

OZTester



The Quarterly Magazine for the Australian Software Testing Community and Supporters

ISSUE 3 OCT - JAN 2013

COMPLIMENTARY



In this special Testing Education issue:

My 20 Year Software Testing Education Journey

OZWST - Collaboration in Software Testing

Interview with Sharon Robson, Software Education

Training & Certification: Let's Tidy Things Up a Bit

So What Do the Educators Think of the Education?

A Tale of Two Testers

Tasting Let's Test Review

Our Survey Results...and much more!

OZTester Magazine

Editor: **Colin Cherry**
colinda@ozemail.com.au

Ph 0412 214 240

Editor-In-Chief: **Geoff Horne**
ed@nztester.co.nz
geoffh@isqa.com

Ph. 021 634 900

P O Box 48-018
Blockhouse Bay
Auckland 0600
New Zealand

www.nztester.co.nz/oztester.html

Advertising Enquiries: ed@nztester.co.nz

Disclaimer:

Articles and advertisements contained in OZTester are published in good faith and although provided by people who are experts in their fields, OZTester make no guarantees or representations of any kind concerning the accuracy or suitability of the information contained within or the suitability of products and services advertised for any and all specific applications and uses. All such information is provided "as is" and with specific disclaimer of any warranties of merchantability, fitness for purpose, title and/or non-infringement. The opinions and writings of all authors and contributors to OZTester are merely an expression of the author's own thoughts, knowledge or information that they have gathered for publication. OZTester does not endorse such authors, necessarily agree with opinions and views expressed nor represents that writings are accurate or suitable for any purpose whatsoever. As a reader of this magazine you disclaim and hold OZTester, its employees and agents and Geoff Horne, its owner, editor-in-chief and publisher and Colin Cherry as editor harmless of all content contained in this magazine as to its warranty of merchantability, fitness for purpose, title and/or non-infringement.

No part of this magazine may be reproduced in whole or in part without the express written permission of the publisher .

© Copyright 2013 - OZTester Magazine, all rights reserved.



The Journal For Australian Test Professionals

Welcome to Issue No. 3 of **OZTester** Magazine. This issue is slightly different from the first two in that we are focusing entirely on a single subject - the current state of software tester education in Australia. We have sought out the opinions of several influential people within the Tester education sector including Sharon Robson of **SoftEd**, Toby Thompson of **Discover** and Tony Bailey (CEO of **Access Testing**). We also have contributions from our regular contributors.

In preparing for this issue I also spoke to several recruiters and senior managers responsible for hiring testing professionals and the underlying story was one of dissatisfaction and frustration. My own feelings are that too few companies invest sufficiently in the continued education and development of their staff leaving the majority of the responsibility for this to the individuals themselves. I also think the strategy of many of our larger organisations to outsource and/or off-shore testing is creating a gap in the more junior levels of our profession that will hurt our industry significantly within the next 5 years.

One area that is promising is that of the *context-driven school* (see my article in Issue 2 on **KWST3**)

where significant efforts are being made to improve awareness and education. The **Let's Test Conference** is coming to the Gold Coast next September and this will showcase many of the ideas behind the current thinking of the *context-driven* approach to software testing - more information about this event will be available in the next few issues of **OZTester**.

It is part of our long term strategy to focus entire issues on what matters most to those of us who care about our sector of the IT Industry; however, we can only do this with input from our readers on what they consider significant together with the continued support of our valued contributors.

Therefore, please let us know what you think about the current issue's focus plus what we should address in the future.

So, I hope you enjoy this issue and that it generates debate and discussion both within your workplace and via the various social media sources.

Remember, we have a *LinkedIn* OZTester group and I am always interested in discussing stuff via my *Twitter* account (cherryAKAtism).

- Colin.

IN THIS ISSUE...

Click on title...

- 5 Interview With Sharon Robson, SoftEd**
- 9 OZWST 2013 - Collaboration in Software Testing**
Dean Mackenzie, The System Works Company
- 11 My Twenty Year Software Testing Education Journey**
Colin Cherry, OZTester Magazine
- 14 Training & Certification: Let's Tidy Things Up a Bit**
Sharon Robson, Software Education
- 17 So What Do the Educators Think of the Education?**
Tony Bailey, CEO, Access Testing
Toby Thompson, CEO, Disqover
- 22 NZTester & OZTester Magazine Survey Results**
NZTester Magazine Staff Writer
- 25 A Tale of Two Testers**
Erin Donnell & Jen Hurrell, Codec Software
- 28 Tasting Let's Test Review**
Geoff Horne, Editor, NZTester Magazine
- 30 Twinkle Twinkle Little Star(West)**
NZTester Magazine Staff Writer

REGULAR FEATURES

- 4 OZTester Magazine Announcements**
- 19 OZTester & NZTester Magazine Back Issues**
- 20 Testing Events**

The NZTester Editor finally joins the 21st century:

@GeoffHorne99



OZTester Announcements

Calling all

PROGRAMME TEST MANAGERS

There's now a LinkedIn group for all test professionals operating at Programme Test Management level (or at least aspire to)

Click on the title above.

Over the Tassie, in the latest issue of NZTester, first anniversary issue....

- The Three Pillars of Agile Quality & Testing
- Using Rich Analytics to Reduce Test Cycle Time
- Interview with Chris Wellington, IntegrationQA
- Managing Risk Acceptance
- Testing @ Orion Health
- KWST, STANZ & StarWest Reviews

Click on the cover to download.



This issue's interviewee is:

Sharon Robson

Testing Practice Lead,
Software Education



Our interview this edition is with Sharon Robson, Head of Testing for Software Education, otherwise known as SoftEd. Sharon has enjoyed spells as a Test Team Manager with Hewlett-Packard amongst others.

OZTester: Can you please describe SoftEd?

Software Education (aka SoftEd) is a specialist training company that focuses on training and coaching for all aspects of the Software Development Life Cycle. We offer a complete curriculum for Software Testing, Business Analysis and Agile as well as a number of short courses in areas such as project management, leadership and programming.

Our primary focus is on the training so that our participants have the knowledge and competency to make sure that they can apply their knowledge in the workplace. We are responsive to the needs of our customers to develop training plans and coaching to suit.

OZTester: What services does SoftEd offer?

Mainly we focus on training, where we have a testing curriculum with 15 defined courses to hone testing skills, but we are also available as short term coaches or change assessors (usually for process or skill assessments in our main areas) but we also do coaching, mentoring and transformation assistance in Business Analysis, Design and Development, Agile and Software Testing.

In the testing space specifically we focus on skills building, testing process assessment and assistance in growing your testing capability.

OZTester: What do you believe makes SoftEd different?

Our focus is on skills and direct application of knowledge, for example one of the latest testing course we wrote is totally different to traditional testing training. We have built a lab based course that covers the ISTQB Foundation Syllabus as well as Context Driven approaches, agile and automation. The focus has been that to test software you have to use software. Learning is not done at a purely theory level, it is done at the “learn-by-doing level”. In my opinion the only way to learn is by doing the work, not reading about it and that’s how our courses are focused and delivered.

We have a central group of courses for core capabilities, vital to every tester, but we also have other courses that cater for specialist skills such as SQL, automation, and Risk Based Testing. We enhance this by working with others like Jonathan Kohl to incorporate their expertise in areas such as Mobile Application Testing.

OZTester: What do you think makes a Trainer come to work for SoftEd?

It’s a multi-dimension thing! We must all be practitioners in our field, at SoftEd you cannot train in the topic unless you have real-world experience in the topic. We then look for people who are keen to share, explore and grow. One thing that trainers need at SoftEd is an open mind and a lust for new ideas, knowledge and techniques.

It is a hard journey to go on as a trainer. You are constantly examined, evaluated and tested by the people in our classes, by our peers and by ourselves. You have to have a reasonably resilient ego, but also a deep internal trust that what you are

doing or sharing is a good thing, for both you and your class attendees. If it's not good for you, it becomes very obvious in the classroom.

If you look at a gathering of trainers at SoftEd (which sadly we don't get to have very often) you will see some general characteristics no matter what training discipline we specialise in. These are sharing; exploring; deep thinking and evaluation; consideration of context; respect for others and a great propensity to laugh at each other and ourselves! It is very important that we say grounded and not get too caught up in what we do and how we do it.

OZTester: Where do you believe the challenges for Australian testing companies lay?

Testing companies (companies that do testing either internally or as part of their product lifecycle) are everything and everyone...and they want certainty and quality. The challenge is that both are very context dependent! Testing operates on many levels. Elizabeth Hendrickson calls it Checking and Exploring, James Bach refers to Checking versus Finding. Checking is the ceremonial approach to testing, for example ensuring that the requirements have been met, correctly. Finding is the use of skill and knowledge to identify issues not categorised anywhere else. Both are needed, both are poorly understood and managed within a lot of organisations.

However, the key challenge to me is the lack of understanding about testing. If we consider information to be raw data, knowledge to be the use of the raw data in a particular way and context to be the integration of knowledge into a specific environment, then I think that people don't understand testing deeply enough. I think they see the raw data (test scripts, charters, defects, tools etc.) and assume they "know" testing; they don't take that data and consider it relative to a goal or target to build their knowledge of testing; and they don't understand testing in the context of the value that it can provide to the organisation.

I think the faster development cycles are a real challenge for Australian companies, but a boon to the testing community. Faster feedback loops provide more and better information for testing. All we have to do is learn how to maximise valuable

information to the organisation while minimizing, but not eradicating, the time it takes to find it.

OZTester: Where do you believe Australia's approach to testing is going well?

I'm very lucky that I get a trans-Tasman view and I am able to engage with teams in both Australia and New Zealand. In general, I see that the level of understanding of the value of testing has really grown. I think organisations are seeing test teams contributing more important and focussed information and providing more value to the organisation. I think there is also a greater recognition of the specialist skills and approaches that testers have.

As I get to work with a lot of teams and meet a lot of brilliant people I've seen some really amazing applications of testing. I've seen a company that has built and implemented its own model based testing approach that is staggering in its coverage, cohesiveness and value. I've worked with organisations that understand the incredible value of having testing as part of the development lifecycle at every stage, with great engagement and enormous respect for the values that testers deliver. I'm seeing the rise of the concept of a "Test Architect" – a role that weaves enterprise level architecture approaches with test strategy and ensures that projects and products are built with a focus on quality as well as delivery at the architect level. I've seen organisations do 180deg. turns and change the testing engagement model to be test first (not last), then build (not only at unit level) but all the way through the lifecycle. I've also seen the growth of the use and application of test tools and approaches that have made the test teams smile! All of which lead me to think that testing is going wonderfully well in our hemisphere.

We have a strong and dynamic "can-do" attitude in Australia and New Zealand and over the years we have shown that, applied that, and grown from it. I find testers in this region to be articulate, keen, eager to learn and wanting to contribute more and more to both the community and to their organisations. It's very impressive!

OZTester: Where do you believe Australia's approach to testing could improve?

Improvement should be a constant thing and I am really keen to see the continued growth of the mindsets and community growth that we are seeing now. I think the secret is inclusive, not exclusive schools of thoughts and approaches.

I have noticed that there seems to be an aspect of elitism in some of the discussions and forums around testing in general and it seems to be quite prevalent at the moment, not just in Australia and New Zealand, but globally. I think we need to be more open minded about ways of thinking and talking about testing, rather than strict adherence to "one-true-way", more of an inclusive approach to thinking and learning and sharing about testing approaches and techniques.

I think that people need to consider the resources at their fingertips. There are heaps of external resources such as the internet, testing forums, Twitter, as well as user groups. But if people look at internal resources as well they can see a wealth of knowledge....knowing about testing does not preclude knowing about development practices, analysis practices or management practices. I think testers in Australia should seriously consider changing their language, changing who they speak to and change how they engage with the rest of the teams in the organisation.

I also think the current push for testers to be developers (or test script automators) is also flawed. There is a role for a tester who knows how to script; but not every tester needs or can grow those skills either. To me, the skill in testing is designing great test cases that exercise the test conditions that provide valuable information to the team or organisation. Writing scripts is something that we can get other people to do.

I think we need to look at how best to use the testing skill too. Late lifecycle engagement is the worst time for testing to be brought on board. The value of early testing is enormous and the injection of testing skills and mind sets right at the beginning of the lifecycle provides an enormous Return-on-Investment (ROI).

OZTester: Do you believe that overall the standard of testing in Australia is improving?

Yes...testing is the ultimate "trust" job – people have no choice but to trust their testing teams and to trust the information that the teams deliver to them. I think that testers are realising this and understanding that to deliver better value they need to hone their skills, enhance the information they are providing and deliver the best possible results that they can. The only way to do that is to learn and grow, look for and deliver improved services and not, absolutely not, stick to a niche!

I think with the investment in learning and the great learning approaches that are available to testers these days there is almost no option except to improve! There is this skills expansion and value injection happening in the community and I think we should leverage that as much as possible. I think it is vital to continue to grow testing as a role in software development and a vital skill set in any team. It is happening now, and it should be nurtured to continue to happen.

OZTester: Where do you believe the next initiatives in testing lay? What's coming next? In Australia? Internationally?

I think the next big focus will be on value-based testing (VBT) – bred from the world of Agile and testers understanding (and helping others to understand) that testing adds value throughout the SDLC, not just at the end (which is the worst time to test). Up-skilling people to be able to apply their testing skills without touching a keyboard. To be able to think using testing skills on the fly, in conversations, in requirements definition, in project planning, in architecture and design discussions.

The testers need to be adding value to the organisation, team and product otherwise they are impediments and hurdles to be avoided. This is my driving force; to enable and empower testers to be able to contribute at these sessions.

All the information I see from global engagements are about cross-functional conferences, tools and teams. We need as testers to be able to proactively integrate the testing skill into the delivery team at all levels. I think the next steps should be learning to

think about testing well, really well, and then articulate our thoughts clearly to the rest of the team.

Locally I see growth of hands-on interactive sessions as being a huge step forward and a continuing approach to how testers learn and share knowledge. The upsurge of weekend-based testing activities, crowd-sourced testing and the use of skills outside of the testing team is an indication of where we are going. Global communities are now closer together than ever so the testing community is growing closer, the skills we need continue to expand and we need to continue to foster their growth as well.

OZTester: Do you have a testing horror story to share?

Oh dear.....many, varied....my main failure was terrible!

We had a very script driven approach to one of our techniques (installing a new service on a server) with a complex procedure that was documented step by step; the test script included inputs and expected result. We had someone run the script manually, ticking off each step, watching the screen as the steps were completed, checking that the screen outputs matched the expected results. At the end of the scripted tests being completed the final step was to confirm the words "install completed successfully" were displayed. Sadly, I didn't realise that some install scripts had these words hard coded in the install script – so they displayed no matter what! And I also didn't realise that the tester had run the script (manually) and had watched the screen display error messages, failure statements etc, but because at the end, the expected result was displayed, the tester then moved onto the next step.

The installation test "passed" as all of the actual results matched the expected results but no matter how hard we tried we could not get the system to work. We ran the scripts many many times and it cost us days of heartache, until one day I sat with the tester to run the script together to see what was happening during the install.

The horror was that 1) The script did not include the lee-way for someone to realise that something was going wrong, 2) The tester didn't ask if it was meant to be like this and finally that the developers had hard coded success messages! That's why I like to

work with testers and empower them to think, understand and have the courage to say "I don't think this is right".

Thanks Sharon, some very useful insights here. I think you've hit the nail on the head with the inclusive/exclusive concept. There has recently been all too much confusion around "what's best; this way or that way" and along with others, NZTester and OZTester Magazines have attempted to address. Whilst the outcome is nearly always the somewhat frustrating "it depends", this does get us thinking more around the types of testing required, whether they're product- or project-focused and the general context within which the testing effort needs to be applied.



OZWST 2013 - Collaboration in Software Testing

by **Dean Mackenzie**

"I'm sick and I'm mad as hell about the way people are trying to teach collaboration now."

Rob Sabourin, OZWST 2013

With those impassioned words, **Rob Sabourin** set the tone for the Australian Workshop on Software Testing (OZWST) 2013 – a challenging, topical and at times uncomfortable examination of how testers in the software development industry view the nebulous activity, skill and concept of "collaboration".

OZWST is an annual peer conference organised and facilitated by the tireless **David Greenlees**. It's conducted in a similar fashion to the Los Altos Workshop on Software Testing conferences, which were among the forerunners of peer conferences in the testing field. This format differs from the typical convention, where one person speaks for the majority of the allotted time and fields a few questions at the end. At a peer conference, the ratio of participation is radically altered. One person presents an experience report for around 15-20 minutes on a particular experience relevant to the theme, which is then extensively questioned and discussed by the small audience (this session is known as "open season"). The questioning and subsequent discussion can extend for upwards of an hour – and this is where the real learning and insights can occur.

Day 1

The first report for OZWST was delivered by **Kim Engel** on how she faced a mechanically unique challenge in automation. The initiatives and strategies she employed in gaining informal engagement with stakeholders, raising enthusiasm and driving development for a "skunk-works" project demonstrated a conventional approach towards communication (meetings, formalised questioning) isn't always the most effective.

Pausing discussion only for morning tea, the conferees returned to listen to **Bruce McLeod** expound his experiences with trouble-shooting automation problems while leading a team. His

account centred on the concept that there are times where "independent innovation" (a one-man solution) can be far more effective than a collaborative effort.

One of the best parts to any conference is often outside the prepared agenda – it's the conversations and networking that occur outside the meeting rooms. OZWST 2013 was no exception, and the lunch room of Google Headquarters, replete with pool table, table tennis table, mini kick football table and cappuccino maker, was the perfect environment for discussing all things work, test and software-related.

After lunch, **Lee Hawkins** struggled through a sore throat to deliver a report on training an outsourced team in exploratory testing while dealing with language and cultural barriers. While he was softly spoken, his story spoke loudly on the possibilities that such a task could be done with any tangible success. In fact, a lengthy discussion arose on being able to specifically measure success of these kinds of activities (beyond simply saying "yes, it's been successful") that proved to be one of the most interesting points that were raised.

The conclusion of Day 1 involved a group discussion around the definition of collaboration, and signs of successful and not-so-successful collaboration. It was an engaging and lively debate, with plenty of point and counter-point. One of the most interesting threads that swirled around the room discussed how parties can have opposing goals but still collaborate to achieve an outcome, provided there is some common element between those parties.

Day 2

Sunday morning saw the conference resume without skipping a beat. **Henrik Andersson** joined the workshop, having just arrived in Sydney for the Tasting Lets Test (TLT) conference that was being held the following day. Although he spent much of his time working on TLT material, his insightful questions and observations throughout the day opened a lot of new avenues for discussion.

Paulo Lai took us through “a day in the life of a Google tester” – a fascinating and microcosmic example of working as a tester in the Google environment. It was amazing how many people that someone in his role needed to interact, co-operate and deal (in different time zones, of course) with over the course of an eight hour day. In these kinds of circumstances, the ability to “context switch” rapidly but effectively becomes an important (if not key) skill.

Rajesh Mathur followed on, recounting a not-so-pleasant experience of “anti-collaboration”. It was a compelling story for all the wrong reasons, but nevertheless was enlightening to hear his attempts to bring hostile groups from different countries together in an effort to achieve a minimally acceptable outcome.

Rob Sabourin, who was the content owner for the conference, gave his own “mini” experience report on how he worked with organisations that had conflicting goals to co-operate over a migration testing project. In light of Rajesh’s encounter, it made for an illuminating contrast as he had success in using an indirect approach with stakeholders to encourage involvement in collaborating with the other party, as well as obtaining their input into the testing. Noting that cultural differences could often be a bigger impediment to collaboration than language, it was also interesting to see how Rob’s experience shared a link with Lee’s and Rajesh’s, even though they were all different stories.

After another enjoyable lunch (the catering having been provided by an AST grant), the workshop broke away from the experience report format yet again. The participants divided into two groups to attack a pair of “open experience reports”, provided by **Andrew Dempster** and **Alessandra Moreira**, that detailed a couple of weighty test-related issues. Although on face value dis-similar, it was interesting to see how the problems decomposed, shedding the outer guise of technical problems and at their core being communication and relational issues – two important components to collaboration.

As the shadows lengthened outside, **Mark Tolfts** presented the final talk for OZWST 2013, recounting how he and others in a development team worked together to track down a bug that had been initially blamed upon poor testing. Two points seemed to jump out from his report – that the “heat”

by senior management provided a good deal of motivation to the collaboration, and the systematic problem-solving approach that Mark and the others used, decomposing the problem step-by-step rather than adopting the unfortunately more common “shot-gun” method.

Wrap-up

How do you define collaboration? Is it something that just happens? Can it be forced? What sorts of skills might help facilitate collaboration?

It was a difficult topic – to some it seemed quite straight-forward (and was it really a worthwhile topic?), while others couldn’t get their heads around it. OZWST spent two days deeply exploring the subject, and discovered no definitive answer. Such a mundane and theoretical exercise was not the purpose of the conference, however. It was the learning that each person unearthed that would be much more valuable – a new understanding of some of the complexities, risks, opportunities and skills needed to work in a collaborative fashion.

And, best of all – they debated, learned and grew collaboratively!

Dean Mackenzie is a Senior Test Analyst with The System Works Company and is based in Brisbane. He has progressed through a career in software testing with a genuine record of co-ordinating and contributing to the planning, design and implementation of system tests and user acceptance testing (UAT) for software releases, updates and special projects with enterprise-wide implications.

Dean can be contacted at dean.mackenzie@optusnet.com.au

My Twenty Year Software Testing Education Journey

by **Colin Cherry**

For me, *being educated* is not simply about being taught, it's about taking information from as many sources as possible and applying it in an appropriate manner. If this is done effectively, it becomes self-perpetuating and works like compound interest - the more you put in, the more you gather in return. My education as a software tester has often gone beyond this and taken on a form of *emersion*, where I have become totally obsessed with certain aspects of our craft.

My software tester education journey has taken over 20 years; beginning as a *novice* Tester and becoming, among other things, a Program Test Manager on various major Programs of Work, Test Practice Director at one of the largest software testing consultancies in the world and a Keynote Speaker at various international Software Testing Conferences. Along the way I also gained a qualification in Training & Assessment, after which I developed several of my own Tester Training Programs and delivered these to 1,000+ testers in Australia.

My transformation from Developer to Tester began in the late 80's, in the UK, while I was working on a major banking program and we needed to formalise our testing approach in order to gain user sign-off. This was my first encounter with the concept of UAT; more than 5 years would pass before I received any formal training though. Fortunately, my initial training providers were world-class and today are still recognised leaders of our profession - enormous gratitude goes to Dot Graham, Mark Fewster and Ross Collard for their ongoing support and friendship.

In March 1990, I left a very secure 6-year freelance banking assignment in Surrey, England for a new opportunity on the other side of the world in Melbourne, Australia. While my new role was initially a Development Team Leader, I quickly moved into the Testing team and was soon promoted into my first Test Management role. I then spent most of the 90's running various testing initiatives, the highlights of which were the merger

of three second-tier Australian banks and the restructure of the Testing department for a major Australian domestic and international airline. It was during this time that I developed my own training programs and attended a Test Automation introduction course in the UK.

It was around the time of the airline job that I was introduced to my first **EuroSTAR** Testing Conference in 1999, in Barcelona. I had a fortunate encounter at the conference when I met Donna O'Neill (CEO of IV&V Australia and renowned Software Testing Trainer from Sydney - via Boston, USA!) and we teamed up a few years later to create the **AsiaSTAR** Testing Conference concept.



In 2000, I returned to the UK for a year, spending the first 8 months designing and implementing the *soft launch* of an online supermarket and then redesigning the annual training and development program for a major testing consultancy. The 10 years between 1996 and 2006 were probably the most intense with respect to me expanding my software testing education as I grew in confidence. I also received many offers to work on major testing programs and building testing capabilities within many large organisations in Australia. It was not long after this that I was offered a job with the company that I had been wanting to work for all my life - IBM.

IBM was **the** benchmark computer organisation when I was a fledgling developer and to be sought out by them to work on several major projects was very special. It gave me access to so much knowledge and information from around the world and culminated in me redesigning their Australian graduate software testing module.

The real highlight of my time at IBM was my involvement in a project turnaround opportunity in Stockholm, Sweden where I spent almost 9 months redefining a Testing program and gaining user acceptance of a solution that had struggled to meet its implementation targets for almost 2 years. The education *bonus* for me on this project was the opportunity to learn Swedish and what really surprised me was that I discovered a whole new way to learn a different language. I was never good with languages in school and I've not had many opportunities since to immerse myself in a new language. So, what I discovered was that if I just took the words on face value and didn't try to translate them into English I picked things up far quicker. If only I'd known that 40 years ago!!

Being involved in the **AsiaSTAR** Conference in Australia in 2001 was a major part of my education strategy as it allowed me to share my passion with hundreds of other like-minded software testers. We were the first multi-stream Testing Conference in Australia with over 45 sessions delivered over three days each July. **AsiaSTAR** was hugely successful during its four year life, but sadly our backers in Europe decided to re-focus their efforts back on home soil. Those four years expanded my software testing knowledge far beyond what I ever imagined and led me to meet so many inspirational people. Although several conferences have filled the gap since **AsiaSTAR**, (IMHO) none have been as successful.

Another aspect of my software testing education has been a focus on formal qualifications and certification. I gained certification in ISEB/ISTQB (Foundation level) in 2000 and CSTP (Certified Software Testing Professional) in 2003. I found the CSTP syllabus and exam far more useful in terms of expanding and testing my knowledge. Sometime around 2002/3 I was invited to contribute to the ISTQB Advanced Syllabus however heavy work commitments at the time prevented me from doing so. With over 300,000 currently ISTQB certified it is by far the most widely recognised software tester benchmark.

To say my software tester education odyssey has been fulfilling would be an understatement. To say it has been special would still not approach my feelings. Software Testing has been *my life* for almost 25 years and I'm still as passionate about improving my skills and the skills of others

as I was at the beginning. We are so fortunate today to have access to so much material that the problem is no longer *finding* education resources but *discerning* which is worthy of recommendation and the investment of time and money.

Colin has been around a bit and that's an understatement if there ever was one. He is currently the **OZTester Magazine** Editor and an esteemed software testing advisor. Colin can be contacted at colinda@ozemail.com.au

TAKE THE NEXT STEP.



ISTQB CERTIFICATION SPECIALISTS

Offering ALL current ISTQB Career Path Courses

- ISTQB Foundation Certificate
- ISTQB Advanced Test Manager
- ISTQB Advanced Test Analyst
- ISTQB Advanced Technical Test Analyst

new

Online ISTQB Foundation
Available Now!

DISQOVER.COM.AU

P: +61 3 9836 2901
E: INFO@DISQOVER.COM.AU

 **Disqover**
Improving Software Quality

Training and Certification: Let's Tidy Things Up a Bit

by **Sharon Robson**



In a recent blog from the very active NZ Testing Community, I read that training and certification are often viewed as being one and the same. It made me realise that there appears to be some confusion in our industry around the separation between the two. In this article, I am hoping to clear some of these up. To me there is a huge difference between training (the imparting of knowledge) and certification (the assessment of knowledge). I cannot agree more with the current topics being discussed in the testing world. I do not think that the certification at foundation level truly reflects knowledge of testing but then I wonder whether it was really ever designed to.

Lumping training providers and certification bodies into the same category shows a lack of understanding of the process and of knowledge of the subject. Based on my experiences as a member of a certification body, a customer of training providers and now as a training provider myself, I hope I can impart how it all works. I have been very lucky to have seen both sides of the fence and appreciate the opportunity to share this knowledge with you.

What is the difference between training and certification?

Training and certification are terms both revolving around a common reference point, this reference point is called “knowledge”. The role of training is to impart or share knowledge, the role of certification is to test knowledge. First of all we must all realise that they are very different roles with very different approaches and objectives. Training is a divergent activity; it opens up vistas and unexplored horizons to us. Certification, on the other hand is a convergent activity; a method of fencing knowledge into a few questions to be answered thus narrowing knowledge to a set of learning objectives to achieve or otherwise (pass and fail).

What is certification used for?

Testing in general and with the International Software Testing and Quality Board certification (ISTQB, the most popular) specifically, the role of certification is to provide a common and global baseline of knowledge acquired. It is possible to gain ISTQB certification at any level (Foundation, Advanced and Expert) with no training as attending training is not a prerequisite for any exam. All that is required for the Foundation and Advanced Certification is to sit and pass an exam. This exam is a series of multiple-choice questions designed to assess the candidate's knowledge. Questions are set at different levels (K Level) designed to assess the candidate's depth of knowledge on a particular topic. K1 is a simple definition or recall of a term; K2 requires the candidate to understand a concept or pick a concept that has been paraphrased; K3 is the application of a skill to provide an answer and K4 is the selection of a skill or technique to apply to provide an answer based on analysis of the question and context. The higher the K Level, the more difficult the question and therefore more knowledge is required to be successful in the assessment thereof.

However time and again the industry has seen that candidates who hold the Foundation certification do not know how to apply even basic testing skills nor do they understand the terminology. Why is this? Because there is no training required to sit these exams. Anyone can download the syllabus and sit the exams at any time. The premise behind this approach is probably that there are a lot of very skilled testers out there who need not spend the time nor money to go through training. However what it leads to is an awful lot of under-skilled and in fact, unskilled certified testers!

Certifications are used by testers to land new jobs and by hiring managers to short list testing

candidates. Prior to a basic certification, the knowledge of what makes a good tester, what a tester does and who is a good hire, was very difficult to determine. By setting up certification schemes, organisations were able to attempt to baseline skills and knowledge.

At the very best, Foundation Certification proves that the candidate has heard of testing, may have read the syllabus and has taken the time to get certified. It proves nothing about skill or exposure to knowledge about the industry.

Training on the other hand is all about imparting knowledge - ideally that that the candidate requires. But therein lies the conundrum: if the candidate wants nothing more than sufficient knowledge to pass the exam, should that training have to be undertaken in order to sit the exam?

Why training is so expensive?

Knowledge has long been treated as a commodity and in our merit-based societies, the acquisition of knowledge is seen as highly desirable. As such, training providers are often seen as the “bad guys” because we charge for acquisition of knowledge. As a training provider, I have to tell you that a huge investment goes into building and delivering knowledge. It cannot be done lightly if quality is required. The better the quality of the training, the the more comprehensive the knowledge shared. The better the quality of knowledge required, the better the training needs to be.

If the role of training is to impart knowledge and the role of certification is to test it, it is key that there be a clear distinction between training and assessing, otherwise the assessment itself can be suspect. In the world of testing we have a prime example with the ISTQB. ISTQB consists of a central body with local bodies providing an examination/assessment framework for the generation and administration of their exams. Associated with these bodies are training providers who are separate entities designed to impart the knowledge that is to be assessed. These are usually commercial organisations who have paid money to be approved and counted as an “accredited” training provider. This means that accredited courses that the training provider delivers have been assessed as suitable in style and coverage to meet the needs of the certifying body.

Self-training by studying the syllabus only can be undirected, unstructured and unproductive. Informal training such as study groups, online forums and networking events can be driven by strong personalities or misguided understandings. They often lack cohesiveness and structure and make little allowance for new people to the group. How often do we see the poor neophytes attempting to ask basic questions online and getting criticised for it? Formal training will follow a structure, a known path and is specifically designed to meet the candidate’s needs. However as we know, formal training can be more expensive.

When it comes to training, candidates must ask themselves “what do I want from training?” and find a training course that delivers on that objective. If it is certification; then expecting anything more is not really viable. If the objective is to gain knowledge and acquires skills then a course that prepares you for certification is not a good fit either.

However the question always arises: how do we prove that someone has acquired knowledge? Training providers have a commodity to supply to candidates....knowledge. Not all candidates want to acquire that knowledge prior to attempting certification nor is all knowledge structured to grow the candidates either.



Some training providers will have courses that are specifically focused on passing the exam. All the course covers is what is needed to gain a pass mark; no context, no background to the learning and little or no application of the skills. This is an exercise in teaching key words, phrases and tricks to correctly answering questions presented in the exam.

Little, if any testing knowledge is imparted in these courses. I, for one, dislike delivering these courses as they are not about testing, they are about passing an exam. Training should be much more than that...it should be about thinking, adapting and learning skills to be used consistently at the workplace - multi-dimensional training not single-dimensional, focused on passing the exam. And here's the challenge: to train testers by going beyond the syllabus and indeed, beyond the collection of works produced by industry bodies. This requires skill to find, compile and pass on.

The training of software testers should as a minimum I believe, require the use of software. I cannot understand how a software testers' skill can be adequately assessed without it! I also don't believe that people should necessarily know how to do things intuitively. Training needs to be structured to grow knowledge, not present information as a finished state. Good training should provide not only the understanding needed for the direct application of skills but also that for the continued growth of those skills. One of my key learnings from my university days was not the knowledge I acquired in the subjects I studied but the knowledge of how to acquire knowledge, how to learn and how to expand my horizons through my learning.

In this day and age we are often restricted by time, so a training provider builds a course that will provide that information for you in a structured and cohesive manner. The provider also goes to the effort of constructing a framework of notes, slides, exercises, references and further reading to assist in future knowledge acquisition. This, is why training costs.

What options do we have?

I believe certification was designed to provide a stopping-off point for future knowledge acquisition. It has unfortunately in some cases been used as a short cut for both candidates and hiring managers to circumvent appropriate learning around testing and how to assess a tester's abilities. I think the answer is to take certification out of the theoretical paper-based world where we use words to assess skill and move it into the practical world where we use the application of skills to assess. The challenge will then be how to assess that? How do we build a mechanism that allows comparative assessment that clearly shows the difference

between someone with the skills and otherwise. I believe that herein lays the next differentiator for training providers to pick up on.

Conclusion

Knowledge acquisition should be a constant and an ongoing quest for anyone looking to grow their career. For me, one of the first things I recognised when I became a tester was that I did not know enough about the role nor about the skills I required. I sought out knowledge and found training that allowed me to sit an exam and become ISEB Foundation Certified (yes, before ISTQB!). As part of that process I also found more references, knowledge sources and information beyond the syllabus. I found a community of thinkers and sharers of knowledge. Some of them shared their knowledge for free, others charged for it in the form of subscription fees, book purchases or course costs. I generally found that the old adage "you get what you pay for" was very true. A lot of the free information that was available was not necessarily good information. It was food for thought but I often had to reject it on one level or another. Thankfully not all of it was like that....there was pure gold amongst the dirt! However it took a long time to find those gold nuggets that I now freely share those with anyone who asks.

Editor's comment: *I've always seen the ISTQB Foundation Certification as merely a baseplate. The "meat" comes in the Advanced and Expert levels where skills and knowledge are assessed by other as well as multiple-choice questions.*

Sharon Robson is the Software Testing Practice Lead for Software Education having been in and around IT for over 20 years. She was a founding board member of the Australia New Zealand Testing Board (ANZTB) and Head of Marketing for the International Software Testing qualifications Board (ISTQB). During her time with ANZTB Sharon was actively involved in the establishment of the local certification scheme and exam question generation. Once she became a training provider, Sharon moved into promoting the need to acquire knowledge and has spent most of the last decade attempting to reconcile training needs with certification goals. Her main focus is on sharing knowledge that will help people enjoy their jobs more. Sharon can be contacted at sharonr@softed.com

So What Do The Educators Think About the Education?

OZTester Magazine asked a couple of testing education companies (among other services) principles to give their views on the state of testing training in Australia

Tony Bailey, CEO, Access Testing

I find Tester Education in Australia to be at best limited with far too much focus on commoditised technical certification rather than education for thinking, understanding and quality. There is clearly an emphasis on ISTQB Certification for testers and at a very base level (namely as a glossary of testing terms) this has its place but I'm not a fan of the commoditised approach to education. Unfortunately, this commoditisation is endemic within our industry and I guess the flaws of this approach can be really seen when a relatively smart person can spend approximately \$7,500 on courses and invest approximately one month of their time (with only optional prerequisites for testing experience) and end up with the three separate ISTQB Certifications (Foundation, Advanced Test Analyst & Advanced Test Manager) with still no hands on testing experience. This is clearly wrong and does nothing to instil confidence in our Industry. We can all be thankful that nurses and doctors don't operate this way!

In most IT University degree courses, testing is barely touched upon so I guess we can't be too surprised with the level of tester education at present. I also find it incredible that there is virtually no focus on the "softer skills" in most tester education. The report writing, the communication skills, the interpersonal skills are all basically left to the individual to develop and in an industry where we are only too happy to hide behind a desk, tool, email or report, this creates many problems.

The industry needs to realise that as technology becomes more and more pervasive, technical skills are only a necessary condition for success, not a sufficient one. Learning the "how" is required but learning the "why" is more important. Testing

nowadays is more about economics, management, communication and meeting customer expectations than ever before and these are the precise areas that our technical education focus completely avoids. This leaves testers in a bubble largely removed from the people they should be communicating with the most - the business and customers. It's hard to be relevant in a vacuum.

Toby Thompson, Managing Director, Discover

I was a pretty poor student in general – I was an average student at school which was partially down to an undiagnosed hearing deficiency and partly due to a lack of interest.

Where I did excel though was with subjects I was truly enthused by – I aced subjects like Ancient Greek and Latin. These aren't subjects most people would jump up and down about – and possibly I wouldn't have, were it not for the teachers. These people just loved their subjects and this was evident in the way they presented with passion and an endless generosity to share their expansive knowledge. Had it not been for their enthusiasm I imagine I would have been less interested in the subjects.

I fondly recall an elderly history teacher, who often regaled us with stories of his years of active military service during the Second World War, whenever we got bored of what was in the school curriculum, we used to ask him about his experiences – the entire class were hanging off his every word – we were so excited to go to his lessons for that very reason.

Passion for and experience in the subject being taught are extremely important attributes for the educator. You don't want your audience to switch off – you want them to be engaged and motivated and eager to learn more.

I love to teach. I have been teaching on software testing themes for more than thirteen years – I have also been a testing practitioner in the field and as a result much of my teaching is drawn from real world experience – sometimes I borrow other peoples’ real world experiences because they tell a better story and convey more meaning with regard to the topic I am exploring with the class. I relish learning and educating myself on all things from the software testing arena – I collect articles, books, presentations and heaps of tools and I share them with my students. They are not specific to a syllabus or curriculum but just interesting and helpful pieces of the complex testing jigsaw.

Thirteen years ago when I started teaching the topic, testing was by no means as popular as it is today – most people I met back then had never heard of testing and if they had they weren’t entirely sure what it was. There was no real shared terminology or clear understanding as to why it was needed so it was often seen as the lowest part of the food chain often treated with suspicion and disdain. Thirteen years of testing education later and the testing world is almost unrecognisable. This must in part be down to the education of aspiring and existing testers in class-rooms and via a myriad of other sources.

In class-room education the role of the teacher is of paramount importance – if the teacher doesn’t enjoy or believe in their work then the students suffer as a result. If they can’t bring their teachings to life then students will struggle to identify with the concepts being taught. It is the teacher’s responsibility to encourage, facilitate, inspire and address each student’s requirements – to keep the student engaged and enthralled and motivated.

This kind of education starts in a classroom but certainly shouldn’t end there. When that class room session has stopped the education should live on and the student also has a responsibility to go and educate themselves and to explore and experiment with the new ideas they have learned.

To become a complete tester in just a few days of tester training is an unlikely outcome – for any kind of training for that matter – I do however believe that you can gain many new insights and ideas and broaden your testing horizons by listening to not only what the teacher is trying to explain but also by listening to the experiences of your peers.

The teacher’s role in education is obviously

important – but the responsibility lies squarely on the shoulder of the people who have come to learn. There is a lot to be said for self education and much of this can be done on the job, via mentors, and much can be gleaned from domain specific literature, attending conferences, special interest groups, meet ups, forums, blogs, social media and so forth.

In my many years of teaching I have asked the question to my students – how many of you own a software testing book, go to testing conferences, proactively learn about your profession? The response is often they’re just too busy working in testing to learn about their profession. Life gets busy and learning is often put on the backburner – however some of the learning just might make their life easier, might help them articulate the complexities of what they do to senior management, to have a better understanding of the role they play within the development life cycle.

Interestingly I also often get – I wish the project managers / developers / business analysts could come on this course then they would understand what testing is really about. I have constantly encouraged my students to go and teach and educate these types of stakeholder – a bottom up approach – what better way to learn about what you do then to teach those that sponsor you or those that you interact with

There are many sources of testing education out there – trust me I have trawled through reams of testing related literature on the internet and beyond.

You can discover an array of tools, processes and techniques that you can add to your skill set. I have always held the belief that if you can learn one valuable lesson from any educational source and implement it – in whatever way suits the context – then there is value in that education.

The software testing discipline is one that is still not being comprehensively covered at the tertiary education level which is why it is so important for people in the testing community to keep learning so they can find better ways to execute their work. I have learned so much about this ever growing discipline – there is so much more to learn and so much more to teach. Whether it is through the ISTQB or another medium – I say it is never too late to become better informed about what you do and to improve your utility in the software testing field.

OZTester & NZTester Back Issues

Click to download

OZTester #2



OZTester #1



NZTester #4



NZTester #3



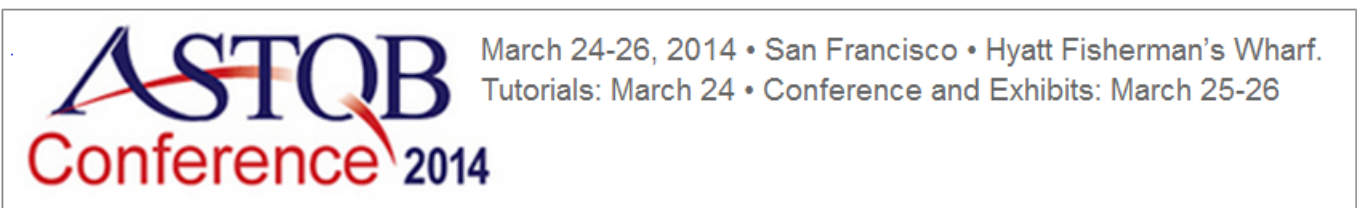
NZTester#2



NZTester #1



Testing Events



TESTING MOBILE APPLICATIONS

Effective Testing for Smartphones and Tablets

Created and delivered by Jonathan Kohl, an industry recognised thought leader in mobile application testing.

SYDNEY 3-4 February **MELBOURNE** 6-7 February 2014 \$1870 inc GST

This MasterClass is specifically designed to empower testers to:

- make the paradigm shift from web and PC application testing to mobile testing
- understand how mobile technology works, and how to exploit it when testing
- quickly discover and identify problems that are important to mobile users
- use strategies to choose which device platforms to test
- add structure and guidance to better manage mobile testing activities.

BOOK NOW

Go to www.softed.com/courses/TMA

Email info-au@softed.com

Phone **1800 145 152**

* Every 5th person attending the same course (date and location) from the same organisation attends for free.



**software
EDUCATION**
www.softed.com

*For terms and conditions visit www.softed.com/register/terms

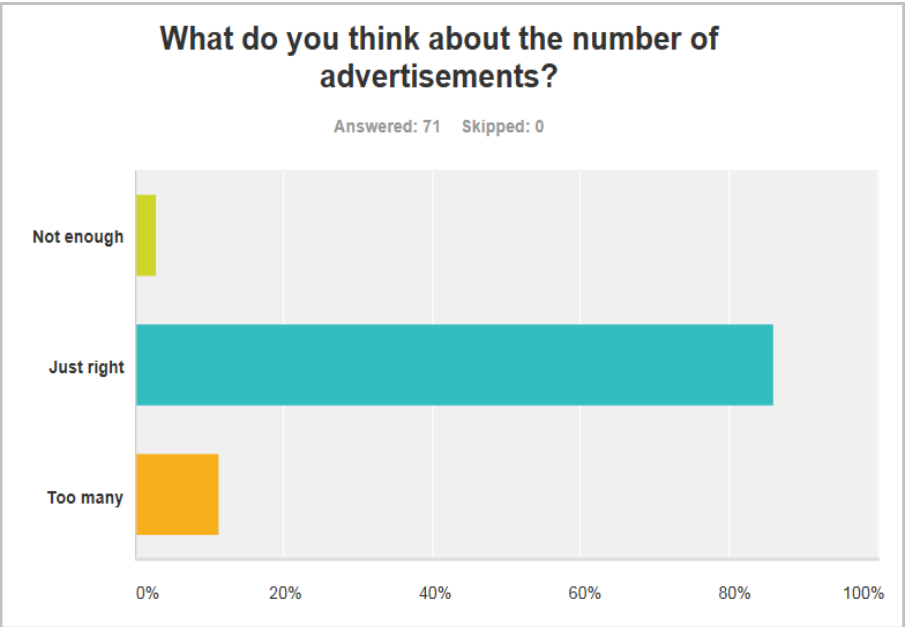
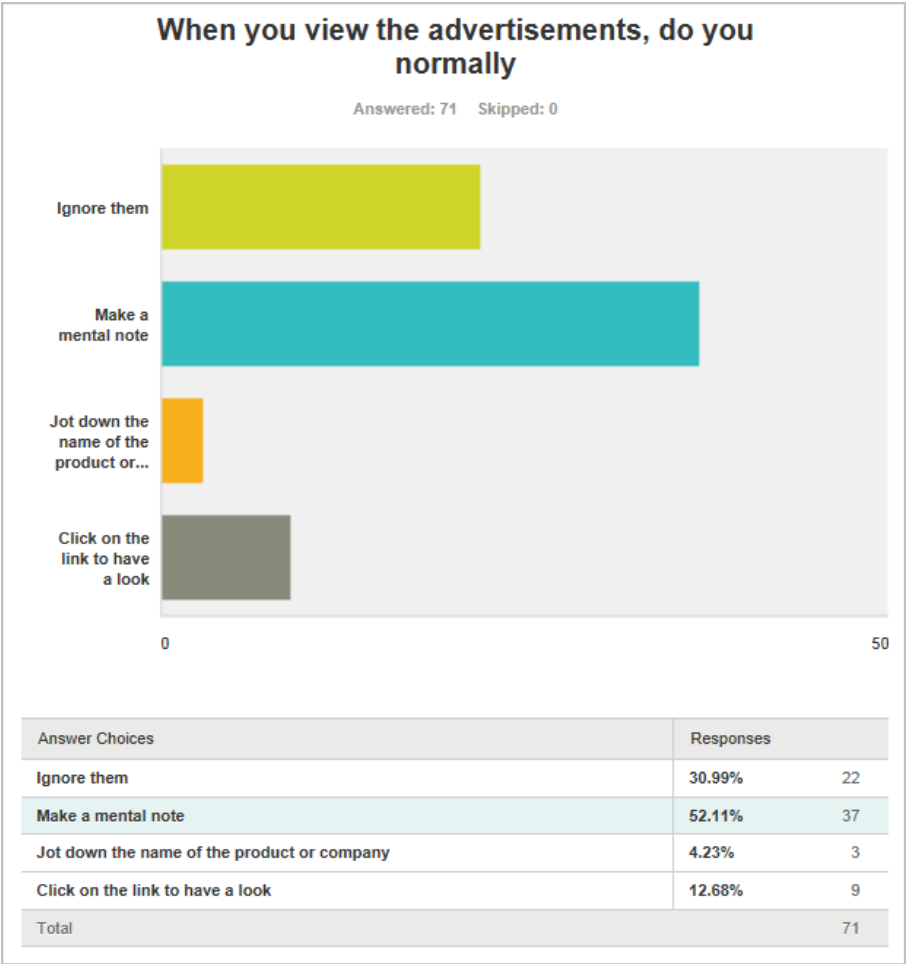
Copyright © 2013 Software Education Holdings Ltd. Correct as of October 2013.



OZ & NZTester Magazine Survey Results

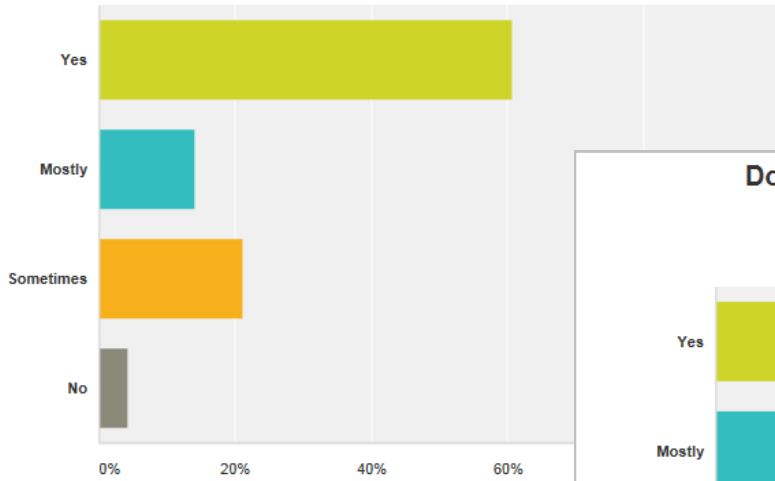
Back in April, to coincide with the release of NZTester 3 and OZTester 1, we set up our first survey. As always, we were keen for further feedback on the magazines’ content and what our readers would like to see in future issues. Space limitations prevent us from publishing the full set results so we’ve selected those that we believe would be of the most interest. The full set can be found [here](#).

We would like to thank everyone who completed the survey, all seventy-one of you. The results in many cases were as we expected however there were a few surprises, as there always are.



Do you find the guest interviews of interest?

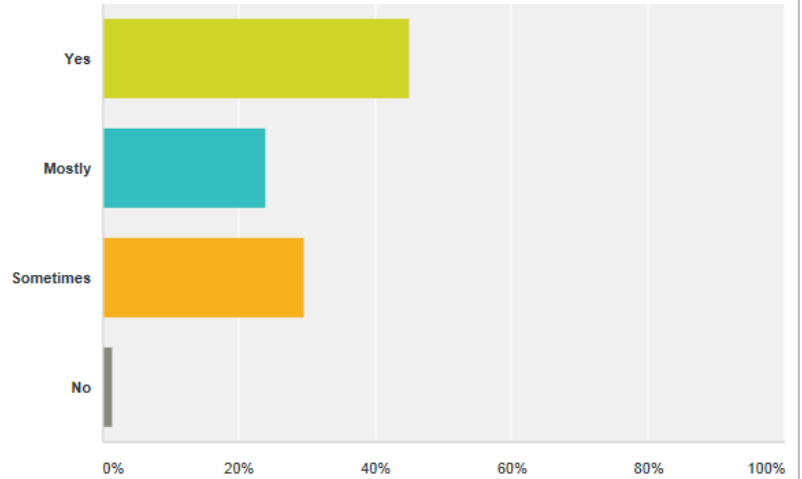
Answered: 71 Skipped: 0



Answer Choices	Responses
Yes	60.56%
Mostly	14.08%
Sometimes	21.13%
No	4.23%
Total	

Do you find the conference reviews of interest?

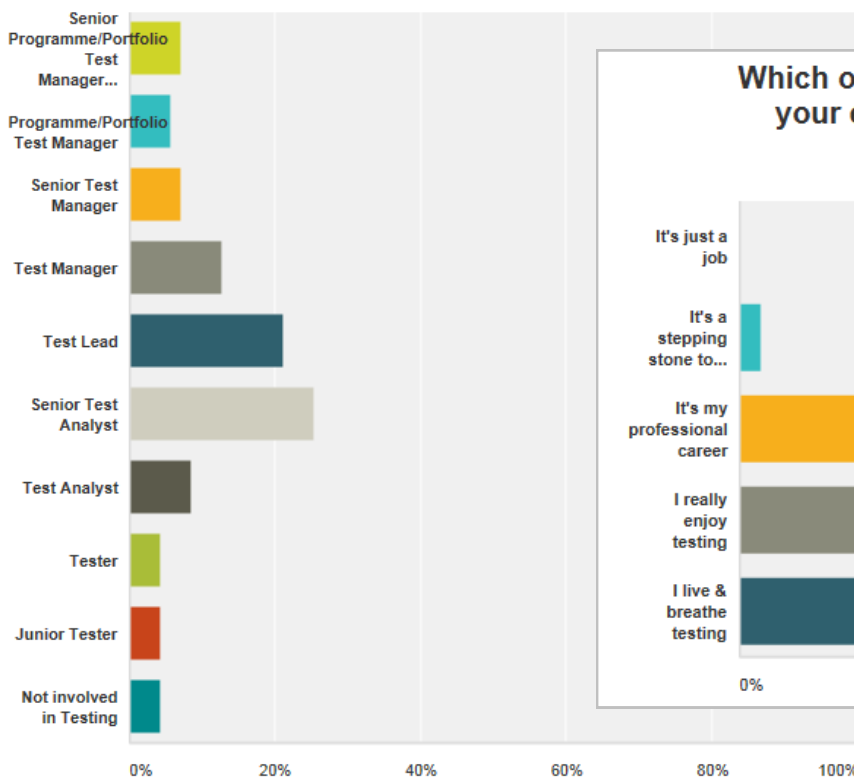
Answered: 71 Skipped: 0



Answer Choices	Responses
Yes	45.07%32
Mostly	23.94%17
Sometimes	29.58%21
No	1.41%1
Total	71

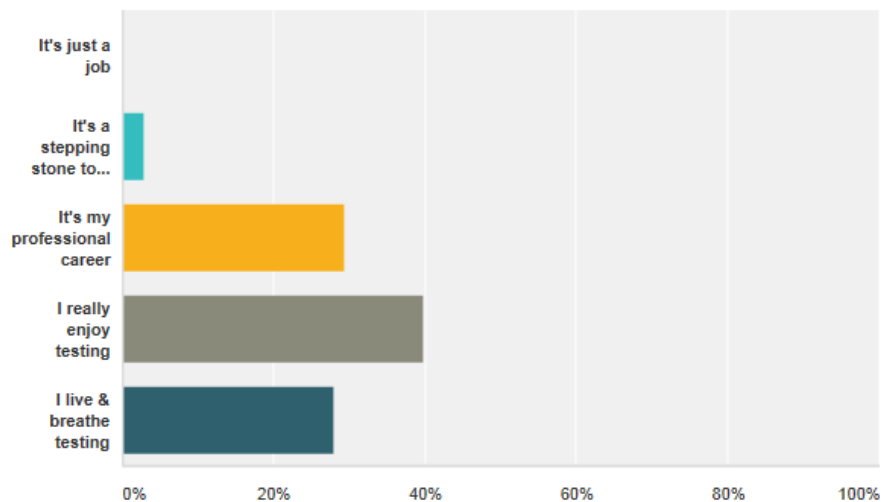
What testing level do you consider yourself at?

Answered: 71 Skipped: 0

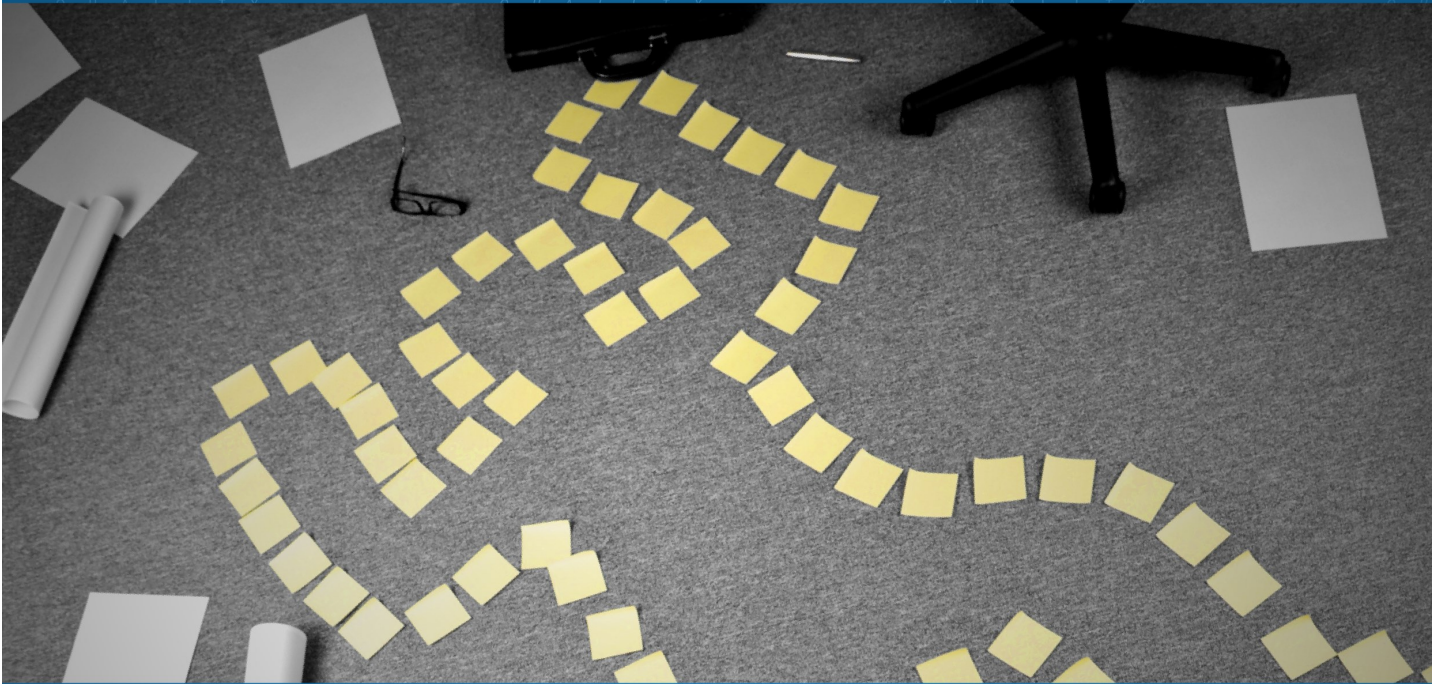


Which of the options below best describes your enthusiasm/passion for testing?

Answered: 68 Skipped: 3



IMMEDIATE OPENING: FULL-TIME QUALITY ASSURANCE MANAGER



Don't work yourself to death. Manage your testing effort with TestTrack TCM.

TestTrack TCM puts you in control of test planning and tracking, providing better visibility over the testing effort and giving you more time to manage your team.

- Know instantly which test cases have been executed, what your coverage is, and how much testing remains.
- Manage suites of platform-specific compliance tests, functional tests, and performance tests in one central location.
- Assign tests to your QA team, track results, and report on performance and workload.
- Use test variants to target multiple platforms with the same test case for more efficient test case management.
- Link defects to failed test runs and improve test coverage by creating test cases based on existing defects with seamless TestTrack Pro integration.
- Schedule automated tests to run 24x7.



Visit www.seapine.com/oztester to learn more.

**Free
30-Day Trial**

© 2013 Seapine Software, Inc. All rights reserved.

A Tale of Two Testers

by Erin Donnell (right) & Jen Hurrell (left)



Like many others, our entry into testing was a happy accident! We started out temping on a small software project in 2008. Jen had just sold her restaurant and started on the project as a proofreader. Erin was taking a break from university and started as a data analyst. Over time, we were co-opted to help with testing – despite the fact neither of us knew what a tester did or why testing software was necessary!

2010 rolled along, the project went into maintenance mode and most of the team disappeared with it. Suddenly, as the sole ‘survivors’, we were in charge of testing (and just about everything else.) We have worked closely together ever since. Back then, neither of us dreamed we would do testing in anything other than a temporary ‘fill-in’ capacity. But a few twists and turns later, in 2012, someone said “you guys are testers, right? Can you come and test this?”

We had great domain knowledge on our old familiar project and felt relatively comfortable testing it. However, we both felt uncertain when these new projects came our way. How do we test with no prior knowledge? With no requirements? How do we plan this, where do we start, and how do we know we’re done? Thus began the education journey that has made us into the testers we are today.

To explain why we did not at this point just ask someone a bit higher up how to go about all this, our testing ‘team’ is a team of two - Erin and Jen...sometimes called ‘Jerin’ by our colleagues! We are based in Auckland. We knew no other testers in Auckland or anywhere else for that matter. At this point the amazing testing community we are beginning to find remained undiscovered. And so, the tale begins....

Neither of us had ever read a testing book – so this seemed like a safe place to start! We raided Amazon and the local libraries. We started with Lee Copeland’s “A Practitioner’s Guide to Software

Testing”, an easy-to-read introduction, and loved James Bach’s ‘Lessons Learned in Software Testing.’ Our lightbulb moment came when reading Gerald Weinberg’s ‘Perfect Software’ and we realised testing is a cognitive activity as much as it is a technical one! Once we got started, a frenzy of Googling testing articles ensued, along with a raft of discussions (and arguments...) with each other about all things testing.

One of these arguments was about Twitter. Jen flatly refused to have anything to do with social media. After careful manipulation and cajoling on Erin’s part, she conceded.

Surprisingly, joining Twitter turned out to be one of the best decisions we could have made. It led us to discover different testing techniques, opinions about testing, and opportunities for learning about testing. Some examples: CodeAcademy to learn Javascript and CSS, Hexawise to learn combinatorial testing, and Skype testing coach Ilari Aegerter, who helped us immensely. Even if you are not a fan of social media, we would encourage testers to consider joining - It is a great way of slowly introducing yourself to the wider testing community.

Fast forward a few months, we stumbled across an advertisement for a James Bach talk in Wellington. To our dismay, the talk was full. Having read his blog, we knew James wasn’t one for following the rules. So, Erin decided to e-mail him and ask if he could let us come, anyway. We offered to sit on the floor. We were stunned when James came back ten minutes later and said “If you aren’t allowed in, then I won’t speak”! So, off we went to Wellington. We met James Bach and he introduced us to Aaron Hodder. This was our first ‘in-person’ connection to the testing community in New Zealand.

Meeting James Bach and hearing him speak was a serious turning point. His enthusiasm for testing was infectious and for the first time, we considered that testing could be a real (and exciting) career.

Encouraged by James' talk, we decided to enrol in the BBST (Black Box Software Testing) Foundations course. This is a four-week online course run by the Association of Software Testing. It involves online lectures, group assignments, multiple choice quizzes, and a long-answer final exam. It is fast-paced, and is designed to encourage critical thinking about testing. The course has a reputation among those who have taken it as being fairly 'intense' – and we can attest this is definitely the case. We both really enjoyed this course, and passed it, despite much nail-biting over the final exam! It was also great to link up with testers from all around the world who had a range of different contexts and perspectives, ranging from automation testers to less tech-savvy testers, testing everything from games to military drones.

Although BBST gave us a good theoretical background, we felt like we needed an outlet to practice some of the new techniques we were discovering. We signed up to uTest - a crowdsourced testing platform where testers are paid to find bugs on short-term projects. uTest has given us exposure to technologies that we would not have otherwise had the opportunity to work with in our day-to-day job, e.g., mobile testing. It's definitely not a get-rich-fast scheme but in a relatively short space of time, it paid for Erin to go to 2013 Tasting

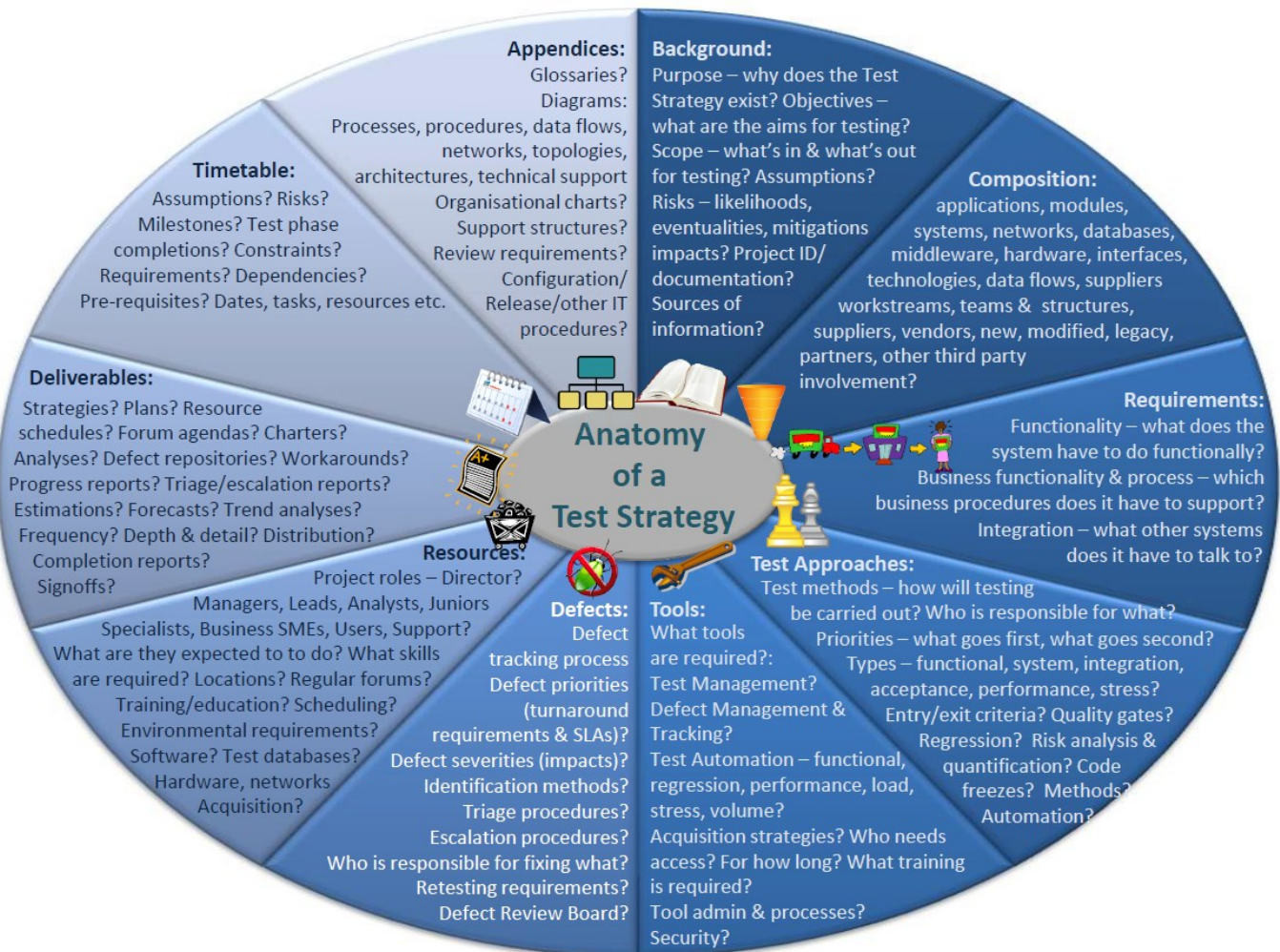
Let's Test in Sydney with Jen.

A surprising outcome of our chat with James Bach was being invited to the 2013 Kiwi Workshop for Software Testing (KWST3) – a peer conference. When we received the invite, it talked about industry leaders. Our reaction was "Us? Industry leaders in software testing? You have to be kidding!" – followed by many discussions about whether or not we would go, angst about preparing an experience report, and concerns that we would be ripped to shreds in the 'open season' debate. Needless to say, we went. The total opposite happened. We were made to feel totally welcome, in an environment where we could share our ideas and experiences freely.

Looking back on our journey so far, we are amazed at the learning opportunities we have had. We have learned how to ask questions, cope with confusion, and mix and match scripted and exploratory approaches as needed. Perhaps most importantly, we have learned how to seek out the information and help we need as we need it. There truly are many opportunities for self-education in testing - whether it be from books, courses, or mentoring from other testers. Many of these options cost nothing more than time, dedication, and sometimes, the courage to ask for help.

Jen Hurrell and Erin Donnell are both self-employed testers. They are based in Auckland and currently work together at Codec Software. Jen can be contacted on jenhurrl1@gmail.com and Erin on ecdonnell@gmail.com

Click on the infographic for a downloadable .pdf



© Copyright 2013 - NZ Tester Magazine, all rights reserved.



TEST YOUR WORLD WITH ENTERPRISE TESTER

"Enterprise Tester is by far the best tool out there for agile test case management and integration into open tooling such as JIRA."

Tom Gilmour -
Head of Quality Infrastructure, PayPal

Developed by your local Atlassian Experts



catchsoftware.com/oztester

Tasting Let's Test Review:

by **Geoff Horne**, Editor, NZTester Magazine

The Let's Test conferences are the brainchild of Swede Henrik Anderssen who established the Let's Test model and kicked the first one off in Sweden last year. A student of the Context-Driven Testing (CDT) school, Henrik saw the opportunity to create a community of like-minded test professionals and, partnering with locals of the same thinking, will be bringing a Let's Test conference to Australian shores next year.

This one-day "teaser" held in Sydney on 5 August was designed to introduce Let's Test to the local community and if the feedback is anything to go by, I would suggest the flagship conference scheduled for September 2014 will be a resounding success.

No matter what readers may think about Context-Driven Testing and its half-brother Exploratory, there is no doubting that Henrik and his henchman (sorry, and henchwomen) are absolutely bona fide, sold-out "passioneers" and "empassioneers" (how about those for terms, perhaps I should copyright?!) for software testing and the quest for better ways of doing so oozes from their every pore (especially under the lights when presenting!).

The day was kicked off by Adelaide-based CDT-man David Greenlees (see David's articles in NZTester 1 and 3) who outlined the ground rules for the day and introduced each speaker. Richard Robinson then explained how the test lab was going to work and laid down the gauntlet to those attendees who felt up to the challenge of Andrew Robins' robots and Lee Hawkins' dice games among others. If you ever get the opportunity to play the testing "dice game", do so - it is a brilliant way of demonstrating how to think creatively when considering different testing combinations and options (hint: the rules sometimes change).

The first presenter of the day was my fellow classic rock fan from Canada, Rob Sabourin. I always enjoy Rob's sessions; he has the knack of explaining complex situations with no more than four words. One of Rob's key messages is that software testing,

even within the context of a single project, needs to quickly adapt to the changing landscape upon which the testing effort is operating. We cannot assume that everything will stay as it is at the outset of a test exercise and that a finger needs to be constantly kept on the pulse throughout. I've often likened it to having a "tiger by the tail" - as soon as you grab him, he's off at top speed. Through every twist and turn he takes, you have to hold on for dear life because if you let go, chances are he will have disappeared into the undergrowth before you've even hit the dust.

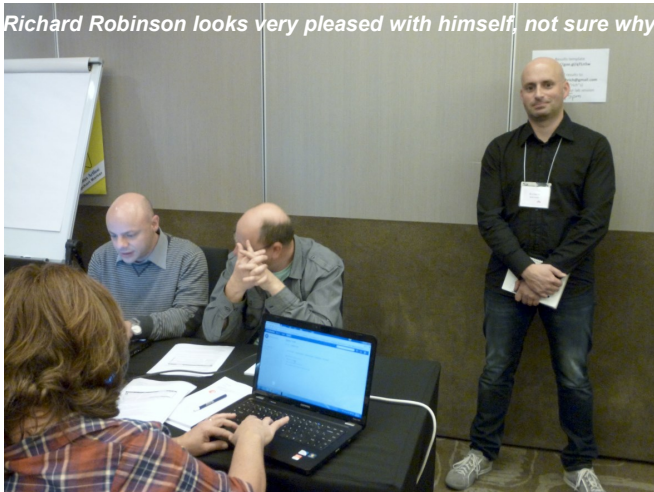
Next up was Mr Exploratory himself, Erik Petersen. I've known Erik for many years having shared the StarEast/West podium with him many times. Erik is generally credited with being the first test practitioner to structure an exploratory testing approach into a commercial value proposition. For his session, Erik talked about his CPIE model of Collation, Prioritisation, Investigation and Experimentation as a framework for conducting exploratory testing. He then moved the audience through to the test lab to demonstrate first hand. I found this approach quite refreshing as the session resulted in the immediate outworking of the practical application from the theoretical concept.



l. to r. Erik Petersen, Tim Coulter & Henrik Anderssen

During the lunch break, I had the opportunity to meet, amongst others, Hong Kong-based OZTester writer Rajesh Mathur (see OZTester 2). Rajesh is currently Test Delivery Manager for Cathay Pacific Airlines where his responsibility encompasses all testing not just applications software, as we were to find out from his session later in the day.

The dreaded après déjeuner slot was admirably filled by Anne-Marie Charrett who delivered an impressive presentation around building a Context-Driven Test Team. Anne-Marie introduced me to a new word: “autodidactic”, which had me reaching for my iPhone Oxford Concise English Dictionary - it wasn’t there however I think I got the drift (a propensity to slip into teaching mode) . I can imagine working in one of Anne-Marie’s teams would be a delight as she has such a gentle way of communicating difficult concepts – not like this old bull with a sore head!



Tim Coulter, all the way in from New York, then took us through his six-year employment history showing how each employer had fared when stacked up against CDT concepts. Far from this being a critique, Tim showed where each had the opportunity to improve their quality initiatives and what the results were when they either did or did not take up the opportunity. Some of the outcomes were a little scary!

Rajesh was next up, using his current employ at Cathay Pacific as an example of how the essence of CDT is really nothing more than project-appropriate application of skill and judgement. He showed how Cathay Pacific has deployed such practices when e-enabling its aircraft fleet (the connecting of all aircraft to the ground digitally via wireless capabilities). Rajesh also showed how small issues can have such huge consequences using the Arian 5 rocket explosion in 2005 as the example. The Arian project was 10 years in execution with a cost of over \$7 billion, valued at over \$500 million and lasted a mere 29 seconds in flight. The cause? The attempted conversion of a 64-bit integer into a 16-bit integer creating an overflow exception. I often use similar examples to demonstrate how small software changes do not necessarily equate to small testing efforts.

Finally, Henrik took the platform to talk more about the aims and objectives of Let’s Test and to encourage everyone to spread the word for the flagship conference next year. With Richard following to award prizes for the test lab competitions and David to round off proceedings by thanking all who had made Tasting Let’s Test a success, the day was over. Or was it? By the time I departed I’m Angus Steakhouse in Cockle Bay around 11:00pm, a hardy core was still going, sustained by beer and red wine (I shall not name and shame, you know who you were eh Rob!).

No matter how much I might think I know, I always learn something new at these events. While each session offered its valuable insights, I found Rob Sabourin’s illustrations around changing testing contexts of particular value, especially for when I next need to impart these understandings at senior management levels. As usual though, I think I got the most benefit from picking the brains of the experts as, while an experienced test manager, I’m still figuring out how to fold these learnings into my arsenal of tools and techniques. Every opportunity I get to so enhance, I endeavour to take and I’m seeing more and more how i) I could have better approached some of my past programmes of work and more importantly ii) I might do better on future efforts.

It was good to catch up with old friends: David, Richard, Erik, Andrew & certainly not least, Rob and be introduced to new ones; Henrik, Anne-Marie, Rajesh, Tim and many others from the 80-odd attendees. All in all, a very successful day at a very professionally executed event.

A few gems from the proceedings for your amusement:

Richard Robinson:and on the topic’s boards outside we’ll have, um well, topics.....

Henrik Anderssen: ... you might not know that I have a very dark background. Rob Sabourin: Really? *(in case you’re wondering, we do have Henrik’s permission to print this!)*

Rob Sabourin: (when referring to what to do when faced with stakeholders and management who remain unconvinced of our testing explanations) ...if all else fails, let the Wookie win (from Star Wars 1977, for those of us old enough to remember)

Twinkle Twinkle Little Star(West)

By NZTester Staff Writer

Another Star conference rolled around; it seemed only a few weeks since StarEast and StarWest crept up quickly, so quickly in fact that I still hadn't sorted expenses from the last sojourn! There was to be no road trip this time however as StarWest was hosted in Los Angeles, well Anaheim actually, so it was simply fly-in/fly-out.

Like StarEast earlier in the year, StarWest was certainly no "little" affair and topped 1,000 attendees with the usual gallery of "rogues" (me included) on the bill. I reprised my Testing the Data Warehousing tutorial and stepped in to substitute for Wayne Yaddow who was not able to make it this time (and who's book I used a lot in developing my tutorial, thanks Wayne) for a track presentation. In addition, I seemed to get busier this time as not only did I participate in the Lightning Talks but was also interviewed by the lovely Dawn Haynes for the StarWest live broadcast site. It was all capped off with what I was told was a great rendition of "Stray Cat Strut" on the karaoke night (even though it was actually a live band) – just about falling off the stage in the process (do hate it when that happens)!

On the serious side though, I did note that the exhibitions this time were more balanced between tools and services companies however still with most tool vendors focusing on mobile platforms. Others displayed and demonstrated their wares around web-based test automation along with a few long-standers which cover all platforms eg. HP, SmartBear etc. It was pleasing to see Catch Software exhibiting with EnterpriseTester as I think this is the first time, certainly in my recollection, that an Australian or NZ company has been there on the floor at a Star conference outside of Australasia.

I attended most of the sessions I wanted to get to, outside of those that were on at the same time as mine. The one that sticks most in my mind is Jon



Hagar's presentation around the new IEEE 29119 Standard which I understand is to supercede a number of long-term standards including good ole IEEE 829. Watch out for input from Jon on this!

Other sessions that I was impressed by were Matt Heusser's Reducing the Cost of Software Testing, Live Site Quality by eBay's Jon Bach (lots of parallels here with New Zealand's TradeMe, see NZTester 3) and the World Quality Report: Emerging Testing Trends from Sultan Syed (Capgemini), Mary Johnston (Capgemini) and Michael Cooper (Hewlett Packard). The latter was of particular interest to me as some of you will know, this is one of my "pet" subjects. The full report can be downloaded [here](#) and I was surprised to see some of the growth statistics around these emerging trends; some of which could be argued have already emerged and are now prodding at mainstream eg. SaaS, testing in the Cloud.



StarWest also provided the opportunity to catch up with fellow Star alumni in Scott Barber, Rob Sabourin, Paul Holland, James Bach, Dale Perry, Julie Gardiner, Dot Graham, Johanna Rothman and Dawn Haynes...and yes, we are all Star alumni 'cos



So much for one-one!



l.to r. Ken Gardner & Mike Harding (SOASTA)

it said so on our conference badges (even though mine fell off). And I was amused by one particular one-on-one session with James where no less than five attendees clustered around, waiting for the next gem to emerge – so much for one-on-one!

It was also a pleasure to meet Ken Gardner, the founder of SOASTA at their function on the Wednesday night. As per StarEast, SOASTA hosted the evening and it was good to meet up with others who hold an interest in mobile automation testing. Growth in this sector is showing no signs of abating and the available tools seem to be getting cleverer and cleverer. As always though, I find the key to successful test automation often lays in the design of the surrounding framework as opposed to the features of any given tool. Nothing I've seen yet convinces me that this is not the case when automating on mobile platforms even though some tools cover a wider range of operating systems and browsers than others. Plus some are easier to integrate with external frameworks than others so it's all really horse-for-courses.

I resisted the temptation to skip an afternoon and sneak over to my favourite Disneyland rides – not

only did I not want to miss what was going at the conference but I figured that riding Space Mountain with a round of kidney stones (argh!) might not be the most painless proposition. So sorry Mickey, my \$US130 remained firmly in my wallet this time (and paid for four evening meals)!

As always, StarWest was more than ably chaired by Lee Copeland of SQE and in fact I do not recall an SQE conference I've attended that has not been Lee-chaired. His laid back approach puts everyone at ease and able to enter into conference-mode in double quick time and like Disneyland next door, attendees are able to escape the pressures and realities of everyday testing - even if only for a few days!

All-in-all another successful and enjoyable conference; I'm privileged to have been invited back for the next StarEast in May 2014 so looking forward to doing it all again. By then, we should have made some very exciting announcements that I hope to take full advantage of next time so watch this space!

Email me with any queries [here](#).



The three screens of StarWest!



Good to see Catch Software on the StarWest floor

And now it's your turn...

If you would like to be involved with and/or contribute to future OZTester issues, you're formally invited to submit your proposals to me at colinda@ozemail.com.au.

Articles should be a minimum of ½ A4 page at Cambria 11pt font and a maximum of 2 A4 pages for the real enthusiasts. If you wish to use names of people and/or organisations outside of your own, you will need to ensure that you have permission to do so.

Articles may be product reviews, success stories, testing how-to's, conference papers or merely some thought-provoking ideas that you might

wish to put out there. You don't have to be a great writer as we have our own staff writer who is always available to assist.

Please remember to provide your email address which will be published with your article along with any photos you might like to include (a headshot photo of yourself should be provided with each article selected for publishing).

As OZTester is a free magazine, there will be no financial compensation for any submission and the editor reserves the sole right to select what is published and what is not.

Please also be aware that your article will be proof-read and amendments possibly made for readability. And while we all believe in free speech I'm sure, it goes without saying that any defamatory or inflammatory comments directed towards an organisation or individual are not acceptable and will either be deleted from the article or the whole submission rejected for publication.

Feedback

OZTester is open to suggestions of any type, indeed feedback is encouraged. If you feel so inclined to tell us how much you enjoyed (or otherwise) this issue, we will publish both praise and criticism, as long as the latter is constructive. Email Geoff on ed@nztester.co.nz and please advise in your email if you specifically do not want your comments published in the next issue otherwise we will assume that you're OK with this.

Sign Up to Receive OZTester

Finally, if you would like to receive your own copy of OZTester completely free, even though we're real low tech right now. there's still a couple of easy ways: 1) go to www.nztester.co.nz/oztester.html, or 2) simply click [here](#) - Ed.



Click on title

- 13 Discover**
- 21 Software Education**
- 24 Seapine Software**
- 26 Catch Software (Enterprise Tester)**