



SOFTWARE ENGINEERING LECTURE 2

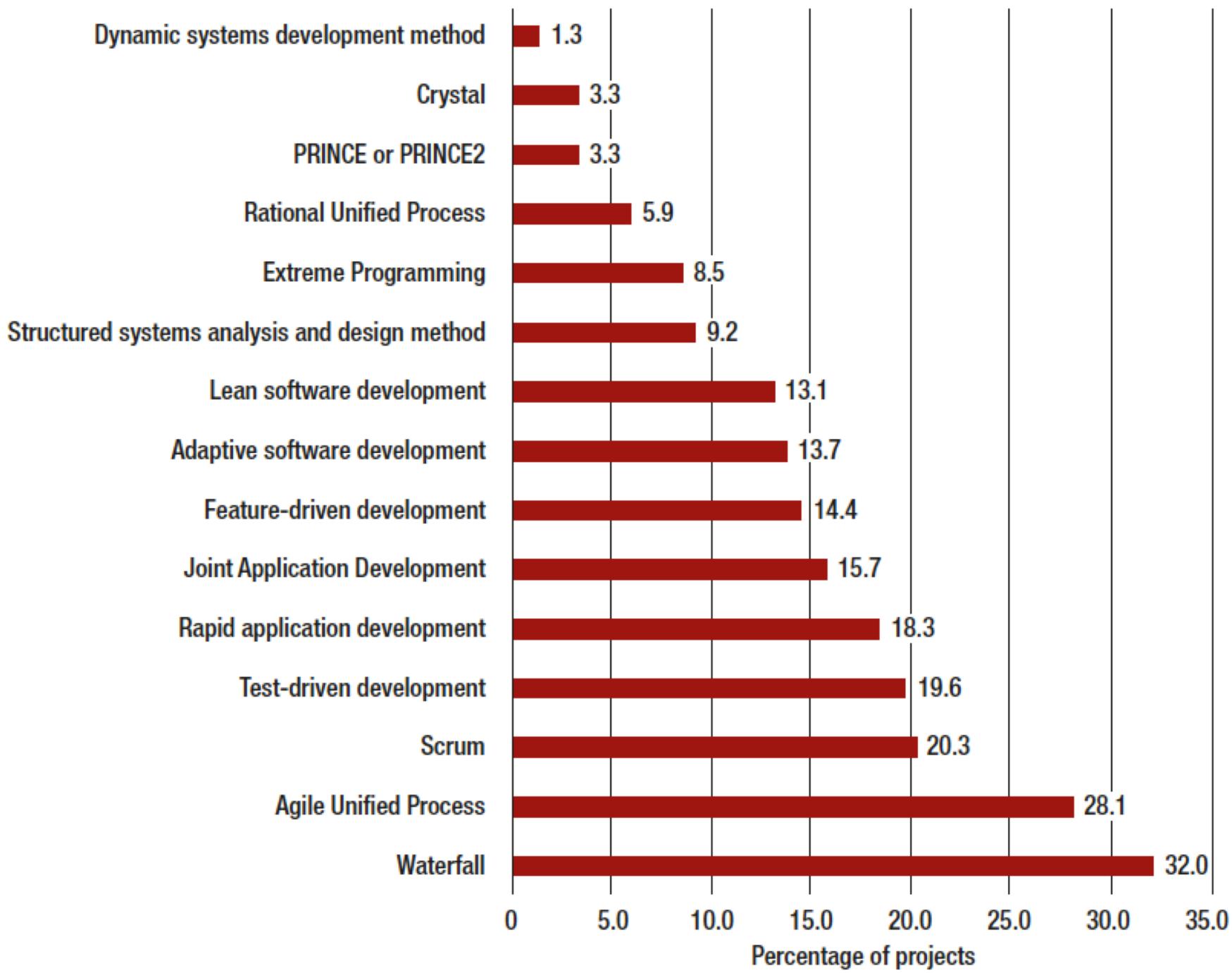
CARSTEN RUSENG JAKOBSEN

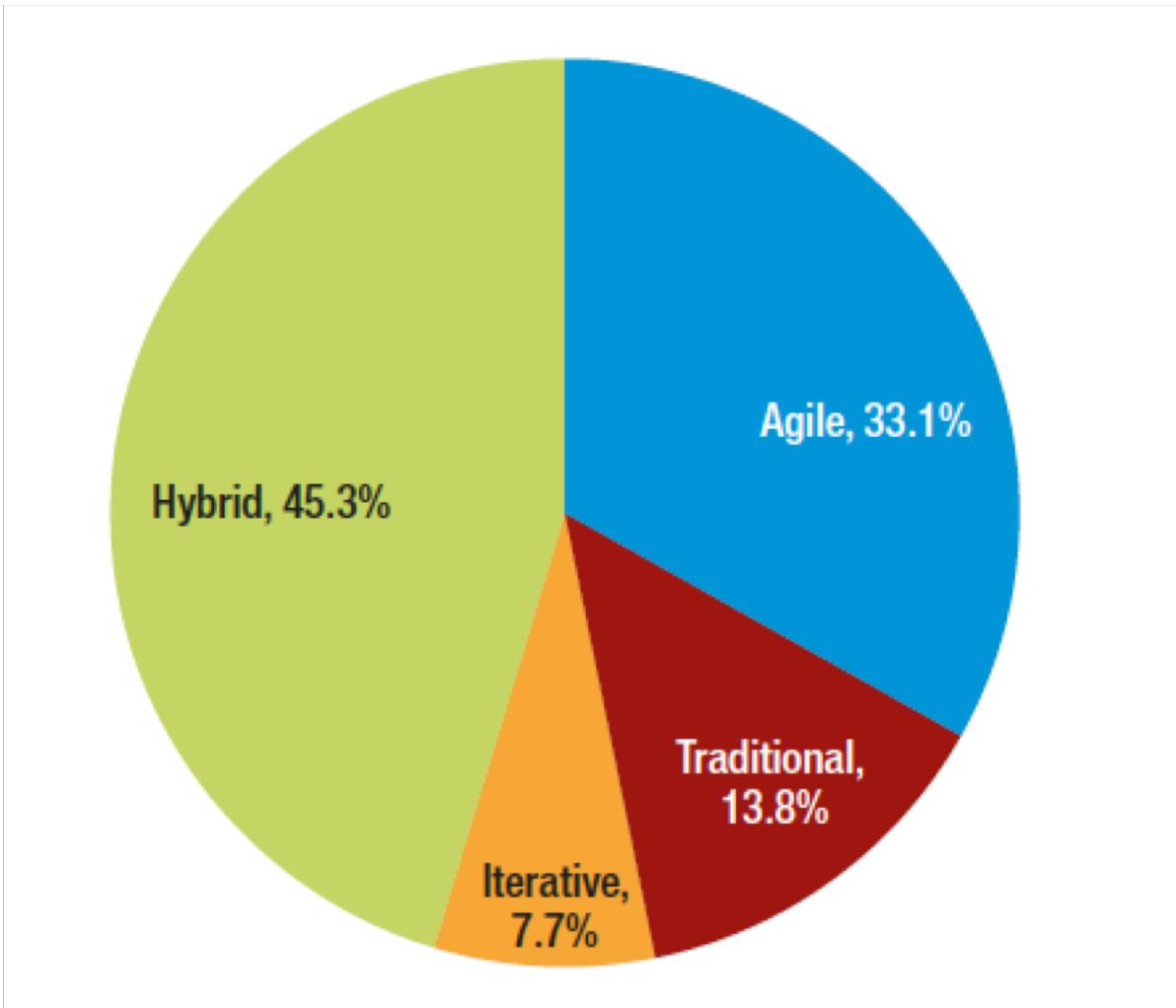


AALBORG UNIVERSITY
DENMARK

Agile

- Agile Manifesto
- XP
- Scrum





Most projects use a mix of predictive (waterfall) and agile methods

XP techniques

Concept	Description
Incremental planning	Requirements are recorded on story cards and the stories to be included in a release are determined by the time available and their relative priority. The developers break these stories into development 'Tasks'.
Small releases	The minimal useful set of functionality that provides business value is developed first. Releases of the system are frequent and incrementally add functionality to the first release.
Simple design	Enough design is carried out to meet the current requirements and no more.
Test-first development	An automated unit test framework is used to write tests for a new piece of functionality before that functionality itself is implemented.
Refactoring	All developers are expected to refactor the code continuously as soon as possible code improvements are found.

XP techniques, contd.

Pair programming	Developers work in pairs, checking each other's work and providing the support to always do a good job.
Collective ownership	The pairs of developers work on all areas of the system, so that no islands of expertise develop and all the developers take responsibility for all of the code. Anyone can change anything.
Continuous integration	As soon as the work on a task is complete, it is integrated into the whole system. After any such integration, all the unit tests in the system must pass.
Sustainable pace	40 h work week. Large amounts of overtime are not considered acceptable as the net effect is often to reduce code quality and medium term productivity
On-site customer	A representative of the end-user of the system (the customer) should be available full time for the use of the XP team. The customer is a member of the development team and is responsible for bringing system requirements to the team for implementation.

XP: On-site customer

- A real customer sits with the team
- Answers the developers' questions
- Writes user stories and functional tests
- Provides value to the project by being ready available as a knowledgeable source on everything to do with the customers and users and their business

XP: Pair Programming



EXTREME PROGRAMMING

THE TWO OF YOU WILL
BE A CODE-WRITING
TEAM.

www.dilbert.com

STUDIES PROVE THAT
TWO PROGRAMMERS
ON ONE COMPUTER
IS THE MOST PRODUC-
TIVE ARRANGEMENT.

© 2002 United Feature Syndicate, Inc.

SOMETIMES I CAN
WHISTLE THROUGH
BOTH NOSTRILS. I'VE
SAVED A FORTUNE
IN HARMONICAS.

XP - User story template

User Role (*Who?*)

As a <type of user>, I can <immediate goal> so that <reason>.

Desired Function (*What?*)

End Result (*Why?*)

*Who, What, Why...
what's **not** here?*

[DRAMATIC SHIFT!!!] By utilizing user stories (and in particular this template), we're saying that we're going to work collaboratively with the customer or business partner to discover what they want (in the course of analysis) and deliver it during the sprint. The past way was for marketing (or customer) to have an iron-clad set of tightly defined requirements and we would chunk those up and meet them!!

- **As a user** I want to be able to set the alarm on my cell phone so I can get up in the morning.
- **As a snoozer** I want to be able to activate ‘snooze’ when the alarm goes off, so I can sleep 10 minutes more.
- **As a user** I want to set the alarm so I can get up at the same time every morning.

Courtesy Gertrud Bjørnvig

XP - What makes a **good** story?

Ron Jeffries' 3 Cs

Card

Conversation

Confirmation

Backside of user story has accepttest criteria

Bill Wake's **INVEST**

Independent

Negotiable

Valuable

Estimable

Small

Testable

XP: Story-based planning

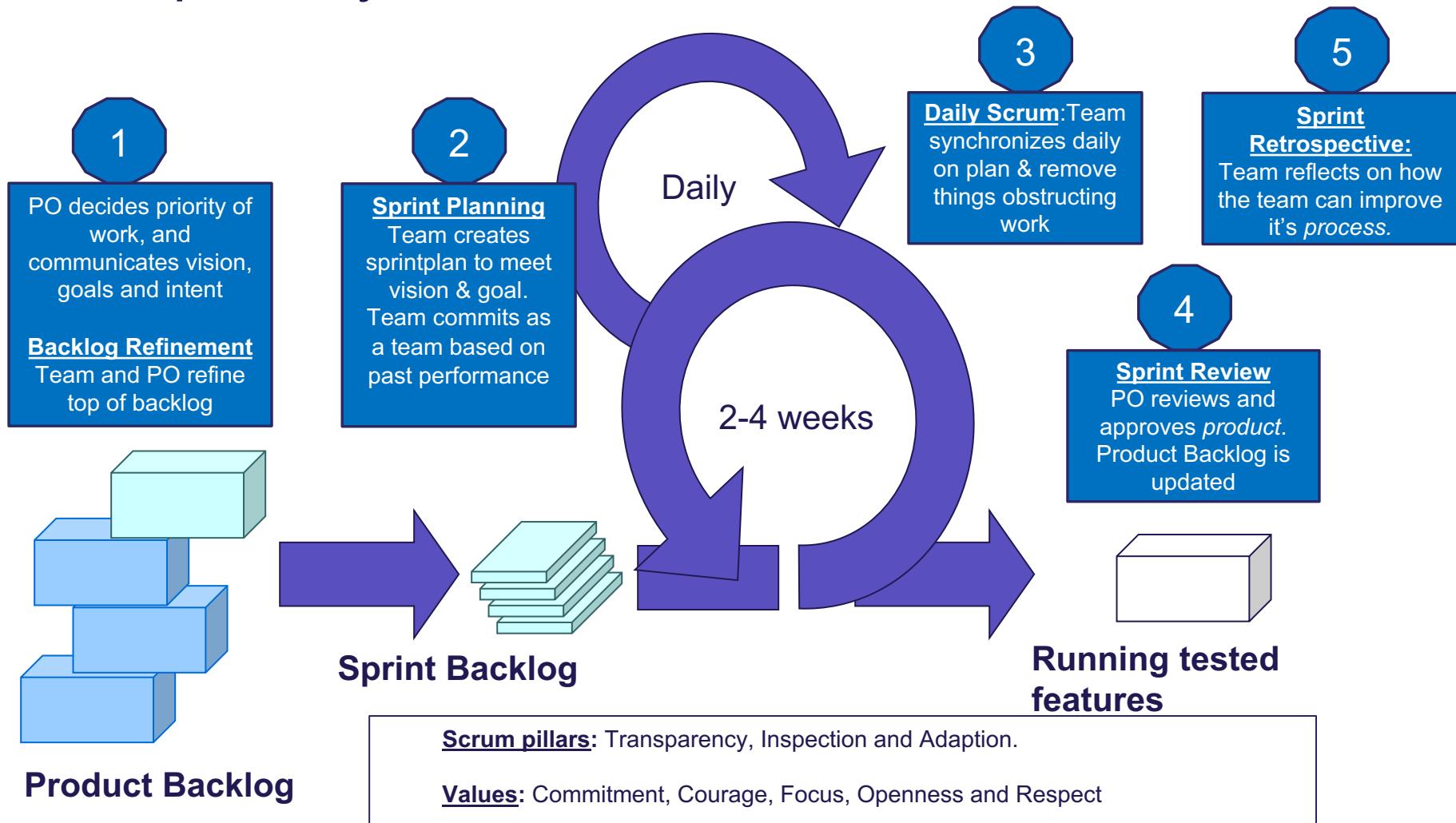
- The planning game is based on user stories
- The project team read and discuss the stories and rank them in order of the amount of time they think it will take to implement the story
- Stories are assigned ‘effort points’ (also called story points) reflecting their size and difficulty of implementation
- The number of effort/story points implemented per day is measured giving an estimate of the team’s ‘velocity’
- This allows the total effort required to implement the next release to be estimated

Scrum core concepts

It's all about respect, commitment, empowerment and being responsible

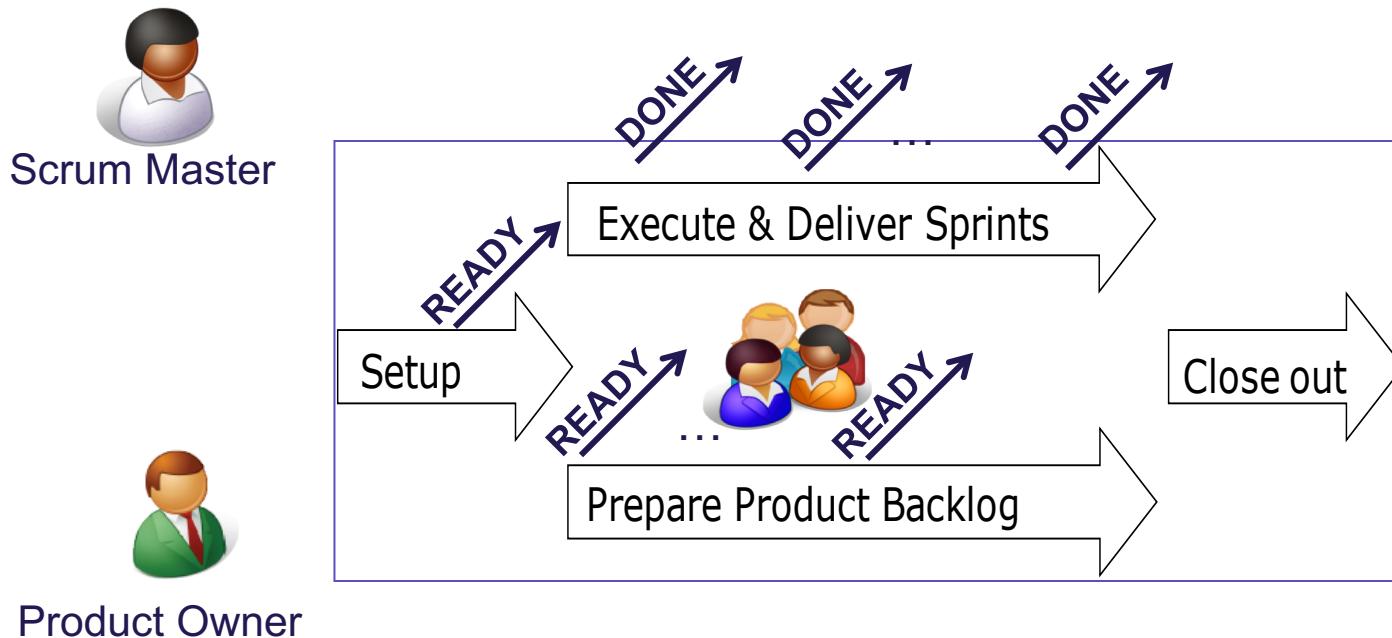
- Separates two concerns
 - In what sequence will work be done (PO)
 - How much work will we commit to (Team)
- Culture of open honest communication
 - Team player over individual players
- Inspect and adapt
 - More focus on goal/results than estimation precision or original plan

Scrum – Inspect & adapt based on transparency



Scrum has two concurrent tracks

Success depends on a balance in progress of the two tracks



When READY is well implemented a positive spiral emerge
When READY is ill implemented a negative spiral is started

Scrum Roles

- **Decision Maker: Order and content of Product Backlog**



Product Owner

- Accepts or rejects work results
- Responsible for *vision and profitability* of the product
- Defines the features of the product, decides on release date and content - *release planning*.



Scrum Master

- **Facilitator of Scrum Process**

- Ensures that the *Scrum process* is followed.
- Enables close cooperation across all roles and functions
- Removes barriers
- Shields the team from external interferences
- *NOT* a project manager

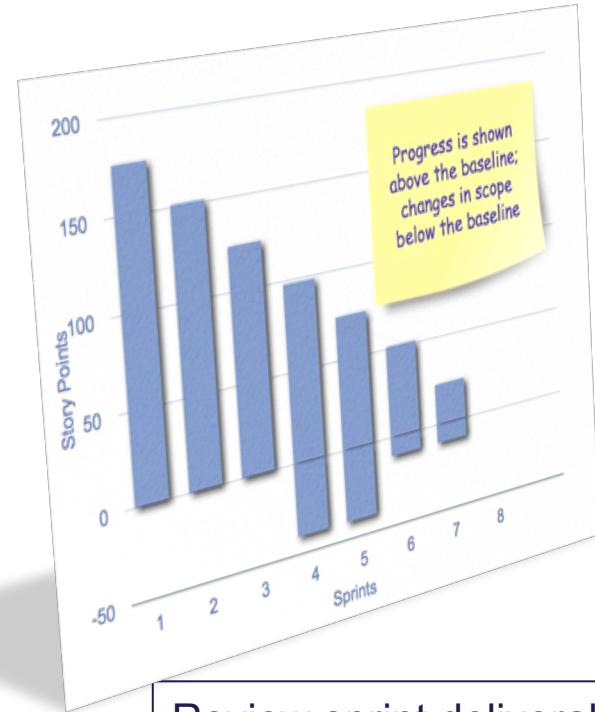


Team

- **Professionals who owns their own sprint plan**

- Cross-functional, seven plus/minus two members
- Selects the sprint goal and specifies work results
- Has the right to do everything within the boundaries of the project guidelines to reach the iteration goal
- Organizes itself and its work
- Demos work results to the Product Owner

Scrum Product Owner activities



Prioritize Product Backlog

Review sprint deliverable

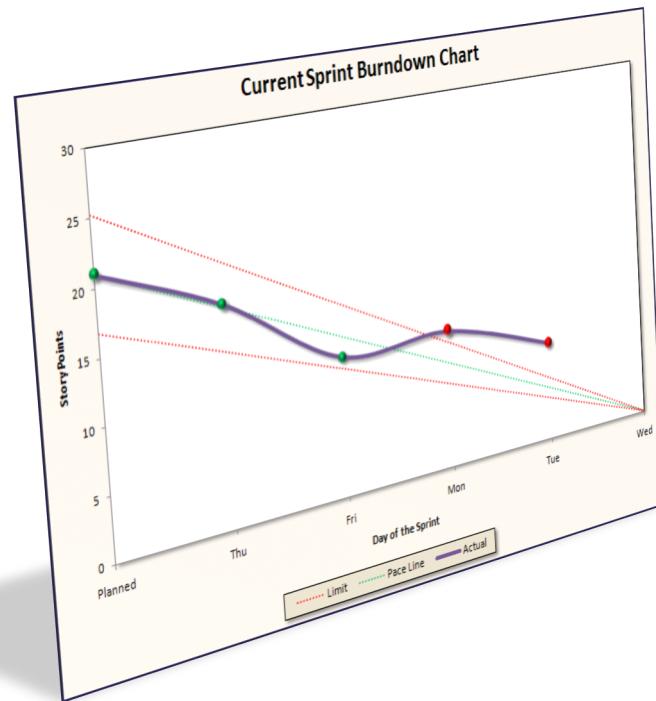
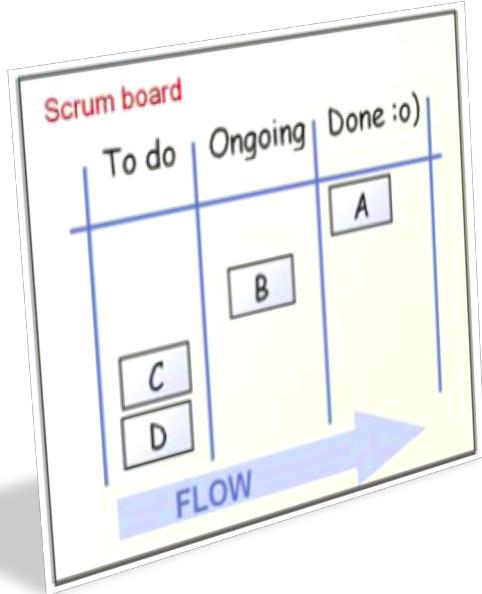
Prepare Product Backlog for next sprint(s)

Provide clarification or decisions to work in sprint



Refinement meetings

Scrum Master activities

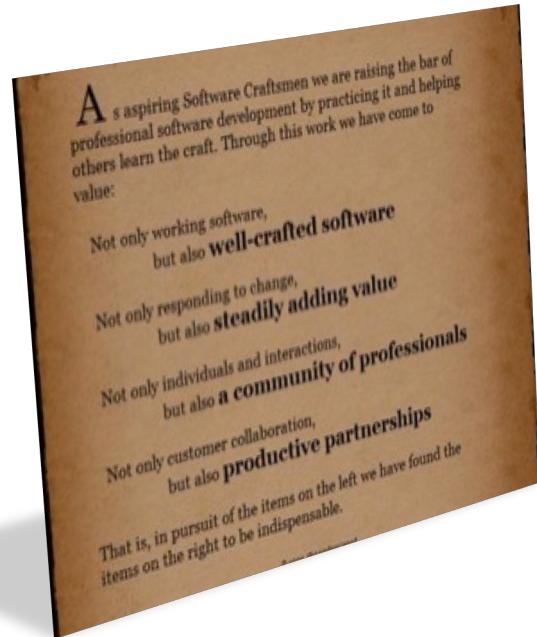


Facilitate Scrum meetings and coach development team and Product Owner



Remove impediments the team were unable to resolve
Protect team – help all to follow scrum process

Scrum Team activities



Sprint Planning

Sprint Review

Sprint Retro.

10% of team effort

Support Product Owner in clarification of work for future sprints

Deliver to sprint commitment

90% of team effort

Scrum

Concept	Definition
Development team	A self-organizing group of software developers, which should be no more than 7 people. They are responsible for developing the software and other essential project documents.
Potentially shippable product increment	The software increment that is delivered from a sprint. The idea is that this should be 'potentially shippable' which means that it is in a finished state and no further work, such as testing, is needed to incorporate it into the final product. In practice, this is not always achievable.
Product backlog	This is a list of 'to do' items which the Scrum team must tackle. They may be feature definitions for the software, software requirements, user stories or descriptions of supplementary tasks that are needed, such as architecture definition or user documentation.
Product owner	An individual (or possibly a small group) whose job is to identify product features or requirements, prioritize these for development and continuously review the product backlog to ensure that the project continues to meet critical business needs. The Product Owner can be a customer but might also be a product manager in a software company or other stakeholder representative.

Scrum, contd.

Concepts	Definition
Scrum	A daily meeting of the Scrum team that reviews progress and prioritizes work to be done that day. Ideally, this should be a short face-to-face meeting that includes the whole team.
Scrum Master	The Scrum Master is responsible for ensuring that the Scrum process is followed and guides the team in the effective use of Scrum. He or she is responsible for interfacing with the rest of the company and for ensuring that the Scrum team is not diverted by outside interference. The Scrum developers are adamant that the Scrum Master should not be thought of as a project manager. Others, however, may not always find it easy to see the difference.
Sprint	A development iteration. Sprints are usually 2-4 weeks long.
Velocity	An estimate of how much product backlog effort that a team can cover in a single sprint. Understanding a team's velocity helps them estimate what can be covered in a sprint and provides a basis for measuring improving performance.

Planning Poker Estimation – Lego example









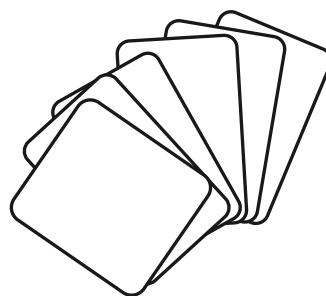




Scrum - Planning Poker

Getting started

1. Each person in your team should have a set of cards with the numbers “1, 2, 3, 5, 8, 13”
2. Each team member holds a set up so only they can see them
3. Estimates are made by the people doing the work, the PO will be present to answer questions and clarify product needs.
4. Estimates are done for each story individually
5. All team members show their selected card at the same time
6. If difference is more than 2 numbers in the series, the person with lowest and highest number present their reasons
7. Entire teams replay and continue until at most 1 number difference, e.g. lowest 5 highest 8



of pieces

1

2

3

5

8

13

21

34

55



Time

1

2

3

5

8

13

21

34

55



Scrum – Other Techniques

1. Daily Scrum
 1. What did you do Yesterday?
 2. What will you do today?
 3. Any impediments making it harder to reach the sprint goal
2. Sprint Review (Product focus)
 1. Only work that meets Done-criteria is presented (no exceptions)
3. Sprint Retrospective (Proces focus)
 1. Tune in (Say a color for how well our proces was in last sprint)
 2. Timeline (place positit notes with events (+/-) on timeline)
 3. Brainstorm (What to Keep, Stop, Try)
4. Many XP-techniques fit well into Scrum