

Lessons Learned Implementing Agile

SQDG

November 17, 2009

Janet Gregory, DragonFire Inc.

With material from Lisa Crispin



Copyright 2009 Janet Gregory, DragonFire

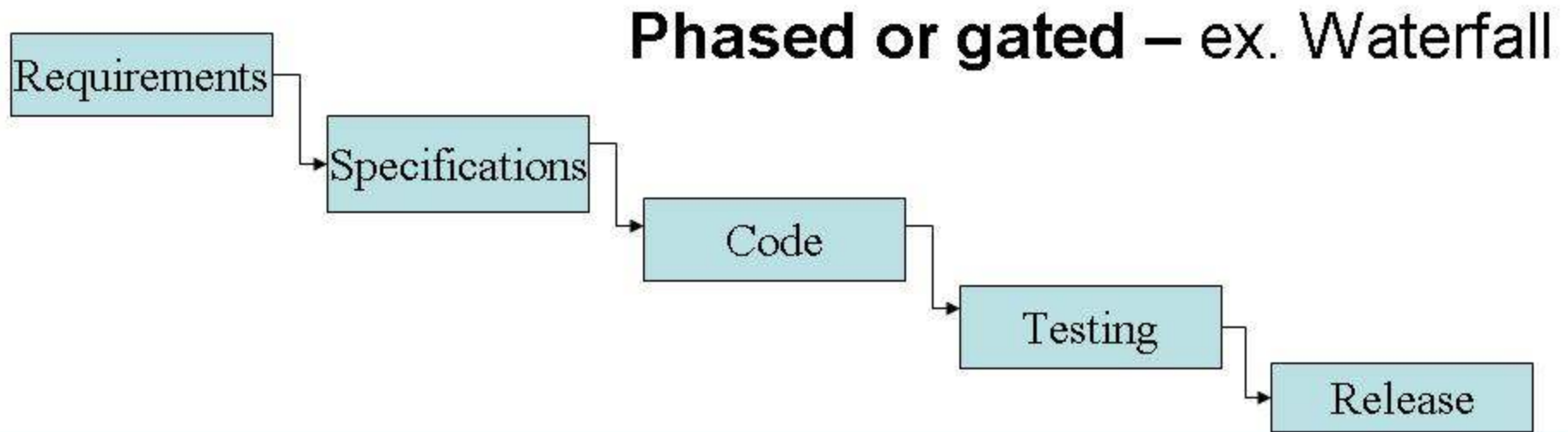
Takeaways

- Symptoms vs.. problems
- Some lessons learned
- How to recognize some common problems
- Practical steps you can take today

..... But, first a bit of agile so everyone is on the same page

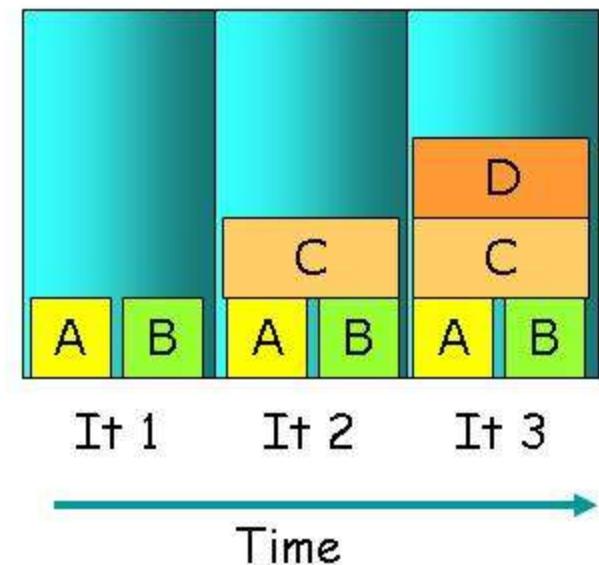


Traditional vs. Agile SDLC



Agile: iterative and incremental

- Each story is expanded, coded and tested
- Possible release after each iteration



Agile is a term to describe methodologies that:

- have short iterations
- encourage active customer participation
- demand whole team collaboration
- test features as they are coded
- deliver business value at regular intervals
- adapt their processes based on feedback

...and so on.....



Are you solving a problem or a symptom?



- Question -- What is the real problem?
- Ask ---Why, why, why, why....and why?
- Use experts
- Too many times we solve the wrong thing
- Use retrospectives to identify symptoms



Problem 1: Calling it “Agile”

Teams call it agile, and say it doesn't work.

Start by questioning ...

- Do they understand the values and practices or take the myths as reality?



Are You Doing Agile? How Do You Know?

Questions I ask...

- How big are your iterations?
- Do you have continuous integration?
- Are your stories “done” at the end of an iteration?
- What does “done” mean to you?
- Do you have a potentially shippable product every iteration?
- Are your regression tests automated?



Before you claim you are doing agile, ...

- Really understanding what agile means
- Ask experts
- Follow critical practices
 - Short iterations – 2 weeks
 - Potentially shippable product every iteration
 - Collaborate: testers, programmers, customers



#2: The team structure didn't really change

- Still have a separate test team
- Not everyone participates in planning sessions
- Still have functional silos



Encourage the Whole Team Approach

- The team committed to testing, quality
- The team solves problems
- All team members participate in planning sessions
- Anyone can do any task



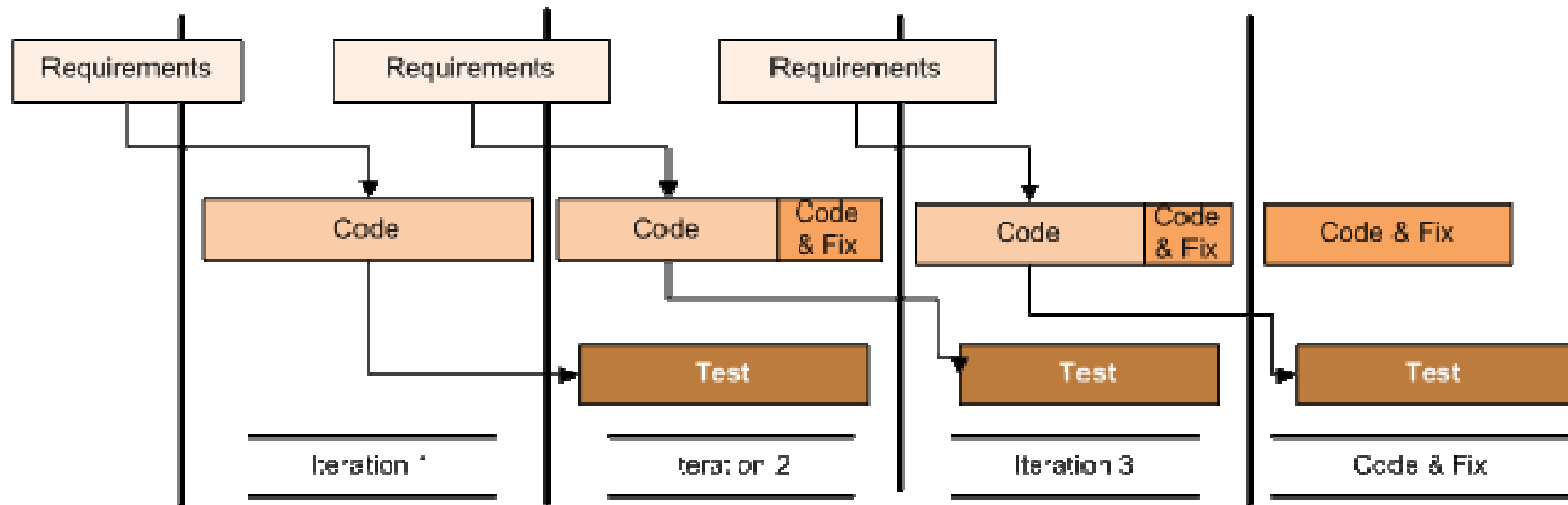
Acknowledge training needs

- Changed Roles
 - Functional Managers, testers, programmers, customer
- Agile principles
- Communication strategies
- New testing terminology



Problem 3: Practicing Mini-Waterfall

- Your testing is at the end of the iteration or..
 - in the next iteration



Symptoms

- Stories are not “done”
- Your team has defined “done-done”
- Bugs are left until later to fix
- Testing feedback is too late to change anything

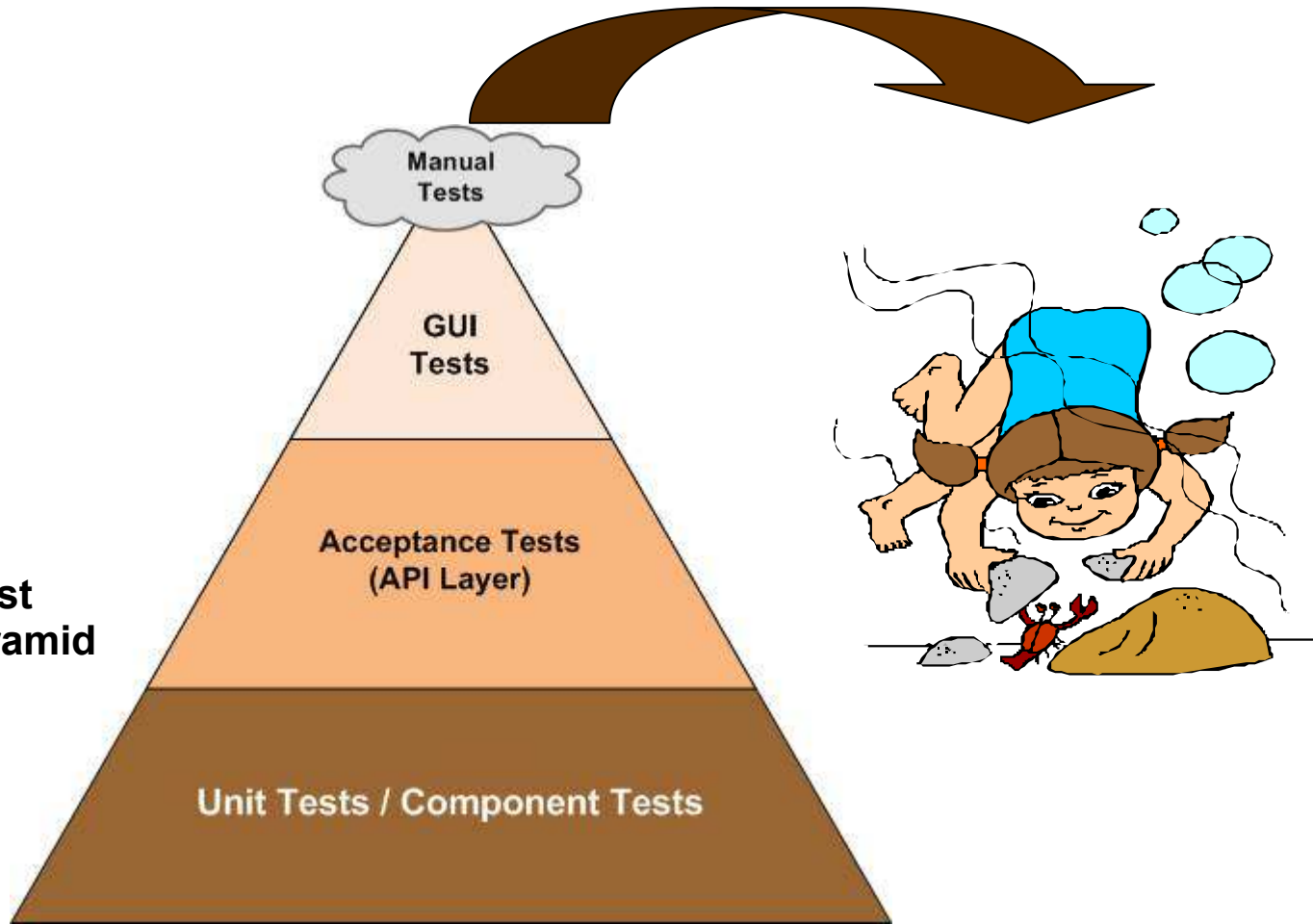


Try ...

- Breaking stories up smaller (<3 days)
- Learn to slice the stories vertically
 - Create feature teams
 - Rather than component teams
- Define acceptance tests during planning
- Give tests to programmers before coding starts



Automate your regression testing to enable you to do more exploratory testing.



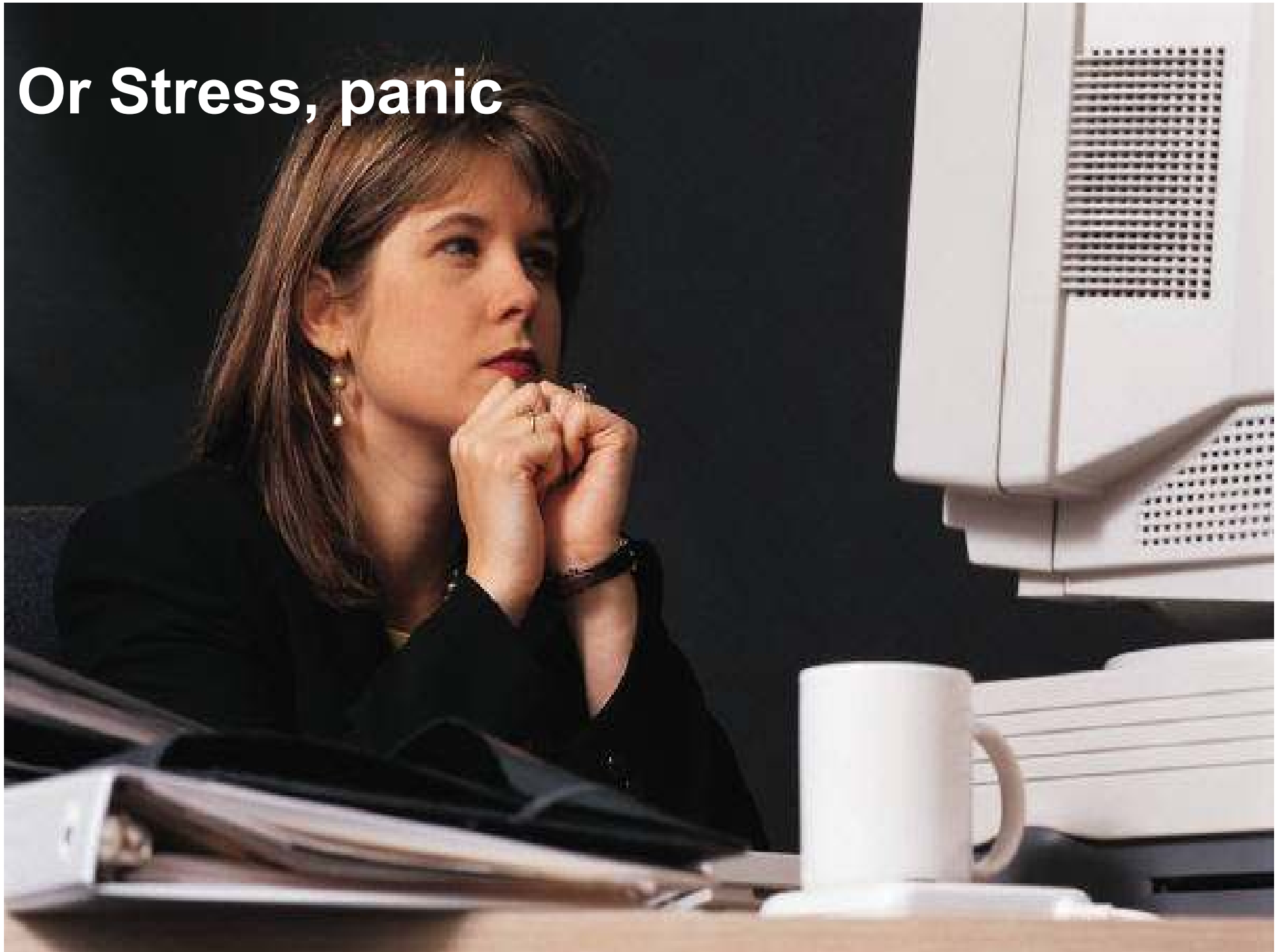
Mike Cohn's Test Automation Pyramid



Problem 4: Complacency



Or Stress, panic



Recognize complacency or panic when...

- Old habits resurface
- New people influence your process negatively
 - Ex. you start building requirements documents again
- Forget to keep the code clean
- Forget to keep the tests running green



Focus

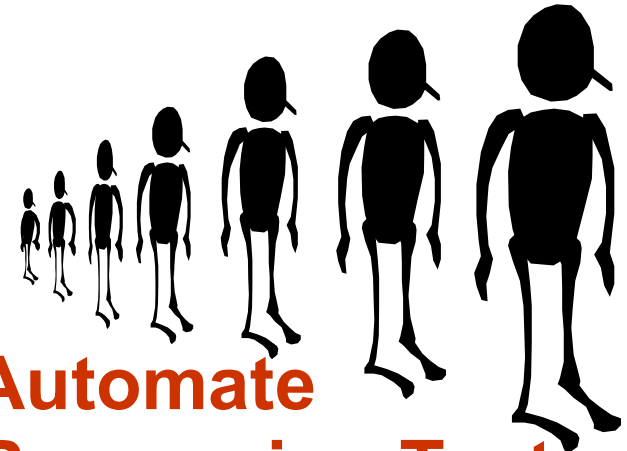
- Instil a learning organization mentality
- Shu Ha Ri
- Make the process visible
- Make the metrics visible
- Understand the why behind the process
- Have a coach who is monitoring the process
- Use your retrospectives to find problems

守破離





**Whole Team
Participation**



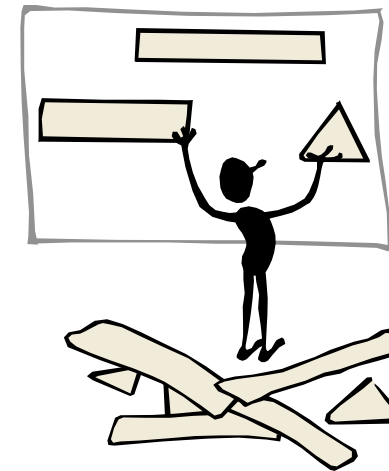
**Automate
Regression Tests**



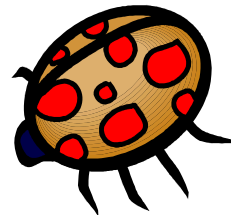
Collaboration



**Practice, prevent
complacency**



Be involved



Provide Feedback

Copyright 2009 Janet Gregory, DragonFire



Now Available

*Agile Testing: A Practical Guide
for Testers and Agile Teams*

By Lisa Crispin and Janet Gregory

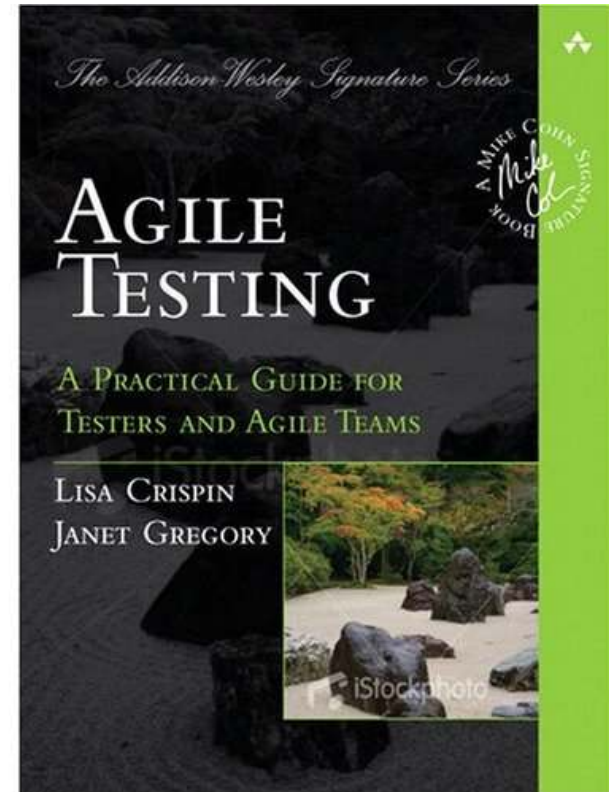
www.agiletester.ca

My contact info

www.janetgregory.ca

<http://janetgregory.blogspot.com/>

janet@agiletester.ca



Copyright 2009 Janet Gregory, DragonFire

Agile Resources

- www.lisacrispin.com
- agile-testing@yahoogroups.com
- www.testobsessed.com
- www.testingreflections.com
- www.mountangoatsoftware.com – Mike Cohn's web site (and all his books)
- Mary Poppendieck and Tom Poppendieck, *Lean Software Development*, 2003 Addison-Wesley (series of 3)
- Jean Tabaka, *Collaboration Explained*, 2006 Addison-Wesley
- Lisa Crispin and Tip House, *Testing Extreme Programming*, 2002 Addison-Wesley
- Agile Manifesto: <http://agilemanifesto.org/>

Let's talk about your problems ?

