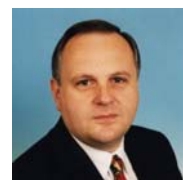


Issue
35**THE
TESTER***December 2010 Issue***Next Conference: Wednesday 8th December 2010****Susan Windsor****Gojko Adzic****Geoff Thompson****Les Hatton****Fran O'Hara****Erkki Pöyhönen****John Kent****'The Keynotes - 6 of the best!'****Conference presentations**

We have a really strong and varied program for the December conference, entitled **'The Keynotes – 6 of the best'**. Every presentation will be of Keynote quality. We hope you will enjoy it.

To open the day **Les Hatton** will ask the question *'Would anybody notice if testers did something else in life?'*. I hope that the answer is yes. I am looking forward to his inimitable and unmissable style.

This is followed by **Gojko Adzic** who is talking about *'Winning Big with Agile Acceptance Testing'*. I have seen this presented as a workshop and I think it is a really valuable real-life experience.

The morning is rounded off by **John Kent** who is going to talk about *'How to Improve Test Team Effectiveness using Test Entities'*. John has been working on Test Entities for several years. You may have seen his EuroSTAR presentation in The Hague.

After lunch **Geoff Thompson** will present a bonus session. The Theme is *'What Influences me in Software Testing'*. Expect some fun!

Erkki Pöyhönen, who has travelled from Finland, will start the afternoon with a talk entitled *'Paradigm shifting without a clutch - What I'd do differently?'* He has taken his inspiration from the world of Dilbert, and reworked this classis lesson for software testing.

This is followed by **Fran O'Hara**, who has travelled from Ireland, to talk to us about *'Scrum – A Tester Perspective'*. I think this will have great relevance and help us understand the changing world of development.

The final speaker of the day is **Susan Windsor**, who is going to show us *'How to Create Good Testers'*. This is more challenging than it sounds and she will help us to first understand what good testers are.

In this issue**CONFERENCE PROGRAM**

- Agenda (p3)
- Abstracts (p4)

HOW TO

- Register to attend the SIGiST conference (p2)
- Borrow a book from the SIGiST library (p11)

ARTICLES

- **Gojko Adzic**
Lowering the barrier to entry for agile testing using patterns (p9)
- **Susan Windsor**
Agile or Waterfall
- Can Project Profiling Help? (p12)
- **Peter Morgan**
Ham, Eggs and SCRUMmy cakes (p15)

TESTING EVENT CALENDAR

- Testing events and conferences (p14)

FROM THE EDITOR

Matt Archer, Editor

Our first article this month comes from **Gojko Adzic** - an active member of the Agile Alliance Functional Test Tool (AA-FTT) programme. In his article, '**Lowering the barrier to entry for agile testing using patterns**', Gojko examines the barriers to adopting agile testing and the work the AA-FTT programme is doing to help remove those barriers. If you enjoy Gokjo's article, I can personally recommend his books ('Test Driven .NET Development with FitNesse' and 'Bridging the Communication Gap'). Both make a though provoking read for anybody working in an agile environment.

Our second article has been written by **Susan Windsor** of Gerrard Consulting. I really like Susan's article as it avoids championing a single process and instead addresses the most important question... what process is right for me? Read Susan's article, '**Agile or Waterfall - Can Project Profiling Help?**' to see how project profiling can help you decide.

In our final article, '**Ham, Eggs and SCRUMmy cakes**', **Peter Morgan** looks at the often overlooked and subtle factors that make SCRUM the success process that it is.

If you have been inspired by any of the articles in this edition and would like to write an article for *The Tester* yourself, then please email me.

Matt Archer

The Tester Editor
BCS Specialist Group in Software Testing
matthewjarcher@googlemail.com

WEB LINKS

BCS SIGiST website:
www.SIGiST.org.uk

SIGiST Standards Working Party:
www.testingstandards.co.uk

SIGiST LinkedIn Page:
<http://www.linkedin.com/groups?mostPopular=&gid=3466623>

LINKEDIN AND TWITTER

The SIGiST is now using social media platforms to improve communications both to members and between members.

Our LinkedIn Group (link below) will carry details of our conferences as they become available. It will also provide a place where people can discuss testing topics, make requests about future conferences, find employment opportunities (there are a few jobs advertised already) and generally keep up to date with our chosen industry. If you are already a member of LinkedIn then simply visit the SIGiST group and make a request to join. If you're not a member then go to <http://www.linkedin.com/> to create an account first.

If you use Twitter you can follow us @SIGiST.

<http://www.linkedin.com/groups?mostPopular=&gid=3466623>

CONFERENCE BOOKING INSTRUCTIONS

If you would like to pay online, you can use our new online booking and payment system.

www.bcs.org/events/registration

If you would like to pay by cheque, you can download a booking form.

www.bcs.org/upload/pdf/sigist-bookingform.pdf

If you have a query relating to making a booking, please contact Gemma Stanley-Evill, Specialist Groups' Officer.

Tel: (01793) 417656

gemma.stanley-evill@hq.bcs.org.uk

CONFERENCE AGENDA

'The keynotes – 6 of the best!'

Wednesday 8 December 2010
Royal College of Obstetricians and Gynaecologists
27 Sussex Place, Regent's Park, London NW1

Time	Session	Length
08:30	Registration open, Coffee and Exhibition Hall	
09:25	<u>Welcome & Introduction</u> Stuart Reid – Chair of the BCS SIGiST	5
09:30	<u>Would anybody notice if testers did something else in life?</u> Les Hatton - Professor of Forensic Software Engineering CISM, Kingston University	60
10:30	<u>Networking Session</u>	15
10:45	Tea / Coffee Break and Exhibition Hall	30
11:15	<u>Winning Big with Agile Acceptance Testing - Lessons Learned from 50 Successful Projects</u> Gojko Adzic - Neuri	45
12:00	<u>How to Improve Test Team Effectiveness using Test Entities</u> John Kent - Simply Testing	45
12:45	Buffet lunch, Exhibition Hall and Networking	75
14:00	<u>What Influences me in Software Testing?</u> Geoff Thompson - Experimentus	15
14:15	<u>Paradigm shifting without a clutch - What I'd do differently?</u> Erkki Pöyhönen- Teito	45
15:00	<u>Scrum - A Testers Perspective</u> Fran O'Hara - Sogeti	45
15:45	Tea / Coffee Break, Exhibition Hall and networking	30
16:15	<u>How to Create Good Testers</u> Susan Windsor – Gerrard Consulting	45
17:00	<u>Closing Remarks</u> Stuart Reid – Chair of the BCS SIGiST	5

ABSTRACTS AND BIOGRAPHIES

Would anybody notice if testers
did something else in life?

Les Hatton, Professor of Forensic Software Engineering CISM, Kingston University

This talk is in two parts. First, the reason for the title is that given the appalling quality of many systems, (as I will illustrate here with copious and mostly recent tales of the front line), it seems fair to say that there simply aren't enough of us to make any significant difference. It is perfectly possible to produce excellent systems but far too often, we as a society simply choose not to. Some of the reasons why will become obvious from the examples given but one of the main ones is that systems design and implementation has been hi-jacked by management consultants and infested by nonsensical, ephemeral jargon at the expense of reality and the end user.

In the second half, I cease ranting for a moment and return to the scientific method. Testers have long used the knowledge that defects appear to cluster but have you ever wondered why this is so? As I shall demonstrate, it turns out to be a beautiful property of a complex system related to power-law behaviour and completely independent of the technology used to build that system, so you can indeed depend on it to improve your testing efficiency.

Finally, is this talk fully compliant with all procedures and end to end mentored with regard for all modalities by an appropriately standardised milieu of consensual sub-committees engaging all stake-holders with a measure of gravitas although pushing the envelope outside the box to address all low-hanging fruit before close of play? I really don't care, I'm a tester, I just break stuff.



Prof. LES HATTON is on his third scientific career. He started in 1973 at the Met Office with a shiny new PhD in computational fluid dynamics, but decided the scientific civil service was not for him and left in 1974 to 'see the world' as an exploration geophysicist based in Houston, Texas and then London. After bouncing around a bit on the world's oceans and visiting various generally unsavoury places, he co-founded a geophysical company in 1979 which was sold along with his soul in 1984. He received the 1987 Conrad Schlumberger Award for his work in computational geophysics. Shortly after, he switched careers again to pursue growing interests in software and systems failure, a nice indoor job with no heavy lifting based in his garden shed, shared with his eldest son's gerbils. He became Professor of Forensic Software Engineering at Kingston University in 2004. He has been voted in the "leading scholars of software systems engineering" by the prestigious US Journal of Systems and Software, and since December 2009, is on the editorial board of IEEE Software. In his spare time he is the guitarist and harp player in the Juniper Hill Blues Band.

Winning Big with Agile Acceptance Testing – Lessons Learned from 50 Successful Projects

Gojko Adzic, Neuri

After an extensive research of agile acceptance testing implementations for his new book, Gojko Adzic presents great war stories that will inspire you to improve your software development process. Learn why and how teams all over the world succeed in bridging the communication gap between business stakeholders and implementation teams and how they got users, developers and testers to collaborate in defining great requirements and acceptance tests to produce software fit for purpose. Gojko will also present a summary of the most important success patterns for different contexts and talk about how to solve most common implementation issues with agile acceptance testing.



*Gojko Adzic got bitten by the agile testing bug five years ago. Since then, he has helped numerous teams implement these practices, written two books on the subject (*Test Driven .NET Development with FitNesse* and *Bridging the Communication Gap*) and contributed to several opensource projects in the agile testing space. At the moment, Gojko is working on his third book, titled *Specification by Example*.*

Gojko is a frequent speaker at leading software development and testing conferences and runs the UK agile testing user group. His company Neuri Ltd helps ambitious teams from web startups to large financial institutions implement specification by example and agile testing practices.

How to Improve Test Team Effectiveness using Test Entities

John Kent, Simply Testing

Do you spend too much time writing detailed test scripts and too little time increasing test coverage by analysing the system design? Do you know the most efficient way to build tests? Do you consider how much detail should be written into the steps in a test script? Do you know when you should build test scripts and when it would be better to stick with test cases or even just test conditions? Do you know the reasons for writing test scripts? This presentation will offer answers to these questions using an entity model of software testing as an oracle.

The most scarce commodity in software testing is time. It makes sense then, to be able to write and execute tests in the most efficient way possible. Very often however, no consideration is given to the way we build tests; rather testers just dive in and write scripts. The entity model of software testing has been developed to improve tester productivity and make testing more efficient. It attempts to fully define the test assets we create and the relationships between them. It maps out the entities we use from requirement specifications, test conditions and test cases to test results. An interesting theory with few practical uses? Not so, because the entity model demonstrates the relative benefits of different approaches to building and running tests and offers ways to improve. The presentation will enable testers to improve their testing by giving them a route map which will show the way to more efficient testing.



*John Kent, a leading consultant, specialises in test automation and test management. He developed the Liberation automated test framework and was involved in the development of the T-Plan test management tool. He wrote the **The View From Kent** column in *Professional Tester Magazine* and is co-author of the **Official Netscape Guide to JavaScript1.2** (Netscape Press). John regularly presents to international audiences on software testing subjects including his own automation training course.*

What Influences me in Software Testing?

Geoff Thompson, Experimentus

Geoff has enjoyed working within the software testing discipline for 20 years now and in this short bonus session he will encapsulate why he started down that road and share what keeps him motivated today.



Geoff has a real passion for the right software testing. He is a founder member of the International Software Testing Qualification Board (ISTQB), is currently the Chairman of The UK Testing Board. He co-authored the BCS book *Software Testing - An ISEB foundation*. In 2008 Geoff won the European Testing Excellence Award. He is a popular speaker having recently delivered keynote presentations in India and Australia as well as being a regular speaker at EuroSTAR and the BCS SIGiST.

Paradigm shifting without a clutch – What I'd do differently?

Erkki Pöyhönen, Teito

The Testing field is rich. So rich that normally we do not believe how rich it can be -- rich with different groups of people with varied needs, rich with development challenges and rich with solutions. The people with different background and different paradigms seldom communicate well. That means also different views about generic concepts, like what is considered a reasonable defect report, a good test case or even feasible testing!

I've had the pleasure to experience testing on several, very varied contexts; first product development in a small company, then large-scale development in formal ways, later IT, back to R&D, doing training, consulting, waterfall, Agile. All new situations bring fresh challenges but also a new and widening view to testing. Whenever I asked for advice from a consultant I got the stock answer "it depends", at some point I learned why? and what else to say.

In this talk you'll hear about the changes I had to adjust to, what I learned, how my testing thinking has changed along the years -- including what were my worst mistakes and also my best lessons in testing.



Erkki works as software testing consultant and test manager at the service company Tieto Oyj, having 1300 testers globally. Before testing services he has worked as a programmer, tester, line manager, management consultant, trainer and webmaster. As a keen facilitator of professional networks he has been a founding member of ISTQB, FiSTB (Finnish Software Testing Board) and FAST (Finnish Association of Software Testing) TTL (Finnish Information Processing Association, similar to BCS but smaller); while bootstrapping the FAST on 2001 he had SIGiST as a model and benchmark for success. He's a frequent speaker on testing events and promoter for testing profession in Finland. Professionally most memorable achievements so far include being voted by his peers as "Tester of the Year" in Finland on 2008 and programme chair role of EuroSTAR2004. He's proud father of 4 young adults, married, and enjoys photographing Finnish nature and singing in a choir on any available time.

Fran O'Hara, Sogeti

Of all the agile methods, Scrum seems to be spreading the most. However, Scrum does not say much about testers and testing and, in many cases, organisations using agile struggle to implement effective approaches to testing and achieving the productivity enhancements with the required level of quality. This is because of issues such as:

- Implementing SCRUM as a effective work management approach but without designing/adopting appropriate development and test practices within the incremental framework
- Partial implementation of agile methods and agile mindset sometimes resulting in negative quality implications
- No clear Scrum test strategy

This presentation will:

- Explain the essence of Scrum, its roles, practices and the mindset changes required
- Define how the tester fits into the Scrum team
- Present the key learning points so far in Scrum from a test perspective including testing without detailed requirements, testing strategies in incremental/iterative environments and how to integrate agile practices such Test Driven Development (TDD) and exploratory testing in Scrum



Fran O'Hara is practice manager/principal consultant with Sogeti Ireland. Sogeti now incorporates Insight Test Services which Fran co-founded in 2003 to provide test consulting, training and managed test services. Fran is also a director of Inspire Quality Services. He specialises in pragmatic approaches to process improvement, agile and associated best practices. Fran is a regular speaker at process improvement and testing conferences. He is an ISEB/ISTQB tutor, a trained SEI CMM lead assessor, a certified ScrumMaster and TickIT auditor, a fellow of the Irish Computer Society and co-founder of the Irish SIG in Software Testing - SoftTest.

Susan Windsor, Gerrard Consulting

During this presentation I will cover three key questions:

1. What is a good tester?
2. Why would you want to be one?
3. How can you create or become one?

Firstly, I don't believe there is an industry wide definition of "good tester" that fits every situation; but I do believe good testers exist, and that they are very special and valuable individuals! So, I'll share what I believe constitutes a good tester in some different situations and you can identify how this relates to your world. I'll also give my opinion on how this relates to our obsession for certification.

As for wanting to be a good tester, I will ask why you wouldn't! It's our work that provides the most valuable management information on the planet; gives confidence to stakeholders that their system is going to bring the benefits they want; and the skills we need cover the entire project life-cycle. If only we could unlock our minds from the tethers others try to constrain us with.

Finally, I will share my experiences of how to create good testers. For each of the situations I defined earlier, I'll expand on how your organisation can identify people to target for a career move into testing. Maybe you'll identify how to improve your own career prospects too!



Susan has recently merged her business with Paul Gerrard and is now a Principal with Gerrard Consulting, responsible for the provision and delivery of testing consultancy. Prior to running her own company (for 5 years), Susan managed the Testing Service's for IBM. Overall, she has 35 years in IT, the last 15 years dedicated to testing. Susan has spoken at many industry conferences in the past, including EuroSTAR, SQSTest, Softest in Ireland, Unicom, ExpoQA in Madrid and at the BCS SIG.

LOWERING THE BARRIER TO ENTRY FOR AGILE TESTING USING PATTERNS

Gojko Adzic, Neuri

Agile acceptance testing, acceptance test driven development, behaviour driven development and specification by example are great ways to build quality into products in short iterations or flow-style development. I'd go as far as saying that they are an essential part of a successful testing strategy with Scrum, Extreme Programming, Kanban and related processes. Although the basic ideas behind all these processes have been around since the eighties, the adoption in the community has been less than stellar. Editors of InfoQ, one of the leading agile media sites, suggested that the whole thing is just theoreticall. [1] A new effort by the Agile Alliance Functional Testing Tools (AA-FTT) programme promises to lower the barrier to entry and help teams get started with all these ideas easier.

While researching for my upcoming book Specification by Example, I collected more than 50 case studies of how successful teams approached testing and specifications in short iterations or flow-style development. Once I tried to put that into a consistent story, the level of confusion caused by inconsistent naming suddenly became apparent. Many teams had very similar elements in their processes, but used different names for them. All the successful teams used a set of examples as a target for development, acceptance criteria when it is done and regression check later. Some teams called this set of examples an "acceptance test", some called it "feature specification", and some focused on a tool or a mechanism to capture the examples, such as "feature file" or "fitnesse page". Some names, such as "continuous integration" create a confusion because in this context they do not apply to integration tests but to functional checks. Some very important ideas, such as using a set of examples to aid change impact analysis and support, do not really have a name at all yet.

My angry blog post about this[2] caused a lengthy discussion on the AA-FTT mailing list.[3] This resulted in an effort by the AA-FTT programme to finally define a set of process patterns and a consistent language for the practices associated with agile acceptance testing, acceptance test driven development, behaviour driven development and specification by example.

AA-FTT to the rescue

AA-FTT[4] is a programme of the Agile Alliance, created in 2007 to build awareness and community around functional testing tools applied on agile projects. Although the programme name contains the word tools, the focus of the people involved in it has shifted towards processes and practices. Process consolidation was the key topic at the workshop in Orlando during Agile 2010. Following the discussion on the AA-FTT mailing list after the workshop, we decided to attempt to consolidate the practices and the language around the processes.

Patterns in software development were very popular in early nineties,[5] and although the interest in them has waned over the last decade they are a great way to organise process ideas and practices, so we decided to try out patterns workshops.

Jennitta Andrea, Elisabeth Hendrickson and I organised an initial patterns writing workshop in London in early October,[6] closely followed up by another workshop in Berlin during Agile Testing Days. We invited industry experts coming from different backgrounds, contexts and working with different tools to join us and kick start the process. Linda Rising, with more than 15 years of experience in patterns workshops, luckily agreed to facilitate the initial two workshops and help us get started.

The initial pattern ideas workshop produced around 100 interrelated themes,[7] on everything from project management over collaboration to test automation. This struck me as overly complex considering that the Gang of Four book, probably the best pattern language example in software, captured the key ideas of object oriented design in only two dozen patterns. Linda Rising suggested that we will see several layers of patterns evolving, with higher lever patterns capturing a whole range of ideas.

We then ran writers workshops in London and Berlin to start nailing down what these patterns actually mean. Working in groups of six, we reviewed suggested patterns and proposed refinements. This demonstrated how the same ideas can be

implemented completely differently and that “best practices” really do not exist. We often got caught up in discussions on the limits of applicability of particular ideas which helped us better define generalised patterns and the contexts in which they apply.

Conclusions

The first thing that really became obvious to is that the community has to stop thinking about the tools and step back to look at a wider picture of the processes that have evolved over the last decade. Tools are there to assist us with a process, so focusing on tools and not on the process is just causing problems in adoption for new teams.

From the discussions during the workshops and my research, it is also obvious that this thing, whatever it ends up being called, is much bigger and more important than just functional testing. It involves collaboration, specification and long term product change management. I guess this will lead to renaming the programme from Functional Testing Tools to something more process oriented.

Inconsistent and illogical nomenclature only creates confusion and a barrier to entry. As the patterns workshops help us refine the process patterns and create the pattern language, I expect this barrier to entry to be significantly lowered. Patterns should help teams at least take the first step easier and overcome the initial problems, which are in my experience often caused by a gross misunderstanding.

Next steps

Follow-up workshops have already been scheduled in the US and there are talks of workshops in Germany and Finland. I plan to run at least one writers workshop next year in London, so if you are interested in attending get in touch by e-mail to gojko@neuri.co.uk,

We all agreed that the content will be published under the creative commons license, so I expect the initial set of patterns to show up soon on a public collaborative editing platform. At the time when I wrote this, we still did not have a specific URL for it but monitor the AA-FTT mailing list[3] to get notified when the web site comes up.

[1] <http://www.infoq.com/news/2009/06/automated-acceptance-tests>

[2] <http://gojko.net/2010/08/04/lets-change-the-tune>

[3] <http://tech.groups.yahoo.com/group/aa-fft/>

[4] <http://www.agilealliance.org/programs/agile-alliance-functional-test-tool-program-aafft/>

[5] <http://c2.com/cgi/wiki?HistoryOfPatterns>

[6] <http://aafft-2010patternworkshop.crowdvine.com>

[7] <http://specificationbyexample.com/patterns.php>



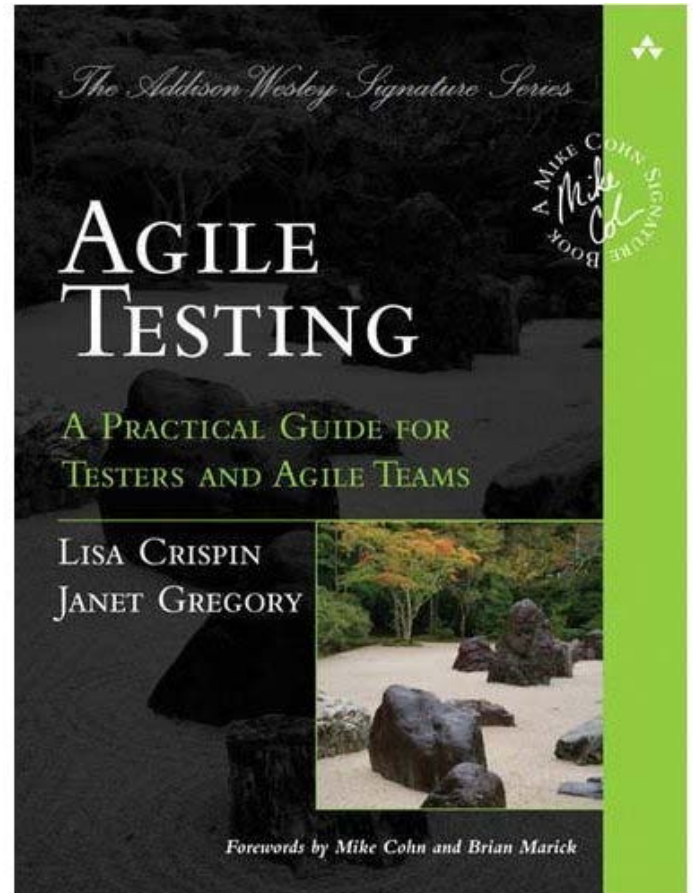
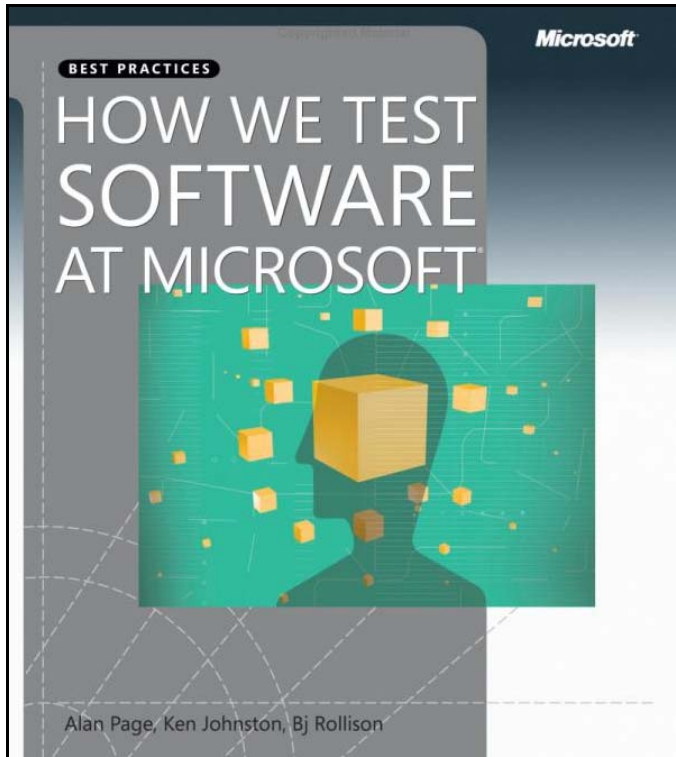
Gojko Adzic got bitten by the agile testing bug five years ago. Since then, he has helped numerous teams implement these practices, written two books on the subject (*Test Driven .NET Development with FitNesse* and *Bridging the Communication Gap*) and contributed to several opensource projects in the agile testing space. At the moment, Gojko is working on his third book, titled *Specification by Example*.

Gojko is a frequent speaker at leading software development and testing conferences and runs the UK agile testing user group. His company Neuri Ltd helps ambitious teams from web startups to large financial institutions implement specification by example and agile testing practices.

BORROWING A LIBRARY BOOK

Looking for a testing book but not sure which topics are covered? Or are you trying to decide which testing book to buy? Or do you simply want to increase your testing knowledge? If the answer to any of these questions is 'yes' then the SIGiST Library could help!

The SIGiST Library has lots of testing books covering a variety of topics and they are available to borrow for a period of 4 weeks - free of charge. Extended loans are allowed as long as the book has not been requested by another SIGiST member.



Topics include (amongst others) Requirements testing, Reviews/Inspections, Test Management, Test Techniques and Test Process Improvement.

If you would like to know more about the library and books available, or for any queries, visit...

<http://www.bcs.org/server.php?show=ConWebDoc.11675>

AGILE OR WATERFALL - CAN PROJECT PROFILING HELP?

Susan Windsor, Gerrard Consulting

In IT, we all strive to meet the challenge of improving the success of system delivery. The priority areas for me are:

- Delivering what is actually required by the business
- Reducing the cost of re-work
- Increasing confidence in timelines

After so many years it's surprising, and perhaps disappointing, that we're still struggling. However there has been a great deal of effort gone into meeting these challenges.

As testers, we've set up independent test functions; strived to create a profession; become specialists in different disciplines; and introduced new techniques such as risk based testing.

The development approach itself has migrated from formal, highly structured waterfall, through to iterative prototyping and more recently to agile development. All have their strengths and weaknesses in different situations and yet so many organisations appear to be making the decision to switch to agile as a knee jerk reaction, without understanding what it actually means. Some, having switched to agile are reverting to waterfall because agile didn't work for them – hardly surprising is it!

Then there are projects that can't adopt all of the agile principles (such as test driven development and continuous user involvement) but still want to benefit from some of the activities such as defining requirements by examples (stories and scenarios), having short iterations of development, and so on. The advocates of each development approach become very competitive, vying for our attention and assuring us that if we don't do it "their perfect way" we're somehow failures.

In reality, we're all after the same thing and different projects have different needs. Some lend themselves more to agile, some to waterfall and some to somewhere in between – iterative development perhaps? How can this decision be taken, how does that impact the testing approach; and what can we do as testers to contribute to the decision making process?

In a testing improvement project we worked on earlier this year, something our client found to be very helpful was a project profiler and associated test framework. It's a simple concept that was both pragmatic and flexible – and therefore successful. Designed as a tool for project and test managers it helps to identify levels of formality and risk. As testers yourselves, I suggest you look into doing something similar within your organisation. It helps to ensure you've involved at the start of the project (which we always want to achieve) by having something valuable to support the project manager and sponsors.

The first thing we did was remove all the development process "labels" so we didn't talk waterfall and agile; that removed the emotion from the process! We then developed a framework of "Project Aspects" and defined a set of model answers for each of them. You can then quickly circle the model answers that represent the closest fit to your project. Inevitably, all the model answers won't fit neatly into a single column but you're very likely to get a bias towards one of them. In addition, the thought process you go through is very valuable. Let's assume you've got a bias towards the column that represents agile development, you can take the test framework associated with agile and use that to develop a specific test approach for your project. Then, for Project Aspects that don't fit into the agile framework, seek additional activities from the relevant test framework to build into your project test approach.

Our project profiler and test frameworks are too large to fit into this article; even if they did it would make sense for you to develop your own so that it's specific to your organisation; your decision making process; your skills;

and so on. However, there are some examples provided below of the Project Aspects and Generic Risks with their respective model answers. Can you identify which is the agile column? Let me give you a clue; it isn't column D, that represents having insufficient information to make a decision!

Project Aspects	A	B	C	D
Sources of Knowledge	New system replaces a well understood existing system; users have a clear vision of system goals and prefer to document their requirements up front	Users want to collaborate to jointly define requirements and meet them incrementally	Users put the onus of requirements elicitation on the project; requirements and the solution will evolve	Inexperienced users who are unable or unwilling to collaborate with requirements gathering
Requirements Stability	New system is a functional replacement of an existing system or a well-defined process (requirements can be fixed early on)	New system replaces an existing system with enhancements or an established (but not necessarily documented) process	New system supports a new business need; business process exists but will change/evolve; users have experience of requirements	New system supports a new business need; business process is not yet known; users have no experience or requirements
Responsibility for Acceptance	Users will take responsibility for UAT and have UAT experience	Users will be responsible for UAT but have no test experience	Users will take part in UAT or witness tests at critical periods, and will review the outcome	Users are unwilling/unable to take part in UAT; reluctant to make the acceptance decision or not known
External Dependencies	More than one or new external suppliers responsible for development; and supplier testing	Single, known supplier responsible for development and supplier testing	In-house development, no external dependencies	Dependencies on external suppliers, their responsibilities or competence not yet known
Visibility, Formality	High visibility/risk to general public; formal progress reporting required at board level; fixed scope and deliverables; formal approvals and sign-offs	High visibility/risk to business; formal progress reporting required; some defined deliverables, some deliverables will emerge/evolve; some approvals and sign-offs	Relatively low business-risk; informal progress reporting is acceptable; partial solution may suffice, incremental/iterative delivery	Potentially high visibility, high risk project; uncertain impact on the business
Generic Risks	High Exposure	Medium Exposure	Low Exposure	Unknown
Business Engagement	Available ad-hoc only; nothing significant	Available part time (1-2 days a week) on the project	Dedicated full time on the project	Not known
Project Complexity	Complex and/or critical business processes; conflicting goals/requirements	Stakeholders agreed on goals and architectural vision	Clear well-understood goals, low architectural and system complexity	Business goals and system complexity are not well defined or understood
Risk of System Failure	Failure would cause major/widespread disruption to the business or public embarrassment	Failure would cause widespread/minor or local/major temporary business disruption	Failure would cause temporary, minor, local disruption to the business	Impact of failure is not known

To complete the profiler, include all project aspects; generic risks and product risks that are relevant to your organisation. If you're using risk based testing, you will already have relevant product risks. I strongly recommend that you try to construct a relevant profiler for yourselves. The real value I found was not only in being able to identify the most appropriate development method and test approach, but in the thought processes required to complete the profiler and how much the outputs helped have conversations with others about the rationale behind the thinking.

Finally, don't get hung up on labels and trends. Construct a test approach that is valid for each project you're involved in and use "examples" where ever possible to aid communication. Good luck! I'll be really interested to hear how you get on.



Susan has recently merged her business with Paul Gerrard and is now a Principal with Gerrard Consulting, responsible for the provision and delivery of testing consultancy. Prior to running her own company (for 5 years), Susan managed the Testing Service's for IBM. Overall, she has 35 years in IT, the last 15 years dedicated to testing. Susan has spoken at many industry conferences in the past, including EuroSTAR, SQSTest, Softest in Ireland, Unicom, ExpoQA in Madrid and at the BCS SIG.

TESTING EVENTS CALENDAR 2010 / 2011



expo:QA
<http://www.expoqa.com/>
15 - 18 November 2010
Madrid, Spain



EuroSTAR
<http://www.eurostarconferences.com/>
29 November - 02 December 2010
Copenhagen, Denmark



TestExpo
<http://www.testexpo.co.uk/>
7 December 2010
London, UK



SIGiST
<http://www.bcs.org/server.php?show=nav.9264>
8 December 2010
London, UK



UK Test Management Annual Forum
<http://uktmf.com/>
26 January 2011
London, UK



SIGiST
<http://www.bcs.org/server.php?show=nav.9264>
16 March 2011
London, UK



IEEE International Conference on Software Testing, Verification and Validation
<http://sites.google.com/site/icst2011/>
21 - 25 March 2011
Berlin, Germany



TAICPART
<http://www.taicpart.org/>
25 March 2011



STARWEST
<http://www.sqe.com/stareast/>
1 - 6 May 2011
Orlando, US



BCS Scottish Testing Group
<http://www.bcs.org/server.php?show=nav.9729/>
Spring / Autumn
Edinburgh or Glasgow, UK



ignite UK
<http://www.ignite-conferences.com/uk/index.aspx>

(ignite worldwide <http://www.ignite-conferences.com/>)

HAM, EGGS AND SCRUMMY CAKES

Peter Morgan, Software Tester

There are two types of people that can attend our daily development SCRUM meetings: pigs and chickens. Why these are called 'pigs' and 'chickens' is a well-known story, but in case anyone has not heard it, here it is, courtesy of Wikipedia:

A pig and a chicken are walking down a road. The Chicken looks at the pig and says "Hey, why don't we open a restaurant?" The pig looks back at the chicken and says "Good idea, what do you want to call it?" The chicken thinks about it and says "Why don't we call it 'Ham and Eggs'?" "I don't think so" says the pig, "I'd be committed but you'd only be involved."

So, 'pigs' have a voice at the SCRUM because they are committed, but 'chickens' are only involved, so for the most part stay silent. On our project, there was a mixture of Agile and traditional approaches; I was the tester in the only Agile team, a team of three, geographically removed from the rest of the project. For that reason, the daily stand-up SCRUM was actually a telephone conference call. These two matters, (Agile and traditional [almost Waterfall] within the same project and telephone conference call SCRUM meetings) presented some interesting challenges for the whole team and particularly the 3-man Agile cell, but that is not the purpose of this article. As the tester, I was very much the link between our development team and the rest of the project, and towards the proposed implementation date, spent 2 or 3 days a week with the majority of the team, in Central London. Some of the overall project team in London were regular or occasional participants in the daily SCRUM, as 'chickens'.

The SCRUM meeting is actually a very good discipline, with its three focus points for each 'pig' in the 15 minute session:

- 1) What did you do yesterday?
- 2) What are your targets for today?
- 3) Any blockers?

It holds everyone to account, and disseminates information, so that all (including the 'chickens') know what is going on, and where we expect to be at the same time the next working day. However, we have a particular rule that also applies to SCRUM meetings, whether these are true stand up sessions, or like ours, a telephone conference call. Anyone who is late buys cakes!

If it is truly a stand up session, provided that some of your SCRUM colleagues are in close proximity, or the meeting area is within view, making the SCRUM meeting on time is not normally a problem. However, conference calls are different, and one that has cost me dearly, both through the pocket and around the waist. There I am, grappling with the latest release, seeing if my 'observations' have been addressed, and what effect this release has on the embryonic User Guide when bingo! My development colleagues sitting no more than 2 metres away have picked up their phones, connected and are awaiting my participation, already thinking "cakes".

Excuses for non attendance are permitted, provided that these are given at the previous SCRUM, or are allowable (my 4 nights' unexpected stay in hospital was fortunately an acceptable excuse). However, the vagrancies of London Transport are not valid. "You're late, and we want our cakes!" Recently I had a real problem when I had two consecutive days in London. On the first, there was a problem with the Circle Line, so I was 6 minutes late. My fault, I should have left enough time to walk the whole way (Oh, he is a hard SCRUM master!), so cakes it would be. The SECOND day, the tube was slightly delayed, so I was faced with buying cakes on the way to the office, and being late again (meaning cakes the day after as well), or turning up with no cakes. I chose the latter, and asked those on the conference call whether I should buy cakes for the London participants or the West Country team, with one of the London 'chickens' to decide. Problem solved, cakes were bought in London and available that afternoon.

Nobody (neither 'pigs' or 'chickens') likes to buy cakes for being late. It is not the cost, or even the inconvenience (£6.00 for a very good selection, and 6 minutes maximum). No, it is the ignominy of having been found out. The starting time constraint of the SCRUM meeting is an example of the positive power of peer pressure. Avoiding the disapproval of one's colleagues can be a tremendously motivating force – try turning up at a document review meeting without having even opened the document to see this disapproval with all its brutal force. Avoiding disapproval can be the factor that keeps the Agile development on track, that maintains the velocity of the development or even increases it. The thought of explaining to colleagues why you have not done that which you contracted to do (at the previous day's SCRUM) is a big motivator. It is almost as much a motivator as the idea of (not) buying cakes. Almost, but not quite!



***Peter Morgan** passed the first ISEB Practitioner Certificate software testing exam, in 2002. He worked for a testing consultancy and wrote their (old-style) 'Practitioner' course, teaching the Foundation level. A member of the ISEB accreditation panel, Peter is a hands-on tester, working on a free-lance basis within the UK. He has presented at EuroSTAR conferences, attending and occasionally speaking at the London SIGiST.*

