool. Overall 31% responded that they received many benefits, 67% stated that they received some benefits but expected more and only 2% didn't receive any benefits at all. This seems to be a reasonable score, and compared to earlier surveys an increase in perceived satisfaction level can be observed. Participants were also asked to rate their satisfaction level for individual tools on a scale of 1 to 10. Looking at the individual tools (Table 2 shows an overview by tools per phase) one can see that the results correspond to "some benefits but expected more score".

On a scale of 1 to 10 no really high scores can be observed, although the highest level of satisfaction is clearly in the area of test execution tools with benefits in the area of better and more effective testing (quality) and more efficient testing (less effort). As to be expected tools in the area of test management and control have the highest satisfaction score in the area of control. Overall this is only an average score for satisfaction and a challenge for the upcoming years.

People's favourites

Perhaps one of most interesting questions asked by the survey is 'which tools do people indicate as their favourites?'. This of course could be seen as a top five recommendation list. Use the practical experiences from your colleagues, rather than taking advice from a tool vendor. The list of people's favourite tools is corrected for the number of implementation occurrences and also shows the area and level of benefit that testers can receive, ranging from one star (some benefits) to three stars (great many benefits).

Record and playback is still the tool with the highest potential. Table 3 shows that although record and playback is often mentioned as a tool that improves testing efficiency, it is also indicated that these tools provide more control to the test processes and allow for more effective testing. It surprised me that three tools that relate more to developers (coverage, configuration management and static analysis) are on the list. One may also notice that a lot of benefits are in the area of control and quality and that test tools are certainly not only for improving efficiency.

Shelfware

There are many organisations that have successfully chosen and purchased test tools, but many organisations have not achieved any benefit from their investment because their tools have ended up not being used, ie on the shelf or "shelfware". This is not only a problem for test tools, but also a common problem for many types of tools. During the survey the respondents were asked whether they had any shelfware, and if so what tools had become shelfware. No less than 26% of the organisations claimed to have some sort of shelfware. Although this is still a high number, the percentage is substantially lower than reported in earlier surveys by Dorothy Graham; 50% in 1995, 45% in 1997 and 40% in 1998 (Fewster and Graham, 1999).

Figure 1 shows the type of tools that end up as shelfware. Record and playback (42%) is by far the winner. In general one can say the test tools that require a substantial implementation process are mentioned here. It seems not every organisation is yet aware of the fact that there is a lot more to implementation than buying a tool. A thorough selection and implementation process is a critical success factor for beneficial test tool support (Veenendaal, 2002). It is interesting to see that record and playback is both number one on the "people's favourite" list and number one on the shelfware list. Apparently it is either implemented correctly and thoroughly and as a result has great benefits or it does not receive the required amount of resources and as a result becomes shelfware.

More tools?

Finally it was asked whether people would like more tools. This of course is an especially interesting question for tool suppliers, since it indicates what people are looking for. No less than 75% stated that they would like more tools. The three tools that were mentioned most are record and playback, test management and code coverage.

For record and playback and code coverage there seems to be a logical explanation. They both still today have limitations regarding hardware and software platforms and programming language.

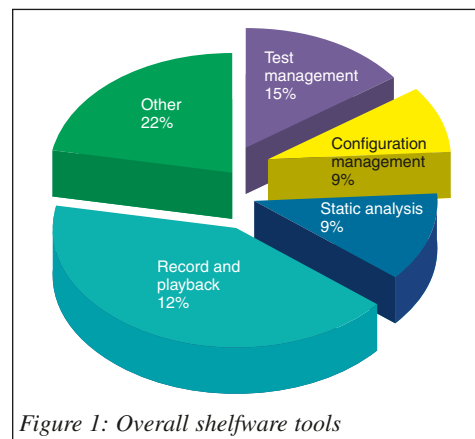


Figure 1: Overall shelfware tools

Many organisations in the area of information systems are looking for coverage tools that support 4GLs and not for tools that support "just" C, C++ and Java. In the area of technical applications the usage of a standard (non-intrusive) record and playback tools is still limited due to various hardware and software constraints. Perhaps an interesting challenge for the tool suppliers!

References

Fewster, M. and D. Graham (1999), *Automating Software Testing: Effective use of test execution tools*, Addison-Wesley, ISBN 0-201-33140-3

Veenendaal, E. van (2002), *The Testing Practitioner*, UTN Publishing, ISBN 90-72194-65-9

Automated Testing Just Got Easier

Seapine
QA Wizard™

REGRESSION TESTING TOOL

- ▶ Powerful capturing capabilities
- ▶ Easy database connectivity
- ▶ Powerful checkpoint engine
- ▶ Browser-based toolbars
- ▶ User Friendly Graphical Interface
- ▶ Multiple filtration systems
- ▶ Organisational tree
- ▶ Highlight Indicators

Rapidly create automated test scripts for Windows and Web applications

CONTEMPORARY
Contemporary plc

Call now on 01344 297 613
Download evaluation at: www.seapine.co.uk



Seapine Software™

The testing project and its infrastructure

Regular columnist **Felix Redmill** on the real reason things go wrong: built-in risks

Software testing, from unit to system testing, and in some cases also including acceptance testing, is usually considered to be a component, or sub-project, of a system development project. But, just as a sub-system is recognised to be a system in its own right when it is the focus of interest, so a testing sub-project is a project in its own right. Certainly it should be perceived as such by the test team and test manager. Yet, within an organisation, the test team is often treated as an in-line group and its manager as an in-line manager.

So what? Why is it important for testing to be perceived as a project in its own right? Because it needs to be managed as a project. It is not a routine function; it is subject to disruptions to its plans at short notice when developers do not deliver software on time; and it therefore requires frequent re-planning and dynamic management of staff, resources and contingency activities. Further, as late delivery of software to the test team may not be accompanied by an extension of the time in which to test it, even the testing objectives may have to change.

In a project, two significant resources are the budget and the time allowed for the delivery of the product - the tested software. But budget and time are also constraints, and in modern projects, in which productivity expectations are often unreasonably high, they are likely to be constricting. At the best of times, small errors can escalate into large problems with substantial undesirable results, but under tightly limiting circumstances the potential for this to occur is amplified. There is little leeway for error.

Adequately trained and experienced technical people usually do a good job. When they make mistakes, they usually discover and correct them, thus arriving at a satisfactory resolution even if some time has been lost. Typically, the more frequent and far-reaching project problems result from management issues and, most frequently, from deficiencies in the 'project infrastructure', which may be defined as possessing three components: the people, communication, and documentation infrastructures, and these form the subject of this article.

In one project, a problem occurred that only the director who had sponsored the project could resolve. But the project manager discovered that the director was abroad and would not be back for a week. When he returned, the director was too busy to hear about his project, and when he took time for it he complained at being expected to involve himself in it. He had authorised the project, he said, and that was the limit of his participation. It hadn't occurred to him that his project's strategic problems were his responsibilities and that his involvement needed to be permanent. The project manager had not drawn up the people infrastructure or formally informed the project participants of their roles, and the director was not the only senior manager who was unaware of his responsibilities to the project. Senior managers had been invited to Project Board meetings, but the project manager had accepted their absence without complaint, so their conviction that they were not needed was reinforced. In the end, with the director's authority appropriately focused, the problem was solved in half a day, but the delay had cost the project a great deal more than that.

The project manager's comment was that such things could happen in any project and were unforeseeable. Such things can indeed occur in any project, but, far from being unforeseeable, what happened was entirely predictable. The problem did not start with the director's unavailability but with the project manager's failure to put an infrastructure in place. The trouble is that many, if not most, project managers do not think in terms of infrastructure.

A testing project is as much in need of a defined infrastructure as any other. Regarding people, there are the test team and the test manager, all of whom should know their responsibilities. If the testers are graded by seniority or salary bands, the implications that these have on project responsibilities should be explicit. Then there are the less obvious project participants. Suppose that some software is delivered too late for its testing to be carried out as planned; who decides on what action to take? Should the test plan be reduced, and if so, how? Should the time be extended? Perhaps the decision will depend on the criticality of the software in question, but

it is not usual for such risk-based planning to have been done. In some circumstances, responsibility for the decisions may rest with the overall project manager (and if so, why did he or she not ensure that the software was developed on time in the first place? But that's another issue). If the test or project objectives need to be changed, definition and authorisation of the changes may rest with an appropriate director, and in some cases a senior customer representative may need to be called in. They therefore should be aware of their long-term responsibilities to the project so as to make themselves available at short notice if a problem arises, and to keep themselves appropriately briefed at all times. If test planners need to be involved, special provision may have to be made for their recall if they have already moved to other projects. Such possibilities should be foreseen by the project manager.

The appropriate involvement of each participant should be established at the Initiation stage of a project, and the basis for doing this is to define the respective responsibilities - and the relationships between participants - within the project. Then all participants should be informed of what is expected of them and their agreement formally obtained. Is this too formal an approach to project management? Not at all; only proper formality at the start will achieve smooth running later. Getting the specification right for the 'people part' of the project is as important as preparing the technical specification. It is not sufficient for the test project manager to assume that the people infrastructure for testing has been put in place by the overall project manager. In most cases it will not have been.

Defining the people infrastructure informs the design of the communication infrastructure. If a director may have to make crucial decisions at short notice, he or she needs appropriate information, not only at the time of the decision but throughout the project. But what form should the information take? Most senior managers demand frequent - weekly or monthly - project reports, but they should also guarantee their attendance at certain types of project meetings. Discussing progress and problems, and making decisions on matters arising, cannot satisfactorily be replaced by the exchange of documents.

Safety-Critical Systems Club

Raising awareness of safety issues and good development and testing practice

regular events • newsletter three times per year • co-ordinator: Felix Redmill

12th Annual Safety-Critical Systems Symposium

with one-day Tutorial on Software Metrics
by Professor Norman Fenton

17-19 February 2004 • Birmingham, UK

To join, or for information on the Club or the Symposium, contact
Mrs Joan Atkinson, tel 0191 221 2222, email csr@newcastle.ac.uk

Reference to a director is merely an example. Each project participant requires certain information in order to discharge formal responsibilities and certain further information for interest and motivation. How should the relevant information be communicated in each case? Which participants need to be in touch with which others? Are meetings necessary or should documents be used - and, if the latter, what form should they take? What meetings should be held periodically throughout the project, and what meetings may need to be called ad hoc? What documents are required for recording, transferring and storing information? Asking and answering such questions at the project's Initiation stage informs the project manager's design of the communication infrastructure so that it facilitates the collection of appropriate information, its timely transfer to those who need it, the making of decisions, and the solving of problems as they arise. At the same time, the communication infrastructure should be designed to preclude the wasteful collection and transfer of unneeded information.

A high proportion of project information is recorded and transferred in documents, and the document infrastructure too needs to be planned and designed. Why have a ten-page report when all that is required is a single-page brief? Most people would agree that this would be absurd. Yet time is persistently squandered on creating and managing unnecessary documentation. If a junior member of staff is called on to prepare a document that will be read by 'the director', he or she is likely to invest (and almost certainly waste) a great deal of time in its preparation. Only if the purpose of the particular type of document is defined as being for briefing only, and its length as being of no more than one page, will the writer waste less time over it.

And so it is with all types of document. For each type of meeting within the project, should minutes be taken, and, if so, should they record all discussion or only agreed actions? It is useful, at the start of a project, to design a template for each type of document to be produced, showing the required headings and giving guidance on the expected nature and volume of content. When, during the project, new types are found to be necessary, they too should be defined. Many test teams are small, and test managers may put progress reports onto a notice board rather than copying them to all team members. Such informality is not out of place, as long as it is the practice defined by the project manager and understood and expected by all team members.

Document definition is not limited to purpose, length and content. Numbering and filing systems are also essential. Such things are often considered so trivial that they indefinitely retain a low priority and are never implemented, and it is not unusual for severe problems, occurring six months or more into a project, to have their origins in their lack. Then, as with the people infrastructure, the project manager and others may shrug and remark that such things could never have been foreseen.

A distribution system, defining whom each type of document should be sent to, and the mode of dispatch, is also a necessity. Without it, mistakes are likely, many resulting in confusion, late decisions, and project delays. In some cases, a recall system is useful as well. Recalling (and perhaps destroying) obsolete versions of some documents, for example standards, can go a long way towards obviating the need to repeat a stage of the project because out-of-date guidance was used.

A document infrastructure is particularly important in a testing project because informa-

tion (much of it in the form of documents) must be transferred not only within the project but also across the project boundary into the larger parent development project. Often test results must be documented in different forms: in overview for senior managers, in detail for developers, and somewhere in between for the project manager. A format and a protocol are also required for sending feedback to developers, particularly if it may include complaints.

All the informational relationships in the project need to be planned or predicted by the test project manager and used in the definition and implementation of the three components of the project infrastructure. Creative imagination can be as important an attribute to a project manager as leadership.

A project arises out of the need to achieve certain objectives - in a testing project, the testing of software. But only the least effective project management is limited to scheduling and directing technical activities. A significant part of a project manager's job is to design and implement an appropriate project infrastructure to *facilitate* their accomplishment, and to monitor its operation for both efficiency and effectiveness. If senior management understood this, they would not have to wait for a late or inferior product before knowing that a project is in trouble; they would make early judgements on the quality of the project infrastructure. Indeed, they would train project managers to think in terms of infrastructure.

A good project infrastructure smoothes the path of testing. Without it, things go wrong - things that those who do not understand the infrastructure's importance believe to be unpredictable. Projects come to grief because of the risks that we build into them, and the worst of these are often caused by the lack of infrastructure management.

A career
in testing

Automation
specialist

Web tester

Advanced tester

Advertising feature

When choosing training courses to develop your career, place your emphasis on the most important factor: **content**.

Training to help
you **do** testing, not
just know about it

QBIT offers the only
available training
and qualifications in
practical manual
testing skills.

Delegates on the
**Putting Theory Into
Practice** series of
courses analyse
information, produce
documents and
present findings and
recommendations
exactly as on a live
project.

Their performance is
recorded and their
work is examined by
a board of assessors
after the course.

Merit and distinction
grades are awarded
for excellence.

Several large testing
organisations use
these courses as an
integral part of their
training programme.

QBIT
LIMITED

www.qbit.co.uk

Bug hunting is not a career

A very damaging idea

"A good test is a test that finds a fault"
(various sources)

Oh how this idea set back the career of the tester! It was highly doubtful in the early 1980s, when it first emerged in books on software testing, and is now completely out of step with reality.

The professional approach

Here are some better definitions:

"A good test is a tested test

A good test is repeatable

A good test represents the real world

A good test influences design

A good test represents the expected behaviour

A good test represents the desired behaviour

A good test is economical to run

A good test is understood by developers

A good test is understood by users

A good test is directly related to the business need"

(Quentin, *The Tester's Handbook*)

In order to achieve all these, testers must do far more than find bugs. Their work should be part of a defined formal process, and their training should develop a wide range of general and specialist skills.

A real career path in software testing is now widely recognised and well defined. The role of the **test analyst** covers analysis of requirements, designs and systems and design and execution of tests. The work of a group of test analysts is planned and scheduled, and results reporting is coordinated, by a **test team leader**. The test strategy is set and continually adapted according to project progress and change by a **test manager**.

The QBIT Generic Test Process

All QBIT's training is based on the well-known V Model. We believe that all projects should aspire to this and therefore a test process which is suitable for use by both system and acceptance testers and which is compatible with the V Model is required. The QBIT Generic Test Process has been created to meet this need; it has traceability at its centre and fits exactly with the modern view of the test analyst. It covers analysis for test conditions, test design, test scheduling and test execution, including regular status reporting at each step of the process.

The V Model approach divides testing neatly into analytical and empirical testing. Analytical testing uses reviewing, prototyping, simulation, modelling, role playing and model office. These are all on the left hand side of the V Model. The expensive, time-scale wrecking empirical testing is significantly reduced by strong, supportive analytical testing. Some organisations have found that empirical testing can be reduced to a single cycle because there are so few defects in the delivered system as a result of the analytical testing.

The test analyst

The work of the test analyst should be fundamental and vital to the development project. The main products of the test analysis - the tests - should start influencing design before the design is created. In fact the tests can even be used as requirements; this approach is certainly more agile than maintaining a specification of requirements document which becomes more and more elaborate as project change occurs.

To create tests which describe the business process in a way that both users and developers can understand, a test analyst must be trained in true analytical and design skills.

Senior tester

Developer

Database
developer

Business
analyst

A knowledge of the basic theory of software testing, as taught in courses such as those leading to the **ISEB Foundation Certificate**, is essential to ensure that a person understands the fundamentals of the job, but more training is needed to make them able to *do* it.

QBIT's certificated course **Putting Theory Into Practice: Test Analyst** develops the essential practical skills of analysis and design needed to derive the test conditions and so construct usable and effective test scripts. This process involves three distinct tasks: (1) to establish whether a document is complete, consistent, accurate, unambiguous and compliant with standards; (2) to establish that all attributes described are required, justified and traceable to business requirements; and (3) to design tests suitable for use later as empirical tests on the delivered system.

The test team leader

The test team leader plans, estimates, schedules, consolidates results and reports the progress of the testing effort. Other essential skills include the use of advanced test techniques, choosing and using tools, liaising with project and organisation management and formal communications, as well as the motivational and other people skills needed to support the work of the test analysts.

Again, QBIT provides a range of theory courses to support the advancement of testers into a team leading role, but also a certificated course, **Putting Theory Into Practice: Test Team Leader**, which

develops and tests the practical planning, estimating, scheduling, monitoring and controlling skills needed by senior testers.

The test manager

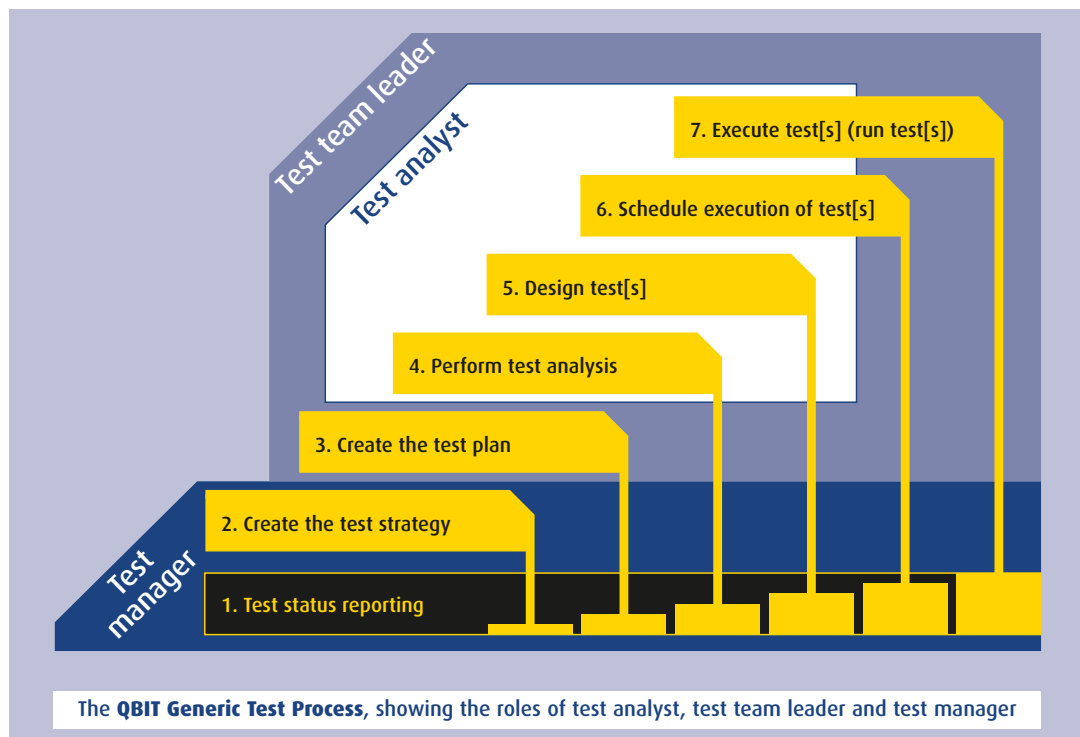
The specialised abilities needed by the manager are related to the product, business model and working culture of their organisation; QBIT offers courses on all aspects of modern quality software development. However the focus of all test managers is on strategy: budgets, politics and process improvement. The QBIT Generic Test Process has been designed to include all of these, and the certificated course **Putting Theory Into Practice: Test Manager** is a masterclass in strategies and reacting to change at project level. The senior and very knowledgeable delegates who attend this course are tested with three complex case studies and their work is assisted, observed and assessed by two expert and highly experienced presenters.

The way forward

Hunting for bugs by exploring systems late in development may be fun but it divides testers from developers and business, is expensive, and will never gain much reward or respect. Testers need to dissociate themselves from this image and move to a central project role, supporting development through a series of well-defined and carefully-chosen quality control points.

About QBIT

- QBIT is the test training specialist; we don't offer consultancy or recruitment and are completely independent of tools vendors etc
- QBIT offers a portfolio of 34 different testing courses for testers, developers and business analysts
- QBIT has presented over 120 courses leading to the ISEB Foundation Certificate in Software Testing since October 1999. Over 1,500 candidates have taken the course and examination with QBIT
- QBIT's presenters have trained over 25,000 testers worldwide
- QBIT's current managing director Geoff Quentin was the founder chairman of the BCS Specialist Interest Group in Software Testing and is author of **The Tester's Handbook**



For full specifications of all QBIT courses please visit www.qbit.co.uk



The enemy within

Sarah Salzman, Compuware's Technology Manager for Automated Software Quality, on the dangers of dormant code and how to find it

Over the past few years security has become one of the most talked about topics in the IT sector. Businesses have made huge investments in technology to try and ensure hackers cannot penetrate their networks and cause serious financial damage. However, the enemy can also lurk from within and steal millions without the company even knowing. For example it cost the Jasper State Bank \$2.7million before they realised that two former employees were engaged in fraudulent activities.

Organisations are tackling the issue of fraud by carefully vetting and monitoring employees (including following up on all references) that have access to sensitive financial systems. Employee vetting is a critical part of the recruitment process because typically with development projects there is a high turnover of staff. People tend to move around with high regularity in this industry. The predictable churn of staff within a project, in addition to project managers hiring contractors (to address the peaks and troughs or to utilise staff with a particular skills set required at that point) is a cause for concern from a security perspective. Having a high turnover means that there is a larger potential/opportunity for people to place malicious code into your applications. The problem is that even vetting and monitoring employees does not always protect against former staff that may still be in a position to commit fraud.

Most organisations do not view current or former employees as an immediate threat, because they are very diligent about managing passwords and access rights when people move on. However, many may be unaware that systems or applications can be developed in such a way that features are built in to enable someone to steal millions of pounds from company accounts or destroy applications without leaving a trace. Fraudulent application developers can insert lines of code that remain dormant for several years, then when/if they leave the company the code may become active. This code is often carefully hidden away within the application and therefore will only be found if people are specifically looking for it.

Presently most businesses do not have the controls or processes in place to protect against criminals who have the technical knowledge to insert fraudulent code into IT systems. Most security managers would not be able to find this type of threat because very specialised and technical knowledge is generally required to discover this malicious code. Methods can however be employed within the specialist application testing or quality assurance teams to trap fraudulent code.

By adopting testing practices that work in combination with code coverage analysis tools, the test teams determine:

- 1 the functional completeness of test procedures being used
- 2 test results based upon code coverage measurements and metrics.

In taking your test procedures further by drilling down to assess the lines of code that have been run during a test, you can visually and easily determine which elements of code have and have not been tested. For example, in figure 1 you can

see what percentage of the code has been covered during the test run.

Next it's important to be able to drill down further to examine exactly which components have or haven't been tested. The diagram below illustrates this easily. The red lines of code have not been tested, whereas the green sections have. This kind of detailed information can stand as tangible evidence (at both summary and detailed level) to prove which code has, and more importantly, has not been tested (figure 2).

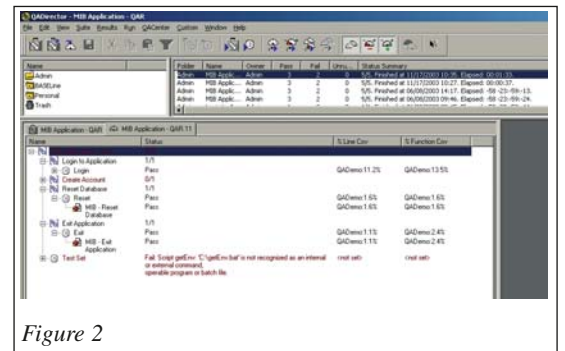


Figure 2

Companies need to turn to the testing teams and look at the testing processes they have in place. It tends to be the norm for people to test only active code (code that actually contributes to the running of an application), but this needs to change if businesses are to protect themselves from rogue developers. They need to make sure that more of the code is tested, so that dormant code is identified, examined and analysed for fraudulent attributes. Embedding code coverage analysis as an element to the testing strategy will ensure that the testing teams can highlight easily lines of code that may not have been tested, prompting a revision in test procedures to generate more testing scenarios aimed at uncovering fraudulent code. Obviously increasing the amount of testing will cost money, but if you consider that the Association of Insurers recently said that fraud costs the UK economy upwards of £15 billion per year, you can see why it is crucial that businesses act to ensure that their application development and testing processes are watertight, guaranteeing fraudsters cannot use dormant code to steal money from the business.

PT

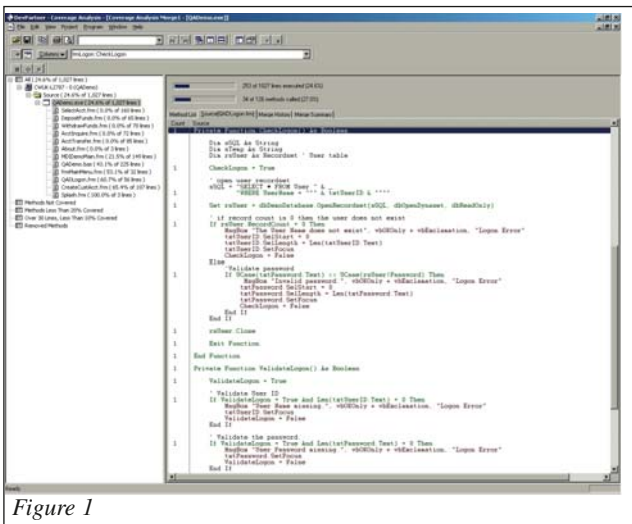
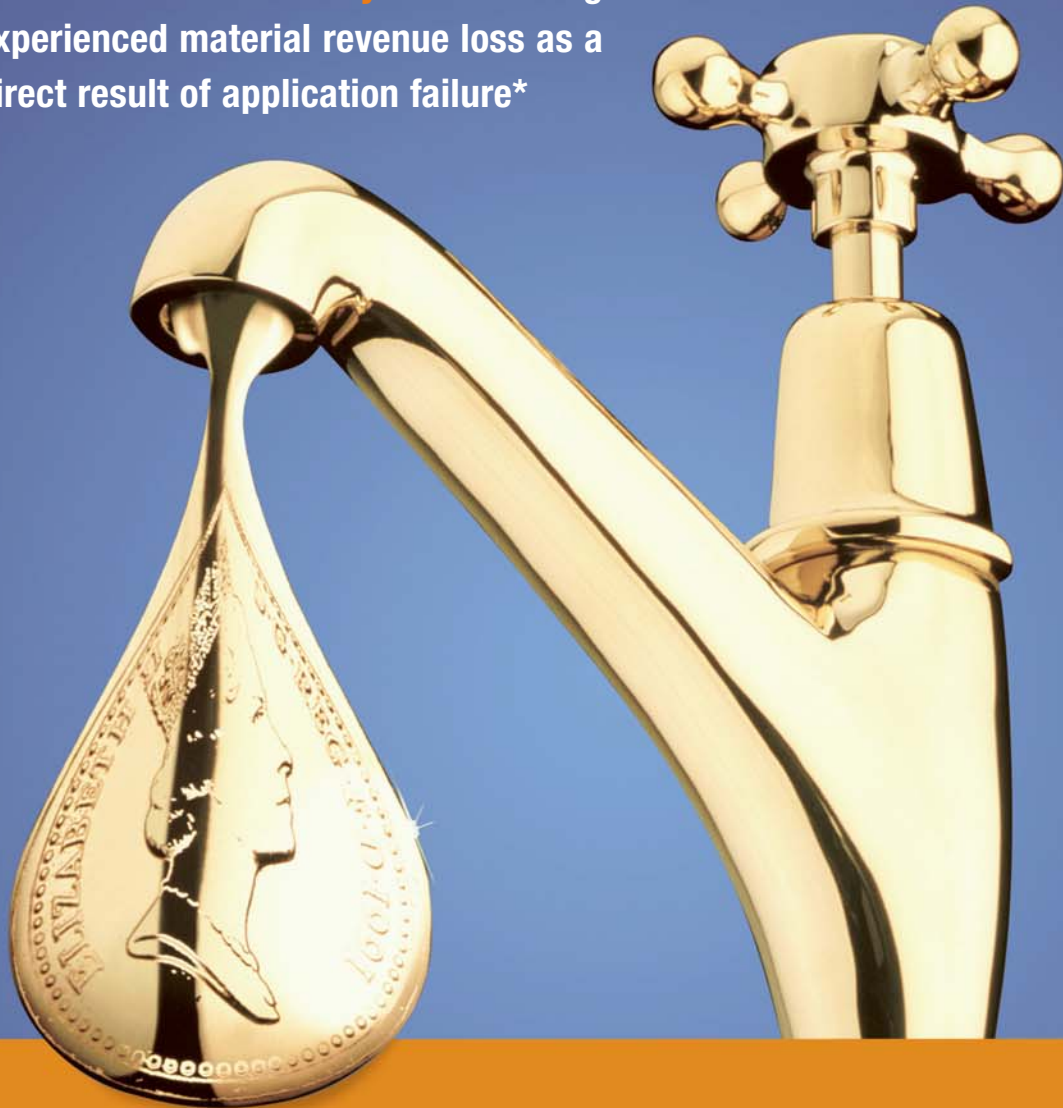


Figure 1

Are your applications leaking money?

64% of IT Executives say Yes... having experienced material revenue loss as a direct result of application failure*



In today's competitive environment, IT departments need to release increasingly rich feature sets across complex distributed infrastructures. To reduce the risk of costly errors, analysts such as Forrester Research and Patricia Seybold, recommend an Automated Software Quality (ASQ) solution.

To learn how your software projects can be a third less expensive** and to download your ASQ information pack with Patricia Seybold white paper, visit:

www.compuware.co.uk/money

* Forrester Research - 2003 ** Patricia Seybold Group - 2003

COMPUWARE
www.compuware.co.uk



BS 7925 and the Yeti

In the last issue we asked if testers found BS 7925 useful. It turns out they can't find it at all

• Readers' views on any testing-related subject are always welcomed. Please write to letters@professionaltester.com

So where is it?

In the editorial of Professional Tester issue 16 you ask for any examples of putting BS 7925 into use.

I'm unable to do this as searching the BSI web site (<http://www.bsi.org.uk>, using the field at the top right of the screen) for 7925 returns only the message "You searched for '7925'. No matches found".

This makes it quite difficult to examine and possibly implement! Please can you point me towards this standard?

—Colin Deady

We would have thought the BSI main site was the most obvious place to look as well. However the standard can be found and purchased at <http://bsonline.techindex.co.uk> but be warned, it's not cheap. Alternatively, the final draft version produced by the SIGiST Standard Working Party, which is almost the same (see below) can be downloaded free from <http://www.testingstandards.co.uk>.

Standards slipping

There are small typographical errors in the British Standards publication which mathematically make very large differences compared with the SIGiST's Standard Working Party final draft. On pages 40 and 41, "2n" should be "2ⁿ" (two to the power n). I have reported this to BSI several times, without satisfactory answer.

—Eric J. Sargeant

A foundation for what?

My primary reasons for taking the ISEB Foundation Certificate course and exam were curiosity, and to be able to demonstrate proficiency in my daily business: quality assurance and control. In addition I wanted to get a competitive edge on the job market. I think I achieved these goals by passing the exam.

I think it is a good thing that a certification programme for testers exists, and that much more should be done much more to get the Certificate known in other countries such as mine (Switzerland).

There is only one aspect which I find curious. If the first level (the Foundation Certificate) is supposed to introduce the principles of software testing why is so much importance given to what's in the standard BS 7925? This standard is focused on component testing, which few testers will do: rather they are involved in integration and/or system testing. Component testing in most companies is done by developers.

Although I recommend this course for everybody interested in software testing, I think it would be more useful if at this level the main focus was on how to do system testing and if the main subject of the course and the exam was how to create test cases based on requirements and use cases.

The only thing that really fell flat during the course I attended was that the exam took place after lunch in the afternoon and after new chapters were discussed in the classroom in the morning. There was no time at all to study the new learning matter. I think it would be better to either do the exam on the fourth day or to use the third day only to review chapters discussed in the previous days.

Also, trying to pass an exam right after lunch is probably not the best timing.

Another improvement could be to provide a reference book such as Andreas Spillner and Tilo Linz's Basiswissen Softwaretest. [As far as we can find out this book is currently available only in German -Ed]

I passed the exam on September 17, 2003 with a well-known UK-based testing training provider whom I am happy to recommend to others. If I had the money I would do the Practitioner Certificate with them as well.

—Torsten Zelger, Switzerland

Little improvement after 20 years

Geoff Quentin *questions whether refining how testing is done is really changing anything*

Test process improvement interests me as the first training course I taught, in 1981, was called Improved Testing Processes. It was intended primarily for developers who needed to learn about boundary value analysis, equivalence partitioning and check lists and reviews as well as the usual statement and branch coverage stuff.

As a programmer trained in structured programming, who was lucky enough to learn about data flow diagrams from the guru and the forerunner of SSADM from its originator, I used to consider the improvement of testing a critical issue. Now I am not so sure.

In the early 80s COBOL was king and personal computers were in their infancy. The development process was well understood and the mainframe environment very disciplined. Obviously things have now changed and I am convinced that it is not test process improvement that is needed but development process improvement. By this I do not mean coding but the whole development process including the business analysis.

If the goal is to deliver working systems that meet the business need and are delivered on time and within budget, many organisations are falling far behind. They are too quick to embrace new technology, scornful of established approaches and intent on inventing a new approach while failing. The National Audit Office analysis of recent failures (see <http://www.nao.gov.uk>) makes depressing reading; it lists as culprits poor change control, poor configuration management and poor plain old fashioned quality control.

Success is totally dependent on management being capable of managing the development process. The biggest complaint from testers attending my courses about their projects is "not enough time". This will not be improved by test process improvement but by management comprehension of the correct way to run a project with the correct allocation of time to the various activities including the essential quality control work.

We have plenty of well defined and documented test processes. We have development life cycles and documentation standards. There are books galore and new testing books that say much the same as was said 20 years ago turn up each month. We have enough review approaches and incident classification systems. We have plenty of laws about satisfactory working environments, working weeks and staff social care. We have training schemes and qualifications for all that might be involved in the development and support of systems. What we lack is management taking it seriously. **PT**

Examining testing

The British Computer Society's [Victoria Reinthal](#) tells us about the thinking behind, current status of, and future plans for the ISEB qualifications for testers

In 1990, the Information Systems Examination Board (ISEB) was formed and became fully owned by the British Computer Society after 23 years of successful partnership between the National Computing Centre (NCC) and ISEB's predecessor, the Systems Analysis Examination Board. The BCS's mission was to make ISEB qualifications add value to professional careers by providing both the means and the platform for recognition and enhanced career development. It is a mission which has seen candidate numbers grow from 3,000 in 1990 to 100,000 today.

ISEB's success has not been in its candidate numbers alone but also in the diversification of examination subjects. Almost 50 qualifications are available covering essential subjects such as Service Management, Data Protection, Information Security and Project Management, which have seen interest grow at a phenomenal rate in recent years. Software testing certification was the brainchild of a particularly proactive BCS Specialist Interest Group in Software Testing; the same group defined the standard that later became the British Standard on Software Component Testing (BS 7925) and that underpinned the certification programme. Initially envisaged as a three-level progressive qualification, the Software Testing Certificate was launched at the Foundation level in 1998. Since then 15,000 certificates have been awarded, making this a spectacularly successful certificate.

The Foundation level certificate, with a syllabus requiring two and a half days of tuition, fits neatly into three days of training at a broad but theoretical and relatively superficial level. This has made it accessible, both for

training providers and trainees, and the number of accredited providers has risen to over 30.

The Practitioner Certificate, by contrast, requires 56 hours of tuition and around half of this is expected to be at a practical level. Most training providers have opted for a 10 day package of training, with a few packaging the delivery into 8 days. The significant increase in training time and cost of the Practitioner Certificate has naturally led to a slower start, and Practitioner candidate numbers have grown fairly steadily, reaching 390 by September 2003. With around 170 candidates expected to take the December 2003 examination, it seems the Practitioner Certificate is settling into a steady growth pattern.

The Practitioner Diploma is currently being worked upon by ISEB's dedicated product development section and it is expected that the first syllabus will be delivered in 2004. *[No details of content or format were available when this issue went to press - Ed]*

One of the biggest challenges ISEB currently faces is coping with demand. At Practitioner level, creating a pool of examination questions has taken enormous effort; so far production of questions has just kept pace with the demands of a quarterly examination and there are no 'unused' questions available for future exams. In the 18 months since the Practitioner exam was first offered the world of software testing has moved on and syllabus updates will be required to keep the qualification in line with current practice. Input and feedback from ISEB candidates is always welcomed.

In fact most of the inspiration for new qualifications originates outside ISEB! An

individual, a group of like-minded people or a BCS Specialist Group, for example, may have an idea and approach ISEB. The proposal will be considered and if accepted will be subject to rigorous scrutiny with a detailed business case produced to prove the market for and viability of the qualification. If this proves satisfactory a BCS Product 'Gateway' Board will approve and support the case and the BCS will fund the development of the qualification.

The individuals involved in the design and development of ISEB qualifications are always specialists and recognised professionals in the specific field and will be drawn from a wide cross section of the IT community to represent commerce, industry, academia, government, training organisations, BCS Specialist Groups and other relevant professional bodies or institutions. These people will form the initial working party who will usually become the core members of the resultant examination board and who set the examinations and manage the qualification once it has been launched.

Another area which ISEB is considering is the provision of qualifications for the continuous professional career development of IT professionals and looking at topic areas which cut (horizontally) through all the existing subjects, for example, soft skills, team motivation and financial management.

Fundamental to all of the work of ISEB, at strategy, operational and development levels, is the essential need to monitor the market, not only to check that the current qualifications still satisfy the original requirement but to detect new trends within the subject area and the changes in technology that will present new opportunities. **PT**



information systems examinations board

ISEB Software Testing Certificates

These qualifications provide visible and objective evidence of an individuals knowledge and ability to perform software testing.

It incorporates the following:

- **The Foundation Certificate*** – is for anyone with an interest in testing. It provides visible evidence that the individual understands the basics of software testing.
- **The New Practitioner Certificate** – is for experienced testing practitioners. This certificate demonstrates a depth of knowledge of testing topics and the ability to perform testing activities in practice.

www.iseb.org.uk/pt

For information contact Customer Support:

TEL +44 (0)1793 417542

Email: isebenq@hq.bcs.org.uk

quoting reference number: 834/0104

THE BCS IS A REGISTERED CHARITY: NUMBER 292786

*Computer Examinations available





Help yourself

Alan Richardson of Compendium Developments works as a tester all day, then goes home and does more testing at night, always looking for new ways of improving his skills

Most of the testers I know have educated themselves on the topic of software testing. All of the best testers I know never stop educating themselves and continuing to learn the skills of their trade through study and practice.

Viewing practice as unnecessary is easy when we are involved in a particular skill day in and day out. But we are often doing the same old thing, and the learning experiences may be fewer than we realise. We may be reinforcing old habits rather than increasing our learning and expanding our knowledge about what we do. The choices we make for testing should be justified, and not be habitual responses to the notion of, or need for, testing.

Although one aspect of the current testing zeitgeist is certification, I recruit by looking at a candidate's experience on their CV and, during an interview, by trying to determine what insights they have learned from that experience.

Work situations may not always be the best place to learn. Learning requires the ability to get it wrong, to experiment and to try new things. Sometimes this is going to require doing and studying testing outside of the work environment. Take the time and give yourself permission to learn.

Test process improvement can start by improving yourself and becoming the best tester you can be. By taking responsibility for the state of your own skills and your own skill development, you can rapidly become a much more effective, knowledgeable and experienced tester.

Self-education methods

Early in my career, I thought I knew a lot about software testing, just from reading. I was reading everything I could get my hands on. Sadly some of my knowledge was superficial as I hadn't learned how to apply it, and I made mistakes along the way. The books and papers I was reading were not enough for me to know how to apply what I was learning. And, I admit, I have created testing processes that were inflexible, overly formal and bureaucratic. I have previously set up a lot of hard work for myself, and my fellow testers, that the situation I was in didn't require.

Now though, I have learned to better use the knowledge that I have and apply it more effectively and contextually to the situation I am in.

The point? Reading and study is incredibly important and useful, but never underestimate the need for putting this knowledge into practice and learning from your own experiences of doing testing.

Reading

There are of course many testing books available. My own favourites include:

Beizer: Software Testing Techniques

Binder: Object Oriented Software Testing

Kaner, Bach, Pettichord: Lessons Learned in Software Testing

Black: Managing the Software Testing Process
[all very well-known and widely available. Ed]

But don't trust me; ask other testers what they recommend, check what people say about the books on www.amazon.com, flip through the books on bookshop shelves and see how readily the contents apply to your workplace and environment. When you buy a book, you are investing in its contents, not just with the money you pay for it, but the time you spend studying it, so evaluate it carefully before you commit.

And as you read the books; make notes on, think about, and disagree with what you read. Just because they've written a book doesn't make them right, does it? Testers are never the most trusting of people; it makes sense to validate the information we get.

Stay up to date and read plenty of relevant magazines. as well as the obvious one you are reading now, see *STQE*, (now called *Better Software*, www.stqemagazine.com).

Don't limit yourself to software testing magazines. I get a lot of benefit from:

Software Development Magazine
(www.sdmagazine.com)

CrossTalk (www.stsc.hill.af.mil)

Harvard Business Review
(www.hbsp.harvard.edu)

The latter covers management, influence and process, all vital information for the effective tester.

And don't forget the web. There are an astonishing number of websites on software testing out there. I'll just list a few to get you started:

www.stickyminds.com: the companion site for *Better Software* magazine and one of the best testing sites on the web;

www.qaforums.com is an extremely useful discussion site;

www.testingeducation.com is a repository of software testing courses which you can use as self-education tools;

www.testingstandards.co.uk is a great place to pick up information on British software testing standards.

You can bring a lot of your own wisdom to this field by reflecting on subjects that you do, or study, and seeing how they relate to software testing. All the best testers I know bring in ideas and techniques from other areas of study, for example philosophy, logic, physics, martial arts, psychology and strategy. There will be parallels and lessons that you can identify that will expand your knowledge, approaches and attitudes to software testing.

Doing

"Half an hour of playing with awareness is worth six hours of playing without"

—Joseph O'Connor, *Not Pulling Strings*, 1989, Metamorphous Press

In my spare time, I download stuff off the Internet and test the **** out of it. This is commonly known as beta testing and is a great way to practice what I have read and studied.

By beta testing I get to test different types of software, on different hardware platforms and operating systems, in ways that I might never be exposed to in my normal working life. I can identify, and learn, new tools that will help me in my daily testing life, and I can

return to my normal testing projects with new experiences and attitudes that will give me more options in the way that I test.

By doing we learn; what works under what circumstances, to identify what doesn't work, and what things we do that can be made more efficient. By practising, we typically do things with more attention, and we learn, not just the results of the act, but from the experience of the performance.

Learning from experience

So how do you learn from experience? Well, you could wait till something similar rolls around and hope that you remember what you did. Or you can take a more direct and structured approach to your learning:

- reflect
- document
- communicate

We have all had experience with testing, and we all know about post implementation reviews. So conduct some for yourself on your own experiences. Think about the testing you have done and reflect on it; what went well? What went badly? What would you do differently now? Is there a generalised principle there? Ask yourself plenty of questions, and document them.

I have a notebook that I carry around everywhere with me, so that I can document any thoughts that relate to my understanding of

testing. I would encourage you to do something similar. Documenting your thoughts helps you phrase them in such a way that you can communicate them to others, but more importantly, it allows you to review and evaluate them yourself more objectively, and you can recall them more readily in the future.

What skills?

Often we think of skills required for testing in terms of test design and test execution, and that can be translated into knowledge of test techniques, tools, programming/scripting, and heuristics.

A more detailed list of skills would add: planning, designing, managing, communicating, understanding quickly, estimating, influencing, and honesty.

All of the above can be studied. All of the above can be practiced, and life will undoubtedly present you with plenty of opportunities for practising them. All of the above will help make you more effective as a tester.

Conclusion

"...the best way of learning is not through the computation of information. Learning is discovering, uncovering what is there in us."

—Bruce Lee, Artist of Life, 2001, Tuttle Publishing

- testing is a skilled profession requiring a variety of skills, all of which can be improved through practice
- testing is a doing thing
- self-education comes from experience and study
- education and learning never stops.

By presenting these self-education methods, I am not ruling out training courses, learning on the job, discussing testing with other testers, going to conferences or certification. On the contrary, use every opportunity and avenue available to learn as much as possible.

But I firmly believe that you will gain the most by taking control of your own education process. Relentlessly pursue a self-education process and voraciously expand the scope of your learning into areas that you do not experience every day. Continually question your current test processes and beliefs about testing. Learn continually and never stop learning.

I've questioned and observed the best testers I know to determine how they got to be as good as they are. Each and every one of them did, and continues to, pursue a programme of self-education. To the best of my ability, I'm doing this. And I know that as you do this, you can surprise yourself with what you can already bring to your learning process and the scope of what your learning process can encompass. PT



Join Us

e-testing[®]

consultancy

Testers

If you are a top Test Manager, Test Consultant or Test Analyst, it's time to put your load testing tools and lifecycle expertise to the test by working with clients across Financial Services, Telecommunications, Banking and Retail, for a dynamic third-party software systems testing consultancy.

Trainee Test Consultants

Build a career where your skills are in demand. Enjoy the benefit of ISEB accredited training in testing and working with proven methods and advanced automated tools.

e-testing Consultancy can provide you with the opportunity to build a stimulating career working on different platforms, technologies and exciting software projects.

So send your up to date CV to jobs@etesting.com or complete the on-line registration form by visiting our web site www.etesting.com.

International relations

President of the International Software Testing Qualifications Board **Tilo Linz**, Head of ISEB's Professional Products Business Unit **Pete Bingham**, and Chair of the ISEB Software Testing Group International Panel **Andy Redwood** answer our questions about the future of their testing qualifications in Europe and worldwide

PT: Please describe ISTQB's mission and history, and its relationship to ASQF.

Tilo Linz: The International Software Testing Qualifications Board was officially founded in Edinburgh in November 2002. The members of the ISTQB are the national testing boards of Austria, Denmark, Finland, Germany, Poland, Sweden, Switzerland, The Netherlands & Belgium, the UK and the USA. Other countries considering joining include Spain, Australia and New Zealand. ISTQB's mission is to co-ordinate the national 'certified tester' initiatives and provide uniformity and comparability of the teaching and exam contents in all countries involved. The national testing boards are still responsible for translation, issuing and maintaining the ISTQB curriculums in their country language and for organizing and executing examinations in their countries. They assess the training offered in their countries according to defined criteria and accredit training providers. ISTQB and its national testing board members thus guarantee a high quality, internationally recognized standard for software testing qualifications.

Andy Redwood: ISTQB is a not-for-profit organisation set up to allow the existing national software testing examination bodies to come together as a single entity and allow every member nation to examine software testing against the same standards. In this way certificates at different levels of expertise will be equally valid in every member country.

What are the relationships between ISTQB, ISEB and ASQF?

TL: ISTQB and its national testing board members are responsible for the syllabuses contents and for defining and maintaining certification procedures and criteria. But the business of accreditation and certification is normally delegated by a board of professional local certification and accreditation bodies. Like ISEB in the UK (www.bcs.org/iseb), ASQF in Germany (www.asqf.de) is an accreditation and certification body offering certified tester exams in German. ASQF also provides administrative services to the German Testing Board and to ISTQB.

How long have ISTQB qualifications for software testers been available, how many people have gained them, and how do you view their acceptance by employers etc?

AR: Currently all certificates are issued by individual national examination centres. This will remain the case until the international syllabuses are complete. This is scheduled for spring 2004. At this time the individual syllabuses will be replaced with a single set and all member countries will set examinations with questions based on the international syllabuses at each examination level. Member countries are allowed slight variations to syllabuses to cater for translation issues. This will create a great opportunity for employers and software testing professionals. Software testers for example will be able to take Foundation level in any member country and have the flexibility to take an Advanced level in any other member country. Training providers can supply their services in any member country. Candidate certificates will be dual-branded, bearing logos of both ISTQB certification and the national examination body.

This opens up the resource market internationally for employers who wish to take advantage of well trained and highly qualified Testing professionals.

It was announced in early December that an agreement to cooperate has been reached between ASQF and ISEB. What is this agreement and what will happen now?

AR: Historically the examinations have been slightly different. This agreement was designed to move to common practices and procedures to ensure that all members within ISTQB can move forward together. ISEB and ASQF have both offered sources of material to the ISTQB working parties to allow international syllabuses to be delivered. Agreements were reached on sales and marketing arrangements.

Pete Bingham: The existing Examination Bodies are ASQF in Germany and ISEB in the UK. They have the largest number of Software certificates issued. In the UK for example, the British Computer Society (which runs ISEB) has issued over 15,000 Foundation Certificates.

Some people in the industry, including certain employers and educators who perceive shortcomings in the ISEB Foundation Certificate syllabus, were hoping that ISTQB would offer an alternative basic qualification with a different syllabus. Will this now happen, or has ISEB's monopoly simply been perpetuated?

AR: The working parties set up to deliver the international Foundation syllabus are representatives for all member countries. We hope to deliver Foundation and Advanced level syllabuses that will allow testing theory and practical work that reflects business practices across many sectors and disciplines to be examined. ISEB are a major player in certification, not just software testing but many other disciplines. I'm not sure they have ever been a monopoly in Europe where testing is concerned.

TL: Yes, there are some shortcomings in the current Foundation and Advanced syllabus. Nevertheless the existing ISEB and ASQF syllabuses are a very solid basis to build upon. So ISTQB will not invent completely new syllabuses but improve and extend these materials. We invite all interested parties to submit their suggestions and change requests to the ISTQB working parties.

PB: The BCS, with its ISEB qualifications, is very keen to ensure that the testing industry adopts the most appropriate practices, and that people who work in the software testing field can demonstrate their competence through certification. ISEB runs examinations across a broad spectrum of qualifications internationally. A good example where a standard has been adopted internationally is in IT service management (ITIL®). ITIL is the acronym for the "IT Infrastructure Library" guidelines developed for the British government. Today, ITIL is the de-facto global standard in the area of service management. I see the global best practices being created for software testing following similar lines, ie the adoption of a single standard embodied in the ISTQB single syllabus.

What is ISTQB's opinion, and/or your personal opinion, of the ISEB Foundation and Practitioner Certificate syllabuses?

AR: Many volunteers worked very hard to get the ISEB syllabuses and the examination process to where it is today. There is no doubt that software testing in the UK as a profession is the better for it. We now have an opportunity to build on what has been done, not only within ISEB but within Europe, the USA, Australia and New Zealand to refine and improve the testing knowledge on which the examinations are set.

fulfil your resolution

For almost 20 years, Newell & Budge has remained at the forefront of information technology, extending innovative services and solutions to the UK and Ireland's leading companies.

Newell & Budge's Testing Solutions Business is a leader in the testing market, with major programmes being driven throughout the UK for high street banks, major insurers, leading Telco's, utilities and government. As a result, we are looking to recruit more of the UK's top calibre testing professionals on both a permanent and contract basis.

You will be ISEB accredited (or ready to complete shortly after joining us). You will have outstanding communication and leadership skills, with a proven ability to work independently and on your own initiative.

With a passion for testing, you will be looking to advance, joining a company renowned for its quality, professionalism and true market independence.

Excited by what you could achieve in 2004? We are.



TESTING
SOLUTIONS
www.nandbtesting.com

UK Software and Business Testing Specialists

- Consultants
- Risk-based Specialists
- Programme Managers
- Service Managers
- Test Analysts
- Mercury Tool Specialists
- Specialist Sales Staff

To apply, email your CV (Ref: TS01) to our recruitment team at jobs@newellandbudge.com.

Alternatively, send your CV and covering letter to: Newell & Budge Recruitment, Queensway House, 1 Queensferry Terrace, Edinburgh. EH4 3ER

Ref: TS01

Edinburgh - London - Birmingham - Glasgow - Belfast - Dublin

The perfect holiday studio flat in the heart of **Nice** (old town)

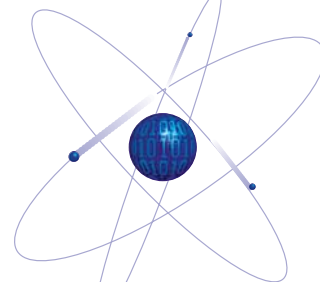


Light and spacious • sunny terrace • double sofabed
fully equipped kitchen • modern shower room
exposed beams • lots of charm • 5 mins to beach
easy access to airport, the famous restaurants,
markets and galleries of Nice, the fabulous
Cote d'Azur and the 'perched villages'
ideal all the year round • car not necessary



£235-£335 pw
depending on season.
For details please call
Jennifer Murray on
01273 888033 or email
jenmurray@20powis.fsnet.co.uk

GUI testing with no scripts. *None.Zero.Zip.*



MITS.GUI & MITS.Comm

Technology totally unique on earth, in the universe, and . . . well actually we're making assumptions beyond our solar system, but you get the idea.

OMSPHERE, for the inside track in an upside down world.

OMSPHERE

San Francisco CA 94013 USA www.omsphere.com
phone (+US) 415 439 5272 fax (+US) 360 397 7221

PB: For some time at the BCS we felt that the ISEB syllabus needed to be more international and we were working on syllabus development that would fix this. However, we feel strongly there is no merit in perpetuating competing syllabuses as this would confuse the market. We are very keen to see a single standard (like ITIL in service management) and the BCS/ISEB supports the ISTQB effort one hundred percent. It really is better that the ISTQB builds on the considerable work and expertise that built ISEB, rather than work from 'ground up', so that the lessons learnt over the last five years are not lost.

Do you envisage that testers in the UK will gain either an ISEB or ISTQB qualification, or both? Should holders of the ISEB Foundation Certificate consider studying for an ISTQB qualification?

AR: All examination centres from ISTQB member countries will offer those who pass the examinations ISTQB certificates. For example those issued by ISEB will also have the ISEB and ISTQB logos. Those issued by ASQF will have the ISTQB and ASQF logos and so on. The standard of the examinations will not be degraded in any way by changing to an international offering, so those in member countries that already have Foundation or Practitioner (Advanced) level certificates will be equivalent to International certificates.

TL: The same holds for training providers. Each training provider accredited by an ISTQB member board or the boards national accreditation body will be recognized by ISTQB. Also in the future this accreditation will be done by the current national bodies. But criteria and procedures will be set by ISTQB.

PB: I think it is very likely that there will be a strong correlation between ISEB and ISTQB syllabuses, so that candidates who already have the ISEB Foundation certificate might automatically be entitled to the ISTQB certificate. It is also possible that there will be a divergence between the two so that existing ISEB certificate holders would need to take an ISTQB exam if they want the international certificate. ISEB will review the situation when the ISTQB syllabus is completed. There will be a transition phase when training providers will need to get their training courses 'ISTQB compliant' and ISEB will run ISTQB examinations. When this happens, UK candidates will receive 'co-branded' ISEB/ISTQB certificates.

Who is responsible for setting the syllabuses for ISTQB qualifications and how will they be examined?

AR: The International syllabuses are being compiled by an ISTQB working party. Within the working party, Thomas Muller (chair of Swiss Testing Board) is responsible for the Foundation level and I'm responsible for the Advanced (what ISEB up to now has called 'Practitioner'). Some members of the working party have responsibility to compile different

sections and others are reviewers. The final draft must be approved by the ISTQB Board.

What documentation is currently available (sample exam papers, syllabuses, guidance for training providers etc)? If not yet, when will it be available? How can readers find out more?

AR: Progress is being made on the testing body of knowledge. This is a database that is being used to ensure that subject matter gets into the right syllabus. This is being worked on now. A structure has been suggested for the Foundation and Advanced level syllabus and a new working party is being formed, headed by Erik Van Veenendaal (Holland & Belgium) to look into the format of the Expert level syllabus. We are just inviting training providers input into the working parties to ensure that appropriate training programs can be accredited. Information is available from www.istqb.org or from Robert.Treffny@asqf.de. ISEB also provides information for training providers at www.bcs.org/iseb, in the 'Training Provider Regulations' section.

Does ISTQB intend to have a mechanism for training providers, employers, learners etc to influence syllabuses and examination methods? Under what circumstances will the syllabus change and how will training providers be expected to react to that change?

AR: All ISTQB member countries are expected to have their own mechanism in place to allow people to contribute to the Body of Knowledge and allow the working parties to alter and release updated syllabuses within and change and release process. In the UK the Testing community will sit within the ISEB Software Testing Group. The ISEB Group has 3 panels – International, Examination and Accreditation. Anyone can be nominated to be a member of a panel by an existing panel member and a panel vote. In this way ISEB maintain an information stream through to ISTQB and back to the panel members.

Changes to the syllabuses will be controlled by the ISTQB board. The change control process is already established. We are not expecting the syllabus to undergo a high frequency of change, so training providers will be given adequate time to adjust training courses. The whole process must of course work for everybody in the loop.

It has been announced that ISTQB will be developing further, higher level qualifications for testers. Please describe these. How will they relate to ISEB's Practitioner Certificate and Diploma?

AR: ISEB has not begun work on the Diploma level and as an ISTQB member country can adopt the Expert level instead. ISTQB has only just instigated the Expert level working party and it would be premature to speculate on this.

ISEB publishes very prescriptive instructions for training providers and issues official accreditation to those whose course follows

those instructions. Will ISTQB use a similar model? If not, how will its relationship with training providers be managed? Will ISTQB offer training for candidates or training providers itself?

AR: As far as I'm aware the model is the same within ISTQB as it was for ISEB. All training must conform to the syllabus. Accreditation will be performed against training provider's material to check it conforms to the syllabus at each examination level. Trainers will be accredited as capable training specialists.

TL: ISTQB will not run any accreditation itself, but will only verify whether an accreditation process (eg ISEB's or ASQF's accreditation procedure) conforms to ISTQB's set of requirements on such a process. If it does, ISTQB will recognize those accreditations.

PB: Although the current accreditation processes are quite prescriptive, they work very well to uphold the high quality standards that are necessary for an examinations institute like ISEB. It has taken many years to refine the processes to where they are today. But, that is not to say they cannot be improved further and the BCS is keen to simplify the current accreditation processes if it benefits the industry without diluting the quality standards. ISEB, under the auspices of the BCS, works hard to maintain fairness and independence, which is why we monitor the examination statistics of all our training providers. I would expect ISTQB to follow the same principles with the other national Boards.

ISTQB's published syllabuses appear similarly prescriptive, even to the extent of instructing trainers how many minutes to spend on each topic area. What is your opinion on the argument put forward by some that this is a combination of textbook and lesson plan, not a syllabus, and inhibits the ability of trainers to improve their courses by using innovative teaching methods, discussing alternative opinions etc?

AR: There are many differing opinions on the pros and cons of the existing ISEB syllabuses. Before the ISEB syllabuses there wasn't anything to bring all subject matter together and now there are documents that describe what should be trained and some guidelines on appropriate time to apportion. That was a significant step forward.

TL, AR & PB: We now have an opportunity to move forward, incorporate new ideas from practitioners, academics, trainers and those in the business community throughout the world, to help formulate a set of syllabus guidelines, flexible enough to meet the needs of a testing industry, which must align to global business practice. This is the challenge ahead.

Interview by Edward Bishop

PT



Tescom Direct is the new specialist recruitment arm of Tescom UK

Are you struggling to find Testing Professionals? Whether it is contract or permanent staff, Tescom Direct offers a tailored recruitment service to help you

The Tescom Direct Advantage:

- 13 years of proven Testing and QA expertise
- Tescom understands your staffing needs better than any general placement agency
- The most rigorous qualification process in the industry
- Face to face interviewing with our HR Director
- Environment specific technical testing

Please contact Tescom Direct via jobsuk@tescom-intl.com or on 020 7022 6708

These are testing times - these are Tescom times

With over 650 test consultants working in 12 countries globally Tescom are the world's leading Independent Software Testing Consultancy. We provide software testing, quality and performance assurance solutions to clients across a range of industry sectors including Financial Services, Telecoms, Retail, Life Science and Gaming and Leisure.

Due to our continued success we are actively recruiting across all levels

- Test Analysts
- Senior Testers
- Test Team Leaders
- Test Managers

If you have over 2 years experience of software testing within mobile telecomms, retail, EPOS or Peoplesoft then we will want to talk to you. The majority of our clients are based within the South East, M25 and Thames Valley. As an alternative we are also looking for experienced Unix Testers to work in Denmark.

Configuration Management

Tescom are recruiting for an experienced Configuration Management Consultant. You must be conversant with architecture, methodologies and processes as well as various tools such as Merant, Rational etc.

Excellent salaries, training and career progression are offered.

DIRECT APPLICANTS ONLY - NO AGENCIES PLEASE

Please email jobsuk@testcom-intl.com or post a copy of your CV to

21-22 Great Sutton Street, London, EC1V 0DN

The bottom line

RadView's senior sales executive Nick Blamey's tips on making the case for automated web testing and measuring the return on investment in it

How does your company measure the value of testing their web applications? Do they only notice when an application fails or when a business partner or end user calls with a complaint? What if tests could provide management and marketing with information to directly affect the company's bottom line? QA and testing teams would be feted as heroes.

Let's face it, most companies measure the value of web application testing in simple terms: usually fewer tech-support calls and emails to the webmaster. In fact, management probably doesn't consider the role that testing plays until an application slows, fails, or exhibits a glitch.

Solving problems during these moments proves your worth, but what if you could provide your boss with return-on-investment (ROI) numbers from testing and provide strategic information to the marketing department? Companies are constantly looking at possible actions and purchases with business value.

Test and QA engineers can participate in strategic business goals by providing ROI data. This article presents ideas for strategically evaluating software test vendors, examples of ROI from testing scenarios, and some of the benefits realised by automated testing.

Evaluating software test vendors for ROI:

Space requirements

Carefully consider the resource footprint of the test software as well as the ability of the software to scale. Will you have to buy additional machines or memory to run the test software? What's the memory usage per virtual client? One company was able to cut hardware and memory needs in half by getting more virtual users out of each PC. This was because the test software vendor they chose had a smaller footprint and additional functionality for easy scalability.

Consider the hidden costs of Testing

How much time will a test engineer need for training? Will the engineer have to spend a week away from the office to learn a new system or a proprietary scripting language? Can they get up to speed and be productive quickly, or will it take a month or so of time,

in the office, jammed in between other projects, which will slow the start of the web application testing?

Once testing is underway, will it require constant rework to continue to run? Look for testing software that offers flexible options and uses standard languages, such as JavaScript, as well as visual point-and-click recording to create and manage test scripts, in order to shorten the learning curve, jump-start the testing process, and keep it moving.

ROI discovered through testing:

Before testing begins, what's your hardware capacity?

If you load test your web application with the idea of optimising your company's hardware configuration, you may find you have more capacity for testing than you thought. One international investment research firm was planning to outsource the testing of a new web site to verify the performance and reliability of the application and to determine the hardware requirements. They decided to perform the testing in-house. Through the load testing process, they determined that its base hardware estimate was sufficient to operate the web site.

Testing for memory leaks

In the same way that a leaking roof drains money from a homeowner, memory leaks on a web application can tie up a team with needless technical tasks. For example, a major financial services firm discovered through load testing that cookie verification in the production web site was causing site instability. Prior to this discovery the testing team was scheduling daily reboots of twenty-two of the site's servers. With the new testing results, the company saved time and energy each day.

Benchmarking information for sales and marketing

Sales and marketing don't usually make the commute to the test department. If they knew what you could offer, you might see them more often. For example, one customer used load-testing software to collect data on competitors' sites' performance. This provided the marketing and development departments with hard data for competitive response times for their company's application. In another

case, a multinational company was working on a major sale of its software, but the buyer would not budge until it had scalability results. Testing of the vendor's application on a simulated environment sold the product. If you work for a software vendor, does marketing have or need scalability information on the company's products?

ROI benefits of moving to automated testing:

Move to automated testing

While moving to automated testing may require a capital outlay to purchase new software, there is an easy-to-compute equation to support this time-and-money-saving method. Just consider the average manual testing labor costs versus automated testing labor costs. For example, if you are testing an e-commerce site, it's easy to compile the figures for money lost if the site goes down for a half day. Compare this possible business loss to the salaries of your testing staff for a six-month period. Now compare the business loss figure again to the salaries of your testing staff over a three-month (automated) testing period. This should yield a return on investment in your staff and any additional software purchases.

Test productivity

Everyone talks about productivity, but how do you calculate the cost savings realised through improved testing efficiency? There are two ways to tackle this issue, one is to compare a prior project's work to a current engagement (in terms of the length of time it took to write the test scripts, etc). The second method is to examine the test cycle. Can the team reuse scripts? If yes, this would limit the amount of new coding work needed and would improve productivity by reducing the amount of time needed to develop tests.

What's needed is an internal evaluation of the approximate amount of time it takes to write scripts versus reuse of existing scripts. The results of good test procedures ensure a company's reputation through a quality product – and make a bottom-line impact by saving and making money. PT



Beware of imitations

Unfortunately, being able to spot when you've been sold a dummy isn't always so easy. Certainly, we have all suffered the experience of buying a copy or imitation item. They rarely last and unfortunately never retain their value like the real thing. At IS Integration, we understand the value you put on your business, we know that quality comes first and we know you can't afford to take risks.

Don't buy a dummy, invest in the real thing.



integration **IS** working together

Test Process Improvement | Test Strategy Implementation | Managed Test Teams | Automated Tools Implementation |
Specialist Test Resource | ISEB Qualified Consultants

www.isintegration.com | info@isintegration.com | 08707 501 502

The next generation

Most commercially-available training in testing is highly vocational, but testing also needs to develop as an academic subject if it is to improve and grow.

Professor Mike Holcombe describes how it is taught at Sheffield University

Software testing is a very neglected subject in the computer science curriculum at British universities. Few teach it as a stand-alone subject to any depth and for most it is a footnote in a more general software engineering or programming course. Thus it is hardly surprising that most computing graduates have a very sketchy understanding of the subject. Their experience is likely to be of informal, ad hoc testing which they have picked up in the course of assignments and project work. The use of any testing tools in universities is most uncommon.

At Sheffield, our philosophy to software testing and quality assurance is based on two key factors:

- a thorough theoretical understanding of the major principles of the subject
- real exposure to practical testing in the course of commercial software development activities.

We teach Computer Science and Software Engineering as both three year honours degrees (BSc and BEng) and four year honours degrees (MComp, MEng). A Computer Science and Artificial Intelligence degree and as well as a number of advanced full time MSc degrees are also available.

The courses are based on what I call a mushroom model. In the first two years all students do the same set of courses which cover the basics of theory and practice in programming, design, testing, project management, artificial intelligence etc. In the third and fourth years, students can specialise in research-oriented subjects which are based mainly around our research strengths: software engineering, machine learning and natural language processing, speech technology, robotics, graphics, computational biology – and software testing. In this way we can ensure that our students reach a very high academic standard.

A lot of what we do is based around group projects which start on day one, year one. This is a unique and very important aspect of our degree courses.

The course runs alongside studies in introductory programming (in Java), mathematics,

artificial intelligence and network architectures. It starts with a module called *Requirements Engineering* and during this students learn about understanding the requirements gathering process, how to write a requirements document and some of the basics of design. They work in their groups on a project called *Crossover* which will involve them in building a complete working system from scratch. Their tutors are the clients and the systems are fairly simple ones such as a theatre booking system, a stock control system, etc. Essentially these are simple databases with suitable user interfaces. By the end of the first semester they should have produced a requirements document – with emphasis on non-functional requirements as well as functional – and a basic architecture and design and some prototype user interfaces. They use tools – for example *Select* is used for the design and *Forte* is used for the user interfaces.

“Few universities teach testing to any depth. It is hardly surprising that computing graduates have a very sketchy understanding of the subject”

In the second semester Unified Modelling Language (UML) is taught by formal lecture. Students are also expected to broaden their knowledge by taking modules from another department: philosophy and chemistry are popular choices. To complete the second part of *Crossover* students must carry out:

- basic system testing
- principles of testing
- psychology of testing
- functional, structural and statistical testing methods
- implementation testing
- test harnesses and unit testing

- integration testing
- creation of use cases and test cases
- formal specifications and test cases
- implementation and maintenance
- acceptance testing and delivery
- error reporting
- version management.

All of these projects follow the waterfall model and are broken up into stages of 2-4 weeks. The name ‘crossover’ refers to the fact that each team receives, at the start of a new stage, the system documentation from another group – usually on a different system. This is intended to reinforce the message that clear documentation is vital and also gives students experience in reading other people’s material.

In the second year there is a more refined version of *Crossover*. This, the *software hut*, involves students working in groups writing software for an external business client. The following subjects are also studied in the second year:

- functional programming
- machines and languages
- systems analysis and design
- database technology
- symbolic reasoning
- adaptive robotics
- abstract data types
- professional issues
- HCI and graphical interfaces
- pattern processing.

The *software hut* places a lot of emphasis on testing because the last thing we want is clients coming back with maintenance issues after the students have finished! Another way that we ensure quality is that each client has six competing development teams and chooses the best solution at the end of the project. A prize is awarded to the winning teams.

We also use the project as a vehicle for our empirical research in software engineering. The students have to archive everything using CVS (see www.cvshome.com) including minutes of meetings, timesheets, system documentation, tests etc. Strict coding standards are imposed.

**The UK's premier
software testing exhibition**

**Thistle Tower, London
Thursday 25 March 2004**



TestExpo™

TestExpo is the **evaluation tool** for everyone interested in the latest tools, products and techniques for use by software testers. It features **free presentations** from the leading consultancies, **free demonstrations** by the leading suppliers and **free discussions** with leading experts. **By visiting TestExpo you can do a whole year's legwork in one day.** There is no better way to find out about and compare the best the market has to offer, and to meet the top companies in testing, all under one roof.

sponsored by  **n focus**
making **IT** work

Visiting TestExpo is FREE.
Order tickets at www.testexpo.co.uk

This archive is an excellent resource for researchers looking into the way projects develop and we try to compare different approaches to development. For example, recently we have got half of the teams to use Extreme Programming (with very strong emphasis on early testing) and the other half a more traditional design-led approach. We hope to be able to make more valid conclusions about the merits of the two approaches than those of most other experiments which do not include real clients and thus miss the point in my view.

In year three the programme is more flexible and students can choose from a wide range of courses. One of these is software measurement and testing. It is based around the following topics:

- the category partition method
- the cause effect graphing method
- implementation based testing
- mutation and statistical testing
- state-based testing
- software measurement theory
- applied software measurement
- measurements and models in software engineering
- collection and presentation of measurement data
- measurement in the management of software engineering.

Students who progress to the four year programme can then take part in the innovative *Genesys* project. This involves students running a company that offers IT consultancy and software development services to outside organisations. The emphasis of the work will be on learning how small IT companies are created and managed, the legal and financial frameworks within which such companies operate, the practical management of the companies and their successful trading. Testing plays a vital role in this work; we use state-of-the-art methods to ensure that our products are of the highest quality. Full details can be found at www.genesys.shef.ac.uk.

At the end of this course the students should:

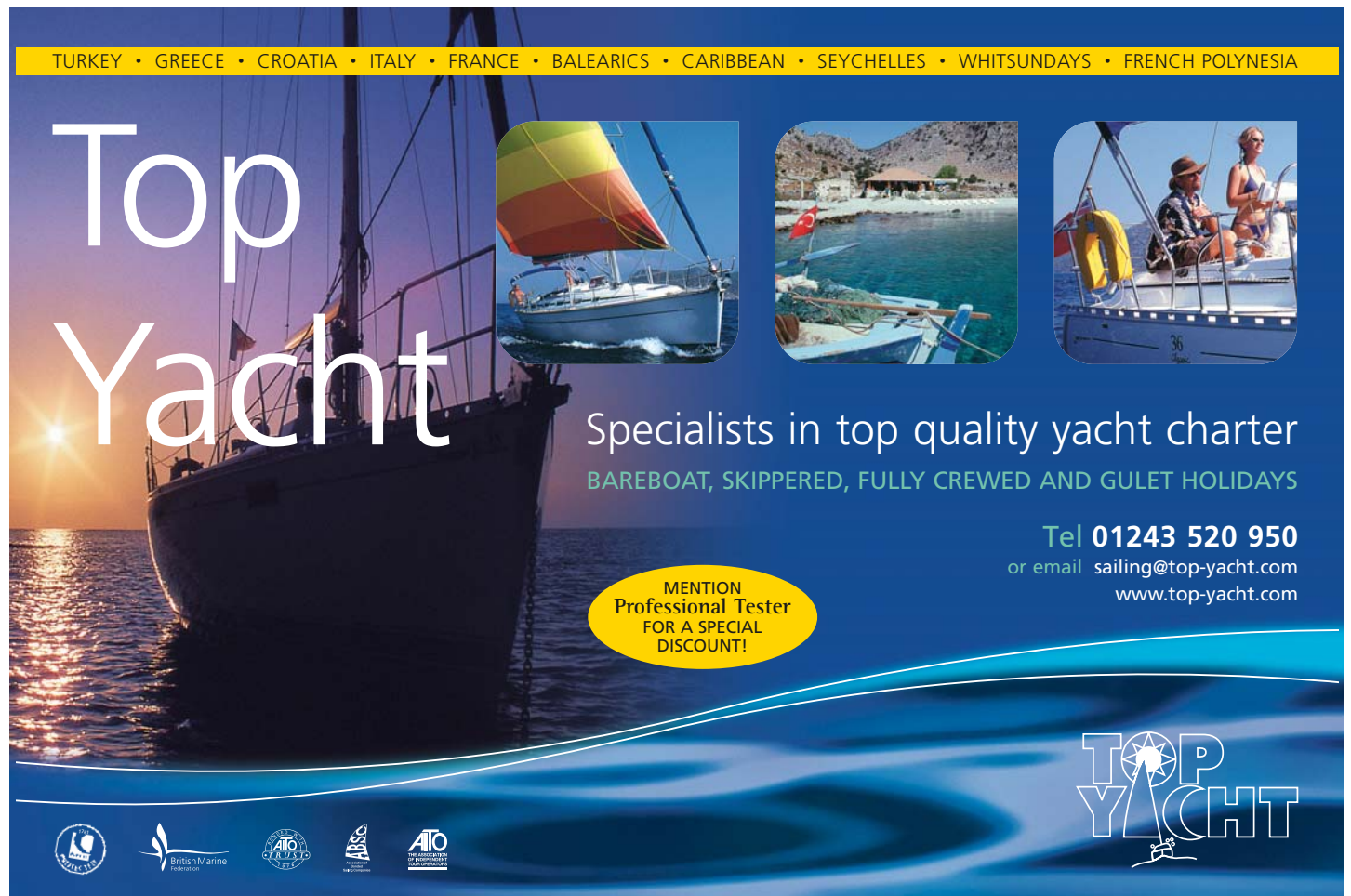
- understand the business issues that underlie the development of software systems
- be able to reconcile business and technical requirements from clients, and plan work accordingly
- be able to evaluate the requirements for software systems, both at the level of individual systems and of the organisations which require to use them
- be able to develop successful software systems for clients, starting from imprecisely stated requirements

- be able to select and apply appropriate quality management measures at every stage during the development of software systems.

The content is based entirely based on the activities necessary to run the software development organisation *Genesys*, and to undertake the projects that it accepts from external clients. The participants in manage the organisation at regular weekly meetings. These involve them reviewing *Genesys*' operations as a business, deciding which possible projects to accept, and also reviewing the projects that are being worked upon. At the technical level the structure is based on the set of current projects, with requirements being set for how these should be managed and how the technical work for them should be carried out.

The company has a number of groups: marketing, systems administration (the company has its own premises and networks running an open source environment), research and development and around seven project groups each with one or more clients or clients. R&D builds and consolidates the company infrastructure; new testing and management tools are built for use by the project groups. We have been awarded a grant by IBM to develop the Eclipse architecture for agile development methodologies, particularly XP. R&D also carry out QA on the individual project teams. PT

TURKEY • GREECE • CROATIA • ITALY • FRANCE • BALEARICS • CARIBBEAN • SEYCHELLES • WHITSUNDAYS • FRENCH POLYNESIA





Top Yacht

Specialists in top quality yacht charter

BAREBOAT, SKIPPED, FULLY CREWED AND GULET HOLIDAYS

Tel **01243 520 950**
or email sailing@top-yacht.com
www.top-yacht.com

MENTION
Professional Tester
FOR A SPECIAL
DISCOUNT!

5th ICS TEST[®]

INTERNATIONAL CONFERENCE
ON SOFTWARE TESTING

April 21 - 23, 2004
Congress Center
Stadthalle Düsseldorf
Germany
www.icstest.com

WEDNESDAY, APRIL 21, 2004

TUTORIALS

Monday April 19, and Tuesday April 20, 2004 are reserved for full-day tutorials on selected issues of software testing.

MONDAY, APRIL 19, 2004

RTCA DO-178 / EUROCAE ED12B
Software Considerations in Airborne Systems and Equipment Certification (2 days)

Cheryl Dorsey
DigitalFlight (USA)

TUESDAY, APRIL 20, 2004

Requirements-Based Testing

Hans Schaefer
Software Test Consulting (N)
Dirk Meyerhoff
SQS (D)

Non-Functional Software Testing in Practice

Bogdan Bereza-Jarocinski
BBJ Test (PL)

Test Strategies and Planning

Eric van Veenendaal
Improve Quality Services (NL)

EXHIBITION

KEYNOTE

The Boeing-Approach to (Independent) Software Verification and Validation
Lawrence Day, Boeing (USA)

TRACK 1.1
Test Management

TRACK 1.2
Quality Assurance for Airborne Systems

TRACK 1.3
Test Preparation

KEYNOTE

BTO knocks IT into shape
Karsten Ludolph, Mercury Interactive (D)

THURSDAY, APRIL 22, 2004

KEYNOTE

Benchmarking of Quality Processes at Bosch
Gerhard Fessler, Robert Bosch (D)

TRACK 1.4
Reviews and Inspections

TRACK 1.2
Quality Assurance for Airborne Systems

TRACK 1.5
Test Processes

KEYNOTE

Quality in a Global Environment: Delivering Business Value
Theresa Lanowitz, Gartner (USA)

TRACK 1.6
Test Automation

TRACK 1.7
White Box and Static Testing

TRACK 1.8
Tester Education and Motivation

Conference Banquet

EXHIBITION

FRIDAY, APRIL 23, 2004

EXHIBITION

TRACK 1.9
Embedded Systems

TRACK 1.6
Test Automation

TRACK 1.10
Test Process Improvement

KEYNOTE

The Role of Simulators in Validation of Real Time Software
Kjeld Hjortnaes, ESA (NL)

TRACK 1.9
Embedded Systems

TRACK 1.6
Test Automation

TRACK 1.11
Testinfrastructure

IN PARALLEL WITH 4TH CSVHC[®] AND SQM[®] 2004 (WWW.CSVHC.COM, WWW.SQM.DE)

SUPPORTING ORGANISATIONS



ORGANISED BY



Leading the way in specialist testing



Consulting

Through early involvement in testing programmes our experienced consultants have a proven track record in delivering significant business benefits to our clients.

Recruitment

A tailored flexible service for all your test recruitment needs. Our expert team take care to understand your exact requirements to ensure we provide the right people at the right time.

Education

We provide ISEB accredited courses, specialist industry workshops and seminars to motivate, retain and develop testing staff, thereby enhancing skills and productivity.

Managed Services

By combining our capabilities we offer a unique solution to the requirement for flexible levels of skilled testing resource. This enables improved planning and cost reduction.



For more information please contact:

T: 01293 44 00 22 E: info@missiontesting.com

W: www.missiontesting.com

Mission Testing is part of The Capita Group Plc.