

Scrum

Mark C. Paulk, Ph.D.

May 21, 2009
MSE Students

Questions from the Cave

Scrum in general

- **In brief, what is the motivation behind Scrum?**
- **What is the problem in software engineering that Scrum tried to solve?**
- **How will Scrum help with planning and tracking?**
- **What are the fundamental components in Scrum?**

Scrum in practice

- **Will Scrum work well with TSP?**
- **When would you use Scrum over other processes?**
- **Does it work with things besides software?**
- **What are common pitfalls to look out for when adopting Scrum?**
- **How do you know that you are using Scrum correctly?**

The Agile Alliance

Agile Manifesto

“We are uncovering better ways of developing software by doing it and helping others do it.

Through this work we have come to value:

- ***individuals and interactions*** over processes and tools
- ***working software*** over comprehensive documentation
- ***customer collaboration*** over contract negotiation
- ***responding to change*** over following a plan

That is, while there is value on the items on the right, we value the items on the left more.”

<http://www.agilealliance.org>

Principles of Agile Software Development

A. Cockburn, “Learning From Agile Software Development,” Crosstalk: The Journal of Defense Software Engineering, October & November 2002.

- 1) Different projects need different methodology trade-offs.
- 2) A little methodology does a lot of good; after that, weight is costly.
- 3) Larger teams need more communication elements.
- 4) Projects dealing with greater potential damage need more validation elements.
- 5) Formality, process, and documentation are not substitutes for discipline, skill, and understanding.
- 6) Interactive, face-to-face communication is the cheapest and fastest channel for exchanging information.
- 7) Increased communication and feedback reduces the need for intermediate work products.
- 8) Concurrent and serial development exchange development cost for speed and flexibility.
- 9) Efficiency is expendable in non-bottleneck activities.
- 10) Sweet spots speed development.

Scrum

A process for incrementally building software in complex environments.

Backlog – all outstanding work for a product area

Sprints – 30-day increments of work that produce a deliverable

Scrums – daily status check meetings

*K. Schwaber, [Agile Project Management with Scrum, 2004.](http://www.controlchaos.com)
<http://www.controlchaos.com>*

5

Three Scrum Roles

ScrumMaster

- project master?
- coach, facilitator, expediter

Product Owner

- customer point of contact
- “whole team”

Development Team

- small, co-located?
- generalists?

6

Scrum Sprint Rules

Use small interdisciplinary teams

Build clean interface software

Intelligent management required

Solid systems architecture and framework upfront

Prototype all new tools and technology

Develop infrastructure first

Each Sprint results in an executable

Develop, document, and test in parallel

7

Daily Scrum Questions

What did you do since the last Scrum?

What got in your way?

What are you going to do before the next Scrum?

8

Daily Scrum Protocol

Daily, same place and time, 15 minutes

Only three questions

All pigs (committed) must respond

Chickens (involved) can attend, but must be silent

No new backlog can be introduced externally

Backlog can be added internally

9

Other Scrum Practices

Product Backlog, Release Plan

Sprint Planning Meeting (4+4 hr), Sprint Backlog

Backlog/Release Burndown Charts

Sprint Review Meeting (4 hr)

Sprint Retrospective Meeting (3 hr)

10

Why Scrum is Powerful

Focus is on team's work and team's work only

Daily communication of status occurs

Enables low-overhead empirical management

Makes impediments visible

Someone is willing to make decisions and remove impediments real-time

11

Associated Practices

Defining "done"

- design reviews
- code reviews
- unit tested

Practices commonly associated with other agile methods

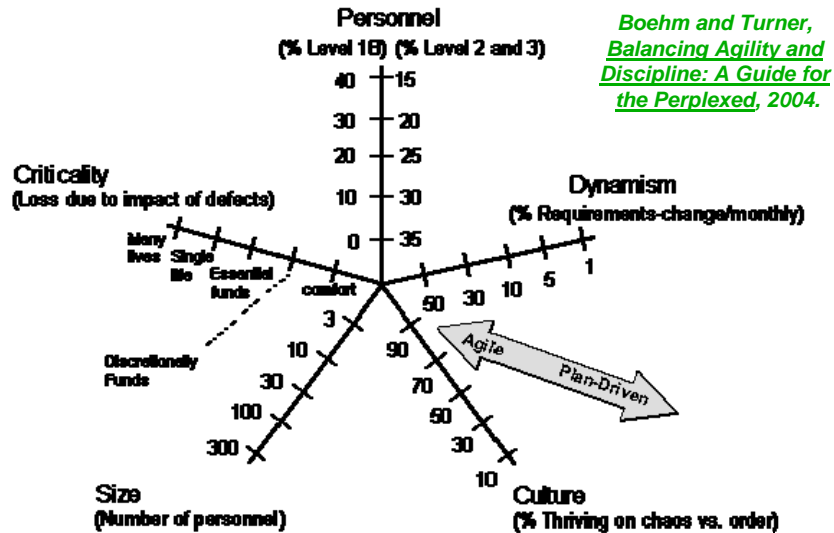
- test-driven development
- pair programming

Good engineering and management practices in general

- top-10 risks list
- customer-supplier relationship

12

Agile vs Plan-Driven Tradeoffs



13

Cultural Misfits (Using the DoD as an Example...)

Regulatory requirements for a level playing field raise challenges for evolutionary and incremental development...

The need by the contracts officer for a requirements specification...











Progress payments defined from a waterfall mentality...

Barriers – regulatory and cultural – to a collaborative customer relationship...

Protests from competitors...

14

Aligning Agility and Discipline

-  Early and continuous delivery
-  Changing requirements, even late in development
-  Work with business people (customers and end users)
-  Motivated individuals with support they need . . . *People CMM*
Face-to-face conversation
-  Working software
-  Sustainable development, constant pace
-  Continuous attention to technical excellence and good design
-  Simplicity
-  Self-organizing teams . . . *CMMI / IPPD*
-  Reflect on how to become more effective, then adjust behavior

15

Questions from the Cave Reprised

Scrum in general

- In brief, what is the motivation behind Scrum? ✓
- What is the problem in software engineering that Scrum tried to solve? ✓
- How will Scrum help with planning and tracking? ✓
- What are the fundamental components in Scrum? ✓

Scrum in practice

- Will Scrum work well with TSP?
- When would you use Scrum over other processes? ✓
- Does it work with things besides software?
- What are common pitfalls to look out for when adopting Scrum?
- How do you know that you are using Scrum correctly?

16

Questions and Answers



17

Contact Information

Dr. Mark C. Paulk
Institute for Software Research
Carnegie Mellon University
407SCR 115
5000 Forbes Avenue
Pittsburgh, PA 15213

mcp@cs.cmu.edu
Mark.Paulk@ieee.org

<http://www.cs.cmu.edu/~mcp/>



18