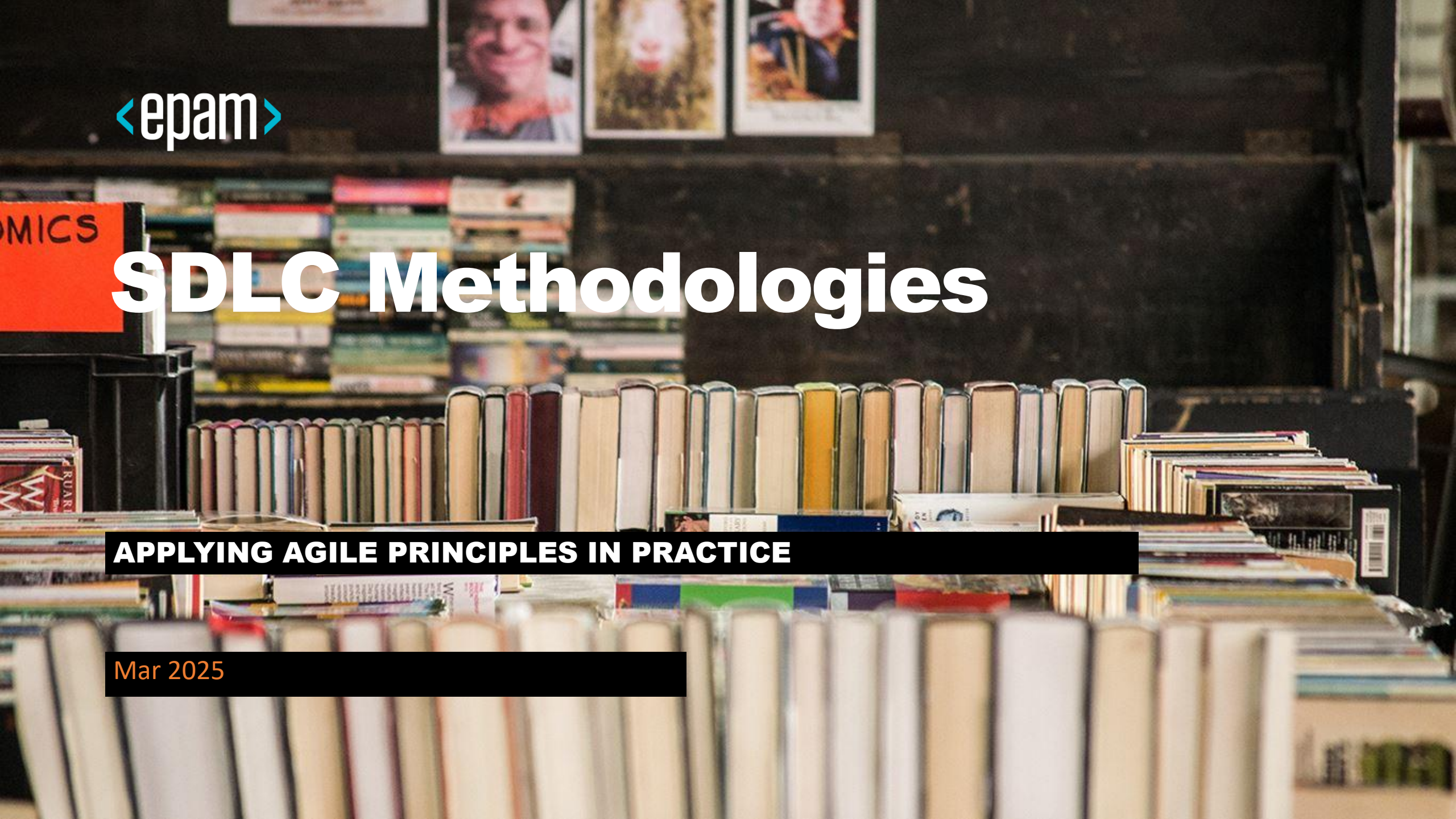


SDLC Methodologies

APPLYING AGILE PRINCIPLES IN PRACTICE

Mar 2025

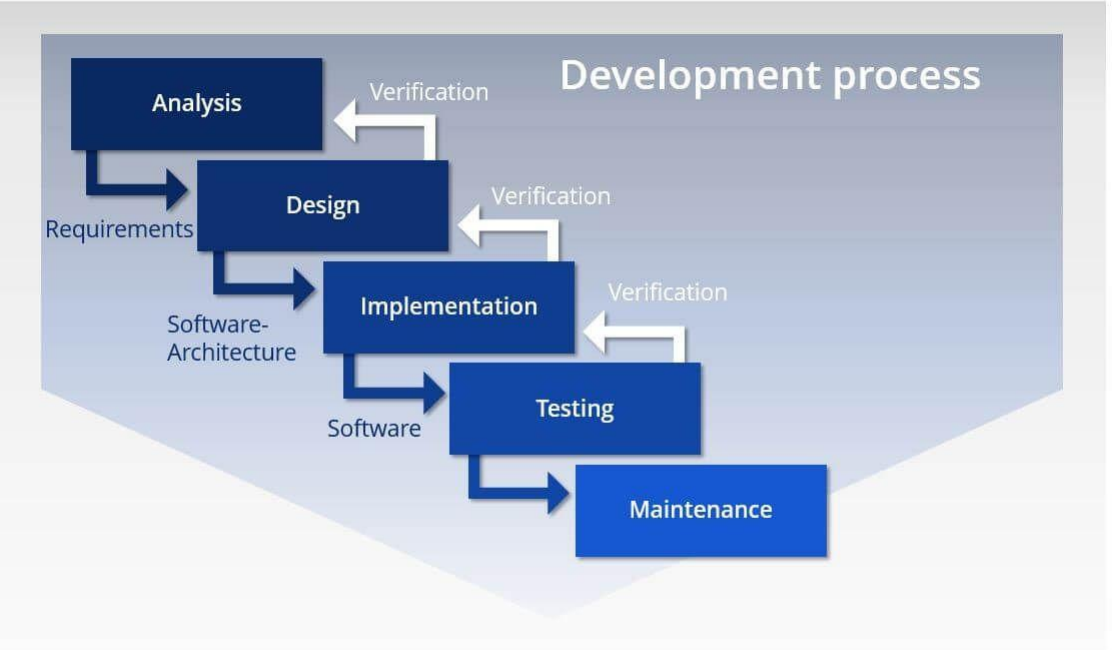
SDLC METHODOLOGIES

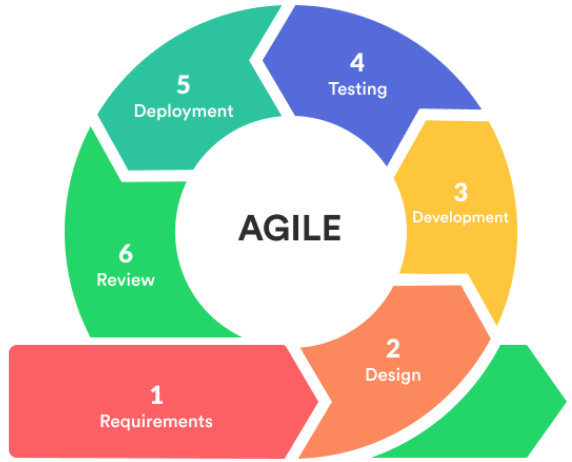
SDLC Methodologies

A methodology is clearly defined processes or software development life cycle to ensure the product is of high quality

It identifies phases and structured flow from one phase to another phase.

Waterfall and Agile are two popular methodologies, but of very different models





AGILE METHODOLOGY

AGILE IS ABOUT MINDSET

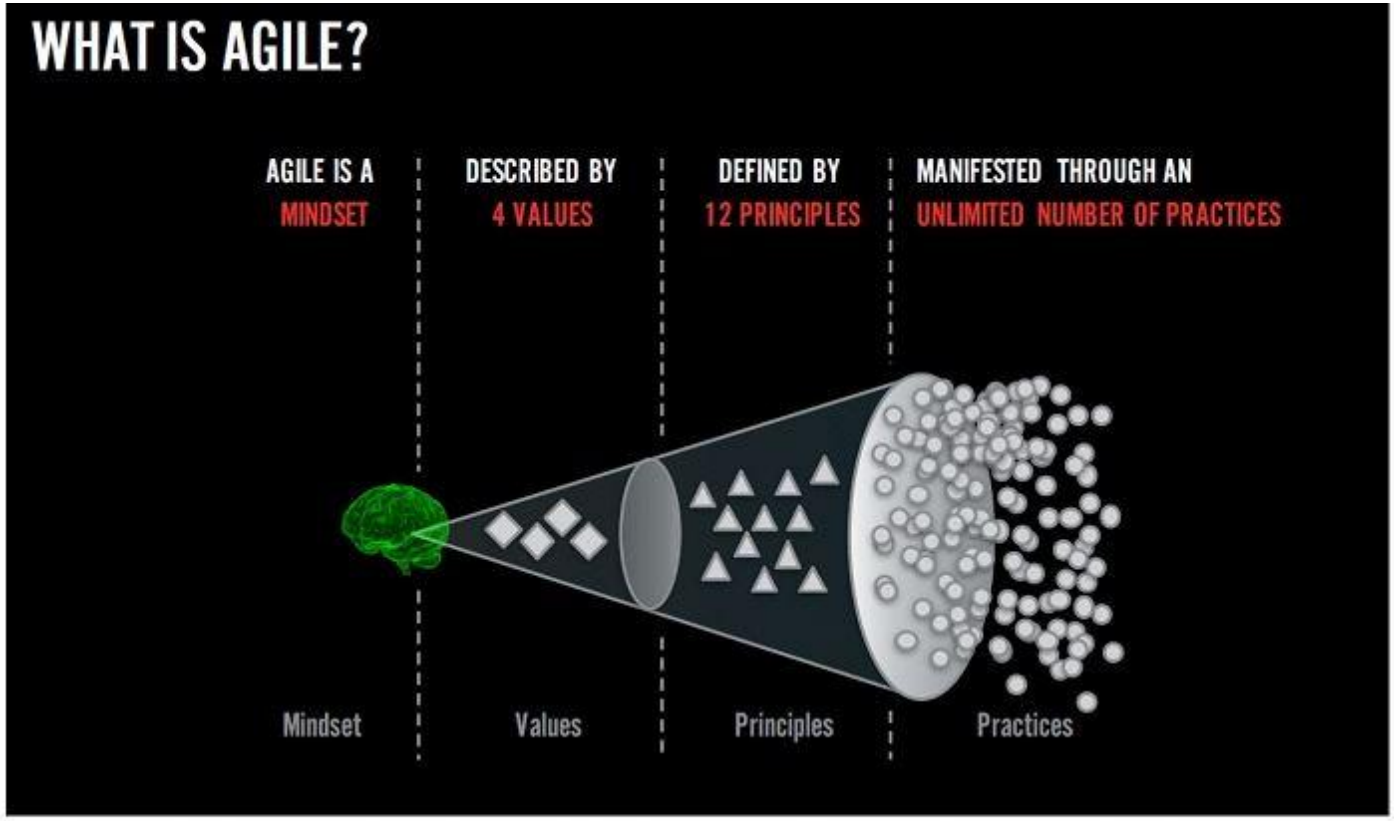


Image from Ahmed Sidky, Riot Games and ICAgile reproduced with permission
Agile is a mindset, described by the four values and the twelve principles of the Agile Manifesto, and manifested through an unlimited number of practices, tools and processes. Implementing the practices, tools and processes without the Agile mindset, values and principles of the Agile Manifesto is not Agile.

http://www.forbes.com/sites/stevedenning/2016/08/13/what-is-agile/amp/

Agile umbrella

Scrum
Lean software development
Kanban (process + method)
Extreme Programming (**XP**)
Continuous Integration (**CI**)
Continuous Delivery (**CD**)
Feature Driven development (**FDD**)
Test Driven Development (**TDD**)
Crystal Clear
...

Lightweight approaches

Scrum-of-Scrums
Scrum at Scale (**Scrum@Scale**)
Large-scale Scrum (**LeSS**)
Scaled Agile Framework (**SAFe**)
Disciplined Agile Delivery (**DAD**)
Dynamic Systems Development Method (**DSDM**)
Agile Project Management (**AgilePM**)
Agile Unified Process (**AUP**)
Open Unified Process (**OpenUP**)
...

Fuller approaches (beyond 1 team)

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 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 in the items on the right, we value the items on the left more.

Kent Beck – Mike Beedle – Arie van Bennekum – Alistair Cockburn – Ward Cunningham – Martin Fowler – James Grenning – Jim Highsmith Andrew Hunt – Ron Jeffries – Jon Kern – Brian Marick – Robert C. Martin – Steve Mellor – Ken Schwaber – Jeff Sutherland – Dave Thomas

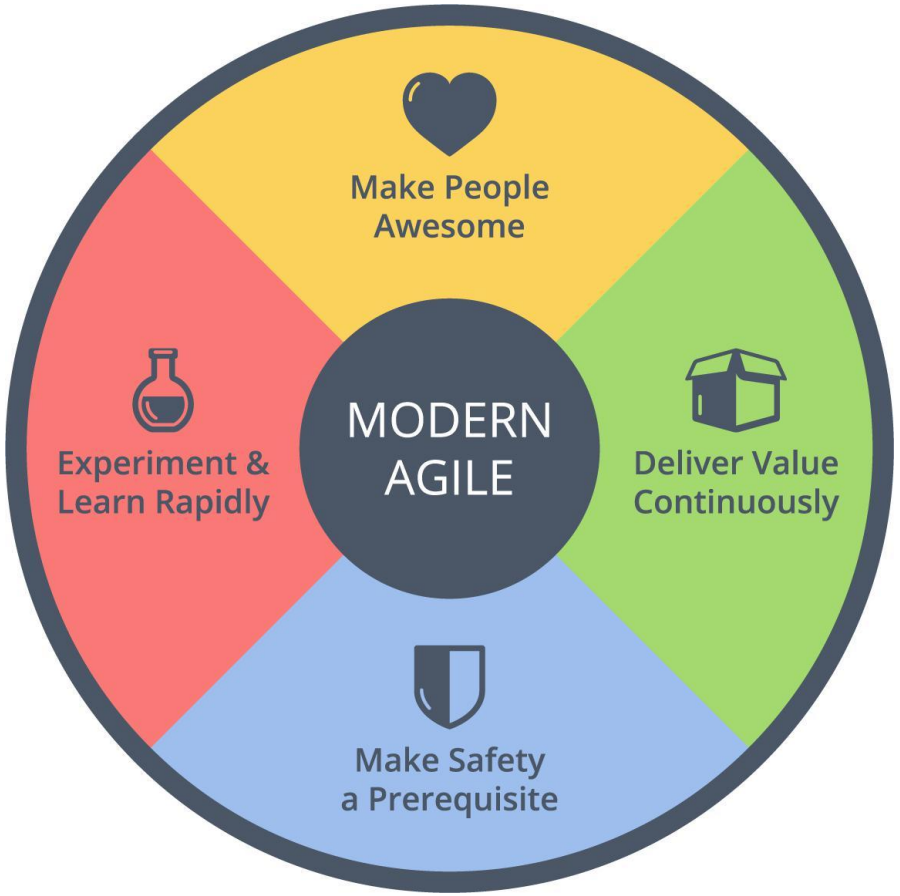
MANIFESTO FOR AGILE SOFTWARE DEVELOPMENT

INDIVIDUAL AND INTERACTIONS **OVER** *PROCESSES AND TOOLS*

WORKING SOFTWARE **OVER** *COMPREHENSIVE DOCUMENTATION*

CUSTOMER COLLABORATION **OVER** *CONTRACT NEGOTIATION*

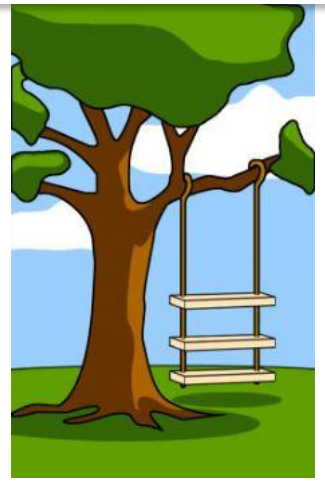
RESPONDING TO CHANGE **OVER** *FOLLOWING A PLAN*



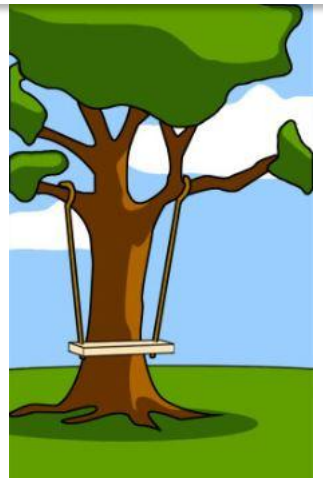
PRINCIPLES OF THE AGILE MANIFESTO

1. Our highest priority is to **satisfy the customer** through **early** and continuous delivery of valuable software.
2. **Welcome** changing requirements, even late in development. Agile processes harness **change** for the customer's competitive advantage.
3. **Deliver working software frequently**, from a couple of weeks to a couple of months, with a preference to the shorter timescale.
4. Business people and developers must **work together daily** throughout the project.
5. Build projects around **motivated individuals**. Give them the environment and support they need and **trust** them to get the job done.
6. The most efficient and effective method of conveying information to and within a development team is **face-to-face** conversation.
7. **Working software** is the primary measure of **progress**.
8. Agile processes promote **sustainable** development. The sponsors, developers, and users should be able to maintain a **constant pace** indefinitely.
9. Continuous attention to **technical excellence** and good design enhances agility.
10. **Simplicity**--the art of maximizing the amount of work not done--is essential.
11. The best architectures, requirements, and designs emerge from **self-organizing** teams.
12. At **regular** intervals, the team **reflects** on how to become more effective, then tunes and **adjusts** its behavior accordingly.

HAVE YOU EXPERIENCED THIS BEFORE?

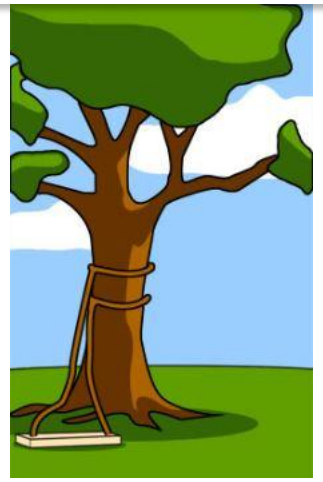


How the customer explained it

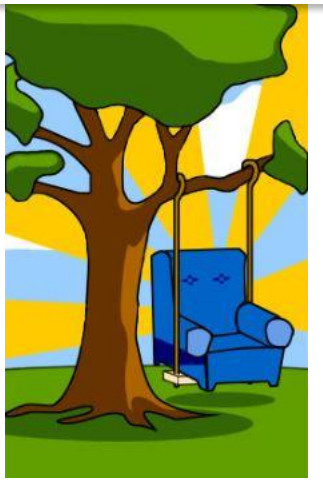


How the project leader understood it

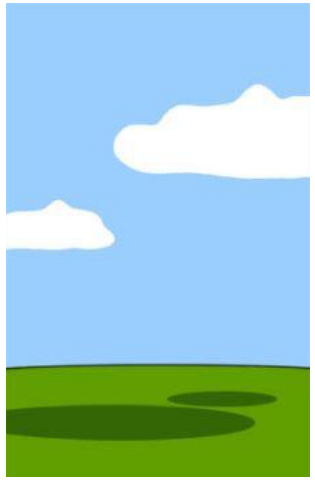
How the analyst designed it



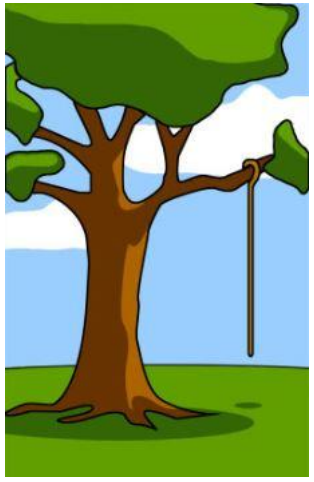
How the programmer wrote it



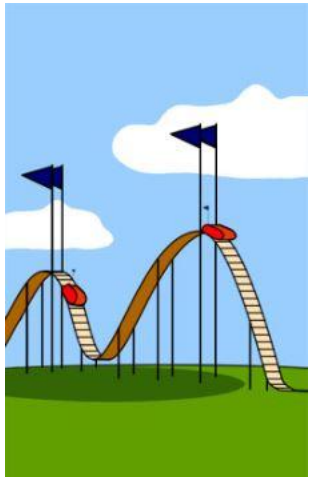
How the business consultant described it



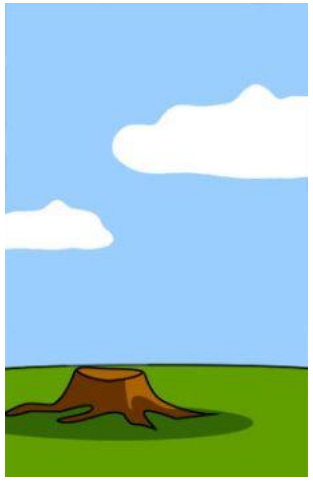
How the project was documented



What operations installed



How the customer was billed



How it was supported

What the customer really needed

Understanding vs Knowledge

https://www.youtube.com/watch?v=MFzDaBzBlL0

AGILE DEVELOPMENT

https://www.youtube.com/watch?v=1iccpf2eN1Q

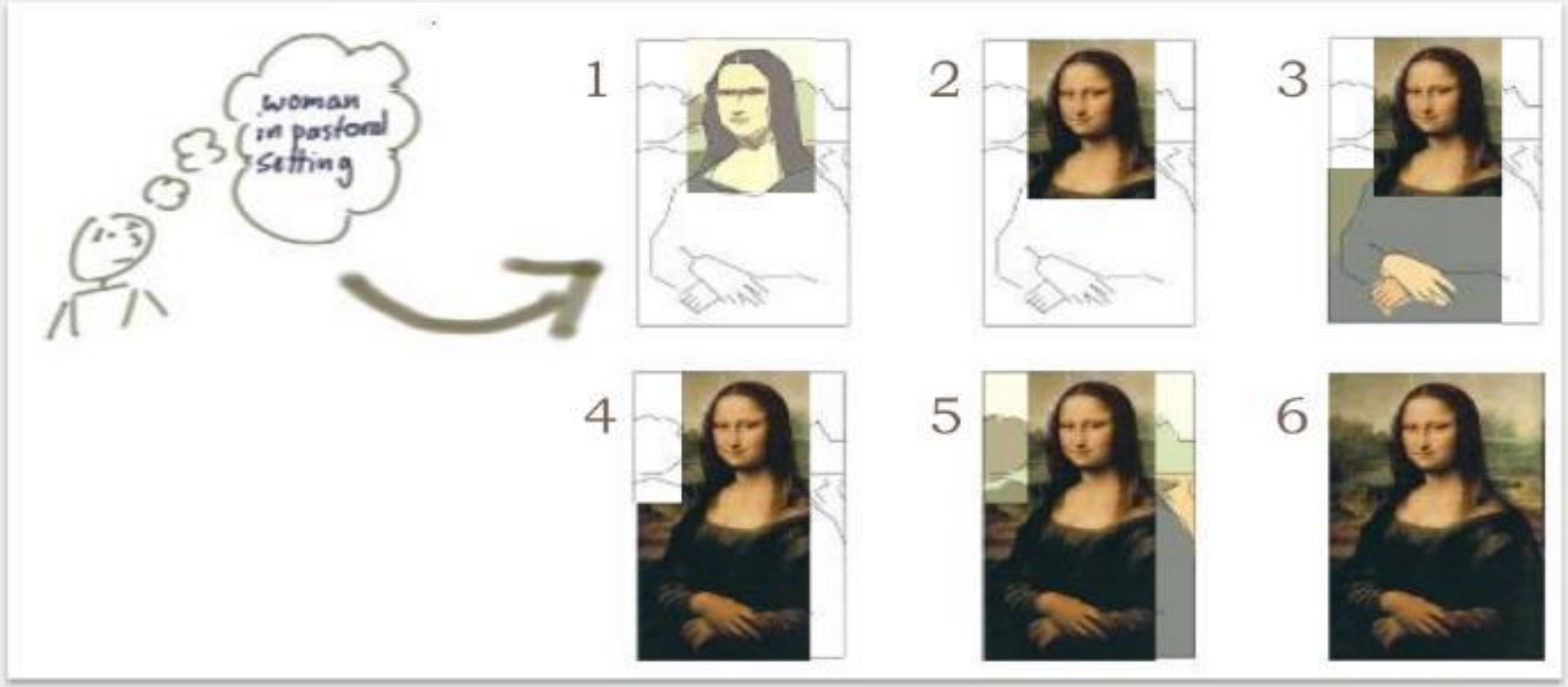
INCREMENTAL DEVELOPMENT

— Incrementing Mona Lisa (Source: Jeff Patton)

ITERATIVE DEVELOPMENT

— Iterating Mona Lisa (Source: Jeff Patton)

ITERATIVE & INCREMENTAL DEVELOPMENT



— Drawing Mona Lisa Iteratively and Incrementally

DO YOU COLLABORATE OR COOPERATE?

COLLABORATE

Cross functional team

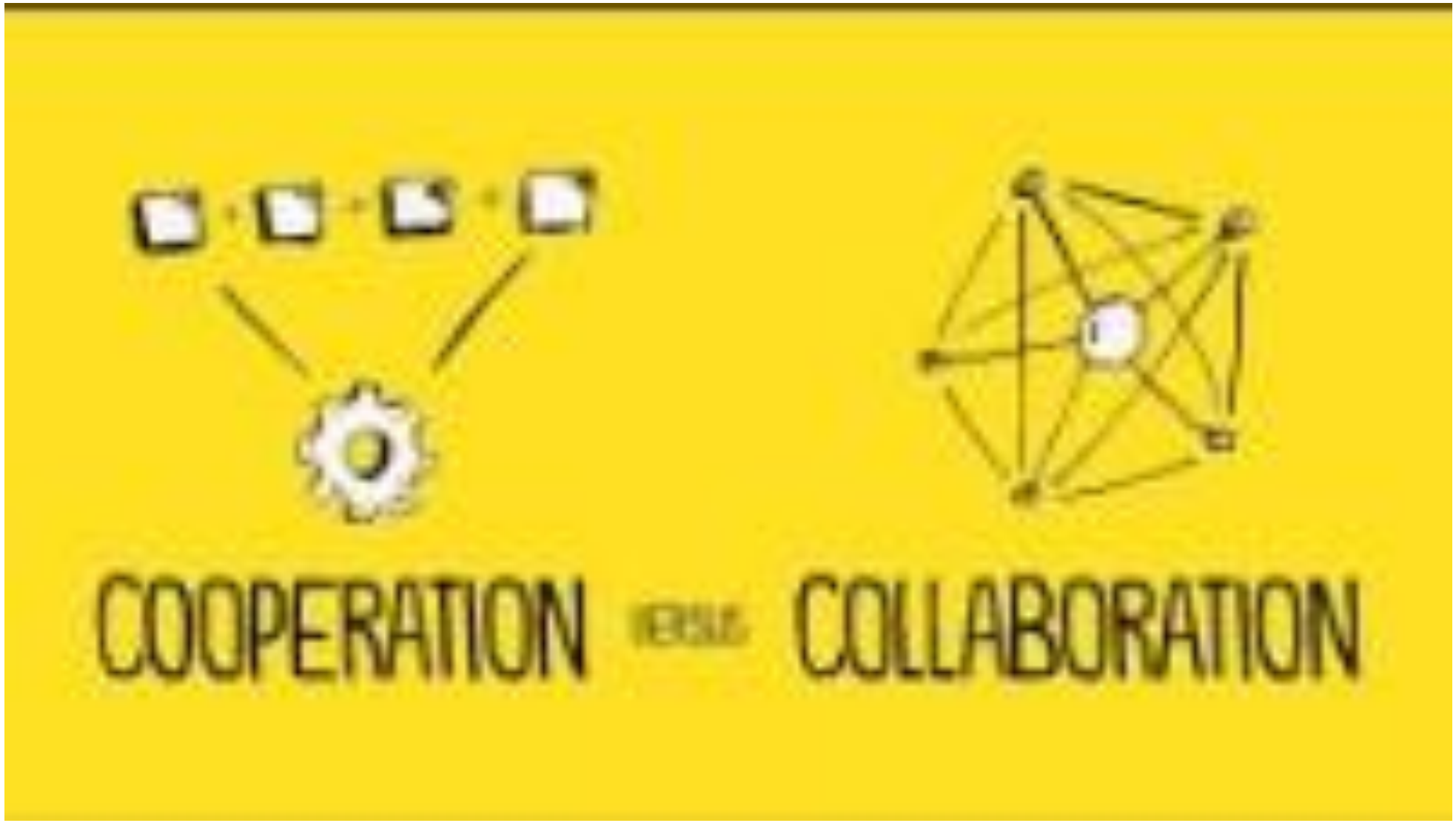
Functional

Common functional expertise

Cross-Functional

Representatives from the various functions

Individuals with different skill-sets



https://www.youtube.com/watch?v=Gr5mAboH1Kk

SCRUM IN PRACTICE

SCRUM ELEMENTS

Roles


PRODUCT OWNER


SCRUMMASTER


DEVELOPMENT TEAM

Events


SPRINT RETROSPECTIVE

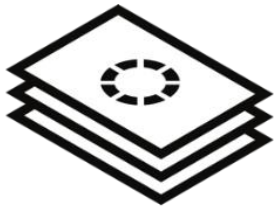

SPRINT REVIEW


SPRINT PLANNING

DAILY SCRUM

SPRINT

Artifacts


PRODUCT BACKLOG


SPRINT BACKLOG


PRODUCT INCREMENT

Other


RELEASE BURNDOWN


IMPEDIMENT

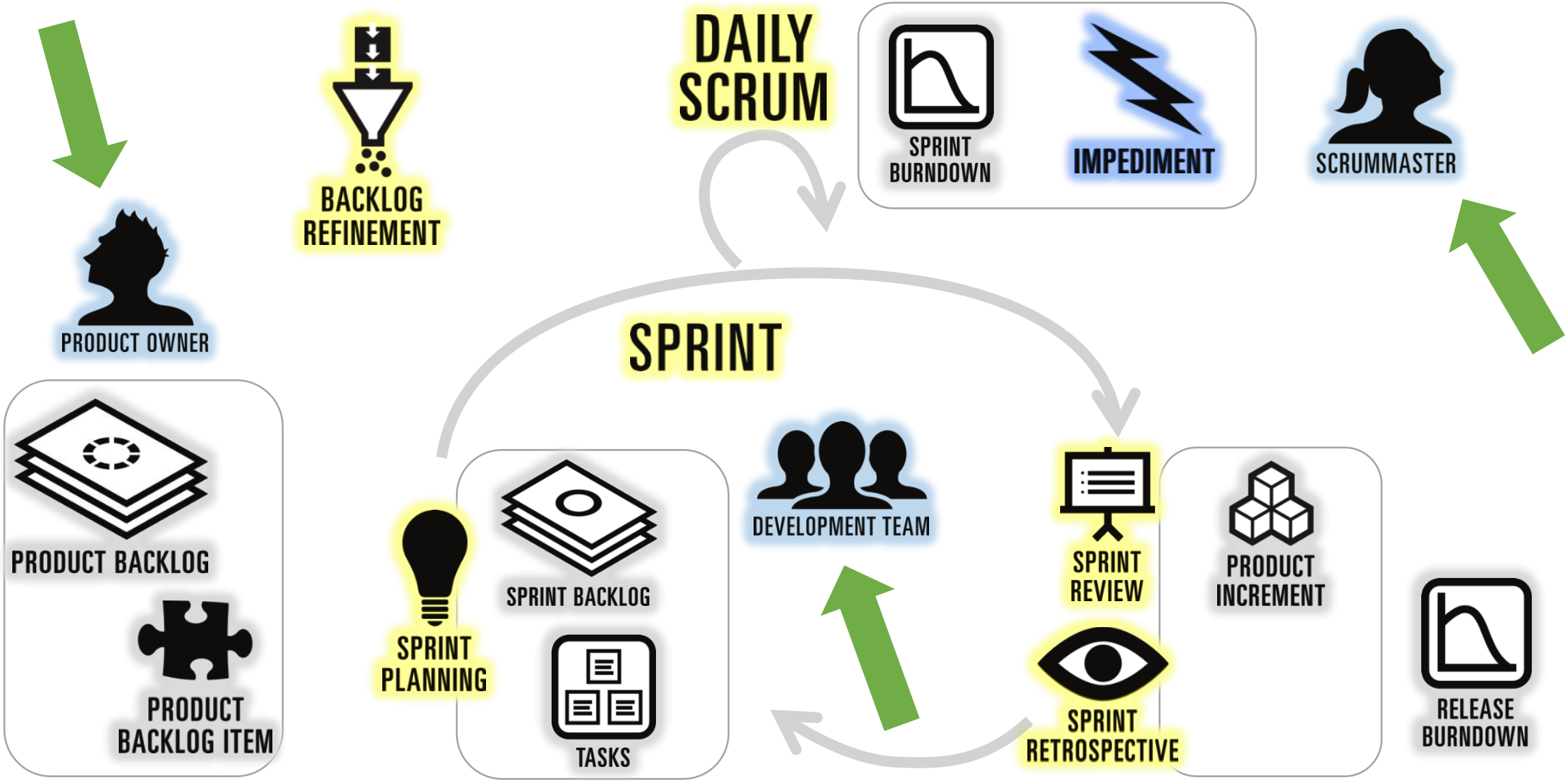

SPRINT BURNDOWN


BACKLOG REFINEMENT


TASKS


PRODUCT BACKLOG ITEM

SCRUM ROLES


DAILY SCRUM
SPRINT BURNDOWN
IMPEDIMENT
SCRUMMASTER
BACKLOG REFINEMENT
PRODUCT OWNER
SPRINT
PRODUCT BACKLOG
PRODUCT BACKLOG ITEM
SPRINT PLANNING
SPRINT BACKLOG
TASKS
DEVELOPMENT TEAM
SPRINT REVIEW
SPRINT RETROSPECTIVE
PRODUCT INCREMENT
RELEASE BURNDOWN

SCRUM ROLES

PRODUCT OWNER

Goal

- Maximizes product's value
- Drives product vision
- Single point of contact for decisions about the product, the *what*
- Takes into account all stakeholders
- Accepts the product
- Responsible for ROI

SCRUMMASTER

Feedback

- Removes impediments
- Helps the team adhere to the Scrum process
- Facilitates Scrum events
- Supports the team as servant leader
- Coaches the team in being agile
- *Not a project manager*

DEVELOPMENT TEAM

Context

- Is responsible for delivering the product, the *how*
- Self-organizing
- Cross-functional
- Provides transparency about process and progress
- Is responsible for continuous improvement
- Ideal team size: 7 +/- 3

SCRUM ARTIFACTS

1. **Product backlogs**

   List of new features, enhancements, bug fixes, tasks, or work requirements needed to build a product
   Maintained and refined by product owner

2. **Sprint backlogs**

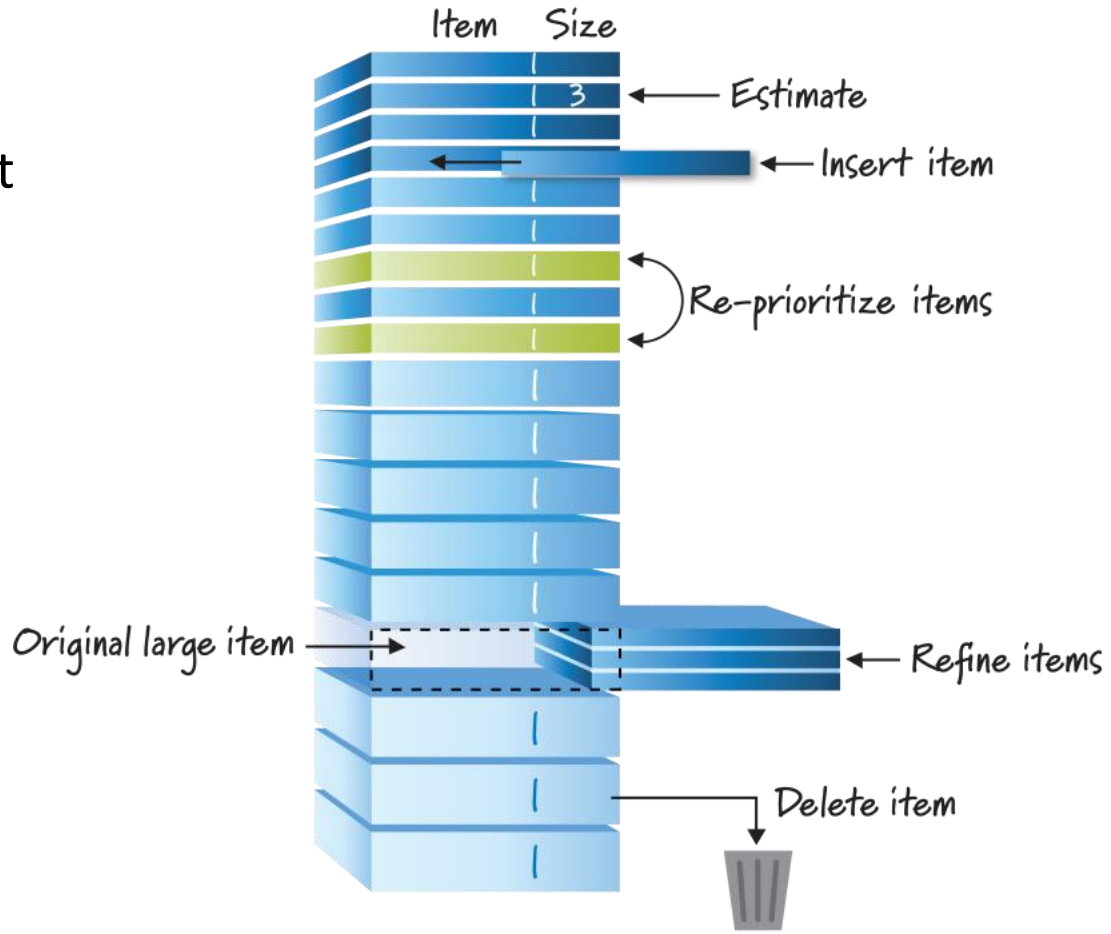
   Set of product backlog tasks chosen for sprint
   Updated during sprint planning
   Remaining sprint tasks will standby in the sprint backlog for a future sprint.

3. **Product Increment**

   An increment release at end of each sprint which was decided in sprint planning
   Each increment is a recognizable, visibly improved, operating version of the product

BACKLOG REFINEMENT

5-10% of each Sprint should be spent on refining the Product Backlog, with the Team and Product Owner. This is ideally done in a focused workshop. It is often done near the middle of each Sprint and at the end of it.

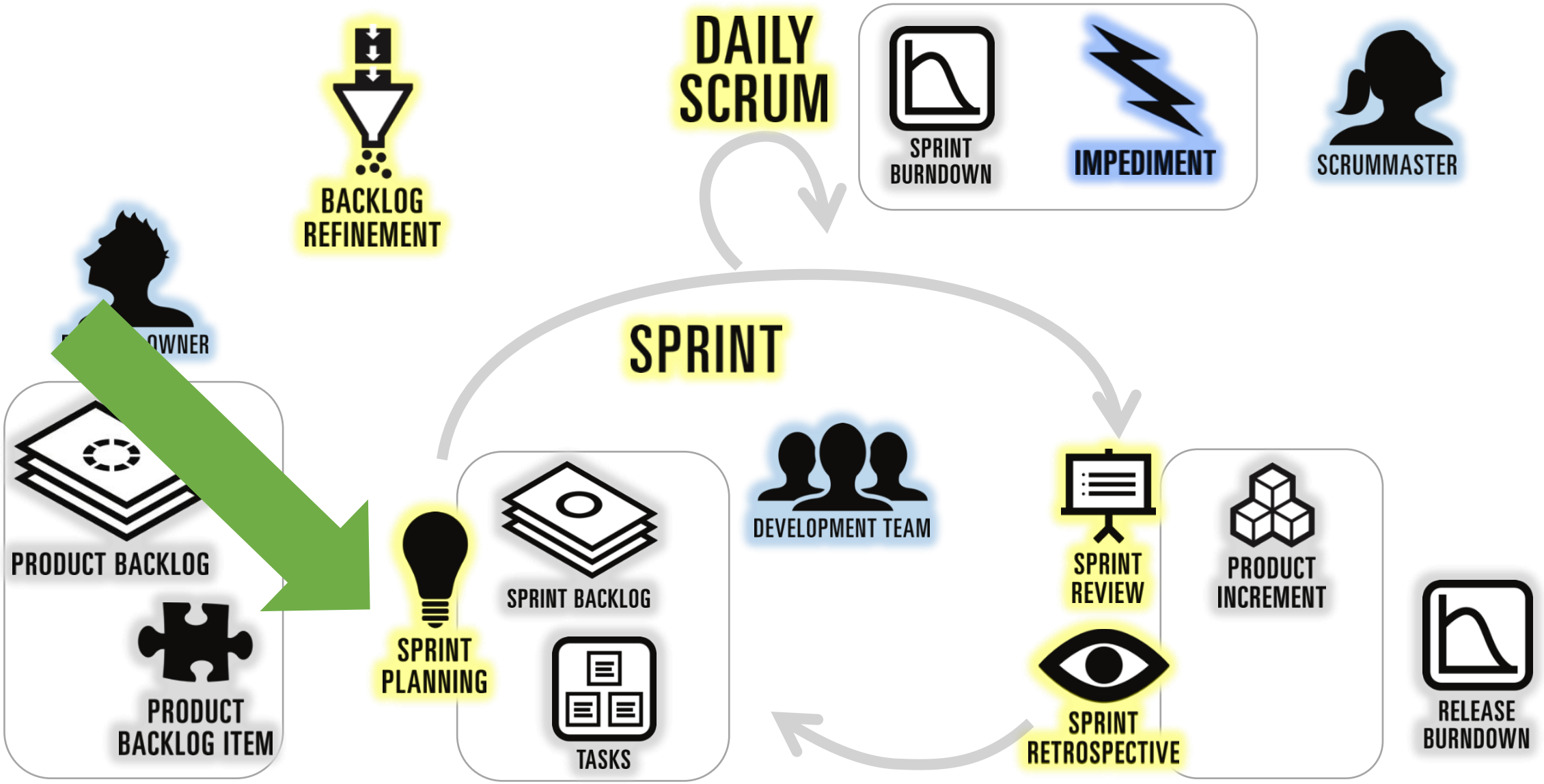


SPRINT

Objective: To deliver a Done, usable increment that meets the Sprint Goal

- Sprints are the heartbeat of Scrum, where ideas are turned into value
- A container that includes all the other events
- The Sprint has a timebox of one month – Usually 2 weeks
- A Sprint could be cancelled if the Sprint Goal becomes obsolete. Only the Product Owner has the authority to cancel the Sprint.
- During the Sprint:
  - No changes are made that would endanger the Sprint Goal;
  - Quality does not decrease;
  - The Product Backlog is refined as needed;
  - Scope may be clarified and renegotiated with the Product Owner as more is learned.

SPRINT PLANNING



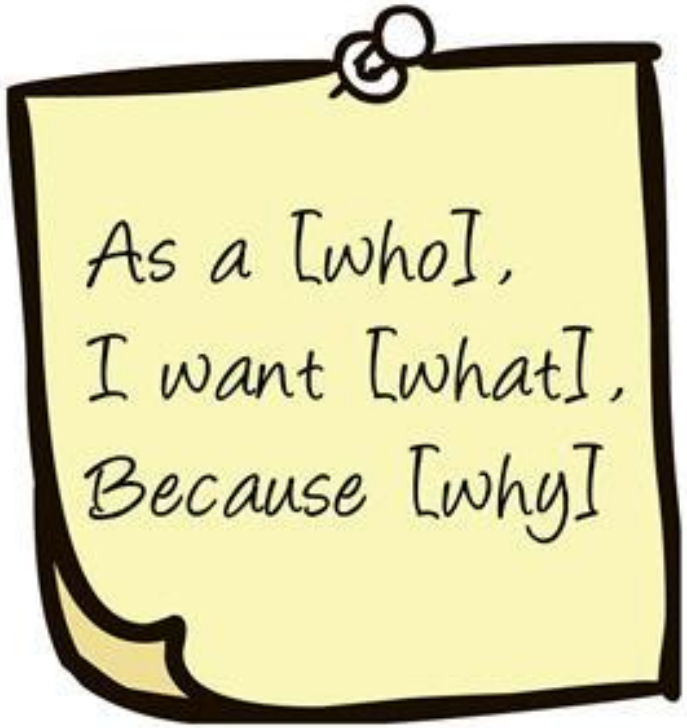
SPRINT PLANNING

- Maximize the delivery of value for the upcoming sprint
  - Take prioritization into account
  - Plan for completed user stories (according to DOD)
- Use spikes to mitigate risk
- Break stories down in technical tasks
- Plan for what is possible
- Velocity = average amount of story points the team can deliver in a sprint

Sprint Planning
Max 8 hours for 30d sprint

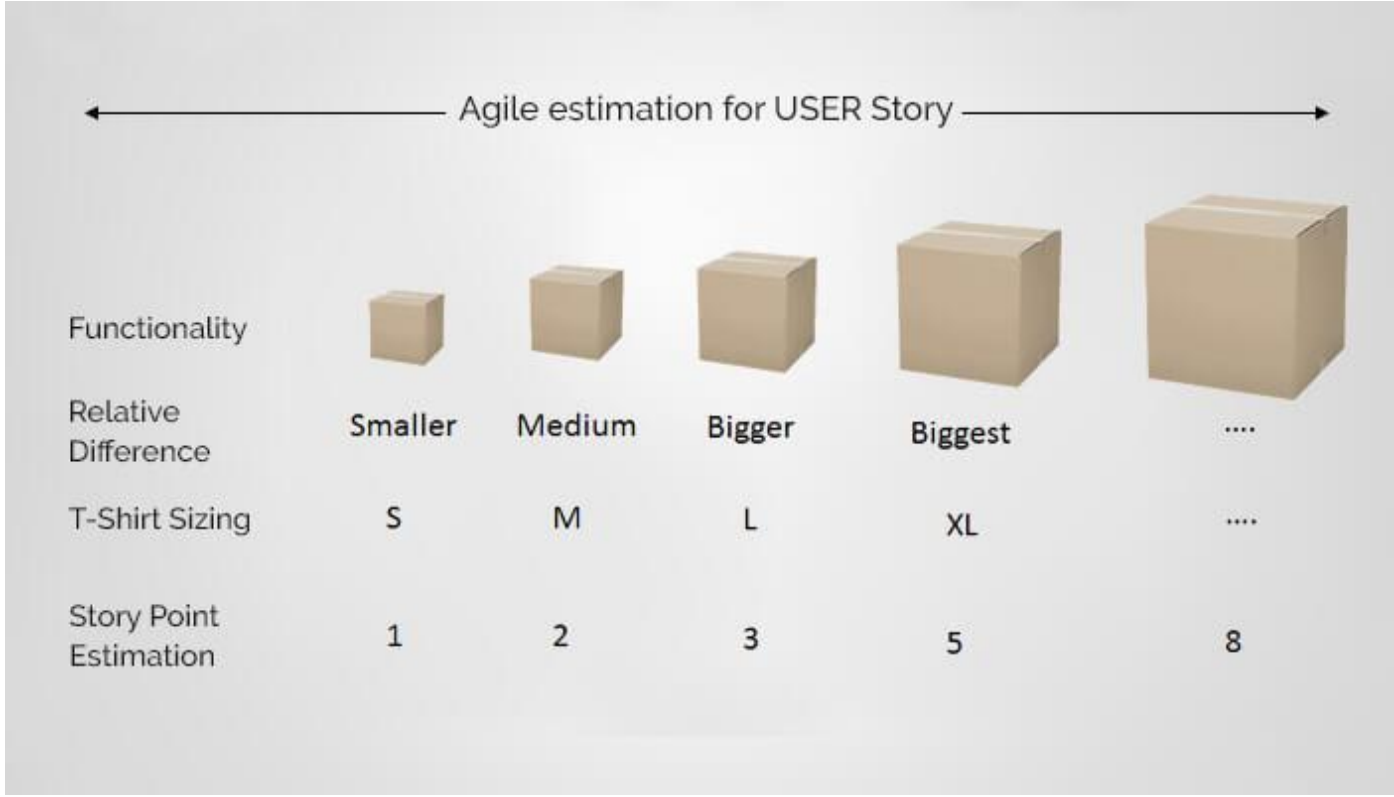
ABOUT STORIES

- “In order to <reason>, as <role> I want <something>”
- A user story describes functionality that will be valuable to either a user or purchaser of a system or software.
- From the end user’s perspective
- Just enough information
  - Card/Conversation/Confirmation
- I.N.V.E.S.T.



Planning Poker vs T-shirt size

- Relative estimate
- Total effort/complexity
- Whole team contributes



Planning Poker

Story points as the Size Range in terms of Fibonacci sequence

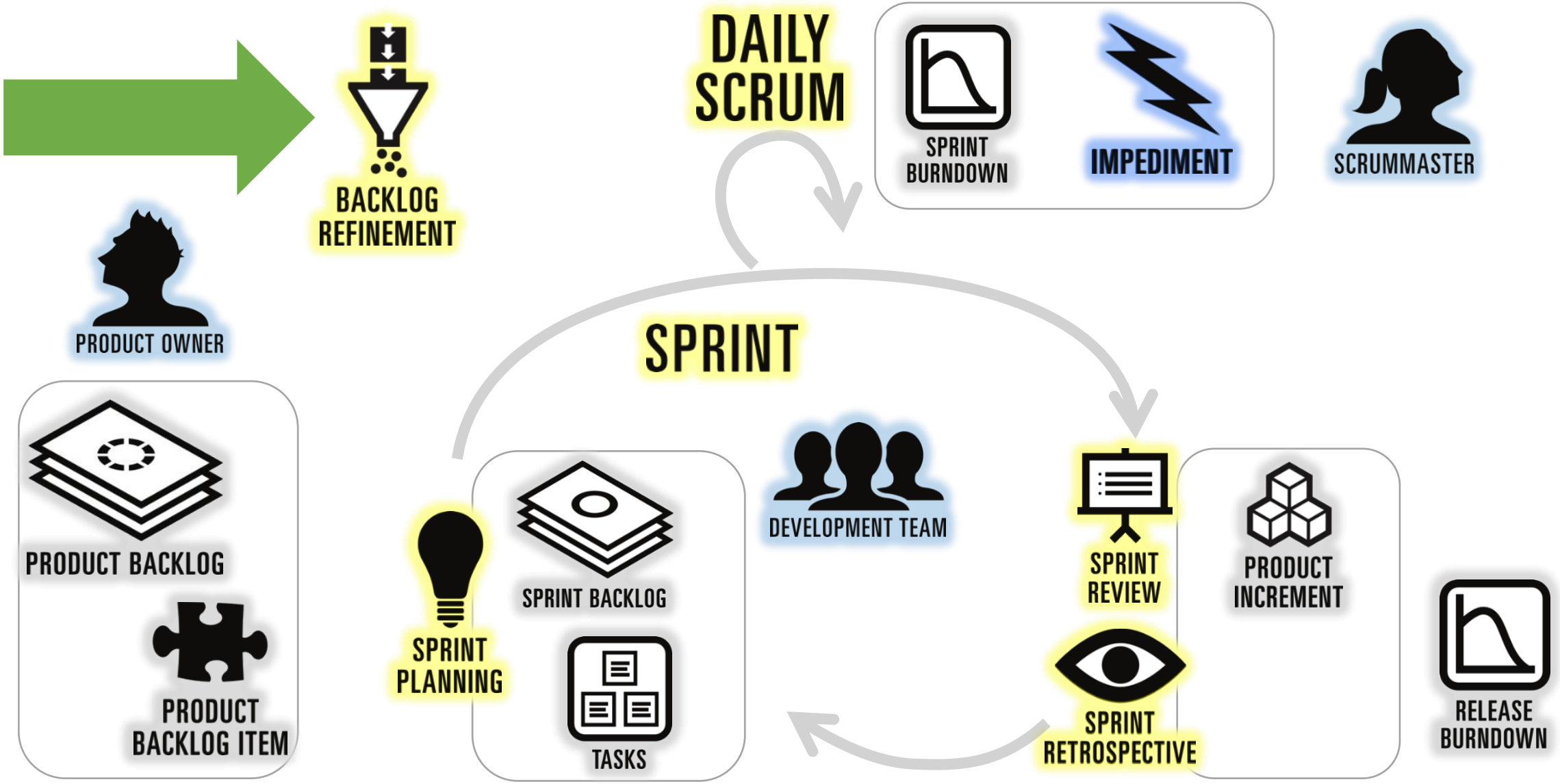
Story points – 1, 2, 3, 5, 8, 13, 20 , 40, …. 100

T-shirt size

Story points as the Size Range in terms of X-L

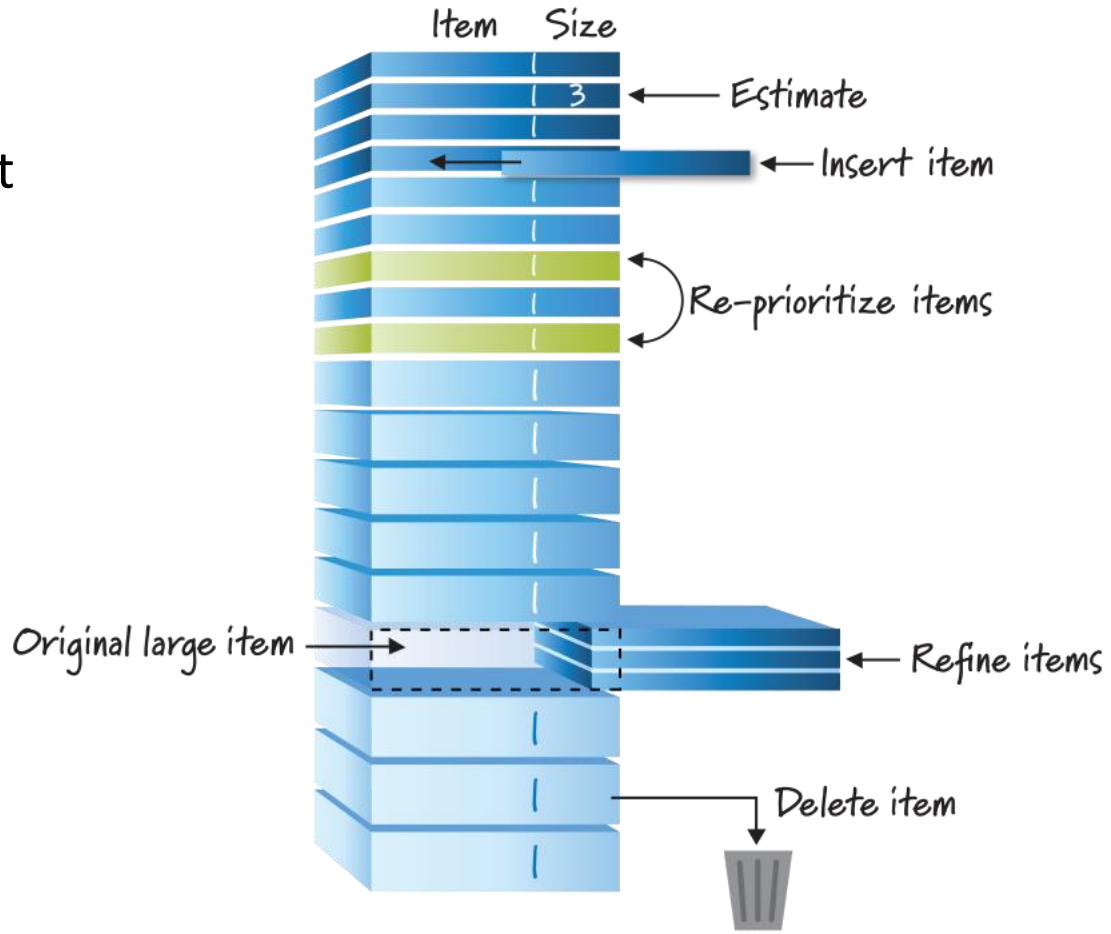
Story points – XS, S, M, L, XL, XXL, XXXL

BACKLOG REFINEMENT

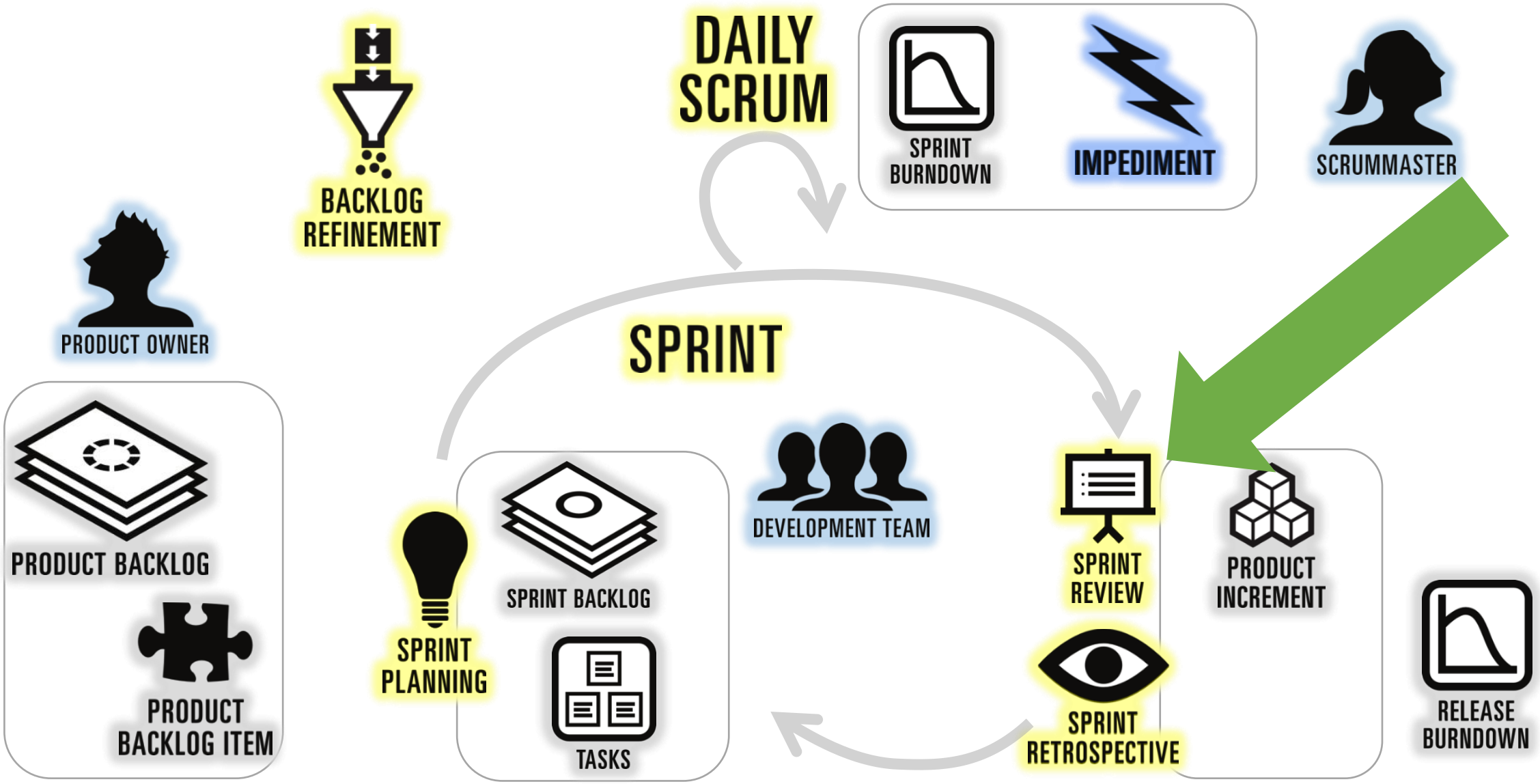


BACKLOG REFINEMENT

5-10% of each Sprint should be spent on refining the Product Backlog, with the Team and Product Owner. This is ideally done in a focused workshop. It is often done near the middle of each Sprint and at the end of it.



SPRINT REVIEW

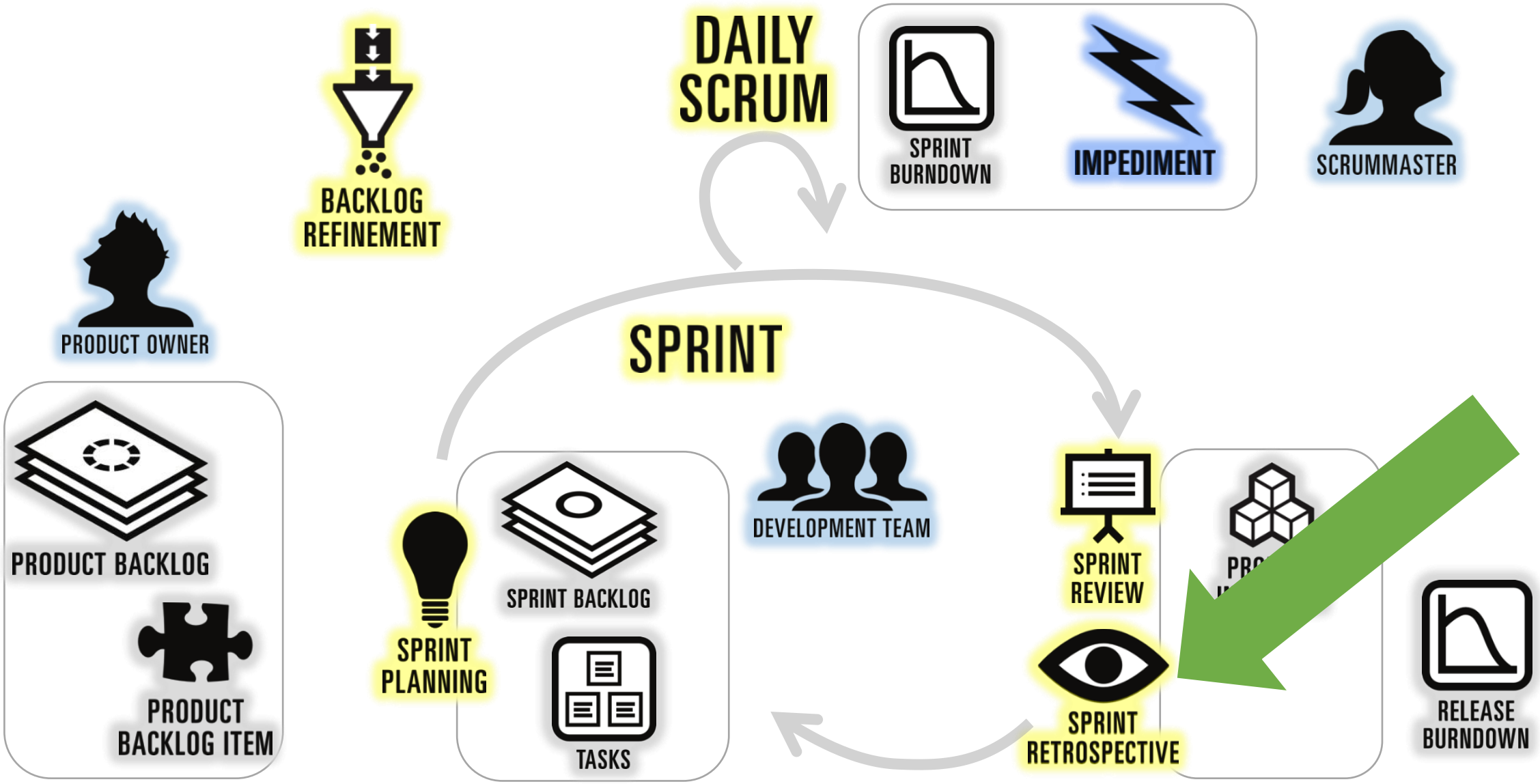


SPRINT REVIEW

- The delivered functionality is demonstrated in the demo meeting.
- Attendees include the Scrum Team and key stakeholders invited by the Product Owner
- Opportunity for the stakeholders to collaborate and provide feedback
- Opportunity for the team to build trust

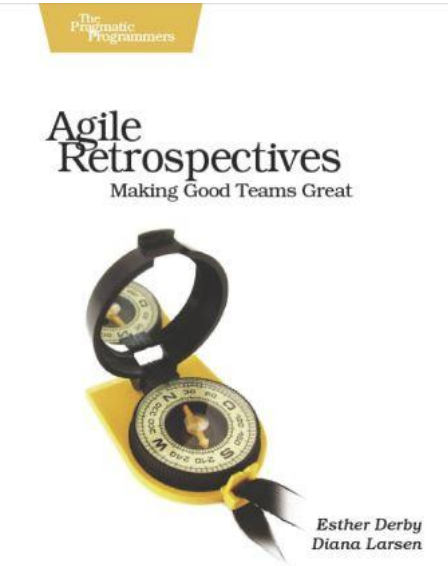
Sprint Review
Max 4h for 30d sprint

SPRINT RETROSPECTIVE



SPRINT RETROSPECTIVE

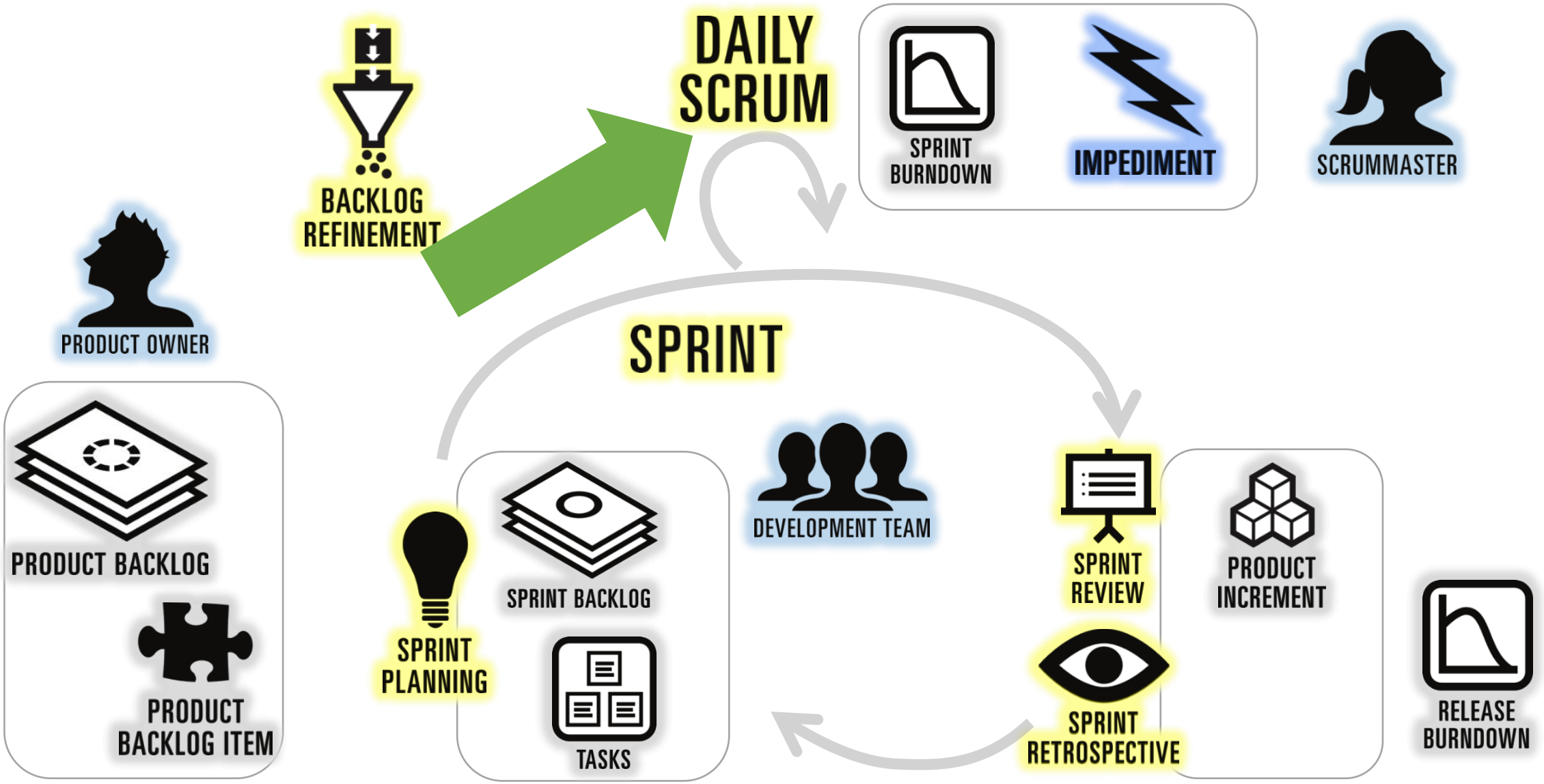
- Feedback about “how” you delivered
- What worked well
- What can be improved
- How can we improve
  - Generate insights
  - Identify actions



Sprint Retrospective
Max 3h for 30d sprint

Regardless of what we discover, we understand and truly believe that everyone did the best job they could, given what they knew at the time, their skills and abilities, the resources available, and the situation at hand – *Norm Kerth*

DAILY SCRUM


DAILY SCRUM
BACKLOG REFINEMENT
SPRINT BURNDOWN
IMPEDIMENT
SCRUMMASTER
PRODUCT OWNER
SPRINT
PRODUCT BACKLOG
PRODUCT BACKLOG ITEM
SPRINT PLANNING
SPRINT BACKLOG
TASKS
DEVELOPMENT TEAM
SPRINT REVIEW
PRODUCT INCREMENT
SPRINT RETROSPECTIVE
RELEASE BURNDOWN

Daily Scrum

A short meeting everyday – max 15 mins

We discuss 3 important things in daily stand up

1. What did I do yesterday?
2. What will I do today?
3. Are there any blockers/impediments?

Three Pillars

Transparency - To make decisions, people need visibility into the process and the current state of the product - Sprint Reviews Provide Transparency.

Inspection - To prevent deviation from the desired process or end product, we need to inspect what is being created, and how, at regular intervals - Sprint Reviews & Retrospectives Offer Inspection Opportunities.

Adaptation - the ability to make timely adjustments based on inspection results and changing circumstances. It involves responding effectively to feedback, addressing issues, and adapting plans and processes to optimize value delivery

SCRUM VALUES

Commitment - Scrum teams work together as a unit

Courage - Scrum teams must feel safe enough to say no, to ask for help, and to try new things

Focus - Focus means that whatever scrum teams start they finish--so agile teams are relentless about limiting the amount of work in process

Openness - Scrum teams consistently seek out new ideas and opportunities to learn

Respect - Scrum team members demonstrate respect to one another, to the product owner, to stakeholders