Agile Project Management with Scrum

Ken Schwaber

2004

CONTENTS

1 BACKDROP: THE SCIENCE OF SCRUM

1.1 EMPIRICAL PROCESS CONTROL

> If the commodity is of such unacceptable quality as to be unusable, the rework is too great to make the price acceptable, or the cost of unacceptably low yields is too high, we have to turn to and accept the higher costs of **empirical process control**. In the long run, making successful products the first time using empirical process control turns out to be much cheaper than reworking unsuccessful products using defined process control

Three legs of empirical process control:

- visibility
  - aspects of the process that affect the outcome must be visible to those controlling the process. Not only must these aspects be visible, but what is visible must also be true (certain functionality is labeled 'done'?). It doesn't matter whether it is visible that this functionality is done if no one can agree what the word "done" means.
- inspection
  - The various aspects of the process must be inspected frequently enough that unacceptable variances in the process can be detected. Processes are changed by the very act of inspection. The other factor in inspection is the inspector, who must possess the skills to assess what he or she is inspecting.
- adaption
  - when process are outside acceptable limits and that the resulting product will be unacceptable, adjustment must be made as quickly as possible to minimize further deviation.

'code review' as an example of an empirical process control.

1.2 COMPLEX SOFTWARE DEVELOPMENT

three most significant dimensions:

- requirements
  - simple: one customer, one developer
  - usually: stakeholders (glossary) with different needs that change, sometimes who start to understand what they want when they are provided with someone else's impression of what they want.
- technology
  - simple tech rerly used in software development

  - usually more than one piece is of need

- people
  - skills, intelligence levels, experience, viewpoints, attitudes, and prejudices, complexity level goes through the roof

usually intersection between requirement complexity and tech complexity define the total complexity of a project.

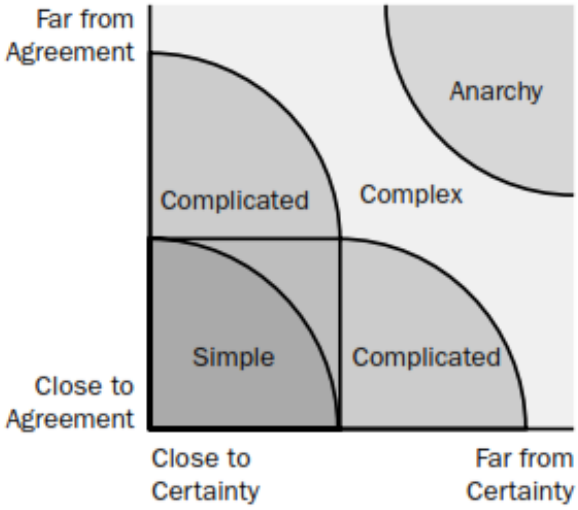

Figure 1.1: complexity assessment graph

The team takes a look at the requirements, considers the available technology, and evaluates its own skills and capabilities. It then collectively determines how to build the functionality.

2 ITERATION

- At the start of an iteration, the team reviews what it must do.
- It then selects what it believes it can turn into an increment of potentially shippable functionality by the end of the iteration
- team is then left alone to make its best effort for the rest of the iteration
- At the end of the iteration, the team presents so that the stakeholders can inspect and timely adaptations ca be made

3 SCRUM ROLES

Scrum implements this iterative, incremental skeleton through three roles.
All management responsibilities in a project are divided among these three roles.

1. Product Owner
2. Scrum Master
3. Team

3.1 PRODUCT OWNER

- achieves initial and ongoing funding for the project
- responsible for representing the interests of everyone with a stake in the project and its resulting system.
- responsible for using the Product Backlog (glossary) so that valuable functionality is produced first.
- responsible to those funding the project for delivering the vision in a manner that maximizes their ROI, he formulates a plan for doing so that includes a Product Backlog

3.2 SCRUM MASTER

- responsible for the Scrum process, for teaching Scrum to everyone involved in the project
- implementing Scrum so that it fits within an organization's culture and still delivers desired benefits
- ensuring that everyone follows Scrum rules and practices

The people who fill these roles are those who have committed to the project.
After the Sprint review and prior to the next Sprint planning meeting, the ScrumMaster holds a Sprint retrospective meeting with the Team

3.3 TEAM

- responsible for developing functionality
- self-managing, self-organizing, and cross-functional
- responsible for figuring out how to turn Product Backlog into an increment of functionality within an iteration
- collectively responsible for the success of each iteration and of the project as a whole

Team tells the Product Owner how much of what is desired it believes it can turn into functionality over the next Sprint.

3.4 TWO GROUPS

important distinction in scrum

- 'chickens': are intereseted in the project, but not on the hook
- 'pigs': are on the hook

It should always be clear who is on the hook and who is just a kibitzer.
Who is responsible for the ROI, and who has a stake in the ROI but isn't accountable?
The rules of Scrum distinguish between the chickens and the pigs to increase productivity, create momentum, and put an end to floundering.

4 SCRUM FLOW

4.1 SPRINTS

All work is done in Sprints, each Sprint is an iteration of 30 consecutive calendar days.
Each Sprint is initiated with a Sprint planning meeting, where the Product Owner and Team get together to collaborate about what will be done for the next Sprint.
Selecting from the highest priority Product Backlog, the Product Owner tells the Team what is desired.

4.1.1 SPRINT PLANNING

Sprint meeting should not last longer than 8 hours.
The Sprint planning meeting has two parts:

- first four hours
  - are spent with the Product Owner presenting the highest priority Product Backlog to the Team
  - The Team questions him or her about the content, purpose, meaning, and intentions of the Product Backlog
  - When the Team knows enough, selects as much Product Backlog as it believes it can turn into a completed increment of potentially shippable product functionality by the end of the Sprint.
- second four hours
  - the Team plans out the Sprint
  - The tasks that compose this plan are placed in a Sprint Backlog
  - the tasks in the Sprint Backlog emerge as the Sprint evolves

directly after the meeting, the sprint has started, 30 day target is tried to be met.

4.1.2 DAILY SCRUM

Every day, the team gets together for a 15-minute meeting called a Daily Scrum.
each Team member answers three questions:

- What have you done since last Daily Scrum?
- What do you plan on doing between now and next Daily Scrum?
- What impediments stand in the way of you meeting your commitments to this Sprint and this project?

purpose of the meeting is to

- synchronize each team member daily
- to schedule any meetings that the Team needs to forward its progress

4.1.3 REVIEW MEETING

At the end of the Sprint a Sprint review meeting is held, it is an informal meeting.
four-hour meeting at which the Team presents what was developed during the Sprint to the Product Owner and any other stakeholders
purpose of the meeting is to

- functionality is presented is intended to bring people together
- collaboratively determined what the Team should do next

4.1.4 SPRINT RETROSPECTIVE

After the Sprint review and prior to the next Sprint planning meeting, the ScrumMaster holds a Sprint retrospective meeting with the Team. three-hour meeting: revise the Teams development process to make it more effective and enjoyable for the next Sprint.

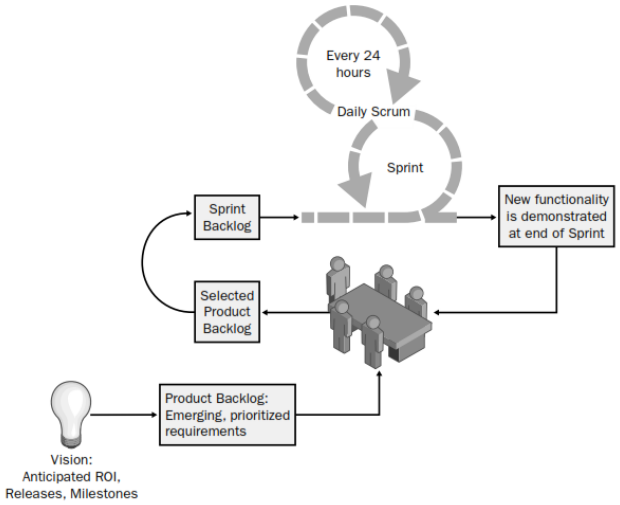

Figure 4.1: scrum process overview

- The Product Backlog is a list of functional and nonfunctional requirements
- it will deliver the vision of maximizing the ROI (glossary)
- is prioritized so that the items most likely to generate value are top priority and is divided into proposed releases
- Changes in the Product Backlog reflect changing business requirements and how quickly or slowly the Team can transform Product Backlog into functionality.

Scrum hangs all of its practices on an iterative, process skeleton. The heart of Scrum lies in the iteration.

Scrum addresses the complexity of software development projects by implementing the inspection, adaptation, and visibility requirements of empirical process control with a set of simple practices and rules, which are described in the following sections.

The people who fill these roles are those who have committed to the project ('pigs'). Others might be interested in the project, but they aren't on he hook. Scrum makes a clear distinction between these two groups and ensures that those who are responsible for the project have the authority to do what s necessary for its success and that those who aren't responsible can't interfere unnecesarily ('chickens').

All work is done in Sprints

defined process control	Laying out a process that repeatably will produce acceptable quality output
Daily Scrum	team gets together for a 15-minute meeting
empirical process control	When defined process control cannot be achieved because of the complexity of th
product Backlog	The list of functional and nonfunctionalrequirements
ROI	return on investment
stakeholders	those with an interest in the software and how it works