Monday – Workshops and Tutorials

Time	Sigmund Freud	Gustav Mahler	Karl Kraus	Max Reinhardt 1	Max Reinhardt 2	Max Reinhardt 3	Arnold Schönberg
09:00	Welcome						
09:15 10:45-11:15 Coffee Break	Story Maps in practice: enable early feedback to build what really matters Christian Hassa (Tutorial)	Leading Self-Organizing Teams Andrea Provaglio (Workshop)	RefTest2013 – International Workshop on Refactoring & Testing Francesca Arcelli Fontana, Steve Counsell, Alessandro Murgia, Roberto Tonelli (Academic Workshop)	Lean Startup in Practice Manuel Küblböck, Doreen Timm (Workshop)	Restructuring Code: From “Push” to “Pull” Nicole Rauch, Andreas Leidig (Workshop)	Implementing Continuous Delivery Patrick Kua, Christian Trabold (Tutorial)	PhD Symposium (Part 1) Organizer: Johanna Hunt
12:45	Lunch						
13:45 15:15-15:45 Coffee Break	Specification by Example Christian Hassa (Tutorial)	Seeding New Organizational Culture Olaf Lewitz, Steve Holyer, Ivana Gancheva (Workshop)	RefTest2013 – International Workshop on Refactoring & (continued)	Lean Startup in Practice (continued)	Property-Based Test Driven Development Nat Pryce (Workshop)	Implementing Continuous Delivery (continued)	Research Dojo: Collaborative Approaches for our Agile Community Johanna Hunt, Xiaofeng Wang (Workshop)
17:15	Break						
	Welcome Reception						

Tuesday – Conference Day

Time	Sigmund Freud	Gustav Mahler	Karl Kraus	Max Reinhardt 1	Max Reinhardt 2	Max Reinhardt 3	Arnold Schönberg
09:00	Opening						
09:15	Opening Keynote: Listen Up! *David Anania*						
10:15	Tuesday Session Introductions						
10:30	Coffee Break						
11:00	*Exec & Managers* Effective planning and uncertainty management across the enterprise *Hendrik Esser* The Software Development Improvement Program: Agile Practices and Training at ABB *Per Branger*	*Vendor Track Sessions* Agile Fixed Price *Dr. Nikolaus Kawka and DI Richard Brenner, Zuehlke Engineering*	The Extreme Hour Revival *Erik Lundh*	*Teaching and Learning* Barriers to Learning in Agile Software Development Projects *Jeffry Babb, Rashina Hoda and Jacob Nørbjerg* Early Start in Software Coaching *Thomas Vikberg, Arto Vihavainen, Matti Luukkainen and Jaakko Kurhila* Introducing Programmers to Pair Programming: A Controlled Experiment *A. S. M. Sajeev and Subhajit Datta*	Growing Coaches for the Future *Mark Summers*	Agile Leadership in Action - Workout Practices for Leading Self-Organized Teams *Michael Leber, Ángel Medinilla*	Growing software from examples *Seb Rose*

12:30	Lunch						
13:30	*Exec & Managers* Extending agile into the rest of the organization *Gregory Yon* Agile Working Groups (AWGs) – A Sustainable Engine of Enterprise Agile Transformation *Jorgen Hesselberg*	Vendor Track Sessions	Rapid Software Architecture Exploration *Michael Keeling*	*Development Teams* Team Performance in Agile Development Teams: Findings from Focus Groups *Torgeir Dingsøyr and Yngve Lindsjørn* The Practice of Not Knowing for Sure: How Agile Teams manage Uncertainties *Denniz Dönmez and Gudela Grote* Key Challenges of improving agile teamwork *Nils Brede Moe*	Agility and Flow: Enablers, Influencers and Impediments of Organization Flow *Ken Power*	Agile Career Development from the trenches – lessons learned *Ionel Condor*	Testing with a stranger *Markus Gärtner*
15:00	Coffee Break						
15:30	*Exec & Managers* Fishbowl Discussion: Agile organizations generating higher profit, innovation and growth	The camel and the needle eye – or the challenges of integrating agile approaches into a standard system development process *Peter Faßbinder*	*Lightning Talks*	We don't think the way we think *Manuel Kiessling*	*Panel Discussion* The Impact of Agile on Technical Debt	The Scrum Principle – managing beyond Scrum *Boris Gloger*	Scenario based testing- multiple browsers, devices, environments *Jodok Batlogg*

16:30	Break
16:50	Open Space Opening
18:30	Agile Connect - the large crowd ballpoint-game
19:00	Open Space Night

Wednesday – Conference Day

Time	Sigmund Freud	Gustav Mahler	Karl Kraus	Max Reinhardt 1	Max Reinhardt 2	Max Reinhardt 3	Arnold Schönberg
09:00	Morning News						
09:15	Wednesday Session Introductions						
09:30	Open Space	Open Space	Distributed Product Owner Team for an agile medical development *Andrea Heck*	*Agile Practices* Effects of Negative Testing on TDD: An Industrial Experiment *Adnan Causevic, Rakesh Shukla, Sasikumar Punnekkat and Daniel Sundmark* Investigating the Impact of Experience and Solo/Pair Programming on Coding Efficiency: Results and Experiences from Coding Contests *Dietmar Winkler, Martin Kitzler, Christoph Steindl and Stefan Biffl*	BE BRAVE *Gitte Klitgaard*	Serious Games for change and engagement *Cesario Ramos*	How to Narrow Down What to Test *Zsolt Fabok*

10:30	Coffee Break						
11:00	Open Space	Open Space	How Agile Localization can be a competitive business advantage *Vasco Duarte*	*Experiences and Lessons Learned* Visualizing and Managing Technical Debt in Agile Development: an Experience Report *Paulo Sérgio Medeiros Dos Santos, Amanda Varella, Cristine Ribeiro Dantas and Daniel Beltrão Borges* How are Agile Methods and Practices Deployed in Video Game Development? A Survey into Finnish Game Studios *Jussi Koutonen and Mauri Leppänen* Inter-Organizational Co-Development with Scrum: Experiences and Lessons Learned from a Distributed Corporate Development Environment	Designing the Best Process for your Team *Michael Keeling*	Our Scrum is Great! (And Our XP Practice Ain't Bad.) Now what? *Steve Holyer & Diana Larsen*	Brutal Refactoring Game *Adrian Bolboaca*

				Raoul Vallon, Stefan Strobl, Mario Bernhart and Thomas Grechenig			
12:30	Lunch						
13:30	Keynote: Make Impacts, Not Software *Gojko Adzic*						
14:30	Break						
14:50	Open Space	Open Space	Multi-project management with dispersed agile teams *Maximilian Hantsch-Köller*	Self-Experimentation for Continuous Improvement *Tom Perry*	*Panel Discussion* Collaboration Tools for Agile: Learning and Practice	Agile Contracts – Can We Make Them Possible? *Andrea Provaglio*	Discussion and Q&A to “Impact Mapping and Story Mapping” *Gojko Adzic, Christian Hassa*
15:50	Coffee Break						
16:00	Open Space		The Art and the Real Option of Doing Nothing and Trusting People, *Steve Holyer and Olaf Lewitz*	Open Space			Pecha Kucha – Your Presentation Skills *Erik Lundh*
17:30	Break						
ca 19:00	Leaving for Conference Dinner						

Thursday – Conference Day

Time	Sigmund Freud	Gustav Mahler	Karl Kraus	Max Reinhardt 1	Max Reinhardt 2	Max Reinhardt 3	Arnold Schönberg
09:00	Morning News						
09:15	Keynote: Agile development outside the SCRUM team: integrating professional cultures *Helen Sharp*						
10:15	Thursday Session Introductions						
10:30	Coffee Break						
11:00	Open Space	Open Space	1000 Words – Illustrating Project Challenges with Visuals, *Tarang Baxi*	*Large Scale Projects* A Metrics Model to Measure the Impact of an Agile Transformation in Large Software Development Organizations *Jeanette Heidenberg, Max Weijola, Kirsi Mikkonen and Ivan Porres* Perspectives on Productivity and Delays in Large-Scale Agile Projects *Deepika Badampudi, Samuel*	Coding Dojo Challenge: Refactoring *Emily Bache*	Retrospectives as a tool for change management *Jutta Eckstein and Diana Larsen*	Do you remember how to play? *Jakob Wolman*

				A. Fricker and Ana M. Moreno Continuous Release Planning in a Large-Scale Scrum Development Organization at Ericsson *Ville T. Heikkilä, Maria Paasivaara, Casper Lassenius and Christian Engblom*			
12:30	Lunch						
13:30	Open Space	Open Space	*Panel Discussion* Where is Extreme Programming (XP) Today?	*Architecture and Design* Micro Patterns in Agile Software *Giulio Concas, Giuseppe Destefanis, Michele Marchesi, Marco Ortu and Roberto Tonelli* Feature Usage Diagram for Feature Reduction *Sarunas Marciuska, Cigdem Gencel, Xiaofeng Wang and Pekka Abrahamsson*	Thinking about TDD: A Russian Doll, Full of Clockworks *Erik Lundh*	StarWars! Jedi vs. Sith – hands on Agile vs. Waterfall *Olaf Lewitz*	Product Portfolio Game *Antti Kirjavainen*

				The Effect of Complexity and Value on Architecture Planning in Agile Software Development *Michael Waterman, James Noble and George Allan*			
15:00	Coffee Break						
15:20	Closing Open Space						
15:50	Break – Change Room Setup						
16:00	Closing Keynote: Agile Fluency: What We've Learned About How We Learn *James Shore*						
17:00	End of Day						

Friday – Workshops and Tutorials

Time	Sigmund Freud	Gustav Mahler	Karl Kraus	Max Reinhardt 1	Max Reinhardt 2	Max Reinhardt 3	Arnold Schönberg
09:00 10:30-11:00 Coffee Break	Solution Focused Coaching and Retrospectives *Ralph Miarka, Veronika Kotrba* (Tutorial)	Complex Projects aren't plannable but controllable *Jutta Eckstein* (Workshop)	Integrating Agile and User-Centered Design *Dina Salah* (Academic Workshop)	Fast Can be Safe: Coaching Agile Medical Software Development *Nancy Van Schooenderwoert, Brian Shoemaker* (Tutorial)	Designing good test cases *Emily Bache* (Workshop)	The Taming of the Script *Emmanuel Gaillot and François de Metz* (Tutorial)	PhD Symposium (Part 2) *Organizer: Johanna Hunt*
12:30	Lunch						
13:30 15:00-15:30 Coffee Break	Take Your Agile Adoption to the Next Level *Steve Holyer, Diana Larsen* (Workshop)	Agile Kaizen *Ángel Medinilla* (Tutorial)	AgileES2013 *Xiaocheng Ge, Nour Ali, Xiaofeng Wang* (Academic Workshop)	Fast Can be Safe: Coaching Agile Medical Software Development (continued)	Driving Development With Tests *Emily Bache* (Workshop)	The Taming of the Script (continued)	Research Challenges in Large Scale Agile Development *Torgeir Dingsøyr, Nils Brede Moe* (Academic Workshop)
17:00	End						

Monday – full day (09:15 – 17:15)

Lean Startup in Practice

Manuel Küblböck and Doreen Timm

Workshop

Summary

Keen to find out what the Lean Startup buzz is all about? This workshop is for you. Apart from a bit of theory and an experience report, this session focuses mainly on the handson application of the Lean Startup ideas; How can you quickly validate your assumptions about the product and the market? How can you define experiments, whose results can actually serve as a basis for deciding if it makes sense to continue with or change the current strategy? In this workshop, you have the opportunity to get your own impression of Lean Startup by applying the methodology yourself.

Presenters

Manuel Küblböck develops software products. He is constantly on the lookout for ways to do this more successfully and in a shorter amount of time. He believes that making the feedback cycle from customers to developers even shorter is the key to doing so. In this quest he has used agile and lean practices and methodologies for many years, while continuously adding new ones to his tool box. He appreciates the Lean Startup method for understanding a business model as a set of assumptions that need to be validated by experiments, rather than a plan that is set in stone.

Twitter: @manuelkublbock

Doreen Timm works as Product Owner and coaches her customers in similar roles. Her background is in design and product management. When she started to work for it-agile, she discovered her passion for agile software development, especially when combined with Lean Startup and UX techniques.

Twitter: @doreenitent

Implementing Continuous Delivery

Patrick Kua and Christian Trabold

Tutorial

Summary

With agile software development, building software is often no longer the bottleneck. More often than not, the bottleneck is in getting that software production. Continuous Delivery aims to address the “last mile” software houses often have. In this tutorial, we will explore through a series of short talks, facilitated discussions and some hands-on practice some of the ideas behind Continuous Delivery. Leave this tutorial with a greater set of skills to apply in your own organisations. Participants should bring laptops (all platforms supported) for part of the session.

After this tutorial, we expect participants can:

- Describe what Continuous Delivery is
- Draw the organisational support required for successful Continuous Delivery
- Setup a build pipeline
- Identify and add stages to a build pipeline
- Describe the impact Continuous Delivery as on other aspects to the software process
- List out tools that can help automate infrastructure
- Describe strategies for safe rollouts
- Identify data migration strategies
- Describe a set of antipatterns for Continuous Delivery

Presenters

Patrick Kua is a frequent speaker at conferences, and is author of The Retrospective Handbook. When he is not sharing with the community, he works as Tech Lead for ThoughtWorks helping clients deliver better software. You can find him on twitter as @patkua or read the articles on his blog, http://www.thekua.com/atwork

Christian Trabold works as a Senior DevOps Consultant for Thoughtworks helping customers implement Continuous Delivery. He's been working in the industry for over ten years, is passionate about pushing new ideas and forward thinking in the areas of DevOps and infrastructure automation and enjoys writing books like the TYPO3 Kochbuch published by O'Reilly.

You can find out more by following him on Twitter @ctrabold, reading his blog at www.christian-trabold.de or following his open source contributions on GitHub.

RefTest2013 – International Workshop on Refactoring & Testing

Francesca Arcelli Fontana, Steve Counsell, Alessandro Murgia and Roberto Tonelli

Academic Workshop

Summary

Refactoring and its testing implications/crossover have emerged over recent years to become important inter-related research topics with high industrial resonance. Many issues and problems still remain unexplored in these two highly-related fields incorporating topics such as the theoretical underpinnings of refactoring, Test Driven Development, empirical studies, refactoring of test artefacts, code smell analysis and patterns (both design and micro) to name just a few. Refactoring also has a growing importance in the monitoring of system's evolution and the propensity of systems to minimise maintenance effort and fault propensity; the over- riding question, still largely unanswered, is whether we can quantify the benefits of refactoring.

The purpose and goal of the RefTest Workshop is to bring together industrial practitioners and academics in a setting where current issues in refactoring and test crossover can be presented, relationships with testing discussed, results from current research in refactoring/testing disseminated and future directions distilled.

Topics of interest for the RefTest Wien workshop are as follows:

- Industrial System Experiences with Refactoring/TDD
- Pair Programming for Refactoring and Writing Tests
- Refactoring Theory and Models
- Empirical Studies of Refactoring (OSS)
- Refactoring Test artefacts
- Code Smell Evaluation and Eradication
- Design Patterns and Refactoring
- Micro-patterns and Refactoring
- Refactoring and System Evolution
- Software Metrics for evaluating the Refactoring process

Researchers and practitioners are invited to submit full papers (approximately 8-12 pages) or position/discussion papers (max 4 pages) in Springer format, guidelines: http://www.springer.com/series/7911 format. All submissions will be peer-reviewed. At least one of the authors of accepted papers should attend the Workshop to present the paper. You may submit papers at http://reftest.wordpress.com/submission/

We welcome academic and industrial contributions.

We are currently in discussions with a primary Software Engineering Journal, indexed in Scopus and Web of Science, to produce a special issue from selected papers from the Workshop.

For more information, visit the workshop website

Questions about the workshop may be addressed to roberto.tonelli@dsf.unica.it

Organizers

Francesca Arcelli Fontana has received her master and PhD degrees in Computer Science at the University of Milano, Italy. She currently holds the position of Associate Professor at the University of Milan -Bicocca. Her actual research activity principally concerns the software engineering field. In particular, her attention is focused on software evolution and reverse engineering, design patterns detection for reverse engineering, program comprehension, and system migration towards SOA architectures.

Alessandro Murgia received the laurea degree in Electronic Engineering in 2007, and his Ph.D. in Information Engineering in 2011 at University of Cagliari. He was post-doc for two years at the same department, and visiting student at Brunel University for three months. He is currently enrolled at the Antwerp University. His research interests are data mining from software repositories, software metrics, software bugs, and text analysis.

Roberto Tonelli received the laurea degree in Physics in 1995 and his Ph.D. in Physics in 2000, working on complex dynamical systems. He received his second Ph.D. in Software Engineering. He has been visiting researcher at University of Maryland, and post- doc researcher at University of California Berkeley. He is currently with the Electrical and Electronic Department at University of Cagliari – Italy, working on complex software networks and software quality. He has published over 60 Conference and Journal papers.

Steve Counsell was awarded his PhD in 2002 from the University of London where he was a lecturer from 1998. He moved to Brunel University in 2004 and has published over 135 Conference and Journal papers in the areas of software metrics, refactoring and software engineering topics in general. Steve is a Fellow of the British Computer Society and much of his research is in collaboration with industrial companies and organisations. He is currently a Senior Lecturer at Brunel and is involved in a number of grants awarded from the UK research councils.

Monday morning (09:15 – 12:45)

Story Maps in practice: enable early feedback to build what really matters

Christian Hassa

Tutorial

Summary

Build – Measure – Learn is one of the most important mechanisms of agile software development. However, this mechanism is often crippled in nowadays projects, where traditional approaches of requirements gathering are bloating up product backlogs that cannot be prioritized anymore in a meaningful way. The results are customers not interested in iteration results, release to production that happens only at the end of the project, and feedback from customers when it is already too late and the budget is burned up.

Story mapping is a method that aligns user stories along desirable outcomes, so that customers can give sooner meaningful feedback, and release to production can happen earlier. The method helps slicing and prioritizing user stories, and addresses the product design aspect that is missing when just working with a product backlog. The method is highly visual and facilitates shared product ownership among product owner, team and customer.

This workshop gives a hands-on introduction to story mapping. Participants will be introduced to the concept and build their first story map during the workshop. The workshop concludes with things I've learned so far in using story maps in own projects, and what further benefits story maps can provide.

Presenter

Christian Hassa is specializing in agile requirements engineering methods, particularly story mapping and specification-by-example. He is working for several years as Product Owner and Scrum Coach, mainly in the public and financial sector. As managing partner at TechTalk, a .NET services and consulting company focusing on agile delivery, his role is leading the evolvement of methodology and practices and facilitating their application in projects and at clients.

Leading Self-Organizing Teams

Andrea Provaglio

Workshop

Summary

Have you adopted an Agile approach, but people in your team are still not showing the level of participation and commitment you were hoping for? Decisions-making is still not working right? Leadership is confused? The team's morale is not taking off? Development is still deadline-driven?

Then you are not reaping many of the benefits that an Agile mindset may bring into your organization.

Attend this workshop to learn — through short lectures, open discussions and practical exercises — why Self-organization is so important, what factors may prevent it from happening and how to create an environment that fosters it.

Presenter

Andrea Provaglio is an independent professional in the field of Agile/Lean software development, with 20+ years of experience in three different continents.

I help my clients to develop better products and to create better teams; and I support individuals and teams who want to improve technically and relationally.

My clients are from different domains, ranging from large organizations such as the European Commission in Brussels, to small and dynamic IT companies.

I currently work in Europe; I also worked for four years in the U.S. on a O-1 visa for “extraordinary abilities in Science”.

More information at: http://andreaprovaglio.com/about

Restructuring Code: From “Push” to “Pull”

Nicole Rauch and Andreas Leidig

Workshop

Summary

Developers often experience code that is both hard to read and to track. At the same time, they strive to write “clean code” that follows certain principles like single responsibility, encapsulation and coherence. While most developers know these principles by name, they often find it hard to establish them, especially in legacy code. They feel the need to restructure parts of the application, but they have no idea how they could perform such a large-scale restructuring, and sometimes they might not even have an idea of what the resulting code should look like. One can find lots of information on how to perform small refactorings in legacy code, but there is almost no guidance for performing large restructurings and redesigns.

Our strategy addresses a specific coding pattern, which we have found in many applications: Fields in objects are written from multiple locations in the code. This has various effects: The value of a field at a given time is not clear, and tracking back all the methods that write the field is difficult. Also, it is hard to tell how and when the results have been determined because there is no unique location in the code where the transformation from the input values to the individual output values takes place. We call this way of coding “Push” because the code actively pushes its results into some object’s fields. These results are determined in a procedural and sequential way. The resulting code is rather fragile and tends to be buggy. Changing and extending the code is difficult and often turns into a “Shotgun Surgery”.

Instead of using the “Push” style, the output values can be calculated on-demand when retrieving them. This way, the algorithms for each value are clearly visible and very descriptional. We call this style “Pull” because the values are pulled by other parts of the code. The resulting code has proven to be readable, understandable, testable, changeable and extensible.

This workshop intends to give a concrete example of how a large restructuring can be performed in a systematic, structured and reproducible way, taking small and safe steps. The participants can directly experience its advantages by implementing “Pull” style code themselves. If this strategy is carried out consequently, the resulting code follows the abovementioned principles and becomes object-oriented, with small methods in objects communicating with each other, in one word: the code has become “clean code”.

Participants should have one computer per pair, containing a Java IDE. There should additionally be one keyboard and mouse per pair.

Presenters

Nicole Rauch is a software developer with a solid background in compiler construction and formal methods. Over the last two years, she worked on the restructuring of large legacy code applications with different technologies. Alongside her occupation as developer, she took part in organizing some self-organized conferences related to agile coaching.

Andreas Leidig has broad experience in object oriented software development and agile practices. He started programming years ago with Smalltalk and worked as an agile coach for a few years. Nowadays he is working full-time in developing enterprise software products.

Nicole and Andreas co-operate at work and in user group activities. Their passion for producing great software led them to organize a conference on Software Craftsmanship, known as SoCraTes. Subsequently they initiated Softwerkskammer, a Germany-wide user community for software craftsmanship, and Nicole founded the regional Softwerkskammer branch in Karlsruhe.

PhD Symposium

Monday afternoon (13:45 – 17:15)

Specification by Example

Christian Hassa

Tutorial

Summary

Many teams struggle with the implementation of user story acceptance criteria and having a shared understanding about the expected story outcomes. This often results in missed stakeholder expectations, ad-hoc assumptions made by the team during implementation and conflict between team members and the product owner around testing.

In this tutorial, you will learn how specification-by-example and acceptance test driven development will address team conflict, missed stakeholder expectations and overall increasing the level of clarity on the project end-to-end. The presentation will cover the theory behind ATDD, case-studies and practical experience from real projects and several hands-on exercises to try out the presented concepts.

You will leave this tutorial with a fundamental understanding of specification-by-example and its benefits, as well as concrete pointers on how to get started using it in your own projects.

Presenters

Christian Hassa is specializing in agile requirements engineering methods, particularly story mapping and specification-by-example. He is working for several years as Product Owner and Scrum Coach, mainly in the public and financial sector. As managing partner at TechTalk, a .NET services and consulting company focusing on agile delivery, his role is leading the evolvement of methodology and practices and facilitating their application in projects and at clients.

Seeding New Organizational Culture

Olaf Lewitz, Steve Holyer and Ivana Gancheva

Practitioner Workshop

Summary

Most Agile transitions fail. Apparently something devours the benefits for breakfast … it’s culture. Instead of facing that beast head-on, we will guide you on a path to tame it and let it work with you.

We will start with simple frames to help you understand what you are dealing with and choose to aim high and transform your organization’s culture together.

You will learn how to tame your own beast with five different approaches for transforming it into a new culture. Two will be experienced in practice: You will learn how your community can create a persona for the organizational identity. This helps everyone as an aspirational model for your transformation. Finally, you will create and share a personal vision, as each of us knowing what we want is key to be able to get it.

Presenters

Ivana Gancheva is an Agile developer, project manager and business influencer. She establishes links between developers and business. She is a passionate leader and community contributor. She is a relentless critical thinker who focuses on listening between the lines. She strives for continuous improvement, inspires and links people. She lives in Trondheim, and adores rollerblading.

As a developer, **Steve Holyer** was initially sceptical of Agile development, but embraced it enthusiastically when he realized Scrum allowed him to deliver better software in a saner way. He is a “by the book” scrum master who doesn’t want to stop at “scrum by the book”. In his ongoing journey he has worked to master the art the “Scrum master”, learned the importance of XP practices and advised lean startups in the technical and social aspects of doing agile development with remote teams. Steve works with agile42 coaching organisations and teams finding their path to truly transformative agile fluency.

Olaf Lewitz is an independent agile coach and linchpin. He inspires people to improve the way they work. He inspires managers to make their organisations more effective. His motto is that of Nanny McPhee: “When you need me, but do not want me, I must stay. When you want me, but no longer need me, then I have to go.” You can find him on Twitter as @OlafLewitz.

Property-Based Test-Driven Development

Nat Pryce

Practitioner Workshop

Summary

A hands-on programming workshop (i.e. bring your laptop) to explore how property-based testing influences the TDD process & the designs it creates.

The TDD/BDD community has not embraced property-based testing, despite tools being available for most programming languages. TDD practitioners “triangulate” a property of the system by testing it against multiple examples, selected manually. Triangulation does not explicitly define the property being tested, giving little scope for automating the selection of test data. It is easy to write tests with poor condition/decision coverage. People reading a test must infer the property from the data and name of the test, which can be difficult.

The property-based testing literature has focused on testing for quality assurance. It does not describe property-based testing tools being used to drive development, focus attention, measure progress, and obtain design feedback. (Saff & Boshernitsan’s article on JUnit Theories is a notable exception).

We will investigate:

- How do we go from examples to properties? Must we start with examples and generalize to properties, or can we start with properties in some cases?
- How do we change properties as the system grows? Do we start with general properties and specialize them, or specialized properties and generalize? How does that compare to adding examples?
- Can property-based testing drive design? If we focus on properties, do we get different designs than if we focus on examples? Are some design styles easier than others to describe in terms of properties?
- Does property-based testing work better when working topdown/outsidein or bottomup/insideout?

Presenter

Nat Pryce is a co-author of Growing Object-Oriented Software Guided by Tests. An early adopter of XP, he has written or contributed to several open source libraries and tools that support TDD and was one of the founding organizers of the London XP Day conference. He has worked as a programmer, architect, trainer, and consultant in a variety of industries, including sports reportage, marketing communications, retail, media, telecoms and finance. He has delivered systems ranging from embedded devices to large compute farms supporting global business.

Research Dojo: Collaborative Approaches for our Agile Community

Johanna Hunt and **Xiaofeng Wang**

Academic Workshop

Summary

In Coding Dojos developers gather to learn from each other and improve their skills via deliberate practice. For this Research Dojo we encourage attendees of XP2013 to join us to do the same and collaboratively design research.

There are many similarities between software development and conducting research. Both are knowledge intensive: how to come up with a good idea (either a software product or a research question to be answered), how to implement the idea and how to evaluate it. Both development and research are also collaborative processes, involving communicating and cooperating with colleagues and peers.

Agile approaches have yet to be effectively applied in the research world. Even researchers studying agile methods typically do not use such practices purposefully in their daily work. If agile practices are effective for collaborative work in software development teams, then why not use them also for research? How can we conduct research into agile without applying the lessons learned to our own work?

We are particularly interested to explore:

- parallels and differences between conducting research and developing software
- approaches to conducting small iterative pieces of research in practice
- using agile ideas and values (such as increased feedback loops) to improve research practice

The primary objective of this workshop is to explore and demonstrate how to use agile practices to conduct research collaboratively. This is not a standard Academic workshop however. This is a call to come and collaborate, no matter what your background, and help design a short piece of research with us.

Organizers

Johanna Hunt, Eventyr Limited, has been sharing her general willingness to help out with the UK and international Agile community since 2006. She has presented and facilitated sessions across the US and Europe on subjects ranging from the advantages of agile approaches for mobile application user experience design through to the use of traditional storytelling approaches for retrospectives, and is currently focussed on coaching strategies for 'large problems'. She has yet to find time to complete her PhD (on the impact of Agile transformation on team and organisational culture) due to her passion for team facilitation and coaching out in the 'real' world.

Xiaofeng Wang, Free University of Bozen-Bolzano, is an agile researcher. Her PhD study was focused on understanding agile practices through the perspective of complex adaptive systems. After the completion of PhD she has been actively involved in agile research community and served various XP and Agile conferences. In particular she has run a workshop at XP2009 for both agile researchers and practitioners to review the history of agile research and explore the emerging topics for future research. It is always an intriguing question for her as how to better conduct agile research that is both relevant to practice and of high quality.

Tuesday

Opening (09:00 – 10:30)

Listen Up! Hearing versus Listening: A Distinction in Being Human

David Anania

Keynote

Summary

Have you ever been "deep in conversation" with someone—whether an old friend or business colleague or someone you've just met—and have had absolutely no idea what they were talking about…because you weren't listening? Think of all of the moments that have gone by in your life—personal and professional—where you may have passed up on valuable information: instructions on completing a task, a different approach to reach a desired goal…someone saying "I love you". Priceless information that could potentially be life changing…and, even if we pretended to…

we didn't listen.

David Anania (Blue Man Group, Della Miles, MamaMiaMusic) draws from a unique blend of music, life experiences and hands-on participation to create a clearing for understanding the difference between "hearing" and "listening" and how we, as human beings, can become present, engaged and inspired by potential from the countless connections that we experience in our everyday lives.

Presenter

David Anania — drummer, singer/songwriter, producer—has been a member of Blue Man Group since 2000 (NY, Berlin) and has performed with BMG all over the world, including: Wango Tango Music Fest (2001, Dodger Stadium in front of 45,000 people), Moby's Area II Tour (2002, w/Moby, David Bowie and Bustah Rhymes), The Complex Rock Tour and The Royal Variety Performance (in front of Queen Elizabeth II). In August of 2010, David co-wrote drum music for Blue Man Group's world premiere performance of "We Are One" a composition by internationally renowned DJ, Paul van Dyk (o2 Arena, Berlin).

David is also an accomplished composer in his own right, having completed his solo EP "Reinvent the Feel" in 2010 and, in Autumn of 2012, scored, performed and produced the original soundtrack for the full-length documentary, "Return to Pearl" (Snap 180 Media). Outside of his rhythmitizing for Blue Man Group, David drums for Della Miles (Whitney Houston) and Jaimi Faulkner (2008 Australian Blues/Roots Performer of the Year) and is a sought-after solo-drum performer ("Timetraveler" Performance) and clinician for DrumCraft Drums, Roland V-Drums, Paiste Cymbals and Vic Firth Drumsticks.

Since June of 2009, David has been orchestrating successful Team Building Events throughout Western Europe, combining his specialized musical and leadership abilities to create an atmosphere where a group of people can come together, leave their inhibitions at the door and boldly go where no other team has gone before.

Tuesday morning (11:00 – 12:30)

Effective planning and uncertainty management across the enterprise

Hendrik Esser

Manager, Portfolio and Technology Management for the Mobile Core, Ericsson (Germany)

Summary

“Life is what happens while you are busy planning it”. While planning is important, it at the same time needs to be effectively adaptive to changes.In a large agile organization planning needs to be done on many levels, starting from the (company) strategic level over product portfolio and product releases down to features and sprints. The complex task of planning is distributed over different roles and teams across an organization. There are practices for the planning on all levels. However, often it is not understood how the plans interact with each other and how all plans converge to an overall, continuous planning flow.This session will explain how such a large planning flow can work. It is based on the practical experience from agile portfolio management in a large organization over 2 ½ years.The session will cover the principles of planning at scale, an overview of the different levels of planning, the interaction between the different levels and how the feedback coming from the implementation of plans and strategies is used for learning, refinement and adaptation.

Presenter

Hendrik Esser has worked as a developer, project manager and in a variety of management roles at Ericsson since 1994. In 2008 he was a key contributor to the agile transition of the 2000 people organization he works in. He today supports the Ericsson enterprise transition to lean and agile by consulting and teaming up with other parts of Ericsson. He also is active as a speaker at agile conferences and as a member of the Agile Alliance’s Supporting Agile Adoption work group.

The Software Development Improvement Program: Agile Practices and Training at ABB

Per Branger

Manager, ABB (Sweden)

Summary

Software content is often a differentiating factor for the products we sell. Software development must be agile and follow best-in-class practices. ABB’s Software Development Improvement Program (SDIP) offers free classes for these best-in-class software solutions. Whether a manager needs to get a new employee up to speed, help a current employee increase productivity or cross-train a member of their staff, the SDIP classes are the starting point. SDIP provides engaging instructors, research-backed training methods, and a variety of delivery methods. All classes align with the 11 basic practices to which all software development groups within ABB are expected to adhere.

Presenter

Per Branger works in ABB to improve software development, from small teams to relatively large teams in a geographical distributed environment. He has experience from different roles, as software developer, product manager and line manager.

Agile Fixed Price

Dr. Nikolaus Kawka and DI Richard Brenner

Presentation: 45 minutes

Summary

In this talk, Zühlke will introduce customer/supplier collaboration models for agile projects and ways to create fixed price contracts without losing the advantages of agile methods.

We look at the variables: budget, feature set and time and will show contract constructs where one of the variables, preferably feature set can stay variable.

Topics like „money for nothing”, “changes for free” are examined and explained. Zühlke can go back here to extensive project experience in Switzerland, Austria and Germany.

Join the conversation: blog.zuehlke.com

Current Events: zuehlke.com/events
Current Videos: youtube.com/user/zuehlkeengineering

Barriers to Learning in Agile Software Development Projects

Jeffry S. Babb, Rashina Hoda, and Jacob Nørbjerg

Research Paper

Summary

The adoption of agile methods promises many advantages for individual, team, and organizational learning. However, environmental, structural, and organizational/cultural constraints often find teams adapting agile software development methods rather than engaging in full adoption. We present results from two quantitative studies of teams and organizations that have, in many cases, adapted agile software methods to suit their needs through the omission or alternation of aspects of the method. In many cases, aspects of an agile method that is most related to learning were those that were modified or omitted. This paper utilizes the results of these studies to identify common and emergent barriers to learning. Often these barriers to learning exist according to organizational culture and the extent to which that culture influences attitudes, norms, and behaviors pertaining to learning. We present these barriers to learning and provide insight to the causes, effects, and potential ameliorations for these barriers.

Presenter

Jacob Nørbjerg is an Associate Professor at the Department of IT Management, Copenhagen Business School. His research is directed towards the organization and management of systems development, and software process improvement. He currently works on a project about software development practices in very small companies. Much of his research is carried out in close collaboration with industrial partners.

Department of IT Management, Copenhagen Business School, 60 Howitzvej, 2000 Frederiksberg, Denmark, jno.itm@cbs.dk

Further Authors

Jeffry Stephen Babb is an Assistant Professor of Computer Information Systems in the College of Business at West Texas A&M University. He has a PhD in Information Systems from Virginia Commonwealth University in Richmond, Virginia. His research interests are in software systems development, small-team development, mobile applications development, and information systems security.

Department of Computer Information and Decision Management, West Texas A&M University, 2403 Russell Long Blvd. Canyon, Texas USA, 79016, jbabb@mail.wtamu.edu

Dr. Rashina Hoda leads the SEPTA research group in the Electrical and Computer Engineering department at The University of Auckland, New Zealand. Her research interests include Agile software development, self-organizing teams, and software project management. Rashina received her PhD in Computer Science from Victoria University of Wellington, NZ. More: www.rashina.com

Electrical and Computer Engineering, The University of Auckland, 38 Princes St,
Auckland, New Zealand, r.hoda@auckland.ac.nz

Early Start in Software Coaching

Thomas Vikberg, Arto Vihavainen, Matti Luukkainen, and Jaakko Kurhila

Research Paper

Summary

Demand for software coaching and coaches is increasing. As our programming courses are organized according to the Extreme Apprenticeship method, it is relatively safe and straightforward to allow students to participate as coaches in our CS1 course even as early as during their second semester. Safety is ensured by the hierarchical structure of CS1 course personnel that ensures enough peer and faculty support for students undertaking the task of coaching. We briefly describe the Extreme Apprenticeship method as well as the organization and the learning objectives in our coaching environment. Results acquired from student coaches (N=46) indicate that the learning experience of coaching is highly valued and deemed especially educational.

Presenter

Matti Luukkainen has completed his Computer Science PhD on formal verification at University of Helsinki 2003. Most of his industrial experience originates from a major Finnish telecommunications company, where he worked during the waterfall era. To redeem his past, he has gathered hands-on agile software development experience by cleaning toilets and emptying trash cans in a bleeding-edge startup company focusing on bringing the best of cloud services to mobile applications. He currently evangelizes programming and agile software engineering best practices at the University of Helsinki, and is actively involved in researching and developing Computer Science Education, with a focus on people-oriented teaching approaches such as the Extreme Apprenticeship method.

Department of Computer Science, University of Helsinki, Finland, mluukkai@cs.helsinki.fi, http://www.cs.helsinki.fi/rage

Further Authors

Thomas Vikberg, Arto Vihavainen, and Jaakko Kurhila, Department of Computer Science, University of Helsinki, Finland, {tvikberg, avihavai, kurhila}@cs.helsinki.fi, WWW home page: http://www.cs.helsinki.fi/rage

Introducing Programmers to Pair Programming: A Controlled Experiment

A. S. M. Sajeev and Subhajit Datta

Research Paper

Summary

Pair programming is a key characteristic of the Extreme Programming method. Through a controlled experiment we investigate pair programming behaviour of programmers without prior experience in XP. The factors investigated are: (a) characteristics of pair programming that are less favored (b) perceptions of team effectiveness and how they relate to product quality, and (c) whether it is better to train a pair by giving routine tasks first or by giving complex tasks first.

Our results show that: (a) the least liked aspects of pair programming were having to share the screen, keyboard and mouse, and having to switch between the roles of driver and navigator (b) programmers solved complex problems more effectively in

pairs compared to routine problems, however, perceptions of team effectiveness was higher when solving routine problems than when solving complex problems and (c) programmers who started pair programming with routine tasks and moved on to complex tasks were more effective than those who started with complex ones and moved on to routine ones. We discuss how these results will assist the industry in inducting programmers without prior pair-programming experience into XP process environments.

Presenters

Professor **A. S. M. Sajeev** holds a bachelor degree with first class honours in Electrical Engineering from Cochin University (India), a MTech with Honours in Computer Science from the Indian Statistical Institute, Calcutta and a PhD in Computer Science from Monash University, Australia. He is a Fellow of the Institution of Engineers, Australia. Sajeev is currently the Chair in IT/Computer Science at the University of New England, Armidale, Australia. Previously, he has taught at the University of Newcastle, Australia and Monash University, Australia. His research interests include empirical software engineering, mobile systems and business information systems.

University of New England, Australia, sajeev@une.edu.au,
http://mcs.une.edu.au/~sajeev

Subhajit Datta has nearly a decade and half of experience in software development, design, research, and teaching across different organizations in India, the United States of America, and now Singapore. Subhajit received his Ph.D. and M.S in computer science from the Florida State University and his Bachelor of Electrical Engineering from Jadavpur University. He is the author of the books Software Engineering: Concepts and Applications (Oxford University Press, 2010) and Metrics-Driven Enterprise Software Development (J.Ross Publishing, 2007) and numerous research publications. Latest information about his interests and activities are available at: www.dattas.net

Singapore University of Technology and Design, Singapore, subhajit.datta@acm.org, http://www.dattas.net

Growing Coaches for the Future

Mark Summers

Workshop: 90 minutes

Summary

Do you want to understand where you are on your journey as a ScrumMaster or coach? Then come and explore. Maybe you need a way of helping ScrumMasters grow in your organisation? Then this session will provide you with a framework. We will take a number of key competencies of a ScrumMaster/coach and explore the journey from newbie to experienced coach. This will be an interactive session to explore yourself and share ideas to help each other develop as ScrumMasters and coaches. This is based on work done with ScrumMasters and coaches within organisations.

This session focuses primarily on how to grow coaches, taking a look at the competencies required for a coach and using that knowledge to support the development of ScrumMasters/coaches through a number of initiatives.

In such a development programme the first thing we need to do is to help them understand where they are on their journey, test first! We have developed a set of ScrumMaster/coaching competencies that can be used to self-assess with the aid of an experienced coach. In the session everyone will get the chance to use the competency framework. For each competency the attendees will assess themselves as Shu, Ha or Ri.

Following this group exercises will explore how these competencies can be developed.

We will then share our experiences of different initiatives we have run to help coaches grow. We will present data based on the competency framework showing the development of some ScrumMasters and coaches that we have been coaching.

Short Bio

Mark Summers is an Agile coach and Certified Scrum Coach. Mark is passionate about helping individuals, team and organisations be successful. Mark's goal is to too create world class development organisations and teams that deliver value to their customers through collaboration, sound engineering practices, empowerment and a dedication to quality.

Get more information about this session on IdeaScale.

Agile Leadership in Action - Workout Practices for Leading Self-Organized Teams

Michael Leber, Ángel Medinilla

Workshop: 90 minutes

Growing software from examples

Seb Rose

Workshop: 90 minutes

Summary

There are a wealth of methods that use specifications, examples and tests to drive out the design and implementation of software systems: TDD, ATDD, BDD, SbE and more. Beyond a common feeling that the use of the T-word (test) is unfortunate (because it serves to distort the intent and distract the focus of the practitioner) there is little agreement. A further impediment, from a development perspective, has been the partitioning of the techniques into business-facing (ATDD, BDD, SbE) and technical (TDD). All methods that make use of executable examples require the participation of developers and share a common subset of pitfalls and gotchas.

This session will demonstrate how we can use executable examples to productively grow software. We will examine simple, yet effective, techniques that should be used irrespective of the layer at which you are working, while highlighting concerns that are specific to the business and technical layers. We will emphasise that the techniques are frequently common to all the methods and point out the occasional exception. Tooling will only be discussed to the extent that it empowers a particular technique.

In conclusion, we will make a case for a more inclusive nomenclature that emphasizes the shared underpinnings of all example-based techniques.

Short Bio

Seb Rose is an independent software developer, trainer and coach based in the UK. He specialises in working with teams adopting and refining their agile practices, with a particular focus on delivering software through the use of examples. He first worked as a programmer in 1980 writing applications in compiled BASIC on an Apple II. He has worked with many mainstream technologies since then, for many well-known companies, such as Amazon, IBM, NCR, HBOS, Standard Life and Aegon. He is a regular conference speaker as well as a contributing author to O'Reilly's "97 Things Every Programmer Should Know"

Get more information about this session on IdeaScale.

The Extreme Hour Revival

Erik Lundh

Workshop: 90 minutes

Summary

A team of volunteers set up an agile workflow and define, plan, design and deliver two versions of a product within 2+4 timeboxes. The audience acts as a market asking for new features while they see the product growing.

At the very first XP2000 conference Peter Merel and Kent Beck convinced the brave organizers to do a virtually unknown live team exercise invented by Peter, an agile simulation where two teams built two versions of a product within 90 minutes of setup simulation and teardown. The Extreme Hour has since been one of our favourite agile simulations that many people have honed. The Extreme Hour agile simulation enables an agile team to set up. Defined, plan, design and deliver two iterations of a product where focus is on getting a product together within the timebox in a can-do problem solving mindset. In the early days we did a big Extreme Hour where a conference audience of 180 acted as the market, throwing in requirements during the simulation.

Short Bio

Erik Lundh has more than 25 years experience in software development. Erik has worked with mature innovation firms and start-ups, from small to large organizations such as Ericsson and ABB. Erik programmed industrial just-in-time (Lean) systems in the 1980’s, was a “process and management guy” in the 1990’s, and spent the fun part of the 2000’s as an agile evangelist and coach. In 2006 Erik was invited to Ericsson’s first major agile transformation of 2300 R&D people at 10 sites in 5 countries.

Get more information about this session on IdeaScale.

Tuesday afternoon (13:30 – 15:00)

Extending agile into the rest of the organization

Gregory Yon

Senior Product Engineer, Knowledge Management/Engineering Operations Systems, Johnson Controls (Germany)

Summary

Join Gregory Yon as he relates the tale of how agile moved throughout his organization:

- Story 1: “Let me tell you a story about scrum, on engineering level.”
- Story 2: “Let me tell you a story about scrum, on Simultaneous Development team level.”
- Story 3: “Let me tell you a story about scrum, on Senior Management level.”

Presenter

Gregory Yon is a Scrum Master and as a Senior Product Engineer working in the Knowledge Management/Scrum department as an Agile Coach and Kepner Tregoe Facilitator.

Before that he worked as a Six Sigma Black Belt in the Robust Engineering, Supply Chain Management, and IT (ERP support and programming) department. He holds a degree in Finance and Information systems.

He is part of the Engineering Operations Systems department in Europe of Johnson Controls Automotive Experience, a global leader in automotive seating, overhead systems, door and instrument panels, and interior electronics.

Agile Working Groups (AWGs) – A Sustainable Engine of Enterprise Agile Transformation

Jorgen Hesselberg

Senior Manager, Enterprise Agile Transformation, Nokia (Chicago US)

Summary

This talk covers the critical role, practical responsibilities and tangible deliverables of the Agile Working Group (AWG), a dedicated set of internal Agile people responsible for driving a sustainable transformation of Agile at the enterprise level. Drawing from practical, real-world experiences from Nokia and other leading companies transforming how they work and create products that delight the customer, this presentation aims to share ideas that can be applied immediately to any large organizations considering Agile

Presenter

Jorgen Hesselberg is the Senior Manager of Enterprise Agile Transformation at Nokia. With over fifteen years of experience in a variety of business environments from proprietary hedge funds to Fortune 50 companies, Jorgen specializes in helping organizations achieve value through leadership, passion and effective project management processes and techniques.

Team Performance in Agile Development Teams: Findings from 18 Focus Groups

Torgeir Dingsøyr and Yngve Lindsjørn

Research Paper

Summary

How to make teams perform well is increasingly important in software development, as agile development methods prescribe development in small teams. Team performance has been studied in a number of research fields, and there are many models of what enables team performance. A central question then is how relevant these models are for agile development teams. This article investigates the following research question: What factors do agile software practitioners perceive to influence effective teamwork, through a focus group study with 92 participants in 18 groups. The main findings are that what agile practitioners perceive foster and hinder team performance seems to comply well with what is stated in an existing research-based model. However, agile practitioners seem to place insufficient focus on backup behaviour. Agile practitioners place much emphasis on physical and technical infrastructure of the development team as enablers of team performance.

Presenters

Dr. Torgeir Dingsøyr works with software process improvement and knowledge management projects as a senior scientist at SINTEF Information and Communication Technology. He has published more than 60 refereed papers in this field, including articles in IEEE Transactions on Software Engineering, IEEE Software, Communications of the ACM, Information and Software Technology and Empirical Software Engineering. In particular, he has focused on agile software development through a number of case studies, co-authoring of a systematic review of empirical studies, co-editing of the book “Agile Software Development: Current Research and Future Directions” and co-edited the special issue on Agile Methods of the Journal of Systems and Software. He wrote his doctoral thesis on “Knowledge Management in Medium-Sized Software Consulting Companies” at the Department of Computer and Information Science, Norwegian University of Science and Technology, where he is now Adjunct Associate Professor.

SINTEF, NO-7465 Trondheim, Norway, torgeird@sintef.no

Yngve Lindsjørn has long experience from research, education and management within ICT, both in public and private sector. He is now a research scientist at the Department of Informatics at the University of Oslo. He works with software process improvement, with focus on effective teamwork in agile software teams. He is currently leading a research project on teamwork in software teams, with the University of Oslo, SINTEF and four software companies, funded by the Research Council of Norway.

University of Oslo, Norway, ynglin@ifi.uio.no

The Practice of Not Knowing for Sure: How Agile Teams manage Uncertainties

Denniz Dönmez and Gudela Grote

Research Paper

Summary

Uncertainties are ubiquitous in software development. They impact almost every aspect of a development project. Most uncertainties are viewed as threats to project efficiency and there are strong calls to their reduction. However, uncertainties can pose opportunities for creativity and innovation in some situations. The literature has been dominated by discussions that focus on requirements uncertainties. We aim to extend these discussions by drawing attention to additional types of uncertainties, namely resource, task, and output uncertainties. In this empirical study we

investigate the potential of agile software development methods to manage these different types of uncertainties, and examine the mechanisms available to development teams. Our results reveal how some agile teams seized mechanisms to harvest positive and mitigate negative impacts of uncertainties. Drawing upon these results, we discuss several antecedents of successful uncertainty management.

Presenter

Denniz Dönmez is a PhD researcher at the Organization Work and Technology Group at ETH Zurich's Department of Management, Technology, and Economics. He works under the supervision of Prof. Dr. Gudela Grote, and studies mechanisms of decision-making, collaboration and leadership in self-organising teams. A major focus of his work is risk and uncertainty management. He has been working with agile software development since 2010.

Department of Management, Technology, and Economics ETH Zurich, Switzerland, ddonmez@ethz.ch

Further Author

Gudela Grote, Department of Management, Technology, and Economics ETH Zurich, Switzerland, ggrote@ethz.ch

Key Challenges of improving agile teamwork

Nils Brede Moe

Research Paper

Summary

Inspect and adapt is essential to succeed with agile software development. Our objective was to understand the challenges software process improvement in agile software development teams. We designed a multiple case study consisting of five projects in three software product companies that applied Scrum. We collected data in semi-structured interviews. We found that long-term quality was often in conflict with short-term progress, specialization hinders self-management, process related problems are difficult to solve and there are major organizational barriers to self-management. The main conclusion drawn from this work is that software process improvement challenges in agile software development are the problems of increasing redundancy to create conditions for the team to self-manage, to learn how to learn, and to improving agile software development as a large long-term organizational change project.

Presenter

Dr. Nils Brede Moe works with software process improvement, agile software development and global software development as a senior scientist at SINTEF Information and Communication Technology. His research interests are related to organizational, socio-technical and global/distributed aspects. His main publications in the field of agile software development include several longitudinal studies on self-management and teamwork and co-editing of the books “Agile Software Development: Current Research and Future Directions” and “Agility Across Time and Space: Implementing Agile Methods in Global Software Projects.” Moe was also a co-organizing chair of the 11th International Conference on Agile Software

Development (XP2010) and co-producer of the research at work stage at Agile2011 and Agile India 2012. He wrote his thesis for the degree of doctor philosophiae on "From Improving Processes to Improving Practice – Software Process Improvement in Transition from Plan-driven to Change-driven Development."

SINTEF, Strindveien 4, NO-7465 Trondheim, Norway, nils.b.moe@sintef.no

Agility and Flow: Enablers, Influencers and Impediments of Organization Flow

Ken Power

Workshop: 90 minutes

Summary

Organization Flow is about achieving the Lean concept of Flow at an organization level. The complexity of enabling flow increases as we move from a single product focus to a cross-organization portfolio focus.

This session will describe some core metrics that help organizations, large and small, to understand their capacity and flow. These metrics tell a story about what is happening in the organization. Seeing, interpreting and sharing these stories enables organization learning, continuous improvement and ultimately helps us increase throughput and flow across the value network.

We will look at lessons from these stories, including impediments to flow, and how you can use a Kanban system to identify and manage them. Through this, we will look at ways to understand who and what in the organization is really influencing flow.

You will come away with a core set of metrics you can use to understand flow in your organization, a framework for identifying impediments to flow, and the means to understand and interpret the stories that are being told through these metrics and impediments.

Short Bio

Ken Power is co-founder of Cisco's CMBU Agile Office, where he is an internal Lean and Agile consultant/coach, working with the organization's leadership and teams around the world on their continuing agile and lean journey. He is a frequent speaker at major international Lean and Agile conferences and has authored several papers on Lean and Agile product development, including winning the Best Paper award at Agile 2011. Ken is a Fellow of the Lean Systems Society.

Agile Career Development from the trenches – lessons learned

Ionel Condor

Talk: 90 minutes

Summary

Career Development is sometimes left behind or considered an auxiliary ingredient when we talk about building great self-org teams.

No wonder why good professionals leave companies, why many of so called Senior Developers stopped learning many years ago, why sometimes Architects are only able to draw pictures, why Juniors are not finding mentors along the way, why Managers have no clue about the latest technologies or practices, why companies are having visions for their products but not for their staff and at the end what we can do to have true Professional Developers in our companies.

As a Development Manager I am going to list some of the lessons I have learned in the last 6 years of doing Agile Career Development:

- How a company traditionally see career development and the questions we had trying to adopt more Agile principles
- Developing & supporting competence:
  it all starts with hiring, continues with optimizing the whole department, self-

development, game storming, coaching, growing standards, techniques to learn more deliberately, collaborative learning, managing focus

- Dilemmas and false friends along the way: specialist/generalist, architect/manager, Peter's principle, career aspirations vs skills forecasting, career path anti-patterns
- A never ending journey and how we implemented the Dreyfus model
- Career Development metrics: metrics we found useful, patterns, anomalies, outliers and why "inspect and adapt" rocks again!
- Recommendations

This presentation is full of recipes that have started to work and are subject to continuous improvement in the environments where I work.

Short Bio

Ionel Condor (@ionel_condor) is a passionate software professional working in this industry since 1998. He likes to apply & promote healthy engineering practices and project management that make sense, clean code, scalable solutions, all driven by self-org teams that continuously improve their abilities and skills. He started as a Java developer and very soon in his career he has been given the chance to act as a lead. He worked with many teams spread across multiple countries and from the early days of his career he has been involved in all the aspects of building and growing teams. He truly believes that a technical manager needs to keep in touch with both technology and management trends, he likes to read and review books and also to blog and tweet about growing software teams. Ionel's favourite activity is to act as an Engineering Manager (7 years of experience and counting), currently for SDL Language Technologies, based in Cluj, Romania. Ionel is married and has a 2 years daughter. He likes Italian music, medicine and evangelical theology.

Get more information about this session on IdeaScale.

Rapid Software Architecture Exploration

Michael Keeling

Workshop: 90 minutes

Summary

For agile teams who desire effective yet lean software engineering practices, Rapid Software Architecture Exploration is a set of structured, lightweight architecture design, discovery, and description practices designed to efficiently explore the design space and articulate a working solution.

Unlike prevailing architecture-centric practices such as the Quality Attributes Workshop (QAW), Rapid Software Architecture Exploration practices allow your team to maximize design discovery speed, reduce costs, and increase agility without sacrificing engineering effectiveness. This is accomplished by focusing on essential architectural information, encouraging team and customer collaboration, and emphasizing systemic thinking throughout the team.

In this session I will summarize the theory behind enabling rapid software architecture exploration and describe several methods regularly used by my team at IBM as a part of our rapid exploration practice. These methods include, lean versions of common architecture-centric workshops (such as QAW), methods adapted from other disciplines such as Stakeholder Maps from UX, novel workshops such as the Prioritized Properties Web, and lightweight representations such as the Architecture Haiku.

By the end of this session you will have learned several architecture-centric practices that can be applied directly with your team back home.

Short Bio

Michael Keeling is a professional software engineer with over eight years of development experience. As an applications engineer at IBM, he helps optimize organizations’ access to information by designing and configuring enterprise search solutions. Prior to this, he filled a few software development roles including a systems analyst for Black Knight Technology for five years and has experience with analysis tools, real-time systems, and web applications. He has a BS in computer science from the College of William and Mary and a Master's in Software Engineering from Carnegie Mellon University. Michaels' interests include the pragmatic application of software engineering methods, software architecture and design, and the human aspects of software engineering.

Get more information about this session on IdeaScale.

Tuesday afternoon (15:30 – 16:30)

Fishbowl Discussion: Agile organizations generating higher profit, innovation and growth

Summary

AGILITY GENERATES 30% HIGHER PROFITS

"An overwhelming majority of executives (88%) cite organisational agility as key to global success. Other studies support this idea as well: research conducted at MIT suggests that agile firms grow revenue 37% faster and generate 30% higher profits than non-agile." — Report by The Economist magazine.

We will wrap up our day dedicated to issues facing executives and managers with a "fishbowl session" involving all track speakers, organizers, and participants in an open, yet structured discussion to gain further insights for all of us.

Adopting and enabling Agile in the enterprise has significant impacts that go beyond the Agile team. In this collaborative group discussion, we seek to provide insights into the results of agile transformation. We aim at identifying some of the most significant outcomes of organizational design, policies, practices that are relevant to enable higher profit, innovation and growth.

We invite executives and senior managers from all industries and areas to join us in discussing the organizational gains that you have experienced in adopting Agile and, more importantly, how you have realized them.

The camel and the needle eye – or the challenges of integrating agile approaches into a standard system development process

Dr. Peter Faßbinder

Experience Report: 60 minutes

Summary

Although large companies like the Siemens AG have well-established processes for developing their products, systems and solutions, they want to benefit from the advantages of agile development as much as possible.

However, for an organization with globally spread development sites, a great variety of project types, and numerous standards and regulations to be satisfied, introducing agile practices into selected projects is only the first step of the journey. The crucial mountain that has to be climbed consists of integrating the agile approach efficiently into the standard system development process without jeopardizing any boundary conditions.

As an expert department for process consulting within the Siemens AG, we have implemented agile approaches in many Siemens business units. This experience report focuses on the three main challenges we were confronted with during that journey:

- What is the optimal process architecture for integrating agile development into the overall process landscape, and still addressing the variety of project types that often require different development approaches to go side by side?
- What are suitable approaches for integrating system requirements and system testing into the agile way of handling requirements and testing?

- How can agile roles be integrated into the standard system development process with its many traditional roles?

We will discuss the most important aspects of these challenges and demonstrate possible solutions and implementation examples.

Short Bio

Peter Faßbinder is a principal consultant for advanced process optimization at Siemens Corporate Technology. He leads process improvement projects and fosters process innovation within the Siemens AG. His current focus topics are agile system processes, integrated process solutions, and the management of global processes. Prior to joining Siemens, Peter worked as project leader and architect at T-Systems, IBM Global Services and Telecom New Zealand.

We don’t think the way we think

Manuel Kiessling

Talk: 60 minutes

Summary

When thinking about ‘How much is 2 + 2?’, you are using a completely different think system than when thinking about ‘How much is 23 x 59?’.

And, when calculating how many animals of each species Moses took with him in the ark, you might come up with a very quick answer of ’2′, but also, you might have been fooled by one of your think systems, because it was Noah who built the ark, not Moses.

As we will learn, there are a lot more – and often much more profound – failures caused by the properties of and the interplay between both systems.

Over the last years, these have been at the centre of several important research projects in cognitive psychology, and their findings tell us a lot about how and why humans actually arrive at the decisions they make.

Knowing the psychological basis for reactions, judgments, recognition, choices, conclusions, and much more, is a key factor in succeeding when building and leading agile teams and introducing change in the overall organization – this talk will equip you with the knowledge and mental tools you need.

Short Bio

Manuel Kiessling is a software architect, test driven software developer, and agilist who helps introducing Scrum and XP into organizations. He currently works as an IT team lead in Cologne, Germany.

Get more information about this session on IdeaScale.

The Impact of Agile on Technical Debt

Panel

The term “Technical Debt” was coined over 20 years ago by Ward Cunningham in a 1992 OOPSLA experience report to describe the trade-offs between delivering the most appropriate – albeit likely immature – product, in the shortest time possible. Since then the repercussions of going into “technical debt” have become more visible, yet not necessarily more broadly understood. This panel will bring together practitioners to discuss and debate the impact of agile on technical debt – and more importantly, on debt relief!

The Scrum Principle – managing beyond Scrum

Boris Gloger

Talk: 60 minutes

Summary

What is managers role in an agile company? What is the function of me as a manager when my teams will work with Scrum and Kanban – when they are more self-motivated, more self-directed?

We need managers and they are key to the survival also of an agile organisation so we want to give you the missing answer to the above questions.

In this talk three executive manager: from the customer, from the software development company and from the Scrum consultant company, will explain in a interview style approach to what the role of managers will change.

1. The manager as leader beyond people management -> Orientation, Fokus and Collaboration
2. The manager as domain specialist -> Education, giving technical direction not by demand but by training its people, and work with them.
3. The manager as business expert -> Helping Product Owner to do their jobs much better, instead of forcing them to do what managing want.
4. The manager as specialist -> The manager was a specialist – he will only gets respect from his people, if he is a specialist in something.
5. The manager as governance creator -> How do we establish a governance framework that is transparent for everybody.

And again – not based on some nice theory but related to real live experience of three companies.

Short Bio

Boris Gloger is the European Scrum-Pioneer. He has been directly trained by Ken Schwaber and was certified as the first Scrum Trainer. Boris is constantly adapting and developing the Scrum practices for use in different cultural areas of the world and has set standards in the German-speaking and Scandinavian part of Europe as well as in Brazil and South-Africa. Besides educating trainers, teaching teams and individuals in the use of Scrum, Boris and his team of consultants put the emphasis on the organizational changes and transformations that come along with Scrum implementations. Beyond the boundaries of the method itself Boris – as a consultant, author and speaker – always considers the company-wide implications and impacts of Scrum. To address different issues on agile and Scrum he published 3 books: Scrum, Scrum for HR and Agile Contracts (available in English soon)

Get more information about this session on IdeaScale.

Scenario based testing- multiple browsers, devices, environments

Jodok Batlogg

Talk: 60 minutes

Summary

A case-study on how Lovely Systems tied together existing open source tools to provide a solution which provides the stake holders of an agile software project a live view on the current implementation status of their system. The framework was developed in the course of a software project for a German enterprise customer.

Built on top of Jenkins, Robot Framework, Python and various other open-source components, the framework makes requirements described as use case scenarios testable and monitorable. Test-Cases are automatically generated from functional descriptions. The solution allows multiple teams working on multiple distribution channels of the same system to work together and report the outcome of each development iteration on a central dashboard.

Each use-case scenario is tested automatically for every defined frontend channel. For example, the use case “user buys product” is tested on multiple browsers and on over 20 mobile devices running on Android or IOS. Due to the reason that the defined scenarios are actually the test definitions, it is ensured that requirement specification and testing stays in sync at any time. The results are displayed in a dashboard, which is focusing on simplicity. The dashboard can easily be integrated into wikis like confluence or simple html pages.

Short Bio

Jodok Batlogg is partner and cofounder of Lovely Systems. He is an agile being and shows strong technical background. He is involved in various international open source activities. Before Lovely Systems and being CTO of StudiVZ he also served in the Board of Directors of the Zope Foundation and Plone Foundation.

Get more information about this session on IdeaScale.

Tuesday afternoon/evening (16:50 – 24:00)

Open Space

Wednesday – all day

Open Space

Wednesday morning (09:30 – 10:30)

Effects of Negative Testing on TDD: An Industrial Experiment

Adnan Causevic, Rakesh Shukla, Sasikumar Punnekkat, and Daniel Sundmark

Research Paper

Summary

In our recent academic experiments, an existence of positive test bias, that is lack of negative test cases, was identified when a test driven development approach was used. At the same time, when defect detecting ability of individual test cases was calculated, it was noted that the probability of a negative test case to detect a defect was substantially higher than that of a positive test case.

The goal of this study is to investigate the existence of positive test bias in test driven development within an industrial context, and measure defect detecting ability of both positive and negative test cases. An industrial experiment was conducted at Infosys Ltd. India, whose employees voluntarily singed up to participate in the study and were randomly assigned to groups utilizing test driven development, test driven development with negative testing, and test last development. Source code and test cases created by each participant during the study were collected and analysed.

The collected data indicate a statistically significant difference between the number of positive and negative test cases created by industrial participants, confirming the existence of positive test bias. The difference in defect detecting ability of positive and negative test cases is also statistically significant. As a result, similarly to our previous academic study, 29% of all test cases were negative, contributing by revealing as much as 71% of all the defects found by all test cases. With this industrial experiment, we confirmed the existence of a positive test bias in an industrial context, as well as significantly higher defect detecting ability of negative test cases.

Presenter

Adnan Causevic is a PhD student at Mälardalen University, Sweden, member of the Software Engineering Division at Mälardalen Research and Technology Centre (MRTC). He graduated from the Sarajevo University, Faculty of Electrical Engineering, Computer Science and Informatics Department in 2006. His research interests include software verification and validation, software testing, agile testing and empirical studies in software engineering.

Mälardalen University, Sweden, adnan.causevic@mdh.se, Staff Page

Other Authors

Rakesh Shukla, Infosys Ltd. India, rakesh_shukla@infosys.com, http://www.infosys.com

Sasikumar Punnekkat, Mälardalen University, Sweden, sasikumar.punnekkat@mdh.se, Staff Page

Daniel Sundmark, Mälardalen University, Sweden, daniel.sundmark@mdh.se, Staff Page

Investigating the Impact of Experience and Solo/Pair Programming on Coding Efficiency: Results and Experiences from Coding Contests

Dietmar Winkler, Martin Kitzler, Christoph Steindl, and Stefan Biffl

Research Paper

Summary

Developing working software is a key goal of software development. Beyond software processes, following traditional or agile approaches, coding strategies, i.e., solo and pair programming, are important aspects for constructing high quality software code. In addition developer experience has a critical impact on coding efficiency and code quality. Pair programming aims at increasing coding efficiency, code quality, and supports learning of development team members. Several controlled experiments have been conducted to investigate benefits of different development strategies, learning effects, and the impact on code quality in academia and industry. Nevertheless, reported study limitations and various results in different contexts require more studies to fully understand the effects of experience and programming strategies. Coding contests can be promising approaches to (a) involve different participant groups, e.g., junior and senior programmers and professionals, and (b) can represent a well-defined foundation for planning and executing large-scale empirical studies. In this paper we present coding contests as a promising strategy for conducting empirical studies with heterogeneous groups of participants and report on a set of findings from past coding contests. Main results are (a) that the concept of coding contests is a promising way for supporting empirical research and (b) the results partly confirm previous studies that report on the benefits of pair programming and development experience.

Presenter

Dietmar Winkler is working as researcher and lecturer at the Christian Doppler Laboratory for "Software Engineering Integration for Flexible Automation Systems" (CDL-Flex) and the Institute of Software Technology and Interactive Systems at the TU Vienna. In 2003 he received an MS in computer science from TU Vienna, Austria. He worked as a guest researcher at the Czech Technical University, Department of Cybernetics in 2007 and received a PhD research scholarship at Fraunhofer IESE in Kaiserslautern, Germany, in 2008. Since 2010 he is researcher for quality management and software process improvement at the CDL-Flex at TU Vienna. Moreover, he is working as software engineering and process and quality management consultant in software and systems engineering industry. His research interests include Software Engineering, Engineering Processes, Quality Management, and Empirical Software Engineering.

Laboratory for Software Engineering Integrationfor Flexible Automation Systems, Vienna University of Technology, Institute of Software Technology and Interactive Systems, Favoritenstrasse 9-11/188, 1040 Vienna, Austria, dietmar.winkler@tuwien.ac.at, http://qse.ifs.tuwien.ac.at/~winkler and http://cdl.ifs.tuwien.ac.at

Martin Kitzler is a student in the computer science masters-program of the Vienna University of Technology. In his master thesis he writes about the using coding contests to evaluate benefits and drawbacks of pair programming. Since 2011 he is working for Catalysts GmbH as a software engineer, specialized on Java Enterprise Solutions in agile environments.

Catalysts GmbH; Huemerstraße 23, 4020 Linz, Austria, martin.kitzler@catalysts.cc

After earning his PhD at the Johannes Kepler University Linz, **Christoph Steindl** joined IBM Global Services as an IT Architect. Soon he became a core member of the (IBM internal) community Agile@IBM. In 2005 he quit his job and started working as an agile consultant. His company Catalysts fully embraces agile and lean practices. In 2007 organized the first Catalysts Coding Contest (http://contest.catalysts.cc). By now, several hundred software developers participate 3 times per year.

Founder and CEO of Catalysts, Catalysts GmbH; Huemerstraße 23, 4020 Linz, Austria, steindl@catalysts.cc

Stefan Biffl is an associate professor of software engineering at the Institute of Software Technology and Interactive Systems, TU Vienna. He received MS and PhD degrees in computer science from TU Vienna and an MS degree in social and economic sciences from the University of Vienna in 2001. He received an Erwin-Schrödinger research scholarship and spent one year as researcher at the Fraunhofer IESE, focusing on quality management and empirical software engineering. Also, in 2001 he received the Habilitation degree Venia Docendi for his work on empirical software engineering in project management and quality management. In 2006 he worked as guest researcher at Czech Technical University, Department of Cybernetics. Since 2010 Stefan Biffl is the head of the Christian Doppler research laboratory “CDL-Flex”.

Laboratory for Software Engineering Integrationfor Flexible Automation Systems, Vienna University of Technology, Institute of Software Technology and Interactive Systems, Favoritenstrasse 9-11/188, 1040 Vienna, Austria, stefan.biffl@tuwien.ac.at, http://qse.ifs.tuwien.ac.at/~biffl and http://cdl.ifs.tuwien.ac.at

BE BRAVE

Gitte Klitgaard

Talk: 60 minutes

Summary

You have to be brave in order to change the world!

This holds true professionally and personally. Being brave is not about removing fear or not being afraid – it is about doing what is necessary even when you are afraid.

In our jobs (and in life), we face uncertainty & resistance, need to handle conflicts. This holds even more true in agile, where changing is part of everyday. Being part of an agile transition while most managers are still doing traditional management styles requires bravery. Succeeding is faster when being brave – including failing fast

Standing by who you are even when your surroundings find you strange, having the courage to be different, making yourself vulnerable, speaking up when there is something important to you, sharing your fears and joys requires courage and cojones.

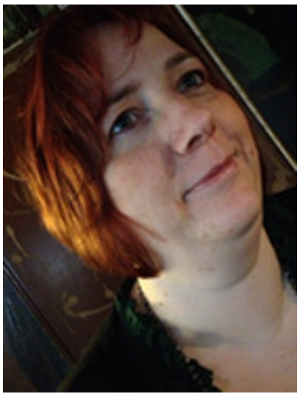
I live this personally; I have to be brave and I even have a tattoo with the title; and it is also a big learning I try to pass on to the people I coach. Being brave and vulnerable is necessary.

In my session I want to talk about why I have to, give examples of how I am courageous, and why being brave is a big part of being agile.. I will explain the meaning of brave and courageous.

I hope to have a session with lots of questions, comments and discussions.

Examples: Telling the team that it is my first time with distributed teams and that I need help with Indian culture. Asking for help. Telling stories of failures and scary stuff. Telling the truth to clients which is not necessarily what they want to her.

Short Bio



Gitte Klitgaard Hansen is an Agile Coach, Pirate, Dragon Lady, Hugger, and much much more. She is agile; live it and love it. Preferred starting point is scrum and then adapting is the key. Gitte has 10 years experience in different aspects of software development from testing, analysis, and processes to currently doing Agile Transformation. She wants to change the world by helping people in their work life; getting more work done, making the right product, doing it right and very important: have fun doing it She is a geek and very passionate about a lot of stuff 😃

Get more information about this session on IdeaScale.

Serious Games for change and engagement

Cesario Ramos

Talk: 60 minutes

Summary

If you are looking for ways to change the way you work and improve your products and customer service along the way this session is for you. We will present a case study where we used numerous Serious Games to change how people work so that better products and services are offered.

We discuss how we used Serious Games to improve management teams, engage employees, improve collaboration across departments, improve working processes and create more fun at the workplace. We describe how we worked with more then 100 people to discover breakthrough improvements using Innovation Games. We will discuss how we did it, which games we used, how we tailored the games? and what we learned.

Short Bio

Cesario Ramos works as an independent Agile Coach for various companies in The Netherlands. He is the originator of the Agile Lean Europe Bathtub conferences and a Professional Scrum Trainer from Scrum.org and Innovation Games® Qualified Instructor.

Get more information about this session on IdeaScale.

Distributed Product Owner Team for an agile medical development

Andrea Heck

Talk: 60 minutes

Summary

We are developing medical imaging and workflow software in an agile way with development teams distributed to several countries. One of the major challenges is how to set up and communicate within the Product Owner team. There we have to deal with the distribution, e.g., have the Product Owner either onsite with her peers or with her Scrum team, travelling, or with proxy. We need people who are good in two different fields of knowledge: medical and software development. As a third issues, the environment of the customers may be different in different countries.

We have ramped up local Product Owners in different countries, have found local collaboration customers, and have developed a set of communication channels and workshops how to synchronize Product Owners in the team, share a common vision and backlog with their Scrum teams, and collaborate with customers locally and globally.

Short Bio

Andrea Heck is agile coach in and beyond a Siemens Healthcare business unit. She has started and leads this agile transition since 2008. She has been working as Scrum Master and does a lot of coaching and facilitating. Co-founder of the Siemens-wide Agile @ Healthcare conference. Presented at OOP2012 and XP2012. She has been working in the software industry since she graduated in computer science in 1991, developing software, integrating, managing and coaching software development teams and organizations. She is interested in people and what makes them change their minds, in big changes as well as in continuous improvement.

Get more information about this session on IdeaScale.

How to Narrow Down What to Test

Zsolt Fabok

Workshop: 90 minutes

Summary

Nowadays testing legacy code, especially writing automatic test cases costs a lot. This looks like an extra expense in the short run, but saves a lot of trouble in the long run. However, not every organisation can afford to spend expensive coding time on testing that is considered having as no real value to the customer. Especially when we are talking about a large piece of untested legacy code.

The best way to do this effectively is to test those parts of the code that really need to be tested. In my presentation I'm going to share several methods that can be used to find areas that are worth testing so that organizations do not have to spend more effort on testing legacy code than what is absolutely necessary.

Short Bio

Zsolt Fabok is a team leader and a blogger. He has always been curious about software development processes and how to bring the best out of them and create better products. When he started to get familiar with the Agile, Lean and eXtreme Programming methodologies six years ago, he immediately became an enthusiast. He spent the last four years mastering Kanban and helping large and small organizations to successfully adapt its principles. He is the founder of the Budapest Lean and Kanban Meetup and the limitedwipsociety.hu groups.

Get more information about this session on IdeaScale.

Wednesday morning (11:00 – 12:30)

Visualizing and Managing Technical Debt in Agile Development: an Experience Report

Paulo Sérgio Medeiros dos Santos, Amanda Varella, Cristine Ribeiro Dantas, and Daniel Beltrão Borges

Research Paper

Summary

One of the main tenets that make agile methods effective is the right balance between the importance given to the people developing the software and the engineering practices dedicated to keep its quality. This equilibrium, however, can be hard to achieve. And, unfortunately, the tension between engineering practices and delivering value sacrifices the technical side most of the times. This paper reports the experience of an architecture team of a software development department with 25 agile teams in supporting technical decisions regarding technical practices. After trying other approaches to technically support the teams the architecture team invested in the technical debt metaphor to guide the department teams' technical decisions. The main motivation to use technical debt was its acknowledged potential in driving software development and maintenance decisions, especially those long term maintenance tradeoffs which are usually less visible to developers and decision makers in general. We propose the use of a "technical debt board" with main technical debt categories to manage and visualize the high-level debt, combined with tools to measure it at low-level (software metrics and other kind of static analysis). We have found that our approach improved the teams' awareness about the technical debts, stimulated a beneficial competition between teams towards the debt payment and enhanced the communication regarding technical decisions.

Presenters

Paulo Sérgio Medeiros dos Santos, Federal University of Rio de Janeiro, System Engineering and Computer Science Department, Cidade Universitária – Centro de Tecnologia. Rio de Janeiro, Brazil, pasemes@cos.ufrj.br

Amanda Varella, Cristine Ribeiro Dantas, Daniel Beltrão Borges, Petrobras, Exploitation and Production Business Solutions, Centro 20031-912 Rio de Janeiro, Brazil, amanda.varella@petrobras.com.br, cristine.dantas@petrobras.com.br, daniel.borges@petrobras.com.br

How are Agile Methods and Practices Deployed in Video Game Development? A Survey into Finnish Game Studios

Jussi Koutonen and Mauri Leppänen

Research Paper

Summary

Agile methods and practices are largely deployed in software engineering. Game development shares many features that have given rise to the emergence of agility in soft- ware engineering. There is, however, a lack of understanding of the extent to

which agile methods and practices are actually deployed in video game development and with which impacts. This paper reports on a survey into Finnish game studios. It shows that Scrum and, to a lesser degree, XP and Kanban are frequently used in the game studios. The most positive impacts of agility concern communication, quality of video games, and finding fun and implementable features earlier.

Presenter

Jussi Koutonen, Software designer by occupation. Gamer in heart.

Department of Computer Science and Information Systems P.O. Box 35 (Agora), FI-40014 University of Jyväskylä, Finland, jussi.koutonen@sysdrone.fi

Co-Author

Mauri Leppänen, Department of Computer Science and Information Systems P.O. Box 35 (Agora), FI-40014 University of Jyväskylä, Finland, mauri.leppanen@jyu.fi

Inter-Organizational Co-Development with Scrum: Experiences and Lessons Learned from a Distributed Corporate Development Environment

Raoul Vallon, Stefan Strobl, Mario Bernhart, and Thomas Grechenig

Research Paper

Summary

Distributed development within a single organization adds a lot of overhead to every software development process. When a second organization joins for co-development, complexity reaches the next level. This case study investigates an agile approach from a real world project involving two unaffiliated IT organizations that collaborate in a distributed development environment. Adaptations to the regular Scrum process are identified and evaluated over a six-month-long period of time. The evaluation involves a detailed problem root cause analysis and suggestions on what issues to act first. Key lessons learned include that team members of one Scrum team should not be distributed over several sites and that every site should have at least one Scrum master and one product owner.

Presenter

Raoul Vallon received a MSc in Business Informatics in 2011 and a MSc in Software Engineering & Internet Computing in 2012 from Vienna University of Technology. Currently he is working on his PhD study focussing on the investigation and improvement of agile and lean processes for large distributed industrial software development project environments. In the last five years he has analyzed, coached, and worked in several lean and agile projects. At INSO he is the coordinator for the AMMA (Amazing Makers) research group with a strong focus on processes for software engineering teams.

Research Group for Industrial Software, Vienna University of Technology, Vienna, Austria, raoul.vallon@inso.tuwien.ac.at

Further Authors

Stefan Strobl is a research assistant at the Vienna University of Technology. He is currently working in the field of software engineering with a strong focus on maintenance and evolution. A specific highlight is the adoption of agile practices under particularly difficult (and sometimes extreme) circumstances. Further interest lie in software internationalization and software product lines – all from an academic as well as industrial perspective. Furthermore Stefan is a committer on the Eclipse Babel Project.

Mario Bernhart is the coordinator for the BUSY (Building Systems) software engineering research group at INSO. His research interest covers everything useful and practical in large-scale and real-world software engineering. Currently he is working on code review systems and the re-engineering of legacy systems in aviation and other domains. In 2012 he was a visiting researcher at the MIT Complex Systems Research Lab (CSRL) of Prof. Nancy Leveson. He is an Eclipse committer and regular speaker at EclipseCons (2009-2011), and from 2009 to 2012 he was leading the Eclipse Mylyn Reviews project (successor of ReviewClipse) which now provides the Gerrit connector for Eclipse. From 2000 to 2005 he worked with Frequentis AG on Taptools and was the lead engineer for the successful smartStrips electronic flight strip (EFS) system.

Thomas Grechenig is a senior architect in large IT systems and nation-wide IT-infrastructures. He is a Professor for Industrial Software Engineering at the Vienna University of Technology. He and his teams have planned, designed, and built several large scale IT solutions, e.g. in health, railway systems or public transport applications. In science and research the focus of interest goes towards enhancing the stability and fine-tuning of the IT-mass-concept in all its critical sectors (usability, security, IT-infrastructure, performance, integration and interoperability).

Designing the best Process for your Team

Michael Keeling

Workshop: 90 minutes

Summary

The best processes are those that encourage teams to naturally do the right things at the right times. Amazing processes like this don't happen by accident; they are specifically designed to encourage desirable behavior while discouraging harmful behavior. By carefully choosing the process's affordances — practices or artifacts that direct our thinking toward a specific goal — a team can tailor a process that makes success intuitive. The session will begin by presenting the core concepts behind affordance-driven process improvement before diving into a collaborative workshop.

During the workshop teams will get a chance to put theory to practice and apply design and ideation methods to agree on team values as well as to brainstorm practices that will help them promote those values and trigger the right behaviors. Teams can directly apply the concepts and methods taught in this workshop to their team retrospectives and any other process improvement initiatives.

Short Bio

Michael Keeling is a professional software engineer with over eight years of development experience, currently working at IBM. As an applications engineer, he helps optimize organizations' access to information by designing and configuring enterprise search solutions. Prior to this, he filled a few software development roles including a systems analyst for Black Knight Technology for five years. He has a BS in computer science from the College of William and Mary and a Master's in Software Engineering from Carnegie Mellon University. Michaels interests include the pragmatic application of software engineering methods, software architecture and design, and the human aspects of software engineering.

Get more information about this session on IdeaScale.

Our Scrum is Great! (And Our XP Practice Ain’t Bad.) Now what?

Steve Holyer and **Diana Larsen**

Workshop: 90 minutes

Summary

Take Your Agile Adoption to the Next Level

Your Agile transition was a success. That was two years ago. Now all development is done in Agile teams. Your teams deliver value consistently, and most of your teams are mastering the XP technical practices. Yet, something feels stagnant. Are your teams delivering value to the market cadence? Do all your managers and key business experts feel their expertise is incorporated into the teams? Is your dream of a truly “lean startup” style organisation beginning to seem like just that–a dream?

Is your “Agile” the new “Waterfall”? Countless presentations, experience reports and workshops have been given about successfully transforming a company with one flavour of “Agile” or another. This time we turn the focus onto kickstarting the next, continuing Agile transition long after the Agile transformation mission was accomplished.

As coaches we are being called into organisations that have mastered the mechanics of Scrum, Kanban and XP. And yet they feel something is missing. We will review the literature on continued transitions and present our own research and experience. We’ll identify milestones on the long-term path towards gaining true fluency in Agile Software Development. Presenting examples drawn from many different types of teams we have met and coached, we will help you identify specific pointers for moving the language of the Agile transition into sustained continuous improvement and for kickstarting the Agile rollout after the Agile rollout.

Short Bios

As a developer, **Steve Holyer** was initially sceptical of Agile development, but embraced it enthusiastically when he realized Scrum allowed him to deliver better software in a saner way. He is a “by the book” scrum master who doesn’t want to stop at “scrum by the book”. In his ongoing journey he has worked to master the art the “Scrum master”, learned the importance of XP practices and advised lean startups in the technical and social aspects of doing agile development with remote teams. Steve coaches organisations and teams finding their path to truly transformative agile fluency.

Diana Larsen consults with leaders and teams to create work processes where innovation, inspiration, and imagination flourish. Diana brings focus to the human side of organizations, teams and projects. She activates and strengthens her clients’ proficiency in shaping an environment for productive teams and thriving in times of change. Diana co-authored Agile Retrospectives: Making Good Teams Great! Past Chair (2007-2010) and current board member of the Agile Alliance Board of Directors, she co-founded the “Agile Open Northwest” conference and the international “Retrospective Facilitators Gathering”. Diana discovers solutions and possibilities where others find only barriers and obstacles.

Get more information about this session on IdeaScale.

How Agile Localization can be a competitive business advantage

Vasco Duarte

Workshop: 90 minutes

Summary

Was the release of your project ever delayed when localization problems were found too late? Or worst, delayed subsequent products because of this delay? Or the fact that UI specifications quickly get out of date, leaving us with very poor quality testing by localization testing vendors. In most projects localization is still done in “waterfall” mode. Localization teams are typically involved at a very late stage of the development cycle.

We have lived through many of these problems, and we believe that Localization and Agile software development were born to be together. If you want to know how localization can be done better and faster come and see us.

Short Bio

Vasco Duarte currently an Agile Coach at Avira, Vasco Duarte is an experienced Product and Project Manager. Having worked in the software industry since 1997, Vasco has also been an Agile practitioner since 2004, he is one of the leaders and catalysts of Agile methods and Agile culture adoption at Avira and previously at Nokia and F-Secure. Vasco's contributions to the development of the Software industry and professions can be read at his blog: http://SoftwareDevelopmentToday.blogspot.com Or you can follow Vasco on Twitter: @duarte_vasco

Get more information about this session on IdeaScale.

Brutal Refactoring Game

Adrian Bolboaca

Workshop: 90 minutes

Summary

Did you know you can have legacy code after only 15 minutes? This is why we will be brutal with the coding smells. We will take the time to refactor soon and often.

The facilitator will be your benevolent dictator. His wish is to have the cleanest code possible. This is why any code smell must be removed immediately when spotted. This is why, while coding, the facilitator will stop you whenever he spots a coding smell. **Adding functionalities is forbidden until you refactor the smell away.**

You will need a laptop with a working and testing environment installed for your preferred language. This event will be language agnostic. Also you should have locally installed a source control software (git, bzr, mercurial etc).

Short Bio

Adrian Bolboaca was involved in developing software for domains like energy, e-commerce, banking, customs and ERP/CRM. He has been working with companies from Netherlands, Romania, Italy, France and Germany, and he is knowledgeable in software technical domains like: clean code, unit testing, test driven development, simple design, emergent design, working effectively with legacy code. As a continuous learner and challenger of existing ideas and concepts, Adrian is a supporter of movements that give new ideas on how to continuously improve software and that embrace the values of software quality and efficiency. He is fluent in Romanian, English and French, but when it comes to programming languages he supports language agnosticism, since he strongly believes that a programming language is only a tool towards higher-level software concepts. Adrian works as a technical and organizational trainer and coach at Mozaic Works.

Wednesday afternoon (13:30 – 14:30)

Make Impacts, Not Software

Gojko Adzic

Keynote

Summary

Software is everywhere today, and countless software products and projects die a slow death without ever making any impact. Today’s planning and roadmap techniques expect the world to stand still while we deliver, and set products and projects up for failure from the very start. Even when they have good strategic plans, many organisations fail to communicate them and align everyone involved in delivery. The result is a tremendous amount of time and money wasted due to wrong assumptions, lack of focus, poor communication of objectives, lack of understanding and misalignment with overall goals.

There has to be a better way to deliver! Gojko presents a possible solution, impact mapping, an innovative strategic planning method that can help you make an impact with software.

Presenter

Gojko Adzic is a strategic software delivery consultant who works with ambitious teams to improve the quality of their software products and processes. He specialises in agile and lean quality improvement, in particular agile testing, specification by example and behaviour driven development.

Gojko's book Specification by Example was awarded the #2 spot on the top 100 agile books for 2012 and won the Jolt Award for the best book of 2012. In 2011, he was voted by peers as the most influential agile testing professional, and his blog won the UK agile award for the best online publication in 2010.

Gojko is the author of Impact Mapping, Specification by Example, Bridging the Communication Gap, Test Driven .NET Development with FitNesse and The Secret Ninja Cucumber Scrolls

Over the last thirteen years, he has worked as a developer, architect, technical director and consultant on projects delivering financial and energy trading platforms, mobile positioning and e-commerce applications, online gaming and complex configuration management systems.

Wednesday afternoon (14:50 – 15:50)

Multi-project management with dispersed agile teams

Maximilian Hantsch-Köller

Experience Report: 60 minutes

Summary

How do you deliver safety-critical applications for air traffic control to multiple parallel customer projects out of a distributed workforce? Where can agile methodology help?

Last year, Frequentis introduced agile development methods to coordinate projects and product development for air traffic management safety-critical applications out of dispersed and distributed teams in two countries. This talk will present the approach used, questions that arose, decisions taken, the observed results, impact on team motivation, and lessons learned throughout the process. Witness the transformation of a waterfall-driven business into an increasingly agile organization!

Short Bio

Maximilian Hantsch-Köller graduated from Vienna University of Technology and has worked in numerous small and large-scale organizations from start-ups to international multi-cultural corporations. With a long-standing experience in software development and broad understanding of communication networks and technologies, Maximilian has been leading classic and agile teams in both highly structured and dynamically flexible set-ups. Maximilian is keen on growing teams and organizations, implementing the appropriate processes & communication, and allowing the magic of team spirit to unfold.

Self-Experimentation for Continuous Improvement

Tom Perry

Talk: 60 minutes

Summary

Have you ever struggled to convince a team that your ideas have merit? How can you tell if the change you proposed at the last retrospective will actually pay off? How can we build a mindset of experimentation into team behavior? What if there was a powerful way to demonstrate through your own actions the legitimacy (or fallacy) of your ideas?

Self-experimentation offers us a way to test out our ideas and demonstrate their validity. It offers us a way to demonstrate, through modeling the desired behavior, a mindset of experimentation and validation. Self-experimentation has a long history in disciplines like psychology, chemistry and medicine.

Tom Perry shares strategies for setting up self-experiments of your own. Starting with identifying antecedent behaviors, definition of the reinforcement and other behavior modification mechanisms, with hands on examples he teaches how we can use self-experimentation to improve the performance of our selves and our teams.

Short Bio

Tom Perry has been working in software development for over 20 years. He has worked on teams at startup companies, large corporations in the Fortune 100 and the State and Federal Government. His background includes testing, development, project/program management, agile coaching/mentoring and training. As part of his involvement in the greater agile community, he led the Seattle eastside chapter of the APLN. Tom speaks at a wide variety of software development conferences all over the world on a wide variety of Agile and Lean topics. Check out his blog for more information about his writing, presentations and current projects http://agiletools.wordpress.com or follow him on twitter @tlperry

Get more information about this session on IdeaScale.

Collaboration Tools for Agile: Learning and Practice

Panel

Tools play an important role in agile development – particularly as projects increase in scale and scope. This panel is designed to attract both academics and industry practitioners to share and discuss their challenges and experience – both in teaching elements of agile as part of a curriculum – and in practice to deliver products that delight customers. While code-oriented tools have existed since the early days of waterfall development, new collaboration methods and tools have emerged to facilitate and speed collaborative development and support (where support may be taken in the context of learning agile – or as a maintenance activity in the context of solving customer issues).

Agile Contracts – Can we make them possible?

Andrea Provaglio

Talk: 60 minutes

Summary

One of the clearest signs of a strong, mature Agile culture in a software development organization is the degree of Agility in the contractual agreements with its clients — and vice-versa.

However, there is often so much resistance that, apparently, just the mere thought of an Agile contract defeats the envisioning capacity of many organizations.

There are of course solid reasons for this, but the three questions we'll try to answer in this session are: are these reasons so solid, after all?; are there alternatives?; and are there ways to make these alternatives more appealing for all parties involved?

With these questions in mind, we'll explore the nature of our traditional contractual forms and the culture they come from, in different situations; we'll see which is our responsibility in making and signing these contracts; we'll see some real-life alternatives and ways to facilitate a transition to a more modern forms.

Finally, we'll set aside enough time to share ideas about this subject with the audience.

Short Bio

Andrea Provaglio is an independent professional in the field of Agile/Lean software development, with 20+ years of experience in three different continents. He helps his clients to develop better products and to create better teams; and he supports individuals and teams who want to improve technically and relationally. Andrea's clients are from different domains, ranging from large organizations such as the European Commission in Brussels, to small and dynamic IT companies. He currently works in Europe; Andrea also worked for four years in the U.S. on a O-1 visa for "extraordinary abilities in Science". More information at http://andreaprovaglio.com/about

Get more information about this session on IdeaScale.

Discussion and Q&A to "Impact Mapping and Story Mapping"

Gojko Adzic, Christian Hassa

Wednesday afternoon (16:00 – 17:30)

The Art and the Real Option of Doing Nothing and Trusting People

Steve Holyer and **Olaf Lewitz**

Workshop: 90 minutes

Summary

Learn to sit around doing nothing and trusting people. You will be surprised, and delighted, by what emerges.

One of the very best things you can do as an agile leader, coach or Scrum Master is to sit there doing nothing and trusting people. Doing nothing allows a team to trust itself, and trust enables you and the team to discover more emerging options.

Trust is the basis of establishing a learning culture, which is necessary for real collaboration and high performance. And, doing nothing is harder than it seems. It means being certain the team has the experience and knowledge and knows the constraints it needs.

Trusting people means really believing that they can do it… and continue to do it (learn/grow/deliver) better.

Because, experience shows, they can! When Agile teams “smell” trust, they can accomplish just about anything. But beware, developers are really good at smelling fake-trust. Merely saying “I trust you” is only the imitation scent of faux-trust. In this session we will explore real trust.

AS AN Agile Leader I WANT to trust People SO THAT they can trust themselves to find better ways of developing software.

Short Bios

Steve Holyer coaches organisations and teams finding their path to truly transformative agile fluency. He creates containers of trust helping communities and healing to emerge.

Co Speaker: **Olaf Lewitz** is an independent agile coach and linchpin. He inspires people to improve the way they work. He inspires managers to make their organisations more effective. His motto is that of Nanny McPhee: “When you need me, but do not want me, I must stay. When you want me, but no longer need me, then I have to go.” You can find him on Twitter as @OlafLewitz.

Get more information about this session on IdeaScale.

Pecha Kucha – Your Presentation Skills

Erik Lundh

Short talks: 60 minutes

Summary

Like it or not, if you are going to influence more than a few people you need to be good at presentations. Not everyone can get a spot in TED but the Pecha Kucha is another great challenge. The format was invented 10 years ago at a Tokyo design school to help design students better focus on their message.

You create 20 slides. They are put in a slide deck out of your control and advanced automatically every 20 seconds. You tell your story with that timing.

The strict format forces you to distill your story and use your 20 visual impressions much more carefully.

You might even find that it is hard to fill 20 slides once you found your true core message.

Pecha Kucha was a very popular event at XP2010 where we saw well-known agilists like Mary Poppendieck and Rachel Davies take on the challenge.
http://www.pechakucha.org/

Short Bio

Erik Lundh has more than 25 years experience in software development. Erik has worked with mature innovation firms and start-ups, from small to large organizations such as Ericsson and ABB. Erik programmed industrial just-in-time (Lean) systems in the 1980's, was a "process and management guy" in the 1990's, and spent the fun part of the 2000's as an agile evangelist and coach. In 2006 Erik was invited to Ericsson's first major agile transformation of 2300 R&D people at 10 sites in 5 countries.

Get more information about this session on IdeaScale.

Wednesday evening (19:00 – 22:00)

Conference Dinner

Thursday morning (09:15 – 10:30)

Agile development outside the SCRUM team: integrating professional cultures

Helen Sharp

Keynote

Summary

Building successful cross-functional development teams brings clear benefits. However full integration of different functions within the development team is not always practical. For example where a database specialist needs to support several development teams, where design coherence needs to be maintained across a suite of deliverables, where testers need an independent view of the release, and so on. Different approaches have been adopted, e.g. merging processes and adapting techniques, but it is a complex endeavour requiring a holistic view aimed at integrating professional cultures, and that is hard. Focusing on one area, that of User Experience (UX) design, this keynote will explore the different ways in which integration has been approached, and consider possible ways to address the challenge.

Presenter

Helen Sharp is Professor of Software Engineering in the Computing Department of The Open University, where she leads the Empirical Studies of Software Development research group.

Her main research interest focuses on the human and social aspects of software engineering, leveraging expertise in both

Interaction Design and Software Engineering. In particular, she is interested in the practice of software engineering and therefore has established and maintained a wide range of industrial collaborations. Helen is actively involved in the development and organisation of several academic and industrial conferences in software engineering, HCI and agile development, including SPA, Agile; XP; OOPSLA (now called SPLASH); and HCI. Helen was Associate Editor in Chief for IEEE Software with responsibility for human and social aspects from 2008 to 2010 and is now co-editor of the Voice of Evidence column.

Thursday – all day

Open Space

Thursday morning (11:00 – 12:30)

A Metrics Model to Measure the Impact of an Agile Transformation in Large Software Development Organizations

Jeanette Heidenberg, Max Weijola, Kirsi Mikkonen, and Ivan Porres

Research Paper

Summary

As the adoption of agile and lean methods continues to grow, measuring the effects of such a transformation can be valuable but challenging due to the many variables influencing the outcome of a software project. In this paper we present a metrics model developed for measuring the effects of an agile and lean transformation on software development organizations. The model was developed iteratively in cooperation with industry partners within the Cloud Software Finland research project. The resulting metrics model is applicable to projects of any size, complexity

and scope, using metrics that support agile and lean values. The model can be used to measure both past and on-going projects, regard- less of whether the process model used is plan driven or agile. In order to evaluate the metrics model, the proposed model has been piloted in an industry setting. We aim to further apply the metrics model in case studies with different industrial partners.

Presenter

Jeanette Heidenberg (1,2,3), Max Weijola (1,2), Kirsi Mikkonen (3), and Ivan Porres (1,2)
1 Abo Akademi University, Department of Information Technologies, Joukahaisenkatu 3-5 A, 20520 Turku, Finland, Givenname.Surname@abo.fi
2 Turku Centre for Computer Science, Joukahaisenkatu 3-5 B, 20520 Turku, Finland
3 Ericsson R&D Center Finland, Hirsalantie 11, 02420 Jorvas, Finland, Givenname.Surname@ericsson.com

Perspectives on Productivity and Delays in Large-Scale Agile Projects

Deepika Badampudi, Samuel A. Fricker, and Ana M. Moreno

Research Paper

Summary

Many large and distributed companies run agile projects in development environments that are inconsistent with the original agile ideas. Problems that result from these inconsistencies can affect the productivity of development projects and the timeliness of releases. To be effective in such contexts, the agile ideas need to be adapted. We take an inductive approach for reaching this aim by basing the design of the development process on observations of how context, practices, challenges, and impacts interact. This paper reports the results of an interview study of five agile development projects in an environment that was unfavorable for agile principles. Grounded theory was used to identify the challenges of these projects and how these challenges affected productivity and delays according to the involved

project roles. Productivity and delay-influencing factors were discovered that related to requirements creation and use, collaboration, knowledge management, and the application domain. The practitioners' explanations about the factors' impacts are, on one hand, a rich empirical source for avoiding and mitigating productivity and delay problems and, on the other hand, a good starting point for further research on flexible large-scale development.

Presenter

Samuel A. Fricker is assistant professor in the Software Engineering Research Laboratory (SERL) at Blekinge Institute of Technology (BTH). He has more than ten years experience as senior consultant, global process responsible, lecturer, and senior researcher with companies at any scale, from startups to Fortune500. Samuel's research interests are in software product management and requirements engineering.

Blekinge Institute of Technology, 371 79 Karlskrona, Sweden, samuel.fricker@bth.se

Further Authors

Deepika Badampudi, Blekinge Institute of Technology, 371 79 Karlskrona, Sweden, deba10@student.bth.se

Ana M. Moreno, Universidad Politécnica de Madrid, 28660 Boadilla del Monte, Madrid, Spain, anamaria.moreno@upm.es

Continuous Release Planning in a Large-Scale Scrum Development Organization at Ericsson

Ville Heikkilä, Maria Paasivaara, Casper Lassenius, and Christian Engblom

Research Paper

Summary

Scrum development at large-scale requires a release planning process that supports the agile way of working and planning. Most of the existing release-planning processes are plan-driven and ill suited for a large Scrum organization. This case study describes how release planning was conducted in a 350-person Scrum development organization with over 20 teams at Ericsson in 2011, and the related challenges and benefits. Data was collected with 39 interviews, which were transcribed, coded and analysed. The release planning process was continuous and characterized by regular scoping and prioritization decisions, and by incremental elaboration of features. The challenges were the overcommitment caused by external pressure, managing non-feature specific work, and balancing be- tween development efficiency and building generalist teams. The benefits were the increased flexibility and decreased development lead time, waste eliminated in the planning process, and increased developer motivation.

Presenters

Ville T. Heikkilä is a researcher and PhD student in Aalto University School of Science, Department of Computer Science and Engineering. His professional interests are focused on the advances in software development methodologies and specifically in agile and lean methods. Currently, he is studying models for scaling agile software development methods to distributed and multi-team development environments. He has a master's degree from the Helsinki University of Technology, Finland.

Department of Computer Science and Engineering, Aalto University, Helsinki, Finland, ville.t.heikkila@aalto.fi

Maria Paasivaara is a post doc researcher at Aalto University School of Science, Department of Computer Science and Engineering. Her research interests include global software development, agile software development and scaling lean and agile to globally distributed multi-team projects. She has a PhD in Computer Science from the Helsinki University of Technology, Finland.

Department of Computer Science and Engineering, Aalto University, Helsinki, Finland, maria.paasivaara@aalto.fi

Further Authors

Casper Lassenius is an associate professor at Aalto University School of Science, Department of Computer Science and Engineering. His research interests include software product development, agile methodologies, and global software development. He has a PhD in Computer Science from the Helsinki University of Technology, Finland.

Department of Computer Science and Engineering, Aalto University, Helsinki, Finland, casper.lassenius@aalto.fi

Christian Engblom is a member of the Ericsson Finland R&D management team. After joining Ericsson in 1979 he has held the roles of project manager, product manager and line manager within the R&D, as well as the market unit areas. Currently, he holds a position as driver for the Lean and Agile transformation within the R&D organization in Finland and is actively involved in Lean & Agile change wave going through the entire Ericsson R&D world. His professional interests include agile and lean software development. He has a master's degree from Tekniska Läroverket i Helsingfors, Finland.

Oy LM Ericsson Ab, Kirkkonummi, Finland, christian.engblom@ericsson.com

1000 Words – Illustrating Project Challenges with Visuals

Tarang Baxi

Workshop: 90 minutes

Summary

Project teams typically track a lot of data related to throughput, quality, scope, cost, etc. You probably already use some of this data for project dashboards and information radiators like burn-ups, burn-downs, defect distribution charts, etc. When specific challenges come up however, more creative use of the data may be called for. Simple, hand-crafted visuals can often help you effectively tell your stakeholders the story of what's going on, possible causes, and potential next steps. In this hands-on workshop, you get to exercise your visual thinking and visual communication skills. We introduce some simple visual thinking techniques like Look-See-Imagine-Show and the <6><6> Rule, and then let you apply them in a project simulation, so that you can practice hand-rolling simple visuals that speak volumes (no fancy tools needed!).

Short Bio

Tarang Baxi works with ThoughtWorks as a Project Manager and Business Analyst. He has over 11 years of experience in a variety of roles spanning analysis, project management, IT strategy consulting, business process consulting, and methodology coaching. He has extensive experience working with distributed software delivery teams. Tarang has presented and run workshops at a number of agile conferences including Agile India 2010, Agile India 2012 & 2013, XP2012 and Agile2012.

Get more information about this session on IdeaScale.

Retrospectives as a tool for change management

Jutta Eckstein and Diana Larsen

Workshop: 90 minutes

Summary

Retrospectives are known as a tool for learning and improving over time. In more traditional projects, retrospectives are scheduled at the end of the project, in agile projects at the end of each and every iteration. Many of us have experienced retrospectives as great means for detecting good, bad and missing practices and as a handle to make tacit knowledge explicit (for the good practices) and to define actions in order to deal with the bad and the missing practices.

We've used retrospectives in all these ways, yet as well as a tool for introducing change in a corporation. The change introduced might be the switch to agile, but it could be as well a change in the organizational structure or anything else. Using retrospectives for starting a change, enables the change agent to respect the existing experience, to look for patterns that should be kept despite the change because they're part of the organization's (or team's) DNA (also known as fractals) or to ensure that there is something attractive in this change for everyone (known as attractors). Typically retrospectives are used to learn from the past joint experience. In contrast, retrospectives as a tool for change management (a) isn't necessarily based on a -joint experience and (b) is meant to start something new and thus not solely focused on incremental improvement.

In this session we want to explore the different possibilities, but also the challenges and still missing links in using retrospective as a tool for change management.

Short Bio of Jutta

Jutta Eckstein is an independent coach, consultant and trainer from Braunschweig, Germany. Her know-how in agile processes is based on over fifteen years' experience in project and product development. She has helped many teams and organizations all over the world to make the transition to an agile approach. She has a unique experience in applying agile processes within medium-sized to large distributed mission-critical projects. This is also the topic of her books 'Agile Software Development in the Large' and 'Agile Software Development with Distributed Teams'. She is a member of the AgileAlliance and a member of the program committee of many different European and American conferences in the area of agile development, object-orientation and patterns. In 2011, Jutta has been elected into the Top 100 most important persons of the German IT.

Short Bio of Diana

Diana Larsen consults with leaders and teams to create work processes where innovation, inspiration, and imagination flourish. With more than fifteen years of experience working with technical professionals, Diana brings focus to the human side of organizations, teams and projects. She activates and strengthens her clients' proficiency in shaping an environment for productive teams and thriving in times of change. Diana co-authored Agile Retrospectives: Making Good Teams Great! She publishes articles and write occasional blog posts at "Partnerships and Possibilities". Past Chair (2007-2010) and current board member of the Agile Alliance Board of Directors, she co-founded the "Agile Open Northwest" conference and the international "Retrospective Facilitators Gathering". Diana discovers solutions and possibilities where others find only barriers and obstacles.
Get more information about this session on IdeaScale.

Do you remember how to play?

Jakob Wolman

Workshop: 90 minutes

Summary

The days when bearded men solved problems and came up with new innovations by thinking really hard individually have passed. The best creative ideas come from teams of people working together. This type of cross-functional collaboration requires new skills among team members and leaders. How do we catch the ideas and spark the fire of creativity? By doing as we have always done: playing and having fun.

In this session I will tell you about how I put games to work in our organisation. We use collaboration games for everything from coming up with new concepts together with customers to inspecting and adapting processes. In this session I will show pictures and tell stories from gamestorming workshops. You will see real examples of how games have brought people together and helped teams define a path to reaching their goals.

I will also give you the tools you need to get started with your own play sessions. I will explain how I put together workshops using the 7P's (Purpose, People, Process, Product, Prep, Practical, Pitfalls). I will give you examples of core games that can be used in any context. Hopefully I will wake up the child in you that wants nothing more than starting to play.

Short Bio

Jakob Wolman (prev Klamra) is an agile advocate and gamestorming facilitator, wishing to inspire others and to boost team productivity. He developed his passion for agile and collaboration games while working as a Java consultant, realizing that his team could be more successful if they just changed the way they worked and interacted. Since then Jakob has been supporting and leading teams with the aim to perform small miracles. He is a highly appreciated consultant with several successful teams and a few fast failures on his resumé. Jakob was on the program committee for Øredev 2012. He currently works as a team leader and occasional coder at Jayway. Jakob is a passionate long distance runner who doesn't believe in conventional running shoes.

Get more information about this session on IdeaScale.

Coding Dojo Challenge: Refactoring

Emily Bache

Workshop: 90 minutes

Summary

In this hands-on session we will be working on a rather smelly piece of code which helpfully has a fairly comprehensive suite of automated tests. Refactoring is one of the key skills of Test-Driven Development, and this is your chance to really practice it. The idea is not to rewrite the code from scratch, but rather, by taking small refactoring steps, gradually transform the code into a paragon of readability and elegance.

We'll be stepping into the Coding Dojo together, which is a safe place designed for learning, where it doesn't matter if we make mistakes. In fact all the code will be thrown away afterwards. You should feel free to experiment, try out different refactoring approaches, and get feedback from your peers. The great thing about this Kata is that since the tests are very good and very quick to run, they will catch every little refactoring mistake you make. You should experience how programming is supposed to be – smooth, calm, and always minutes away from committable code. The last part of the session is the retrospective, when we discuss what we've learnt, and how we can apply our new skills in our daily production code.

The code kata we'll be looking at is "Tennis", and the starting code is available here: https://github.com/emilybache/Refactoring-Katas/Tennis. (Java, Python, C++ etc). If you want to practice using your familiar development tools, please download the code and set up a project in advance of the session. Otherwise you can just come and use the cyber-dojo practice environment, which I'll have set up in advance.

Short Bio

Emily Bache has been a software developer and test engineer for nearly 15 years, working in organizations as diverse as small startup and large corporation. These days Emily is an independent consultant specializing in automated testing and agile methods. Emily has recently published a book, “The Coding Dojo Handbook”, based on her work teaching and learning skills like Test Driven Development. Emily has run Coding Dojos at conferences, in companies, with user groups, and has facilitated several Code Retreat events. Emily regularly speaks at international conferences such as XP, Agile Testing Days, ACCU, and currently acts as program chair for Scandinavian Developer Conference.
Get more information about this session on IdeaScale.

Thursday afternoon (13:30 – 15:00)

Micro Patterns in Agile Software

Giulio Concas, Giuseppe Destefanis, Michele Marchesi, Marco Ortu, and Roberto Tonelli

Research Paper

Summary

In this paper we present a study on micro patterns in different releases of two software systems developed with Object Oriented technologies and Agile process. Micro patterns can be a useful metrics in order to measure the quality of software by showing that certain categories of micro patterns are more fault prone than others, and that the classes that do not correspond to any category of micro patterns are more likely to be faulty. In our study we present some empirical results on two case studies of systems developed with Agile methodologies, and compare them to previous results obtained for non Agile systems. In particular we have verified that the distribution of micro patterns in a software system developed using Agile

methodologies does not differ from the distribution studied in other systems, and that the micro patterns fault-proneness is about the same. We also analyzed how the distribution of micro patterns changes in different releases of the same software system. We demonstrate that there is a relationship between the number of faults and the classes that do not match with any micro patterns. We found that these classes are more likely to be fault-prone than the others even in software developed with Agile methodologies.

Presenter

Giulio Concas, Giuseppe Destefanis, Michele Marchesi, Marco Ortu, and Roberto Tonelli, Department of Electrical and Electronic Engineering (DIEE) University of Cagliari, Cagliari, Italy
{concas,giuseppe.destefanis,michele,marco.ortu,roberto.tonelli}@diee.unica.it

Feature Usage Diagram for Feature Reduction

Sarunas Marciuska, Cigdem Gencel, Xiaofeng Wang, and Pekka Abrahamsson

Research Paper

Summary

Feature creep, if not managed well, cause software bloat. This in turn makes software applications become slower. Currently, soft- ware industry urgently requires mechanisms and approaches to reduce unnecessary or low value features. In this paper, we introduce a modelling notation, so called Feature Usage Diagram, and an approach to identify and visualize the required information for decision makers when reducing features. We conducted a case study using a real web application to validate and evaluate the Feature Usage Diagram elements and notation. The results showed that the Feature Usage Diagram is easy to learn and understand. Moreover, by visualising useful information, it has potential to support developers when making decisions for feature reduction.

Presenters

Sarunas Marciuska is a PhD student at Free University of Bolzano, Italy. His research focus is to identify and to visualize software features and their value. The newest advancements of his work can be found on the following website: www.featurereduction.org. In addition, he is a co-founder of a social movie recommender system – nextrailer.net.

Free University of Bolzano-Bozen, Marciuska@inf.unibz.it

Xiaofeng Wang is a lecturer at the Free University of Bozen. Her research areas include software development process, methods, agile software development, and complex adaptive systems theory. Her doctoral study at the University of Bath investigated the application of complex adaptive systems theory in the research of agile software development. She publishes in major IS journals and conferences, including Information Systems Research (ISR), Journal of Information Technology (JIT), IEEE Software, the International Conference on Information Systems (ICIS) and the European Conference on Information Systems (ECIS).

Free University of Bolzano-Bozen, Xiaofeng.Wang@unibz.it

Further Authors

Cigdem Gencel is a senior researcher at the Faculty of Computer Science of the Free University of Bolzano, Italy. Her research focus is providing novel solutions to real industrial challenges, particularly in the areas of software size and effort estimating, software measurement, software project management, requirements elicitation methods and process improvement. She is a member of the COSMIC Measurement Practices Committee.

Free University of Bolzano-Bozen, Cigdem.Gencel@unibz.it

Pekka Abrahamsson, Free University of Bolzano-Bozen,
Pekka.Abrahamsson@unibz.it

The Effect of Complexity and Value on Architecture Planning in Agile Software Development

Michael Waterman, James Noble, and George Allan

Research Paper

Summary

A key feature of agile software development is its prioritisation of responding to changing requirements over planning ahead. If an agile development team spends too much time planning and designing architecture then responding to change will be extremely costly, while not doing enough architectural design puts the project at risk of failure. Striking the balance depends heavily on the context of the system be- ing built, the environment and the development teams. This Grounded Theory research into how much architecture agile teams design up-front has identified system complexity as an important factor in determining how much planning a team does up-front, while system size, although related to complexity, has a much less direct impact. Furthermore, when determining how much design to do up-front, value to the customer can be a more important factor than overall development cost. Understanding these factors can help agile teams to determine how much up-front planning is appropriate for the systems they develop.

Presenter

Michael Waterman is a software engineering PhD student at Victoria University of Wellington, New Zealand. His research topic is the investigation of how the software industry deals with the apparent paradox of up-front architecture planning and agile software development.

Victoria University of Wellington, New Zealand,
Michael.Waterman@ecs.vuw.ac.nz

James Noble, and George Allan, Victoria University of Wellington, New Zealand, kjx@ecs.vuw.ac.nz, George.Allan@ecs.vuw.ac.nz

Where is Extreme Programming (XP) Today?

Panel

Extreme Programming, introduced by Kent Beck in his book “Extreme Programming Explained: Embrace Change” (October, 1999), was intended as an experiment-removing factors such as: geographic location; customer multiplicity; incompetent developers; large teams and disinterested customers. At its heart were set of values and practices which evolved and grew as evidenced by the publication of a multi-volume series of texts on XP (including a second edition of Beck’s original book). This panel will look back at the roots of XP and discuss the influences and future evolution – perhaps offering comments on questions of scale and scope – such as contexts involving: large teams or teams-of-teams; non-co-located teams; multiple customers; complex systems.

StarWars! Jedi vs. Sith – hands on Agile vs. Waterfall

Olaf Lewitz

Workshop: 90 minutes

Summary

In Agile transitions in huge organizations we’re often confronted with Software Architect and Design departments. These departments are a part of the traditional waterfall oriented companies and represent a typical silo, as well as Test- and Verification departments. They work for almost all teams and it’s their part of work to create specifications and tons of documents during a design phase. While introducing agile, these departments are often providing a lot of resistance in adopting agile as it means a lot of changes for such departments.

Unfortunately it's a popular opinion, that Software Architects are not longer needed in an Agile environment and by the way – this is a reason why they are so often refuse to adapt agile…

As this is a common misunderstanding, we experience in this session how roles shift during an agile transition and we'll prove that the knowledge of all waterfall-roles are also required in an agile environment.

In Addition, when introducing Agile to the traditional Waterfall-world, most of the time Agile Coaches have to explain in theory why developing products in an agile way has huge advantages compared to the traditional waterfall approach. Discussing the theory is very often like a never ending story of who has the best arguments.

To be able to compare both approaches customers ask for reference projects and an opportunity to see how it works in real life.

In this session participants are able to discover advantages and disadvantages of a Waterfall approach compared to an Agile approach in real time.

This Session is inspired by Martin Fowlers great article – 'who needs an architect'

Short Bio

Olaf Lewitz is an independent agile coach and linchpin. He inspires people to improve the way they work. He inspires managers to make their organisations more effective. His motto is that of Nanny McPhee: "When you need me, but do not want me, I must stay. When you want me, but no longer need me, then I have to go." You can find him on Twitter as @OlafLewitz.

Product Portfolio Game

Antti Kirjavainen

Workshop: 90 minutes

Summary

Agile development has solved a lot of problems: how to develop software effectively both in terms of productivity and shorter time-to-market, how to plan development releases in an ever-changing business environment, and how to improve worker morale and visibility.

The prevailing concern for leaders in the industry is how to steer an agile development organization: “How to manage product portfolios and roadmaps or strategies in an effective way that takes advantage of the potential of agile?” Old ways of strategic planning are too weighty and rigid, causing conflicts with agile development.

In this session I will present a learning game I designed to demonstrate the problems of traditional product (or project) portfolio planning processes. We will also explore ways to solve this problem with the game. The solutions explored involve transparent product business model and customer descriptions, business goal-centred business value estimation and impact mapping.

More info on the Product Portfolio Game: http://learninggamedev.wordpress.com/2013/02/28/product-portfolio-planning-game-playtest-3

Short Bio

Antti Kirjavainen is an agile coach at Houston Inc., Antti helps clients such as The Nordea Bank as well as Houston’s own development teams and leaders to get results working together towards common goals. He has worked with software R&D since 1998 and with Agile since 2006. Antti has a wide range of interests: product development, lean-agile software development, game design, and designing organizations to support knowledge work better. Antti is also an executive committee member of Agile Finland association.

Get more information about this session on IdeaScale.

Thinking about TDD: A Russian Doll, Full of Clockworks

Erik Lundh

Workshop: 90 minutes

Summary

Another take on Inside-Out/Outside-In thinking in test-driven development. Two simple, connected thought patterns, with examples, that helps you avoid serious but all too common mental pitfalls of different takes on test-driven development like TDD, BDD, ATDD. Too often both beginners and seasoned TDD-ers skip all the tests for the hard part of a piece of code. We find that the hard The Russian Doll pattern helps us think in clean onion layers of tests. We learn to look out for and recognize a Clockwork and switch design space to TDD its gears, not just the interface, thus saving a lot of debugging.

Short Bio

Erik Lundh has more than 25 years experience in software development. Erik has worked with mature innovation firms and start-ups, from small to large organizations such as Ericsson and ABB. Erik programmed industrial just-in-time (Lean) systems in the 1980’s, was a “process and management guy” in the 1990’s, and spent the fun part of the 2000’s as an agile evangelist and coach. In 2006 Erik was invited to Ericsson’s first major agile transformation of 2300 R&D people at 10 sites in 5 countries.

Get more information about this session on IdeaScale.

Thursday afternoon (16:00 – 17:00)

Agile Fluency: What We've Learned About How We Learn

James Shore

Keynote

Summary

When a team first adopts an Agile method, they stumble a bit. Nothing comes easily. With time and practice, the team gains fluency and Agile becomes second nature. But this fluency is just a plateau. No matter how practiced the team is, there is more to learn and another plateau waiting.

In this keynote, James Shore explores fluency and excellence in Agile. He'll talk about how he's seen teams grow in his 14 years of leading Agile teams and the stages of growth those teams go through. From creating value, to delivering, to optimizing, and beyond, James will look at the practices, techniques, and change necessary as teams stretch their understanding of Agile and achieve new levels of excellence.

Presenter

James Shore teaches, writes, and consults on Agile development processes. He led his first Agile team in 1999 and was an early adopter of Extreme Programming in 2000. Today, he focuses on helping people understand how all aspects of Agile, from technical, to business, to social, fit together to create successful software projects.

James is an inaugural recipient of the Agile Alliance's Gordon Pask Award for Contributions to Agile Practice and is co-author of The Art of Agile Development (O'Reilly, 2007). In 2012, InfoQ named him as one of the "most influential people in Agile." He writes about software development on his "Art of Agile" blog at http://jamesshore.com and he hosts the screencast series "Let's Code: Test-Driven JavaScript" at http://www.letscodejavascript.com.

Friday all day (09:00 – 17:00)

Fast Can be Safe: Coaching Agile Medical Software Development

Nancy Van Schooenderwoert and Brian Shoemaker

Practitioner Workshop

Summary

This workshop focuses on the intersection of Agile practices and the realities of safety-critical, regulated product development.

Are FDA/ IEC software quality and validation requirements stopping your teams from being Agile? It doesn't have to be that way.

There is no contradiction between being fully Agile and fully compliant with all the medical regulatory requirements in the USA or in Europe. This workshop will cover varying levels of safety-critical development for those with an intermediate or expert level of Agile practices knowledge. Our focus is on medical product development but the concepts behind those regulations apply to other safety-critical industries as well.

Who Should Attend

- Agile Coaches
- Managers
- Team leaders and technical leads

- Software Developers
- Requirements analysts and business analysts
- Validation and Quality Assurance personnel

What you will learn

- Perceived problems of Agile in a regulatory environment
- How coaching Agile medical work differs from Agile in other industries
- Overview of regulations (European and/or US, depending on attendees' interests)
- How Agile development improves speed and quality without losing regulatory compliance
- How to stay Agile and still use standards, SOPs, and risk management effectively

Organizers

Nancy Van Schooenderwoert was among the first to apply Agile methods to embedded systems development, as an engineer, manager, and consultant. Beginning in 1998 she has led Agile change initiatives beyond software development in safety-critical industries such as medical devices, flight simulation, factory automation, and has coached clients in the art of Agile technical and management leadership. A contributor to Agile Times and Cutter IT Journal, she also served on IEEE 1648 Agile Customers committee, and as a track co-chair for Agile 2013. She speaks at numerous conferences worldwide, and is past president of Greater Boston's premier Agile user group, Agile New England. Nancy is founder and Principal Consultant at Lean-Agile Partners, Inc.

Brian Shoemaker consults for healthcare products companies in computer system validation, software quality assurance, and electronic records and signatures. He has conducted validation both on product software and on internal software, developed software quality systems, audited software quality processes (including agile methodology), and evaluated 21 CFR Part 11 compliance.

He has had clients in clinical diagnostics, medical device engineering, medical imaging, medical-device fabrics manufacturing, contract lyophilization, clinical trial software, dental prosthetics, and bone-repair implants. He has worked with companies in Germany and Switzerland as well as the U.S. Previous to founding ShoeBar Associates, Brian had quality roles at PPD Informatics, Doxis, Inc., and Behring Diagnostics, Inc. Brian earned his Ph.D. in chemistry from the University of Illinois; he has achieved the ASQ Software Quality Engineer certification.

The Taming of the Script

Emmanuel Gaillot

Tutorial

Summary

JavaScript is a living paradox; it seems to spawn the best and the worst of the web: sometimes highly interactive browser-hosted applications that marvel by their richness and fluidity, other times monsters behaving randomly depending on which browser is used, or memory gluttons sluggishly dragging their over- weight self. How could this be possible?

It appears JavaScript isn’t a language one learns, but a language one endures. Server-side application coders are asked to improve the user experience by adding behaviors “in the browser.” What could be wrong with this? A programmer is a programmer, in a language or in another. Problem is, JavaScript isn’t like “another language” (Pythavaby++, or one of its equivalents). To use JavaScript like “another language” is like working the wood against its grain. JavaScript is a functional programming language, which underlying paradigm differs radically from object orientation. It is a language whose strength lies in its easiness to manipulate DOM elements and to associate data sections (in HTML format) to presentation styles (in CSS format), in response to events generated by the user or the server.

It is these “Good Parts” of JavaScript and the associated way of thinking that this tutorial presents. Through exercises of increasing difficulty, attendees will learn how to get a better understanding of JavaScript and its benefits, how to apply agile engineering practices (like TDD, loose coupling and emergent design) in this somehow peculiar context, and how to further their learning on their own.

Intended audience

This tutorial is meant for (eXtreme) programmers who don’t use JavaScript as their primary programming language and who wants to get a firmer grasp on their use of it – especially in situations where they have to code features that will be executed in a web browser.

Benefits of attendance

Attendees will have the opportunity to learn

- how to use Jasmine to test-drive JavaScript programs
- how to apply Functional Programming in JavaScript
- how to deal with events and callbacks
- how to reduce coupling in the code and to separate the scripts from the HTML page
- how to use jQuery and CSS selectors to manipulate the DOM
- how to find reliable documentation related to JavaScript on the Internet

Presenter

Emmanuel Gaillot works as a team coach, (extreme) programmer, facilitator, trainer and systems jiggler. For the past ten years he has been helping software makers to be better at, prouder of, and happier about their work. A speaker at many conferences on Agility, Emmanuel also organizes the annual Agile Open France conference. He has co-founded (and still assiduously attends) the Coding Dojo in Paris. Emmanuel works in Paris at /ut7, a co-operative business he learns to hack with his fellow colleagues. He currently focuses on learning exotic languages, shaping self-organizing structures and setting up co-learning spaces.

Friday morning (09:00 – 12:30)

Solution Focused Coaching and Retrospectives

Dr. Ralph Miarka and **Veronika Kotrba, MC**

Tutorial

Summary

The attendees will experience a solution focused retrospective setup, different to the work by Larsen & Derby. After the sessions, the attendees could apply this different format and they understand why and how it works. We introduce the participants to the values, principles and practices from solution focused coaching (De Shazer & Kim Berg). We also introduce scales as means to provoke team transparency and to instigate a conversation about the change that the team / team members want to see happen. This approach leaves the responsibility of change in the hands of the motivated individuals.

Attendees will take back:

- knowledge about the solution focused approach (values, principles and practices)
- a short method for self-reflection as opening
- question techniques from professional coaching
- questions for asking towards the desired future goal, including positivity
- using scaling to visualize differences between now and then
- questions to enquire about examples of the desired future already in the present or past
- questions to initiate first little steps towards the desired goal
- experience report about using such retrospectives
- lasting experience by doing a retrospective themselves in the session

We will work with the participants through a retrospective on the topic “How to improve the effectiveness of their retrospectives?”, so that they experience the method themselves in order to decide whether they would like to try it out. Also, learning is enhanced by hands-on experience of the participants.

Presenters

Veronika Kotrba, MC, is a professional coach and trainer for solution focused coaching at the Solution Management Center, Vienna. She is self-employed since she gained her Master’s degree in Coaching and Solution Focused Management in 2006. She works with clients from different background, from industry and public services. Veronika founded the Austrian Solution Circle as a platform for people that apply the solution focused approach in their work.

Ralph Miarka, is a professional coach and trainer for agile teams. He gained his Master’s degree in Systemic Coaching in December 2012. Ralph is CS(MIPOIP) and provided many Scrum trainings alone and as co-trainer. Ralph worked for Siemens AG Austria where he lead project teams successfully using agile principles and practices before he turned independent in January 2010.

We live in Vienna and we are interested in the combination of the solution focused approach and the agile values and principles, as we believe and experience that this will bring the most benefit to our customers. Together we offer the training “Leading and Coaching agile Teams“.

Complex Projects aren't plannable but controllable

Jutta Eckstein

Practitioner Workshop

Summary

Science has finally approved it: Forecasting complex projects is a deception. Moreover, forecasts hinder innovations. Daniel Kahneman, Nobel Prize Winner in Economic Sciences and psychologist verified in many cases, that forecasting of complex projects is impossible. Yet still, we keep losing time trying to do exactly that. Beyond Budgeting (see e.g. http://www.bbrt.org/) came empirically to the same findings and offers a concept for controlling corporations without budgets. Additionally Beyond Budgeting provides advice for controlling even long-term complex projects. Agile methodologies generally recommend developing a long-term plan on a coarse-grained level only and coming up with detailed short-term plans iteratively. I'm working on large and complex agile projects for more than 10 years. However, learning about Kahneman's and the Beyond Budgeting folks' work helped me a great deal in better understanding how planning, estimating and budgeting relate and why the traditional approaches don't work. Of course, we all know how this works in the small, how to plan and steer a project by iterations. Yet, how will you get the budget for starting a large agile project, how do you know how your tiny iterations (across many feature teams) fit the long-term project goal? This is not about small simple projects. It is about complex projects, e.g. projects with 50-300 developers, taking 3-5 years to finish or similarly large product (line) development.

In this session I want to analyze the latest scientific research and explore possibilities of combining Beyond Budgeting and Agile principles so that even complex projects remain controllable.

Learning objectives:

- Understand why forecasting complex projects is impossible
- Become acquainted with the core ideas of Beyond Budgeting

- Develop strategies for the application of Beyond Budgeting in Agile projects

Organizer

Jutta Eckstein is an independent coach, consultant and trainer from Braunschweig, Germany. Her know-how in agile processes is based on over fifteen years' experience in project and product development. She has helped many teams and organizations all over the world to make the transition to an agile approach. She has a unique experience in applying agile processes within medium-sized to large distributed mission-critical projects. This is also the topic of her books 'Agile Software Development in the Large' and 'Agile Software Development with Distributed Teams'. She is a member of the Agile Alliance and a member of the program committee of many different European and American conferences in the area of agile development, object-orientation and patterns. In 2011, Jutta has been elected into the Top 100 most important persons of the German IT.

Designing good test cases

Emily Bache

Practitioner Workshop

Summary

Test Driven Development is a multifaceted skill, that it takes study and practice to master. In this workshop, we'll be looking at one particular aspect of it – how to design good test cases. During this half-day session, we'll work in pairs, and work on the same code kata problem three times, varying the design of the test cases. Between each coding session we'll hold a short retrospective, and swap pairs. The coding problem we'll be working on is the "Gilded Rose" Kata, which is an example of some business logic that could do with refactoring, together with a requirements specification, but no tests.

For the three coding sessions I would like to suggest the following three approaches:

1. classic JUnit style test cases.
2. BDD style test cases using Cucumber.
3. Approval testing.

With each approach, workshop participants implement test cases, then lean on them as they refactor the code. My hope is that participants will explore with me how to create tests that are both useful for refactoring, and act to document the functionality in a more precise way than the requirements specification.

Beginners to TDD get an introduction to the skill, and more experienced practitioners will have a chance to reflect on how the choice of testing approach affects the ease of refactoring, and future maintainability of the code.

To make it easier to get set up and coding quickly, we'll be using the cyber-dojo coding environment. Participants are asked to bring a laptop if possible.

Presenter

Emily Bache has been a software developer and test engineer for nearly 15 years, working in organizations as diverse as small startup and large corporation. These days Emily is an independent consultant specializing in automated testing and agile methods. Emily has recently published a book, “The Coding Dojo Handbook”, based on her work teaching and learning skills like Test Driven Development. Emily has run Coding Dojos at conferences, in companies, with user groups, and has facilitated several Code Retreat events. Emily regularly speaks at international conferences such as XP, Agile Testing Days, ACCU, and currently acts as programme chair for Scandinavian Developer Conference.

PhD Symposium

Integrating Agile and User-Centered Design

Dina Salah

Academic Workshop

Summary

Agile User Centered Design Integration (AUCDI) is the practice of combining User Centered Design (UCD) with agile development processes. None of the existing agile processes explicitly include principles and practices for understanding and eliciting testable and verifiable usability and user experience requirements. Similarly, design practices for satisfying requirements for usability and user experience or practices for evaluating systems developed via agile processes for usability and user experience are generally absent. In addition, substantial differences exist between agile and UCD approaches which pose challenges to integration attempts.

More specifically in regards to XP, it was criticized for being light on the user side of software and apparently is better used with non GUI intensive applications. Moreover, requirements engineering as an activity within XP was not explicitly defined. Furthermore, XP has no explicit process for dealing with interaction design thus the question of how to devise an initial design is largely unanswered. Although agile methods accentuate testing, and XP involves acceptance and unit testing, nevertheless, there is an absence of supportive practices for direct support of usability testing. As a result dealing with usability issues in XP is totally dependent on the expertise of the onsite customer.

The aim of this workshop is to encourage academic and industrial attendees to share their research results and first-hand experience on agile and user centered design integration.

For more information and to submit papers, please visit the workshop website at https://www.cs.york.ac.uk/xpaucdi2013/

Organizer

Dina Salah has been the program chair for the workshop "Agile and User Centered Design Integration Workshop – What Lies Beneath and What Lies Ahead (AUCDI 2013) that was hosted by the University of York's Department of Computer Science (http://www.cs.york.ac.uk/AUCDI/). She is a final year PhD student at the University of York whose PhD is focused on Agile and User Centered Design integration. She has a number of posters and papers on the topic in the international conference on software engineering, international symposium of empirical software engineering, Annual BCS HCI Conference, and Irish HCI conference. Some of her relevant work include the following:

1. Dina Salah and Richard Paige. A Framework for the Integration of Agile Software Development Processes and User Centered Design (AUCDI).6th International Symposium on Empirical Software Engineering and Measurement, Lund, Sweden, September 2012.
2. Dina Salah, Richard Paige, Paul Cairns. Towards a Framework for Agile User Centered Design Integration. HCI 2012, Birmingham, UK, September 2012.
3. Dina Salah. A Framework for the Integration of User Centered Design and Agile Software Development Processes. ICSE 2011: 1132-1133,Hawaii, USA, May 2011.
4. Dina Salah, Richard Paige and Helen Petrie. Practices in the Integration of Agile and UCD Processes – A Reality Check. Agile 2010 – UX Session, Nashville, Tennessee, USA, August 2010.
5. Dina Salah, Helen Petrie and Richard Paige. Towards a Framework for Bridging User- Centered Design and Agile Software Development Processes.3rd Irish HCI Conference 2009,Dublin, Ireland, September 2009.

Friday afternoon (13:30 – 17:00)

Agile Kaizen – Improvement Beyond Retrospectives

Ángel Medinilla

Tutorial

Summary

According to our personal experience after coaching several hundred teams over five years, a great number of retrospectives are seriously flawed. “Pluses and Deltas” have become a common way of cargo-cult Agile, and on very rare occasions the teams have the will and the power to go beyond their close environment and push the envelope of continuous improvement through the whole company.

One of the evident reasons is that the Organization is a complex system, and most of the impediments that the team is facing will have several influences from other parts of the system: not enough budget for new tools, not allowed to take enough time for learning, no understanding of the development process by sales people, unreasonable terms arbitrarily set by project managers…

Another reason is that Agile champions don’t fully understand the reach of continuous improvement, and many times they aim for an isolated “black box” team environment that leads to sub-optimization. And yet another reason is that teams, very often suffering form a too technical approach to human aspects, don’t know how to promote improvement through the organization by effectively managing a change initiative.

This tutorial will show how even the smallest impediments detected by the team have deep meanings to the intrinsic fabric of the organization, and will give several tools and frameworks to design a Kaizen plan that goes beyond retrospectives. Some of the topics that will be discussed include:

- How Kaizen goes beyond process improvement and should include dimensions as product improvement or team improvement
- How a retrospective should not be over if a plan for improvement is not crafted (who, what, when, how)
- The need of discipline, momentum and constant pushing, or why the Scrum Master and the Agile Manager are key to Continuous Improvement
- How Corporate Culture influences Continuous Improvement, and how to promote a Culture of ownership, care and responsibility that fosters Improvement and Innovation
- How Change Management frameworks and tools can help teams, Scrum Masters, Agile Managers and Evangelists to unleash the power of Kaizen in their companies

Presenter

Ángel Medinilla, Agile Coach and trainer, working for companies like Ericsson, Vodafone, Electronic Arts, EADS, Tui Travel … Agile Alliance Member, Scrum Alliance Member, Management 3.0 Licensed Trainer. Author, “Agile Management: Leadership on an Agile Environment” (Springer, http://bit.ly/AMEDBook). Working on second book, “Agile Kaizen”.

Take Your Agile Adoption to the Next Level

Steve Holyer and Diana Larsen

Workshop

Summary

Have you accomplished your Agile Transformation? And yet, is something missing? Let’s find ways to kickstart the next, continuing Agile transformation long after the Agile transformation was deemed a success.

Diana Larsen and James Shore published the article “Finding Your Path to Agile Fluency” last Summer. In this workshop Diana and Steve will work together to help participants gain an experiential understanding of the Agile fluency model. Next we will experience and develop new ideas for helping teams take their Agile adoption to another level. We will also look at restarting a team’s transformation after the initial transformation has been labeled “Mission Accomplished”.

Presenters

As a developer, **Steve Holyer** was initially sceptical of Agile development, but embraced it enthusiastically when he realized Scrum allowed him to deliver better software in a saner way. In his on-going journey he has worked to master the art the “Scrum master”, learned the importance of XP practices and advised lean startups in the technical and social aspects of doing agile development with remote teams. Steve works as an Agile Coach with agile42 coaching organisations and teams to find their path to truly transformative agile fluency.

Diana Larsen consults with leaders and teams to create work processes where innovation, inspiration, and imagination flourish. Diana brings focus to the human side of organizations, teams and projects. She activates and strengthens her clients’ proficiency in shaping an environment for productive teams and thriving in times of change. Diana co-authored Agile Retrospectives: Making Good Teams Great! Past Chair (20072010) and current board member of the Agile Alliance Board of Directors, she cofounded the “Agile Open Northwest” conference and the international “Retrospective Facilitators Gathering”. Diana discovers solutions and possibilities where others find only barriers and obstacles.

Driving Development With Tests

Emily Bache

Workshop

Summary

Test Driven Development is a multifaceted skill, that it takes study and practice to master. In this workshop, we'll be looking at one particular aspect of it – how to build a piece of code incrementally, driving development with tests. During this half-day session, we'll work in pairs, and do the same code kata problem three times, each time varying the choice of test cases and order to implement them in. Between each coding session we'll hold a short retrospective, and swap pairs.

For the three coding sessions I would like to suggest the following three approaches:

1. classic TDD approach, bottom-up, starting with a degenerate case.
2. start with a "guiding test" that outlines the goal, but is too big to make pass straight away. Again build code bottom-up with a classic TDD approach, until the guiding test can easily be made to pass.
3. again start with a "guiding test" but then work outside-in using a London school TDD approach.

My hope is that the workshop participants will explore with me how tests can influence design choices, and whether the style of TDD makes a difference. Beginners to TDD get an introduction to the skill, and more experienced practitioners will have a chance to reflect on how they select test cases, and build a design gradually.

To make it easier to get set up and coding quickly, we'll be using the cyber-dojo coding environment. Participants are asked to bring a laptop if possible.

Presenter

Emily Bache has been a software developer and test engineer for nearly 15 years, working in organizations as diverse as small startup and large corporation. These days Emily is an independent consultant specializing in automated testing and agile methods. Emily has recently published a book, “The Coding Dojo Handbook”, based on her work teaching and learning skills like Test Driven Development. Emily has run Coding Dojos at conferences, in companies, with user groups, and has facilitated several Code Retreat events. Emily regularly speaks at international conferences such as XP, Agile Testing Days, ACCU, and currently acts as programme chair for Scandinavian Developer Conference.

Research Challenges in Large-Scale Agile Development

Torgeir Dingsøyr and Nils Brede Moe

Academic Workshop

Summary

Agile software development methods were made for small, co-located development teams, but are increasingly applied in other settings. Several large projects, with a number of teams that develop complex systems have started to use agile methods. How to apply agile methods to large projects was identified as the “top burning research question” by practitioners at XP2010. This workshop seeks to identify the main research challenges in conducting agile software development in large-scale. Such challenges include team coordination, knowledge sharing an facilitating self management. The workshop will also focus on appropriate research designs for such studies.

See complete call: http://listserv.acm.org/scripts/wa-acmlpx.exe?A2=ind1303&L=seworld&P=R824&1=seworld&9=A&J=on&d=No+Match%3BMatch%3BMatches&z=4

Submissions to: https://www.easychair.org/conferences/?conf=largescaleagile2013

Organizers

Dr. Torgeir Dingsøyr works with software process improvement and knowledge management projects as a senior scientist at SINTEF Information and Communication Technology. He has published more than 60 refereed papers in this field, including articles in IEEE Transactions on Software Engineering, IEEE Software, Communications of the ACM, Information and Software Technology and Empirical Software Engineering. In particular, he has focused on agile software development through a number of case studies, co-authoring of a systematic review of empirical studies, co-editing of the book “Agile Software Development: Current Research and Future Directions” and co-edited the special issue on Agile Methods of the Journal of Systems and Software. He wrote his doctoral thesis on “Knowledge Management in Medium-Sized Software Consulting Companies” at the Department of Computer and Information Science, Norwegian University of Science and Technology, where he is now Adjunct Associate Professor.

Dr. Nils Brede Moe works with software process improvement, agile software development and global software development as a senior scientist at SINTEF Information and Communication Technology. His research interests are related to organizational, socio-technical and global/distributed aspects. His main publications in the field of agile software development include several longitudinal studies on self-management and teamwork and co-editing of the books “Agile Software Development: Current Research and Future Directions” and “Agility Across Time and Space: Implementing Agile Methods in Global Software Projects.” Moe was also a co-organizing chair of the 11th International Conference on Agile Software Development (XP2010) and co-producer of the research at work stage at Agile2011 and Agile India 2012. He wrote his thesis for the degree of doctor philosophiae on "From Improving Processes to Improving Practice – Software Process Improvement in Transition from Plan-driven to Change-driven Development."