

testing experience

The Magazine for Professional Testers

printed in Germany

print version 8,00 €

free digital version

www.testingexperience.com

ISSN 1866-5705

© iStockphoto.com/vgajic



Play the game, test the Game!



Certified Agile Tester

Pragmatic, Soft Skills Focused, Industry Supported

CAT is no ordinary certification, but a professional journey into the world of Agile. As with any voyage you have to take the first step. You may have some experience with Agile from your current or previous employment or you may be venturing out into the unknown. Either way CAT has been specifically designed to partner and guide you through all aspects of your tour.

The focus of the course is to look at how you the tester can make a valuable contribution to these activities even if they are not currently your core abilities. This course assumes that you already know how to be a tester, understand the fundamental testing techniques and testing practices, leading you to transition into an Agile team.

The certification does not simply promote absorption of the theory through academic mediums but encourages you to experiment, in the safe environment of the classroom, through the extensive discussion forums and daily practicals. Over 50% of the initial course is based around practical application of the techniques and methods that you learn, focused on building the skills you already have as a tester. This then prepares you, on returning to your employer, to be Agile.

The transition into a Professional Agile Tester team member culminates with on the job assessments, demonstrated abilities in Agile expertise through such forums as presentations at conferences or Special Interest groups and interviews.

Did this CATch your eye? If so, please contact us for more details!

Book your training with Díaz & Hilterscheid!

Open seminars:

September 26 - 30, 2011 in Mödling, Austria

October 24 - 28, 2012 in Cologne, Germany *

November 07 - 11, 2011 in Mödling, Austria

November 07 - 11, 2011 in Stockholm, Sweden

December 5 - 9, 2011 in Berlin, Germany

February 13 - 17 , 2012 in München, Germany

March 19 - 23, 2012 in Berlin, Germany

April 23 - 27, 2012 in Berlin, Germany

June 11 - 15, 2012 in Cologne, Germany

* (German tutor and German exam)



Díaz Hilterscheid

Díaz & Hilterscheid GmbH / Kurfürstendamm 179 / 10707 Berlin / Germany

Tel: +49 30 747628-0 / Fax: +49 30 747628-99

www.diazhilterscheid.de training@diazhilterscheid.de



Dear readers,

It was a great summer. I really did enjoy it. I was in France visiting my kids, and I was also with them in Gran Canaria, Spain, spending a great holiday at the beach. Only 3 weeks...

My daughter lost her fear of the waves and started to snorkel. She swam after the fish and spent a great time in Las Canteras Beach. A great place for kids and families.

If you've never spent a holiday in Las Palmas, try it once. You will love it!

After the holidays my six year old daughter started school. She had wanted to start for months; she felt she was too "old" for the Kindergarten! After the first day at the school she said to me in a quite complaining and disappointed way: "I've not learned anything today. We just paint things. I did not have maths." I hope that she keeps the will to learn for the next 100 years! ;-)

Back in the office, I have to face – like most of you – a big mountain of work in front of me. I have the feeling that we work more towards the end of the year than at the beginning. I don't know if there is a psychological aspect to this, maybe we just want to end the year... and that's why we work and work and work.

The Agile Testing Days are now coming up. It looks like we will break all the records of previous years. It is amazing to see how many people register and like the conference programme. If you haven't seen it, have a look on www.agiletestingdays.com.

Our team is doing a great job! Have you seen the videos? Don't miss it. The planking one is amazing! In it you will see some company colleagues.

We've decided to move the Testing & Finance from the "finance city" Frankfurt to the "finance city" London! It was the right decision. It was also the right decision to do it with Paul Gerrard and Susan Windsor. It looks like we will rock the City! The call for papers is still open. If you have a good topic, please go ahead and send us the proposal at <http://www.testingfinance.com/>

The program of the Belgium Testing Days is out. Have a look at it. It is really great. Do not miss it (visit www.belgiumtestingdays.com).

Last but not least, I would like to thank all the authors for their articles and columns and all the sponsors for supporting once again the success of testing experience.

I wish you a nice reading and hope that you can find something that helps you carry away that big mountain in front of you.



José Díaz
Editor

A handwritten signature in blue ink that reads "José Díaz". Below the signature, the name "José Díaz" is printed in a smaller, clean font, followed by the word "Editor" underneath it.

Contents

Editorial.....	1
EXPLORE: TESTING THE GAME..... by KiranKumar Marri & Sundaresasubramanian G	4
Semi-formal property verification in games..... by P. Arun Babu & N. Murali	14
Gamification of testing..... by Wayne Ellis	18
Turning into AQA organization..... by Andrei Contan & Cristina Fuia, & Oana Oros	22
Quality, what does it mean?..... by Erik van Veenendaal	27
Test by recreating and running customer scenarios in the lab..... by Haya Rubinstein	30
How to start cutting-edge testing with your sword	34
by Guilherme Motta	
Game On!..... by IJsbrand Kaper MBA	38
“We unite competences”.....	42
Making the move into Agile testing..... by Gerry Coleman	50
Test management issues for systems of systems..... by Patrick Hendrickx & Chris Van Bael	54
Las Vegas game testing..... by Thomas Hiji	58
Product testing in front of the crowd	60
by Derek Wallis	
Masthead.....	65
Picture Credits.....	65
Index of Advertisers.....	65

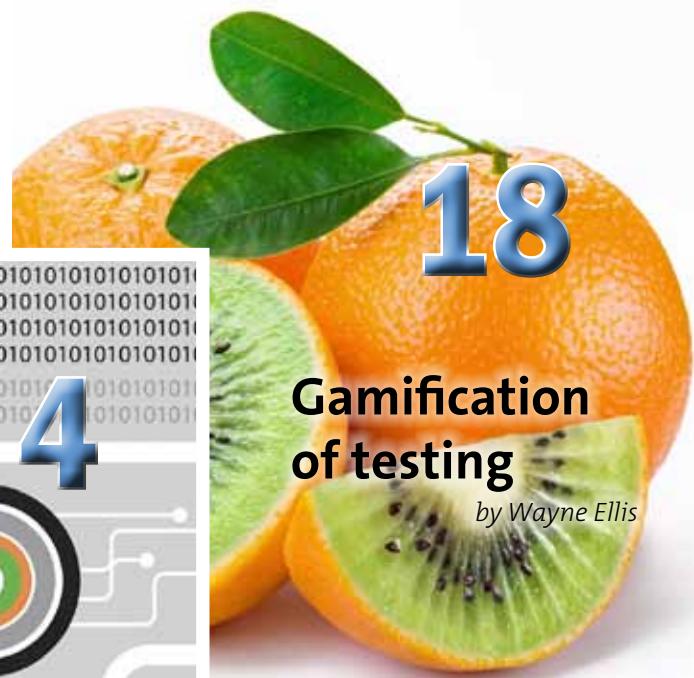
**Test by recreating and running
customer scenarios in the lab**

30

by Haya Rubinstein

**EXPLORE: TESTING
THE GAME**

by KiranKumar Marri & Sun-
daresasubramanian G



18

**Gamification
of testing**

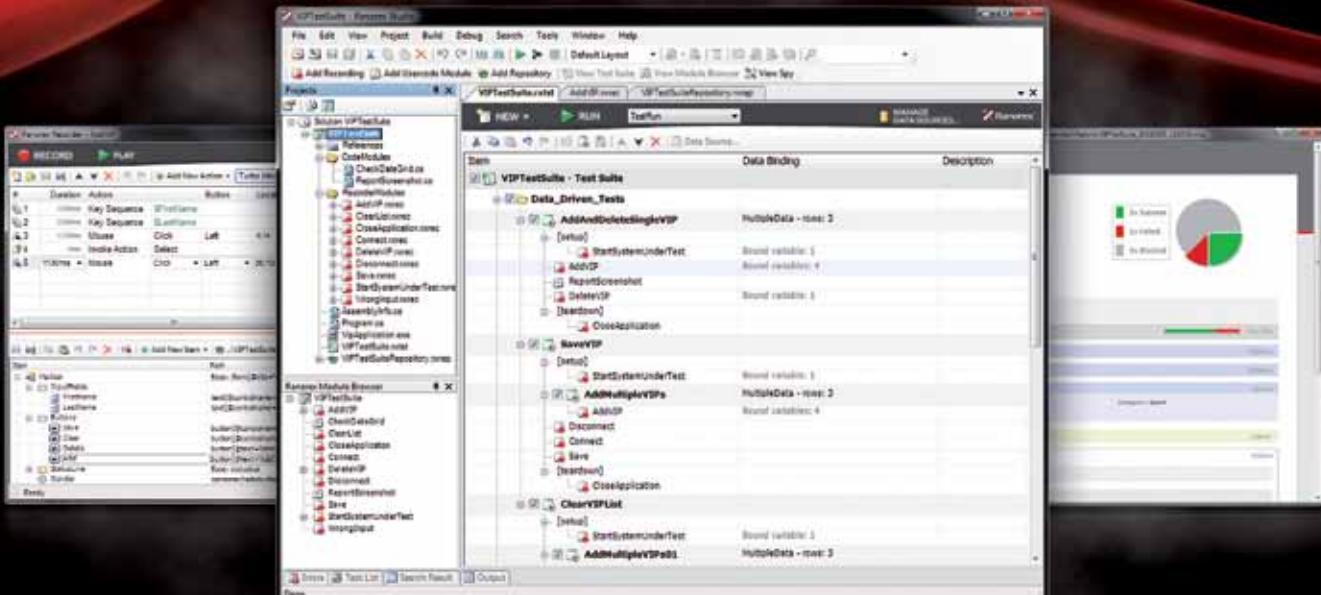
by Wayne Ellis

4

Automate your User Interface Testing

“ Ranorex is the *Best Commercial Functional Automated Test Tool* for .NET and Flash/Flex applications.

— 2010 ATI Automation Honors Awards ”



Record and Edit Reliable Test Actions



Manage and Execute Your Automated Tests



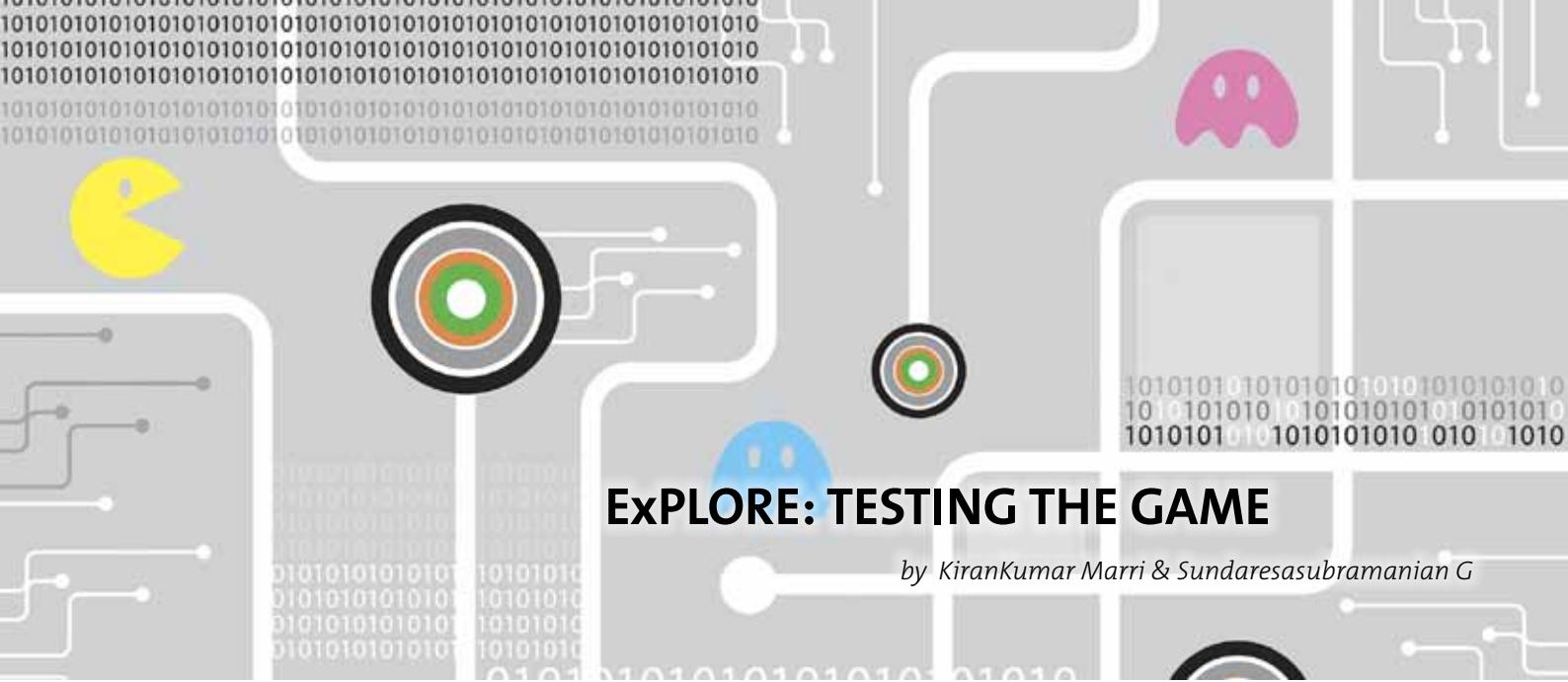
Reproduce Bugs and Maintain Your Tests

- ✓ Use connectors for data-driven tests
- ✓ Build robust test automation frameworks
- ✓ Generate EXEs for pure flexibility
- ✓ Write custom code in C# or VB.NET

Award-winning test automation tools, which allow testing of many different application types, including: Web 2.0, WPF, Flash/Flex, Silverlight, Qt, .NET, 3rd Party Controls, and Java.



Download your 30-day Trial at www.ranorex.com



EXPLORE: TESTING THE GAME

by KiranKumar Marri & Sundaresasubramanian G

It has been widely reported on internet that the “Angry Birds” game has hit the 250 million download mark and the “Pacman” doodle created by Google to celebrate its 30th anniversary has cost a few million hours in productivity in offices worldwide. There are few games on one end of the gaming spectrum which we have grown up by playing, and there are few on the other side, which are simply irresistible and enjoyable. All the other games in between compete with each other for their share in the gaming market. They target a specific user group, render good graphics/sound and apply excellent marketing strategies along with an interesting theme of play.

With the advent of internet, network gaming has become the new world order. While everyone thought of it as the pinnacle of gaming, user centric gaming came into the scenario (such as X-Box, Wii), where the player has a physically active role in the game instead of controlling a few keys. The days are not far off when the fantasies created by movies (player inside the game, player controlled by game, virtual gaming world) will become a reality.

The critical success factor of a game is its underlying development and testing. The expectations of a gamer have increased substantially so that a simple error can spell doom to the future of the game concerned. Also, the tester is really required to live the game from its user's perspective to understand and test it well. We will look at some of the nuances of game testing through this article as we explore them with practical scenarios.

Functionality is the key!!

Functionality like any other software testing is the very basic building block of game testing. It checks the basic working of the game, i.e., the controls (mouse/keyboard if it's a PC game, joystick if it's a controller based game, remote or sensor for user-centric games). Most of the functionality components (except for brand-new games) are taken off the shelf, and so there is a scope of automation for functionality/regression testing as well.

Functionality testing would mostly follow the development pattern. In some game development environments, the software would be developed in a simulator without the real hardware or controller available. After being thoroughly tested, the software is brought out of simulation and tested in a combination of software and real hardware.

This testing requires the most amount of technical knowledge because it involves monitoring controller/network messages and identifying errors, setting up test beds, debugging them, testing the sequence of triggers, etc.

Go with the flow!

Any software game would follow a flow or a pattern or sets of sequences. Testing the flow is of two types: understanding the documented flow and verifying it step by step, or go with the flow based on the tester's intuition. The advantage with the first method of testing is that it helps in thorough verification of the documented flow (happy path). However, the same point is its disadvantage too; the documented path is limited to a set of actions performed by the user and might not take into account various branches of flow or pattern identified by a gamer in real-world.

The second method of testing, which is intuitive is basically not dependent on documentation and tests the game's reaction according to the tester's flow and exploratory concepts. The tester is assumed to be a mid/high level gaming expert in this scenario, and the experience of the tester is really helpful in determining hidden paths (or) cheat codes in a particular flow. However, this type of testing is not exactly repeatable, since the knowledge resides with an individual and not in a document.

The best method is hybrid flow verification – a combination of document based flow testing and intuition based testing. The tester should document the various possible flows occurring in the game and test them thoroughly. In addition to this, an experienced gamer/tester should test the flow or pattern, and all of their inputs should be documented to build future test scenarios.

Some of the basic checks that would be performed in a flow/pattern testing would be to check all levels of the game (including hidden/bonus levels), saving and retrieving a level, soft and hard power reset to test the game's restorability.

Is the content appropriate?

Content varies from game to game depending on the level of its users. Let us take a scenario where the tester is testing a game called “Blaster-Man” by “XYZ media”, which is a novice adventure game involving the use of knives, guns and bombs. The protagonist of the game has to clear various levels after crossing multiple

obstacles on the way. This game targets the age group of 5-11. As part of content verification, the tester should verify that the game is not too graphic or detail oriented (for instance, a character after getting shot just disappears instead of shedding blood, no obscene languages, no loud sounds, etc.).

Let us take the example of another war game (World War I or II) targeting mid/high level gamers (teens & above). The keyword for content verification here is realism – uniforms used, building layouts, weapon designs, spoken language accuracy (specific to the period in history), vehicle design, facial expressions of the characters involved, historical relevance of the challenges designed as part of the game, etc. The game provides leeway for a certain level of violence and language, but it should still be within ‘acceptable’ limits.

There are various standards for content verification defined by game manufacturers. Apart from that there are general standards designed for games to be used by general audience, parental guidance or restricted. The tester’s responsibility is to verify that the content is appropriate to the target audience. The biggest and the most unexpected challenge that any game might face is IP violation or use of copyrighted/trademark material as part of its flow. The tester’s experience and general awareness on IP/copyright laws and material is very essential in ensuring that the content is IP appropriate. For example, placing billboards of real soft drink companies or oil companies in a racing game will look realistic, but might lead to potential law-suits if prior approval or clearance is not obtained from the respective organization.

Play!! Strategize!! Improvise!!

This is the lifeline of any game and the most critical factor in its success. A functionally efficient game with excellent rendering of graphics & sound might still fall flat if the strategy base of the game is weak. This is also the part which requires the most inputs and intuition from the tester. There are linear games where the toughness gradually increases with each level. However, there are strategy based non-linear games which train and equip the player with all basic skills and suddenly throw a series of challenges at them.

Linear games are either low or mid level games – such as racing, novice adventure or sports based. The emphasis for victory is more based on player’s control. The game is predictable beyond a point with challenges thrown here and there. The main factor behind these games is to maintain the interest level of the user. Non-linear games are usually of mid-high or high level such as strategy games, war based scenarios, etc. The emphasis for victory is equally split between player’s control and their presence of mind. The levels are unpredictable and usually hard to win. The main factor behind these games is kindling the ego of the players, thereby maintaining the interest level.

The total play time consumed to complete a level is a very important metric for testers. While testing the playtime for these games, the tester should consider the average play time for other similar industry standard games and benchmark against them. However, the deviation should be measured and validated against the strategy of the game to decide if it is acceptable or not.

Also, the tester should test that there are sufficient challenges/mini-targets available on the way while completing a level. This is to ensure that the same player when playing and clearing the

same level at different times should get different scores. This can be tested by playing/testing the same level several times and ensuring that the score line differs almost every time.

The most important factor in a game that ensures repeat audience is the ‘unexpected’ factor which motivates the players to improvise. For example, double jump in a novice adventure game opens a sub-level full of challenges, or crashing at a specific billboard sign in a racing game opens a new shortcut for the racers. Testers should ensure that there are interesting sub levels available in a game to keep the audience interested. At the same time, it shouldn’t be an overdose which would dilute the main objective of the game.

There should be a clear differentiation between essential intervals in a game and fillers. Essential intervals are the ones which explain the strategy of the next level to the player once a level is cleared (example: war based games). Any interval that is not necessary for the continuation of the game can be classified as filler. Testers should ensure there is an option to ‘skip’ the fillers and that the essential intervals are made as interesting as possible.

The strategy of a game is put to its ultimate test during the user testing where a target audience provides pointed comments/ratings on the game based on their prior gaming experience. However, with the techniques described above a tester can add value during the normal testing phase itself thereby uncovering bugs/strategy flaws that typically get uncovered later.

Sound and light show!!

Graphics and music/sound effects are the two sensory media that help players in associating themselves with the game and hence go a long way in reinforcing their interest in the game.

The primary factor in graphics/video testing is the target audience. For low level games targeting pre-teens and children, the color pattern should be attractive. As the level of the game increases, the focus is more on realistic graphics, since the players at this level will be able to identify and appreciate the game surroundings better. Testers should be adopting a check-list based approach and score the game video based on the game (color pattern LOW/HIGH, usage of bright colors, realistic graphics Yes/No, top view graphics Vs lateral view graphics, intrusion of menus with actual gaming screen, etc.). Another good test will be to check the video resolution against various display systems (monitor, tablet, laptop, computer, televisions of various inches, home theatre, etc.).

Testing music/sound effects gets more complex as the level of the game increases. For low level games, the emphasis is on children or pre-teen friendly music/sound effects and not on realistic sound. However, as the level of the game increases, especially for graphic intensive and interactive games, the lip sync of the voice over with the character needs to be tested. This is true especially in case of movie-based games. Also, the music should not obfuscate the game dialog, and the sounds should be realistic and traceable, i.e., firing of a gun should trigger a gun sound.

In most cases, music or background score is created once the game is fully developed, and so the emphasis will be on the test user group to react to the sound/music and provide feedback.

What is my performance index??

Most of the games developed today are network friendly, and so

the game should support multi-player scenarios without any issues. The main focus of game testers is to test the performance of the game under various loads. This is not very different from software performance testing where the load (number of users) is increased over a period of time and the game's response is measured. At first, the game's response time is tested/benchmarked using simulators (multiple users/simulators present in the same network node). Then the game's response time is measured in real-time scenarios (various players accessing the game via network). Understanding of network's latency time is essential in determining the actual response time of the game in a real-time scenario.

Apart from testing the response time in a network, the game is also tested for any possible memory leaks, i.e., maintaining the integrity of various players in a game. The endurance of the game can be tested by leaving the game unattended for a long period of time and determining the response by providing a sudden input after a long time. The key in all of these tests is that the game should not hang, and even if it is faced with an unexpected input, it should make a graceful exit from the scenario (for example, re-setting to menu screen instead of hanging).

Compatibility

It is self-explanatory that the game should be software/hardware/browser agnostic as much as possible, or various versions of game should be available to the user on an 'on-demand' basis. Testers should be testing the game for all the above parameters explained under multiple combinations of operating systems, hardware and browser. A matrix of all possible permutations should be maintained, and test results should be recorded against each cell of the matrix. If repeating all the tests given above is time consuming, at least the functionality, graphics/video and audio tests should be performed to determine the compatibility.

Other tests

There are a host of other tests that a game tester can normally perform. These are highlighted in the following table:

Type of testing	Comments
Localization	Employing localized testers especially in case of cross-continent or cross-cultural games
Abuse testing	Multiple incoherent inputs through controller and determine game's performance (graceful exit from the scenario expected at the minimum)
Error handling	Testing for all error scenarios in the game to check if error messages are triggered
Independent testing of controller/joystick	Testing if the buttons generate system specified levels of signals (either measured as impedance or signal strength)
Regression testing	Focused/targeted testing around a fixed bug to ensure the unaffected functionality of the game is not broken by the fix
Compliance testing	Any compliance enforced by the game manufacturer, customer or important stakeholder

A game tester should have the quality to ExPLORE the game in order to test it properly:

- Experience the game as a player first
- Progressive thinking
- Logical ability
- Observant
- Reasoning Ability
- Exhaustive verification of features, functionality & game strategy

Practical Example – Authors' perspective of the extent of testing required for some of the games available in the market:

References: Infosys sources

Games/Type of Testing	Kids	Learning - Fun	Sports	Adventure/Strategic	Serious
Reference/ Name of Popular Game	Interactive	Black-Berry: Word Mole	18 wheels of steel; Cricket	Prince of Persia: The two Thrones	Virtual Fitness
Feature Testing	High	High	High	High	High
Sequence or Flow Testing	High – Med	High	High	High	High
Content Testing	High – Med	High – Med	High – Med	High	High
Strategy Testing	Low	Low	Med	High-Med	Low
Visual Testing	High	High	High	High	High
Sound/Audio Testing	Med	Med	Med	Med	Med
Error Handling Verification	Low	Low	Med	High	High
Performance Testing	Low	Low	Med	Med	Med
Moves Testing	Low	Med	High	High	High
Usability Testing	High	High	High	High	High
Localization Testing	Low	Low	Low	Med	Med
Abuse Testing	Low	Low	Low	Low	Med
Interoperability Testing	Med	Med	Med	Med	Med
Controller/joystick Testing	Low	Low	Low	Low	Low
Compliance Testing	Low	Low	Low	Low	Low
Compatibility Testing	Med	Med	Med	Med	Med

> biography



Kiran Marri

BE MS, is currently working as a Delivery Manager at Infosys Limited, Bangalore (NASDAQ: Infy www.infosys.com). He has over 15 years of IT experience in project management, client relations and developer roles. He has published and presented several papers at conferences in the field of project management, software testing, clinical data management and biomedical engineering. He has

also conducted workshops, tutorials on creativity, thought leadership, risk management, test management and defect prediction methods. His current research interest and publications are primarily in specialized testing, test maturity practices and innovation strategies. He received his Bachelors in Electronics & Communication engineering from Madras University in 1993 and a Masters by Research in Biomedical Engineering from Indian Institute of Technology Madras, Chennai in 1996. Kiran is also PMP certified. Kiran Marri can be contacted at kiran.kmr@infosys.com



Sundaresasubramanian

has about 10 years of experience in software testing, embedded systems testing and digital signal processing research & development. Currently, he is a Senior Project Manager with Infosys Technology Limited (NASDAQ: Infy) and part of the Independent Validation Solutions business unit. Before Infosys, he was involved in strategizing and delivery of testing projects in the automotive multi-media domain, and prior to that he was involved in research and development of digital signal processor-based embedded applications. He believes that excellence can be achieved by the eagerness to learn new things and the attitude shown in implementing them at work. He earned his Bachelor of Engineering degree from Manonmaniam Sundaranar University, India. He can be reached at ssubramanian_gv@infosys.com



→ people.steria-mummert.de



Faszination: Consulting

Leidenschaft: Testing

Perspektive: Andersdenkermacher



→ Weitere Infos



Die Andersdenkermacher.

Wir zählen zu den zehn führenden Management- und IT-Beratungen im deutschen Markt. Was uns von allen anderen unterscheidet: wir beraten nicht nur, sondern setzen die Ideen auch in die Tat um und begleiten den Kunden während des gesamten Veränderungsprozesses.

Consultant/Senior Consultant/Principal Consultant

Zertifizierter Software Test Manager (m/w)

Anders denken

Bei Steria Mummert Consulting haben Sie deutlich mehr Freiheiten als anderswo. Wir setzen großes Vertrauen in Sie und lassen Sie von Beginn an selbstständig arbeiten. Sie dürfen Ihren eigenen Kopf mitbringen und – noch viel wichtiger – Sie dürfen ihn auch behalten. Mitsamt Ihrem Wissen, Ihren Erfahrungen und Ihren Ideen.

Anders machen

Als Software Test Manager unterstützen Sie unsere Kunden in der operativen Konzeption und Steuerung von Testprojekten in heterogenen IT-Landschaften. Sie agieren selbstständig im Dialog mit dem Kunden und lenken Ihr Team erfolgreich durch die Testphasen. Dabei übernehmen Sie eine Teilverantwortung für die erfolgreiche Akquise aus den Projekten heraus.

Anders sein

Teamfähigkeit, gesunder Pragmatismus, Engagement und viel Verantwortungsbewusstsein ermöglichen es Ihnen, Ziele schnell zu erreichen. Da Sie als Berater oft beim Kunden vor Ort sein werden, ist die Reisebereitschaft innerhalb Deutschlands eine Selbstverständlichkeit für Sie. Darüber hinaus verfügen Sie über ein erfolgreich abgeschlossenes Studium oder eine vergleichbare Ausbildung, mehrjährige Erfahrung im operativen Test Management, Kenntnisse von Standardvorgehen wie TMap, Zertifizierungen nach ISTQB Advanced Level oder vergleichbare Zertifikate und tiefgehende Erfahrungen im Projektmanagement.

Ihr Einsatzort: Bei unseren Kunden im gesamten Bundesgebiet.

Für Fragen zu dieser Anzeige steht Ihnen Claa Bendels telefonisch unter 040 / 22703-7608 oder -7680 gerne zur Verfügung.

Jetzt online bewerben. Bewerben Sie sich jetzt mit Angabe Ihrer Gehaltsvorstellung unter people@steria-mummert.de oder mit dem Bewerbungsformular auf www.people.steria-mummert.de



Knowledge Transfer – The Trainer Excellence Guild

Díaz Hilterscheid



From User Stories to Acceptance Tests

by Gojko Adzic

• Oct 10 – 12, 2011

Amsterdam



NEW: Implementing a Successful Acceptance Test Procedure

by Yaron Tsubery

• Oct 10 – 11, 2011
• Oct 13 – 14, 2011

Stockholm
Amsterdam



Rapid Software Testing

by Michael Bolton

• Dec 13 – 15 2011

Oslo



Risk-Based Testing

by Hans Schaefer

• Dec 14, 2011

Helsinki

Website:
www.testingexperience.com/knowledge_transfer.php



Agile **TESTING DAYS**

November 14–17, 2011
Potsdam (near Berlin), Germany

www.agiletestingdays.com



November 14–17, 2011
in Potsdam (near Berlin), Germany

The **Agile Testing Days** is an annual European conference for and by international professionals involved in the agile world. This year's central theme is "**Interactive Contribution**".

Please visit our website www.agiletestingdays.com for the current program.

Follow us on Twitter: www.twitter.com/AgileTD

**Agile Testing Days 2011 –
A Díaz & Hilterscheid Conference**

Díaz & Hilterscheid Unternehmensberatung GmbH
Kurfürstendamm 179
10707 Berlin
Germany

Phone: +49 (0)30 74 76 28-0
Fax: +49 (0)30 74 76 28-99

info@agiletestingdays.com
www.agiletestingdays.com

A photograph of a man with short brown hair and a beard, wearing a black t-shirt and blue jeans, sitting at a wooden desk. He is looking towards the camera with a slight smile. On the desk in front of him is an open silver laptop. Behind him is a large bookshelf filled with books of various colors. In the top right corner of the image, there is a graphic element consisting of a white rectangular stamp with a circular postmark. The stamp contains the Agile Testing Days logo, the number '11', the word 'POTS DAM', the date 'Nov 14-17 2011', and the word 'GERMANY'. Below the stamp, the text 'WE WANT YOU' is visible.

**Testers –
Don't loose your head!**

Register for

Agile
TESTING DAYS



The Agile Testing Days 2011 "MIATPP"- Award

Who do you think is the "**Most Influential Agile Testing Professional Person**" 2011?

Please vote until **September 30th** and you could be the winner of **2 free tickets** for the Agile Testing Days 2011!

Vote at www.agiletestingdays.com/award.php!

Agile Testing Days 2011 Photo Contest

The Early Bird season is over – either you missed it or the Early Bird discount was not enough and you want to save some more money for the Agile Testing Days Conference?

Take a **Planking** or **Horsemaning** picture from you or your team and win a **discount of 50%** from the regular prices for the 3 Conference Days (Tutorial Day excluded). The best 3 pictures will earn this fabulous special discount!

Send your picture together with your contact details until **October 15, 2011** to info@agiletestingdays.com *

We wish you luck for the Agile Testing Days 2011 Photo Contest.

Where do you get your inspiration? Here!

Planking Gallery



Horsemaning Gallery



Want to exhibit?

If you'd like to be an exhibitor at Agile Testing Days 2011, please fill in the form which you can find in our exhibitor brochure and fax it to +49 (0)30 74 76 28-99 or e-mail it to info@agiletestingdays.com.

» Download our exhibitor brochure at www.agiletestingdays.com

Exhibitors & Supporters 2011



..T..Systems..

Seapine Software™

NOKIA

XING

SEQIS
The Austrian Software Test Experts!

telerik
deliver more than expected

* You agree that your picture will be uploaded on the Díaz & Hilterscheid Unternehmensberatung GmbH Facebook Fanpage and to the vote results by the community. The picture with the most "likes" until October 31, 2011 will win.

**4 Days
5 Tracks
10 Tutorials
60 Speakers**



"Hooray! We're Agile Testers! What's Next? Advanced Topics in Agile Testing"
Lisa Crispin



"Transitioning to Agile Testing"
Janet Gregory



"Making Geographically Distributed Projects Work"
Johanna Rothman



**"Dealing with Differences:
From Conflict to Complementary Action"**
Esther Derby



"Influence Strategies for Practitioners & "Patterns for Improved Customer Interaction"
Linda Rising



"Critical Thinking Skills for Testers"
Michael Bolton



**"Winning big with Specification by Example:
Lessons learned from 50 successful projects"**
Gojko Adzic



"Introduction to BDD"
Elizabeth Keogh



"Acceptance Testing: From Brains to Paper and Paper to Computer"
Lasse Koskela



"Agile Management: Leading Software Professionals"
Jurgen Appelo

Save the date, discover the place.



Potsdam

The most beautiful place for bright minds

Mr. Net is a Potsdamer! This interactive sculpture, created by the Spanish artist Jaume Plensa, is located on the grounds of the Hasso-Plattner-Institute, an unmatched centre of academic excellence for IT systems-engineering in Germany. The combination of first class alumni, innovative business start-ups, and successful companies makes Potsdam a premium IT location. Come to Potsdam and discover the city for yourself: as a lovely holiday destination, as a multi-faceted event location, or as an attractive place for your business.

Potsdam is a unique mixture of world heritage and high-tech, of science and business, of Brandenburgish cultural milieu and Mediterranean charm. It is an attractive city and a premium location for business with outstanding prospects for future growth. Potsdam is also a growing city with creative potential. Located right next door to the metropolis Berlin, Potsdam offers everything you need to succeed. Potsdam: a great place for great ideas.

This great city welcomes you warmly!



State Capital Potsdam
Department of Business Development
economy@potsdam.de | www.potsdam.de



Semi-formal property verification in games

by P. Arun Babu & N. Murali

Game design, development and testing is artistic and quite informal in nature, hence the use of formal methods has not gained popularity in this community. Even though formal methods have been successfully used to formally specify and verify small safety and mission critical applications, they are not considered well suited for specifying large and complex real-world applications. Most of the practicing programmers and testers find it difficult to use them, due to the steep learning curve and complex tools.

This article shows how programmers and testers can use simple programming languages (especially their favorite scripting languages) in the semi-formal verification of properties in games.

Difference between formal verification and testing

Formal verification is a method of proving certain properties in the designed algorithm, with respect to its specification written in a formal mathematical language/notation. One approach to formal verification is formal proof, which is a finite sequence of steps which proves a certain property of software. In contrast, software testing is an activity (manual or automated) to identify bugs in software which provides overall information about the quality of the software. The main difference between testing and proof is: testing can prove the presence of errors, but not their absence, whereas a proof can.

Testing	Proof
Process of giving input and comparing the program's output/behavior with known output/behavior.	Sequence of steps based on mathematical principles to prove/disprove certain properties in a software
Coverage depends on tester's test cases, and these test cases are not exhaustive for a real-world application.	Coverage depends on assumptions made by proof writer and can be exhaustive.
Based on executing the program under test.	Non-execution based. Proofs are written separately based on the specification.
Specification is generally written in natural language like English.	Specification is written in formal languages/notations.

Testing	Proof
Writing test cases requires relatively less effort than writing a proof.	A lot of effort, understanding and review are required in writing a proof for real-world problem.

Figure -1 demonstrates the difference between proof and testing of Pythagoras theorem.

Why use semi-formal methods?

Game development by nature is chaotic and relies heavily on current culture and human psychology (see Figure -2). Full scale formal methods are difficult to fit in the development methodology used in game development. Due to its rigor, formal methods require a lot of time and analysis, which may not be suitable to a competitive industry like gaming. Moreover, most of the successful formal methods projects have been developed in collaboration with academic institutions, and this may not be feasible for all organizations.

Property verification - Case study: Tic-Tac-Toe

In this section we discuss a case study of a very simple and well known game: the Tic-Tac-Toe.

In this game, computer (X) plays against the human opponent (O) using the following strategy:

The algorithm for computer to choose the next move (in the following order):

1. Search for a winning move, if found: choose the winning move.
2. Search for a blocking move, if found: block the opponent.
3. If center of the board is free, make the center move.
4. Try for a corner move in which opposite corner is also empty : in this order :
 - A corner move, which can give two ways to win.
 - A corner, near which the opponent has placed most the moves.
5. Search for an edge in which opposite edge is also empty.
6. Randomly select any move from left over moves.

The diagram shows two parts: a geometric proof on the left and a Python code snippet on the right.

Geometric Proof:

- A large right-angled triangle has legs of length a and b , and a hypotenuse of length c .
- The area of the main triangle is $\frac{1}{2}ab$.
- The triangle is divided into four smaller triangles and a central square of side $b-a$. The total area is also given by the sum of the areas of these four triangles plus the area of the central square: $\frac{1}{2}(b-a)a + \frac{1}{2}(b-a)b + (b-a)^2$.
- Equating the two expressions for the area of the main triangle leads to the Pythagorean theorem: $c^2 = a^2 + b^2$.

Python Script:

```

1 import math
2
3 - def calculate_hypotenuse (a,b) :
4     return math.sqrt (a*a + b*b)
5
6 - test_cases = [
7     # Our test cases, in the format :
8     # (a,b,c) - sides of triangle
9
10    ( 3, 4, 5 ), # A simple triangle
11    ( 0, 100, 100 ), # Triangle with ZERO length side
12    ( 68, 285, 293 ), # Bigger triangle
13
14    ...
15
16 number_of_failed_test_cases = 0
17
18 - for a,b,c in test_cases :
19     #
20         hypotenuse = calculate_hypotenuse (a,b)
21
22     - if hypotenuse != c :
23         # This test case has failed !
24         number_of_failed_test_cases += 1
25
26     #
27
28 - if number_of_failed_test_cases == 0 :
29     print "All test passed !"
30 - else :
31     print number_of_failed_test_cases, \
32         " test cases failed out of", len (test_cases)

```

Figure – 1: Pythagoras theorem: Proof vs. Testing (in Python) (Derivative work of: http://en.wikipedia.org/wiki/File:Pythagoras_algebraic2.svg)



Figure – 2: Example of a game design whiteboard
(Used with permission http://www.gamasutra.com/view/feature/2662/schadenfreudian_slips_das_.php)

This case study aims to verify the property:

"If computer plays first using the above specification, there is no way computer can lose the game."

Proof of the above property by elimination and exhaustion:

- Generate all possible boards i.e.: with ,X', 'O' and ,,(EMPTY)

- Eliminate boards which cannot be a valid game – (see Table - 1)
- Make a list of possible boards in which computer loses – (see Table - 1)
- Remove duplicate boards from the list of losing boards – (see Figure - 3)
- For each losing board, back track (see figure – 4), and:

- Play again using the above given algorithm.
- Check the played board:
 - If the opponent has a chance to win in the new board :
 - Back track once more and repeat the above steps.
 - Else, remove this board from list of losing boards, as computer would have played this game differently and it would not lose the game.

6. At the end:

- If there are no left out losing boards, then property is proved.
- Else: proof has failed and the property does not hold good.

NOTE: Semi-formal specification, verification and proof for the above problem in Python programming language can be found at author's homepage: <http://www.arun-babu.com/fm.html>

Stage	Number of boards left	Remarks
Initially generated boards	19683	Each slot of board can be filled in 3 ways ('X', 'O' or EMPTY) = $3 \times 3 \times 3 \times 3 \times 3 \times 3 \times 3$
Eliminate boards in which center is not 'X'	6561	As per the algorithm, if computer plays first, it would try to make the center move first.
Number of Xs in board should be equal to number of Os in board	1016	If 'X' plays first and 'O' wins, last move must be made by 'O', hence both must have placed same number of moves.
O should be the winner and X should not win	84	As we are making a list of X's losing boards, all boards in which O loses or X wins, are ignored.
Number of Xs in corners should at least be 1	76	As X starts looking for a corner after placing in center, X should occupy at least one corner
X would have made at least one blocking move before losing	24	X always tries to block O's winning moves. If O wins, there must be at least two ways for O to win, in which one would be blocked by X.
After removing duplicates	3	Eliminate mirror images and rotations
After backtracking and verification	0	Each losing board is backtracked, and computer is let to play the backtracked board as per the given specification. If computer plays the board differently, and does not give the opponent a chance to win, then such a board can never occur.

Table 1 - : Table for number of left boards at each stage of proof

0 1 2	6 3 0	8 7 6	2 5 8
3 4 5	7 4 1	5 4 3	1 4 7
6 7 8	8 5 2	2 1 0	0 3 6
6 7 8	8 5 2	2 1 0	0 3 6
3 4 5	7 4 1	5 4 3	1 4 7
0 1 2	6 3 0	8 7 6	2 5 8

Figure – 3: Duplicate boards (rotations and mirror images)

Conclusion

- Writing a good proof requires thorough knowledge of software specification (preferably written in executable formal/ semi-formal notation/language).
- A formal proof would give designers more confidence than traditional testing.
- When compared to full-scale formal methods, semi-formal methods are more suitable for specifying and verifying real-world games.
- Similar to mathematical proofs, formal proofs also need to be thoroughly reviewed.
- Writing proofs in simple programming languages makes it reviewable by practicing software engineers and testers.

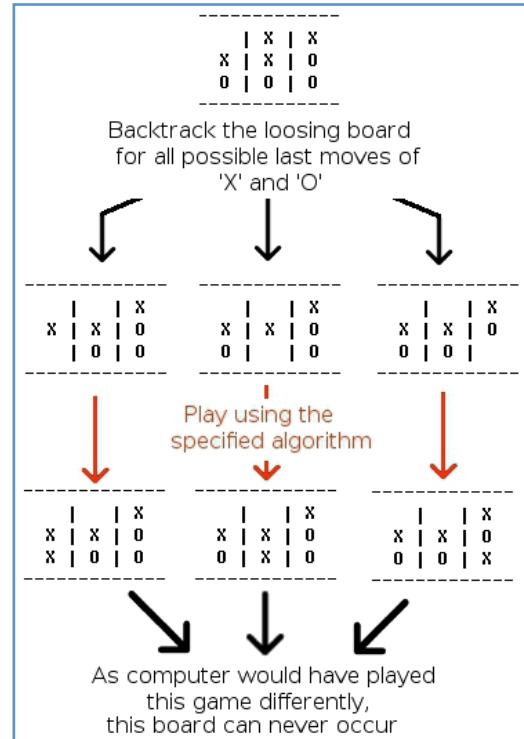


Figure – 4: Backtracking and verifying the losing board

> biography



P. Arun Babu
is a Ph.D. student at the Indira Gandhi Centre for Atomic Research, Kalpakkam, India.

He holds a Masters in Technology (information technology) from the University of Hyderabad. His areas of interests are software quality and reliability.



N. Murali
is currently Head of the Real-Time Systems Division at the Indira Gandhi Centre for Atomic Research, Kalpakkam, India. He holds a Masters in Engineering (computer science) from the Indian Institute of Science, Bengaluru. His research area spans

domains like software engineering, computer networks, distributed control system, network security, human machine interface design, etc.

What do you need to be agile on mainframes?

Test early, test often

savvytest for System z enables you to continuously check the functionality of your software components on z/OS, and thus measure and document the substantial progress of your project after every iteration.

www.savvytest.com

Join our customer's field report on the IBM Rational Software Conference Innovate 2011, Munich, 18th October.



 **savignano**
SOFTWARE SOLUTIONS

Phone +49-7141-5071766
E-mail info@savvytest.com

License ISTQB and IREB Training Materials!



Díaz Hilterscheid

Díaz & Hilterscheid creates and shares ISTQB and IREB training material that provides the resources you need to quickly and successfully offer a comprehensive training program preparing students for certification.

Save money, and save time by quickly and easily incorporating our best practices and training experience into your own training.

Our material consists of PowerPoint presentations and exercises in the latest versions available. Our special plus: we offer our material in 4 different languages (English, German, Spanish and French)!



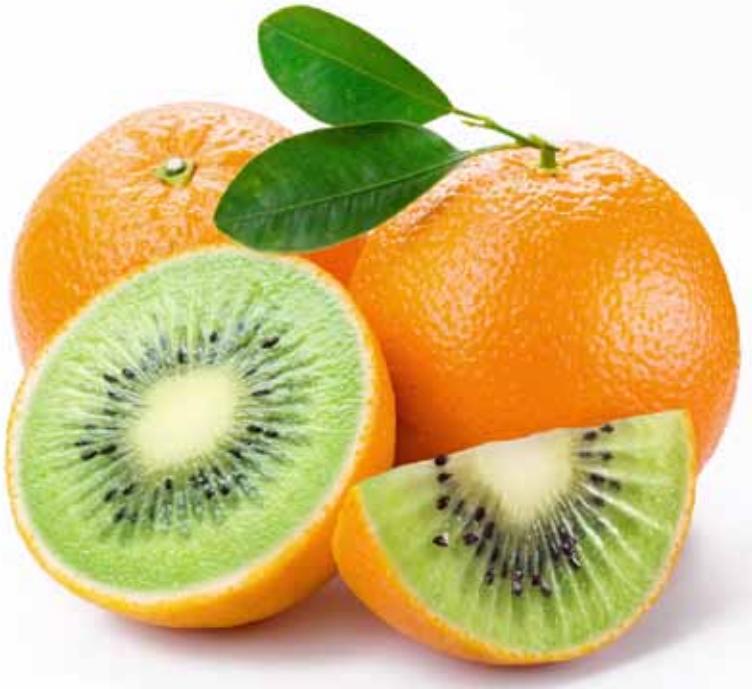
International
Requirements
Engineering
Board



For pricing information, and all other product licensing requests, please contact us either by phone or e-mail.

Phone: +49 (0)30 74 76 28-0

E-mail: training@diazhilterscheid.com



Gamification of testing

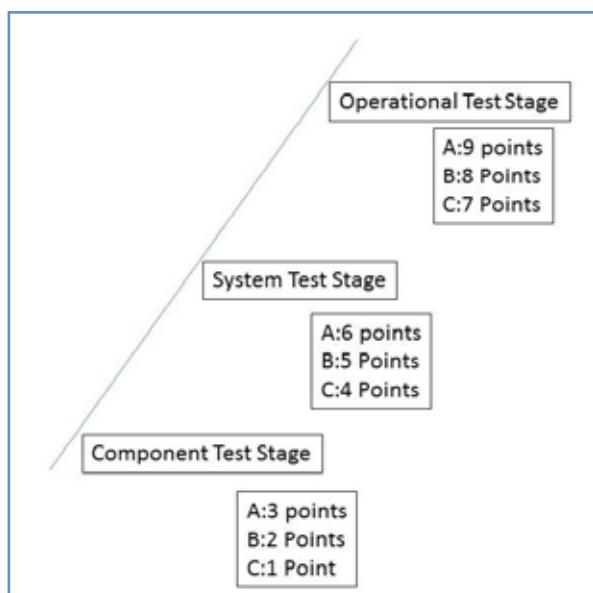
by Wayne Ellis

The newest buzzword around the internet at the moment is gamification, which is to make a game out of everyday activities such as visiting shops/places or carrying out tasks at work. In your normal day you might have introduced gaming already; think about tasks you do, like travelling to work, when the train pulls up do you predict where to place yourself on the platform for the door to stop opposite, in the supermarket do you judge which queue will be the quickest to stand in. It is possible to introduce this technique in testing, to enhance your test team's productivity, accuracy and enjoyment. This article is mainly focused on introducing games for your test team rather than competing against yourself, since where is the fun in beating yourself? Let's take a look at a couple of possibilities of gamifying your testing.

Defect management

Let's assume we are using a simple ABC categorization of defects; where A is for your showstoppers and C is for your minor defects. At the end of the project you can count these up and you will get metrics for who raised what defect and who raised how many.

Looking at diagram 1, you should see half of the V-model, and at each stage of testing you will see different point values for finding defects in that stage. Then instead of estimating the number of



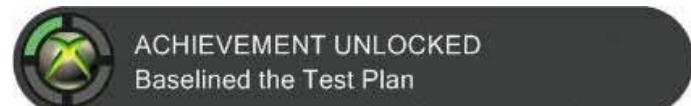
defects you predict to find at each stage, you estimate a point target for each stage; you would expect the component test stage to find more defects and the operational test the least.

With those changes, you suddenly have a game, a competition between each of your test phases, especially if each of these test stages is carried out by separate teams. The component phase team should be able to win by finding all the defects early, but if not, then it's the system test teams to pick up points and so on. With this game it should be showing your test teams the emphasis of finding defects early on in the lifecycle, and that finding defects of all types, will make a good-quality product in the end. With the test teams recording their scores, more detailed metrics should be available, as this score will now matter to your teams, because no one likes to lose.

Of course these rules can be tweaked, for example adding in penalties for raising defects that are not defects to curb cheating and bad defect practices, or by adding bonus points for finding tickets that should have been found earlier but were instead found in later stages.

Testing progress

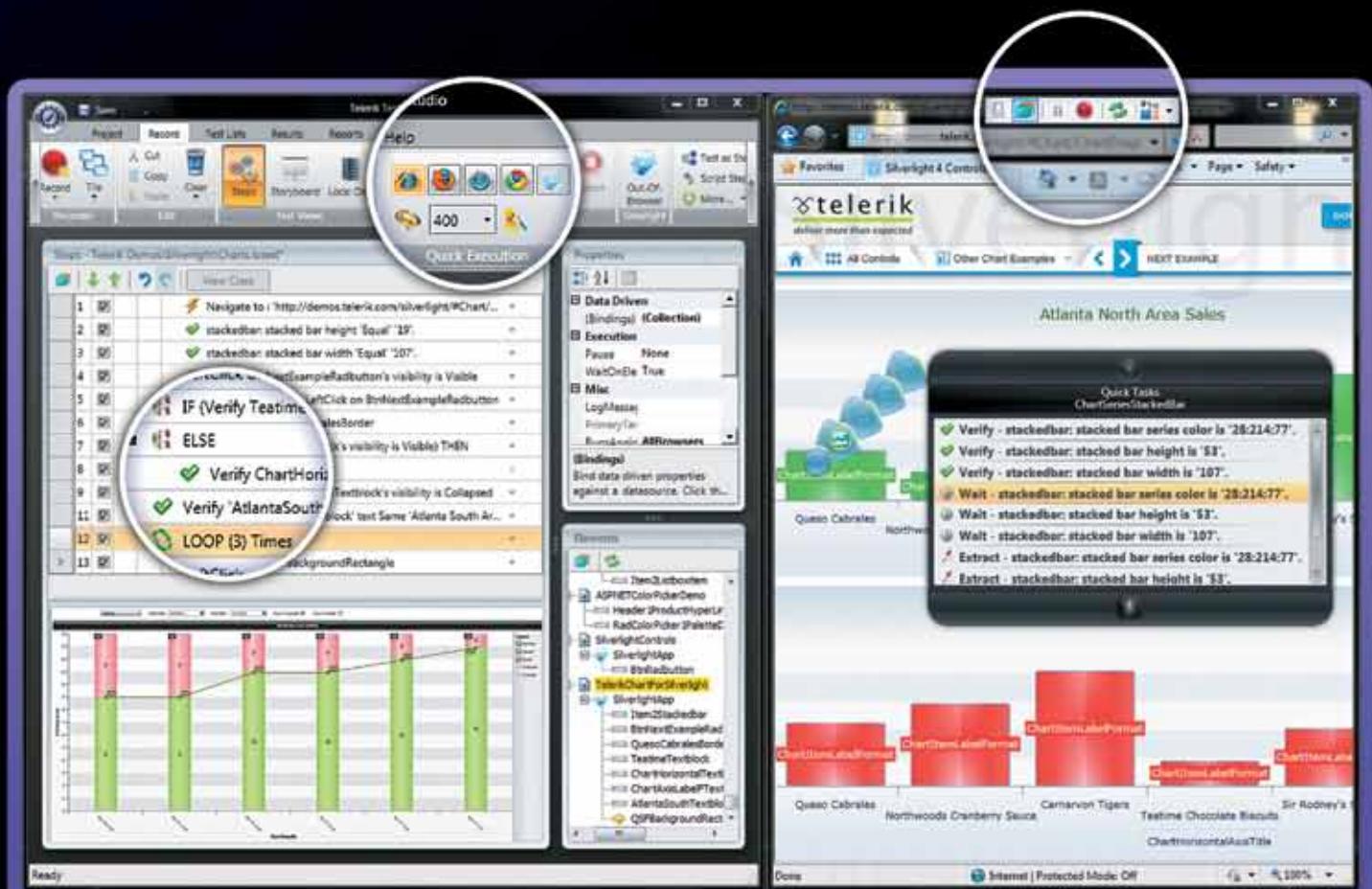
Within video games the player is rewarded with trophies or achievements for tasks done during the game, be it complete a level or carry out a side task or doing a task a number of times. If you look at these achievements as milestones in the timeline of the game (i.e. the achievement of completing a level progressing further in the game), with minor achievements for completing your nice-to-have tasks, then you can see how that be ported into testing.



Work out your milestones, and have them converted into achievement icons as shown in diagram 2. If you give each of your milestones a point value, you can show testing in a progress bar, where completed achievements are displayed with the progress bar, which shows the percentage of completed milestones. In just one page you have a high level of graphical display showing where testing is.

Testing is Now Radically Easier

Telerik Test Studio



Effortless Automated Functional Testing

- Test web and desktop apps
- Next-generation test recording
- Record once, run against multiple browsers

End-to-end App Performance Testing

- Unparalleled insight into your app performance metrics
- Visualize performance stats over time
- Easily convert functional tests to performance tests

Download your free trial and get 20% off Test Studio at:
www.telerik.com/TestingExperience



ACHIEVEMENT UNLOCKED

You Found a Bug!

For your side tasks, as shown in diagram 3, you could display how many defects have been found so far, what documents the testers have reviewed and other such tasks.

Reviews

If you are carrying out formal structure-based reviews of documents, why not try placing the rules into a Bingo Card instead of a list. This puts some excitement into that somewhat dull task, and you can tailor that card to the individual carrying out the review, having them cover off the rules you assign them. For example the project manager should be able to cover off his rule card with the test plan parts of that document.

References:

1. <http://gamify.com/> is a company that has created a game mechanics for web enabled systems, they also have created a Wiki website <http://gamification.org/wiki/Encyclopedia>
2. <http://www.bbc.co.uk/news/business-13749897> BBC article on gamification.
3. <http://achievements.schrankmonster.de/> a website where you can generate your own xbox360 achievement icons.

> biography



Wayne Ellis

Having passed his ISEB foundation and practitioner for software testing, Wayne has worked for Capgemini as a senior test analyst. He has 7 years of experience and specializes in data warehouse testing. He was once described as follows: "Wayne doesn't think just outside the box, he is thinking in a different room to that box altogether".

ORGANISED BY:



READERS OF
AGILE RECORD
RECEIVE A 10%
DISCOUNT
PLEASE QUOTE
REF. AR011
WHEN BOOKING

Finding smarter ways to test is a continuous goal for many testers; and within their span of influence they can achieve amazing results. Whether it is implementation of agile testing, improved test design techniques, exploratory testing (and so on...) - making changes work for the individual tester is relatively easy; having them adopted as mainstream activities is something quite different and a major challenge organisations face. This conference explores the most effective ways of scaling "smarter testing" to benefit the whole team and the organisation.

Details/booking

Visit www.next-generation-testing.com or call + 44 (0)1895 256 484 or email info@unicom.co.uk

The 6th NEXT GENERATION TESTING CONFERENCE (Autumn 2011)

MOVING "SMARTER TESTING" INTO MAINSTREAM

2-3 November 2011, London

www.next-generation-testing.com

Some of the topics that will be addressed and discussed include:

- Agile Project Scaling
- Assuring quality in a virtualised operating environment
- Automating Requirements and Risk Based Testing
- Behaviour Driven Development (BDD)
- Building a QA Team for the Agile Age
- Cloud Testing
- Coping with complex systems
- Expressing Testing in terms of Business Outcomes
- Improved Test Design
- Moving to Agile from traditional projects
- New approaches and ideas for managing testing both traditional and Agile projects
- Quality Management in an Agile Context
- Radically disruptive approaches to Testing
- Test Automation Strategies

SUPPORTED BY:





Online Training

Díaz Hilterscheid

ISTQB® Certified Tester Foundation Level (English & German)
ISTQB® Certified Tester Advanced Level - Test Manager (English)
ISTQB® Certified Tester Advanced Level - Test Analyst (English)
ISTQB® Certified Tester Advanced Level - Technical Test Analyst (English)
ISEB Intermediate Certificate in Software Testing (English)

Our company saves up to
60%
of training costs by online training.
The obtained knowledge and the savings ensure
the competitiveness of our company.
www.te-trainings-shop.com



Turning into AQA organization

by Andrei Contan & Cristina Fuia, & Oana Oros

I have a dream....of a company where product quality is in first place and where client satisfaction is the prime concern. This is what I was saying a year ago when I personally joined the company and when things were a bit blurry. Fortunately for me, everything turned into "I had a dream" because others had the same vision and luckily this vision turned to reality and everything started to shape up. The same vision I was hoping to see became reality in a company where everything used to be focused on fast delivery and rapid patches.

"When joining a company, it is important to understand not only the company vision but also its values, operating style, and culture" a line by Mark Tompkins in his paper on Quality, which got people inspired and enforced the trust among them. It is important to understand the context and how you can contribute to the delivery and the quality of the product. All this will have an immediate positive impact on the processes and, consequently, on the products.

All companies are evolving from a strict and rigid model of designing, developing and testing software, towards a more flexible and customizable approach: the Agile model. This model will mean nothing without the values and the culture around it, but it will have a significant importance if applied into a favorable context or environment.

As a group we must ensure that all employees share the same dream on quality and take a collaborative approach. Quality should be the main responsibility of each organization, whether it's project management, development or QA. In addition, responsibilities should include being the contact person and being accountable for the overall quality of your work, by ensuring continuous improvements and removing any impediments, both internal and external. As long as values like mutual support, trust, teamwork are shared, correctly understood and applied, the level of quality will increase without making any specific effort towards it.

The decision of having a separate QA organization in an Agile environment has its pros and cons, but on a positive note, we will only talk about some of the pros and let the others provide any suggestions or concerns.

One positive aspect is the importance and the visibility of a career ladder. It has the advantage of knowing the direction you are heading for or the path you would like to choose. I find it very useful and encouraging to know that I can move either on the vertical or on the horizontal, as both steps will provide the advantage of a different view and the consolidation of strong technical skills. We have defined levels from basic test case executions up to complex coverage and QA, but also taking into consideration test case generation and expertise in complex automation areas.

Better products can be translated into satisfied customers, which involves a balance between product quality and speed. In a very dynamic world, the time to market is very important if you want be one step ahead. That is why a separate QA organization, correlated with an Agile model, will help you maintain the status of 'no tradeoff' between the quality of your products and the speed of deliveries, ensuring a sustained and fast ROI.

I once heard that "A good tester cannot be replaced by tools, only helped by them", which is exactly the approach that every QA organization should take, and this is exactly what we did. Moreover, we managed to unify QA tools and processes across all products and projects. Also as part of the strategy, we have defined a set of tools following the open-source model, easy to maintain and develop, which perfectly suits our needs, reducing the technical debts as much as possible. All test scenarios are shared across teams, as we use the same repository for storing and maintaining them.

Last but not least, one of the most important assets of a separate organization is the human part with all its values. I'm proud to say that we achieved in building a great community where we share ideas and debate issues, as great ideas are generated through great discussions. We plan to go beyond the company lines and gather QA people across local companies where we talk about conferences, talks, technical challenges, blockers and many more.

How it all became possible?

I will describe further some of the decisions taken by the leader of our new organization and what the effect was on the regular QA engineers. One of the earliest decisions was to automate the most

important features so we can spend less time for integration and regression testing. We already did this within our teams, but the automated tests were not unified across all teams. This lead to the next decision that we need *to pick a tool or a framework so we can all work together* and integrate our tests. The system we are working on is complex and the components are decoupled as each team owns a part of this big system, the interaction is low, but even so, having the tests unified gives us the possibility to integrate the components faster and release sooner.

Being part of the Agile development organization made me gain a lot of experience as a QA engineer from both a technical and also from a communication point of view, but, like in all programming books, once you know how to do something, the next step is to know the best way of doing something. This is how I like to call the new, parallel thread born as a QA organization, it felt like the “best practices” chapter of a programming book. It felt like we, the QA engineers, *stay committed to the Scrum teams* and we will continue doing the day-to-day work.

One of the biggest improvements from the series “best practices” is also the need to shorten the period of time from when the code is ready and tested through the integration phase 24 hours a day, 7 days a week. The approach to achieve the “Lights Out” capabilities **is to split the responsibilities between the QA Delivery teams and a Test Frameworks team.**

The Test Frameworks team is in charge of creating the core testing utilities and tools to be used by all the other testers and to help them automate the most critical functionalities. Once the tools are available, another team would pick **the most suitable compliant framework and used it in doing their day to day work.**

The members of the Test Frameworks team are developers and passionate automation testers - they assure we have a formalized approach to automated testing, to implement centralized test execution, to create general test automation standards, to create utility/test branching/merging standards, to set utility coding standards in-line with development, to implement standardized and centralized automation reporting, to assess automation progress and to support maintenance of the Frameworks Backlog. The Frameworks Backlog contains requirements from the QA delivery teams, as they request utilities and tools which aim to bring value to the QA process.

The benefit of splitting the responsibilities between teams for these tasks is reducing the impact on current QA teams or ‘in-flight’ projects. The QA Delivery teams still focus on their daily tasks and they are still committed to the teams and projects.

On a different note, the QA strategy of our organization was definitely enhanced by reviewing the defect management process.

The strategy taken into consideration when redefining the defects management process was based on the extent that the whole process is created to prevent defects and *defect reporting should be used to improve the process.*

The major goal was to increase quality of our deliverables and customer satisfaction. The defect management process, as described in the following, brought significant benefits in the increase of the business, commercial and technical value.

The process is based on the following general principles:

- *Workflow improvements across the company* – all flows have been redefined and unified to correspond with the defect management process.
- *Feature classification* – another principle that brought consistent value for the business processes was introducing a common language across the company’s projects.
- *Defects Prioritization* – critical defects associated with the system were assessed and a proper prioritization matrix was introduced. People involved in all the projects were aware of the types of defects that are most likely to occur and the ones that can have the greatest system impact.

For each prioritization category, several factors were taken into consideration: functional impact, capacity impact, third party impact, security impact, compliance impact, QA internal impact and, most important, customer impact and total financial impact.

In order for every team to understand and associate their priorities with the defined levels, specific examples were provided by each manager. Once the four categories were defined, a time schedule for the fix of the defect was associated with each category.

P1 – Showstopper – fix within 2 days

P2 – Critical – fix within 5 days

P3 – High – fix in next release

P4 – Low – fix as required

A major process improvement consisted in *introducing a resolution report* (Close Loop) for understanding what caused the defect and whether the defect should have been caught earlier in the process as well as gathering actions to prevent defects in the future. This also helped to investigate what other defects may be present that have not been discovered yet.

The resolution report will contain a summary of the defect and details covering the following topics:

- What is the issue?
- Steps to reproduce
- When should this issue have been caught?
- What actions are required?
- Where are we now?

This process has brought significant value for preventing defects by having a documented understanding about the failures and actions taken.

Reporting production-critical issues was also introduced. All P1 and P2 production issues need to be communicated to Senior Quality Manager / Quality Director within 24 hours, including outages and defects found on live.

A proper repository was created for these reports which offered an increased visibility across the whole organization.

Information collected during the defect management process helps the management team in taking informed decisions, in analyzing defect trends, in taking actions for reducing technical debts.

As with every vision, our roadmap has continuity with lessons learned and actions to be taken. As a final conclusion, I would like to emphasize the fact that overall, it was a good experience being involved in all the changes by providing continuous feedback and suggestions, which enriched people with trust and visibility across organization. This collaborative approach improved ,Quality in building cross-functional tools and processes with a focus towards 'test early, test often' and moving towards a robust continuous delivery process.

> biography



Andrei Contan

currently a PhD student in Software Testing and certified in ISTQB Foundation Level and CSM (Scrum Alliance), has been working as a QA engineer since 2005 and has been involved in various projects across many technologies in different organizations.



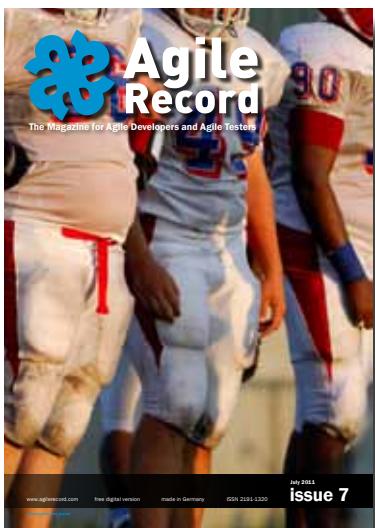
Cristina Fuia

holds a Diploma in Communications Engineering, is ISTQB-certified and CSM certified (by Scrum Alliance). She has been working as a QA engineer in various projects covering multiple technologies and has been involved in Agile testing since 200



Oana Oros

holds a Diploma in Computer Science Engineering, is ISTQB and CSM certified, and has experience as a software tester in various traditional and Agile development projects. Lately she was Senior QA Engineer and acting as QA Manager for multiple Agile projects within Betfair company, and has experience in both manual and test automation.



The Magazine for Agile Developers and Agile Testers

subscribe at
www.agilerecord.com

EUROPE 2012

Testing & Finance

May 16 - 17, 2012
in London, UK

The Conference for Testing & Finance Professionals

www.testingfinance.com

May 16-17, 2012
in London, UK

EUROPE 2012
Testing & Finance

The Future of Finance

The theme of next year's conference is "Testing and Finance in Agile Projects, Social Media and Regulation".

is this year's theme of the eighth Testing & Finance conference taking place in London for the first time in May 2012.

These two days provide a unique chance for software testing professionals in financial institutions to learn, to confer and most importantly, to meet and network with other practitioners. Some of the best known professionals and experts in software testing and quality assurance in finance from around Europe will share their knowledge and experiences at the event.

For the last seven years this event has grown in stature to become the most recognized in the financial sector. To further broaden its appeal, Díaz & Hilterscheid GmbH has decided to move the conference from Frankfurt to London and to expand the scope of the program.

As before, there will be talks about new developments and processes in the areas of software testing and reporting. There will be a higher number of field reports and case studies for recent projects in financial institutions; there will also be leading-edge talks on testing in a regulatory environment, and sessions on risk, goal and profit-based reporting. The 2012 event will be truly international and not to be missed!

The conference is aimed at practitioners who are actively involved in, for example, projects concerning Risk Management or External Reporting for banks. The conference will benefit professionals and decision makers who like to exchange ideas with peers and experts regarding the latest developments in Bank Supervision for instance. The Testing & Finance conference will also suit those who are interested in solving problems that concern the technical assurance, testing, and Quality Assurance of upcoming projects.

We encourage Business Stakeholders, Testers, Test Managers, Quality and Test Assurance professionals to join us and take part.

Speakers Wanted!

Submit your proposal until the **end of September!**

Visit: http://www.testingfinance.com/call_for_papers.php



©iStockphoto.com/titaniumdoughnut

Testing & Finance Europe 2012 – A Díaz & Hilterscheid Conference

Díaz & Hilterscheid Unternehmensberatung GmbH
Kurfürstendamm 179
10707 Berlin (Germany)
Phone: +49 (0)30 74 76 28-0
Fax: +49 (0)30 74 76 28-99
www.diazhilterscheid.de

Follow us on Twitter: www.twitter.com/testingfinance

Gerrard Consulting
PO Box 347, Maidenhead
Berkshire, SL6 2GU
Phone: +44(0) 1628 639173
Fax: +44(0) 1628 630398
www.gerrardconsulting.com

Sponsors, Partners and Supporters



PureTesting
Testing Thought Leadership

gerrard consulting

Díaz Hilterscheid

te testing experience

Agile Record

Quality, what does it mean?

by Erik van Veenendaal

Let me start by making clear that I'm not in the games corner. I do not have a game room where I spend evenings playing like some fathers do. My personal interests are very different. However, I have three children, a 3-year old, a 6-year old and a 9-year old. The latter two are boys and of course have their Nintendo-DS, which they love to play with. If it was up to them they would play for hours and hours.... We also have a Wii in our house, so we can play golf, baseball, table tennis, cycling etc. I must admit, I enjoy the occasional Wii games with my sons. We have bought expensive games, low-price games, games oriented towards younger children etc. One can clearly experience the difference in quality. Some games need rebooting more often, some are just easier to play (more user friendly), some have a better performance, some my sons just like (I sometimes do not have a clue why), etc. For these various reasons we have bought them as being interesting, good quality games (at least that is what we expected). So quality seems to imply different things, what do we mean when we speak of a quality product? This question gets even more interesting if we compare the game industry to other industries. Clearly a medical product has a higher level of quality than any game. And if so, what exactly do we mean or not mean by "a higher level of quality"?

Quality definitions

Before starting testing activities, e.g., defining the test strategy and test approach, there must be consensus about what quality really means in a specific business context. The objectives of the project in terms of quality must be clear. Otherwise what are we aiming for? Only then can wrong expectations, unclear promises and misunderstandings be avoided. Garvin showed that in practice generally five distinct definitions for quality can be recognized [Garvin]. I will describe these definitions briefly from the perspective of testing based on an earlier publication [Trienekens and Van Veenendaal].

The product based definition

Quality is based on a well-defined set of software quality attributes. These attributes must be measured in an objective and quantitative way. Differences in the quality of products of the same type can be traced back to the way the specific attributes, e.g., reliability, performance, maintainability, have been implemented. This is highly related to what is often called non-functional testing. Discuss with stakeholders which attributes are of importance and need to be tested. Often the mere existence of the attributes makes the difference.

The user based definition

Quality is fitness for use. This definition says that software quality should be determined by the user(s) of a product in a specific business situation. Different business characteristics require different "qualities" of a software product. Quality can have many subjective aspects and cannot be determined on the basis of only quantitative and mathematical metrics. This is related to the validation process. It is related to testing with use cases, end-users, beta testing etc.

The manufacturing based definition

This definition points to the manufacturing, i.e. the specification, design and construction, processes of software products. Quality depends on the extent to which requirements have been implemented in a software product in conformance with the original requirements. Quality is based on inspection, using formal test design techniques and (statistical) analysis of defects and failures in (intermediate) products. In this definition testing is covered by verification, e.g., using formal test design techniques such as decision tables and classification tree method, and applying traceability from test design to requirements.

The value based definition

This definition states that software quality should always be determined by means of a decision process on trade-offs between time, effort and cost aspects. The value based definition emphasizes the need to make trade-offs, this is often done by means of communication with all parties involved, e.g. sponsors, customers, developers and producers. This quality definition relates to risk-based testing and the good enough paradigm. How much testing is enough? Which product risks must be mitigated, etc.

The transcendent definition

This "esoteric" definition states that quality can in principle be recognized easily depending on the perceptions and the affective feelings of an individual or group of individuals towards a type of software product. Although the least operational one, this definition should not be neglected in practice. Often a transcendent statement about quality can be a first step towards the explicit definition and measurement of quality. I have always found this one very difficult to use in testing, but perhaps to some extent it is related to the games industry. Why is a game appealing to my sons, I sometimes just do not understand: the transcendent definition?

Using the definitions....

The existence of the various quality definitions shows that it is difficult to determine the real meaning and relevance of software quality and thus the focus of the testing activities. Testing practitioners have to deal with this variety of definitions, interpretations and approaches. I have learned over the years that in discussing the test strategy and test approach, it helps to also start a discussion about product quality. What does it mean for the stakeholders and what is expected? The framework as presented in this column has proven to be highly useful and easy to apply. Of course in practice it is often a mix, the discussion however makes things much clearer to all, and expectations become more aligned. Such a discussion should take place on a project level, but also on an organizational level to drive (test) improvement activities. What does product quality mean for your stakeholders?

[Garvin] D. Garvin, "What does product quality really mean?", in: *Sloan Management Review*, Vol. 26, No. 1, 1984

[Trienekens and Van Veenendaal] J. Trienekens and E. van Veenendaal, "Software Quality from a business perspective", Kluwer Bedrijfsinformatie, 1997, ISBN 90-267-2631-7

> biography



Erik van Veenendaal (www.erikvanveenendaal.nl) is a leading international consultant and trainer, and a widely recognized expert in the area of software testing and quality management with over 20 years of practical testing experiences. He is the founder of Improve Quality Services BV (www.improveqs.nl). At EuroStar 1999, 2002 and 2005, he was awarded the best tutorial presentation. In 2007 he received the European Testing Excellence Award for his contribution to the testing profession over the years. He has been working as a test manager and consultant in various domains for more than 20 years. He has written numerous papers and a number of books, including "The Testing Practitioner", "ISTQB Foundations of Software Testing" and "Testing according to TMap". Erik is also a former part-time senior lecturer at the Eindhoven University of Technology, vice-president of the International Software Testing Qualifications Board (2005–2009) and currently vice chair of the TMMi Foundation.

Probador Certificado Nivel Básico Tester profesional de Software

Formación para el Probador Certificado - Nivel Básico
de acuerdo al programa de estudios del ISTQB[®]

República Argentina



Docente: Sergio Emanuel Cusmai

Co - Founder and QA Manager en QAustral S.A.

Gte. Gral. de Nimbuzz Argentina S.A.

Docente en diplomatura de Testing de Software de UTN - 2009.

Titular y Creador de la Diplomatura en Testing de Software de la UES XXI - 2007 y 2008.
(Primer diplomatura de testing avalada por el ministerio de Educación de Argentina).

Team Leader en Lastminute.com de Reino Unido en 2004/2006.

Premio a la mejor performance en Lastminute.com 2004.

Foundation Certificate in Software Testing by BCS - ISTQB. London – UK.

Nasper - Harvard Business school. Delhi – India.





ISTQB Training



Díaz Hilterscheid

Austria

Date	Days	Training	Place	Provider
2011-11-14	4	Certified Tester Foundation Level	Mödling	SEQIS/Díaz & Hilterscheid

Canada

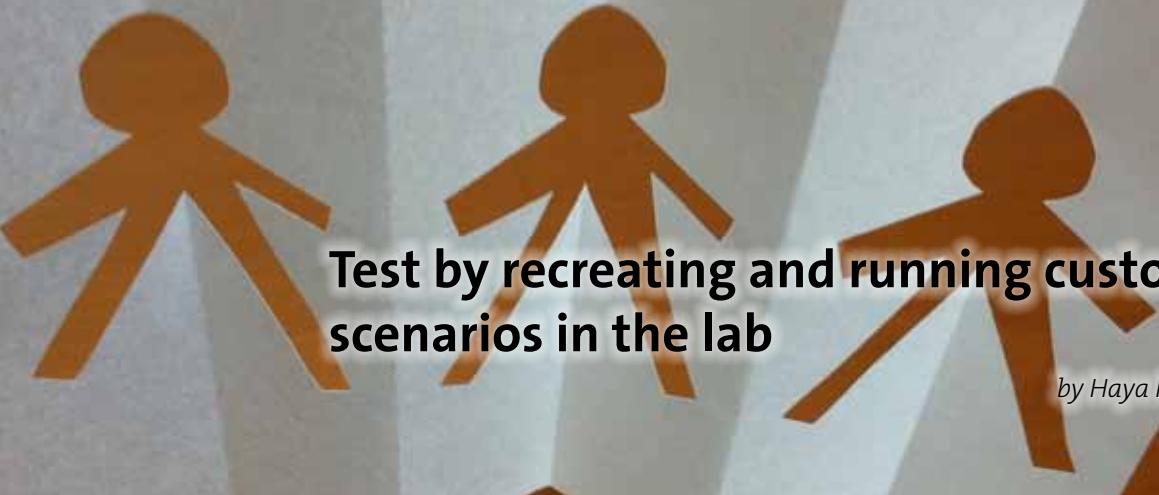
Date	Days	Training	Place	Provider
2011-09-12	5	Certified Tester Advanced Level - Test Analyst	Toronto, ON	SELA Canada
2011-09-26	3	Certified Tester Foundation Level	Toronto, ON	SELA Canada
2011-10-24	3	Certified Tester Foundation Level	Toronto, ON	SELA Canada
2011-11-28	3	Certified Tester Foundation Level	Toronto, ON	SELA Canada
2012-01-23	5	Certified Tester Advanced Level - Test Manager	Toronto, ON	SELA Canada

Germany

Date	Days	Training	Place	Provider
2011-09-12	4	Certified Tester Foundation Level	Munich	Díaz & Hilterscheid GmbH
2011-09-21	3	Certified Tester – Foundation Level	Berlin	qme Software GmbH
2011-10-10	3	Certified Tester Foundation Level	Frankfurt	Díaz & Hilterscheid GmbH
2011-10-17	4	Certified Tester – Foundation Level	Berlin	Díaz & Hilterscheid GmbH
2011-10-19	3	Certified Tester – Foundation Level	Berlin	qme Software GmbH
2011-10-24	5	Certified Tester Advanced Level - TECHNICAL TEST ANALYST	Munich	Díaz & Hilterscheid GmbH
2011-10-31	5	Certified Tester Advanced Level - TEST ANALYST	Berlin	Díaz & Hilterscheid GmbH
2011-11-07	4	Certified Tester Foundation Level	Berlin	Díaz & Hilterscheid GmbH
2011-11-09	3	Certified Tester – Foundation Level	Berlin	qme Software GmbH
2011-11-14	5	Certified Tester Advanced Level - TESTMANAGER	Frankfurt	Díaz & Hilterscheid GmbH
2011-11-21	3	Certified Tester Foundation Level	Munich	Díaz & Hilterscheid GmbH
2011-11-23	3	Certified Tester – Foundation Level	Berlin	qme Software GmbH
2011-11-28	5	Certified Tester Advanced Level - TESTMANAGER	Berlin	Díaz & Hilterscheid GmbH
2011-12-05	4	Certified Tester Foundation Level	Stuttgart	Díaz & Hilterscheid GmbH
2011-12-07	3	Certified Tester – Foundation Level	Berlin	qme Software GmbH
2011-12-12	3	Certified Tester Foundation Level	Frankfurt	Díaz & Hilterscheid GmbH
2011-12-19	3	Certified Tester – Foundation Level	Berlin	qme Software GmbH

Spain

Date	Days	Training	Place	Provider
2011-10-10	5	Certified Tester Advanced Level - TESTMANAGER	Mallorca	Díaz & Hilterscheid GmbH



Test by recreating and running customer scenarios in the lab

by Haya Rubinstein

Are you familiar with the following situation?

You released a new version of your software to the customer. Before release, your team ran thousands of regression tests in each sprint, and tested each new feature carefully, but when the version was released, your customer support team received messages reporting about issues that you wished you had found earlier.

We experienced this with our product.

Like many companies we spent extensive resources on testing. We tested all new functionality that was developed in each release. We developed and ran exhaustive automatic regression tests. We even tested every bug-fix that was checked in for those bugs found in previous versions. We also had a validation stage where an external team installed our software from scratch and ran some common scenarios. However, once our releases hit the market, our customer support still received messages from customers.

Why does this happen?

If you have widely configurable software, that has many interoperating and interdependent capabilities, it is virtually impossible to test all the possible configurations and processes that can be used in your software. The usual modus operandi is to narrow down the possible matrices using tools such as All Pairs, and to use risk based methodologies to discover what needs testing the most.

Risk based testing usually takes into account parameters such as how much is the functionality used, whether the area was changed in the recent development, how critical the functionality is, how stable it was in previous releases, and what impact a failure of this functionality would have on a customer. With risk based testing you need to spend time adjusting the test scope for every sprint or iteration according to the changes in development.

All pairs enables you to only test all the pairs instead of all the single configurations in the matrix, thereby statistically ensuring that you will find ca. 90% of the bugs in the areas you are testing. All pairs is useful for testing one feature at a time.

We discovered that most of the bugs the customers encountered occurred because of a combination of functionalities and configurations that is hard to find using the regular methods we use in our regression and new feature testing.

We decided that in addition to the testing done by our scrum teams in development and in addition to our very wide automatic regression suite, we need to play the same game that our customers play in order to discover the bugs that will affect them the most.

So how do we play the game?

We founded a team whose task is to play the same game as the customer within our lab.

How did we start?

Choose the most representative scenarios.

We decided to look for a set of customers that can represent all the capabilities and processes our software covers. After examining many customers, looking through their messages and mapping what components and processes they use, we selected the ones we deemed either most representative of our customers or most interesting. We looked for both breadth and depth. We involved our customer support and our development in making these selections.

Once we selected the customers, the real work began:

We used a simple three-step process:

1. Learn: Learn the customer scenario
2. Plan: Plan the scenario test
3. Execute: Test, report and summarize

Learn

Step 1: Find out how your customer uses your software.

We prepared a detailed questionnaire to find out details such as the following:



Díaz Hilterscheid

Austria

Date	Days	Training	Place	Provider
2011-09-19	3	Certified Professional for Requirements Engineering - Foundation Level	Mödling	SEQIS/Díaz & Hilterscheid
2011-10-05	3	Certified Professional for Requirements Engineering - Foundation Level	Mödling	SEQIS/Díaz & Hilterscheid

Finland

Date	Days	Training	Place	Provider
2011-10-26	3	Certified Professional for Requirements Engineering - Foundation Level	Helsinki	Díaz & Hilterscheid GmbH

Germany

Date	Days	Training	Place	Provider
2011-10-26	3	Certified Professional for Requirements Engineering	Berlin	qme Software GmbH
2011-11-30	3	Certified Professional for Requirements Engineering	Berlin	qme Software GmbH
2011-12-14	3	Certified Professional for Requirements Engineering	Berlin	qme Software GmbH

Norway

Date	Days	Training	Place	Provider
2011-11-23	3	Certified Professional for Requirements Engineering - Foundation Level	Oslo	Díaz & Hilterscheid GmbH

Miscellaneous

Canada

Date	Days	Training	Place	Provider
2011-11-21	2	Implementing a Successful Acceptance Test Procedure by Yaron Tsubery	Toronto, ON	SELA Canada

Finland

Date	Days	Training	Place	Provider
2011-12-14	1	Risk-Based Testing by Hans Schaefer	Helsinki	Díaz & Hilterscheid GmbH

Norway

Date	Days	Training	Place	Provider
2011-12-13	3	Rapid Software Testing by Michael Bolton	Oslo	Díaz & Hilterscheid GmbH

The Netherlands

Date	Days	Training	Place	Provider
2011-10-10	3	From User Stories to Acceptance Tests by Gojko Adzic	Amsterdam	Díaz & Hilterscheid GmbH
2011-10-13	2	Implementing a Successful Acceptance Test Procedure by Yaron Tsubery	Amsterdam	Díaz & Hilterscheid GmbH

Sweden

Date	Days	Training	Place	Provider
2011-10-10	2	Implementing a Successful Acceptance Test Procedure by Yaron Tsubery	Stockholm	Díaz & Hilterscheid GmbH

- What components does the customer use? In what builds?
- What platforms are they running on?
- What volumes of data does the customer have?
- What roles are there?
- What operations are carried out?
- How many concurrent users are running and what are they doing?
- What are the customer's current pain points?

We request sample data from the customer.

If there is a blueprint document, we request that too.

Step 2: Set up the environment.

Using this information we are able to reproduce the customer environment within our lab.

We use the same platform as the customer as much as possible.

Step 3: Analyze the scenario

We then use some internal tools to analyze the customer data.

We map out all their processes, roles and operations to see what end to end scenarios they use.

We also look through the customer's messages to see what additional pain points they may have and to find out more about their scenario.

Once we have mapped out the scenario, we summarize it in a document.

At this point we can usually see which areas we will need to focus on, and if we can recommend any helpful workarounds for the customer.

Before we continue, we hold a review meeting with some of our team members. Usually an extra set of eyes can see different things that we may have missed.

Plan

Step 1: Decide what to include in your test

We decide what end to end scenarios we need to reproduce. If the customer uses custom code, we may decide to implement their scenario without the custom code they use, since we are not interested in dealing with issues stemming from the code they wrote.

Step 2: Prepare the data

We prepare all the necessary data for the test cases. We create new data in the same format as the customer data to simulate new data when necessary. We may also need to generate some new sample data or to create batch files or automatic processes and recordings. We use tools such as LoadRunner to generate large multi-user operations. Such tools can also help us find performance issues.

Step 3: Incorporate new features

We review all the new features in the current release and find out which ones the customer is interested in, or would be most likely to use in their scenario. We then decide how to incorporate them

within our end to end customer scenarios.

Step 4: Write the test cases

We write test cases that reproduce the customer scenario. We try to run the processes from beginning to end, just as the customer does, on the data that we received from the customer. We create new data in the same format as the customer data to simulate new data.

Once our scenario runs pretty well, we include multi-user tests by manual users from outside our team who test for one hour. At the same time we run scripts with LoadRunner to simulate the large number of users that the customer actually has.

The multi-user tests need to be very well planned.

The manual users are able to notice usability problems that would probably be missed by scripts. The test cases we give them are wide enough for them to cover many different areas. Some of our multi-user test cases have an exploratory nature.

As you have probably discovered, no one tester will test in exactly the same way, so this enlarges our actual coverage.

All our tests will start with an upgrade of the customer data to our currently tested build.

An example for a scenario test could be the following:

Begin with data import from many locations in parallel, continue with workflows that involve many automatic updates and interactive approvals, notifications, and matching operations and end with syndication. In parallel to the automatically run tests, include a manual multi-user test and an automatically generated multi-user search and update.

We also incorporate the new features that may interest the customer into the customer scenario itself once they have been tested and approved by the development product owners, and run them as part of the scenario.

Test

Step 1: Run the test.

All test runs and bugs including the multi-user and exploratory tests are recorded using our company tool. Once a scenario is ready, it can be rerun with hardly any effort, and follow-up on issues from previous runs is simple.

We automate whatever can be automated, and add it to the automatic regression suite too.

We set a higher priority on the bugs we find, since we know that at least one customer will definitely run into the bug, if it is not fixed before release.

Even though we run the scenarios after the sprint has been approved, we discover many interesting bugs on the way.

Many of these bugs are of the really hard to reproduce type and can only be reproduced with a particular configuration.

You are probably familiar with this type of bug.

"To reproduce: Get two people to stand on one leg and hold their ears for two minutes and then jump up and down on the customized trampoline.

Expected results: They will both bounce up and down four times.

Actual results: An error will occur after the third bounce and the trampoline will crash."

Many of the bugs we find are critical for customers. It is much better and cheaper to find them during development, as we do, than to find them after release.

Step 2: Report findings.

At the end of each test period we provide a report on what we did and which bugs we found. We map out the impact and severity of the bugs we found for each customer and a summary for all the customers whose scenarios we ran in this test period. This enables development to see the product quality from a customer oriented perspective.

For example (see below):

We recommend changes to customers that would help them utilize our product better.

We recommend workarounds for known issues for the customer scenario that are not yet fixed.

We communicate to product owners regarding missing features and regarding features that were developed in a way that does not meet customer scenario expectations.

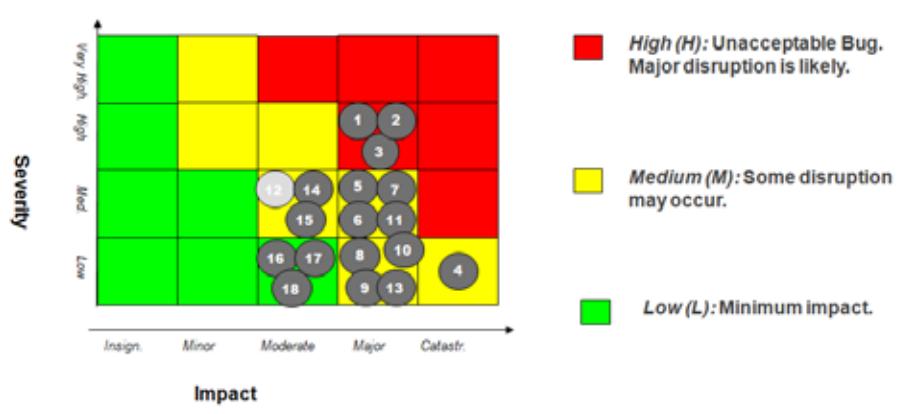
Since we adopted this mode of scenario testing, we have seen a great improvement in our product quality.

> biography



Haya Rubinstein
has ten years of Quality experience in SAP NW MDM, and is currently working on customer interaction prerelease in SAP NW MDM.

CIPR Team List of Top Bugs





How to start cutting-edge testing with your sword

by Guilherme Motta

Every single game has its own context and reality. Some games focus on communities and multi-playing while others are heavily based on single playing. Why would this matter when it comes to testing? We can easily co-relate to our experience testing any other kind of software. When we test an application, we should know what it is made for, what the context is, who the users are today, and who will be using it in the future. With all this information we can set a direction for testing. There is no such thing as a software without issues or a software that is 100% secure. If it is a single player flash based game that expects to get a few players per day, you do not need to worry much about performance, load and stress testing. If you are testing a game for which people killed themselves to get keys for beta testing and sold them all over the internet, it might be useful for the development team to receive information about things such as latency of the testers, how long it took to launch the game, how was the experience to install the game, and so on.

Why do massive multi-player games open their application early to get feedback on their alpha/beta testing phases? Depending on the game, it is likely that there will be already a community in place. From that community you may find: developers, artists (ui/ux folks), testers, fans, supporters and many other different kinds of human beings (sometimes even not humans). If the game is being highly anticipated, there will be thousands of people around the world looking forward to spending their time playing (of course!), experiencing the game in an early stage and also reporting bugs. As these people have different backgrounds, they will approach the game with different perspectives and knowledge on how to test the specific game.

The person who will be testing the game also needs to understand what the business goal of the company is that is releasing that game. Will they follow a F2P (free-to-play) model with an option to buy in game items? Will it have a monthly fee, and if so, how high will this be? Feedback such as „I wouldn't pay 10 US\$ per month for this game“ supported with reasonable arguments is a good example of a valuable feedback. If you know in advance what the plans for the game are, it is good to think as if you were a customer of that game.

Wait a second, isn't that crowd sourcing? People from a community are getting paid to find issues, bugs, features and improvements for the game that is not even in production yet. You might argue that they are not getting money from actively accomplishing this task, but indirectly they are having fun experiencing and playing the game. I enjoyed a couple of games that I was lucky to be able to test and get prizes for doing so. A few examples were in-game cash, „titles“ and/or items that would distinguish myself from other players. This early experience that I had testing massive multi-player games brought my attention to the job of a tester. Years have passed and sometimes I still work as a tester, not like before, not with games, but with enterprise/corporate level software. It is even better when your reward is somehow associated to your performance and results as a beta tester. Metrics such as number of issues found is not enough, unless there is a very strict process of reviewing and requesting the feedback in place.

Among the crowd, there are also people that are not interested in testing at all, but they are there to get information from the application as early as possible to get advantages from it. Security exploits and issues that can benefit players might not be shared with the development team (going against ethic and terms that are generally agreed before testing starts). There are companies that make huge amounts of money (real money!) selling items, in-game money and other intangible/virtual things.

Do games rely only on crowd testing? I'm confident that big players from the gaming market have their own quality assurance team that works before, during and after the alpha/beta testing phases. That is an assumption, though, as I have never worked for a gaming Quality Assurance team.

The gaming community is hard to deal with. Assuming that the scenario is very high hype, a new game from a big player, the expectations will be huge. If alpha/beta testers report too many issues, bugs and features and the development team cannot keep up fixing them, it could also affect the company image and reputation. In 2010, I played a MMORPG that was launched with a lot of severe issues. I bought a new computer and invested a lot of my money and time playing the game and giving feedback to

that game, and it ended up that they launched the game with pretty much the same code base. After a few months I was very disappointed and stopped paying their monthly subscription fee.

After everything mentioned, I would like to share what I believe would be a draft for good guidelines to follow with regard to the goal of testing and improving a game:

1. What is company behind the game? - Research it!
2. What are the business goals of the game (profit, marketing, increase portfolio, revenue, increase cash flow...) - This is hard to get right without context, but it is likely that you can get a clue from forums and developers.
3. What are the core functionalities? - Usually these are used to advertise the game, so it should be easy to find this out.
4. What are the core differences from games that are leading in that market? - Usually this is used to advertise the game, so it should be easy to find out.
5. Functional testing: hands on the game! - This is the most fun part of the process.
6. Whenever you find an issue/bug/feature, write it down. Don't save it for later, write it down and try to reproduce and define how easy it is to reproduce this issue.
7. When you are going to report an issue/bug/feature, add as much information as possible such as your system configurations (hardware/OS/etc.).
8. Write down in one place positive feedback and somewhere else negative feedback as well, try always to link with examples (stay away from evil assumptions!):

- „The game takes too much time to load“ <- This is not worth the feedback as you cannot abstract what „too much time to load“ really means / „The game took 15 minutes to load in my Pentium 1 133MHz with nothing else opened on the machine“ <- This is better feedback and probably more useful for the person that will work on it.

> biography



Guilherme Motta

Affectionate about people, I have been working and learning on IT practical projects for more than 5 years. Have been a passionate gamer since I first got my master system console some 15 years ago. As a very generalist technical guy, I worked with several different companies around the world playing multi-roles functions related to management, business process modelling & mapping, information & infrastructure support, quality assurance, security audits and development of software. Currently, I am doing a MBA at Fundação Getúlio Vargas in Brazil and working at ThoughtWorks advocating Agile as a consultant for one of the biggest retail sales companies of the world.



Lassen Sie sich auf
Mallorca zertifizieren!



Díaz Hilterscheid

Certified Tester Advanced Level TESTMANAGER - deutsch

10.10. – 14.10.2011 Mallorca



Can agile be certified?



Find out what Aitor, Erik
or Nitin think about the
certification at
www.agile-tester.org

Training Concept

All Days: Daily Scrum and Soft Skills Assessment

Day 1: History and Terminology: Agile Manifesto, Principles and Methods

Day 2: Planning and Requirements

Day 3: Testing and Retrospectives

Day 4: Test Driven Development, Test Automation and Non-Functional

Day 5: Practical Assessment and Written Exam



Certified Agile Tester

We are well aware that agile team members shy away from standardized trainings and exams as they seem to be opposing the agile philosophy. However, agile projects are no free agents; they need structure and discipline as well as a common language and methods. Since the individuals in a team are the key element of agile projects, they heavily rely on a consensus on their daily work methods to be successful.

All the above was considered during the long and careful process of developing a certification framework that is agile and not static. The exam to certify the tester also had to capture the essential skills for agile cooperation. Hence a whole new approach was developed together with the experienced input of a number of renowned industry partners.

Supported by

Barclays

DORMA

Hewlett Packard

IBM

IVV

Logic Studio

Microfocus

Microsoft

Mobile.de

Nokia

NTS

Océ

SAP

Sogeti

SWIFT

T-Systems Multimedia Solutions

XING

Zurich



Game On!

Applying game theory concepts for strategic decision making in software testing

by IJsbrand Kaper MBA

Test professionals claiming that they would always make the same decision based on factual project metrics, such as defect rates, coverage ratios and acceptance criteria, will probably hit a few walls over time in their projects or organization. They will be overruled, ignored or not taken seriously. What these people do not realize is that test metrics are only used to support a decision and in fact are only one argument for it among several others. There may be many more arguments to base a decision on. Most of them, unfortunately or fortunately, are never mentioned in the test plan.

This implies that there is always a certain amount of uncertainty involved in the decision making in test projects. It also implies that other people's decisions may influence the outcome. Because of this, decisions in test projects should be strategic rather than only factual.

This article describes a number of ways how concepts from game theory can help test professionals to make better strategic decisions. Had I known these concepts at an earlier stage, they would have helped me to prevent some of the many bad decisions I made in test projects I worked for in the past. Game theory applications in IT are not uncommon, as game theory has been used in areas such as artificial intelligence, e-commerce or networking¹.

Game theory concepts

Game theory is used to model interactions between parties in order to analyze or predict certain behavior of individuals or groups. Game theory is widely used (and being researched) in strategic decision making, such as pricing decisions. Imagine a large ICT consultancy firm in a downwards moving economy, facing the decision whether or not to lower their hourly rates. The payoff may be large when the company is the first to do it (first-mover advantage). The outcome, however, may also be that nothing is sold, because the other players in the field decided to lower their prices beneath the levels of the first-mover. Another appliance of game theory is in the field of research, where research is conducted to see whether investment levels for IT security can be based

on game theory concepts².

The game theory model contains the following aspects: the decision making entities or players who are assumed to be rational, a strategy, an outcome, a payoff and an equilibrium state³.

Firstly, the model assumes multiple players. Simply said, there would not be a game if, there were only one player. In a test project this always applies to the situation, as there are always at least a test professional and a principal (who can be seen as the product owner or customer).

Secondly, it assumes uncertainty of the outcome of the decision, as the outcome is dependent on the combination of strategic choices made by the players. An example would be a case in which a test manager decides to recommend not to move to production with a certain release of the application under test, while at the same time a project manager, given the advice of the test manager, decides to ship the software.

Thirdly, the model assumes a state of equilibrium. This is a state in which players have no reason to adapt their strategic choices, given the strategic choices of the other players in the game.

Strategic decision making and testing

Knowing that decision making can always be strategic, the concepts of game theory may be applied to any situation where decisions are made. Software testing is all about decision making. I would even argue that software testing exists because people want to make better informed decisions. First of all, decision making occurs at the business level of testing. Testing serves as a tool for project managers and product owners to provide them with information to support decisions, such as "Should I move to production now?", or "Should we build this extra feature or not?". Test professionals provide information related to the current status

² Cavusoglu, H., Raghunathan, S., Yue, W.T., Decision-Theoretic and Game-Theoretic Approaches to IT Security Investment, *Journal of Management Information Systems*, Fall 2008, Vol. 25, No. 2, pp. 281–304

³ Madhani, Pankaj M., Salesforce compensation: Game Theory, October-December 2010, *SCMS Journal of Indian Management*

¹ Shoham, J., *Computer Science and Game Theory*, *communications of the acm*, august 2008, vol. 51, no. 8.

quo of the test project in terms of risks or progress against a pre-defined set of acceptance criteria. Secondly, decisions are made at the project level. For example, decisions made on the test approach, appropriate testing techniques and resource allocation.

Thirdly, decision making occurs at the individual level. Every test professional has to make decisions too within the boundaries of their part of the test project. For example, a tester involved in an exploratory testing session, where he or she has to decide whether or not it is best to further 'explore' a specific routine with a specific technique or to move on to another approach.

Where are you in the game

Generally, testers seem to be risk-averse. It is in the nature of their jobs to be like this. I rarely have seen testers trying to persuade their project managers to move to production.

We can further elaborate on the assumption that testers are risk-averse by using a simple model from game theory. The model as shown in the figure represents two players: a test professional and a principal. The strategic choice in the model is whether or not to continue testing (test or not test). The order of actions is that the test professional makes a recommendation and that the principal makes the final decision.

		test professional	
principal	test	not test	
test	1	-2	
not test	0	2	
Expected payoff	0,5	0	

The payoff for the test professional is measured in terms of the level of satisfaction of the principal about the test professional after the project has come to an end. The payoff ranges from -2 to 2, where -2 is 'not satisfied at all' and 2 is 'very satisfied'. The payoff values are relative and subjective values, as it is impossible to assign objective scores to the outcomes.

The four possible outcomes of the model are explained as follows. When the test professional recommends to test, and the principal agrees, the test professional will probably be recognized for his expertise by the principal, leading to a score of 1. In this situation, no value of 2 has been given, as the advice to test is the 'easiest' option for the test professional. The assumption here is that no system can be tested completely. Hence, there is always a valid argument to continue testing. In the situation where the test professional recommends to stop testing, and the principal decides differently, the principal clearly overrules the test professional and therefore will most probably question his expertise, leading to a score of -2. The assumption here is that the principal and the tester have the same project information and that acceptance criteria are clear. On the other hand, if the test professional recommends to stop testing, and the principal decides the same, the level of appreciation is the highest, as the advice to stop testing contains a higher risk for the test professional. This is also represented in the score when the test professional recommends to continue testing and the principal decides not to test. The fact that there are always arguments to continue testing is reflected in the score of 0. It is not 'bad' to recommend to continue to test, but compared to recommend to stop testing it is, again, the safer choice.

By taking the averages of the scores of each decision from the perspective of the test professional, we get the expected payoff for each decision. In this model, the test professional is most likely to recommend to continue testing, as this will have the biggest payoff. This corresponds with most test projects, where test professionals are more likely to test too much, rather than too little.

Beware of the consequences

Because we have learned from a rather simple model that testers tend to be 'better safe than sorry', we can use the knowledge to our advantage and improve our strategic decision making. This is not the same as simply making the same decisions as the principal, but to work on increasing the expected payoff of our decisions. This means that the probability that a certain decision is adopted by the principal should be increased. In the first example, we have assumed an equal probability distribution between the two options (test or not test) of 0.5. In this case we try to manipulate the probabilities by taking corrective actions.

		test professional	
principal	test	not test	
test	1	-2	
not test	0	2	

Suppose that a test professional has valid arguments to recommend to continue testing for a specific piece of software. For example, not all acceptance criteria have yet been fulfilled. However, the last two releases, the product was shipped earlier too and, despite some minor problems, nothing went wrong. Because of this, you as a test professional expect the principal to decide not to test and to ship the software.

As a test professional, there are two possible actions now which will increase the possibility for a higher payoff. The first option is to persuade the principal to decide differently. This might be done by discussing the acceptance criteria and their usage. You might even mention that results from the past are no guarantee for the future. By doing this, the principal may decide differently and be satisfied afterwards. The second option is to adopt the decision of the principal. You might say that, given the experience of the last couple of releases, it might make sense to ship somewhat earlier and perhaps take a good look at acceptance criteria or the intensity of the test approach. In this case, the satisfaction level of the principal will increase from 0 to 2, and the expected payoff is higher too.

What makes decision making strategic?

If software testing was a routine activity, there would not be so many test professionals. The thing is that a test professional who knows how to maximize the expected pay-off will be more effective. This is why more experienced test professionals know that theoretically correct decisions are not always the most effective ones. To illustrate this, consider the following situation. If we look at the first example, it seems the best option to always recommend to continue to test. Now, if you unconsciously take the same decision every release over a certain period, you will become very predictable. In mathematical terms, the expected result in terms of satisfaction will never exceed 1. In real life, the likely consequences are that you will not be listened to anymore, you get overruled or you might even get replaced.

A strategic test professional will make different decisions over time, even if he or she has the same information available. The difference is that a strategic test professional is aware of the possible consequences of his decisions. Because he makes conscious decisions, he is more likely to invest in techniques to increase the effectiveness of his decisions, rather than avoid a possible negative impact. Over time, these test professionals will be more effective. This also explains the fact that some test managers will make decisions which violate the agreements they have made earlier. From a strategy perspective, such a decision is explainable if a test manager realizes what the possible consequences may be and if it will increase the effectiveness of the decision itself or any future decisions.

Towards becoming a strategic test professional

By having read the principles from this article, you can start becoming a better test professional yourself. Perhaps you have ended up in an equilibrium state, according to game theory, which is stable, but not optimal. If your test reports are not being read by your boss, what is the sense of making them? And what will happen when you do have something important to put on the table? Once you understand the basic concepts of game theory, you should take a good look at the decisions you make in your daily practice and take purposeful actions. Will you stay the typical tester who is always right, but also rather risk-averse? Or are you willing to step up and add real value to your test projects?

Are you ready for some serious strategy? Game on!

> biography



IJsbrand Kaper

has over ten years of testing experience in several testing roles. After finishing his Executive MBA at Nyenrode in 2010, he stopped working as a test consultant and started to work as an operational quality manager for the business unit Test Services of VX Company IT Services. In this role he combines a technical management role with a commercial

role. This combination allows him to closely monitor the market and adapt the business unit's product portfolio to the market demand. Currently his areas of attention include app testing and SharePoint testing. Also, he is one of the teachers in Testfactor® training, which has been designed specifically to increase the effectiveness of testers. This summer, IJsbrand expects to become a father for the first time.

Testing IT and the **HASTQB**
united for your growth
by offering you the course:

Testing iT
Hunting Bugs... Opening Business



“ISTQB Certified Tester - Foundation Level”

Objectives:

- To ensure a full comprehension of key and fundamental concepts in Software Testing.
- To provide a foundation for professional development.

Syllabus:

- Testing Foundation, Testing Management, Approaches to Testing, Planning, Basic Performance Tests and Testing Tools.

Testing IT Consulting

“...There is always a better way of doing things, and we will find it...”

Testing IT University

“...Education and Experience is simply the soul of a Tester...”

Testing IT Units

“...TEAM = Together Everyone Achieves More...”

Hunting Bugs...
Opening Business

Information:

info@testingit.com.mx
mexico@hastqb.org
<http://www.testingit.com.mx>

+52 55 55566-3535

Paseo de la Reforma 107, int.102,
Col. Tabacalera, México, D.F., 06030

Do it like James!

Getting Certified Without training?

Mitigate the risk of failure

Online exam preparation for
Foundation and Advanced Levels
with Testing Experience & Learntesting

Products from € 20 to € 200

plus free Foundation self-assessment questions & answers

Exam Simulations, Exercise Videos, Exam Boot Camps

Buy your exam voucher and get some free exam preparation!

www.te-trainings-shop.com



“We unite competences”

Free expert meetings for priceless advantages

Software is omnipresent. Software is base technology. Software itself is value creation and innovation. Be it leisure time or working day: We will hardly find a space in our everyday lives, that aren't changed, improved or even controlled by software applications. We - consciously and unconsciously - rely on software doing what it is meant to be doing: We expect software quality.

Software quality increases efficiency and productiveness. Software quality defines the company image. There are many ways to high quality software: standardization, automation, re-useage, specialization. But in a complicated, quick, creative and sensitive „craftsmanship“ such as software development, personal and its specialist knowledge plays the most important part. Software quality does not stand still. Only with the combined efforts of specialist knowledge standardization, automation, re-useage and specialization a continuous improvement process can be conceived. The Arbeitskreis Software-Qualität und -Fortsbildung (Association for Software Quality and Further Education) – ASQF – has dedicated all its efforts to the idea of software quality – and has been doing so for 15 years now.

In these past 15 years the ASQF has developed into the leading competence network of the software development industry in

the German-speaking countries. Since 1996, when the ASQF originated in Erlangen through European Software Process Improvement of Training Initiative (ESPTI), the association is a prospering network with more than 1,000 members, 12 special interest groups and 32 regional groups even across the borders of the European Union. Members of the ASQF are global players such as Microsoft Deutschland GmbH, highly competitive medium-sized businesses like the SQS AG or the Díaz & Hilterscheid Unternehmensberatung GmbH, institutions like the German Central Bank – and, above all: committed experts, universities and research facilities. Although diverse in their fields of business, they are all united by their common goal to achieve high-quality standards in the information and communication technology.

Knowhow and Networking

The intensive contact between institutions, universities and national committees enables the ASQF as a networking platform to strengthen the exchange of experiences, innovative ideas and knowledge in the areas of the software development and quality management. This exchange is carried out in the specific fields as well as interdisciplinary by connecting software developers, product managers, project managers, quality assurance managers and scientists in regional and subject-related special interest groups in Germany, Austria and Switzerland.



The ASQF has its origins in a project sponsored by the European Union in the context of the European Software Process Improvement Training Initiative (ESPTI) in 1996. The ASQF is the sole shareholder of the iSQL GmbH – International Software Quality Institute – founded in 2004. iSQL GmbH developed certification standards for training in the field of software quality which recognized international and certified the know-how of IT experts world-wide. The ASQF and iSQL are headquartered in Erlangen and Potsdam.

www.asqf.de | info@asqf.de | Am Wechselgarten 19 | 91058 Erlangen | Germany



Networking at the ASQF Testing Day in Frankfurt

Currently no less than 12 special interest groups with their 32 regional sub-groups form the core of his work:

- Agility (new from 2012)
- Automation,
- Automotive,
- Maturity models,
- Medical technology,
- Modelling,
- Project management,
- Requirements Engineering,
- Safety,
- Software Test,
- Service Oriented Architecture (SOA) and
- User Assistance.

The ASQF actively creates knowledge-based communication platforms with more than 100 symposiums every year as well as high-profile conferences like the Automation Days, the Testing Days, the Medical Device Days or the Project Management Days – all free of charge. Renowned speakers discuss new trends, technologies and share their experience. At the same time, participants have the possibility to introduce their experiences and knowledge with an own speech. More than 3000 IT-Professionals and Experts attend the ASQF-Events every year. Longtime members are welcome as well as non-members: As a non-profit organization, the ASQF embraces everyone interested in Software Quality.

Career and Young Professionals

In our common goal for quality in the software development, one group is of particular importance: junior scientists and young professionals are the IT experts of tomorrow. Since 1999, graduates of five partner universities are awarded for exceptional work during their study courses, a short study period or a thesis that incorporates proximity to practical aspects and aspects of software quality. The prize is worth 500 €.

At the same time the ASQF offers chances for start-up businesses and the tense job market for skilled personnel: innovative start-ups can find the right business partners via the ASQF network and get tips for a good start into an entrepreneur's life. Committed graduates, young researchers and qualified employees in want of new challenges can find their new employer here. With the help of the new career portal ASQF quality ambassadors can easily find each other. The portal acts as a platform for job openings in the field of software quality. With the new service the ASQF does its bit to fight against the lack of specialized workforce in the IT industry. The main aim is to use the ASQF network to support the search for skilled employees thus close the gap between job seekers and job offers. Member companies of the ASQF can use the portal to make a specific search for new employees and know their job advertisement will be read by the target group from the IT industry. The job advertisements are open for everyone and can be used by non-members as well.



Anniversary and Award

All this and more is the ASQF since 1996. Thus the 15-year-jubilee of the ASQF is on the agenda this year. On 24th September 2011 the ASQF invites its members, partners and friends in the capital region to celebrate the successful story of the ASQF. The team around ASQF's general manager Stephan Goericke has prepared an exciting and comprehensive jubilee program: Jubilee guests can take part in an exclusive guided tour through the Federal Chancellery and gain new insights into the political processes in Germany. Sport enthusiasts can compete against prominent participants from the fields of sports, broadcasting and television, members of the EAGLES Charity Golf Club, and try to win the ASQF golf cup – titleholder is boxing champion Sven Ottke. The evening is dedicated to a big anniversary ceremony at the Van der Valk Hotel Berlin which will lead into the charity party „Indian Summer“. The focus of the evening, however, will lie on the award ceremony for the first ever „German prize for Software Quality“.



Stephan Goericke, Managing Director ASQF, hands over an ASQF award for exceptional work during study courses

The prize which was created in collaboration the ASQF subsidiary iSQL – International Software Quality Institute – will be awarded to people, companies, initiatives or institutions that have made an outstanding contribution to the preservation, advancement or research in the field of the software quality. Half of the „German price for Software Quality“ is sponsored by Association for Software Quality and Further Education, the other half by the International Software Quality Institute. It is supported by renowned industry members and is endowed with a sum of 5,000€. The prize will be awarded once a year. Proposals for future prize winners can be submitted by everyone sending a two-page letter of recommendation including references to the ASQF. The prize winner is nominated by an independent committee.

However, the ASQF primarily is and will always be a network of active people who are joined in the idea of championing quality standards in information and communication technology. Join us – and become a quality ambassador of the ASQF.

www.asqf.de | info@asqf.de

These are your chances

1. Come to the free ASQF special interest group meetings

IT experts give a dissertation about all varieties of software quality improvement and present new trends, standards and processes and focus on Project Management to the same extent as Usability or soft skills.

2. Find an apprenticeship or a new business partner

Use the special interest groups as a network to find companies which inspire you. As a student you will find interesting apprenticeships and as an employer you will find new business partners or customers.

3. Become an expert/ Be an expert

Are you alumni, an expert in Project Management or an employer? You can share your knowledge with the special interest groups and found or confirm your own expert status.

As a member of ASQF you enjoy these benefits:

- You have exclusive access to the collective expert knowledge from all expert groups
- You get discount for all conferences and exams which are organized by iSQL GmbH and you will get benefits for many other events in which ASQF and iSQL are involved
- A subscription for the SQ-Magazine and the opportunity to publish own contributions
- A 25 % discount for the subscription of the magazine “IT- Governance”, “it-service-Management” and “HMD- Praxis der Wirtschaft” (dpunkt.verlag)
- Access to the career portal

Do you feel confident? Than become a member of the ASQF and take home a free additional 1-year subscription for the magazine “Testing Experience” and take part in the “1.111 ASQF Members” draw, too.

Membership fee (plus VAT):

Membership fee as an student or trainee: 25,00 €

Regular membership fee: 50,00 €

Membership for companies: from 100,00€ (start-ups) to 1.000,00 € (big company)



Belgium Testing Days

March 12–14, 2012
Brussels, Belgium

www.belgiumtestingdays.com

**March 12-14, 2012
in Brussels, Belgium**

QA versus Testing! Antagonism or Symbiosis?

is next year's theme of the **Belgium Testing Days** taking place from March 12-14, 2012 in the Sheraton Brussels Airport Hotel in Brussels, Belgium. The Belgium Testing Days is an annual European conference for and by both national and international professionals involved in the world of Software Testing. Learn from experts and many others who are passionate about Testing during 3 days of talks, learning and discussion and use networking opportunities with your peers and industry experts.

**Belgium Testing Days 2012 –
A Díaz & Hilterscheid Conference
Endorsed by AQIS**

info@belgiumtestingdays.com
www.belgiumtestingdays.com

Follow us on Twitter: www.twitter.com/BelgiumTD

Register Now!
Catch the Early Bird discount and save 15% of the regular price!

Tutorials – March 12, 2012



"Assessments and how to perform them"

Johanna Rothman
09:00 - 17:30



"Mobile Testing"

Karen N. Johnson
09:00 - 17:30



"Making Test Automation Work in Agile Projects"

Lisa Crispin
09:00 - 17:30



"Essential Software Requirements"

Lee Copeland
09:00 - 17:30



"Test Estimation, Monitoring and Control"

Lloyd Roden
09:00 - 17:30

All tutorials include lunch and coffee breaks.



Conference (Day 1) – March 13, 2012

Time	Galaxy 1	Galaxy 2	Galaxy 3	Workshops -Atrium	Sponsor Tracks
08:00-09:00	Registration				
09:00-09:15	Conference Opening				
09:15-10:05	Keynote Goranka Bjedov: "The Future of Quality"				
10:10-11:00	Kris Laes: "When performance testing meets Business people"	Maarten Van Eyken: ",QA'-gile: black-box on a white-board"	Johan Jonasson: "Don't Mislead Your Stakeholders (Even If They Ask You To)"	Dorothy Graham & Mark Fewster: "Test Automation Clinic" - Part 1 -	Sponsor Presenter
11:00-11:25	Coffee Break				
11:30-12:20	Michel Kalis: "Perfomance Testing Case Studies"	Henrik Andersson: "Hi, are you stuck in an agile project?"	Susan Windsor: "How to deliver value from Test Assurance"	Dorothy Graham & Mark Fewster: "Test Automation Clinic" - Part 2 -	Sponsor Presenter
12:20-13:50	Lunch				
13:50-14:40	Sajjad Malang & Catherine Decrocq: "ATDD with Robot framework done right"	Elalami Lafkikh: "Testing Serendipity Testing: The Art of Increasing Defect Detection Likelihood"	Niels Malotaux: "Quality Comes Not From Testing"	Dawn Haynes: "The Search for Software Robustness" - Part 1 -	Sponsor Presenter
14:50-15:40	Wim Demey: "Knock-knock-knockin' on infrastructure's doors"	Matthew G. Sullivan: "Would You Enjoy Reading Your Own Test Reports?"	Jean-Paul Varwijk: "Regulations – Where Quality Assurance meets Testing"	Dawn Haynes: "The Search for Software Robustness" - Part 2 -	Sponsor Presenter
15:50-16:20	Coffee Break				
16:20-17:10	Keynote Johanna Rothman: "QA or Test? Does it Matter? You Bet it Does!"				
17:15-18:10	Lightning Talks Speakers: Lisa Crispin, Scott Barber, Dawn Haynes, Dorothy Graham, Susan Windsor, Lee Copeland				
18:10-18:20/ 18:30-19:20	Cocktail / Improvisation act				
19:20- 22:30	Dinner & Chill Out				

Want to exhibit?

If you'd like to be an exhibitor at Belgium Testing Days 2012, please fill in the form which you can find in our exhibitor brochure and fax it to +49 (0)30 74 76 28-99 or e-mail it to info@belgiumtestingdays.com.

» Download our exhibitor brochure at www.belgiumtestingdays.com

Exhibitors & Supporters 2012



Conference (Day 2) – March 14, 2012

Time	Galaxy 1	Galaxy 2	Galaxy 3	Workshops -Atrium	Sponsor Tracks
08:00-09:00			Registration		
09:00-09:05			Conference Infos		
09:05-09:55			Keynote Karen N. Johnson: "Why it matters what I'm called: Quality Analyst or Software Tester"		
10:05-10:55	Scott Barber: "Applying Educational Assessment to Testing"	Gilles Mantel: "Test automation: the return on investment myth"	Dries Baert: "Offshore tester: Friend or Foe?"	Eveliina Vuolli & Kirsia Korhonen: "Solving practical problems with the quality assurance in the large scale" - Part 1 -	Sponsor Presenter
10:55-11.25			Coffee Break		
11:30-12:20	Peter Morgan: "Planning your career to stay testing into the sunset ..."	Michael Palotas & Dominik Dary: "Test Automation – 10 (sometimes painful) Lessons Learned"	George Wilkinson: "Creating balance as a tester in modern times"	Eveliina Vuolli & Kirsia Korhonen: "Solving practical problems with the quality assurance in the large scale" - Part 2 -	Sponsor Presenter
12:20-13:50			Lunch		
13:50-14:40	Raja Bavani: "Distributed Agile: The Need for QA Mindset in Agile Testing Teams"	Graham Thomas: "Test Process Improvement – Answering the BIG questions!"	Stefaan Luckermans: "Janssen of Janssens, Thomson or Thompson, Dupont ou Dupond?"	Goranka Bjedov: "Advanced Hands-on Performance Testing" - Part 1 -	Sponsor Presenter
14:40-15:30	Björn Vanhove: "Think out of the box with ADQA"	Rik Marselis: "Governance: Controlling quality like a film director"	Adrian Rapan & Tony Bruce: "A Tale Of Two Cities"	Goranka Bjedov: "Advanced Hands-on Performance Testing" - Part 2 -	Sponsor Presenter
15:30-16:00			Coffee Break		
16:00-16:50	Alfonso Nocelli: "Open Source or not open Source that is the Question"	Sigge Birgisson: "Moving the project forward - perform testing and avoid the QA"	Zeger van Hese: "Artfull Testing"		
17:00-17:50			Keynote Lloyd Roden: "Dispelling Testing's Top Ten Myths and Illusions"		
17:50-18:00			Closing Session		

Note that the program is subject to change. Please visit our website for the current program.

Díaz & Hilterscheid Unternehmensberatung GmbH

Kurfürstendamm 179
 10707 Berlin (Germany)
 Phone: +49 (0)30 74 76 28-0
 Fax: +49 (0)30 74 76 28-99
www.diazhilterscheid.de

AQIS bvba

Uilstraat 76
 3300 Sint-Margriete-Houtem (Belgium)
 Phone: +32 16 777420
 Fax: +32 16 771851
www.aqis.eu

Subscribe for the printed issue!



Please fax this form to +49 (0)30 74 76 28 99, send an e-mail to info@testingexperience.com or subscribe at www.testingexperience-shop.com:

Billing Address

Company: _____
VAT ID: _____
First Name: _____
Last Name: _____
Street: _____
Post Code: _____
City, State: _____
Country: _____
Phone/Fax: _____
E-mail: _____

Delivery Address (if differs from the one above)

Company: _____
First Name: _____
Last Name: _____
Street: _____
Post Code: _____
City, State: _____
Country: _____
Phone/Fax: _____
E-mail: _____
Remarks: _____

1 year subscription

32,- €
(plus VAT & shipping)

2 years subscription

60,- €
(plus VAT & shipping)

Date

Signature, Company Stamp



Making the move into Agile testing

by Gerry Coleman

In a fast changing world, organizations have to be agile and flexible to remain competitive. The organization featured here, a financial multinational - 'Munnycorp' - is no different. A couple of years ago, Munnycorp opened a new software development office in Ireland, creating a new Product Development team. The main task for the new team was the rapid establishment of an effective software development operation. This article, based on research I carried out with Munnycorp's Irish team, describes the new Product Development team's experiences in adopting Agile development methods and supporting Agile testing approaches.

Where to start?

Though part of a multinational corporation, a newly established software team, in a separate country from its parent, has the feel of a large start-up company. In this regard, one of the first decisions faced by Munnycorp's new Irish Product Development team was which software development methodology to use. The parent organization used Scrum and were committed to Agile. From the experience of the parent organization, though not mandated, and previous knowledge gained working elsewhere, the newly-formed Product Development team also believed that Scrum would best serve their requirements.

However, this presented many challenges. The majority of the team, in fact all of the testers, came from non-Agile backgrounds, which had the potential to be an immediate impediment.

It has been said that few organizations use Scrum "out of the box". Munnycorp's Irish team was no different. They opted for a local interpretation of Scrum with a number of customizations. The main tenets they would keep; Sprints, daily stand-ups, story-based development, retrospectives and whole team approach but, during the early months, they tried and tested various customizations. These included story points, ideal hours, breaking stories into tasks, having development and test tasks and stories, having development and test estimates on the same tasks and stories, burn up, burn down, testing working half a sprint behind/a sprint behind, etc. They kept some and discarded others until eventually they found a team working agreement and a rhythm that suited.

Initially, there was concern that this customized Agile approach would be detrimental as they were not fully Agile. 'How could this

work?' However, through time via blogs, forums and conferences, it became increasingly apparent that they were 'agile lite' and that was ok as, essentially, so was everyone else.

Within these development constructs, testing and the role of testers is wholly bound up with the implementation of the methodologies. The team's role was to establish the development and testing process quickly whilst simultaneously recruiting and training the staff that would enable them to achieve this goal.

Getting the basics right

Whilst they were a newly developed lab, geographically dispersed from the other main lab, and with a remit and opportunity to challenge, there was a legacy to deal with. A good deal of the testing was more of a 'checking' nature, and there was a largely undocumented test process. There was also a lack of transparency regarding the testing scope/coverage and an inconsistent approach to test case management. This was compounded by a sense of 'we do it like this because we always have' which was holding the team back. While this may have worked in the past with a relatively small, collocated team, with the introduction of an additional lab in a different country and time zone, this was not going to be sustainable.

After an initial period of learning, study and reflection (all whilst working on an active project to go live within 3 months), the Irish team worked with the other international sites to put in place a lightweight testing approach based on a principle of simplicity and logic.

The team (testers & programmers) sitting together in one area was the initial step. Although a generic step, this benefitted the test team greatly, as it immediately quashed the concept of barriers between the programming and test teams. They were **one development team**, sitting together, from Product Owner to Scrum Master to Programmers to Testers. This constant communication improved solidarity and created a bond which immediately made the test team feel involved from the outset.

Furthering the testing approach

The Product Development team undertook the initial testing approach in two simple phases.

- Consolidation and education
- Improvements to their process to introduce a more coherent approach.

Initially, they took the position of understanding the existing testing process from a strategic and execution perspective. As they were also working on a new project with firm delivery dates, it made sense to absorb the majority of the existing process to get them to shipping the product. In a nutshell, they found that the existing testing process was detailed yet unstructured and, at times, inconsistent from project to project and release to release.

One example of this was in the regression approach, which was to run thousands of scripts again and again. These scripts were well written, yet had little high level traceability to what was actually being tested. Having spent some time aligning themselves with the existing process, they set about implementing incremental improvements for themselves.

From the outset, they believed in and implemented, as Crispin and Gregory put it, the ‘power of 3’. Essentially, all decisions related to the project had to involve the test team (along with the Product Owner and Programmers). They found this hugely beneficial as it ensured complete trust and understanding between the teams and made the test team feel very valued from the outset (something the majority of the test team had never felt before in previous companies and methodologies). This prompted the test team to really respond to the challenge and helped their settling into this new environment.

As mentioned earlier, they experimented with a few things including the testing being done at the end of the sprint, and a sprint behind. Initially, the testers argued, “I can’t write any tests, as I don’t know what the product is yet”. This was tried early on whilst they found their feet and as the test team were most comfortable with this approach initially (based on previous experiences). In a way they were working in ‘mini-waterfall’ for a few sprints. However, as they became more comfortable with the move to Agile, and confidence grew in their new world, they abandoned these approaches. They simply had to create a new mindset and move away from a focus on a fully-tested release, to testing the limited functionality that would be received from the developers on day 2 or 3 of a sprint.

They convinced the testers that, by testing what you get from the developers at this early stage, you’re learning about the product to such an extent that you can challenge it and ask why it functions this way, or why this particular feature is absent. In addition, the testers gained more control over the build and deploy process. The testers essentially specified when they wanted the build, thus deciding by the stories in the ‘Ready for Validation’ phase what was in the build and therefore what testing they would run. This allowed testers to stay on top of the project feature testing and control the progress on a regular basis.

They also experimented with how to break up stories and tasks. They went with separate test stories, and tasks, for programming stories and tasks. This was used for a while and then they used the same stories and tasks as the programmers but with programming and test estimates. They currently have settled on separate test stories, but not necessarily with the same names or descriptions as the programming equivalents, given that it was not pragmatic to explicitly test every programming change and quite often numerous programming changes made up a single

testable unit. Again, there was nothing scientific in this, it essentially evolved over time and as they found their way and what suited the teams.

In terms of how the sprint works, testers are involved from day 1. Their work starts with the Test team collaborating with the Product Owner/Development Manager and programming team to flesh out user stories. This allows the test team to then pull together their test cases. Once the initial draft is done, the test team review with the programming team asking two specific questions:

1. ‘In your opinion, is there anything I am missing based on your understanding of the code and requirement?’
2. ‘Is there anything in my test cases, you feel is not required?’

Once agreement is finalized, the test cases become official and are signed off as a team. At this stage, they are no longer “the test team’s tests”, they are the “team’s tests”.

The test team will then automate (where possible) the test cases by putting together a structured data file and storing it in the version control repository. The test cases are then entered into the test case management tool, which is also their Agile planning tool. They have a separate test automation team. Their automation is set up to enable you to point at the structured data file and run the tests, because they’re common classes and methods that the automation team have written in the background. This process (marrying independence then collaboration between teams) allows the testers to work in isolation from the programming team. However, because they use the concept of whole team ownership, all of the team sign off on the tests.

Optimization

The more the Product Development team grew, the more they wanted to try new things and improve the existing test process across the organization. These changes needed to be small at first in order to gain buy-in and acceptance from the existing teams. They decided to focus on traceability as a firm starter, to try and introduce a sense of coverage and understanding of what they test.

The company had previously invested in an Agile test case management tool and, as they had started using it during their first project, the implementation across other teams was more of an education process. They implemented the correct usage of a structured test planning module and creation of test cases per project and feature. They then further introduced the concept of test iterations so as to run test cases, record results in real-time, and put in place an audit of Test execution. This may appear small, but every journey starts with a single step and this was a fundamental step in establishing change.

Another notable improvement was to the release process and the regression testing effort. They did this by encouraging the sharing of these tasks and detailing how they are done. This was met with great acceptance by the other sites, and a sense of team bonding emerged. However, to really shake up the testing in this Agile space, they proposed a ‘common test approach’. With API testing central to their activity, they worked over time to develop a common feature test set. This was then made into a template, using structured data files, and all tests became automated making the testing process smoother. Their long-term goal for this initiative is to optimize this ‘checking’ activity to such an extent

that more space is created for exploratory testing, which they believe has greater benefit to the organization and enables the testers to use their experience in a more effective way.

One of the most important aspects the Irish product development team wanted to introduce was a change in mindset. Beforehand there was a concept that testing=automation. In any company, this is simply incorrect; there are many excellent tests that are manual and these may have been overlooked when there was a drive for 100% automation. Automation is a vital and essential tool, especially in an Agile environment where testing needs to be repeatable and thorough, but it is not the ‘be all and end all’ when it comes to testing. They now still have the vast majority of their tests automated but, as it should be, it is not the first criteria when they approach testing anymore.

Interfaces to the test team

Munnycorp’s Irish team worked out early on that the success of the test team (or any team in any organization) is not solely controlled by that team. Many different types of teams interface with each other and depend on each other to get things done and ship releases of code. As a newly developed lab, they had to reach out early and quickly to these teams to establish relationships and learn processes quickly. They had the inevitable and in-built relationship with the onsite programming team; that was a given. However, as Munnycorp were multi-site, other programming teams could be making changes that may affect their testing. The Irish team, therefore, had to familiarize themselves with programming teams throughout the organization. This was done by picking up the phone, setting up and attending introduction meetings, and finding out as much about the teams as possible.

Another important team they had to interface with was, as mentioned earlier, the automation team. They had to quickly understand the automation currently there and how the team works. This team has its own project backlog based on requests from the various test teams to add functionality to the automation framework.

They also had to reach out to the infrastructure team, configuration management, release management, and database teams. Understanding the teams, their place in the organization and the organization’s ‘way of doing things’ became an integral part of the newly formed lab’s survival. Every team affected the test team more than most, as they essentially need all of these teams for everything from test environments, test set-ups, knowledge of affected areas to test, database set up and schemas, etc. Establishing these relationships was essential to honing an improved test process.

The future

Munnycorp’s Irish team are now 2.5 years into this venture and have made much progress on their journey; a journey involving a mindset change, technical challenges, process improvements and most importantly, personal development. Every tester has grown into a fully-fledged tester during this time. They have become more rounded in their skilling and evolved from UI automators and ‘checkers’. But with every organic process there is a journey of evolution. So what’s next?

The need to remain dynamic and agile is a key feature for the Irish operation, as the company has expanded rapidly in the last 2 years. The Irish team would also like to expand their performance testing regime which, though adequate at present, may be chal-

lenged if business continues to expand at its present rate.

They also want to build on their improvements to date. Some of their initiatives include their build and deployment testing, moving the existing ‘tests’ to ‘automated checks’, their retirement of legacy tests whilst not losing coverage (by migrating to a more traceable feature and services matrix), improvement and enhancement of the automation framework, as well as the continuation of improvements mentioned above.

The overall intention is to make the ‘checking’ process, as Michael Bolton puts it, an everyday habit that goes on in the background, only creating attention when these checks fail. This saved time will allow the testers to further hone their newly acquired testing skills to properly explore the features and services.

The resistance to change from other lanes and sites is diminishing, as the Irish team have shown improvements working locally. There is now an appetite, from the wider organization, to adopt their changes and there are offers to work with them in the next initiatives.

Overall, the Irish team are happy with how their Agile implementation is working. Staff turnover is well below the industry average, and they have focused on recruiting people who want to work in a fast-changing Agile environment. They are optimistic that they have the capacity to adapt to future business changes and that the appropriate development building blocks are in place to enable them to go forward with confidence. There is little doubt that, on their testing journey, they are excited by the road of learning, improvement and opportunity that lies ahead.

> biography



Gerry Coleman

is a member of the Computing and Mathematics Department at Dundalk Institute of Technology (DkIT) where he specializes in teaching Software Engineering. He is Director of the Software Technology Research Centre (SToRC) in DkIT and is a Director of the Irish Software Testing Board. He has over 25 years experience in IT as a practitioner, academic and researcher and received his PhD from Dublin City University for research into Software Process Improvement in the indigenous Irish software industry. Prior to taking up an academic position, he worked for a number of years as a Software Developer, Systems/Business Analyst and Project Manager in the software development departments of a number of financial institutions and was a Senior Consultant, with responsibility for research, and technology training and transfer, in the Centre for Software Engineering at Dublin City University. He is on the programme committee for a number of international conferences and is a regular reviewer for the Journal of Systems and Software and the Journal of Software Maintenance and Evolution: Research and Practice.

Already Certified?

Join the Alumni Scheme and keep your knowledge up to date

Aimed at individuals and business, the Alumni Scheme provides those already certified the opportunity to gain ongoing access to latest versions of course materials plus many other benefits for a nominal annual subscription.

- 50 e-books on Testing and related IT
- Latest version of certificated courses
- Regular testing content updates
- Discounts on Learntesting products
- 12 months 24x7 access
- For Individuals and Business
- The Learntesting Alumni Scheme

Supporting professionalization of the Software Testing Industry

The Learntesting Alumni scheme – lifelong learning and support for Continual Professional Development (CPD) At Learntesting, we take seriously the continual professional development of software testers. After all, if we want to be recognised as a profession, we should start behaving like one! In October 2010, the new release of the ISTQB Certified Tester Foundation Syllabus introduced a 'Code of Ethics' which includes:

"Software Testers shall participate in lifelong learning regarding the practice of their Profession"

There is no concept of 're-certification' in the ISTQB scheme at Foundation or Advanced Level.

The fact that 150,000 individuals have been certified against many different versions of the syllabi over the past 13 years, means that one of the fundamental principles the scheme supports, 'A common language for testing', is compromised. Anyone certifying against the current Foundation syllabus can now be speaking a very different language to someone from several years ago – even if they could all remember it all!

At Learntesting, we have devised an innovative scheme that supports the new ISTQB 'Code of Ethics' and the concept of CPD for Testers. The scheme allows anyone already certified to have ongoing access to the very latest materials plus a whole range of other benefits – all for a nominal sum.

www.learntesting.com





Test management issues for systems of systems

by Patrick Hendrickx & Chris Van Bael

What are systems of systems?

Systems of systems are conglomerates of interconnected systems that work together to achieve a common purpose.

While virtually every system is connected to others, systems of systems are distinguished by the following characteristics:

- Every component system is fully operational and has a purpose of its own.
- The component systems are operated independently of each other, often by different organizations or organizational units.
- Systems of systems are not built in one fell swoop. They are gradually built by connecting existing systems, adding new features, etc.
- A system of systems performs functions and carries out tasks that are not in any component system.
- The component systems are geographically spread.

Example: In the energy sector, the systems of the electricity and gas providers are connected with those of the distribution network operators, the metering companies, etc., effectively forming a system of systems.

This system of systems implements market processes such as:

- the exchange of information regarding energy consumption – so customers can be billed for the energy they use, and
- switches from energy suppliers.

Example: The payment system's infrastructure used to settle financial transactions and transfer funds between banks is a system of systems.

Impact on test strategy

Test management is complex because the component systems are geographically spread, belong to different organizations and are developed using different life cycle models.

For these reasons, the master test plan for the systems of systems typically implements a formal life cycle model with emphasis on management issues such as milestones and quality gates.

Supporting activities such as test incident management must also be rigorously (re)defined because every participating organization will already have its own standards and tools.

If costs are to be distributed across the different organizations, the test manager will be called upon to produce reliable effort estimates.

Advantages

- Testing a system of systems end to end is more representative than testing every system in isolation with the help of simulators or data files generated by the other systems. It may, in fact, be the only way to verify whether the overall process will work or not.
- Testing can be seen as a dress rehearsal that evaluates operational readiness just as much as technical readiness.

Disadvantages and limitations

Tests are very complex because of the complexity of the component systems and interfaces and because the processes span multiple problem domains, multiple organizations with different cultures, and multiple locations.

Designing tests is very difficult because knowledge is typically spread among many experts from different backgrounds. Given the evolutionary nature by which systems of systems are built, documentation may be incomplete or non-existent.

Impact on planning

The component systems are developed using different life cycle models and this complicates test planning. The test schedule will have to include formally defined milestones to synchronize activities.

Since every component system has a purpose in its own right, there will be development activities that are not related to the systems of systems' behavior. These activities may be in conflict with the test planning: they may consume test resources, they may block the test environment, etc.

Impact on coverage

Testing systems of systems is very expensive because of all the

Sie suchen
das Besondere?



Díaz Hilterscheid

preparation and coordination overheads. Realistic coverage goals should be set to keep the cost of testing under control.

Example: During a test project in the energy sector, involving four systems operated by four different organizations, we arrived at a cost of €1000 to €1500 per test case, depending on the length of the test procedure.

There may be different component systems that perform the same task. This should not be overlooked when determining the necessary test coverage.

Example: When testing processes in the electricity and gas market, there may be multiple distribution network operators participating. These operators may use different systems and/or interfaces that must be separately tested.

Impact on defect management

Defect management must be rigorously defined because every organization will already have its own defect management process and tools in place.

Assigning ownership of defects can be very difficult because of the complexity of the system and the tests make it hard to determine the root cause of the problem.

Many test case failures will be caused by inconsistencies in the test data or by tester errors. Ensure that they can be flagged as such, so they do not bias defect metrics.

Impact on test data selection

Test data preparation can be a real challenge when dealing with systems of systems:

- The test environments of the component systems must be interconnected.
- Test data must be compatible in all environments.

Example: Many test environments are initialized with subsets of production data. For system of systems testing, all component systems must refresh their test data on the same date and with the same subset – for instance the same region.

- The component systems will use different coding schemes internally, which complicates test data selection.
- The test environments of the component systems will not be used exclusively for system of systems testing. There may be conflicts with other projects, release calendars, test data refreshes, etc.

Monitoring and control

Test monitoring is not fundamentally different, but the complexity of the tests will make it more difficult to pinpoint bottlenecks or to determine the root causes of failures.

The organizational complexity will make it more difficult to decide on control actions and to implement them.

Since test reports are sent to multiple organizations, the test manager should pay extra attention to objectiveness and political issues.

> biography



Patrick Hendrickx

has spent his entire working life in the offices and the trenches of the IT industry. Prior to becoming a software tester, Patrick worked for several multinationals as a programmer, on-site troubleshooter and software development manager. He has delivered software-intensive systems to a multitude of customers, primarily in the automotive, aerospace and defense industries.

Patrick became a software testing professional in 2002 and since then he has divided his time between test management, training and coaching. Currently he is Field Manager @ ps_testware to assure service delivery towards customers and assure career and competence towards consultants. He holds the ISEB Practitioner Certificate in Software Testing, is a speaker at conferences and author of the book 'Advanced Test Management' (isbn: 978-90-9025727-3).



Chris Van Bael

has been testing software systems for 10 years. He was one of the first Belgian testers to receive the ISEB Practitioner Certificate in Software Testing in 2004.

As a test automation and performance test consultant @ ps_testware he has worked with embedded systems, telecommunication systems and various large-scale web applications.

Currently he is secretary of the ISTQB national board for Belgium and The Netherlands, the BNTQB. He is also a member of the ISTQB Expert Level in the Test Automation working party and author of the book 'Advanced Test Management' (isbn: 978-90-9025727-3).

Wir auch!

Lassen Sie sich anstecken von der kollegialen Arbeitsatmosphäre in einem starken und motivierten Team.
Zum nächstmöglichen Termin stellen wir ein:

Senior Consultants
IT Management & Quality Services (m/w) für SAP-Anwendungen
Deutschland und Europa

Sie haben

- eine fundierte Ausbildung oder ein Studium (z. B. Informatik, BWL, Mathematik) sowie mehrjährige Berufserfahrung als IT-Consultant in einem Fachbereich bzw. in der Organisation eines SAP-Anwenders, eines IT-Dienstleisters oder in einem Beratungsunternehmen in entsprechenden Projekten
- ausgewiesene SAP-Kenntnisse (z. B. SAP ERP, SAP BI oder SAP Solution Manager)
- Erfahrung im Customizing und mit ABAP-Programmierung
- Kenntnisse in der praktischen Anwendung von Methoden und Standards, wie CMMI®, SPICE, ITIL®, TPI®, TMMI®, IEEE, ISO 9126
- Erfahrung in der Führung von großen Teams (als Projektleiter, Teilprojektleiter, Testmanager)
- Vertriebserfahrung und Sie erkennen innovative Vertriebsansätze

Sie verfügen über

- Eigeninitiative und repräsentatives Auftreten
- eine hohe Reisebereitschaft
- gute Englischkenntnisse in Wort und Schrift

Dann sprechen Sie uns an – wir freuen uns auf Sie!

Bitte senden Sie Ihre aussagekräftige Online-Bewerbung an unseren Bereich Personal
(hr@diazhilterscheid.de). Ihre Fragen im Vorfeld beantworten wir gerne (+49 (0)30 74 76 28 0).

Díaz & Hilterscheid Unternehmensberatung GmbH
Kurfürstendamm 179, D-10707 Berlin
www.diazhilterscheid.com

Mehr Stellenangebote finden Sie auf unserer Webseite:

- Senior Consultants Financial Services (m/w)
- Senior Consultants IT Management & Quality Services (m/w)
- Senior Consultants IT Management & Quality Services (m/w)
für Agile Softwareentwicklung und Softwaretest
- Senior Consultants IT Management & Quality Services (m/w)
für unsere Kunden aus der Finanzwirtschaft

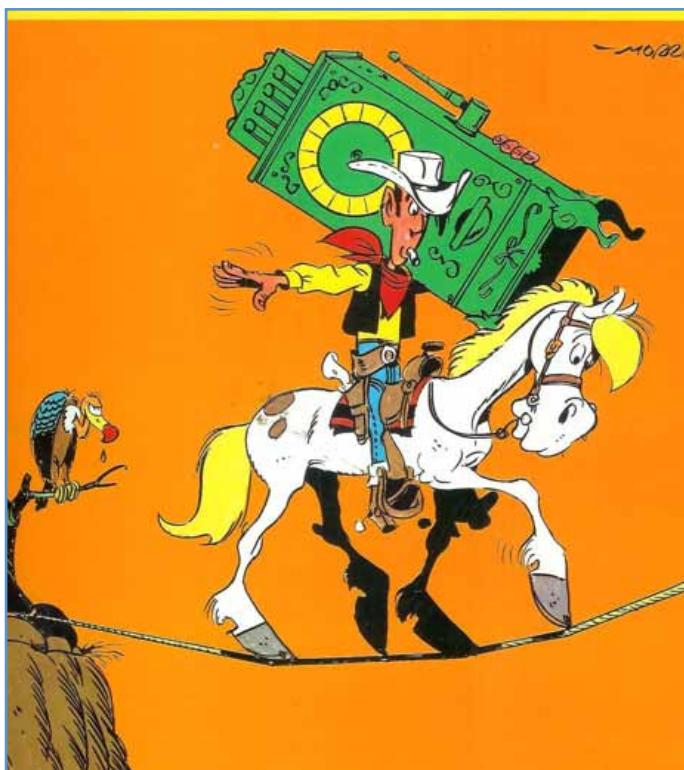


Díaz Hilterscheid

Las Vegas game testing

by Thomas Hijl

Dear readers, many thanks for all the interesting articles about "improving the test process". As stated in my last column, I was curious to see what would be written about this subject. I learned something, but also noticed that there were others having similar ideas about it. I enjoyed reading.



And for now,... something completely different.

"Play the game" first made me think of that beautiful Queen song. I'm a bit of a Queen fan, so that was obvious. However, I could not link "test the game" to the song, so I gave up writing my column about it. My second thought was – VEGAS –. The city of gaming! ... OR was it gambling?

I wondered: Is gambling considered gaming? Or is it a specific type of game? Or can you make any type of game into gambling by adding a financial aspect to it? Anyway,....

I started searching on the internet to see what I would get on "slot machine" + "test". I got a hit on the "Pennsylvania code", which linked me to the Pennsylvania Gaming Control Board. Michael Cruz is director of their gaming lab that tests slot machines. I was really interested to see what he and his organization do in terms of testing slot machines. The following link will get you to a short video presentation about it:

http://videos.cleveland.com/plain-dealer/2011/05/gaming_lab_tests_slot_machines.html

I found it interesting to see, not unexpected, that there are professional test organizations working in this domain. However, I couldn't find anything on the internet that looked like a slot machine test report. Is it still a closed world? Is there still vagueness around this area of business? What I also found was the following:

<http://www.americancasinoguide.com/slot-machines/are-slot-machines-honest.html>

Bottom line of this article: It seems that the gambling industry is still a bit of a "wild, wild west". Lucky Luke was the first to check whether gambling was fair play or not. Nowadays there are control boards, but are they shooting at their own shadows?

Last but not least, I could not find any standards like TMAP for example. I found manufacturing and certification standards, but no test standards. You might have a look at:

<http://www.gamingstandards.com/index.php?page=home>

I guess there are standards, although they might not have been written down or published. To keep the "near-miss" alive, maybe? That's a new interesting term in our profession, don't you think ...

> biography



Thomas Hijl

is acceptance manager and partner at Qwince BV. He started in 1996 as consultant / project manager for Philips Semiconductors (a.k.a. NXP) working on technology projects in the EMEA, APAC and AMEC regions. He switched to test management because of his ambition to improve project deliverables. He now serves clients with managing acceptance throughout the project lifecycle.

ISTQB® Certified Tester

Über 174.000 Certified Tester weltweit.
Wann gehören Sie dazu?

Manche Ausbildungen öffnen Wege, andere eine ganze Welt.

Die Schulung zum ISTQB® Certified Tester führt zu einem weltweit anerkannten und etablierten Zertifikat. Rund 17.700 Fachkräfte alleine in Deutschland haben es schon. Und Zehntausende in weiteren 47 Ländern - von A wie Amerika bis V wie Vietnam.

- Der ISTQB® Certified Tester ermöglicht Karrieren
 - mit ihm liegt erstmals ein internationales und branchenübergreifendes Berufsprofil für Software-Tester vor.
- Der ISTQB® Certified Tester macht die Arbeit leichter - Tester sprechen nun die gleiche Fachsprache, benutzen die gleichen Begriffe.
- Der ISTQB® Certified Tester hilft Kosten zu senken - Durch geschulte Tester werden Fehler bereits in frühen Phasen der SW-Entwicklung entdeckt.

Anmelden ist einfach.

Ein akkreditierter Trainingsanbieter ist sicher auch in Ihrer Nähe:

GTB Premium Partner

- | | |
|---------------------------------------|-----------------------------|
| ● ALTRAN GmbH & Co. KG | ● MaibornWolff et al GmbH |
| ● andagon GmbH | ● Method Park Software AG |
| ● berner & mattner Systemtechnik GmbH | ● Philotech GmbH |
| ● corporate quality consulting GmbH | ● qme Software GmbH |
| ● C1 SetCon GmbH | ● spp.med gmbh |
| ● EXCO GmbH | ● Software Quality Lab GmbH |
| ● imbus AG | ● Sogeti Deutschland GmbH |
| ● Knowledge Department GmbH & Co. KG | ● SQS AG |
| ● Logica Deutschland GmbH & Co. KG | ● T-Systems |

autorisierte Zertifizierungsstellen

- | |
|---|
| DLGI - Dienstleistungsgesellschaft für Informatik mbH |
| iSQI - International Software Quality Institute GmbH |

Alle Trainingsprovider siehe www.german-testing-board.info





Product testing in front of the crowd

by Derek Wallis

I have been a tester at Unum for a little over 10 years, nowadays specializing in automation. I am also a keen car photographer and it is through this that I found myself being given the opportunity to drive a unique \$500,000+ supercar during the famous Goodwood Festival of Speed in England.

Although not consciously thinking about my day job, it set me in good stead for what was about to happen, and looking back the parallels between testing and this drive are there.

We are all used to last minute requirements and this was no different. I was standing with my GTspirit.com editor half way up the famous Goodwood Festival of Speed hill climb, photographing the Le Mans racers as they screamed by when he gets a phone call. Fifteen minutes later we are no longer photographing the cars but are instead down at the Toyota/Lexus stand and there is the distinct chance of a change in job role coming up. I guess the next conversation can be likened to those meetings where everyone in a project gets together in a room and the subject matter experts wax lyrical about their area, or the IT guys talk technical and the rest of the room can only follow the conversation in bits. For me, this is because the editor and PR guy are both speaking fluent Dutch and I have no idea what is being said, although I do pick up the word camera. It transpires that I've been given the opportunity to drive one of their cars.

Product familiarization

As with all types of testing you need to know what is being tested. In this instance it is the Lexus LFA Nürburgring Edition package. This car is still in pre-production, but thankfully Lexus are nearing the end of the project lifecycle so everything works as expected and my role will be that of an end user. My mind automatically runs through a mental checklist for previous experiences that may help. This is quite easy because I passed my driving test many years back and coincidentally my daily drive is a Toyota.

Requirements

So I'm going to undertake product usability testing in what, at the highest level, is just a car and all that is required by me is to drive it up a 1.16 mile hill and back down again. Sounds simple when written like that, but time to go into more detail.

I know how to drive, but I still need to ascertain what is different between this supercar and my own car. Even before getting in I know of some differences, the most obvious being that the steering wheel is on the left, however it still functions the same as my own steering wheel - I turn it and the wheels move in the same direction, although it does mean I need to adjust my road position. This car is also a lot more powerful than my own, so I need to be mindful of that when pressing the accelerator. Finally, this car is a semi-automatic meaning the gear changes are taken care of using paddle shifts behind the steering wheel, meaning my left foot can relax for the journey. Some things are outside of our control, and I can easily adapt my driving to meet these changes in the same way a tester would need to raise or adapt to perceived problems when going through a requirements document.

Objectives

There are three main goals of usability testing: How easy is the product to be used, does it feel right and, most importantly of all, is it fit for purpose. A lot of people think testing is about breaking things, but if that was the case why is it called testing and not, for example, demolishing? My objective was certainly not to break anything, especially because when the car does go into production, it will cost around £400,000, but this particular one is the only one in the world at the moment.

There were four objectives for me, all of which can be traced back to the goals of testing:

- Drive the car up the hill and back without incident
- Ensure there was some good footage for the video cameras that would be attached (this will also provide evidence of the test)
- Show the product off - the public have come to see and hear the cars
- Have fun

If the car failed to meet any of the goals, then the drive (and product) could be classed as a failure.

Test environment

Two distinct test environments existed, the stable one within the

car and the outside environment which is subject to a number of external factors.

I start with evaluating the internal environment. There is a big red button on the center console and I must resist a tester's natural urge to press it and see what it does, especially as it has the word 'reset' on it. This button, along with the rest of the center console, is superfluous to my particular drive and as such can be considered out of scope. Instead I just need to concentrate on what I need to know in order to carry out the task. This version of the product comes equipped with a racing harness and seats plus a roll cage, making getting in a bit cumbersome but not impossible. The roll cage also means there are no sun visors, so a workaround of wearing sunglasses would be needed in bright conditions. Not having tested the previous version, I am unsure if that also suffers from having the bonnet bulge obstructing the end of the car, I'm almost 6' and am unable to see where the car ends, although I have a rough idea and don't intend on testing the boundaries of this. I suspect however that the seating position is due to the bucket seats and is another case of function over form (or safety over comfort). The only other thing to note that I had not picked up in the initial requirements is the electronic handbrake replacing a mechanical one.

The external environment will change during the course of execution. Currently, the car is parked in the 'supercar paddock' surrounded by other cars, and this is by no means the most expensive car present. There are also a few hundred members of the public milling around looking at and taking photographs of the cars. During execution my main concern will be the process of driving out of the paddock to the start line, the exact route of which I am unsure of, but there will be other cars to follow and marshals to guide me. I know the route from the start line to the hill as I had already driven up there twice yesterday, and so I am confident about what should be the most nervy part of the test, providing there is no rain.

Test preparation

I am handed the key by the official Lexus driver and he guides me through the start-up procedure and basic functionality (how to change gear and set it in sport mode). I run through this process a few times and also test the sound the car makes when I press the accelerator, to the appreciation of the crowd. Test execution is scheduled for tomorrow and I'm as prepared as can be, although to reduce the risk of failure I will wear a different pair of trainers as opposed to the basket-ball style ones I have on today.

Test execution

The test execution window was scheduled for 1pm but I had to be at the Toyota/Lexus stand at 12 in order to sign in. Now sporting a silver 'Drivers' wristband the pre-acceptance conditions had been met and it was time to get to the car. I meet my passenger/co-tester whom, despite working for Lexus, had never been in the car before. The marshals give us a 10 minute warning and we get ourselves strapped in and everything set up ready. Am I ready for this? Do I know how to work everything, including the video cameras? A brief refresh of everything and I'm ready to roll.

Phase 1

The purpose of this phase is to exit the paddock without hitting anything, including what is probably a crowd of at least a thousand who have come to watch the cars leave. This I mana-

ge with external help from the professional driver and marshals who keep the crowd under control. I only give the accelerator enough of a press to keep the car moving and follow the car in front. Never be afraid of asking for help from those around you in order to accomplish the task.

Phase 2

We do not make our way straight down to the start line, instead weaving around to another area where we have to wait for what transpires to be almost 30 minutes. This was an unexpected stage in the process and introduces the variable of -Do I leave the engine on or not? I follow the other cars and leave it running as we don't know how long we will be waiting, and this unplanned phase enables a bit of exploratory testing to be undertaken. This mainly involved having another look around the interior and my passenger demonstrating the basics of the onboard computer system. At the request of some onlookers I also tested to see how loud I could make it, along with a number of other cars in the vicinity, thus helping with the two objectives of showing the car off and having fun. If an opportunity presents itself, take it.

Phase 3

Now at the start line and there are thousands of people and cameras watching plus a current Formula 1 driver about 20 cars in front, but I block them all out and turn the video cameras on. Don't get distracted and just focus on the tasks in hand.

Phase 4

I'm at the line and it's time to start testing for real. I had managed to get a ride in another car that very morning which helped provide some reassurance about where to go. Final environment checks made (no rain, sport mode selected and in first gear) and acceptance testing can begin. The professional Lexus driver had reminded me as we exited the paddock that this was to be a demonstration run. Translated this meant a) don't break it and b) show it off to the crowd. With this in mind I try to keep the revs high to make the most noise and drive at a suitable speed according to my skills and the road. The speedometer is in KPH but I have little time to look at it, focusing on where to go, although I do hit 3rd gear along the straight. Somewhere in the vast crowd are my wife and daughter, but I'm not sure whereabouts. Soon we have reached the top without incident, all previous preparation having come in handy. I am unable to check the video footage without a computer and so hope it is ok. Work within your limits and don't crumble under pressure.

Phase 5

I still need to get the car back to the paddock but the journey back down the hill is a relaxed affair in convoy with the other supercars. We wind the windows down and give the crowd and marshals a wave and an occasional burst from the V10 engine. Acknowledge a job well done and where possible give that little bit extra.

Phase 6

Back at the paddock and I don't relish the prospect of reversing the car in the tight confines, but thankfully the professional driver takes over the steering leaving me to just work the pedals. This is good because although the screen in the center dash now shows

the view from the reversing camera, the bright sunshine means I can't see much, and I have no idea how much car is behind me. If there are experts around, use them.

Test closure

Engine off, cameras stopped and smiles all round, it's time to exit the car and hand it back to the manufacturer. I reflect on my first official foray into product testing which just happened to be in front of thousands of people. For once I am not dreading writing up the closure report.

> **biography**



Derek Wallis

originally trained in graphic design and advertising but started in testing when his name was put forward as a business user to help test a new policy admin system. Over 10 years later and having worked through a number of testing roles at the company he can now be found surrounded by many different machines as he runs and develops automated tests. He is both ISEB and HP certified.

IREB - Certified Professional for Requirements Engineering - Foundation Level



Díaz Hilterscheid

Description

The training is aimed at personnel mainly involved in the tasks of software requirements engineering. The tutorial is designed to transfer the knowledge needed to pass the examination to become an IREB CPRE-FL.

After earning a CPRE-FL certificate, a certificate holder can assess whether a given situation calls for requirements engineering. He understands the fundamental characteristics of the discipline and the interplay of methodological approaches, e.g. interview techniques, description tools or forms of documentation.

More information regarding the required knowledge can be found in the IREB Syllabus, which can be downloaded from the IREB web site: <http://www.certified-re.com>

<http://training.diazhilterscheid.com>



Dates*	3 days
14.-16.09.11	Stockhom/Sweden (en)
19.-21.09.11	Mödling/Austria (de)
05.-07.10.11	Mödling/Austria (de)
26.-28.10.11	Helsinki/Finland (en)
23.-25.11.11	Oslo/Norway (en)
28.-30.03.12	Berlin/Germany (de)
06.-08.06.12	Berlin/Germany (de)

*subject to modifications





Prof. van Testing empfiehlt:

DRANBLEIBEN!!!

Besuchen Sie einen der drei folgenden Kurse und sichern Sie sich die **kostenlose Verlängerung Ihres TE-Abonnements** in der Print-Version bis Ende 2012

CTAL-Technical Test Analyst	24.10. – 28.10.2011 in München
CTAL-Test Analyst	31.10. – 04.11.2011 in Berlin
CTAL-Test Manager	28.11. – 02.12.2011 in Berlin

zum Sonderpreis von **1890 €** zzgl. Mwst. plus ggfs. Prüfungsgebühr
anstatt netto **2100 €**.



Masthead



EDITOR

Díaz & Hilterscheid
Unternehmensberatung GmbH
Kurfürstendamm 179
10707 Berlin, Germany

Phone: +49 (0)30 74 76 28-0

Fax: +49 (0)30 74 76 28-99

E-Mail: info@diazhilterscheid.de

Díaz & Hilterscheid is a member of "Verband der Zeitschriftenverleger Berlin-Brandenburg e.V."

EDITORIAL

José Díaz

LAYOUT & DESIGN

Katrin Schülke

WEBSITE

www.testingexperience.com

ARTICLES & AUTHORS

editorial@testingexperience.com

350.000 readers

ADVERTISEMENTS

sales@testingexperience.com

SUBSCRIBE

www.testingexperience.com/subscribe.php

PRICE

online version: free of charge -> www.testingexperience.com
print version: 8,00 € (plus shipping) -> www.testingexperience-shop.com

ISSN 1866-5705

In all publications Díaz & Hilterscheid Unternehmensberatung GmbH makes every effort to respect the copyright of graphics and texts used, to make use of its own graphics and texts and to utilise public domain graphics and texts.

All brands and trademarks mentioned, where applicable, registered by third-parties are subject without restriction to the provisions of ruling labelling legislation and the rights of ownership of the registered owners. The mere mention of a trademark in no way allows the conclusion to be drawn that it is not protected by the rights of third parties.

The copyright for published material created by Díaz & Hilterscheid Unternehmensberatung GmbH remains the author's property. The duplication or use of such graphics or texts in other electronic or printed media is not permitted without the express consent of Díaz & Hilterscheid Unternehmensberatung GmbH.

The opinions expressed within the articles and contents herein do not necessarily express those of the publisher. Only the authors are responsible for the content of their articles.

No material in this publication may be reproduced in any form without permission. Reprints of individual articles available.

Picture Credits

© fat*fa*tin - Fotolia.com	2	© Katrin Schülke	22	© Orlando Florin Rosu - Fotolia.com	54
© volff - Fotolia.com	2	© Katrin Schülke	30	© RallyRider.co.uk	60
© fat*fa*tin - Fotolia.com	4	© Valery Sibrikov - Fotolia.com	34		
© iStockphoto.com/ScantyNebula	14	© Katrin Schülke	38		
© volff - Fotolia.com	18	© photocase.com/emanoo	50		

Index of Advertisers

Agile Testing Days 2011	9	QAustral S.A.	28
Belgium Testing Days 2012	45	Ranorex	3
CAT - Agile Certified Tester	2	Savignano Software Solutions	17
Díaz & Hilterscheid	17	Steria Mummert Consulting	7
Díaz & Hilterscheid	35	Telerik	19
Díaz & Hilterscheid	41	Testing Experience - Knowledge Transfer	8
Díaz & Hilterscheid	55	Testing Experience & Learntesting	53
Díaz & Hilterscheid	66	Testing & Finance 2012	25
German Testing Board	59	Testing IT	40
IREB Training	63	UNICOM Seminars Ltd.	20
iSQI	36		



12.09.11–15.09.11	Certified Tester Foundation Level	German	Munich/Germany
14.09.11–15.09.11	HP Quality Center - Neuerungen in Version 11	German	Berlin
14.09.11–16.09.11	Certified Professional for Requirements Engineering - Foundation Level	English	Stockholm/Sweden
15.09.11–16.09.11	Testen für Entwickler	German	Berlin
19.09.11–21.09.11	Certified Professional for Requirements Engineering - Foundation Level	German	Mödling/Austria
26.09.11–30.09.11	CAT - Certified Agile Tester	English	Mödling/Austria
27.09.11–29.09.11	HP QuickTest Professional	German	Berlin
29.09.11–30.09.11	PILOTSCHULUNG: Kanban Komplett	German	Berlin
05.10.11–07.10.11	Certified Professional for Requirements Engineering - Foundation Level	German	Mödling/Austria
10.10.11–12.10.11	Certified Tester Foundation Level - Kompaktkurs	German	Frankfurt/Germany
10.10.11–14.10.11	Certified Tester Advanced Level - TESTMANAGER	German	Mallorca/Spain
17.10.11–20.10.11	Certified Tester Foundation Level	German	Berlin
24.10.11–28.10.11	Certified Tester Advanced Level - TECHNICAL TEST ANALYST	German	Munich/Germany
24.10.11–28.10.11	CAT - Certified Agile Tester (deutscher Referent + deutsche Prüfung)	German	Cologne/Germany
25.10.11–26.10.11	Testmetriken im Testmanagement	German	Berlin
26.10.11–28.10.11	Certified Professional for Requirements Engineering - Foundation Level	English	Helsinki/Finland
31.10.11–04.11.11	Certified Tester Advanced Level - TEST ANALYST	German	Berlin
07.11.11–10.11.11	Certified Tester Foundation Level	German	Berlin
07.11.11–11.11.11	CAT - Certified Agile Tester	English	Mödling/Austria
07.11.11–11.11.11	CAT - Certified Agile Tester	English	Stockholm/Sweden
14.11.11–18.11.11	Certified Tester Advanced Level - TESTMANAGER	German	Frankfurt/Germany
14.11.11–17.11.11	Certified Tester Foundation Level	German	Mödling/Austria
21.11.11–23.11.11	ISEB Intermediate Certificate in Software Testing	German	Mödling/Austria
21.11.11–23.11.11	Certified Tester Foundation Level - Kompaktkurs	German	Munich/Germany
23.11.11–25.11.11	Certified Professional for Requirements Engineering - Foundation Level	English	Oslo/Norway

- subject to modifications -

more dates and onsite training worldwide in German, English, Spanish, French at <http://training.diazhilterscheid.com/>



Accredited
Training Organisation



International
Requirements
Engineering
Board

