



[Editorial]



Liebe Testcommunity!

Die Fußballeuropameisterschaft ist im vollen Gang und hat bisher schon viele interessante und überraschende Spiele geliefert. Auch wenn man die Leistung der österreichischen Mannschaft eher zu den negativen Überraschungen zählen muss, wurden wir durch die allgemeine Euphorie angespornt, wieder einen qualitativ hochwertigen und für alle Tester interessanten Insider zusammenzustellen.

Um das Thema Agile Methoden kommt man kaum herum, es gibt auch wieder 2 Artikel zu diesem Thema mit vielen Tipps (Seite 5, Seite 7).

Was mich freut ist, dass auch ein Thema, das eher selten in unseren Fachartikeln beleuchtet wird, nämlich Performancetest, gleich 2x vorkommt (Seite 9, Seite 15).

Mit sicherheitskritischen Systemen beschäftigt sich der Artikel ab Seite 11 sowie der letzte Vortrag im Rahmen des ASQF Fachgruppenabends (Seite 14).

Hinweisen möchte ich noch auf die erfüllten Kinderwünsche unseres letztjährigen Charity Events (Seite 10).

Ergänzt wird der Newsletter durch Vorschau und Rückschau auf diverse Events, Prüfungs-, Seminar- und Konferenztermine.

Ich hoffe, es ist für jeden etwas dabei. Ich wünsche allen einen schönen Sommer und allen Interessierten noch spannende EM-Fußballspiele.

Viel Spaß beim Lesen und Gut Test!

◀(Karl Kemminger)

[News aus dem ATB:
Hauptversammlung: Neuer Vorstand gewählt
& Launch der neuen ATB Homepage]



Folgende Ereignisse prägten das erste Halbjahr 2016 im ATB.

Am 18.Mai 2016 fand die **Hauptversammlung** statt, bei der auch der **neue Vorstand** gewählt wurde. Nähere Details dazu [hier](#). Die HV fand diesmal im Zentrum von Wien mit tollem Blick auf Michaelerplatz und Hofburg statt (siehe Foto) - herzlichen Dank an Certible, die diese tolle Location bereit gestellt haben!

Kurz danach ging die überarbeitete und **neu gestaltete ATB Homepage** in Betrieb, welche auch Unterstützung auf mobilen Geräten durch responsive Design bietet. Generell wurden die Texte überarbeitet und auf den aktuellsten Stand gebracht, die Ansprache auf höfliches Du umgestellt, und diverse Optimierungen durchgeführt. Besuchen Sie uns!

Weiters trafen sich die Präsidenten des D.A.CH. (deutschsprachige Boards aus Deutschland, Österreich und der Schweiz) zwecks verbesserter Koordinierung - darüber berichten wir dann in der nächsten Ausgabe!

◀(Helmut Pichler, Karl Kemminger)



[Inhalt]

- Editorial, News aus dem ATB Seite 1
- Goldpartner Seite 2 - 4
- Prüfungstermine Seite 4
- Agile Methoden Seite 5 - 8
- Performancetests Seite 9 - 10
- ATB Charity Seite 10
- Software Competence Center Hagenberg Seite 11 - 12
- SQS Quality Brunch Seite 13
- Neues aus dem ASQF e.V. Seite 14
- ATB Expertentreff Seite 15
- Trainingsprovider Seminare Seite 16 - 17
- Partner Veranstaltungen Seite 18
- Impressum Seite 18
- Konferenzen, Silberpartner Seite 19 - 20

SEQIS Education: Für Sie unser Trainingsangebot im Herbst!

Save-the-date:

*Termingarantie

12.09. - 16.09.2016*: iSQI® CAT Certified Agile Tester

10.10. - 11.10.2016*: iSQI® Certified Agile Business Analysis

17.10. - 19.10.2016*: iSQI® Certified Agile Test Driven Development

07.11. - 08.11.2016*: CMAP® Mobile App Testing, Foundation Level

17.11. - 18.11.2016*: ISTQB® Certified Tester, Foundation Level Extension, Agile Tester

Ihre Vorteile auf einen Blick:

- ✓ Garantierte Durchführung
- ✓ Unabhängig der Teilnehmerzahl
- ✓ Frühbucherrabatte

Machen Sie 2016 zu Ihrem Weiterbildungsjahr!



Für weitere Informationen kontaktieren Sie bitte: sales@SEQIS.com



www.SEQIS.com

IT Analyse. Software Test. Better Results.

[Facebook/SoftwareTestIsCool](https://www.facebook.com/SoftwareTestIsCool)

SQS Academy – Bundle-Angebot für Ihre Weiterbildung



sqs.com

Schauen Sie sich gerade nach einem Seminar um, dass Sie besuchen möchten, oder kennen Sie jemanden, der sich in diesem Jahr weiterbilden möchte?

Unser Bundle-Angebot: Buchen Sie 3 Seminare aus unserem Portfolio und zahlen Sie nur 2.

Buchen Sie Ihre Schulungen zwischen dem 01.07. und 31.08.2016 und Sie profitieren unter dem Stichwort

SQS-Bundle-2016 bis zum 31.12.2016

von unserem besonderen Angebot*.

* Bei unterschiedlichen Preisen erfolgt die Rabattierung auf das günstigste Seminar. Prüfungsgebühren sind von der Rabattierung ausgeschlossen. Rabattierung erfolgt auf den Listenpreis.



Kommende Seminartermine der SQS in Wien:

- ISTQB® Certified Tester: Advanced Level Test Manager, 18.-22.07.2016
- ISTQB® Certified Tester: Foundation Level, 12.-15.09.2016
- ISTQB® Certified Tester: Foundation Level, 03.-06.10.2016
- IREB CPRE Foundation Level, 10.-12.10.2016
- CMAP® Mobile App Testing Foundation Level, 11.-12.10.2016

Alle Seminare von A-Z finden Sie [hier](#)

Wir sind gerne für Sie da! Telefonisch: +43 (0) 1 319 35 23-13 oder per E-Mail: academy-austria@sqs.com



Das ANECON TA-Curriculum

Der berufsintegrierte, modulare Lehrgang für
professionelle Test Automation

[<Mehr Infos>](#)

Bezahlte Einschaltung



SOMMERAKTION

Tool-Schnuppertage:

- Requirements-Management-Tools 06.07.2016 Linz und 10.08.2016 Wien
- Testmanagement-Tools 16.08.2016 Linz
- Testautomatisierungs-Tools 30.08.2016 Linz und 02.08.2016 Wien
- Taskmanagement- und Bugtracking-Tools 12.07.2016 Linz und 23.08.2016 Wien

In jeder dieser Kategorien werden an einem Tag 4-8 unterschiedliche Tools vom Marktführer über Spezialtools bis hin zu Open-Source-Lösungen vorgestellt.

Im Zeitraum vom **01.07.2016 - 31.08.2016** können Sie dieses Leistungspaket zum sensationellen Preis **pro Person um EUR 180,-** buchen.

Inhouse Gruppen bis 6 Personen erhalten dieses Leistungspaket im genannten Zeitraum zum Pauschalpreis von **nur EUR 990,-** (statt Normalpreis EUR 2.800,-). Anfragen bitte an backoffice@software-quality-lab.com

Weitere Informationen zu unseren Seminaren finden Sie auf unserer [Homepage](#) oder in unserem [Seminarprogramm](#)

Bezahlte Einschaltung



If you can't see IT, you can't fix IT!

Get the Cube!

QACube drives competitive advantage with predictive visual analytics & dashboards across projects in popular tools such as HP ALM/Quality Center, Tricentis Tosca Testsuite, Rally Software, Jira and Excel to name a few.

- Automates Data Consolidation & Aggregation
- Provides Visibility to all Stakeholders
- Enables Transparency through the Right KPIs

www.qacube.com

**[Öffentliche ISTQB®
Prüfungstermine,
Certible]**

| | |
|------------|------|
| 08.07.2016 | Linz |
| 19.07.2016 | Wien |
| 05.08.2016 | Wien |
| 23.08.2016 | Wien |
| 09.09.2016 | Wien |
| 27.09.2016 | Wien |

Anmeldung und weitere Termine unter <https://www.certible.com/de/Kalender/>

◀ (Maria-Therese Teichmann)



**[Öffentliche ISTQB®
Prüfungstermine,
iSQI]**

Öffentliche Prüfungen in Wien finden zu folgenden Terminen statt:

- 29.07.2016
- 30.09.2016
- 21.10.2016
- 18.11.2016
- 02.12.2016

Anmeldung und weitere Termine finden Sie auf

<https://www.isqi.org/de/kalenderuebersicht.html>
www.isqi.org

◀ (Christin Senftleben)



**[Öffentliche ISTQB®
Prüfungstermine,
Gasq]**

Informationen über die Prüfungstermine findet man unter <http://de.gasq.org>

◀ (Karl Kemminger)



[10 Tipps und Tricks für den garantierten Erfolg von agilen Projekten]

von Mag. Margaret Grün-Kerr

Diverse Studien belegen es: Agile Projekte sind erfolgreicher als Projekte mit traditionellen Vorgehensweisen, wie beispielsweise dem Wasserfall-Modell. Demnach werden 39% aller agilen Projekte erfolgreich abgeschlossen, während es bei den Wasserfall-Projekten nur 11% sind. Bei jenen als klein eingestuften Projekten (bis fünf Projektmitarbeiter, Projektdauer bis max. drei Monate) werden sogar 58% aller agilen Projekte erfolgreich beendet.

Basierend auf den Studien der Standish Group, den Untersuchungen von Tsun Chow und Dac-Buu Cao sowie eigenen Erhebungen konnten Vorgehensweisen abgeleitet werden, die einen entscheidenden Einfluss auf den Erfolg von agilen Projekten haben. Als Ergebnis dieser Analysen wurden sechs Erfolgsfaktoren für agile Projekte definiert:

- **Agile Software-Entwicklungstechnik:** Das agile Manifest (www.agilemanifesto.org) beschreibt alle Werte, die in der agilen Softwareentwicklung befolgt werden sollen.
- **Kundenbeteiligung:** Eine gute Beziehung zum Kunden ermöglicht Kommunikation auf Augenhöhe. Der Kunde muss in den agilen Software Entwicklungsprozess eingebunden werden, um so das Commitment sicherzustellen.
- **Teamfähigkeit:** Ein gut funktionierendes Team ist die Basis für den Projekterfolg.
- **Teamumfeld:** Das gesamte Team sollte an einem Ort arbeiten, sich selbst verwalten und organisieren.
- **Delivery:** Regelmäßige Software-Lieferungen durch iteratives Vorgehen tragen zum Erfolg des Projektes bei – das Wichtigste wird dabei immer zuerst erledigt.
- **Projekt Management Prozess:** Die tägliche Kommunikation Face-to-Face sowie eine ausgeglichene Work-Life-Balance sind entscheidende Erfolgsfaktoren.

Ausgehend davon wurden in weiterer Folge 10 Praxistipps abgeleitet, die ebenfalls zur Erfolgssteigerung von agilen Projekten beitragen:

1. **Machen Sie die Projekte so klein wie möglich!**

In den oben erwähnten Studien konnte eindeutig nachgewiesen werden, dass kleine Projekte (sowohl agile als auch traditionelle) erfolgreicher sind als große. Deshalb sollten Projekte stets so klein wie möglich gehalten werden, um deren Erfolg zu garantieren – besser ist es, größere Vorhaben in Serien zu realisieren.

2. **Setzen Sie agile Methoden ein!**

Im Vergleich zu Projekten mit traditionellen Vorgehensweisen sind agile Projekte deutlich erfolgreicher. Agile Methoden sind somit ein entscheidender Erfolgsfaktor.

3. **Fördern, fordern und stellen Sie sicher, dass das Team für den Kunden arbeitet!**

Kundenentscheidungen haben höchste Priorität. Deshalb sollten die Interessen der Kunden stets im Mittelpunkt stehen und das Team immer im Sinne des Kunden entscheiden.

4. **Nutzen und fördern Sie das Wissen und die Power des Teams!**

Teammitglieder sind Experten! Sie besitzen hohe Fachkompetenz und -expertise in ihrem Arbeitsbereich, sie verfügen über agile Kenntnisse und hohe Motivation. Nutzen Sie dieses Potenzial und fördern Sie Ihre Mitarbeiter ggf. mit technischen Weiterbildungen.

5. **Ein funktionierendes Team ist Grundvoraussetzung für den Erfolg!**

Zwischen den Teammitgliedern muss eine regelmäßige, offene Kommunikation – im Idealfall Face-to-Face – stattfinden. Dafür sollte das gesamte Projektteam an einem Ort arbeiten. Eine Aufteilung in mehrere unabhängige Teams innerhalb eines Projektes sollte vermieden werden.

6. **Stimmen Sie die Entwicklungsreihenfolge risikobasiert ab!**

Regelmäßige Software-Lieferungen durch iteratives Vorgehen sichern den aktuellen Entwicklungsstand ab. Beachten Sie dabei, dass das Wichtigste stets zuerst erledigt wird.

7. **Achten Sie auf die „Definition of Done“ bei jeder Lieferung!**

Eine Lieferung sollte erst erfolgen, wenn das Ergebnis der „Definition of Done“ entspricht und alle Teammitglieder damit einverstanden sind.

8. **Lösen Sie sich von starrem Management-Vorgehen, seien Sie pragmatisch!**

Idealismus ist nicht gefragt. Arbeiten Sie mit jenen Vorgehensweisen, die funktionieren und nicht nach starren Mustern, die das Management vorgibt.

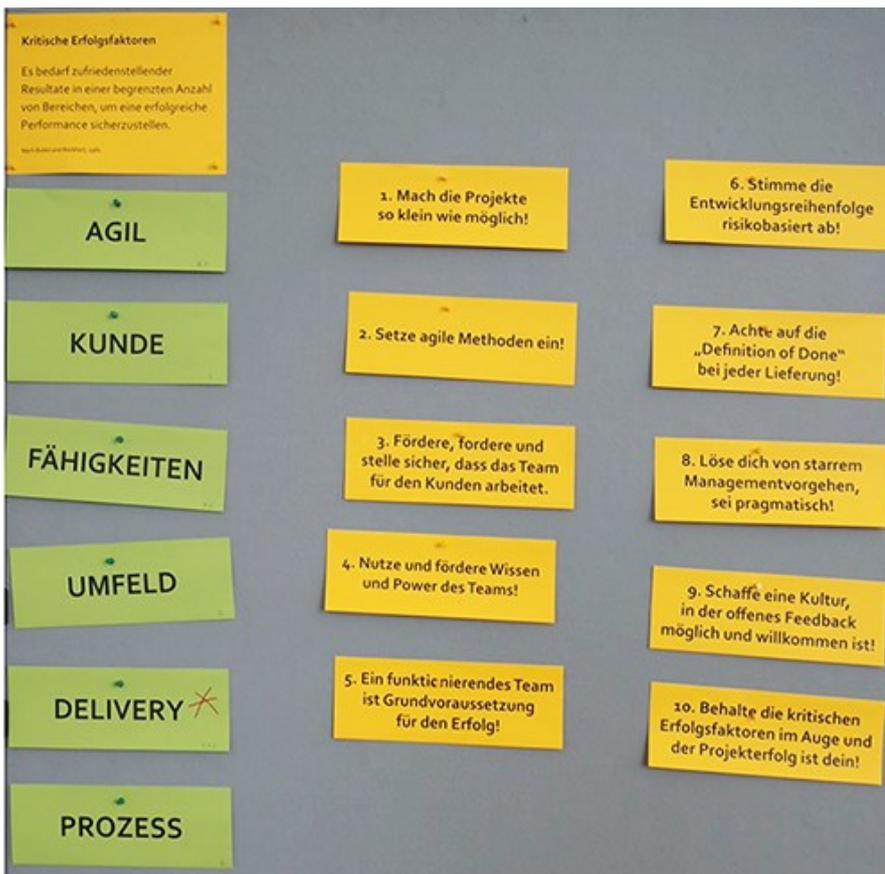
9. **Schaffen Sie eine Kultur, in der offenes Feedback möglich und willkommen ist!**

Ein Team kann nur besser werden, wenn offen Feedback gegeben wird. Sorgen Sie dafür, dass Ihre Mitarbeiter regelmäßig Feedback über ihre Arbeit bekommen, aber auch selbst Feedback geben.

10. **Behalten Sie die kritischen Erfolgsfaktoren im Auge und der Projekterfolg ist garantiert!**

>

Agile Projekte



◀ (Mag. Margaret Grün-Kerr)

[Die Autorin]

Mag. Margaret Grün-Kerr ist als Requirements Engineer bei SEQIS die Schnittstelle zwischen Kunden, Benutzer, Fachbereichen und Entwicklung. Dabei stehen für sie die Bedürfnisse der Menschen und Kunden stets im Mittelpunkt.

Ihre langjährige Erfahrung im Requirements Engineering, der Softwareentwicklung und der Qualitätssicherung/ Software Test bringt sie erfolgreich in jedes Projekt ein und trägt aktiv dazu bei eine qualitativ hochwertige Lösung zu erzielen.



Hier könnte Ihr Inserat stehen!

**Haben Sie ein interessantes Testtool?
Suchen Sie einen qualifizierten Tester für Ihr Team?**

Nutzen Sie die Möglichkeit, sich direkt an die Testcommunity zu wenden.

Kontaktieren Sie das ATB, wenn Sie nähere Informationen zu den Konditionen für Inserate oder Partnerschaften haben wollen.

[The Evolution of Cooperation - Why Agile Approaches Will Prevail]

von Michael Niessl

One of my favorite non-fiction books is Robert Axelrod's *The Evolution of Cooperation*, published more than 30 years ago. The book compares different strategies to win a game called the Prisoner's Dilemma, a classic scenario (explained below) used in game theory. Based on Axelrod's findings, he showed how cooperation can, and most probably will, evolve even if the environment is hostile. Under certain circumstances these findings, prove or at least strongly suggest, that transitions from traditional development methodologies to agile methodologies will succeed even in big organizations, and that agile development approaches will prevail over time.

The Prisoner's Dilemma

The idea of the Prisoner's Dilemma centers around two hypothetical criminal friends. The friends are arrested and faced with imprisonment. Authorities can't present enough evidence to justify a high sentence, so the prosecutor, hungry for success, offers a deal: if one prisoner betrays his friend he will be set free immediately and the other one will be imprisoned for *three years*. This deal is offered to both friends without giving them a chance to coordinate their actions. If both friends rat out each other, the sentence will be *two years* for both of them. If none of them speaks, they will both do just a *year*.

| | | |
|-----------------------------|--|--|
| | Friend 2 keeps silent | Friend 2 rats on his friend |
| Friend 1 keeps silent | 1 year sentence for both | Friend 1: 3 years Friend 2: goes free |
| Friend 1 rats on his friend | Friend 1: goes free Friend 2: 3 years | 2 years sentence for both |

The interesting fact is that, from a game theory point of view, the wisest albeit counterintuitive decision would be to betray your friend. It would guarantee the best possible outcome *independent* of your friend's decision. Scientists have a fancy term for that: in game theory it is called a *Nash Equilibrium*, named after Nobel prize winner John F. Nash whom you may know from the movie "*A Beautiful Mind*".

Evolution of Cooperation

In the beginning of the 1980s, Robert Axelrod asked fellow scientists around the world to submit strategies for a contest based on the Prisoner's Dilemma. He changed one significant parameter though. Instead of just playing just one round of the game, the strategies were compared over the course of a series of rounds. In such an *Iterated Prisoner's Dilemma*, a game strategy can benefit from actions taken in previous rounds – e.g. if one opponent started to cooperate in previous rounds the other opponent can choose to reciprocate that behavior.

Everybody taking part in the contest submitted a strategy. Axelrod would then compare each strategy against another strategy in 200 iterations of the Prisoner's Dilemma. Eventually the scores accumulated by each strategy throughout the tournament led to an overall winner: the best strategy.

It was surprising that despite the rather complex and elaborated strategies (or better, algorithms) used in the tournament, a very simple strategy won in the end. It's called 'Tit for Tat'. The strategy is to cooperate (keeps silent) in the first round and mirror the other player's decision from the previous round for all subsequent moves. After analyzing and publishing the results, Axelrod called for another tournament to see if there was an even better strategy. Again scientists from different disciplines sent in their strategies, knowing well that 'Tit for Tat' won the first tournament. Guess what happened? Surprisingly 'Tit for Tat' won again.

Based on the findings of these two tournaments, Robert Axelrod derived insights about the nature of cooperation. In addition to three other characteristics (retaliating, forgiving, and simplicity) that will not be discussed further at this point, he found that a successful strategy has to be cooperative in the first place. Axelrod made sure to mention that there are certain criteria to be met to guarantee that such strategies succeed. First and foremost, it's the iterative nature of his setup - with the likelihood of the opponents 'meeting again' - that fosters cooperation.

Axelrod took it a step further, showing that even in a hostile environment, where there is likelihood of opponents "playing dirty" cooperative strategies will not only survive but also succeed.

Small clusters of individuals playing cooperatively will be more successful than opponents playing dirty.

The beautiful thing is that it doesn't work the other way round – small clusters of individuals playing dirty will not succeed in a cooperative environment. I see this as a very romantic and optimistic (but still mathematically sound) way to look into the future.

Agile Transitions

Coming back to the tester's world, let's look at the ongoing changes in the development departments of established enterprises around the world. The last couple of years have seen the rise of agile development methodologies. Organizations have adopted, or plan to adopt, agile methodologies and either struggle to implement it or struggle to scale their agile adoption. While methodologies like Scrum look simple on paper, it quickly becomes difficult when R&D departments realize that the whole organization needs to pull together for a smooth transition towards a real agile behavior – it can't be done just by the Devs on their own.

>

Agiler Ansatz

Over the last decade(s), a lot of time, effort and money have been invested into building test centers of excellence (TCoEs) to centralize all the testing activities of an organization to use synergies, bundle skills, and provide a single point of contact for anything involving testing. These TCoEs have also often been outsourced, meaning that there are businesses depending on running TCoEs for a large share of their revenue.

As Scrum accounts for 90%+ of the methodologies adopted during agile transitions, it is a reasonable simplification to look at just that methodology. A typical Scrum team consists of three roles: the *Product Owner*, serving as the voice of the customers, the *Scrum Master*, helping the team to become more effective and efficient following the Scrum processes in delivering product increments, and finally the *Development Team*. There is no tester role in a Scrum team (there's neither a 'tech writer' nor a 'business analyst' role)! Testers are considered part of the dev teams, and going forward the line will get continuously blurry. Having testers embedded in development teams is the exact opposite of having testers clustered in a TCoE. All the efforts, time, and money taken to build these TCoEs are not helpful for a smooth agile transition.

In addition, levels of expectations have increased and the tester's role is constantly evolving. Scrum (and other agile methodologies such as Extreme Programming) force development teams to work in short iterations. In the past, the R&D departments hacked away for months and months, then tried to put everything together, integrate it, make a product out of their work, and finally handed it over to the test team. At that point time was typically short, deadlines were passing by, and testing had to be cut short and/or release dates postponed. This approach was called 'waterfall'. The short iterations of Scrum, called sprints, address the shortfalls of the waterfall methodology by its very nature. A product is developed (and even delivered, to stakeholders at least) every two to four weeks.

Work that is done in such a sprint is not just coding – remember that the testers are embedded into the team. The product increment delivered at the end is designed, developed, tested, documented, and ready to be delivered. It is this approach that, during a transition, puts a lot of pressure on the testers previously clustered in TCoEs and on the organization as a whole. Suddenly testers are forced into test automation – otherwise the simply can't keep the pace. The test is 'shifted left' - it has to start as early as development. Manual testers without too much technical background are suddenly part of a very technical development environment. Testing tools need to be changed in order to cope with these new challenges. Processes need to be adapted or be removed altogether.

What's the advantage? Why would any organization accept so many troubles to go through such a transition? Not surprisingly, it's the usual suspects: shorter time-to-market, higher quality but also quicker feedback from customers, a more flexible reaction to this feedback. In short it is best of both worlds: doing the right things and doing things right. Agile methodologies pay off in many different ways.

Why agile will win

How are the previous two stories, agile transitions and evolution of cooperation connected? Although it may seem to be farfetched, I think the two are closely related. Robert Axelrod's work shows that such transitions can succeed even if, all things considered, it doesn't look very promising in the beginning.

Agile methodologies enforce cooperation. One of the key agile principles is 'Individuals and interactions over processes and tools' – which very simply means that people talk to each other, work together... they cooperate in order to create a better product for their customers faster. Agile methodologies, and thus cooperation, pay off – they reward their proponents. That is one of the key criteria, discovered by Robert Axelrod, needed for cooperation to succeed.

The other very important criteria, the 'shadow of the future' as he called it, where you have to meet again for cooperation to evolve (i.e: your actions today will have an impact on how your colleagues treat you in the future) is an intrinsic characteristic of Scrum. A tongue-in-cheek way to look at it is that a string of sprints is nothing more than an Iterated Prisoner's Dilemma. Sprints guarantee that the 'opponents' will meet again and again and their actions will have an impact on future behaviors.

So although maintaining the status quo as a team (e.g. as the TCoE) might appear to be the best solution individually, it is very likely that a solution that is best for the greater good of the whole organization will succeed over time. All it takes is a small group of individuals working together and an environment that fosters cooperation – as Scrum does.

Start the revolution – you will succeed.

◀ (Michael Niessl)

[The author]

Michael Niessl started as a software tester more than a decade ago. Working for Tricentis, a global player as a test automation and test management tool vendor, he had the chance to work at the very edge between development and test ever since. As an engineering manager and agile coach he is pushing forward the agile transition of Tricentis, facing all the challenges such transitions mean for an organization at whole.



[Performance—Fakten und Emotionen]

von Patrick Eichhorn

Performance Testen ist eine der wohl umfassendsten Aufgaben die einem Testspezialisten gestellt werden können. Er braucht folgende Fähigkeiten:

- a) Ein technisches Grundverständnis im Systembetrieb
- b) Einige grundlegende Programmierkenntnisse
- c) Kenntnisse der zugrundeliegenden Architektur
- d) Kenntnisse der Leistungsdaten, die es zu testen gilt.

Zuerst sollte der Begriff Performance definiert werden. In diesem Artikel nutzen wir die Definition des ISTQB: *A general testing performed to determine how a system performs in terms of responsiveness and stability under a particular workload. It can also investigate, measure, validate or verify other quality attributes of the system, such as scalability, reliability and resource usage.*

ISTQB verwendet einen inhaltlich umfassenden Begriff, es fallen also folgende Tests darunter. Ich bin so frei, eine kleine Guideline zu den Topthemen zu geben:

- a) **Load testing:** Prüft das Verhalten unter einer sogenannten normalen Last. Normal bedeutet, ein durchschnittlicher Tag, mit einer täglichen Spitze und dem normalen „Rauschen“. Bezogen auf die Infrastruktur handelt es sich um die 60% und 80% Auslastungsmarke – ich achte dabei auf CPU, Disk und Netzwerk – weitere Attribute ignoriere ich großzügig.
- b) **Stress testing:** Prüft das Verhalten unter Stress, d.h. das System geht an die 100% Grenze und sogar darüber hinaus. Ziel des Tests ist es, Locking-Mechanismen zu prüfen und das fehlerfreie Abarbeiten der Requests zu gewährleisten.
- c) **Soak testing:** Gilt auch als Endurance-Testing. Als Testmanager im Betriebsumfeld achte ich darauf – dass das System ohne Unterbrechung normale Wochenenden, lange Wochenenden aber auch die Weihnachtszeit ohne Systemeingriff überstehen kann. Normalerweise werden hier Memory-Leaks, Zugriffssperren, Seiteneffekte und anderes erkannt.

- d) **Spike testing:** Dieser Test wird durchgeführt bevor eine Marketingkampagne lanciert wird oder zu anderen speziellen Anlässen. Im Bankingbereich sind dies Tests wie EFTPOS am 24.12. jeweils um 15:00h, wenn alle ihre Weihnachtseinkäufe tätigen oder im Mobilbereich am 31.12. wenn alle die Neujahrsglückwünsche versenden möchten. Es wird ein sprunghafter Anstieg der Nutzung simuliert. Ziel des Tests ist die dynamische Ausweitung der Ressourcenkapazität ohne spürbaren Einfluss auf das Gesamtsystem.
- e) **Configuration testing:** Im Fokus stehen hier infrastrukturelle Themen wie das Load Balancing und Load Strategien die außerhalb der Applikation liegen. Für diese Konfigurationen sollte ein Benchmark gesetzt werden – eine Eichungsmessung oder Null-Linie wogegen verglichen wird.
- f) **Isolation testing:** Eine wichtige Strategie ist es, Wechselbeziehungen zwischen Komponenten (Infrastruktur und Software) frühzeitig zu erkennen. Dies gelingt in der Regel durch Kombinationen der Test-Sets.

Damit diese Tätigkeiten strukturiert angegangen werden können bietet es sich an ein sauberes Monitoring der Produktion aufzuziehen. Im Backend

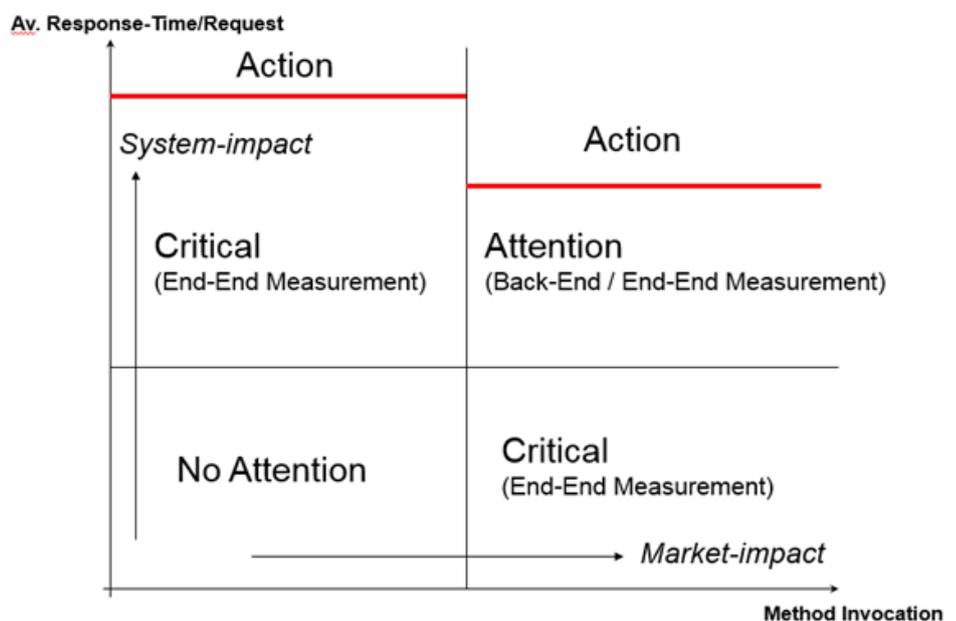
kann ermittelt werden, welche Business-Transaktionen am meisten verwendet werden. Diese Business-Transaktionen (Beispiel: Geldbezug am Automaten) umfassen eine Reihe von technischen Funktionen, die mittels Unit-Test angesteuert werden können. Diese Reihe wird dann im entsprechenden Produktionsmix als Testsuite abgebildet. Jede Businessstransaktion soll so in ein Framework eingebettet werden.

Wie optimiere ich mein Funktionsportfolio für meine Performancemessungen? Es empfiehlt sich ein einfacher Quadrant: auf der einen Seite ist es die Häufigkeit der Verwendung der Funktion auf der anderen Seite die Dauer / Antwortzeit der Funktion. Multiplizieren Sie diese beiden Werte und Sie erhalten eine Rangliste der Wichtigkeit der Funktion (siehe Abbildung).

Sie können jede Funktion in einem der Quadranten einbetten.

Zu guter Letzt gilt es die End-to-End-Zeiten korrekt interpretieren zu können. Dabei hat sich in der Praxis folgende Faustregel bewährt:

- a) Wenn Medianwert und Durchschnittswert innerhalb des 1 Sigma Intervalls liegen, gilt die Funktion als „hart“
- b) Bei Median und Durchschnittswert innerhalb des 2 Sigma Intervalls gilt die Funktion als „ausreichend“.
- c) Liegt die Funktion darüber, ist Handlungsbedarf angesagt. >

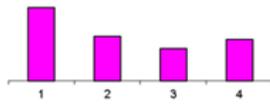


Performancetests / ATB Charity

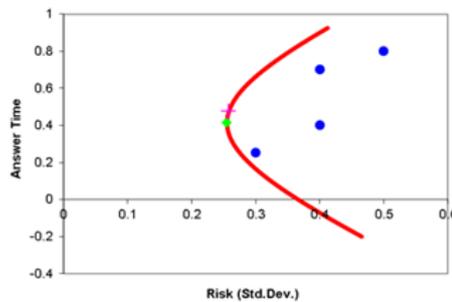
Nachdem Sie nun die notwendigen Zahlen ermittelt haben, können Sie eine klassische Portfolio-Optimierung durchführen, die wie folgt aussehen kann:

Correlated function set

| | Expected | Std.Dev |
|-----------------|----------|---------|
| Login | 0.25 | 0.3 |
| Logout | 0.4 | 0.4 |
| Register | 0.8 | 0.5 |
| Password Change | 0.7 | 0.4 |
| Correlation | 0.5 | |
| Rauschen | 0.2 | about 0 |



Optimized Function Set



StdDev of measured answertimes = Risk

Expected = Average time in case of Median is in range of StdDev

Acceptance on base of optimized function set, based on percentiles

Basierend auf dieser mathematischen Analyse können Sie gezielt den kostenoptimalen Punkt ermitteln, wo die Entwicklung respektive Infrastruktur mit den Optimierungen stoppen muss, da es aus betrieblicher und geschäftskritischer Sicht keinen Sinn ergibt, mehr Zeit und Geld zu investieren.

Take away – speziell für Sie:

- a) Konzentrieren Sie sich auf einige wenige Funktionen – zerlegen Sie diese in messbare Teile
- b) Achten Sie auf die Benutzerführung und das Nutzererlebnis – oft kann bereits ein Stundenglas ein Performanceproblem lösen
- c) Überall wo ein Stundenglas erscheint ist es notwendig einen Performancetest durchzuführen
- d) Den größten Performanceschub erreichen Sie durch eine optimale Nutzerführung
- e) Suchen Sie nach den Klassikern: animierte Hintergründe und große Bilder sind im Performancegeschäft ein „no-go“
- f) Achten Sie auf den Einsatz bestehender Technologien – nach Möglichkeit sollten sowohl File-Streaming (PDF, Video) als auch File-Download situativ konfiguriert und eingesetzt werden.

Ich hoffe, dass Sie mit dieser kurzen Übersicht die eine oder andere Idee für Ihre Teststrategie nutzen können. Viel Erfolg beim Performance Testing.

◀(Patrick Eichhorn)

[Der Autor]

Patrick Eichhorn, Vorsitzender des Swiss Testing Board, Founder und Vice-Chair of TBOK ISTQB ist leitender Testmanager für mobile Apps auf Android, iOS und Windows. Als Betriebsökonom mit einem Master in Business Information Management verbindet er klassische ökonomische Bewertungsmodelle mit Software Testansätzen. Er prüft mit seinem Team bis zu 20 Produktionseinführungen pro Woche mit einem dreistufigen Staging – Installationsumgebung – Vorproduktion - Produktion. Low-Risk Komponenten sind mit vollautomatischen Tests und nachlaufenden Kontrollen in die Produktion zu bringen. Die Systemverfügbarkeit 7*24h war zu 100% gegeben.



Stiftung Kindertraum

[ATB Charity Punsch:
Motto „Vernetzen. Spaß haben und HELFEN“]

Austrian Testing Board
Member of ISTQB®

Auch heuer konnten dank der im Rahmen des ATB Charity Event gesammelten Spenden (der Insider berichtet darüber) mehrere Kinderwünsche erfüllt werden.

Die 4-jährige **Mirjam** profitiert von heilpädagogischem Voltigieren. Bei dieser ganzheitlichen Therapieform wird eine positive Beeinflussung des Befindens, des Sozialverhaltens und der Persönlichkeitsentwicklung angestrebt. Das Mädchen ist bei den Übungen und bei der Versorgung aufmerksam und mit viel Freude dabei.

21 Kinder sind von einem Lama-Alpaka-Projekt begeistert. Highlight waren die neugeborenen Alpaka -und Lamafohlen! Eine große Herausforderung ist es für die Kinder, mit den Tieren den Hindernisparcours zu bewältigen. Sie müssen innerlich ruhig werden, da die Tiere sehr sensibel sind und sonst nicht auf Anweisungen reagieren. Die Tiere helfen, die soziale Kompetenz zu fördern, den Kontakt zur eigenen Gefühlswelt zu finden und die Wahrnehmung zu schulen.

Sabrina (17) freut sich schon auf den Sommer! Sie wird mit ihrer Familie schöne und erholsame Tage in Italien am Meer verbringen.

[Formale Methoden für sicherheitskritische Systeme]

von Dr. Christine Natschläger

In der Medizin, im Transportwesen (vollautomatische Züge, Raumfahrt) oder in der Energieproduktion (Kernkraftwerke) können Softwarefehler schwerwiegende Folgen haben. Verwendet man formale Methoden, dann könnten solche Bugs im Vorhinein vermieden werden. Während die Zuverlässigkeit von einfachen Maschinen sich durch Prüfungen und Tests ermitteln lässt, ist die Beurteilung der Sicherheit von softwaregesteuerten Maschinen mit komplexem Verhalten äußerst schwierig. Hier können Tests nur das Vorhandensein von Fehlern zeigen, nicht aber deren Abwesenheit beweisen. Fehler in Softwarekomponenten können zu den verschiedensten Zeitpunkten im Verlauf des Entwicklungsprozesses entstehen - angefangen bei einer inadäquaten Anforderungsspezifikation über Fehler im Entwurf bis hin zu Implementierungs- und Integrationsfehlern, alle stellen eine potentielle Gefahr im späteren Betrieb des Systems dar. Es gibt daher verschiedene Ansätze, um die Zuverlässigkeit von Software zu erhöhen.

Im Interview erzählt Frau **Dr. Christine Natschläger** warum formale Methoden in Zukunft immer bedeutsamer werden und warum sie aber immer noch ein Nischendasein führen.

Was sind formale Methoden?

Bei der Entwicklung sicherheitskritischer Software hat sich eine Herangehensweise etabliert, die Fehler von Anbeginn vermeiden soll. Bei den formalen Methoden liegt der Schwerpunkt auf der Konzeptualisierung der Software. Sie bauen auf mathematischer Modellierung und formaler Logik auf und werden verwendet, um die Software zu spezifizieren und zu prüfen. Mit formalen Methoden kann man von einem mathematischen Modell ausgehend durch die logische Analyse die Eigenschaften eines Systems voraussagen. Es kann festgestellt werden, ob eine Systembeschreibung (Spezifikation) in sich konsistent ist, ob bestimmte Eigenschaften garantiert werden können und ob die Anforderungen richtig entworfen und implementiert wurden. Außerdem kann die Spezifikation für die Simulation/ Animation, zur Codegenerierung, zur Testgenerierung und zur Testauswertung benutzt werden.

In welchen Bereichen können formale Methoden besonders gut eingesetzt werden?

Zum Beispiel Event-B und ASMs (Abstract State Machines) werden verwendet, um sicherheitskritische Systeme (ein Beispiel wäre die Pariser RER Bahn, die vollautomatisch fährt und noch nie einen Fehler hatte) und auch stark verteilte Softwaresysteme zu entwickeln. Wenn etwa in einer Produktionsstraße mit einer Reihe von Robotern mit gleicher Software ein Roboter ausfällt, so sollen die anderen Roboter den Fehler erkennen und die Aufgaben des defekten Roboters unter sich aufteilen. Für derartig komplexe Anwendungen eignen sich formale Methoden.

Es gibt aber auch andere Anwendungsbeispiele, wie zum Beispiel Dialysegeräte, Herzschrittmacher oder Flugzeugkomponenten. All diese Systeme sind repräsentativ für sicherheitskritische, eingebettete System und die erste Herausforderung besteht in der Modellierung der Systeme. Die zweite Herausforderung ist dann der Nachweis der Einhaltung der Sicherheitsanforderungen und der richtigen Funktion unter Rücksichtnahme auf das physische Verhalten der Geräte.

Wie unterscheidet sich die Entwicklung mit formalen Methoden von der herkömmlichen Softwareentwicklung?

Es verschiebt sich einiges an Aufwand und Zeitverbrauch im Projekt: Es wird mehr Sorgfalt in die Anforderungsanalyse, die Spezifikation und das Grob-Design gesteckt; dieser Mehraufwand kann aber dadurch wieder hereingeholt oder sogar überkompensiert werden, dass während der Implementierung wesentlich weniger Spezifikations- und Designfehler mühsam identifiziert und behoben werden müssen und auch die Testphase schneller durchlaufen werden kann. Bei der Auswahl geeigneter, modell-basierter Methoden kann das für Spezifikation, Validierung und Verifikation erstellte Modell auch zur Unterstützung von Implementierung, Test-Suite-Erstellung, Dokumentation und schließlich auch Wartung wiederverwendet werden.

Welche Eigenschaften hat die Entwicklung mit formalen Methoden?

Der Entwickler muss das System besser kennen!

Der mit der Spezifikation beauftragte Systementwickler ist gezwungen, sich mehr Gedanken über das zu entwickelnde System zu machen und er muss es im Detail verstehen. Sein Zeitaufwand wird wesentlich höher sein als sonst. Da er jedoch genauer darüber nachdenkt, was zu implementieren ist, werden die folgenden Schritte (Entwurf, Implementierung, Integration) weniger experimenteller Natur sein, weniger Fehler verursachen und Zeit einsparen.

Validierung

Mit formalen Methoden kann das System noch vor der Implementierung durch Animation (Simulation) validiert werden.

Verifikation

Durch Verifikation wird überprüft, ob die Spezifikation in sich konsistent (frei von Widersprüchen) ist, ob bestimmte Eigenschaften (z.B. Sicherheitsanforderungen) garantiert werden können und ob die Anforderungen richtig umgesetzt wurden.



Dr. Christine Natschläger moderierte die SCCH Veranstaltung zu „Formale Methoden und Systems Engineering“ (Bildquelle SCCH)

Software Competence Center Hagenberg

Mensch-zu-Mensch-Kommunikation

Mit einer formalen Spezifikation liegen präzise Aussagen (ohne Zweideutigkeiten) über das gewünschte Verhalten vor. Formalisierung unterstützt damit die eindeutige Kommunikation unter den Entwicklern.

Automatische Code Generierung

Manche (operationale) Formalismen erlauben bei einem genügend hohen Detaillierungsgrad eine automatische Code-Generierung aus der Anforderungsspezifikation. Allerdings muss der generierte Code meist noch nachgebessert werden.

Können formale Methoden alle Softwarefehler vermeiden?

Von allen Fehlern, die bei der Inbetriebnahme und im Betrieb gefunden werden, kommen die meisten und die teuersten aus der Anforderungsspezifikation. Die Fehler aus der Implementierung sind seltener und oft einfacher (billiger) zu korrigieren. Die Spezifikationsfehler können sich durch die ganze Implementierung ziehen und große Änderungen nötig machen. Die schwerwiegendsten Fehler bestehen darin, dass nicht das gewünschte Verhalten spezifiziert wurde. Dies kann durch Validierung im ersten Schritt geprüft werden. Der nächstschwerwiegende Fehler besteht darin, dass der Entwurf die Anforderungen nicht erfüllt.

Mit welchem Aufwand ist zu rechnen, wenn man formale Methoden einsetzt?

Der Aufwand bei der Verwendung von formalen Methoden ist bei der Anforderungsspezifikation relativ niedrig und die Anzahl der Fehler, die dabei gefunden werden können, verhältnismäßig hoch. Bei der Implementierung ist ihre Anwendung aufwendiger, und die Anzahl der Fehler, die sie aufdecken können, niedriger. Abhängig von den möglichen Fehlerkosten (fehlerhafte sicherheitskritische Systeme können Leben gefährden), kann aber auch in dieser Phase der Einsatz sinnvoll sein.

◀ (Dr. Christine Natschläger)

[Die Autorin]

Dr. Christine Natschläger hat ein Diplom in Software Engineering im Medizinbereich und einen Master in Information Engineering & -Management. Ihre Doktorarbeit verfasste Sie zu den Themen BPMN (und deontische Logik). Seit 2004 forscht sie am Software Competence Center Hagenberg in den Bereichen Business Process Modelling (mit formalen Definitionen) und Process Mining. Seit 2015 leitet sie den Forschungsschwerpunkt Rigorous Methods in Software Engineering.



Prof. Dr. Klaus-Dieter Schewe erklärt, wie man medizinische Geräte mit Formalen Methoden sicherer machen kann. (Bildquelle SCCH)



Prof. Dr. Egon Börger, Experte für Formale Methoden und Abstract State Machines. (Bildquelle SCCH)

Mehr Information:

In einem jüngeren Beispiel über die erfolgreiche Einführung formaler Methoden berichten die Entwickler von Amazon Web Services und wie sie dadurch Software-Qualität und Produktivität verbessern konnten – siehe dazu: How Amazon Web Services Uses Formal Methods:

<http://cacm.acm.org/magazines/2015/4/184701-how-amazon-web-services-uses-formal-methods/abstract>

Das Software Competence Center Hagenberg war auch Co-Organisator der ABZ 2016 - Die internationale Konferenz, bei der Entwickler und Anwender der wichtigsten modellbasierten formalen Methoden zusammen kommen:

<http://www.cdcc.faw.jku.at/ABZ2016/>

[Nächste Ausgabe]

In der nächsten Ausgabe finden Sie einen Artikel zu dem vom Software Competence Center Hagenberg (SCCH) entwickelten Tool **eKNOWS**, welches die automatische Extraktion und Analyse von Fachwissen direkt aus dem Quellcode ermöglicht.

[Redakteurinnen und Redakteure gesucht]

Haben Sie einen außergewöhnlichen Bug gefunden? Kennen Sie ein Tool, von dem die Testercommunity unbedingt wissen sollte? Haben Sie ein Buch gelesen, das andere auch lesen sollten?

Ja? Dann schreiben Sie an den ATB-Insider. Wir suchen noch Redakteurinnen und Redakteure, die Spaß am Schreiben haben.

Einsendungen an: newsletter@austriantestingboard.at

◀ (Karl Kemminger)

[Rückblick 3. SQS Quality Brunch in Wien]

von Dr. Mohsen Ekssir

Das dritte Event der Veranstaltungsreihe SQS Quality Brunch fand am 02. Juni in Wien im Hotel Regina statt. Diesmal stand das Event gänzlich im Zeichen von SAP-Testing und seiner Zukunft, Best Practices, End-to-End sowie Tipps & Tricks rund um diesen Themenbereich.

Testing 4.0 – nicht nur in der industriellen Fertigung, sondern auch in allen anderen Branchen sind heute immer höhere Schnelligkeit, Effizienz und Flexibilität gefragt. Auch im SAP-Umfeld gewinnen Themen wie Agile, Continuous Improvement, DevOps und Cloud Services weiter an Bedeutung. Dies stellt auch das Testing vor neue Aufgaben. Flexibilität bei hohen Automatisierungsgraden und Wiederverwendbarkeit sind nach wie vor angestrebte Ziele bei der Software-Qualitätssicherung.

An diesem Vormittag stellte zuerst Johannes Kreiner (GF von SQS Österreich und SQS Schweiz) kurz das SQS Unternehmen vor. Dann waren die beiden Experten Heinz Schachinger (SAP) und Jan Berkefeld (SQS) an der Reihe, ihre Vorträge zu halten. Im Anschluss an die Präsentationen diskutierten die beiden Referenten mit Teilnehmern aus verschiedenen Branchen des österreichischen IT-Marktes über die Herausforderungen, Ansätze und Visionen für die Zukunft des SAP-Testings.

Bei diesem Experten-Event waren SAP-Testmanagement und der Einsatz des Solution Managers 7.2, der im Sommer diesen Jahres von SAP freigegeben wird, genauso wichtige Themen wie der visionäre SQS-Ansatz und die Technik des Process based Testings (PbT) zur Effizienzsteigerung des Tests.

Nach der Diskussionsrunde wurden drei Exemplare des Fachbuches „Thought Leadership 2015“ (Whitepaper Book), eine SQS-Publikation, unter den Teilnehmern verlost.

Link zu Vortragsfolien [hier](#)

◀(Dr. Mohsen Ekssir)

Save the Date:

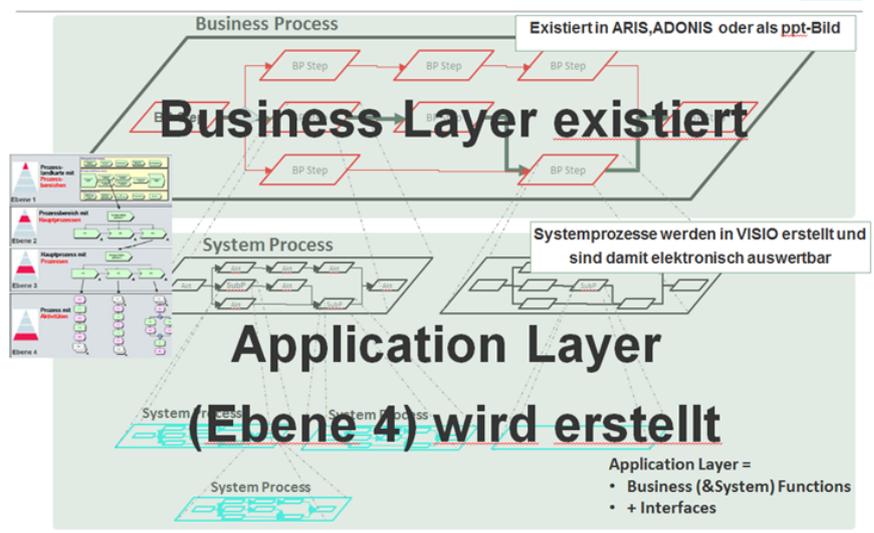
Der 4. SQS Quality Brunch findet am **11. Okt. 2016** in Wien, Hotel Regina statt.

Details folgen.



Process based Testing (PbT)

Vorgehen - Die zwei Layer eines Geschäftsprozessmodells



Die Vortragenden



Redner mit Gewinnern des SQS-Fachbuches „Thought Leadership 2015“ beim Quality Brunch

Die Veranstaltungsreihe **SQS Quality Brunch**, die im Jahr 2015 ins Leben gerufen worden ist, will bei freiem Eintritt den Austausch der Fach-Community rund um aktuelle und brennende Themen zu Software-Test und -Qualitätssicherung fördern. Neben einem kulinarischen Rahmenprogramm bietet der SQS Quality Brunch mit Fachvorträgen von Experten und anschließendem Networking Know-how-Transfer in entspanntem Ambiente.

Neues aus dem ASQF e.V.

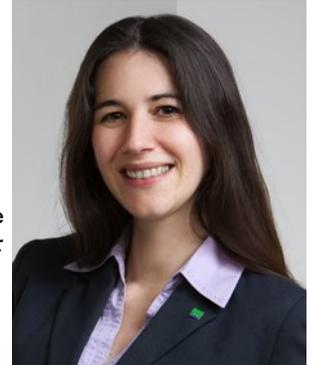
[ASQF-Fachgruppen-Abende]

Die regionale Fachgruppe Software-Test Österreich des ASQF veranstaltet regelmäßig Fachgruppenabende zu ausgewählten Themen. Die Teilnahme ist kostenlos, Anmeldung unter angegebenem Link. Im Anschluss an jeden Abend gibt es einen Imbiss und Gelegenheit zur Diskussion und zu Networking.



Nächster Fachgruppenabend: Di., 27. September 2016, 18:00 - 20:00
FH Technikum Wien, Raum HS_A1.04, Höchstädtplatz 5, 1200 Wien

Thema: Digitalisierung – Qualität aus Sicht der Endkunden
Herausforderungen an Softwaretest in der digitalisierten Welt
Referentin: Barbara Stoik



Abstract: Im Zeitalter der Digitalisierung bestimmt der Endkunde über die Qualität von Software und Prozessen. Gefällt ihm die aufgerufene Seite oder APP nicht, findet er sich nicht zurecht oder sagen ihm die angebotenen Produkte nicht zu, zieht er weiter. Dieser Umstand macht immer mehr Anbietern von Dienstleistungen zu schaffen.

Während man früher die eigenen Mitarbeiter noch schulen und anweisen konnte, muss heute mehr denn je auf Schnelligkeit, Funktionalität und Benutzbarkeit – kurz Qualität gelegt werden. Doch wie bestimmt man den Qualitätsbegriff des Endkunden?

Jeder Mensch ist verschieden, unterschiedliche Kunden haben unterschiedliche Bedürfnisse und nutzen unterschiedliche Medien. Darauf und was diese Veränderungen und neuen Herausforderungen für den Softwaretest bedeuten, wird in dem Vortrag eingegangen.

Barbara Stoik ist seit 2010 bei SQS beschäftigt und leitet das Service Management für die DACH Region. Im Rahmen dieser Tätigkeit, sorgt sie mit ihrem Team für die Weiterentwicklung der SQS Services und den Aufbau der dafür nötigen Skills bei den Mitarbeitern. Zudem ist Barbara als Senior Consultant in verschiedenen Rollen bei unterschiedlichsten Kunden eingesetzt.

Vor der SQS war Barbara Stoik rund zehn Jahre in der Generali VIS Informatik (heute Generali IT) in Österreich als Teil des dortigen Inhouse Test Centers tätig.

Um Anmeldung unter <http://bit.ly/28NMf6a> wird gebeten. **Die Teilnahme ist kostenlos!** ◀(Mohsen Ekssir)



Rückschau auf den ASQF Fachgruppenabend am 21.06.2016 zum Thema „Valide ... was für ein Zustand!) - Qualitätssicherung kritischer Systeme“.

Christine Brandner stellte die Herausforderungen bei der Validierung von kritischen Systemen z.B. in der Lebensmittel- und Pharmaindustrie vor, bei denen aufgrund behördlicher Vorgaben dokumentiert nachgewiesen werden muss, dass diese Systeme ihren Zweck erfüllen. Das betrifft unter anderem die Software zur Steuerung von Produktionsanlagen, Software zur Prozessunterstützung, die Prozesse selbst, die Hardware und Infrastruktur sowie Schnittstellen zu anderen Systemen. Die ordnungsgemäße Validierung wird regelmässig von nationalen Aufsichtsbehörden (z.B. Arzneimittelbehörde) geprüft. Bei größeren Änderungen, aber auch in regelmässigen Abständen (ca. alle 3 Jahre) ist eine Revalidierung notwendig. Die Ergebnisse inklusive der Aussage, ob das System validiert ist oder nicht, werden in einem Validierungsbericht beschrieben. Mehr zu diesem Thema z.B. [hier](#), oder Sie kontaktieren die Vortragende:

https://www.xing.com/profile/Christine_Brandner?sc_o=mx_b_p

◀(Karl Kemminger)

ASQF Fachgruppenabende 2016

- Dienstag, 27.09.2016 (siehe oben)
- Dienstag, 22.11.2016 (Thema Software Archäologie)

[ASQF-Fachgruppenleitung Softwaretest Österreich]

DI Dr. Mohsen Ekssir leitet seit 2010 die ASQF-Fachgruppe Software Test Österreich. Er versucht durch die Veranstaltung der Fachgruppenabende den Wissensaustausch in Bezug auf Softwaretest und Qualitätssicherung in Österreich zu verstärken. Die Fachgruppenabende bieten eine Plattform für einen kostenlosen Wissenstransfer in Bezug auf Softwaretest und Qualitätssicherung. Eine Plattform für Tester, Studenten und Experten, um neue Themen, neue Ideen, neue Methoden, Praxiserfahrungen, Praxisberichte u.s.w. rund um das Thema Softwaretest zu vermitteln und auszutauschen.

Falls Sie interessiert sind, an einem Fachgruppenabend einen Fachvortrag zu halten, kontaktieren Sie bitte gerne den Fachgruppenleiter, Herrn Ekssir, um die weiteren Details zu besprechen:

E-Mail: mohsen.ekssir-monfared@asqf.de
Mobile: +43 699 10909332



[ATB Expertentreff: Last- und Performancetests]

von Alexander Weichselberger

Im letzten Expertentreff vor der Sommerpause haben wir uns mit dem Thema Last- und Performancetests beschäftigt. Vorstellig wurde Siegfried Göschl, Software Engineer, mit seinen Erfahrungen im Projekt des Online-Bankenportals der Erste Bank, mygeorge.at.

Die Grundüberlegungen dazu, warum gerade das Thema Last- und Performancetests (LPT) oft unterschätzt wird, machen klar, dass die Lösung in einem frühzeitigen, durchdachten Vorgehen zu finden ist – ist der Zug einmal in Richtung Go Live am Weg und stößt man dann auf Probleme, kann man kaum noch wirklich „in time“ oder „in budget“ abschließen.

Wie auch immer man das Thema betrachtet, LPTs sind komplex, interdisziplinär, zumeist technisch aufwändig und Tool-getrieben.

Was ist damit genau gemeint?

- **Komplex:** Kann man bei funktionalen Tests noch gut mit Mocking oder nur mit Teilen des Systems zu recht vernünftigen Ergebnissen kommen, ist das beim LPT kaum möglich. Die Testumgebungen sind im Regelfall nicht einfacher als die Produktion, sondern komplexer: Ich brauche ja noch zusätzliche Lastgeneratoren, mehr Monitore, etc. als in der Produktionsumgebung.
- **Interdisziplinär:** Kann man funktionale Tests oft noch „nur im Entwicklungsteam“ erledigen, so sind bei LPTs die Teambzusammenstellungen sinnvollerweise „global“. Neben den Vertretern des Fachbereichs und der Kunden brauchen wir üblicherweise Projektleiter, IT Systemspezialisten, Security Manager, Datenadministrator, Tester, usw.



Alexander Weichselberger.
Moderator der ATB Expertentreffs

- **Technisch aufwändig:** Zusätzlich zur Komplexität kommen noch das Abstimmen mit verbundenen Testumgebungen und die Herausforderung, relevante Testdaten (ohne Widerspruch zum Datenschutzgesetz) zu erstellen, hinzu.
- **Tool-getrieben:** Nachdem die Last nicht manuell aufgebracht werden kann, müssen Simulationen programmiert werden. Das Skripten der Simulationen, aber auch die Unterstützung von Protokollen, das Lizenzierungsmodell und die Möglichkeiten, den Lastverlauf zu beeinflussen bzw. in Echtzeit Monitoring-Feedback zu bekommen und bei der Auswertung durch Root-Cause-Analysen unterstützt zu werden, sind übliche Entscheidungskriterien für LPT-Tools.

Im Rahmen des Treffs wurden noch Fragen zum Verhältnis von Testumgebung zur Produktion diskutiert, welche Lösungsansätze es in Richtung Datenschutzgesetz gibt und wie ein hilfreiches Reporting mit Warnings und Alerts zu gestalten ist. Hr. Göschl hat eindrucksvoll vermittelt, wie bei mygeorge.at bereits von Beginn an („continuous integration“) auf die Performance geachtet wird und mit welchen Tools, Prozessen und Abstimmungen gearbeitet wird.

Im Rahmen der Diskussion hat Hr. Martin Wildbacher, SEQIS GmbH, noch Wissenswertes aus der Testerecke im Zusammenhang mit LPT vorgestellt.

Alles im Allem: Gute Session, viele Fragen beantwortet, alles gut!

What's next?

... der nächste Update zu den Inhalten der heurigen Expertentreffs kommt im Sommer – wir planen in den Monaten nach der Sommerpause mit 3 Treffs die Lücke bis Weihnachten interessant zu füllen. Auf Basis Ihres Feedbacks beschäftigen wir uns mit folgenden Themenbereichen:

- Testcenter und Varianten im Hinblick auf Agile, DevOps, etc.
- Embedded Testing
- Internet of Things & Testing
- Continuous Testing
- Social Skills Improvement
- Security & Co
- BigData / DWH / BI Testing
- Tester Ethik (Stichwort „VW Software“)

Große Themenbereiche also, eine Auswahl bzw. Fokussierung ist daher wohl notwendig. Falls Sie Interesse an einer aktiven Rolle (Vortrag, Diskussionssteilnahme) oder an einem spezifischen Thema haben: Schicken Sie mir bitte einfach eine Email an weichselberger@austriantestingboard.at und wir können uns abstimmen.

Ihr

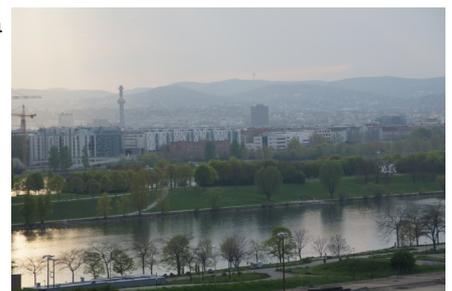
Alexander Weichselberger

ATB Expertentreff

◀ (Alexander Weichselberger)



Martin Wildbacher und
Siegfried Göschl



Blick auf Wien

Voraussichtliche Termine der ATB Expertentreffs 2016

- Mittwoch, 21.09.2016
- Mittwoch, 12.10.2016
- Mittwoch, 23.11.2016

Details zu den Events siehe

[ATB-Homepage](#) bzw. [XING Gruppe](#)

| Kurs | Termin | Ort | Anbieter |
|---|-----------------------------------|--|------------------------------------|
| ISTQB® Certified Tester Foundation Level | 04.07. – 07.07.2016 | Linz, Wien, Graz | Software Quality Lab |
| | 11.07. - 14.07.2016 | Lustenau | |
| | 22.08. – 25.08.2016 | Linz, Wien, Graz | |
| | 29.08. – 01.09.2016 | Lustenau | |
| | 05.09. – 08.09.2016 | Wien | ANECON |
| | 12.09. - 15.09.2016 | Wien | SQS |
| | 03.10. - 06.10.2016 | Wien | |
| | 03.10. – 06.10.2016 | Linz, Wien, Graz | Software Quality Lab |
| | 10.10. – 13.10.2016 | Lustenau | |
| | 14.11. – 17.11.2016 | Linz, Wien, Graz | |
| | 21.11. – 24.11.2016 | Lustenau | |
| | 28.11. – 01.12.2016 | Wien | ANECON |
| | 28.11. - 01.12.2016 | Wien | SQS |
| ISTQB® Certified Tester Foundation Level Englisch | 12.12. – 15.12.2016 | Linz, Wien, Graz, Lustenau | Software Quality Lab |
| ISTQB® Certified Tester Foundation Level Extension, Agile Tester | 12.07. – 13.07.2016 | Linz, Wien | Software Quality Lab |
| | 19.10. – 20.10.2016 | Linz, Wien | |
| | 17.11. – 18.11.2016 ¹⁾ | Mödling | SEQIS Software Testing GmbH |
| | 05.12. - 06.12.2016 | Wien | SQS |
| ISTQB® Certified Tester Advanced Level Test Manager | 18.07. - 22.07.2016 | Wien | SQS |
| | 12.09. – 16.09.2016 | Linz, Wien | Software Quality Lab |
| | 12.10. – 18.10.2016 | Wien | ANECON |
| | 07.11. - 11.11.2016 | Wien | SQS |
| | 28.11. - 02.12.2016 | Linz, Wien | Software Quality Lab |
| ISTQB® Certified Tester Advanced Level Test Analyst | 17.10. - 20.10.2016 | Wien | SQS |
| | 07.11. – 10.11.2016 | Wien | ANECON |
| | 14.11. – 17.11.2016 | Linz, Wien | Software Quality Lab |
| ISTQB® Certified Tester Advanced Level Technical Test Analyst | 04.10. – 06.10.2016 | Linz Wien | Software Quality Lab |
| | 14.11. – 16.11.2016 | Wien | ANECON |
| | 14.11. - 16.11.2016 | Wien | SQS |
| ISTQB® Model-Based Tester Foundation Level | 12.07. – 13.07.2016 | Linz, Wien | Software Quality Lab |
| | 27.09. – 28.09.2016 | Linz, Wien | |
| | 19.10. – 20.10.2016 | Linz, Wien | |
| | 08.11. – 09.11.2016 | Linz, Wien | |

1) Termingarantie. Diese Kurse finden fix statt, unabhängig der Teilnehmerzahl



| Kurs | Termin | Ort | Anbieter |
|---|-----------------------------------|--|-----------------------------|
| Certified Agile Tester® Training (Prüfung in deutsch oder englisch) | 12.09. – 16.09.2016 ¹⁾ | Mödling | SEQIS Software Testing GmbH |
| | 26.09. – 30.09.2016 | Linz, Wien | Software Quality Lab |
| | 10.10. – 14.10.2016 | Wien | ANECON |
| | 21.11. – 25.11.2016 | Wien | SQS |
| | 28.11. – 02.12.2016 | Linz, Wien | Software Quality Lab |
| CATDD® Certified Agile Test Driven Development Training und Zertifizierung | 13.09. – 15.09.2016 | Wien | ANECON |
| | 17.10. – 19.10.2016 ¹⁾ | Mödling | SEQIS Software Testing GmbH |
| | 15.11. – 17.11.2016 | Wien | ANECON |
| IREB Certified Professional for Requirements Engineering Foundation Level | 05.07. – 07.07.2016 | Lustenau | Software Quality Lab |
| | 19.07. – 21.07.2016 | Linz, Wien, Graz | |
| | 20.09. – 22.09.2016 | Lustenau | |
| | 21.09. – 23.09.2016 | Wien | ANECON |
| | 27.09. – 29.09.2016 | Linz, Wien, Graz | Software Quality Lab |
| | 10.10. – 12.10.2016 | Wien | SQS |
| | 11.10. – 13.10.2016 | Linz, Wien, Graz | Software Quality Lab |
| | 18.10. – 20.10.2016 | Lustenau | |
| | 23.11. – 25.11.2016 | Wien | ANECON |
| | 05.12. – 07.12.2016 | Linz, Wien, Graz, Lustenau | Software Quality Lab |
| IREB® Certified Professional for Requirements Engineering Elicitation and Consolidation Advanced Level | 06.09. – 08.09.2016 | Linz, Wien | Software Quality Lab |
| | 05.12. – 07.12.2016 | Linz, Wien | |
| CMAP© Mobile App Testing – Foundation Level | 05.10. – 06.10.2016 | Wien | ANECON |
| | 11.10. – 12.10.2016 | Wien | SQS |
| | 07.11. – 08.11.2016 ¹⁾ | Mödling | SEQIS Software Testing GmbH |
| Certified Agile Business Analysis | 10.10. – 11.10.2016 ¹⁾ | Mödling | SEQIS Software Testing GmbH |

1) Termingarantie. Diese Kurse finden fix statt, unabhängig der Teilnehmerzahl



| Veranstaltung | Termin | Ort | Anbieter |
|---|---------------------|----------------------------|------------------------------------|
| SEQIS Expertentreff „Das Internet der Dinge (IoT): Der nächste QA Paradigmenwechsel, garantiert“ | 15.09.2016 | Wien | SEQIS Software Testing GmbH |
| 360° Testautomatisierung | 06.09. – 07.09.2016 | Wien | ANECON |
| | 02.11. – 03.11.2016 | Wien | ANECON |
| Testgetriebene Softwareentwicklung | 12.07. – 14.07.2016 | Linz, Wien | Software Quality Lab |
| | 11.10. – 13.10.2016 | Linz, Wien | |
| GUI-Testautomatisierung in Theorie und Praxis | 19.10. – 20.10.2016 | Linz, Wien | Software Quality Lab |
| Risikomanagement in Softwareprojekten | 13.10.2016 | Linz, Wien | Software Quality Lab |
| ISAQB® Certified Professional for Software Architecture Foundation Level | 05.09. – 08.09.2016 | Linz, Wien | Software Quality Lab |
| | 28.11. – 01.12.2016 | Linz, Wien | |
| Aufwandsschätzung in Softwareprojekten | 15.11. – 16.11.2016 | Linz, Wien | Software Quality Lab |
| Professionelles Requirements Engineering und Management | 07.11. – 09.11.2016 | Linz, Wien | Software Quality Lab |
| Moderationstechniken im Requirements Engineering | 29.11. – 30.11.2016 | Linz, Wien | Software Quality Lab |
| Software Usability | 26.07. – 28.07.2016 | Linz, Wien | Software Quality Lab |
| | 06.09. – 08.09.2016 | Linz, Wien | |
| Scrum Master Professional | 30.08. – 31.08.2016 | Linz, Wien | Software Quality Lab |
| | 28.11. – 29.11.2016 | Linz, Wien | |
| Continuous Integration and Delivery | 08.11.2016 | Linz, Wien | Software Quality Lab |
| ISPMA® Software Product Management | 28.11. – 30.11.2016 | Linz | Software Quality Lab |
| | 21.11. – 23.11.2016 | Wien | |

[Impressum]

Herausgeber: Austrian Testing Board Alser Straße 4/Hof 1/Eingang 1.5
A-1090 Wien, Austria
Telefon: +43 676 64 35 688 , Fax: +43 2256 65969
Email: office@austriantestingboard.at.

Dieses Magazin richtet sich an Software-Tester im deutschsprachigen Raum. Anregungen, Feedback, Kritik und ähnliches richten Sie bitte an backoffice@austriantestingboard.at
Wenn Sie dieses Magazin abbestellen wollen, nutzen Sie bitte den Abmeldelink im Mail oder senden Sie eine mit Betreff „Storno Magazin“ an backoffice@austriantestingboard.at.
Sämtliche in diesem Magazin zur Verfügung gestellten Informationen und Erklärungen geben die Meinung des jeweiligen Autors wieder und sind unverbindlich. Das ATB übernimmt keinerlei Haftung und Gewähr, insbesondere auch für die Richtigkeit oder Vollständigkeit der darin enthaltenen oder referenzierten Informationen oder deren Anwendung, sowie Druckfehler oder Irrtümer und es werden keinerlei Garantien, Zusicherungen oder sonstige Rechtsansprüche daraus begründet.
Die Redaktion behält sich Kürzungen vor. In keinem Fall spiegeln Leserbriefe die Meinung der Redaktion wieder.

Konferenzen, Iqnite 2016

[Iqnite 2016 Review]

von Yvonne Klee

Auch in diesem Jahr führte die IT-Konferenz iqnite, die vom 26. bis 28. April 2016 im Congress Center in Düsseldorf stattfand, die Software-Quality-Community zusammen. Die Veranstaltung zählt zu den wichtigsten Konferenzen im Bereich Software-Qualität und -Testen im deutschsprachigen Raum und stand in diesem Jahr ganz im Trendthema „Agilität“.



In über 50 Vorträgen und Workshops wurde den Teilnehmern ein buntes und informatives Programm geboten – von aktuellen Trends, Entwicklungen und Herausforderungen im Bereich Software-Qualität bis hin zu unterhaltsamen Vorträgen wie der des deutschen Comedians „Knacki“ Deuser über die Wichtigkeit von Humor oder der des „Menschenflüsterers“ Tobias Beck über „das Erfolgsmodell moderner Mitarbeitermotivation“.

Neben Agilität bestimmten Testautomatisierung, Mobile Testing oder Managed Testing Services die Programmagenda. Die Keynote-Vorträge wurden von hochrangigen Sprechern geführt, unter anderem von Unternehmen wie Colin Moon Communications, Elektrobit Automotive, XING beziehungsweise der Technischen Hochschule Köln sowie der Hochschulen Bremen & Bremerhaven. Weitere spannende Vorträge wurden u.a. von Commerzbank, Volkswagen, Otto und Lufthansa präsentiert.



Neben den Vorträgen, Workshops und dem Abendprogramm gab es auch dieses Jahr eine konferenzbegleitende Ausstellung, auf der Software-Qualitäts-Experten über ihre Produkte und Dienstleistungen informiert haben. Zum ersten Mal wurde in diesem Jahr der „iqnite Award 2016“ für innovative Projekte und Lösungen verliehen.

Eva Wulfert, Testmanagerin bei Creditreform, über die iqnite 2016:

„Ich war vor vielen Jahren schon einmal Teilnehmerin und bin nun sehr überrascht, wie positiv sich die iqnite verändert hat. Da wir derzeit agile Methoden sehr stark fokussieren, war das Schwerpunktthema der Konferenz sehr spannend – gerade weil es auch sehr kontrovers diskutiert wurde.“

Kirsten Roßmann, Entwicklerin bei Otto, über die iqnite 2016:

„Ich habe schon in der Vergangenheit an der iqnite teilgenommen, dieses Jahr zum ersten Mal auch als Referentin. Ich schätze die gute Organisation der Veranstaltung, aber vor allem auch den Erfahrungsaustausch. Ich bin schon seit längerer Zeit in agilen Teams aktiv und kann mir Softwareentwicklung, wie sie früher gemacht wurde, gar nicht mehr vorstellen.“

Dr. Andreas Gerstinger, Safety & Process Engineer bei FREQUENTIS über die iqnite 2016:

„Ich möchte mich für die gelungene Konferenz bedanken und besonders einen Punkt hervorheben: die genialen Keynotes und Vorträge, die sehr unterhaltsam und eindrucksvoll waren. Ich war schon auf Dutzenden Konferenzen, allerdings habe ich nie eine vergleichbare Qualität wie auf der iqnite 2016 erlebt.“

Weitere Informationen zur iqnite 2016: <http://www.iqnite-conferences.com/de>

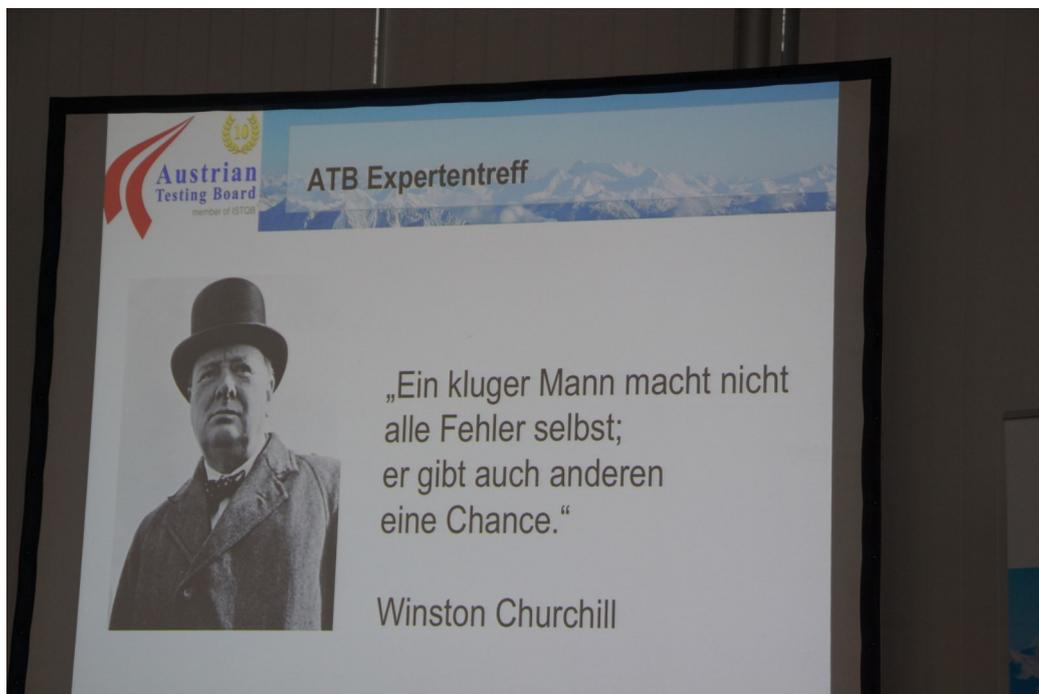
◀ (Yvonne Klee)

Konferenzen / Silberpartner

[Konferenz Kalender]

| Konferenz | Termin | Ort | Deadline Call for Paper |
|---------------------------------------|---------------------|-----------------------|-------------------------|
| Accelerate 2016 | 12.09.—12.09.2016 | Wien | |
| ASOT 2016 | 21.09. - 23.09.2016 | Klagenfurt | |
| QA & Test 2016 | 19.10. - 21.10.2016 | Bilbao / Spanien | |
| EuroSTAR 2016 | 31.10.- 03.11.2016 | Stockholm / Schweden | |
| Agile Testing Days | 05.12. - 08.12.2016 | Potsdam / Deutschland | |
| Software Quality Days | 17.01. - 20.01.2017 | Wien | |

[Spruch]



TRICENTIS
Accelerate 2016
12. & 13. September | Austria Center, Wien

Jetzt anmelden!
tricentis.com/accelerate-2016