

Once upon a time, a team was trying to develop a Linux based Firewall Appliance. The team and the manager agreed to produce a beta version in 4 weeks. That very afternoon, the manager went to one of the developers on the team with a cool idea, which would only be a little bit of work. The developer said yes. But the team did not get its firewall ready on time.

Does this story sound familiar? Well that manager was me and it took me a long time to realize what I had done wrong.

A team is to a group what a laser is to flashlight. Focused, powerful and amazingly versatile. Scrum is a framework for unleashing the power of effective teams.

As Project Manager, Test Manager or SW Developer,
I want to deliver software successfully, efficiently and on time.

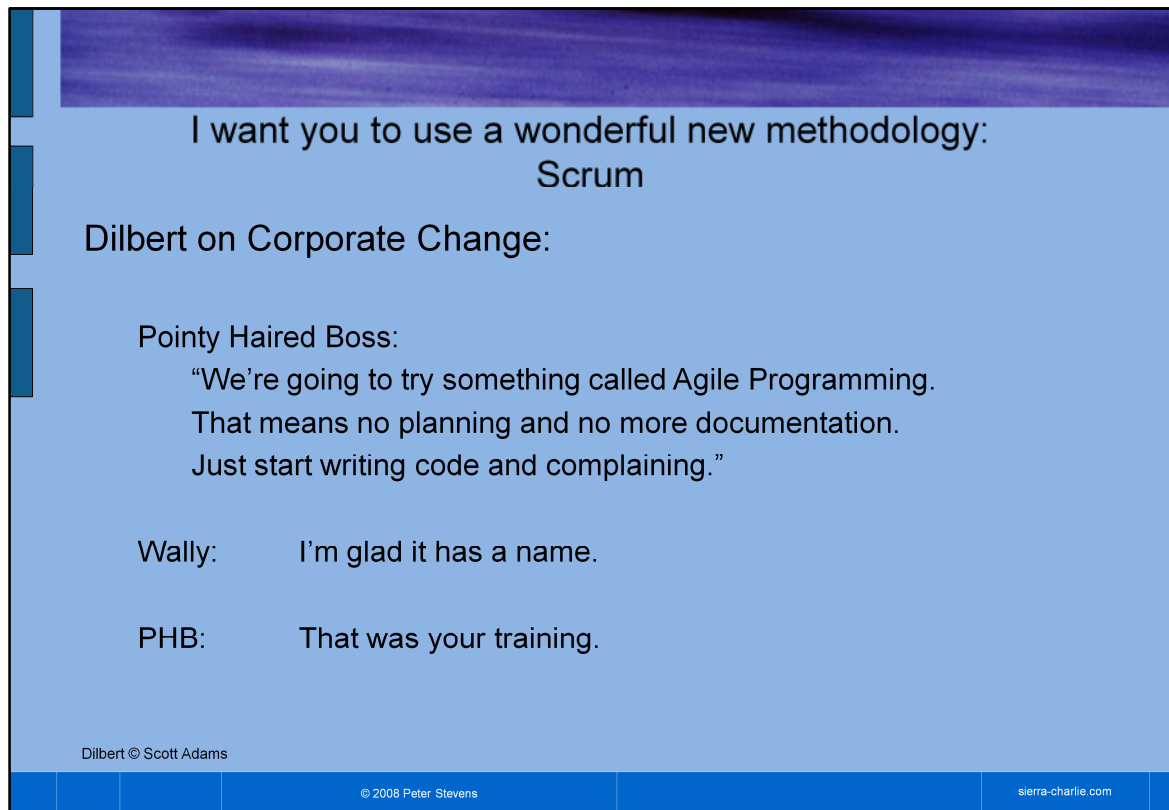


Photo © Julián Rovagnati - fotolia

© 2008 Peter Stevens

sierra-charlie.com

We are all here to build solutions for our customers, right? Or sometimes does it seem like some peop



If this is your (management’s) attitude to Scrum & Agile, you are unlikely to succeed.

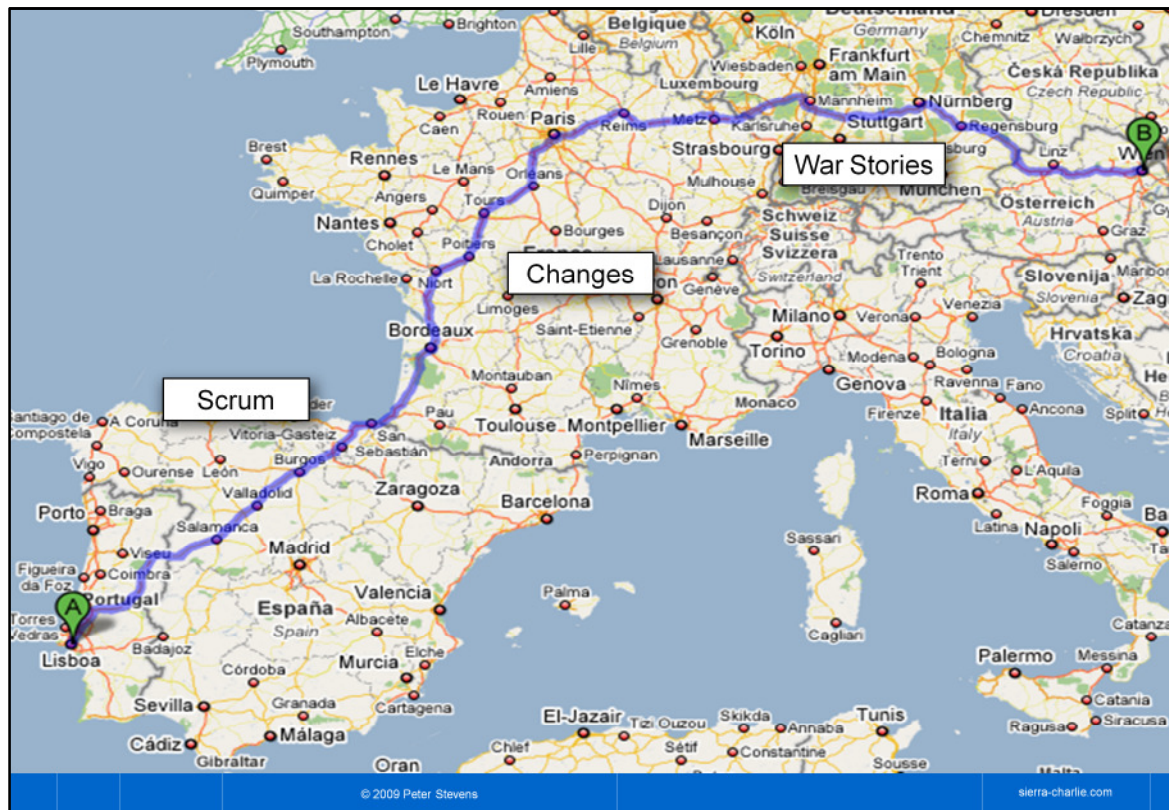
Dilbert on Corporate Change:

Pointy Haired Boss:

“We’re going to try something called Agile Programming.
That means no planning and no more documentation.
Just start writing code and complaining.”

Wally: I’m glad it has a name.

PHB: That was your training.



Implementing Scrum, like implementing a complex project, is a long voyage with many surprises and challenges on the way.

I can't tell you everything today, but I'd like to talk about some of my experiences on the trip.

Scrum was a process which made sense to me. Unlike other processes, which seemed heavy and bureaucratic, Scrum was focused on producing software and working together. At the same time, it implies (or enables) major changes in the development process.

Roles Change: I'd like to take one example: How the role of testing and testers changes

I'd like to tell you a few of my Scrum Stories (and maybe show you some Scars).

If you prefer to develop software in a waterfall,
then please continue doing so.



Photo: (CC) joeslabotnik/flickr

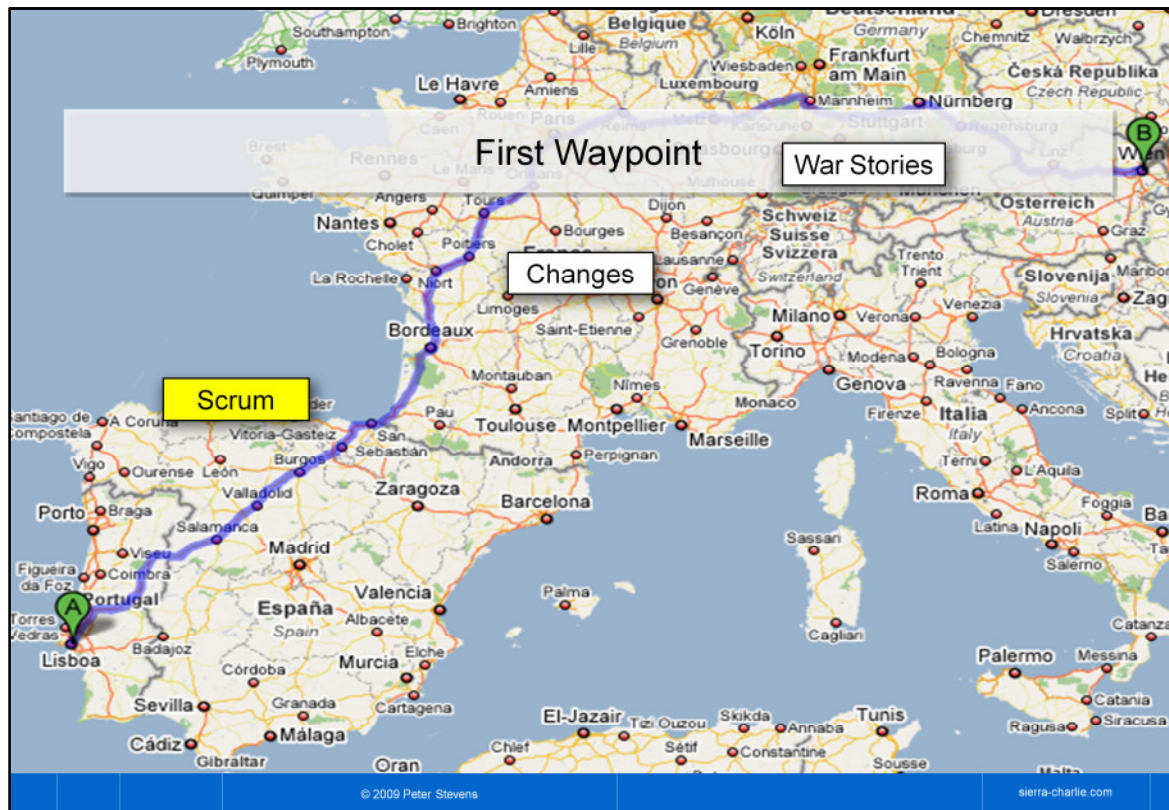
© 2008 Peter Stevens

sierra-charlie.com

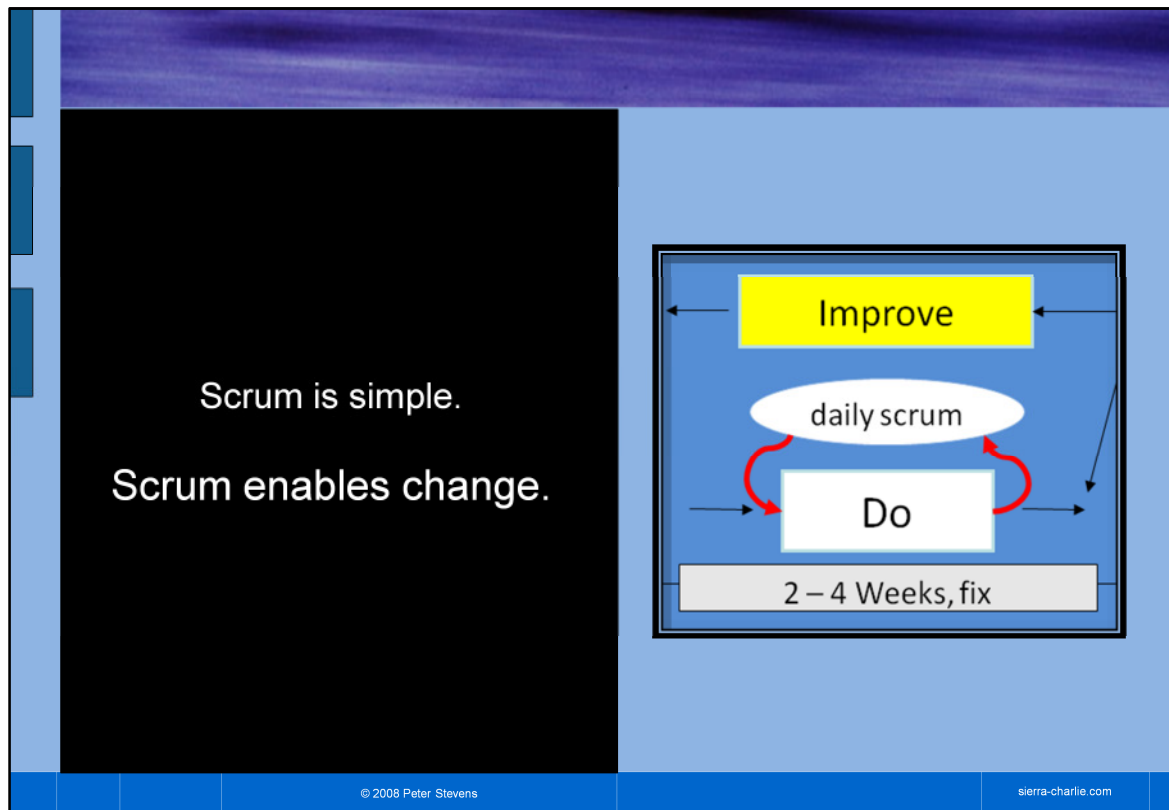
Before we start, I like to visit the waterfall.

Scrum is not for everyone or every project. If you think a waterfall (or some other) approach is better f

Are there any skeptics or pragmatists among you? If so, that's fine. I'd like you to collect really tough qu



To start, I'd like to take a brief look at "what is Scrum?"



Scrum is an agile management framework, which emphasizes teamwork, transparency and discipline.

Scrum is simple: 3 Rolls, 4 Meetings, 5 Artifacts.

Scrum is hard. The rules are demanding and many people don't want to apply the necessary discipline.

Scrum has three clearly defined roles.



Product Owner



Scrum Master



Team

Photos © Aramanda, Forgiss & Michael Chamberlin - Fotolia

© 2008 Peter Stevens

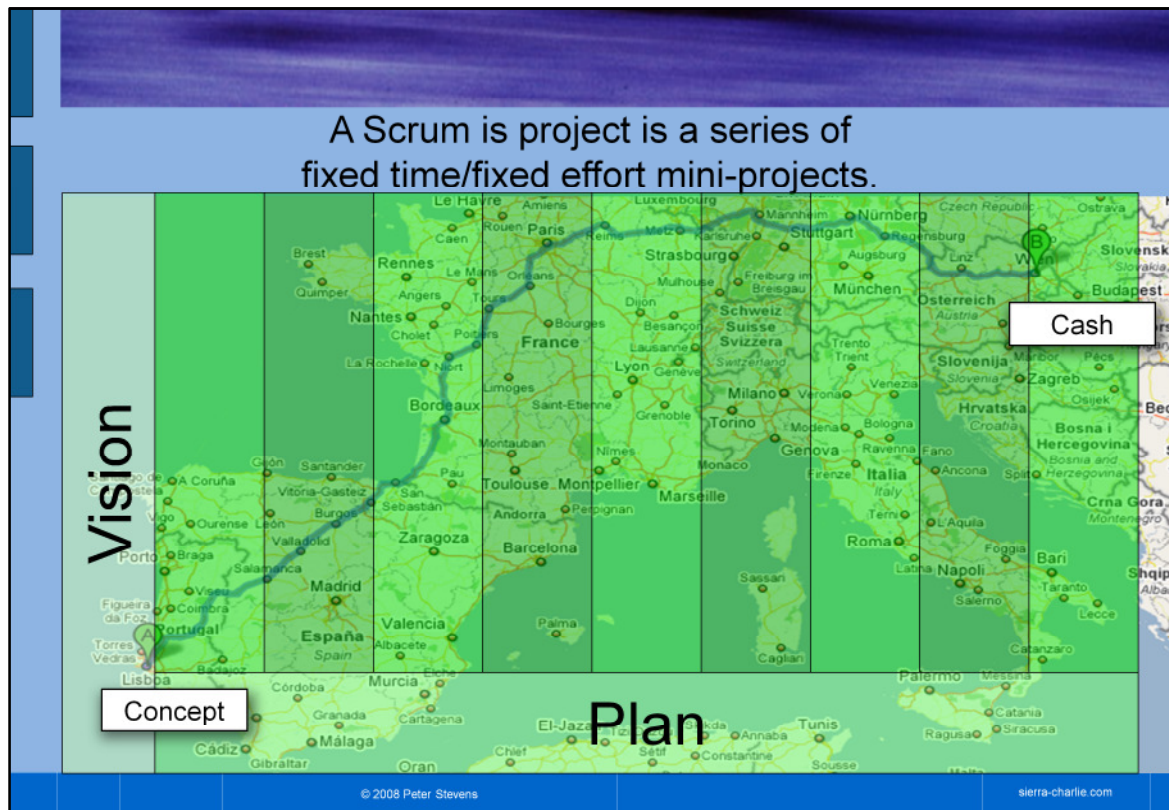
sierra-charlie.com

The Product Owner is responsible for Vision, Focus and Flow... & ROI.

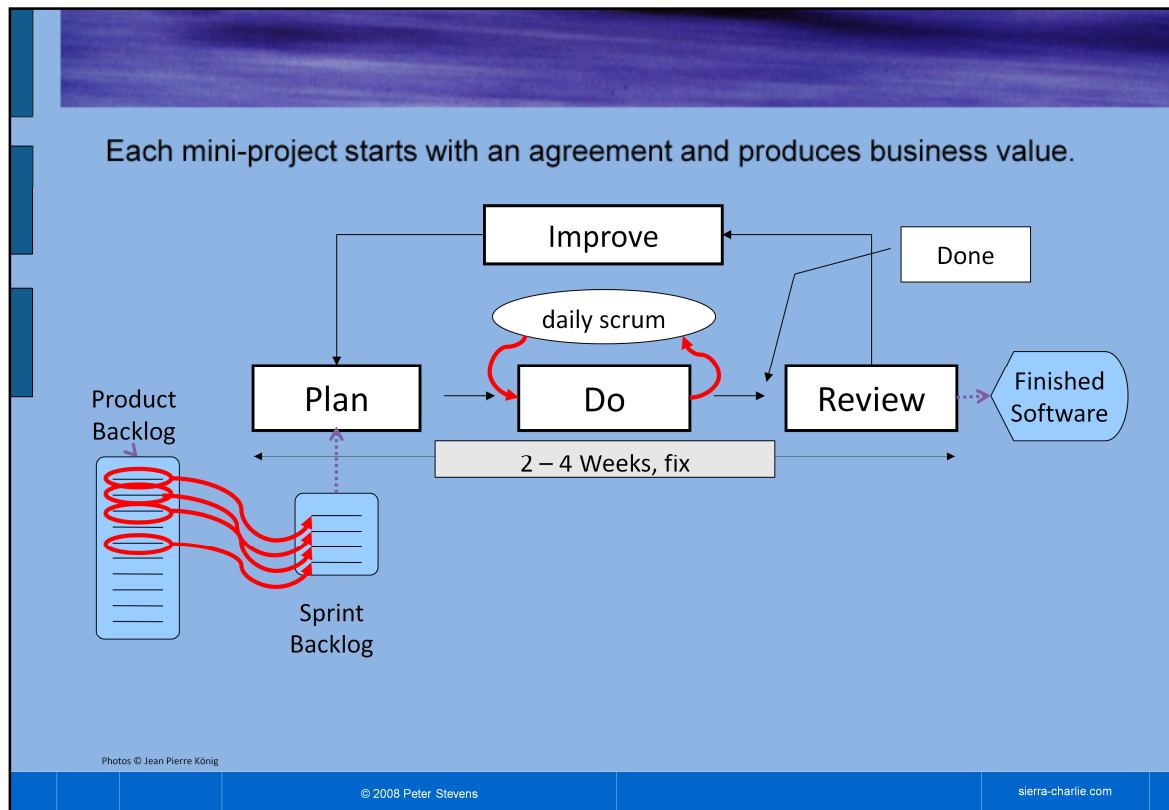
The Team delivers the solution and is responsible for the success of each Sprint.

The Scrum Master removes impediments and helps Product Owner and Team do their jobs better.

The interests of other stakeholders are represented by the Product Owner.



Contrary to popular belief, an agile team probably invests more effort into planning than a waterfall team, but that effort is distributed over the duration of the project.



Start with a wish list. (In Scrum this is the Product Backlog)

Objective is production ready software.

The development period is limited to 4 weeks or less. So it is necessary to limit the volume and size of the functions to be implemented during the Sprint

The Team and Product Owner agree on the scope of the sprint and the definition of done. The team does its best to deliver what it promised.

Inspect and adapt (continuous improvement is applied, daily, monthly and quarterly).

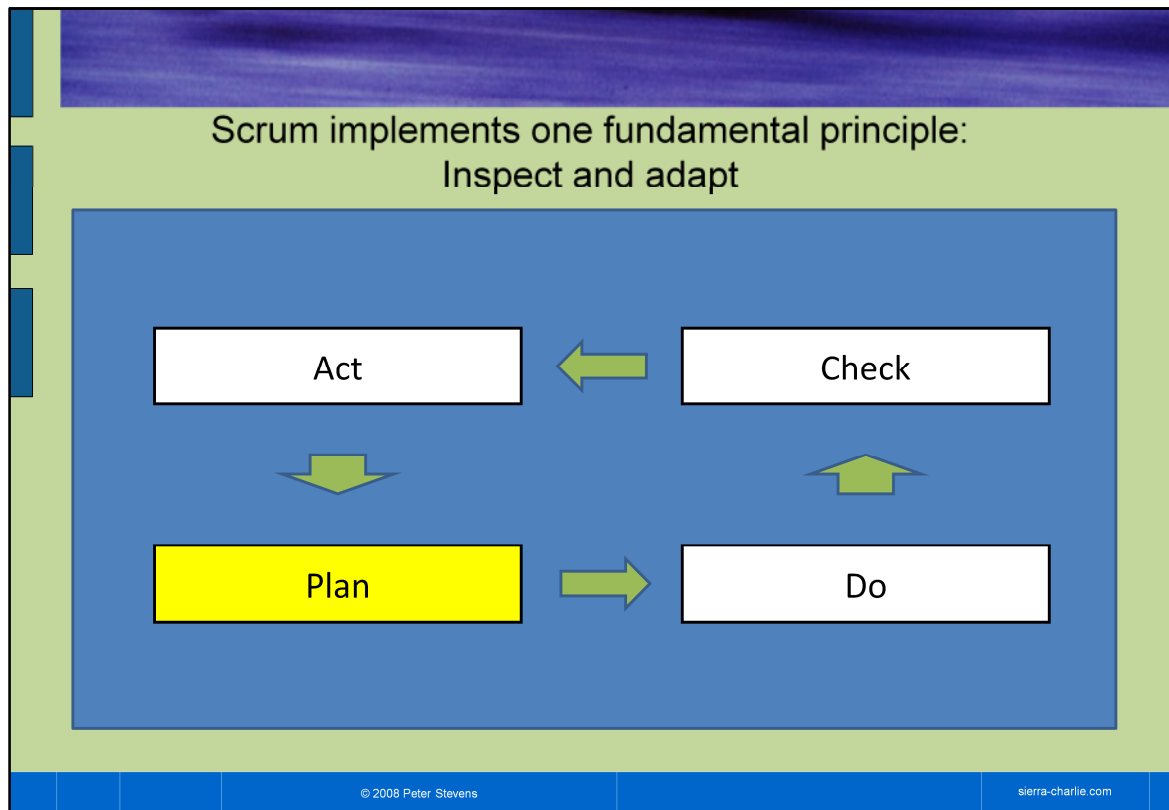
What's different about Scrum?

Scrum implements one
fundamental principle:
Inspect and adapt

Scrum turns excuses into
calls for action.

Scrum redistributes
responsibility.

Scrum turns groups into teams



At the stable, a fence is broken. The horses (cows, whatever) have escaped.

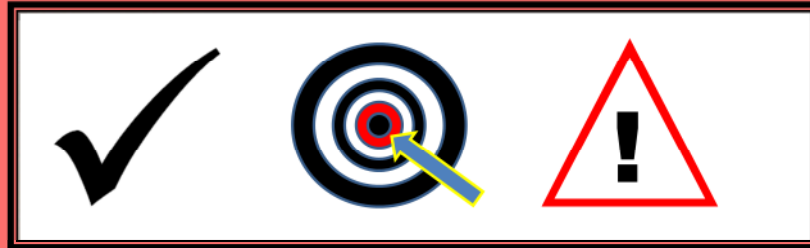
Do: retrieve the horses

Act: Fix the fence

,Don't bother me with the fence! I have to catch the horses!' OK, but what happens tomorrow? The fence is still broken.

You have to catch the horses. And you need to fix the fence.

Scrum turns excuses into
a call for action.



© 2008 Peter Stevens

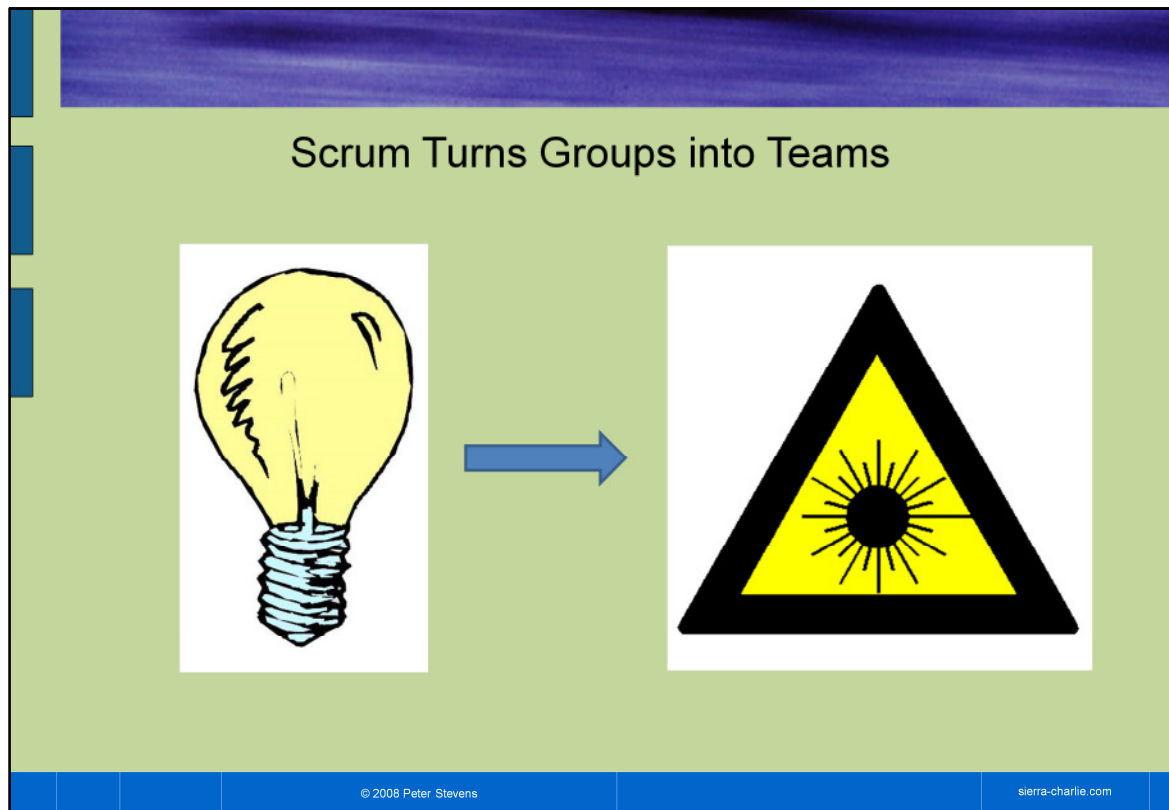
sierra-charlie.com

Scrum systematically identifies impediments and charges with Scrum Master with eliminating them.

Conflict arises quickly. First inside the team. Then with the Product Owner. Then with management or with other teams. These conflicts test management's commitment to Scrum.

Scrum redistributes responsibility. What happened to the Project Manager?

	S-M	P-O	Team
Select Team Members	initial		once established
Select Tools	shared		shared
Plan Tasks			X
Ensure satisfactory implementations			X
Define and Impose Standards			X
Coordinate Work			X
Budget		X	
Scope		X	
Set Priorities		X	
Commit to Delivery Dates		shared	shared
Assign Tasks			X
Communicate with the Customer			X
Remove Impediments	X		



What is the difference between ordinary light and a laser?

A bulb produces white light – the light is at multiple frequencies, going in all directions and produces more heat than light.

Laser – the light is all on the same frequency going in the exactly the same direction. A laser pen can illuminate a point across the room by daylight. A laser can read bits on a DVD. A laser can measure the distance to the moon (which is increasing by 38mm / year).

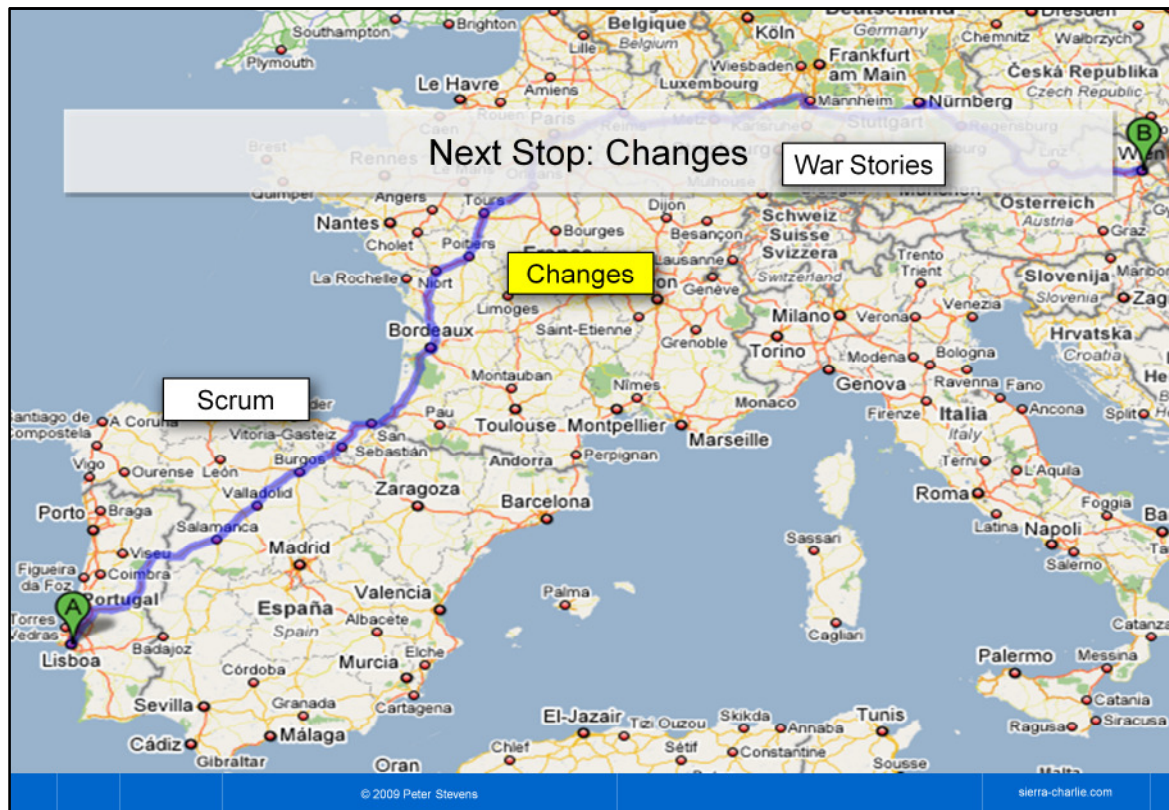
A group is a light bulb – bright individuals, but individuals going in different directions

A team is a laser. Focused, synchronized, with incredible potential.

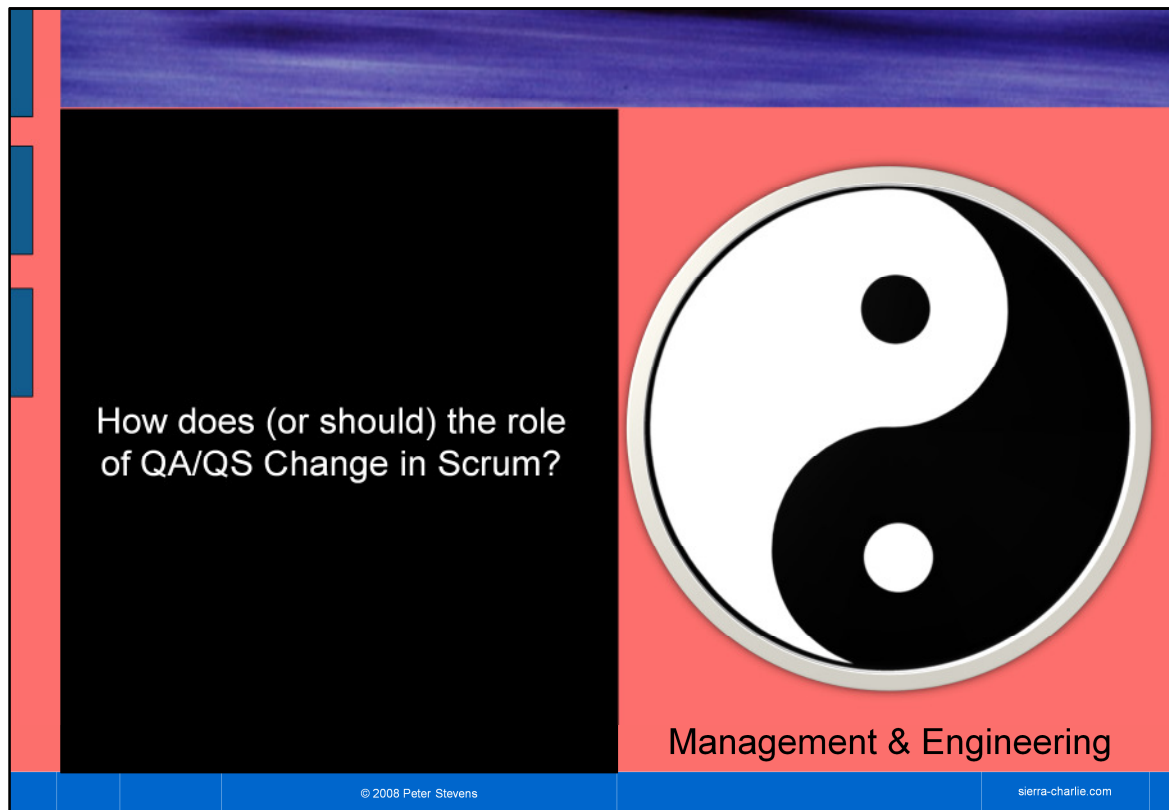
If you do Scrum, you can turn your groups into genuine teams.

Question: if you had to pick just one group in your company and turn it into an laser-team, which one should it be?

Answer: Top Management

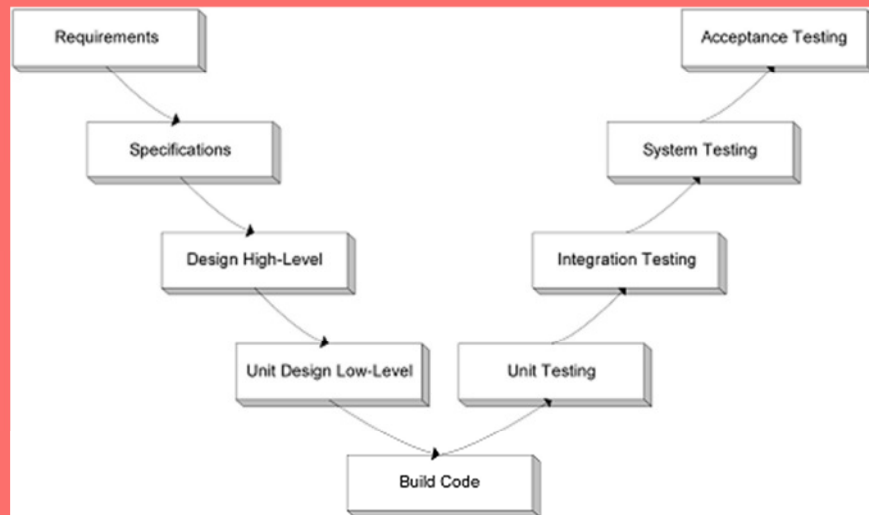


Next waypoint, let's look at how one role can change under scrum



Scrum does not specify engineering practices. But teams quickly realize that Scrum demands A-1 engineering practices. So let's look at how this affects the role of testing.

Why is a phased approach sub-optimal?



Source: <http://www.tfrc.gov/safety/pubs/04080/02.htm>

© 2008 Peter Stevens

sierra-charlie.com

Does anyone here use the V-Model?

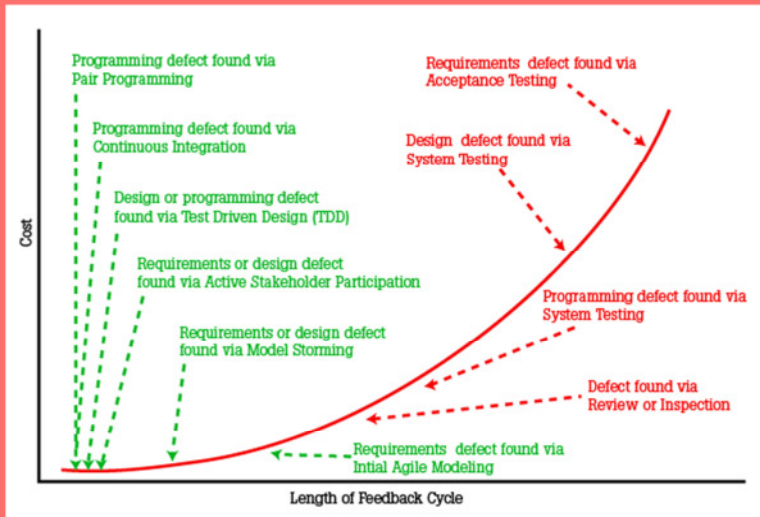
For each implementation level, there is a corresponding test level.

When do the testers write/write down tests?

Do the testers work with or even talk to the developers? “We don’t want the developers guesses about the specs to influence the testers guesses”

What happens when **acceptance** tests are written at the end of the development process?

The longer the delay between making and finding a mistake,
the more expensive it is to fix.



Source: "Pitching Agile to Senior Management" <http://www.ddj.com/architect/199300107>

© 2008 Peter Stevens

sierra-charlie.com

A “Lean” approach to testing

Acceptance Testing

- Does the system conform to business requirements?
 - Are the features present?
 - Are they usable for the user?
- Maybe automated, maybe not

Developer Testing

- Does the software work as the developer intended
 - Do components work properly?

Example

- Calculate Wage for Hourly Staff:
 - An employee receives an hourly wage
 - An employee receives time and half (150%) for overtime
 - An employee receives double time for night shift and Sundays

© 2008 Peter Stevens

sierra-charlie.com

If this were a class, I would ask 8 people to leave the room. The I would invite one to come in. I would give him or her this specification – verbally. The he would pass it on to the person, etc. until...

Until what? What do you think happens as the specs get passed on?

After 4 or 5 hops, the spec is pretty unrecognizable.

But you say wait, we need the spec in writing. OK, here it is in writing. Everyone understand?

Create a test table

Case	Hours worked	Sunday Hours	Hourly Wage	Answer #1	Answer #2	Answer #3
1	40	0	20			

© 2008 Peter Stevens

sierra-charlie.com

1: $40 \times 20 = 800$

Anybody get a different answer?

The simple case is clear.

Case	Hours worked	Sunday Hours	Hourly Wage	Answer #1	Answer #2	Answer #3
1	40	0	20	800		
2	8	8	20			

© 2008 Peter Stevens

sierra-charlie.com

2a: $8 * 40 = 320$

2b: $8 * 20 + 8 * 40 = 480$

Ambiguity in the interface is a source of error

Case	Hours worked	Sunday Hours	Hourly Wage	Answer #1	Answer #2	Answer #3
1	40	0	20	80		
2	8	8	20	320	480	
3	40	8	20			

Problem: Ambiguity in the meaning of the parameters

© 2008 Peter Stevens

sierra-charlie.com

3a: $40 * 20 + 8 * 40 = 1120$ (Sunday rule overrides overtime rule)

3b: $32 * 20 + 8 * 40 = 960$ (40 hours is total hours worked, so no overtime)

3c: $40 * 20 + 8 * 60 = 1280$ (Sunday is overtime and Sunday time)

3d: $8 * 40 + 32 * 20 + 8 * 30 = 1200$ (Sunday is first day of week, so not overtime)

Guessing about the meaning of the spec leads to rework.

Case	Hours worked	Sunday Hours	Hourly Wage	Answer #1	Answer #2	Answer #3
1	40	0	20	80		
2	8	8	20	320	480	
3	40	8	20	1120	960	1280
				1200		

Problem: Ambiguity in the rules

© 2008 Peter Stevens

sierra-charlie.com

3a: $40 * 20 + 8 * 40 = 1120$ (Sunday rule overrides overtime rule)

3b: $32 * 20 + 8 * 40 = 960$ (40 hours is total hours worked, so no overtime)

3c: $40 * 20 + 8 * 60 = 1280$ (Sunday is overtime and Sunday time)

3d: $8 * 40 + 32 * 20 + 8 * 30 = 1200$ (Sunday is first day of week, so not overtime)

This approach is called test driven *design*

Case	Hours worked	Sunday Hours	Hourly Wage	Answer #1	Answer #2	Answer #3
1	40	0	20	80		
2	8	8	20	320	480	
3	40	8	20	1120	960	1280
				1200		

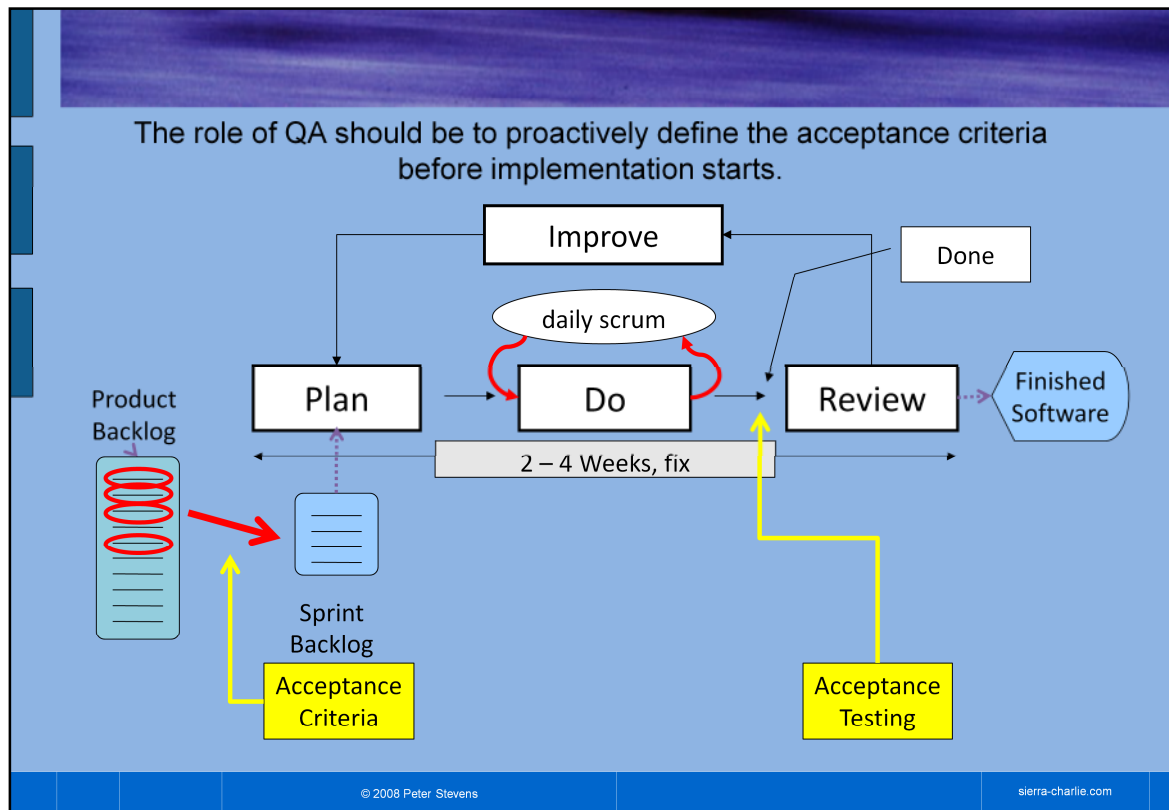
Problem: Ambiguity in the rules

© 2008 Peter Stevens

sierra-charlie.com

Test Driven Design – write tests to specify the product and focus the developers on writing the right thing.

BTW - This result is obvious to anybody who had worked for an hourly wage in North America. What does that say about the problems of offshoring?



Testers are part of the implementation team.

QA prepares the acceptance criteria for each story before it is accepted into a sprint.

QA is transformed from detecting bugs to preventing bugs.

Testing still happens at the end. But it should be a confirmation and one does not expect to find (many) bugs.



And now, I'd like to tell you a few stories.

Scrum Stories (and Scars).

**It can be harder to change
management than the team.**

Buy in from the team is
essential.

You can achieve a cost ceiling or a
required date

Patterns for getting started

Managers are used to talking and thinking for their teams.



Photo courtesy of Laura Ribeiro_@ flickr

© 2008 Peter Stevens

sierra-charlie.com

How do you know that we want to be a self organizing team?



© 2008 Peter Stevens

sierra-charlie.com

You can achieve a cost ceiling or a required date



© 2008 Peter Stevens

sierra-charlie.com

Ingredients:

- Team experienced with agile
- A customer who shared the goal
- A cooperative approach to achieving the goal

Project 1:

- Team working together since 2 years, with Scrum for 3 months
- Very difficult project birth, everything changed between sale and realization
- End Customer was Kanton, with Bund involved, and „interesting“ politics
- Our customer asked for an estimate
- Our Answer: Start in January, finish 6 to 7 sprints later
- We delivered the value on the day promised,
- Scope creep did occur, but this was developed & paid for as a 1.1. release

Project 2

- More or less the same team.
- Initial Estimate „100“
- Customer says „I have 80, can you do it?“ Well, OK, yeah, we think so
- Software was delivered on the day promised
- Extra wishes pushed the cost up to 90, but customer was happy
- End customer had delays from other suppliers – we waited 2 months before we could go live

Getting Started: Introduction Patterns



Photo © - Fotolia

© 2008 Peter Stevens

sierra-charlie.com

Introducing Scrum into your company is a journey.

What are the stops on your journey? The following slides are patterns which have often been applied in other companies



Join a community!



Photo © javarman - fotolia

Lean Agile Scrum Group
swissict.ch/lean
Leanagilescrum.ning.com

Scrum Breakfast
scrum-breakfast.com

Deutschescrum
tinyurl.com/bzo4uy

© 2008 Peter Stevens

sierra-charlie.com

Take a trial run



Photos © Stephen Coburn, - fotolia

© 2008 Peter Stevens

sierra-charlie.com

Treat it as an experiment

Apply some of the practices, e.g.

- daily scrum
- Iterations

Keep the spotlight off

Read a book

<http://tinyurl.com/Scrum-Books>

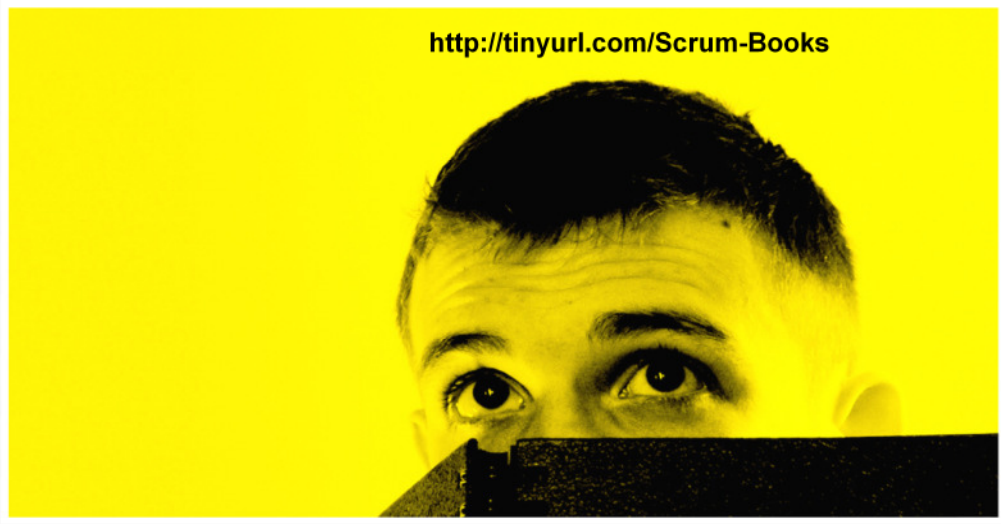
A photograph of a man's face, specifically his eyes and forehead, peeking over the top edge of a dark book. The background is a solid, bright yellow. The photo is framed within a white border, which is itself set against a light green background.

Photo courtesy of bromia@flickr

© 2008 Peter Stevens

sierra-charlie.com

English:

- Agile Project Management with Scrum, by Ken Schwaber
- Agile Estimating and Planning, by Mike Cohn
- User Stories Applied,
by Mike Cohn

German

- Scrum. Produkte zuverlässig und schnell entwickeln
by Boris Gloger
- Scrum - Agiles Projekt-Management erfolgreich einsetzen
by Roman Pichler

Hire a Coach/Take a Scrum Course



<http://tinyurl.com/Scrum-Kurse>

Photo © Wojciech Gajda - Fotolia
© 2008 Peter Stevens
sierra-charlie.com

I hesitate to mention this in a presentation, but every Scrum success story from a large company that I have read or heard has cited coaching and training for their teams as a key success factor.

I offer public Scrum training courses in Switzerland in German and In-House courses in English or French.

Scrum Jumpstart – Getting started for teams, P-O & Scrum Masters. April 6 & 7

Certified Scrum Master – Advanced topics for the Scrum Master,
May 4 & 5 or June 25 & 26

CSM with Scrum Founder Ken Schwaber. June 2 & 3

Practical Product Owner – May 18 & 19

Everything the product owner needs to create great products and
Manage a scrum team

You can also find a directory of Scrum Courses from independent scrum trainers at ScrumAlliance.org.

Save a challenged project



Photo courtesy of [tony.evans@flickr](#)

© 2008 Peter Stevens

[sierra-charlie.com](#)

Usually an external Agile coach (Scrum Master) teaches your team Scrum & “Brings the ship to port”

I don't really recommend this approach (even though this is my bread and butter), but there is nothing like an impending catastrophe to get people's attention and create a willingness to try new things.

Do's & Don'ts

Do's

- Large companies
 - Get a Coach
 - Get Training
 - You
 - Your Team
 - Your Management
 - Get Management Support
- Anybody
 - Respect the sprint contract
 - Respect self organization
 - Retrospective every sprint
 - Short sprints (2-3 weeks)

Don'ts

- Large Companies
 - Try “Stealth” Scrum
- Anybody
 - Refuse to compromise
 - Stop learning
 - Change Scrum
(before you have mastered it)

Questions



Photo © leonardo viti - fotolia

© 2008 Peter Stevens

sierra-charlie.com

Further Reading



Photo © source unknown – please contact me if this image belongs to you

© 2008 Peter Stevens

sierra-charlie.com

Further Reading

- Blogs
 - Peter Stevens
<http://scrum-breakfast.com>
agilesoftwaredevelopment.com
 - Jean-P. Koenig (German)
<http://inside-scrum.blogspot.com>
- Ken Schwaber
 - Agile Project Management with Scrum
 - The Enterprise and Scrum
- Rising & Manns
 - Fearless Change
- Mike Cohn
 - Agile Estimating and Planning
 - User Stories Applied
- Mary & Tom Poppendieck
 - Lean Software Development: An Agile Toolkit
 - Implementing Lean Software Development
- Boris Gloger
 - Scrum - Produkte zuverlässig und schnell entwickeln

Bon Voyage!



Peter Stevens
peter@sierra-charlie.com
+41 44 586 6450

© 2008 Peter Stevens

sierra-charlie.com