

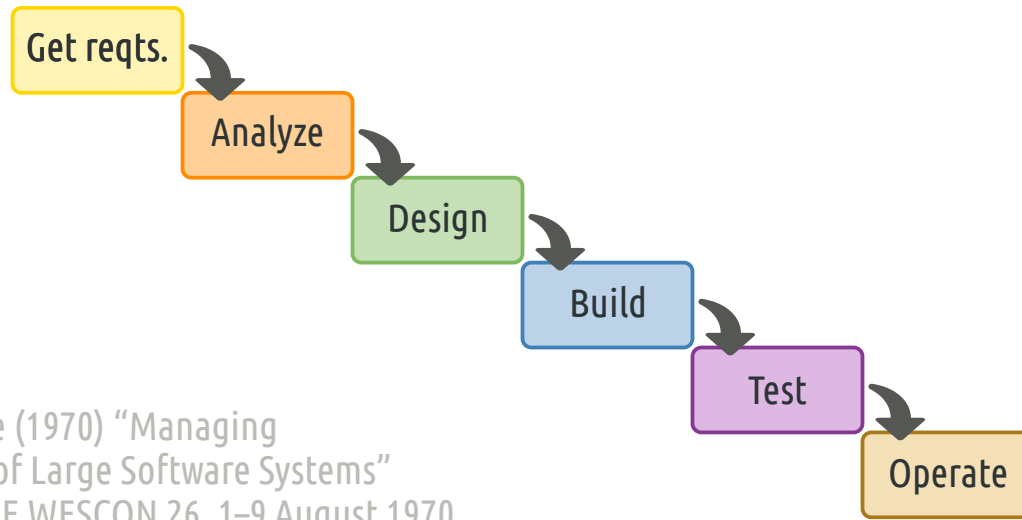


openSE Agility and **SCRUM**

Part 6

What is **SCRUM** ?

- ➔ An alternative to the traditional project management “**waterfall model**” theorized in the 1970's*



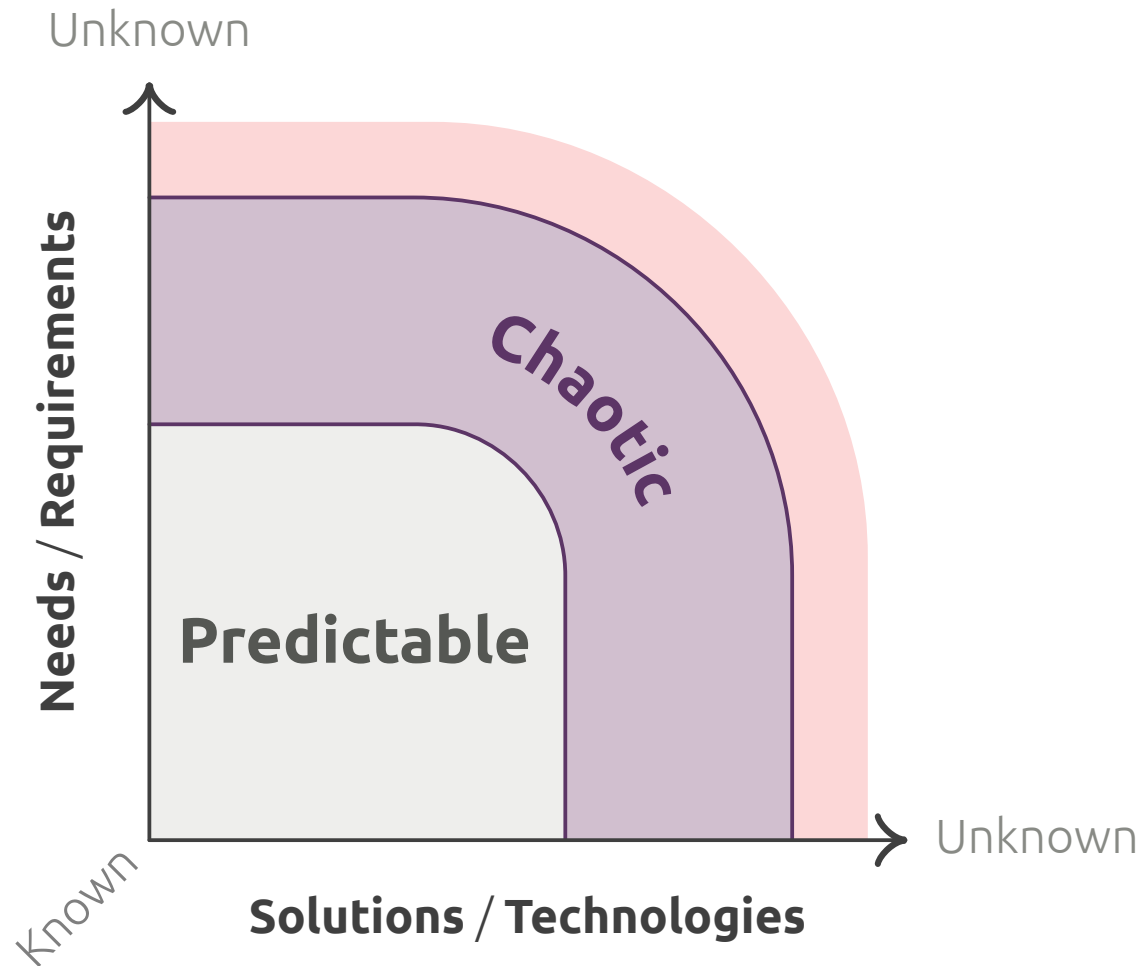
■ *Winston W. Royce (1970) “Managing the Development of Large Software Systems”
Proceedings of IEEE WESCON 26, 1–9 August 1970
(https://en.wikipedia.org/wiki/Winston_W._Royce)



- ➔ **SCRUM** : 
many fixed-duration iterations referred to as **sprints**



What is **SCRUM** ?



What is **SCRUM** ?

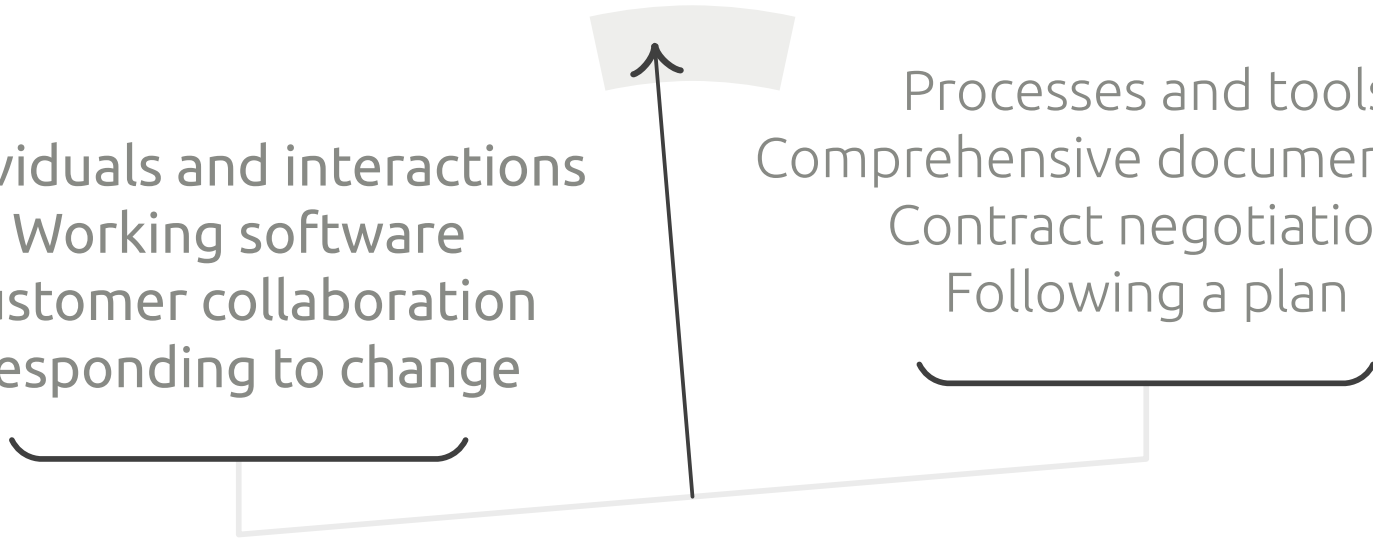
Manifesto for Agile Software Development*



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
Working software
Customer collaboration
Responding to change

Processes and tools
Comprehensive documentation
Contract negotiation
Following a plan



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



What is **SCRUM** ?

Theoretical background

➔ Grounded on the empirical process control theory ➔ **Empiricism***

- 1 Knowledge comes only or primarily from sensory experience
- 2 Decisions shall be made based on this knowledge

➔ Relies on **three pillars**, namely:

Transparency



e.g. sharing
a common
language

Inspection



e.g. frequent
reviews

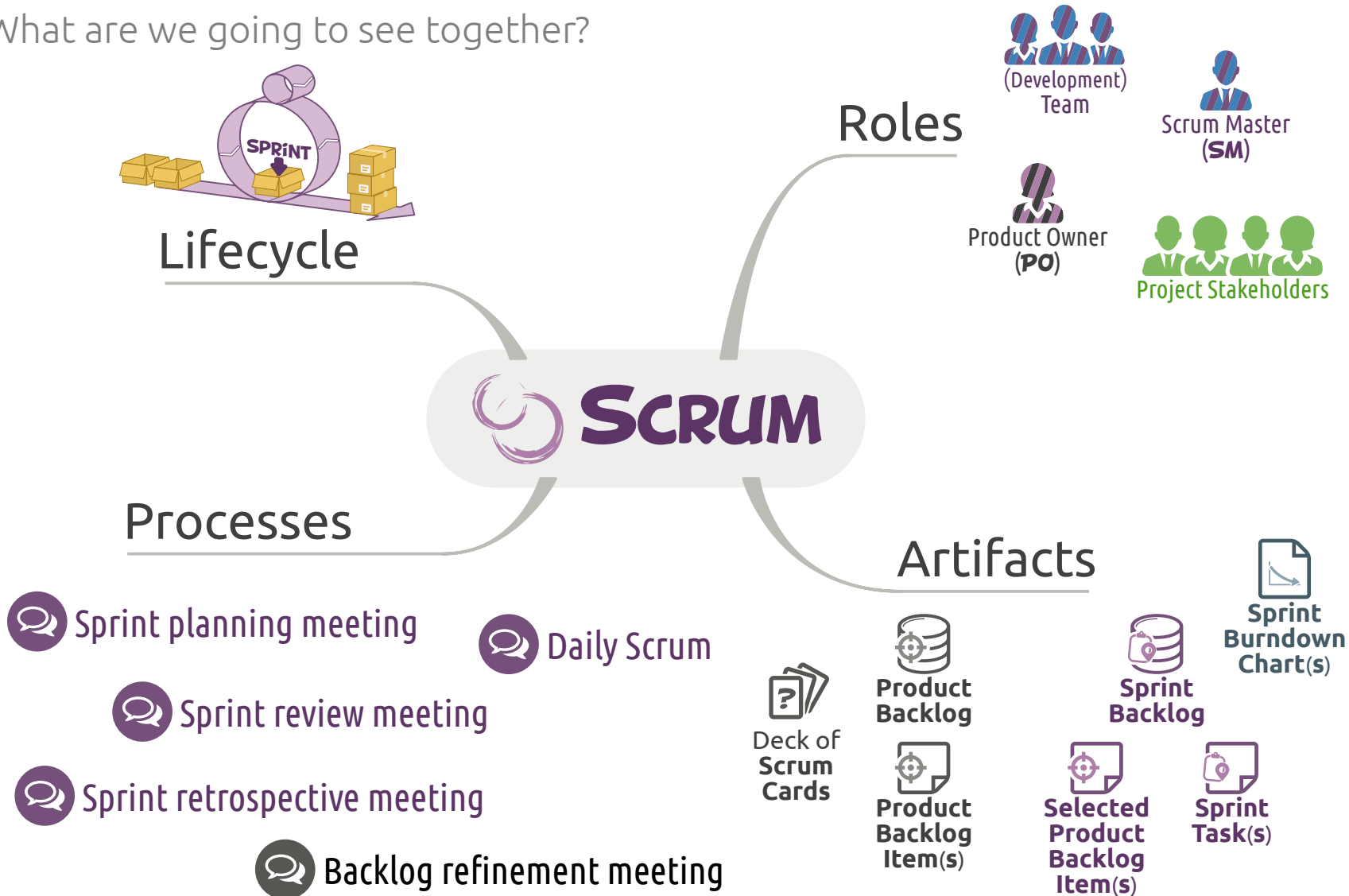
Adaptation



e.g. "along-the-way"
strategy

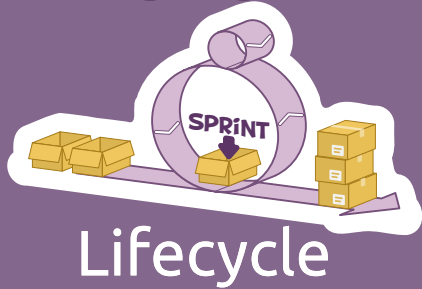
Project Management

What are we going to see together?

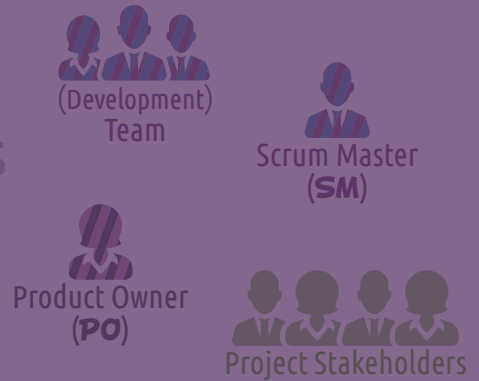


Project Management

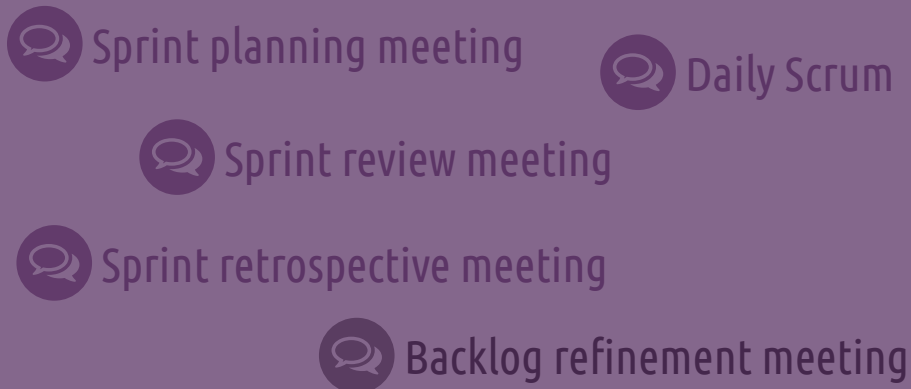
What are we going to see together?



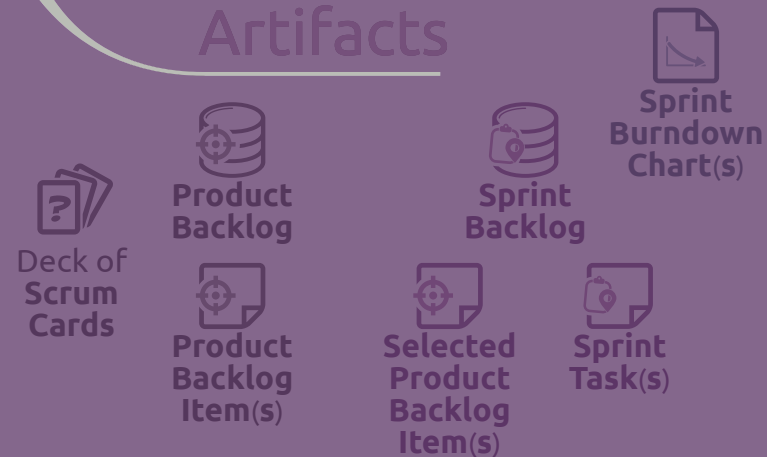
Roles



Processes

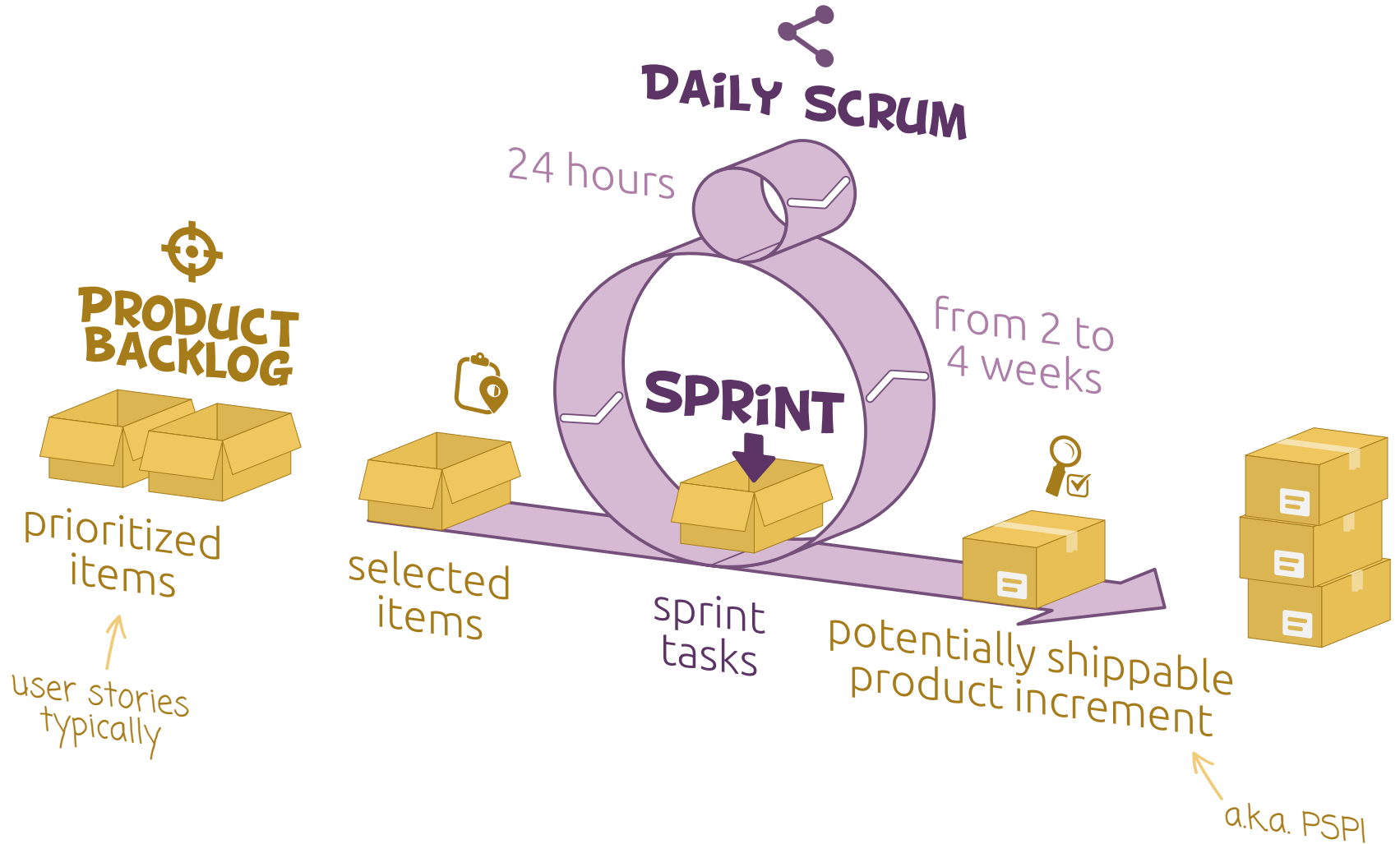


Artifacts



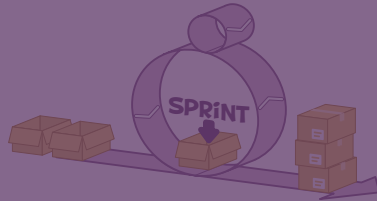
What is **SCRUM** ?

Its lifecycle



Project Management

What are we going to see together?



Lifecycle

Roles



(Development)
Team



Scrum Master
(SM)



Product Owner
(PO)



Project Stakeholders



SCRUM

Processes



Sprint planning meeting



Daily Scrum



Sprint review meeting



Sprint retrospective meeting



Backlog refinement meeting



Sprint
Burndown
Chart(s)



Deck of
Scrum
Cards



Product
Backlog



Sprint
Backlog



Product
Backlog
Item(s)



Selected
Product
Backlog
Item(s)



Sprint
Task(s)

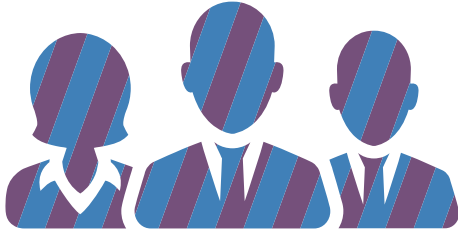
Roles



Product Owner (PO)

- ➔ Is necessarily a **single individual**
- ➔ Is responsible for the “**return on investment**” (ROI), so has authority on **prioritization** of product backlog items
- ➔ Conveys all **requirements** from stakeholders
- ➔ Is the **final arbiter** w.r.t. requirement questions, however, doesn't necessarily knows all the requirements!
- ➔ Owns the **vision** w.r.t. the project deliverable
- ➔ Makes the **business decisions**
- ➔ Is more “**what-focussed**” than “how-focussed”

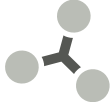
Roles



(Development) Team

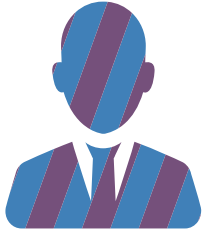
- ➔ Is a **cross-functional team** of 4 to 9 people
- ➔ Is focussed on delivering “**potentially shippable product increment**” at every sprint
- ➔ Is organized as a **collaborating team**, i.e. is self-organized, promotes natural leadership



collaboration (≠  co-operation)

- ➔ Works in a **team room**, and uses intensively **information sharing tools**

Roles

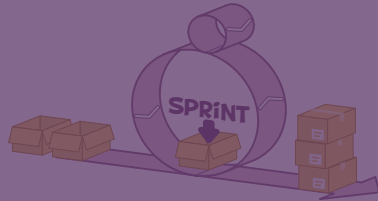


Scrum Master (SM)

- Is also necessarily a **single individual**
- Has **no management authority**, and for this reason, cannot be a line manager nor a project manager
- Is a **facilitator** who protects the development team from distractions and interruptions
- Can facilitate **one team** at a time
- Teaches project participants **how to use Scrum**
- Promotes **improved engineering practices**
- Convenes meetings and enforces ***timeboxes*** (timekeeper in meetings)

Project Management

What are we going to see together?



Lifecycle

Roles



(Development)
Team



Scrum Master
(SM)



Product Owner
(PO)



Project Stakeholders



SCRUM

Processes



Sprint planning meeting



Daily Scrum



Sprint review meeting



Sprint retrospective meeting



Backlog refinement meeting

Artifacts



Deck of
Scrum
Cards



Product
Backlog



Product
Backlog
Item(s)



Sprint
Backlog



Selected
Product
Backlog
Item(s)



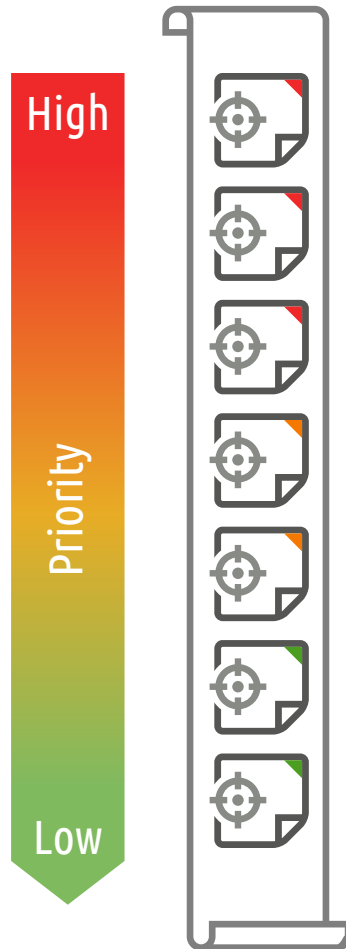
Sprint
Task(s)



Sprint
Burndown
Chart(s)

Artifacts

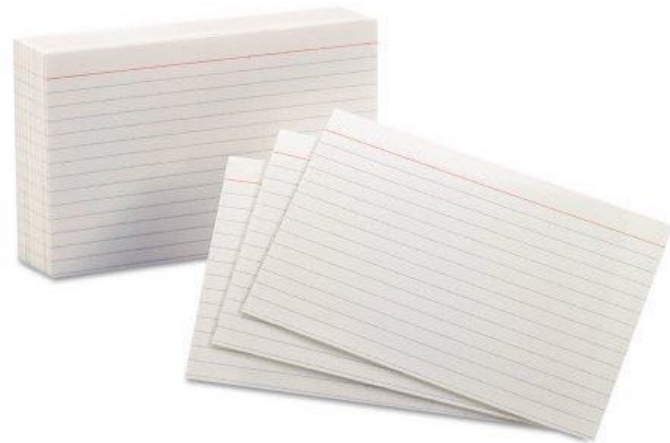
Product Backlog



- ➔ Is a prioritized and dynamic list of **product backlog items**; typically a list of **user stories**
- ➔ Lists all **features, functions, requirements, enhancements, and fixes** related to the product
- ⚠ Is not a list of activities or of tasks!
- ➔ Is owned, ordered (prioritized) and maintained by the **product owner**
- ➔ Is populated by **anyone** (PO, SM, team members, and stakeholders)
- ➔ Evolves as the project progresses (items can be added, removed, modified, morphed, etc.)

User Stories

- ➔ Are informal, short and high level descriptions of **requirements** or **features**, containing just enough information so that teams can produce a reasonable estimate of the effort required to implement each of them
- ➔ Facilitates **sensemaking** – i.e. the gathering of meaningful collective experience – and **communication** among stakeholders
- ➔ Help shift the focus from writing about the requirements to talking about them ➔ concept of “**C**onversation”
- ➔ Are written/owned by the product owner, although they can be initiated by everyone
- ➔ Practically, are traced by means of **C**ards (index cards or sticky notes)



User Stories (continued)

a.k.a. labels or titles

- ➔ Are formulated by means of **user story statements**:

As a **⟨role⟩**, I want **⟨something⟩** so that **⟨benefit⟩**

↑
type of user

↑
goal, capability

↑
reason, rationale

- ➔ Are prioritized, i.e. ordered; therefore feature a **priority index**, e.g.:
 - ⇒ Three-level scale such as: high, medium, low priorities
 - ⇒ MoSCoW scale: must, should, could, won't
 - ⇒ 1-10 scale
- ➔ Are estimated, typically in **story points**, that is a kind of relative scale
- ➔ Feature **conditions of satisfaction** → **Confirmations** that ensure that the objectives of the story have been reached
- ➔ Can either be **not done** or **done** (not in between, e.g. partially done)
- ➔ Can be grouped in **epics** and **themes**

Epics

- ➔ Are large bunches of work containing **several user stories**
- ➔ Are **too big** to be implemented in a single sprint; therefore, they need to be disaggregated into smaller/manageable user stories at some point
- ➔ Are necessarily **lower-priority** user stories

Themes

- ➔ Are grouping of user stories and epics adressing a same family of features

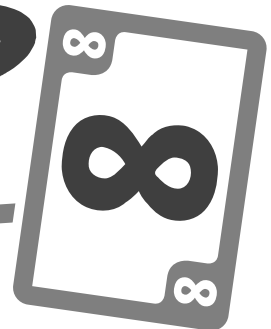
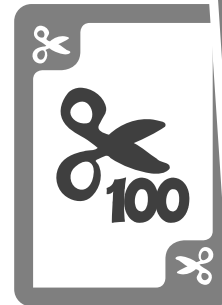
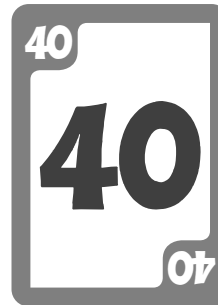
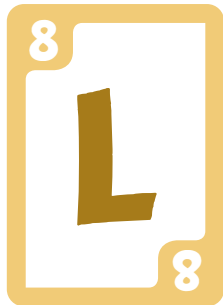
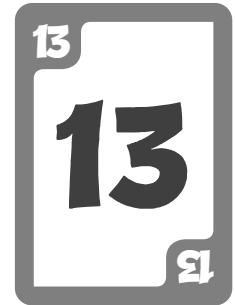
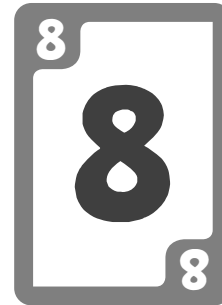
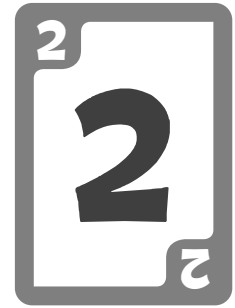
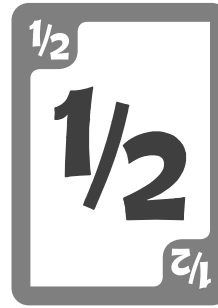
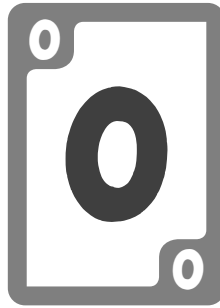
Scrum Estimation Cards

← a.k.a. Planning Poker® or Scrum Poker or Scrum Cards

Fibonacci
scale cards

T-shirt size
scale cards

story
points



Story Point

Definition

- ➔ A story point is a **measure of magnitude**
It's a way to understand the **relative level of effort** of a specific user story as compared to other user stories of the product backlog
- ➔ Story points enable effort to be estimated without trying to estimate how long it will take: **story points $\neq$ ideal person·hours or person·days**
- i To derive an estimate for the **duration** of a project or a sprint: divide the story points for the user stories by the velocity of the team, given by the number of story points achieved over the last sprint(s):

$$\text{duration}_i = \frac{\sum_i \text{story points}}{\text{team velocity}}$$

Ideal (person·)hour — An hour of work where the project participant solely focuses on the task at hand without any interruptions like phone calls, electronic mails, or chat messages. A general agreement sets an average default capacity of six ideal hours per day (for an eight hour working day) for every team member. That is a 75% effectiveness.

Story Point

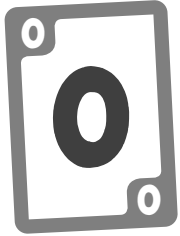
Why use the story point instead of person·hours or person·days?

- 1 Business stakeholders have a difficult time understanding the concept of “ideal person·hours” or “ideal person·days”
They tend to forget the “ideal” part!
- 2 Disparity between how fast certain team members can complete work
An experienced team member who has worked on similar projects for years will think, “I can do that in about an hour”, while the junior one might take more than one day to accomplish the same task
- 3 High degree of optimism present with most highly creative teams
It is easy to think “It’s a day,” while it's really three days’ worth of work

Story points eliminate these problems by disconnecting the size of a user story from the time it will take to implement it

When time is taken off the table, teams tend to be much more effective at estimating relative size

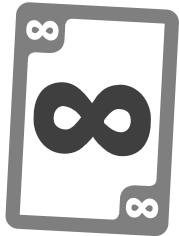
Scrum Cards



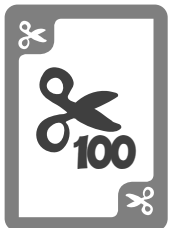
The zero card — Sometimes the list of stories will include one that is very straightforward, where the team believes it will only be a few minutes of work, or perhaps an hour or two.
The story is perceived too small to be an ½ story point value effort!



The question mark card — The team should discuss each story before voting, but sometimes a team member really has no idea what to estimate. In this case the team should discuss the story further and vote again.
As a general rule, if this card is used it means the team needs to discuss the stories more so every team member understands them enough to be able to vote with a story point value.

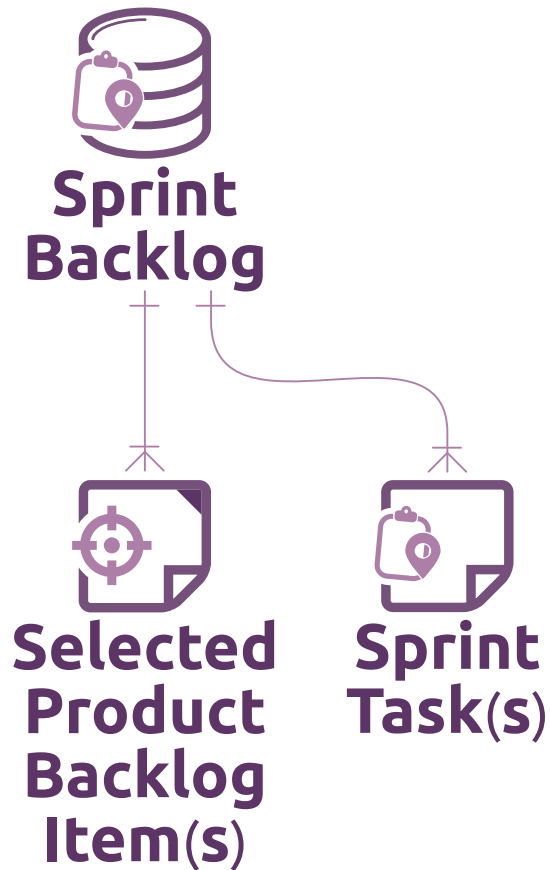


The infinity card — Sometimes stories are so large (epic), insufficiently defined or risky that the team member doesn't feel comfortable placing any sort of story point value on them.
These types of stories either need clarification or need to be scaled down into smaller pieces.



The one-hundred card — Sometimes the team member just sees that the story is an epic and needs to be disaggregated into more atomic stories.

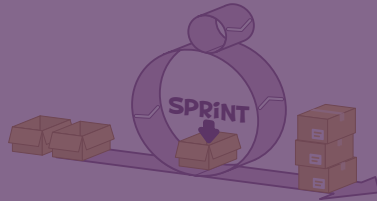
Sprint Backlog & Board



- ➔ Is a list made of **selected product backlog items** to be developed during the sprint
- ➔ Is augmented with the **set of tasks** required to achieve the sprint goal, i.e. demonstrating a PSPI at the end of the sprint

Project Management

What are we going to see together?



Lifecycle

Roles



(Development)
Team



Scrum Master
(SM)



Product Owner
(PO)



Project Stakeholders



SCRUM

Processes



Sprint planning meeting



Daily Scrum



Sprint review meeting



Sprint retrospective meeting



Backlog refinement meeting

Artifacts



Deck of
Scrum
Cards



Product
Backlog



Product
Backlog
Item(s)



Sprint
Backlog



Selected
Product
Backlog
Item(s)

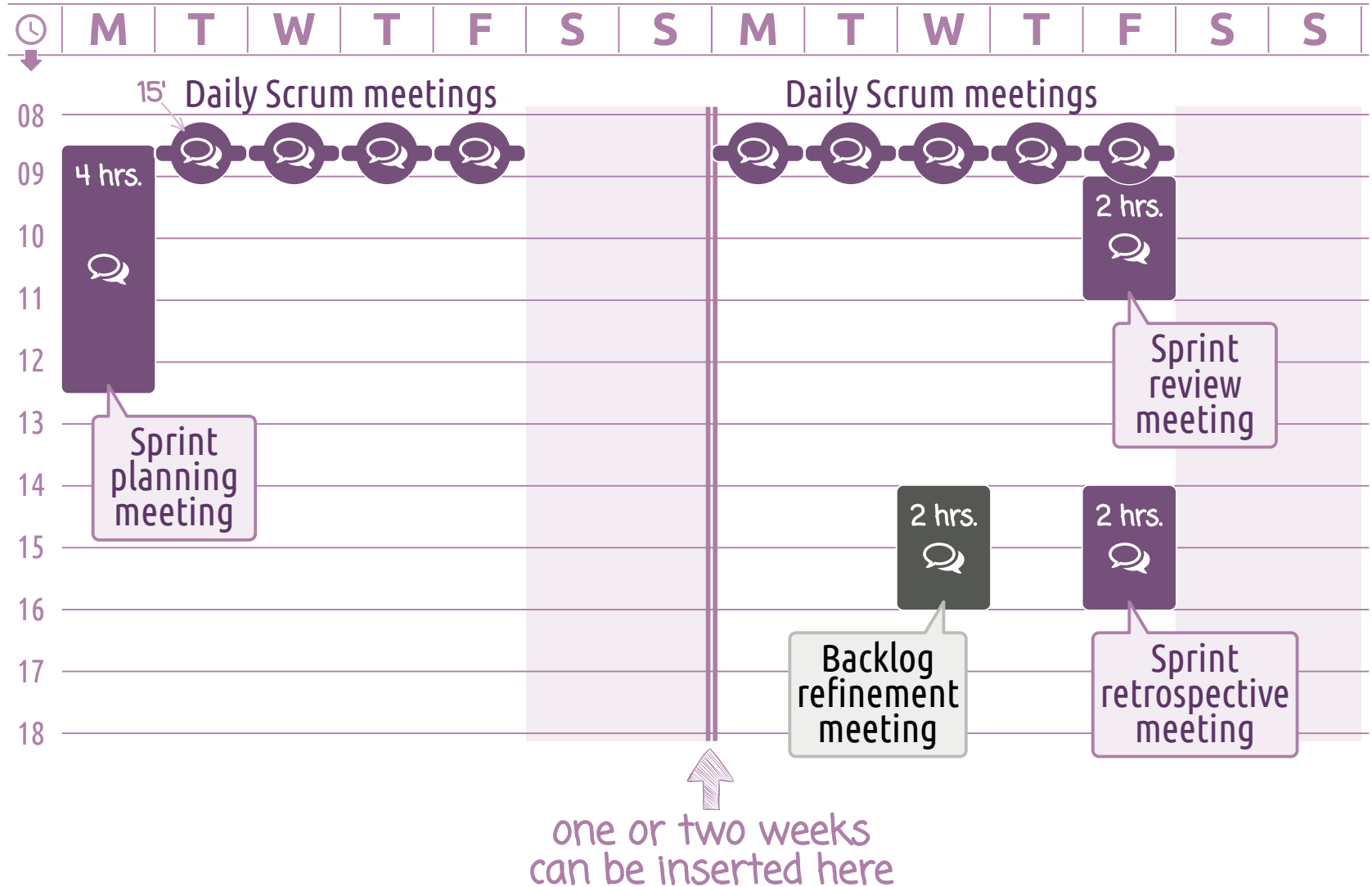


Sprint
Task(s)



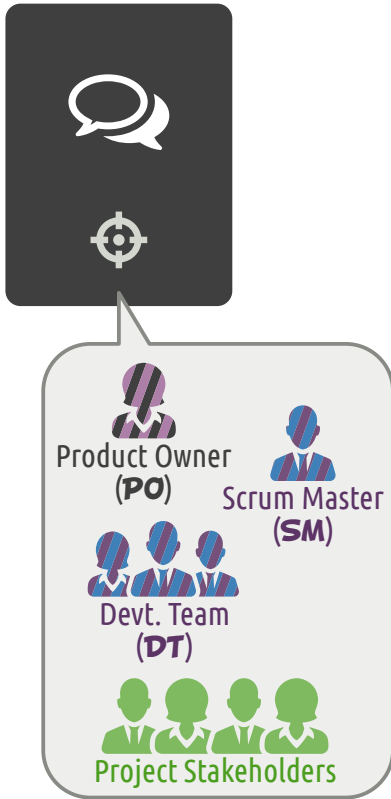
Sprint
Burndown
Chart(s)

Processes Sprint Meetings



Processes

Meetings



a.k.a.
Backlog Grooming Meeting
Backlog Estimation Meeting
Backlog Story Time Meeting

Backlog Refinement Meeting

- ➔ On **day $n - 2$** p.m. (2 hours)
and at least once before the project starts
- ➔ Aims at **looking at the product backlog**, i.e.:
 - ➞ Clarifying items and requirements
 - ➞ Estimating the effort →  Scrum Cards
 - ➞ Decomposing epics into smaller items
- ➔ Each product backlog item shall be:
 - Independent
 - Negotiable
 - Valuable
 - Estimable
 - Small
 - Testable

“INVEST”

“INVEST”

Independent — As the team learns and implements modern engineering practices that reduce item dependencies, the product owner gains the ability to set priorities by business value rather than technical considerations.

Negotiable ... and Negotiated — A good product backlog item shall be negotiable and not an explicit contract for features. Rather, details will be co-created by the product owner and the team along its implementation.

Valuable — A well-formed product backlog item shall be stakeholder- or user-centric. So its business value is clear.

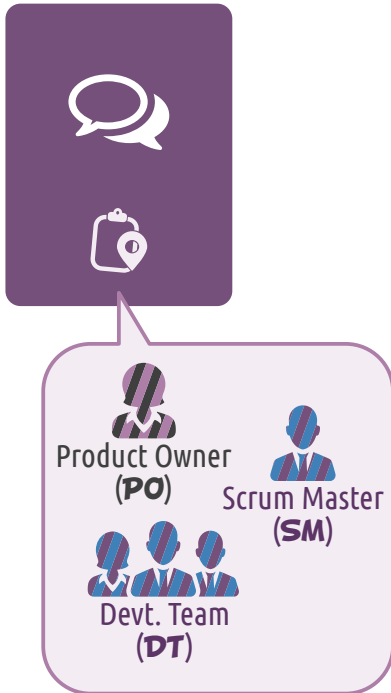
Estimable — If project participants cannot set a rough estimate of the effort required for implementing a product backlog item, this means that it is insufficiently clear and shall be discussed further.

Small — A single product backlog item shall be no bigger than a quarter of the team effort for one sprint. Ideally, it should not exceed two team-days.

Testable — Each product backlog item shall have a clear and bright finish line agreed between the product owner and the team.

Processes

Meetings



Sprint Planning Meeting

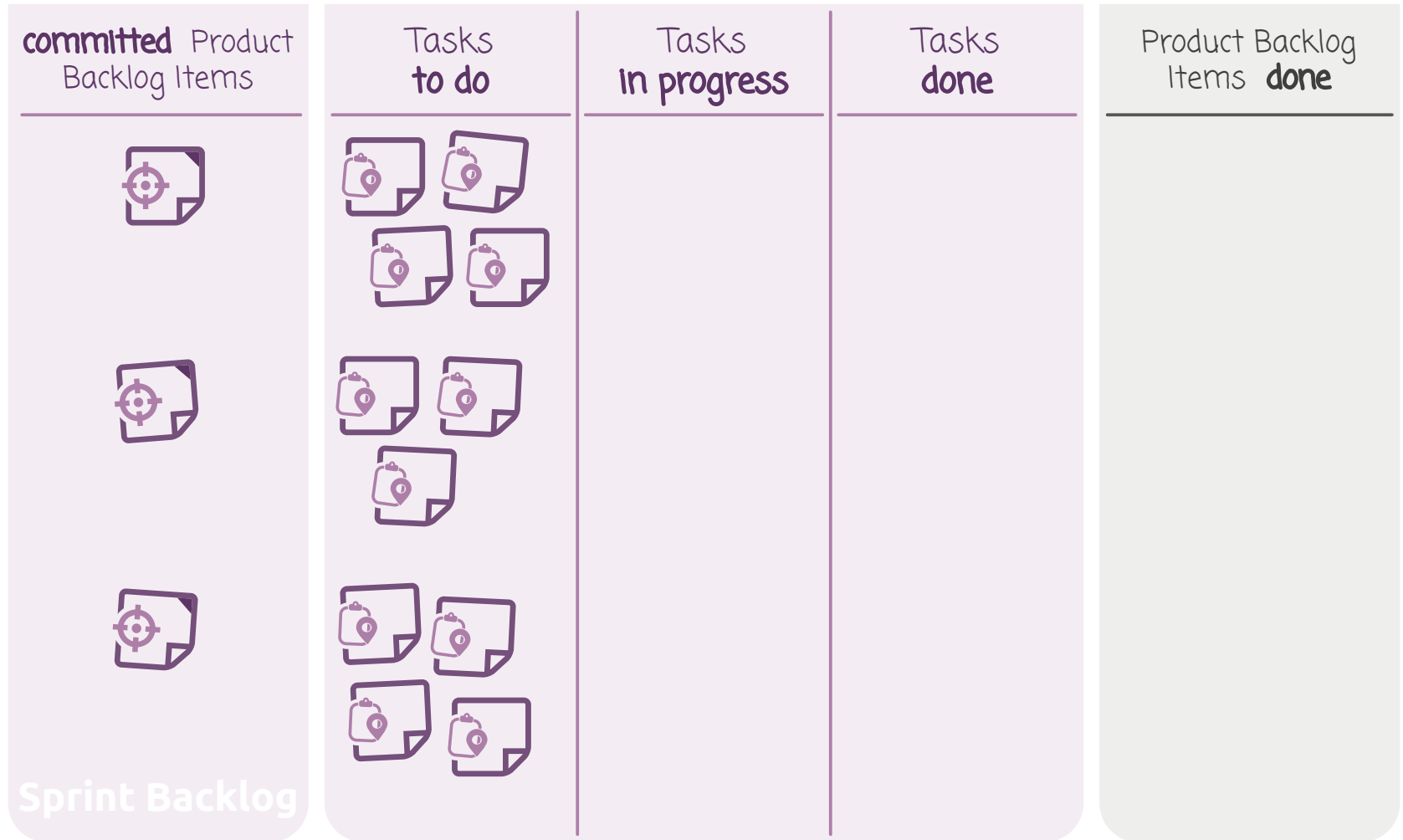
- ➔ On **day 1** a.m. (4 hours)
- ➔ Aims at:
 - ① Selecting and committing the team to a **set of product backlog items** to develop in the sprint → Sprint Backlog
 - ② Coming up with the corresponding **tasks**



1. Sprint goal [PO]
2. Discussion [team+PO]
3. PBI selection [team]
4. Task identification [team]

Artifacts

Sprint Board *after planning meeting*



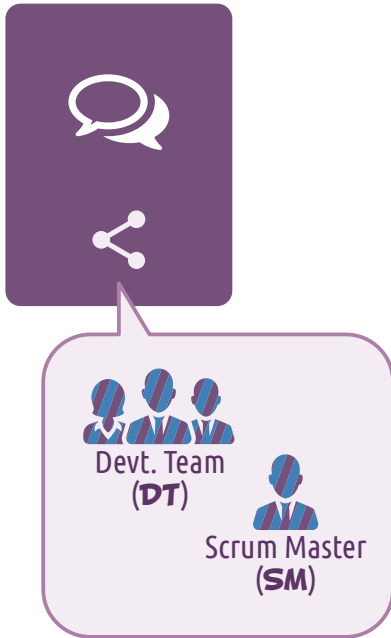
Sprint Backlog

WHAT

HOW

Processes

Meetings



Daily Scrum (Meeting)

- ➔ Every day*, a.m. (15 minutes, stand up)
* except on day 1
- ➔ Aims at reporting to each other:
 - ➔ **W**hat was done since the last Scrum meeting
 - ➔ **W**hat is intended to be done before the next Scrum meeting
 - ➔ **W**hat impedes team members, blocks their progress, reduces their effectiveness



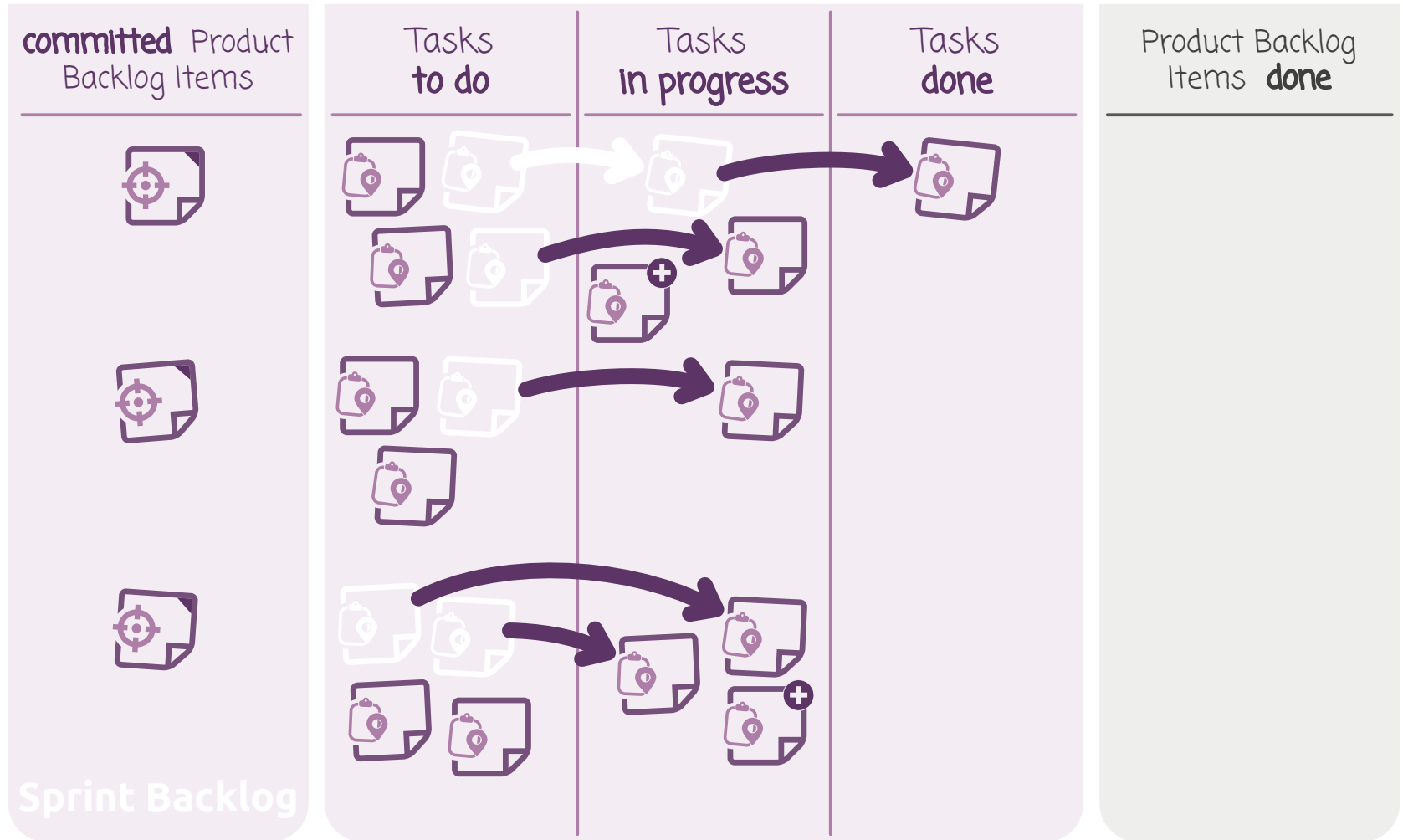
1. 3W-reports [team members]
incl. changes onto the sprint board
↑
Unique agenda item

a.k.a.
Daily Huddle
Daily Standup



Artifacts

Sprint Board *during daily Scrum*



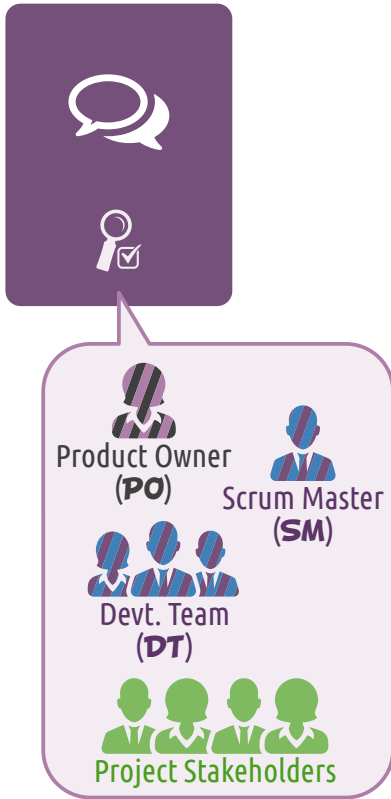
Sprint Backlog

WHAT

HOW

Processes

Meetings



a.k.a.
Sprint Demo Meeting

Sprint Review Meeting

- ➔ On **day n** a.m. (2 hours, rather informal)
- ➔ Aims at:
 - ➞ Demonstrating the increment implemented in the sprint
 - ➞ Declaring which items are done
 - ➞ Getting feedback w.r.t. what was done



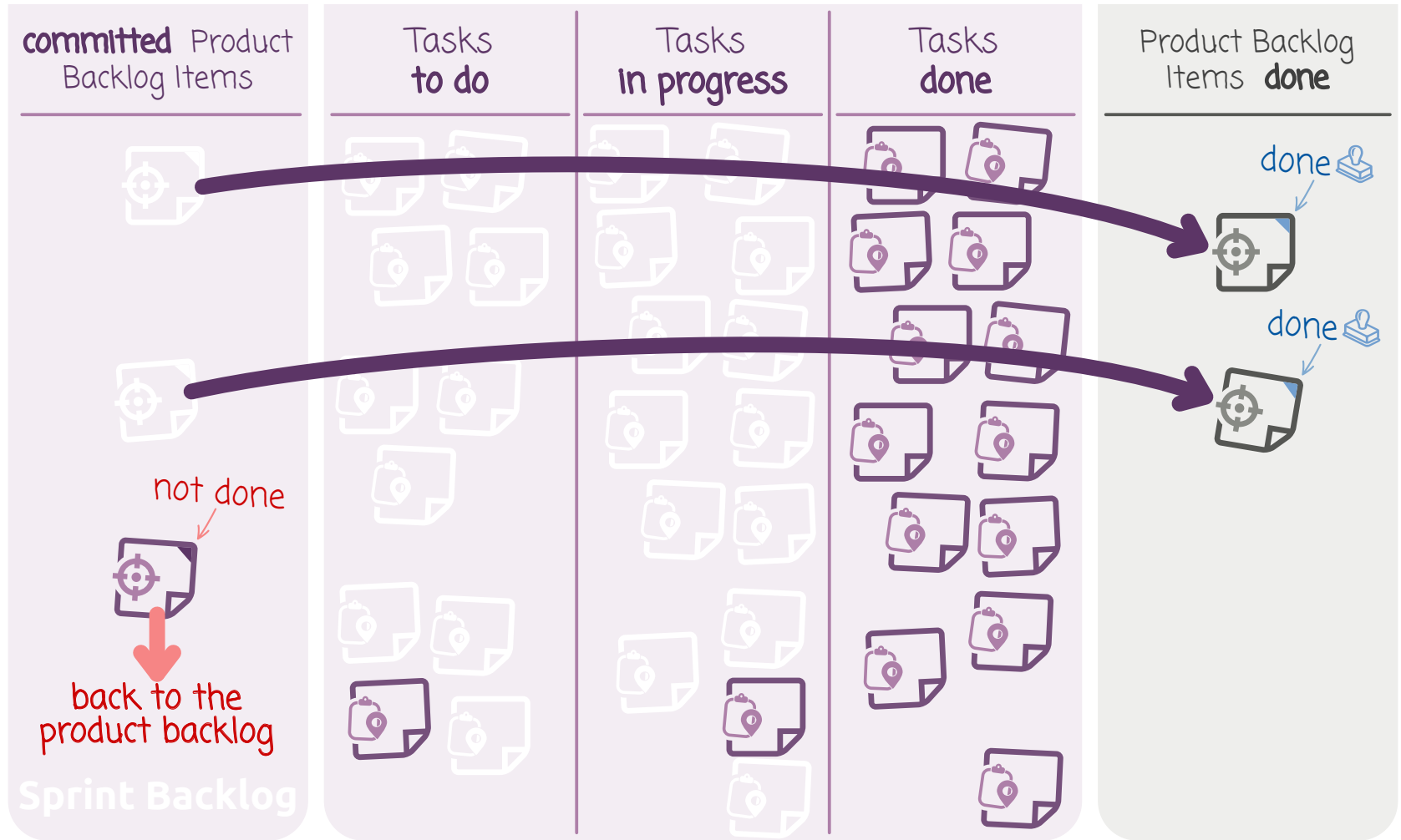
1. PSPI demonstration [team]
2. *Done-declaration* [PO]
3. Velocity tracking (optional)
4. Stakeholder feedback [all]



It is also possible that after the demo the PO realizes that the expected product shall be different.
This should not be a source of frustration for the team!

Artifacts

Sprint Board *during review meeting*



WHAT

HOW

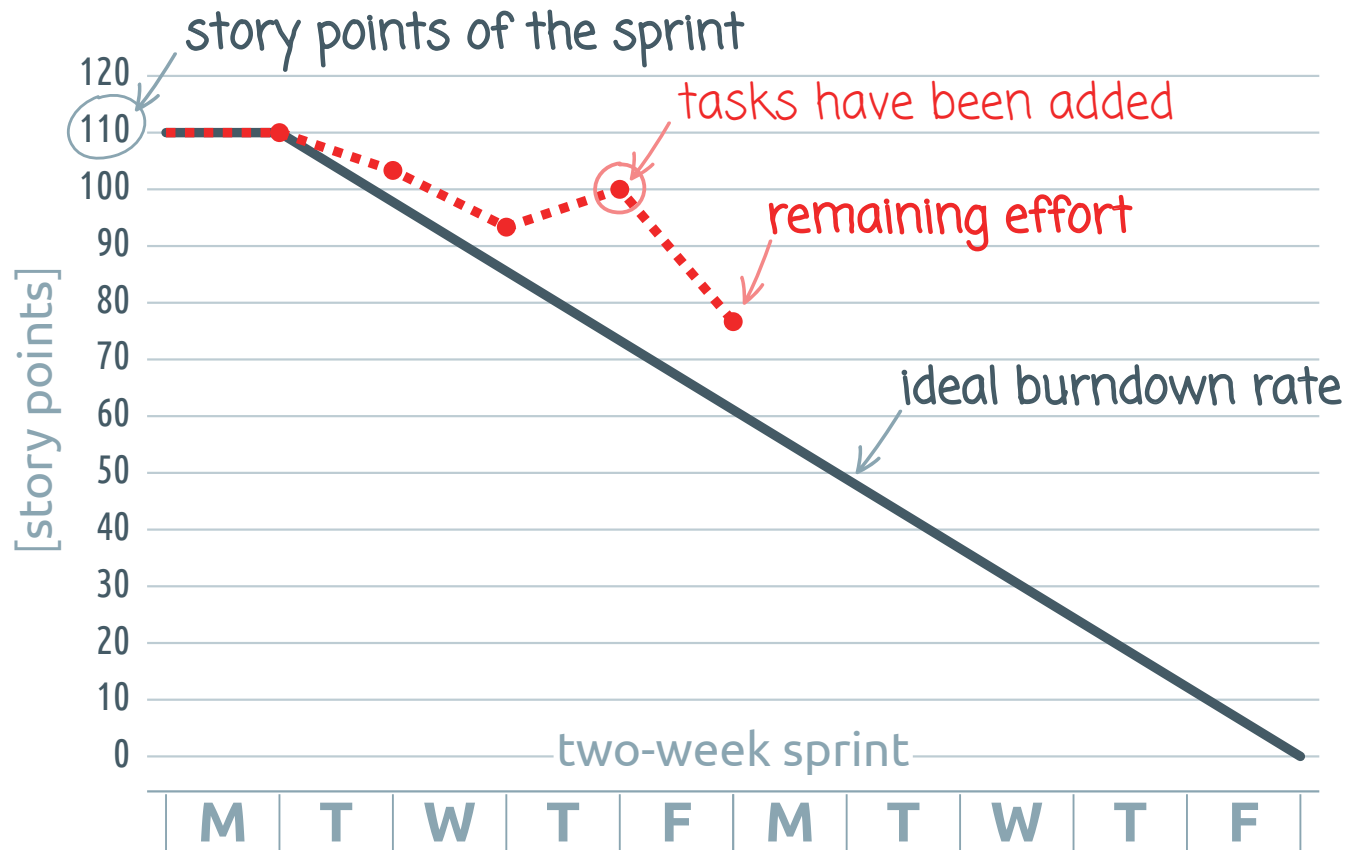
Definition of Done (DoD)

- ➔ Is a **shared understanding of expectations** that the increment (PSPI's) must live up in order to be releasable or deliverable*
- ➔ Is **specific to each team** and/or project
- ➔ Practically, is a clear and concise **list of broad requirements** the product/PSPI shall adhere to for the team to call it complete
- ➔ Is **consistent across all the items** of a product backlog
- ➔ Is somehow equivalent to the concept of **acceptance criteria**, but at a broader level
- ➔ Should be defined in the Project Management Plan

Burndown Chart

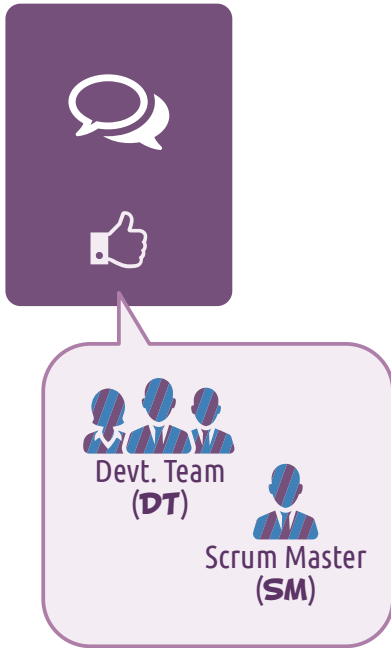


Within a sprint, is a graphical representation of the work left to do versus time*



Processes

Meetings



Sprint Retrospective Meeting

- ➔ On **day n** p.m. (2 hours)
- ➔ Aims at collecting **feedback** on the **process**:
 - ➔ **W**hat went well
 - ➔ **W**hat could be improved
 - ➔ **W**hat did project participants learn
 - ➔ **W**hat still puzzles themand **deciding** what to do...
- ➔ By means of these retrospective meetings, the team shall **take ownership of their own process**



1. 4W-reports [team members]
2. Action plan [team members]

Embedding



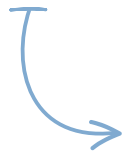
SCRUM

in



openSE

→ Authoring a Project Proposal/Roadmap



Gathering needs/user requirements
Working out solutions and concepts
Developing the product breakdown structure*
Defining product* requirements
Developing prototypes, mock-ups...
Authoring the Conceptual Design Report
etc.

* Product can be replaced by equipment, system, facility, process, service

Embedding



SCRUM

in



openSE

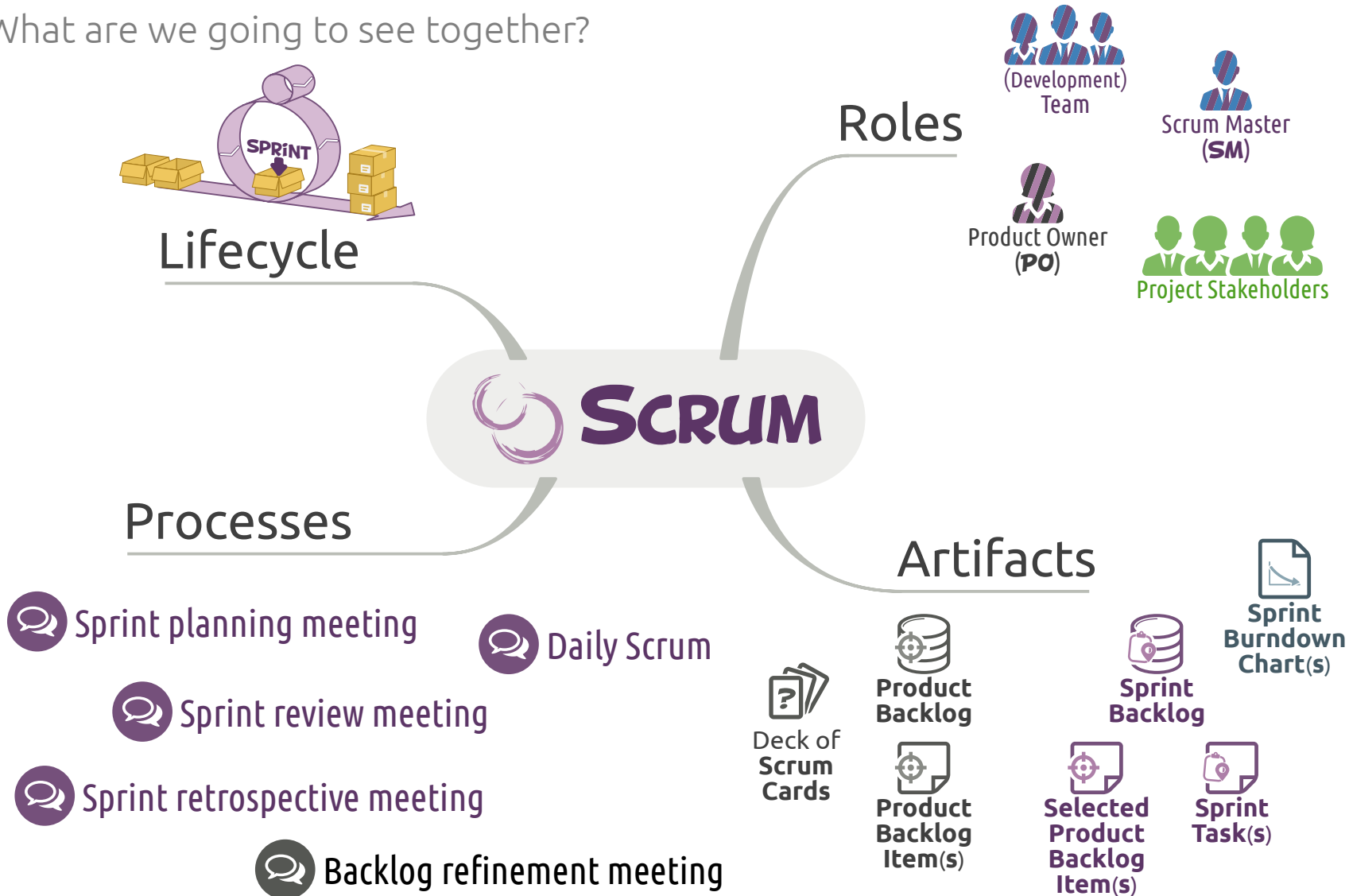
Designing complex sub-systems
Authoring complex documents (e.g. TDR)
Developing further prototypes, mock-ups...
Developing electronics, control systems
Developing software
Commissioning sub-systems
etc.



Running further improvement projects

Project Management

What are we going to see together?



[en]

Scrum

Sprint
Framework
Lifecycle
Roles
Artifacts
Processes
Events
Rules

Stakeholders
Product Owner
Development
Team
Team member(s)
Scrum Master

Product Backlog
—— Item(s)
Sprint Backlog
Potentially shippable
product increment

[fr]

Mêlée (Scrum)

Sprint
Cadre de travail
Cycle de vie
Rôles
Artéfacts
Processus
Événements
Règles

Parties prenantes
Responsable du produit
Équipe de dévelop-
pement
Équipier(s)
Maître de mêlée

Registre (de) produit
Élément(s) du ——
Registre de/du sprint
Incrément de/du produit
potentiellement livrable

Backlog Refinement
Meeting
Sprint Planning
Meeting
Daily Scrum

Sprint Review
Meeting
Sprint Retrospective
Meeting

Need/requirement
User story
Story card
Epic
Story point(s)
Task
Velocity
Burndown Chart
Definition of Done
Acceptation Criteria

Réunion de définition
du registre produit
Réunion de planification
du sprint
Mêlée quotidienne
(Scrum quotidien)
Réunion de revue
du sprint
Réunion rétrospective
du sprint

Besoin/exigence
Récit utilisateur
Fiche de récit
Épopée
Point(s) de récit
Tâche
Rapidité (*vélocité*)
Graphe d'avancement
Définition du fait/fini
Critère d'acceptation



[**cern.ch/openSE**](http://cern.ch/openSE)

[**cern.ch/quality**](http://cern.ch/quality)

[**cern.ch/go/8rMF**](http://cern.ch/go/8rMF)