

## CAST 2018

**A** ACTIVITY   **B** BREAK   **K** KEYNOTE   **L** LIGHTING TALKS   **S** SESSION   **T** TUTORIAL**W** WORKSHOP

---

**AUGUST 6 • MONDAY**

---

07:30 – 08:45	<b>B Breakfast</b>	Horizons
09:00 – 17:00	<b>T Coaching Testing</b> <i>Speakers: Anne-Marie Charrett</i> The coaching that I do focuses on improving skill through questioning and practice to develop a deep understanding of testing and how to perform it.  A coach can learn how to help both testers and developers to: <ul style="list-style-type: none"><li>• Sharpen reasoning and critical thinking</li><li>• Explain your testing and why you tested</li><li>• Understand and deal with ambiguity</li><li>• Deepen your understanding of the testing you perform</li></ul> The coaching model that I use is being developed by myself and James Bach. It uses Socratic questioning to probe testing knowledge, challenging developers and testers alike to think deeper and through practice come to a greater understanding of what testing is as well as how to test in a better way. The intent is for both student and coach to leave coaching feeling enthusiastic about testing, with the motivation to continue self-learning.  The tutorial will examine the coaching model. We will look at the following: <ul style="list-style-type: none"><li>• Socratic Questioning</li><li>• Coaching Task</li><li>• Managing a coaching session</li><li>• Evaluating Coaching</li></ul> Attendees will have the opportunity to observe, analyse, practice and steer coaching sessions throughout the day. This workshop is suitable for testers, developers and leads who want to learn how to coach team members in either a remote or local environment.	Sawgrass

---

09:00 – 17:00	<b>T How to Break Software: Exploring Software’s Common Risks to Increase Robustness</b> <i>Speakers: Dawn</i> Have you ever worked on a project where you felt testing was thorough and complete—all features were covered and all tests passed—yet during the first week of production use, the software had serious issues and problems? Commonly, usability, performance, and robustness issues are among those key areas that are not targeted by requirements and often missed in testing.  Join Dawn Haynes to explore how to more effectively inject robustness testing into your projects to uncover those issues before release. Dawn shows you how—by expanding basic tests and incorporating specific robustness attacks—you can catch many defects that are commonly written, commonly missed, and tend to show up early in production. Dawn also proposes strategies for making robustness risks a project-level concern so these common issues get the priority they deserve and are evaluated before release.  Please bring a laptop with an application you wish to target for testing already set up and configured. Plan to do some exploratory testing and look for opportunities to “attack” your software during the session, as well as attempt a number of attacks from James Whittaker’s book, <i>How to Break Software</i> .  Previous attendees of this workshop have found many valuable bugs to report to their team during the session, and gained valuable insights to raise robustness awareness within their projects. What will you catch at CAST 2018, beachside in Florida?	Dunes 1
09:00 – 17:00	<b>T Testing Fundamentals for Experienced Testers</b> <i>Speakers: Robert Sabourin</i> Testing fundamentals can help experienced testers expose more bugs with less effort.  Through years of experience you have mastered testing in your domain. But are important bugs still slipping by? Can you transfer your skills to new applications? Why can’t others get the job done? Test fundamentals can help.  Rob Sabourin breaks testing fundamentals into five areas, philosophy, scientific method, problem solving, math and rhetoric. Test philosophy improves purposeful testing revealing truths about what testing can and cannot do. Scientific method provides frameworks to advance knowledge confirming or refute conjectures while designing great test experiments. Many problem solving strategies exist based on modeling knowledge and the unknown. Math (discrete, logic, combinations and probability) improves test design and result interpretation. Rhetoric skills improve tester’s communication, argumentation and persuasion.  Applying testing fundamentals focuses testing, closes gaps, eliminates waste and helps you do the right things well. Rob teaches you “how to know about what to test” and “what to know about how to test”.	Dunes 2
10:30 – 10:45	<b>B Break</b>	Seagrapes
12:00 – 13:30	<b>B Lunch</b>	Horizons
14:45 – 15:00	<b>B Break</b>	Seagrapes
<b>AUGUST 7 • TUESDAY</b>		
07:30 – 08:45	<b>B Breakfast</b>	Horizons

09:00 – 17:00

**T Advanced Automation for Agile: UI, Web Services, and BDD**

Dunes 1/2

*Speakers: Angie Jones*

As testing shifts left in an agile world, teams rely on the fast feedback of automated scenarios for continuous integration/deployment. Automation frameworks must be designed to be stable, robust, and flexible. The traditional way of automating UI scenarios in a silo doesn't lend itself to agile practices.

In this hands-on workshop, you will build an advanced automation framework capable of keeping up with the demands of agile development. This single framework will be capable of supporting the automation of UI and web services, as well as Behavior-Driven Development (BDD) and Test-Driven Development (TDD) initiatives.

You will learn how to use:

- Advanced concepts in designing your UI automation such as modeling data within your application and componentizing page objects
- TDD with the context of automation development
- BDD specs for test automation
- Cucumber to write steps that execute BDD specs
- Rest-Assured to employ web services to make your tests quicker and less brittle

09:00 – 17:00

**T Automation in Testing - Check Design**

Seahorse

*Speakers: Richard Bradshaw*

Automation in Testing (AiT) is a new namespace designed by Richard Bradshaw and Mark Winteringham. The use of automation within testing is changing, and in our opinion, existing terminology such as "Test Automation" is tarnished and no longer fit for purpose. So instead of having lengthy discussions about what Test Automation is, we've created our own namespace which provides a holistic experienced view on how you can and should be utilising automation in your testing.

Automated Check Design is the process of designing which checks should be automated, and then how to go about designing them before we implement them. Automated Checks is our preferred terminology within AiT, but if it leaves you confused, change check to test, and re-read it, we'll be talking about the same thing.

What You Will Learn On This Course:

The first half of about what and why. We're going to explore which automated checks should be created, and in turn, which shouldn't and importantly with both, why?

The second half of the day will be spent on how we go about implementing checks, from a design angle, and looking at all the factors that make up an automated check. By the end of the tutorial, attendees will be able to:

- Dissect existing automated checks to determine their purpose and intentions
- Show the value of automated checking
- Describe the anatomy of an automated check
- Discover opportunities to design automation to assist testing
- Distinguish between a check that provides value and one that doesn't
- Appreciate that succeeding with automation requires far more than coding skills

What You Will Need To Bring:

Please bring a laptop, OS X, Linux or Windows with all the prerequisites installed that will be sent to you.

09:00 – 17:00

**T The Whole Team Approach to Testing in Continuous Delivery**

Sawgrass

*Speakers: Lisa Crispin, Ashley Hunsberger*

Is your team puzzling over how to feel confident releasing to production frequently with continuous delivery? Delivering reliable and valuable software frequently, at a sustainable pace (to paraphrase Elisabeth Hendrickson), is a worthy goal. DevOps is a hot buzzword, but many teams struggle with how testing fits in, keeps up, and contributes to the DevOps culture.

In this hands-on workshop, participants will have a chance to practice techniques that can help teams feel confident releasing more frequently. You'll learn how your team can use a test suite canvas to discuss what questions each step in your delivery pipeline needs to answer, to understand the value each step provides. You'll work in groups to come up with new experiments to help shorten feedback cycles, make sure all essential types of testing are done continually, and fit testing into the continuous world. You'll learn that there IS a "test" in "DevOps".

Whether your tests take minutes or days, and whether your deploys happen hourly or quarterly, you'll discover benefits. You'll participate in a simulation to visualize your team's current path to production and uncover risks to both your product and your deployment process. No laptops required, just bring your curiosity.

Learning outcomes:

- Continuous delivery concepts at a high level, and the differences between continuous integration and continuous delivery
- Common terminology and a generic question list to engage with pipelines as a practice within your team
- How to use the test suite canvas to design a pipeline that gives your team confidence to release frequently
- Experience in analyzing pipelines from different perspectives to create a layered diagram of feedback loops, risks mitigated, and questions answered
- Ways your team can design experiments to address the many challenges of testing in a continuous world

09:00 – 17:00

**T Traditional Setup vs. Pairing vs. Mobbing – An Experiment for Testers and Developers**

Sand Dollar

*Speakers: Jan Eumann*

Having worked in software for over a decade I have come into contact with many ways of coding and testing. Together with a colleague I have given session on how pairing can improve productivity – and in many situations I believe it can indeed do that. But should testers and developers pair on things or should they be separated by the proverbial fence? What about working in large groups? Where is the data?

This is where you come in. Be part of something revolutionary while learning new techniques and ways of working.

In this workshop you will experience to work in 3 configurations: as a lone tester/developer, in pairs and in mobs. You will test, fix bugs and enhance a web application. Together, we will compare your experiences and will try to draw conclusions about which configuration works best in which situation.

Key takeaways:

- Learn the basics of pairing
- Learn the basics of mob programming/mob testing
- Learn how to use Chrome Dev Tools to inspect the code and make code changes
- How can we find bugs earlier than in the traditional/evil testing phase?
- How can we collaborate in order to bridge the gap between developers and testers?
- Learn which way of working might be suitable for what kind of task

Attendee Requirements:

Attendees should bring a laptop with the Chrome browser. The workshop can be attended by developers, testers and also product managers.

10:30 – 10:45

**B Break**

Seagrapes

12:00 – 13:30	B <b>Lunch</b>	Horizons
14:45 – 15:00	B <b>Break</b>	Seagrapes

**A** ACTIVITY    **B** BREAK    **K** KEYNOTE    **L** LIGHTING TALKS    **S** SESSION    **T** TUTORIAL  
**W** WORKSHOP

---

**AUGUST 8 • WEDNESDAY**


---

07:00 – 08:00	A	<b>Yoga on The Beach</b> <i>Speakers: Dawn</i> Bring your mat out to the beach, and start your day centered	Beach
07:30 – 08:45	A	<b>Lean Coffee</b> <i>Speakers: Matt Heusser</i> Lean Coffee is a structured, but agenda-less meeting. Participants gather, build an agenda, and begin talking. Conversations are directed and productive because the agenda for the meeting was democratically generated.  The format for a Lean Coffee is very simple. This is intentional. It is meant to be the least structure necessary for a coherent and productive meeting. No more, no less.  <i>From <a href="http://leancoffee.org/">http://leancoffee.org/</a></i>	Horizons
07:30 – 08:45	B	<b>Breakfast</b>	Horizons
09:00 – 09:15	A	<b>CAST Welcome</b> <i>Speakers: Eric Proegler</i> Eric Proegler is a Director of Test Engineering for Medidata Solutions in San Francisco, California.  Eric is the Vice President and Treasurer for the Association for Software Testing. He is also the lead organizer for WOPR, the Workshop on Performance and Reliability. He's presented at CAST, Agile2015, Jenkins World, STARWEST, Oredev, CodeFest, CMG Impact, STPCon, PNSQC, and multiple WOPRs.  In his free time, Eric spends time with family, leads a science fiction book club, sees a lot of stand-up comedy and live music, seeks out street food from all over, collects beer growlers, plays video games, and follows professional basketball.	Sea Oats
09:15 – 10:30	K	<b>Cynefin for Testers</b> <i>Speakers: Liz Keogh</i> Whenever we do anything new, we make discoveries, and often those discoveries force us to change direction and rethink our goals. In a world which embraces uncertainty, and in which innovation means trying things out and iterating more often than analyzing and predicting, what's the role of a tester?  In a world of change, where a quick reaction to problems is often a better approach than a prediction of them, we look at how a tester's mindset and skills can still bring much-needed clarity, ensuring coherence in the experiments we perform and making sure that they're safe-to-fail.	Sea Oats
10:30 – 10:45	B	<b>Break</b>	Seagrapes

10:45 – 12:15

**W An Exploration of Observing - Creating system awareness in our quest for quality**

Dunes 1/2

*Speakers: Louise Perold*

As testers we continually hone our observation of software but in our quest to bridge the gap, are we practicing observing the systems we work in enough? Are we aware of the systemic effects on quality? In this experiential workshop, you will practice awareness, interactions and observations.

*"Men are not moved by things, but by the views which they take of them" - Epictetus.*

The way I think about Bridging between Communities as a tester is that we are the "catalyst for communication" or "CFC". In order to do that, we need to be able to understand, through observing and acting - who is it that we need to communicate to, about what, in what way. How can we create more awareness of the things that are important for quality?

Over the years, I have learned from some incredible observers. I'd like to share some of these techniques and let us have an opportunity to explore and practice them.

At the core is understanding ourselves and what we bring to the observation. Then there are the interactions that we have one on one, and finally what happens as we work within a team and within a bigger system. This workshop will give participants an opportunity to both observe and be observed, and we will touch on the work of Virginia Satir, Jerry Weinberg, Marshall Rosenberg, to name a few.

Takeaways:

New awareness of various levels and layers of observation as well as some techniques to try out.

---

10:45 – 12:15

**W An Introduction to Domain Testing: More than Quick Tests**

Sea Oats

*Speakers: Dwayne Green, Chris Kenst*

We will be conducting several exercises with debriefing sessions afterwards and small presentations focused on filling in the relevant context.

Students can expect to:

- Test a sample application
- Identify, classify and discuss data types used by variables in the application
- Determine if said variables are good candidates for either boundary analysis + equivalence class partitioning
- Generate best representative data and put that information into a classic domain testing table
- Work with other students to generate and catalog data for later use

Learning Objectives:

- What Domain Testing is
- What it is not
- How to apply it in a systematic way

Testing is a complex activity and developing competence requires mastery over many tasks and practice, practice, practice. Yet there are very few places where we get a chance to practice applying test techniques and get feedback on how we are doing. This is that place. When used appropriately, Domain Testing (an umbrella term for Boundary Analysis and Equivalence class partitioning) can increase our efficiency by helping us run less redundant, more powerful tests. How? We're often faced with too many possibilities to test, too much data and Domain Testing helps us create a sample of the best representative data to use. In this workshop, we are going to provide an experience that helps all members of the team better understand the types of data we are using and how to craft powerful tests. We're going to work through a number of examples while testing an application.

Through sampling we'll find the best representative data we think will cause failures and find bugs. It doesn't matter how much testing you do, if you do any, you can learn from this course.

So, what about the rest of the team? What if Developers are more aware of Domain Testing in order to increase the effectiveness of their Unit "Checks"? What if Business Analysts are better prepared to provide examples that include better data points from Domain Analysis? Domain Testing is a technique that the whole team can use to reduce risk in the data that the application uses.

---



10:45 – 12:15

**W Speed-testing using mind mapping**

Sawgrass

*Speakers: Tomas Rosenqvist, Lena Wiberg*

Are your test sessions bogged down by excessive documentation and boring processes? Or do you find exploratory testing too uncontrolled and prefer structured tests and formal test reports? Come to this session to experience how using a visual model can increase your momentum without losing control.

In contrast to ad hoc or “intuitive” testing, Exploratory Testing expects us as professionals to have a systematic approach to testing. Structure and processes tend to be regarded as impediments to momentum, and when time is limited we start cutting corners. Cutting corners risks leaving us with a feeling of not being in control and bugs start slipping through cracks. We say this need not be the case. You can learn how to plan, test, adapt and report with minimal time by using visual models. We would like to invite you to a hands-on Exploratory Testing session, using mind mapping techniques for planning, executing and reporting our results. We will cover:

- Reviewing requirements (provided as a fictional business scenario)
- Planning, prioritizing and re-prioritizing your work
- Hands-on testing the actual web application and drawing conclusions
- Adapting to the unexpected
- Summarizing your findings and recommendations for further work in a concise and useful manner

You will be working individually or in pairs, and are required to bring a laptop. It is highly suggested that you download and install xMind (Freeware) and our Exploratory Testing template before the workshop. No previous experience with mind mapping is required. Having some familiarity with Exploratory Testing is helpful, but not a necessity.

**Takeaways**

- How mind mapping can boost creativity and drive during Exploratory Testing
- Ideas and a template for planning small test sessions
- A chance to find bugs!

12:15 – 14:00

**B Lunch and CASTiesta**

Horizons

Lunch, and then? This year, we're taking a longer mid-day break.

We're asking attendees to reflect on what they've learned, attend the Hallway Track, go to the beach, and prepare for more expert testing conversation in the afternoon.

14:00 – 15:00

**S Coaching Software Testing Using the GROW Model**

Sea Oats

*Speakers: Jose Lima*

Coaching is one of the most talked about topics of recent times. So what do you do when you are tasked of coaching developers on software testing? Come and find out as I explore and tell you about my experience using the GROW model to coach software engineers.

Coaching is an almost essential activity in today's software world. 18 months ago, Redgate Software, the company I work for, decided to shift the testing activities and quality responsibility from testers to software engineers.

The company didn't want to just drop those responsibilities to engineers without some help so they tasked some people to become Quality Coaches, which is my current job title.

As well as talking out about my journey as a coach, I would like to present an experience report about using the GROW model (from John Whitmore) to coach developers, and other roles including user experience designers, on software testing.

It's been one of the most interesting techniques I've heard about and studied before putting it into practice. It can be applied in any context and so far I had moderate success with it, although it wasn't all plain sailing and I still have a long way to go.

Takeaways:

- The different meanings of the word coaching
- That a quality coach does and doesn't
- The 4 GROW model stages
- Report on techniques to help overcome barriers that I've found along the way
- Real work situations where the GROW model helped and others where it didn't, and it shouldn't have been used
- Additional resources that relate in particular to the testing context

14:00 – 15:00

**S Don't Take It Personally**

Dunes 1/2

*Speakers: Bailey Hanna*

Receiving feedback can be tough. It can be hard to remember that it is meant to help improve work going forward, not to point out current flaws. It can be incredibly easy to take feedback or comments personally in the workplace, but what is the impact when we do so?

When we personalize situations we tend to lose sight of the bigger picture. It becomes easier to focus in on minute details and not look at the overall context in which the feedback is being given. The impacts of this lower level of focus can result in wasted time from; chasing the wrong issues, laying blame, making up excuses, refusing to ask for help, and ultimately avoiding discussions around the root cause and ways to improve.

This talk will draw on experiences and examples of situations such as; testing debriefs (tester to tester interactions), bug discovery (tester to developer interactions), and inter team projects (team to team interactions) and discuss tactics for each on staying objective and productive. When we look at feedback for what it truly is - a way to improve - we can build better relationships between communities and teams to make them stronger as a result.

By the end of this talk, attendees will know:

- How to better identify situations where you may be personalizing
- Ways to reorient thinking back to objectified view vs personalized view
- How to improve communication to avoid negatively received feedback

14:00 – 15:00	<b>S Risk Based Testing: Communicating Why You Can't Test EVERYTHING</b>	Sawgrass
<i>Speakers: Jenny Bramble</i>		
Do you feel like your team is under the gun to test everything when a new feature is rolled out? Do you worry that your team mates don't understand why you choose to test the items you do? Do you feel like you can't communicate those risks to them in a way they understand? Are there moments in your life where you deeply question if you can successfully complete the testing requirements of a sprint? Do you just really like cats? If you answered yes to any of those items—this is the talk for you.		
We will define and discuss risk as a tangible metric, striving to break it down into components that you can use to talk to developers, product owners, business people, and any other stakeholders. Having a common language of what risk is and what it's made of allows us to decide what we should test and when we should test it—and then communicate that to our stakeholders. We will also talk about building a risk matrix and why we should even bother. Included will be a heavy dose of jokes, storytelling, anecdotes, and pictures of my cat Dante.		
Top 5 Takeaways:		
<ul style="list-style-type: none"><li>• Definition of risk as a tangible metric (impact/probability of failure)</li><li>• Types of impacts that failure can have on your project and team</li><li>• How to define these terms for your team and facilitate communication</li><li>• Building a useful risk matrix</li></ul>		
15:00 – 15:15	<b>B Break</b>	Seagrapes
15:15 – 16:15	<b>S Cross-team Pair Testing: Lessons of a Testing Traveler</b>	Sawgrass
<i>Speakers: Elisabeth Hocke</i>		
Did you ever wonder how to improve your testing skills? Well, I did. I wanted to learn where I stood in terms of my testing knowledge and at the same time improve my exploration and automation skills. Maybe dive deeper into special areas like security or accessibility testing. I read and thought a lot, but what I felt I was missing was hands-on practice. So I decided to run an experiment.		
My hypothesis: "I believe that pairing and mobbing with fellow testers from the community on hands-on exploratory testing and automation will result in continuously increasing skills and knowledge as well as serendipitous learning. I'll know I have succeeded when I noted down at least one concrete new insight or applied one new technique per testing session and shared that with the community."		
In this talk, I will share the lessons learned on my journey as well as tips for doing pair testing sessions yourself. Let's uncover if my hypothesis proved true, that a testing tour is indeed a feasible and valuable way to improve your testing knowledge and skills!		
Key takeaways:		
<ul style="list-style-type: none"><li>• Hear about lessons learned as a testing traveler</li><li>• Get tips on how to make the most out of pair testing sessions</li><li>• Learn why collocation is not a requirement but an excuse and how you can benefit from geographical dispersal</li><li>• Get inspired to run experiments for your own personal development!</li></ul>		

15:15 – 16:15

**S Recruiting for Potential**

Sea Oats

*Speakers: Lena Wiberg*

In early 2017 I was promoted to QA manager and right off the bat thrown into two recruitment processes. I was terrified. I knew from previous experiences that I am really bad at traditional interviewing techniques and suddenly I could not even hide behind someone else making the decisions. During my career I've interviewed potential colleagues, team members and interns and I've always felt the outcome depended heavily on the candidate's confidence rather than my questions.

Our recruitment process included three interviews and three online tests. I felt it tended to favor glossy resumes and interview-trained professionals as well as being biased towards whatever personality type the recruiting manager had.

I wanted to do something different. Something that used my testing and programming background and that could be used to assess both juniors and seniors on an even playing field.

I started out looking for available exercises but the things I found were limited, generic and all focused on testing in front of other people. This also favors a particular type of person and in addition it wouldn't give me all the answers I wanted.

- How well do they read instructions?
- Do they have the guts to question?
- Can they make reasonable assumptions?
- How do they adapt to something unexpected?
- Can they document and communicate their findings?
- Can they answer questions about their work?
- ...

In this experience report I'll share my thoughts on why traditional interview processes are outdated and I'll show you an alternative way or doing it. I'll talk about successes, setbacks and how we plan to improve the exercise moving forward.

It's about figuring out what makes a tester, how to compare apples to biscuits and how you should always expect the unexpected.

In short: I will talk about putting candidates to the test.

Takeaways:

- Why standard recruitment processes are biased and focus too much on history
- Ideas on how to improve recruitment processes for testers or other roles
- How to design a scope small enough to handle but with enough challenge

15:15 – 16:15

**S SEBTE: A Simple Effective Experience-Based Test Estimation**

Dunes 1/2

*Speakers: Robert Sabourin*

Testers are challenged to provide accurate estimates. Magically test estimates become commitments always held firm even in the most turbulent context. Test estimates are based on shallow project insights. Estimators are often ignorant of key design, usage and environmental factors.

Many traditional project management estimation techniques fail when applied to testing projects. After all, what you test next depends on what you learned before and knowledge evolves as the project continues.

Rob Sabourin shares SEBTE: a simple effective experience-based test estimation method used for over 15 years on a myriad of project types (IT, E-Commerce, Apps), lifecycle models (Agile, structured, waterfall), and technologies (Mainframe, Web, embedded, mobile, hybrid).

Rob shows how SEBTE helps you collect and exploit meaningful historic data as a natural part of your testing work. You will learn how to use SEBTE on charter driven or task oriented work breakdown approaches.

Rob walks you through several entertaining real world examples which will help you put SEBTE into practise right away.

16:15 – 16:30

**B Break**

Seagrapes

16:30 – 17:30

**S Breaking the Surface**

Sawgrass

*Speakers: Susan Finley*

Being introduced to context driven testing is like coming up for air after being held underwater. That burning feeling in your lungs as you strive towards the light. Inching closer and closer, and then finally breaking through the surface and taking a big gasp of air.

You're now above the surface breathing in deep letting go of the frustrations of "quality assurance". You're excited and tell others but they doubt the concepts and aren't interested in CDT. You realize that getting others to breathe deeply with you is not easy.

This talk focuses on the lessons learned from the successes and failures of trying to change Quality Assurance into Context Driven Testing. Providing insights on organizational change concepts and the complexity of teaching others a different way of testing while enlightening your organization on software testing.

Takeaways:

- The importance of the executive sponsor and how they can support you.
- The power of enlisting others to become advocates for your change.
- The power of creating a sponsorship model to help you navigate through an organization and draw support from outside your team.
- Understanding when opportunities present themselves and how to capitalize on them.
- Using concepts like social styles to maximize your communication with others.
- Direction on how to get started and the small things you can begin doing now.

16:30 – 17:30

**S Defined by Your Own Tools**

Sea Oats

*Speakers: Amit Wertheimer*

*"If there's a pencil in your pocket, there's a good chance that one day you'll feel tempted to start using it" - Paul Auster, "The Red Notebook"*

The tools we use are changing the way we see reality and operate. Opening up the browser's dev-tools provides visibility to new areas we are unfamiliar with. Learning about consistency heuristics gives a mean to discuss why a behaviour is (or isn't) desired. Each tool we pick up expands our mind and our options, and many times we don't know we are missing a tool before actually using it.

In this talk I'll share my ways of finding new tools through examples of tools that had significant impact on my abilities as a tester in various ways:

- Encoded knowledge repositories (ZAP or WAVE)
- Transferable skills (xpath, scripting)
- Visibility enhancers (web proxy, resource monitor)
- Simple productivity tools (ditto, excel).

Takeaways:

- We only test what is easy. Tools make stuff easier.
- Take tools out of context to discover new ways to use them
- Explore the tool and find solutions to problems you didn't know you had
- Categories of influence, and how to use them

16:30 – 17:30	<b>S Highs and Lows of Managing a Neurodiverse Testing Team</b> <i>Speakers: Michael Tozer</i> Are all your colleagues quite similar to you? What do you think are the consequences of working with people who have a similar background and similar way of thinking? Or do you have someone in your team who thinks differently and surprises you with their unique way of embracing testing?  Today, change is happening across the industry as more companies are accepting neurodiversity and employing people who think differently. For some this is led by a social imperative; for others a business case as neurodiverse teams tend to outperform other teams.  We have spent this past year developing new opportunities in IT for people with autism. Together, we have carried out software testing for ecommerce, online retailers, not-for-profits and creative digital agencies and also placed software testers within one of Australia's largest banks and a major tech firm. We are seeing strong interest from across the industry from large international organisations to small local firms.  This is a growing challenge but also an opportunity that the testing community is facing right now. So far, we are one year into a five year process. Come and hear what we have learnt from our experiences so far.	Dunes 1/2
17:00 – 19:00	<b>A CAST Cocktail Reception</b> Join us for an evening reception to meet speakers and fellow CAST Attendees!	Horizons
19:00 – 22:30	<b>A Cocktails in the Hall of Heroes</b> Thanks to the generosity of our sponsor PerfTestPlus, we can announce a special addition to the schedule!  Meet in the lobby around 6:45 PM to load buses, they will leave at 7:00 PM sharp! We will take the first 100 registrants to Kennedy Space Center for an evening reception. Please see Valerie Gryfakis at the CAST registration desk to register.  Unfortunately, we can't accommodate any +1s or other guests.  If you are registered and driving please note that parking is free. Follow the signs to guest parking and park in lots 4 or 5.	Kennedy Space Center Visitor Complex

**A** ACTIVITY
**B** BREAK
**K** KEYNOTE
**L** LIGHTING TALKS
**S** SESSION
**T** TUTORIAL

**W** WORKSHOP

---

## AUGUST 9 • THURSDAY

---

07:00 – 08:00	A	<b>Yoga on The Beach</b> <i>Speakers: Dawn</i> Bring your mat out to the beach, and start your day centered	Beach
07:30 – 08:45	A	<b>Lean Coffee</b> <i>Speakers: Matt Heusser</i> Lean Coffee is a structured, but agenda-less meeting. Participants gather, build an agenda, and begin talking. Conversations are directed and productive because the agenda for the meeting was democratically generated.  The format for a Lean Coffee is very simple. This is intentional. It is meant to be the least structure necessary for a coherent and productive meeting. No more, no less.  <i>From <a href="http://leancoffee.org/">http://leancoffee.org/</a></i>	Horizons
07:30 – 08:45	B	<b>Breakfast</b>	Horizons
09:00 – 10:30	W	<b>Finding bugs before implementation</b> <i>Speakers: Katarzyna Balcerzak, Bart Szulc</i> What if you could spot a bug before implementation? Are there tools and techniques that can help you with accomplishing that? We're going to present you several ways of exploring problem space to find incorrect assumptions, unknown unknowns, and risks, that may have resulted in bugs during and after development.  Quality Assurance shifts left. We will show you how at SparteZ and Atlassian we've moved to the very edge. You will be get familiar with several techniques that support learning process, help grow of mutual understanding of a problem. With those techniques equipped you will be able together with your colleagues explore problem space and find invalid assumptions, unknown unknowns, risks no one considered, that may have resulted in bugs during development.  What are the three biggest takeaways you will take from the workshop? <ul style="list-style-type: none"> <li>• Learn about Quality Assistance model and how it works.</li> <li>• Learn what Feature Kickoff is and how to use a deck of cards, as a physical mnemonic, to drive the meeting.</li> <li>• Learn about feature decomposition techniques unveiling complexity from various perspectives.</li> </ul> How this presentation is going to affect you? <ul style="list-style-type: none"> <li>• You will shift left and help your organisation benefit from your critical mind and thinking as soon as feature idea is conceived.</li> <li>• You will start looking at a feature from various perspective and find even more relevant risks and scenarios worth exploring.</li> </ul>	Dunes 1/2

---

09:00 – 10:30	<p><b>W Mobversations - The Power of Observation in a Mob</b></p> <p><i>Speakers: Elisabeth Hocke</i></p> <p>“Amazing things happen when we are open to accepting help from others, and willing to help each other.” Woody Zuill, the initiator of mob programming, nails it here. Never heard of mob programming so far? It’s basically this: “All the brilliant people, working on the same thing, at the same time, in the same space, and at the same computer.” Though this might sound strange to some, it’s a wonderful approach to create learning situations and get the best out of everybody.</p> <p>While more and more teams try out mob programming, observation is a skill that can be easily overlooked. However, it can help you with many things: developing your product to be truly valuable, testing your application, improving how your team collaborates, or even your own behavior. Are you aware of what’s going on? Let’s discover together in this interactive workshop on how a mob can help to gather observations and make better decisions based on them.</p> <p>Key takeaways:</p> <ul style="list-style-type: none"> <li>• Learn the basics of mob programming and practice them hands-on</li> <li>• Sharpen your senses and level up your observation skills</li> <li>• Reflect on what actually happened to make informed decisions on how to improve things</li> </ul>	Sawgrass
09:00 – 10:30	<p><b>W What I Talk About When I Talk About Testing</b></p> <p><i>Speakers: Jim Warchol</i></p> <p><i>The language we use when describing our testing is important to understanding our craft. This workshop is geared toward clarification of the soft skills of our testing practices. We will share best outcomes to document new talking points which we can apply in our evolving careers as testers.</i></p> <p>“What I talk about when I talk about testing” is a workshop geared toward testers and non-testers alike. It will be a short presentation followed by a group discussion. We will share scenarios and situations we have encountered where we had to describe our work, or elaborate on our processes.</p> <p>This will be an overview of the non-technical side of software testing - the soft skills we bring to the agile process, and the communication and perception challenges that we sometimes face in our careers.</p> <p>How do you describe the testing that you are doing to the stakeholders on the project? How do you respond to questions in a job interview? How do you interact during discussions with recruiters, or with account people who are selling your skillset to a client? How do you quantify what you do in order to sell yourself and software testing accurately? The language we use is very important in a craft that is often misunderstood.</p> <p>It’s important to explain that it’s not just what we DO as software testers, it is equally important to show how we interact with other development team members. We need to describe how the testing process often goes much deeper than just ‘checking’ if the software behaves according to requirements. We possess soft skills that are just as important as our technical skills. But how do we convey that?</p> <p>We will have a group discussion about our interactions with the following individuals and uncover a list of good practices:</p> <ul style="list-style-type: none"> <li>* Project Stakeholders</li> <li>* Product Owners</li> <li>* Developers</li> <li>* Scrum masters and project managers</li> <li>* Account &amp; sales people</li> <li>* Talent acquisition / Recruiters</li> </ul>	Sea Oats
10:30 – 10:45	<b>B Break</b>	Seagrapes



10:45 – 12:00	<p><b>K Designing Test. Testing Design.</b> <span style="float: right;">Sea Oats</span></p> <p><i>Speakers: John Cutler</i></p> <p>Over the years, I've noticed many shared experiences between the "specialist" roles: UX, visual designers, testing, data scientists, architects, security analysts, operations, and even agile coaches.</p> <p>In particular I started to notice the similarities between UX and testing, including a core interest in the quality and reliability of user interactions (where humans touch the machine), an appreciation for the complexity of those interactions, and a passion to explore "what could go wrong..." in a quest for better (and safer) outcomes.</p> <p>And one thing everyone shared: trying to "fit" into a framework (Scrum, for most of us) that never really worked out the kinks of our involvement.</p> <p>Which got me thinking...how might we better design the interface to (and the experience of) working with testing? How might we use service design, org design, product thinking, and systems thinking to address the information asymmetries / blind spots that leave testing in a perpetual state of push-based advocacy? What can testing learn from UX? And what can UX learn from testing?</p> <p>My hope with this talk is that we can explore these shared experiences and questions with stories from my career, a bit of theory, a lot of practice, some drawings/doodles, and (likely bad) jokes.</p>
12:00 – 14:00	<p><b>B Lunch and CASTiesta</b> <span style="float: right;">Horizons</span></p> <p>Lunch, and then? This year, we're taking a longer mid-day break.</p> <p>We're asking attendees to reflect on what they've learned, attend the Hallway Track, go to the beach, and prepare for more expert testing conversation in the afternoon.</p>
14:00 – 15:30	<p><b>W Let's Hack Your Team and Organization w/ The Best Inquiry Skills Ever</b> <span style="float: right;">Sea Oats</span></p> <p><i>Speakers: Andrea Chiou</i></p> <p>In this interactive workshop, you will be introduced to and practice some of the basic techniques of Clean Language and Systemic Modeling. Clean Language is a set of 12 basic questions that are non-leading and assumption free. Systemic Modeling is the art of using Clean Language to explore mental models in individuals or groups. It is incredibly powerful in helping a group to discover its own symbols and metaphors, and will lead members to pay exquisite attention to each other. Group communication becomes a platform for sustainable inquiry, learning, clarity and trust.</p> <p>In this intro to Clean Language and Systemic Modeling workshop, get a taste of some of the questions, techniques and models, such as:</p> <ul style="list-style-type: none"> <li>• Metaphor Warmup, have fun loosening up that analytic brain.</li> <li>• Two Lazy Jedi Questions, learn and practice the two most useful questions on earth for inquiry.</li> <li>• An Intro to the Structure of Clean Questions, how is the syntax of clean question in fact 'clean' ? Receive a copy of all of the questions, and how each is used.</li> <li>• Working At Your Best - create and share a symbolic landscape. We'll use the questions on each other to learn each other's best state for working.</li> <li>• Clean Setup - what questions can people can use to prepare for meetings, facilitations, study sessions, workshops, trainings in a collaborative manner - also using symbols and metaphors?</li> <li>• Clean Feedback - you'll learn to separate observation from meaning and impact. This can be used to reflect back on both positive and negative and to think forward about what might work better.</li> </ul> <p>You'll use these techniques to encourage curiosity and diversity. You and your team members will have a powerful tool to use to diffuse conflict and misunderstandings, give effective feedback, setup for great collaborative work. You'll be able to celebrate different opinions and ways of working, and to help enhance each other's thinking.</p> <p>Simple enough to start using right away, rich enough that if practiced with others, it will start to amplify your collective results at work.</p>

14:00 – 15:30

**W Mobile and Chatbots: Effective Testing**

Dunes 1/2

*Speakers: Carl Kibler, Rachel Kibler*

Want to learn about testing mobile apps? Want to break a mobile app? And develop your own heuristics to boot? Come break a chatbot with a mobile dashboard and learn how close collaboration led to the concept and execution of a new app to track mental health!

**Description:**

TYMS (Track Your Mental State) is a chatbot with a dashboard in a mobile app. It gathers information about the end user's mental health daily, to help with tracking things like bipolar. The app is currently under development. This workshop, presented by the tester and the developer, will allow participants to test the app while addressing heuristics for mobile testing (including common issues specific to iOS vs Android), ways to look for bias in natural language processing, and how to think about stress cases when developing test strategies. Participants will develop a mindmap they can use with their own heuristics for mobile testing. And they can report bugs directly to the developer!

The developer will talk about communication skills and styles that are helpful when working closely with a tester, and about how his engineering changed based on what he knew the tester was plotting.

**Takeaways:**

1. understand complexities in mobile and frequent break points
2. create mindmap of heuristics for mobile testing
3. hands-on mobile testing experience of test and prod builds
4. understand how tool-based testing can be helpful (and how it fails)
5. think about alternatives to use case testing

14:00 – 15:30

**W Unlocking Your Creativity for Creating Test Ideas**

Sawgrass

*Speakers: Paul Holland*

Have you ever seen a bug that was found by a customer and thought: "Really?", "Why would they have done THAT?" or "Why didn't I think of that?"

It's possible that you missed testing for that scenario not because of a lack of time or resources but instead because you lacked the creativity to imagine the substantially different ways your customers might try – or will try - to use your product(s).

In this workshop, Paul will show ways to improve your creativity and imagination. Based on years of experience and multiple various workshops and exercises Paul has created some approaches and ideas on how to identify new and creative test ideas. There will be hands on exercises to demonstrate the effectiveness of avoiding reading the detailed specifications (at least initially), of brainstorming, and of using past "test escapes" to prevent future issues.

**Takeaways:**

- Do not start a testing task by reading the detailed specifications or stories
- Use Brainstorming to build on your own ideas and learn from others
- Use your own and others' customer found bugs to help plug holes in your coverage
- Use Idea Generating cards to help you think in different ways
- Give yourself boundaries of space and time to be creative
- Give yourself permission to be curious

15:30 – 15:45

**B Break**

Seagrapes

15:45 – 16:45	L	<b>Lighting Talks</b> <i>Speakers: Dawn</i> Come present a long-standing suspicion, a burning question, or reaction to something you've heard at CAST.  Signups will be available during CAST.	Sawgrass
15:45 – 16:45	S	<b>Testing in Uncharted Waters - A Journey of a Testing Consultant in the World of the ERP Systems.</b> <i>Speakers: Aleksander Lipski</i> A journey I am going to talk about started like this: experienced testing consultant without previous domain knowledge enters the world of enterprise resource planning systems to propose improvements. He fails.  In theory, ERP systems are like any other software systems. The reality is slightly different, they are far away from being similar - starting from the meaning of the technical words, good practices, life cycle, not to mention type of stakeholders. Most of ERP stakeholders have heard about software testing, have been testing for ages but have never met nor worked with dedicated software testers before.  The story I am going to share will be about bridging between these worlds.  Challenging the status quo in any environment can be often described by the quote: "If you want to make enemies, try to change something". When you add to the above lack of previous experience with given software community you will understand the difficulty of that ongoing journey and how it has become already a source of lessons learned for both sides.  Takeaways: <ul style="list-style-type: none"><li>• How to persuade others that software testing is a skill that can bring value from the very first moment</li><li>• How to collaborate and learn from each other when we seem to be on different sides</li><li>• What not to do when you enter a new software community without previous experience nor authority in that domain</li><li>• Get inspired by entering new software communities to broaden your expertise and knowledge</li></ul>	Dunes 1/2
15:45 – 16:45	S	<b>Wearing Hermione's Hat: Narratology for Testers</b> <i>Speakers: Marianne Duijst</i> When we find those pesky bugs in production, hindsight often allows us to see what logical leaps and assumptions we made, and how we could have caught it, had we only known. Narratology is study of the construct of stories. I'll show how to apply its lessons to our software development and testing using examples from the Harry Potter narrative.  Takeaways: <ul style="list-style-type: none"><li>- The basics of narratology: what it is and how to apply it to software development and testing.</li><li>- Ways to reexamine your own assumptions and foster critical thinking</li><li>- The building bricks of storytelling</li></ul>	Sea Oats
16:45 – 17:00	K	<b>CAST Closing</b> <i>Speakers: Eric Proegler</i> Let's say goodbye! We'll miss you - until next time!	Sea Oats