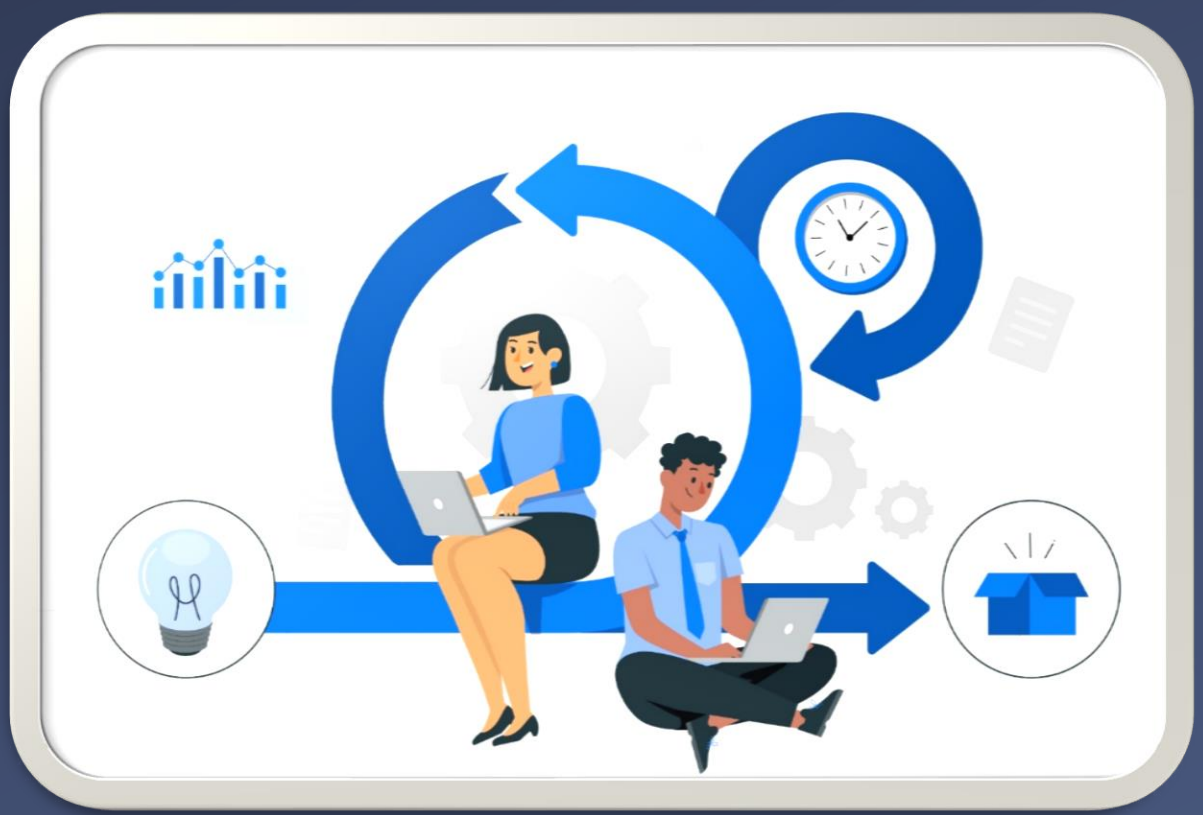


Scrum implementation and execution

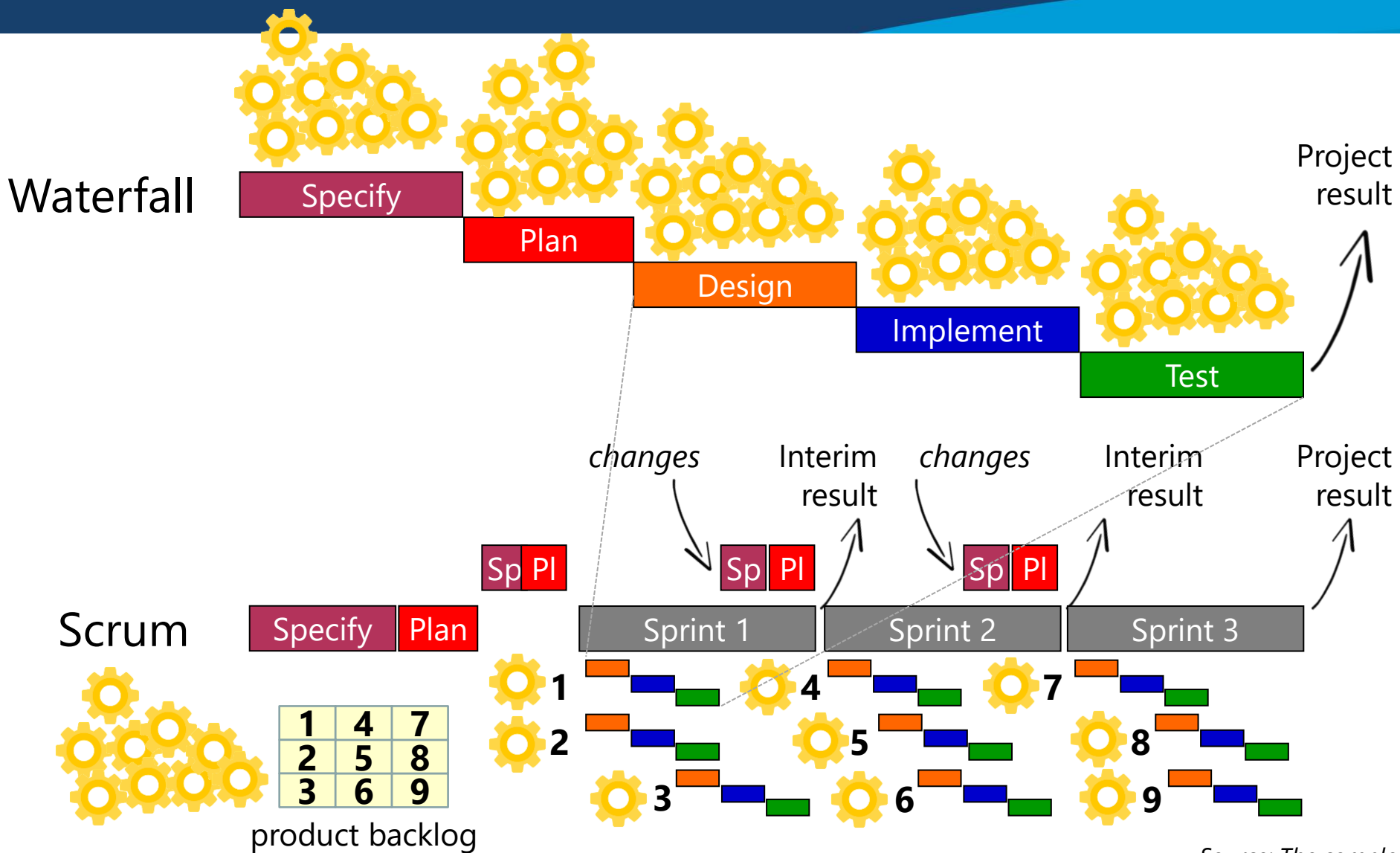


Scrum implementation and execution

- **Characteristics of Scrum**
- Team and organization
- Product backlog
- Sprint planning
- Sprint execution
- Reporting tools and metrics
- Multiple (multidisciplinary) teams



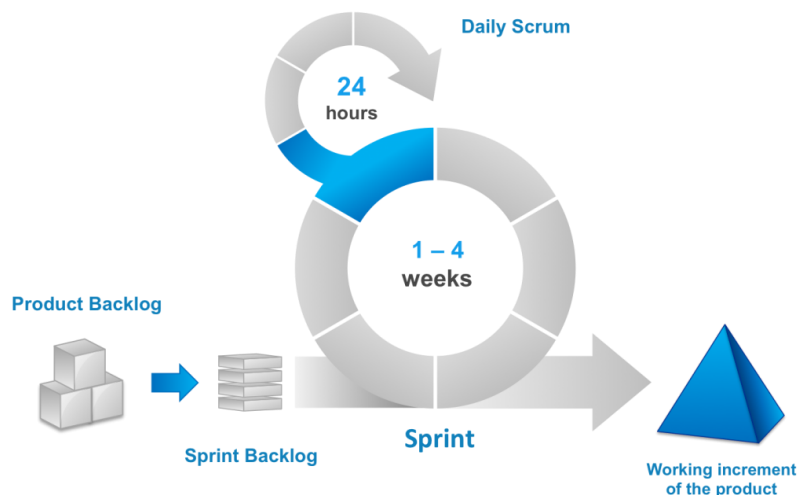
Traditional versus Scrum



Source: *The complete project manager, 2016*

Features Scrum summarized

- Deliver in sprints with fixed duration (timebox).
- Sprint finished: present results to end user in demo.
- Time, money and quality fixed, functionality negotiable.
- Functionality prioritized on product backlog.
- No changes in current sprint, changes OK for future sprints (flexibility ↔ efficiency).
- Cross-functional fixed (full-time) self-organizing Scrum team.
- Daily stand-up meeting (daily Scrum) for planning and coordination.



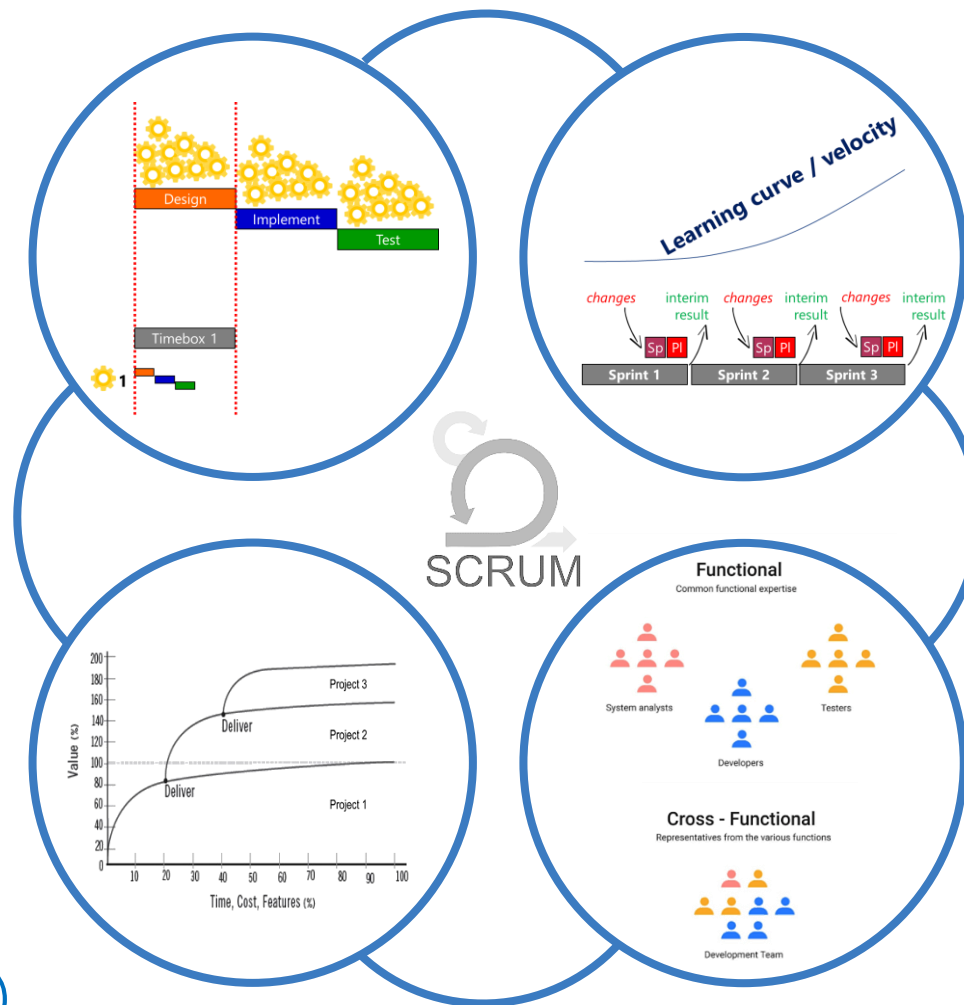
Scrum board (sprint status)

Backlog (user stories)	To Do (tasks)	In Progress (tasks)	Testing (tasks)	Done	KPI's
					Info

Essentials of Scrum

Instead of doing everything a little bit, doing some of it completely.

Deliver value earlier:
earlier feedback and
business revenue
(80/20 Pareto Principle, MVP).



Step by step (per sprint)
learning and adapting.

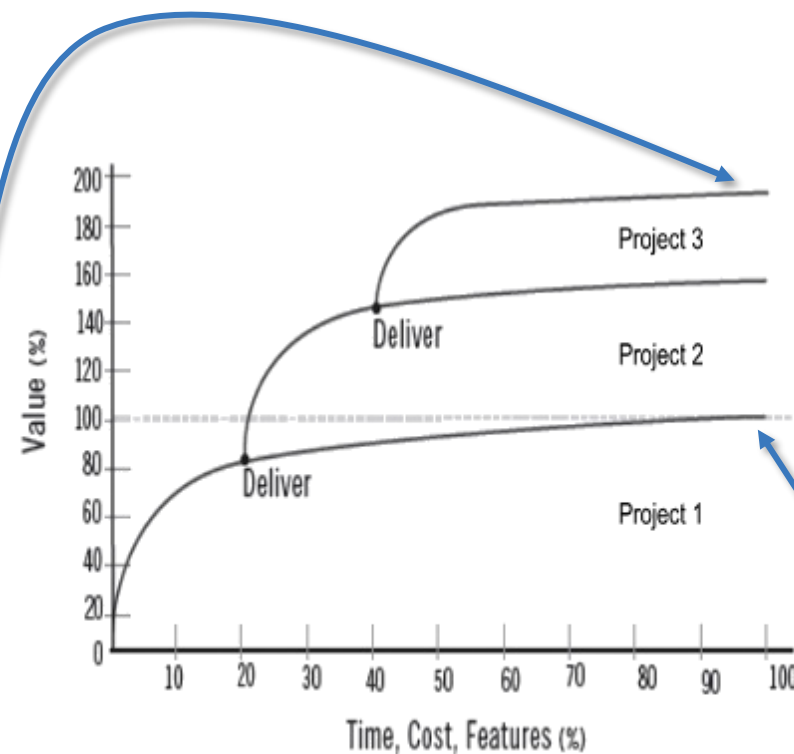
Multidisciplinary
collaboration
(cross-functional team)

Source: PULZ, 2022

Minimum Viable Product (MVP)

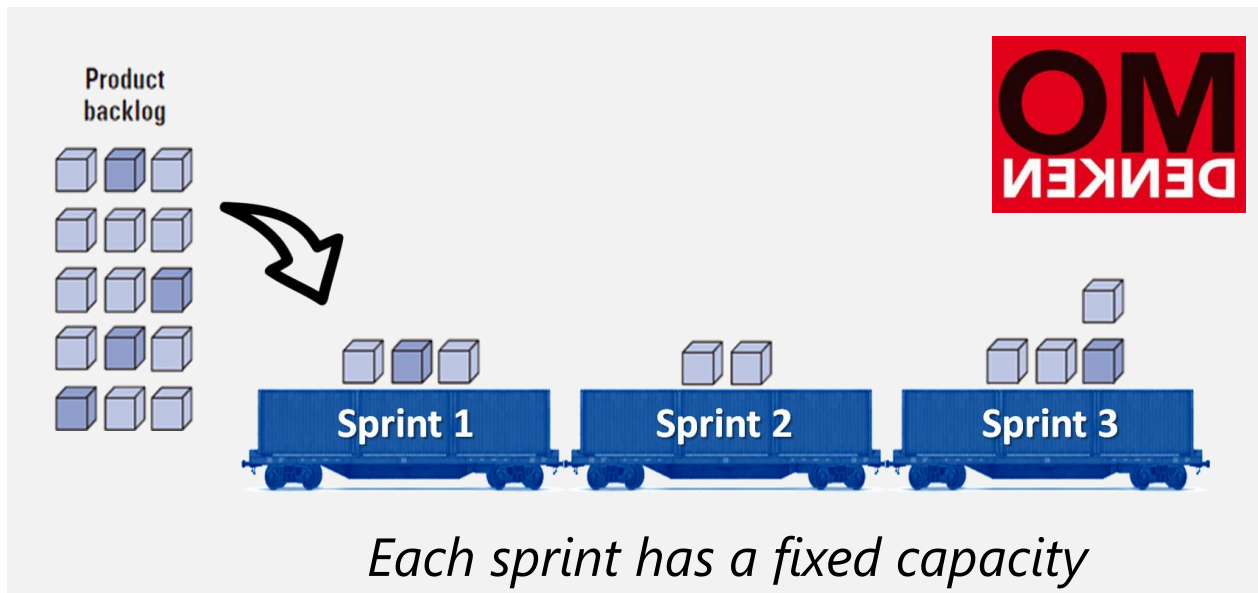
The **MVP**, or **Minimum Viable Product**, is the simplest version of your product that allows you to provide value to the user. Building a simple but well-functioning version of your final product ensures that you can gain experience in the outside world as early as possible, serve the customer sooner (and also have them gain experience) and start making money already.

The MVP is based on the **Pareto principle**, or **the 80/20 rule**. The Pareto principle states, that 20% of the effort creates 80% of the results. By applying this to subsequent releases or projects, you create much more value, than if you wait to launch until your product has all the functionalities.



Source: PULZ, 2022

Managing capacity differently with Scrum



*Bring the work to the team,
rather than the other way around.*

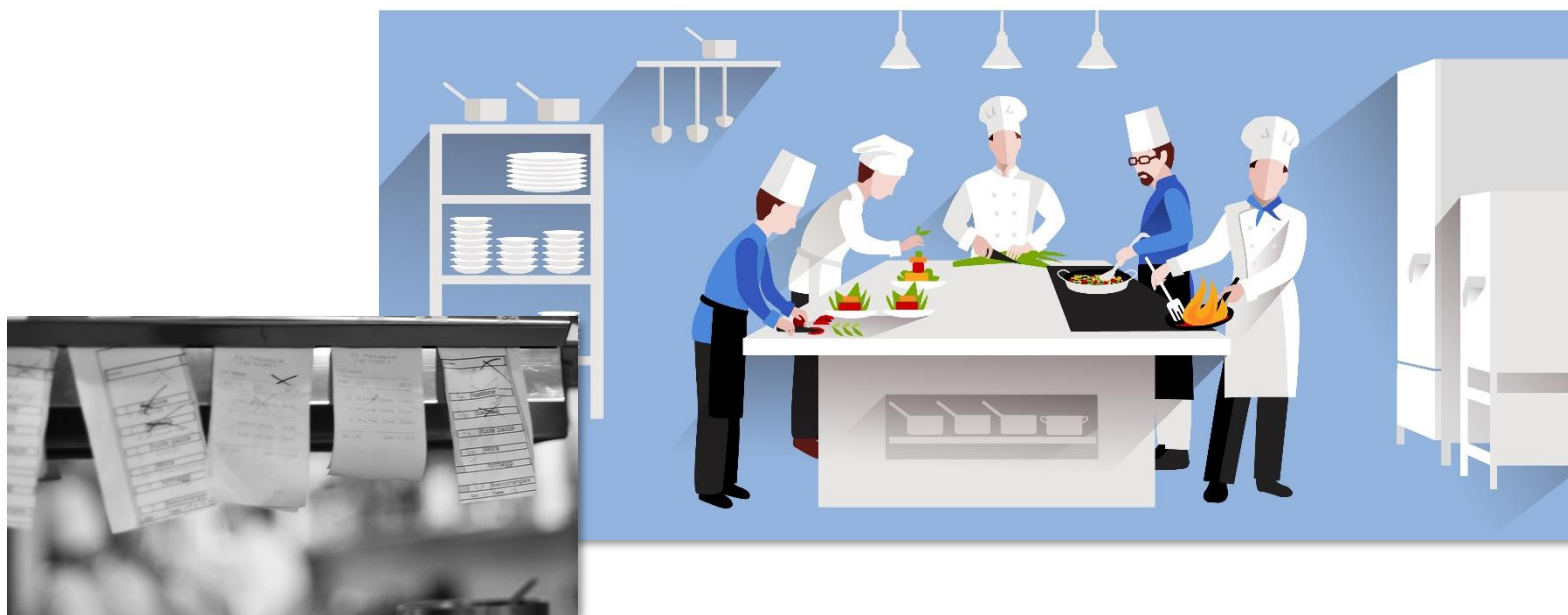
Using Scrum differently dealing with capacity

Instead of seeking capacity per activity and setting up agreements separately: doing the maximum with an available fixed team in a fixed period of time.

Scrum team and organisation



Perhaps a useful and illustrative description of a Scrum team is the way a **kitchen team** works **in a restaurant**. There are **multiple** customers and multiple waiters and waitresses, the various orders come in order on **a single order board** (product backlog). The **self-organizing** kitchen team organizes its own work as efficiently as possible, while being **shielded** from the customers.



Source: PULZ, 2022

So doing Scrum is more than 'working with yellow notes'

Doing **Scrum** is doing the maximum with an available (fixed) capacity, where you accelerate per sprint by learning and removing obstacles.

So at a time when we had come to see everything as (unique) projects, we are basically just going back to *'working in routine factory mode'* using production capacity to the fullest.



Using Scrum to organize instead of plan

► Verlichting voor bomvol traject Eindhoven-Amsterdam

Ten-minute train will start running as of September

Tien minuten een trein tussen Eindhoven en Amsterdam. Het start op 6 september, eerst alleen op woensdag, als test, en vanaf december op elke dag.

Marcia Nieuwenhuis

Utrecht

Treinreizigers kunnen vanaf 6 september kennismaken met de langverwachte tienminutendienstregeling. De NS en ProRail gaan dan testen met elke tien minuten een trein tussen Eindhoven, Den Bosch, Utrecht en Amsterdam. NS-topman Roger van Boxtel sluit niet uit dat andere treinen daardoor vertraging oplopen.

Veertien woensdagen lang gaan de Nederlandse Spoorwegen en spoorbeheerder ProRail met reizigers testen of het haalbaar is om op dit traject zes keer per uur een intercity te laten rijden. Dat betekent dat reizigers tussen deze steden van 's ochtends vroeg tot 's avonds laat elke tien minuten een intercity kunnen nemen.

noemde 'doorboekloos rijden' zoals ook nu gebeurt. Het is belangrijk om de test in de praktijk te doen, mét reizigers, om te kunnen zien hoe dit in de dagelijkse praktijk gaat en hoe we dit stapsgewijs kunnen verbeteren." Mogelijk ontstaat er juist vertraging op de testdagen. Van Boxtel: „Vaker vertraging kan ik niet uitsluiten: een test is om te testen.” ProRail-topman Pier Eringa beaamt dat: „Als de tienminutentrein rijdt is het een feest, maar als je al snel veel problemen. Dan ligt de treinsnelweg plat.”

De dienstregeling op de overige trajecten blijft her-

overstaptijden tijdens de testperiode anders dan straks in de dienstregeling. Vanaf het begin staan de nieuwe tijden in de planner. NS en ProRail zijn al meer dan tien jaar bezig met de voorbereidingen voor elke tien minuten op de drukste trajecten van Nederland. Treindienstleiders, managers en conducteurs kregen er voor te zorgen dat reizigers sneller in- en uitstappen en dat de instapzones op perrons. Eind dit jaar volgend een voorstel aan de po-

—Pier Eringa, topman ProRail



FOTO JERRY LAMPEN/ANP

organizing without a timetable

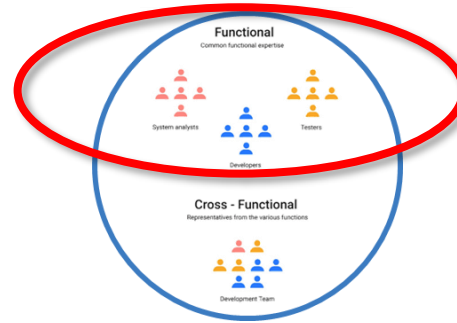
With Scrum
'organizing instead of planning'

If the bus runs once an hour, you look up what time you have to leave home and delays bother you.
If the bus runs every 10 minutes, then you leave 'unscheduled' and delays are less of an issue.

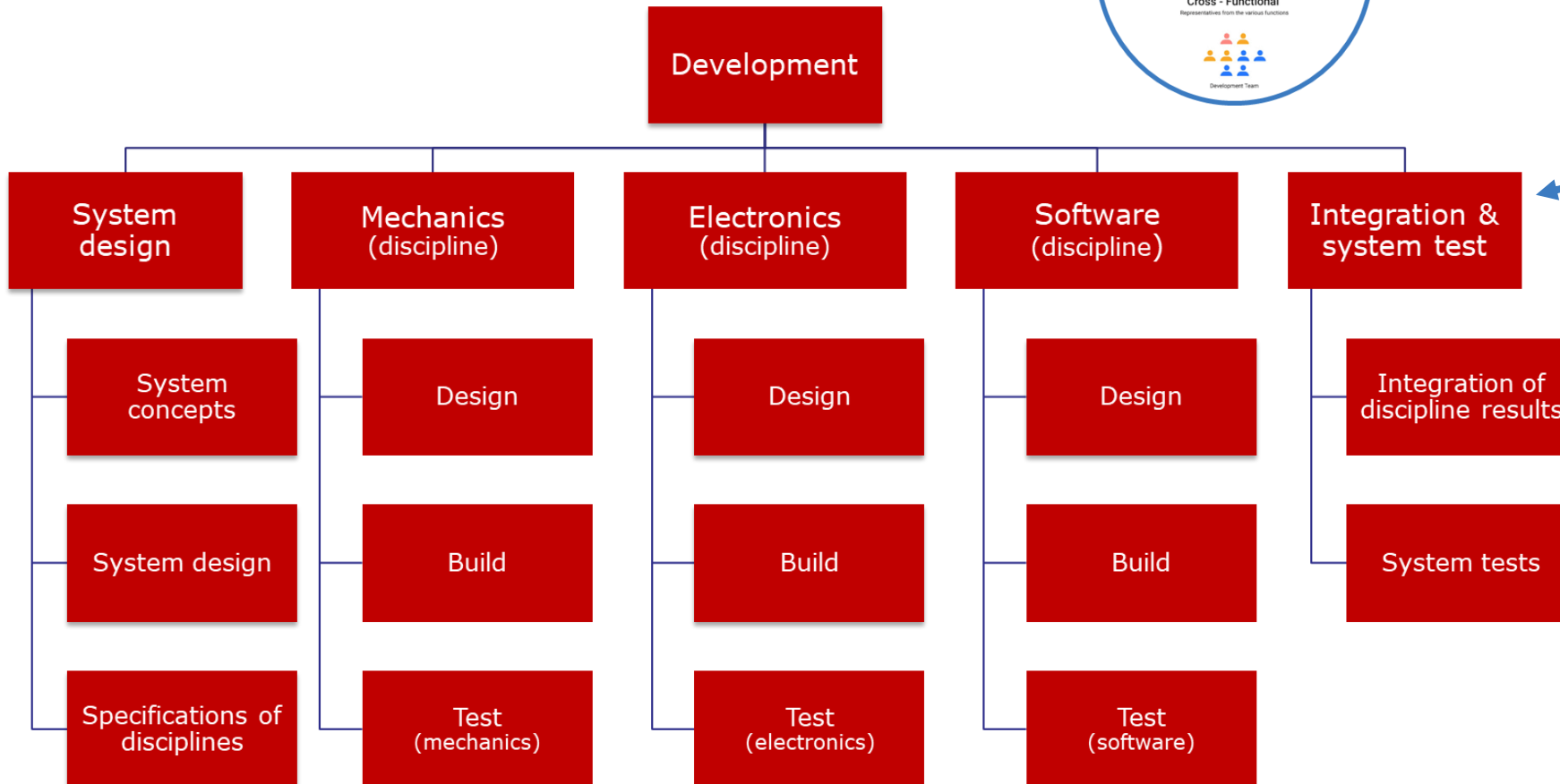
Source: PULZ, 2022

Multidisciplinary teams were not invented by Scrum 😊 (1/2)

Organization set up as monodisciplinary sub-teams that deliver work of a single discipline.



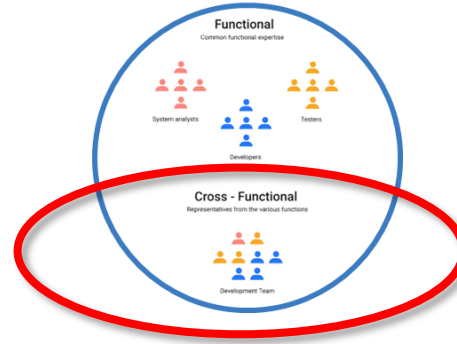
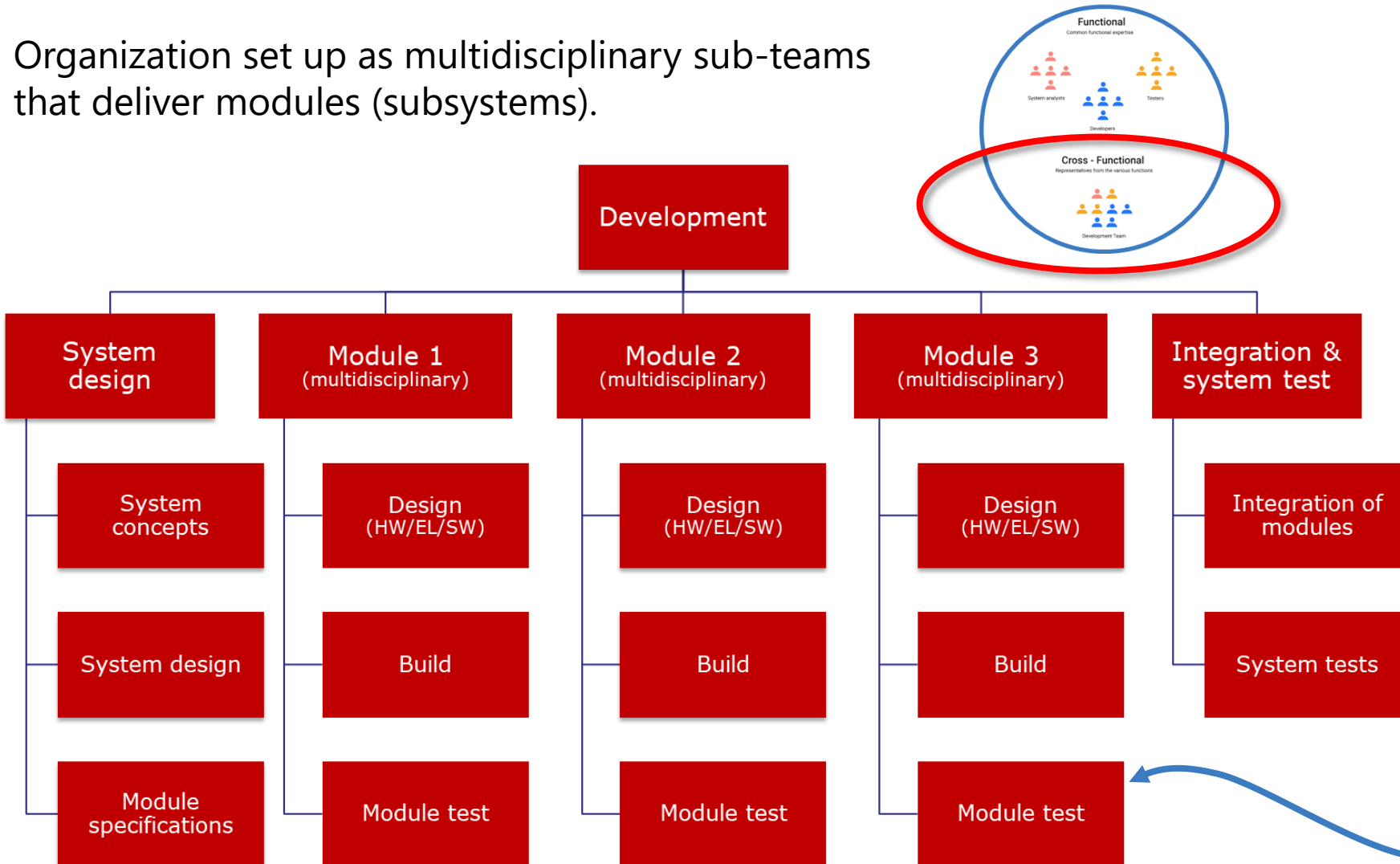
In Scrum, the team should deliver testable results per sprint (preferably at the system or end-user level). This makes perfect sense and logically requires a **multidisciplinary** team. *By the way, you can also implement that without Scrum!*



👉 This breakdown, however, organizes in **mono-disciplinary** teams, so *integration* will take place quite late.

Multidisciplinary teams were not invented by Scrum 😊 (2/2)

Organization set up as multidisciplinary sub-teams that deliver modules (subsystems).



In Scrum, the team should deliver testable results per sprint (preferably at the system or end-user level). This makes perfect sense and logically requires a **multidisciplinary** team. *By the way, you can also implement that without Scrum!*

➡ And this breakdown organizes into **multidisciplinary** teams, so that *integration at the module level* takes place earlier.

Source: PULZ, 2022

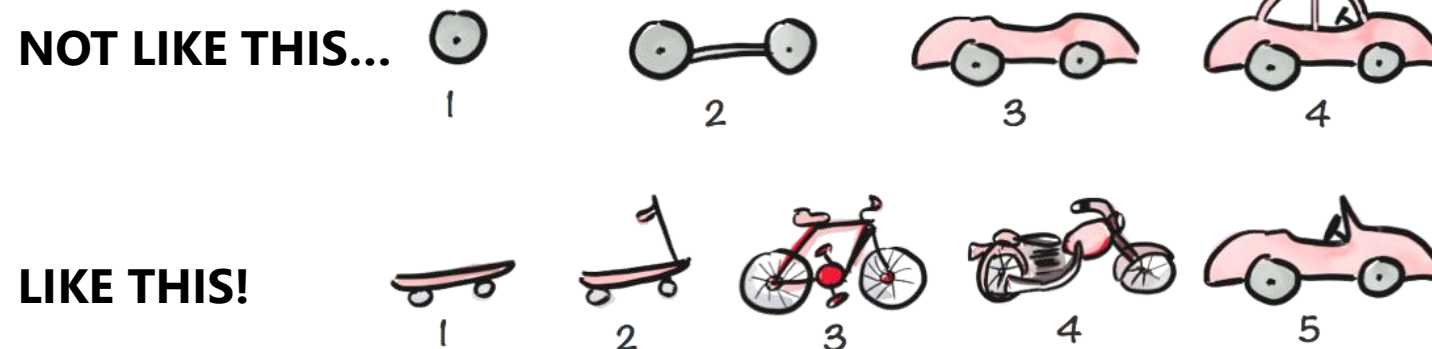
Incremental development



Bear in mind that incremental design in an Agile working method is not the same as building in technical increments. For each step (MVP) it is desirable that the **intermediate results can be used by the (end) user and already add value!**

This requires a different approach to project phasing and marketing.

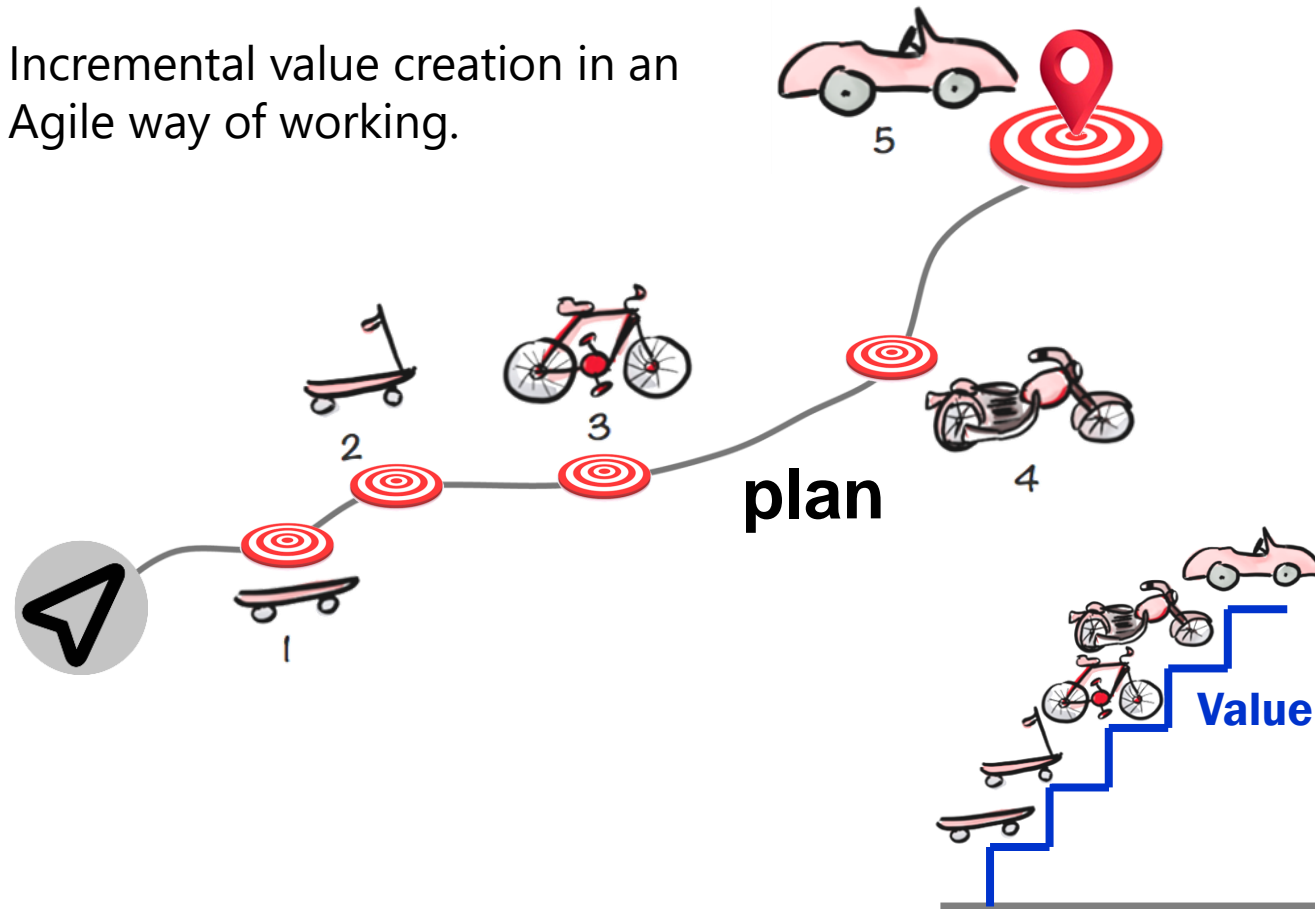
Henrik Kniberg aptly illustrates that in this picture:



Source: Kniberg, 2014

Incremental development

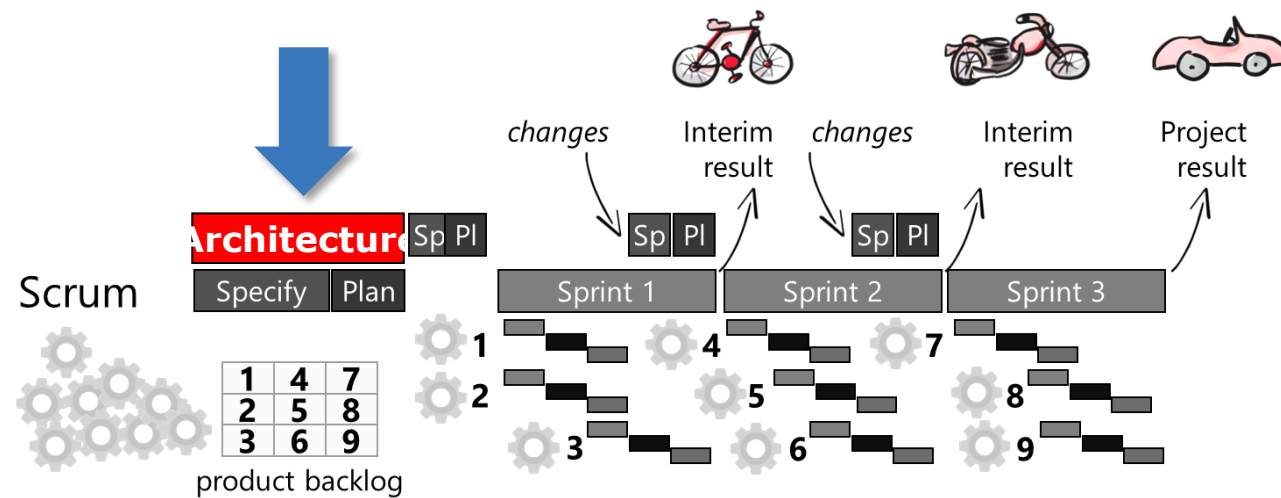
Incremental value creation in an Agile way of working.



Source: PULZ, 2022

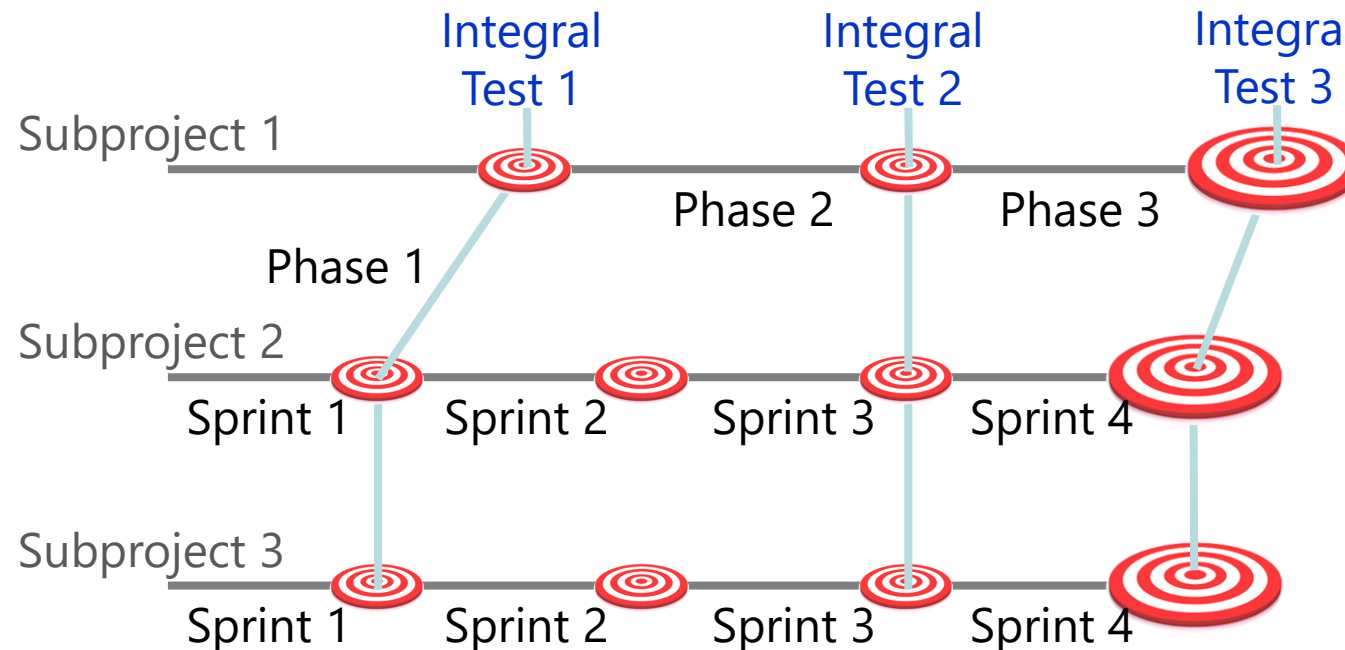
Incremental development

Incremental development does not come naturally. Prior to the execution phase, the **architecture/infrastructure** must be set up so that the intermediate results can actually be tested and thus be **operational/installable/secure/understandable/etcetera**.



Incremental development

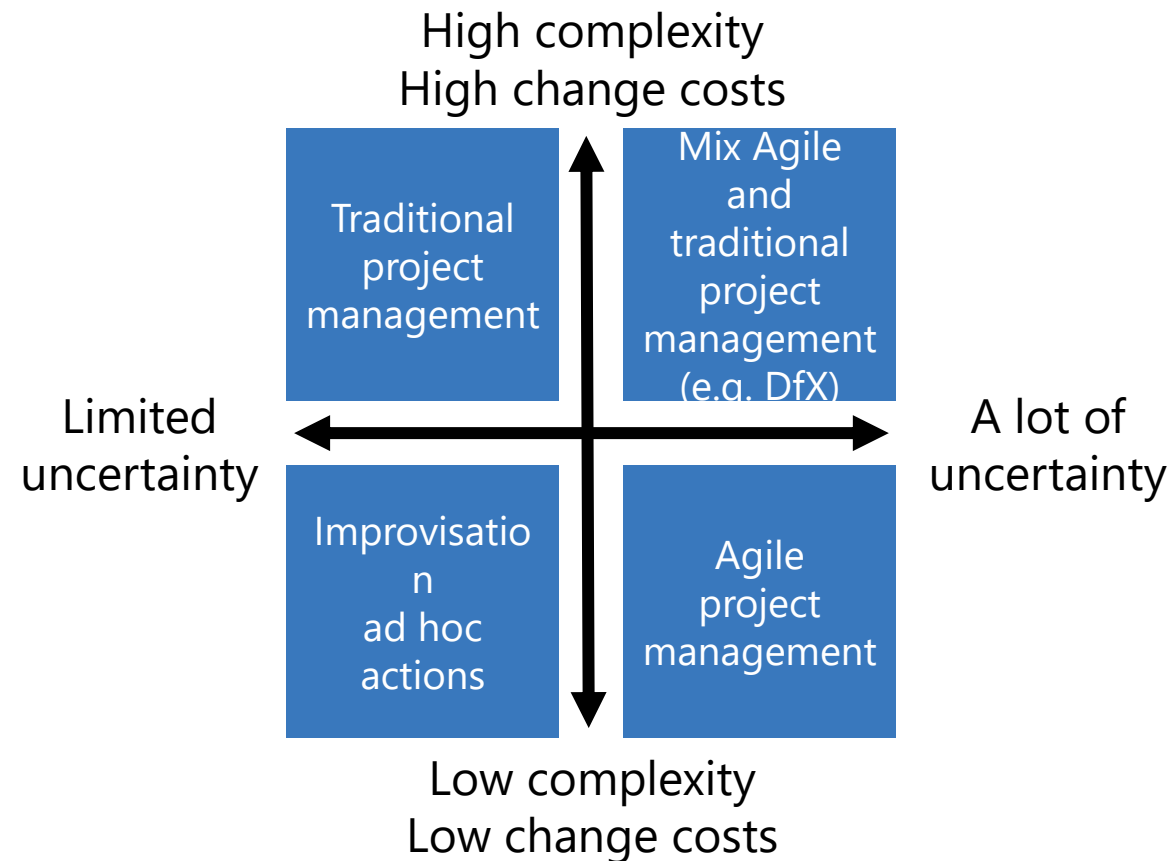
Moreover, it is important that the incremental sprint results **are aligned with the other (sub)projects** in terms of content. **This way**, there is integral value creation and risk reduction and you avoid big-bang integration. By the way, the other subprojects do not have to be Agile projects.



See also the *Multi-disciplinary Heartbeat* at the end of this in-depth block.

Incremental development

A high number of iterations is not always convenient. **The number of iterations** and **associated change costs** should be **balanced**. One of the reasons why you will always encounter traditional techniques in addition to Agile project management.



Source: PULZ, 2022

Scrum implementation and execution

- Characteristics of Scrum
- **Team and organization**
- Product backlog
- Sprint planning
- Sprint execution
- Reporting tools and metrics
- Multiple (multidisciplinary) teams



Scrum team (roles)



SCRUM TEAM



Scrum team (roles)

Scrum describes two main roles in addition to the **cross-functional team**: the **product owner** and the **scrum master**.

The **product owner** manages and prioritizes the **product backlog**. In doing so, he/she represents the customer's wishes and must have the right mandate from the client. The product owner is explicitly part of the team, ensuring that the voice of the customer is integrated into the daily communication (via the daily stand-up meeting). So the product owner spends about 50% of the time with the team (explaining functions, managing backlog) and 50% of the time with the customer (understanding features/requirements and communication between project and customer).

The team and the product owner are supported by the **scrum master**. This is a different role than the project manager, as the scrum master does not direct the team, but acts **as a coach and facilitator**; in an Agile environment, the development team needs to organize itself to efficiently achieve the assigned goals.

Source: The complete project manager, 2016

Scrum team (roles)



Product owner accountabilities

- One person, not a committee.
- Managing and prioritizing the product backlog.
- Answering questions from the developers.
- Maximizing the value of the product by creating a visible and well understood product backlog.
- Representing the needs of many stakeholders.
- Creating and communicating clear product backlog items and a product goal.



Scrum master accountabilities

- Accountable for the Scrum Team's effectiveness.
- Ensuring that the rules of Scrum are respected.
- Facilitating the removal of obstacles.
- Leading, training, and coaching the organization in its Scrum adoption.
- Coaching the team to become more self-organizing and cross-functional.



Developers accountabilities

- Creating a plan for the sprint backlog.
- Delivering quality by adhering to a Definition of Done. *No blind checks!*
- Working to deliver a high quality and functional increment each sprint (preferably at system level).
- Holding each other accountable as professionals.
- Be open, transparent and adaptable to achieve the sprint goal.

Source: PULZ, 2022

Scrum team and organization



As already illustrated by the fundamentals, Agile/Scrum provides a wonderful framework for the team for the purpose of **self-organization** through the prioritized product backlog, regular feedback based on intermediate results, daily stand-up meetings and clarity on the trade-offs of time, money, quality and functionality.

The team is **supported and challenged** to determine its own mode of implementation and to monitor the progress of the work. Hence the **facilitating role** of the scrum master.

So the **project manager** has a **different role** than in a traditional environment. It is sometimes argued that the project manager is no longer needed in an Agile organization, but that only works in small projects where the team communicates directly with the client via the product owner and/or scrum master. **If the project organization becomes more complex**, there is a need for a role that manages capacity, stakeholders, interaction of multiple product owners, synchronisation of subprojects and setting up the technical architecture.

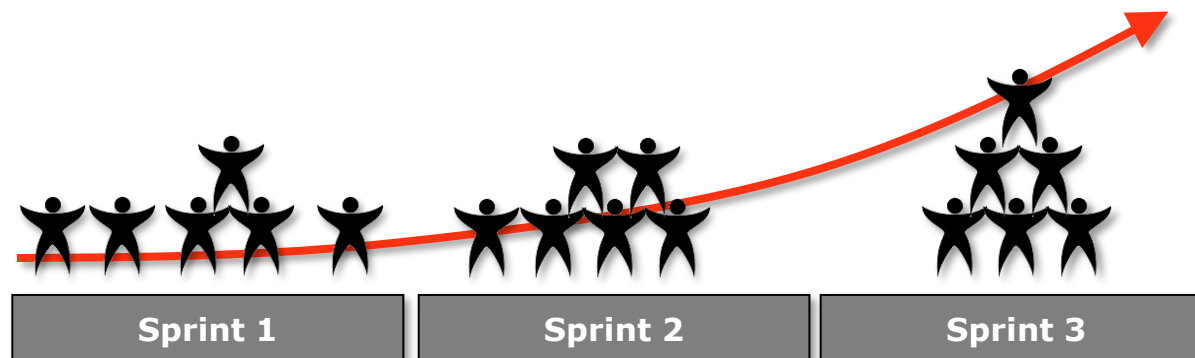
Then again, that is often a 'project management-type' role...

Scrum team and organization



Make sure the team contains all the disciplines to get the job done and keep it together at all costs!

Agile means flexible, but that is **not without any obligations**. For instance, it is desirable that the team can produce the result independently (and is therefore not mono-disciplinary) and that the learning effect is guaranteed by not changing the team after delivering results of a sprint. In organizational terms, it is sometimes quite a task to put this into practice.



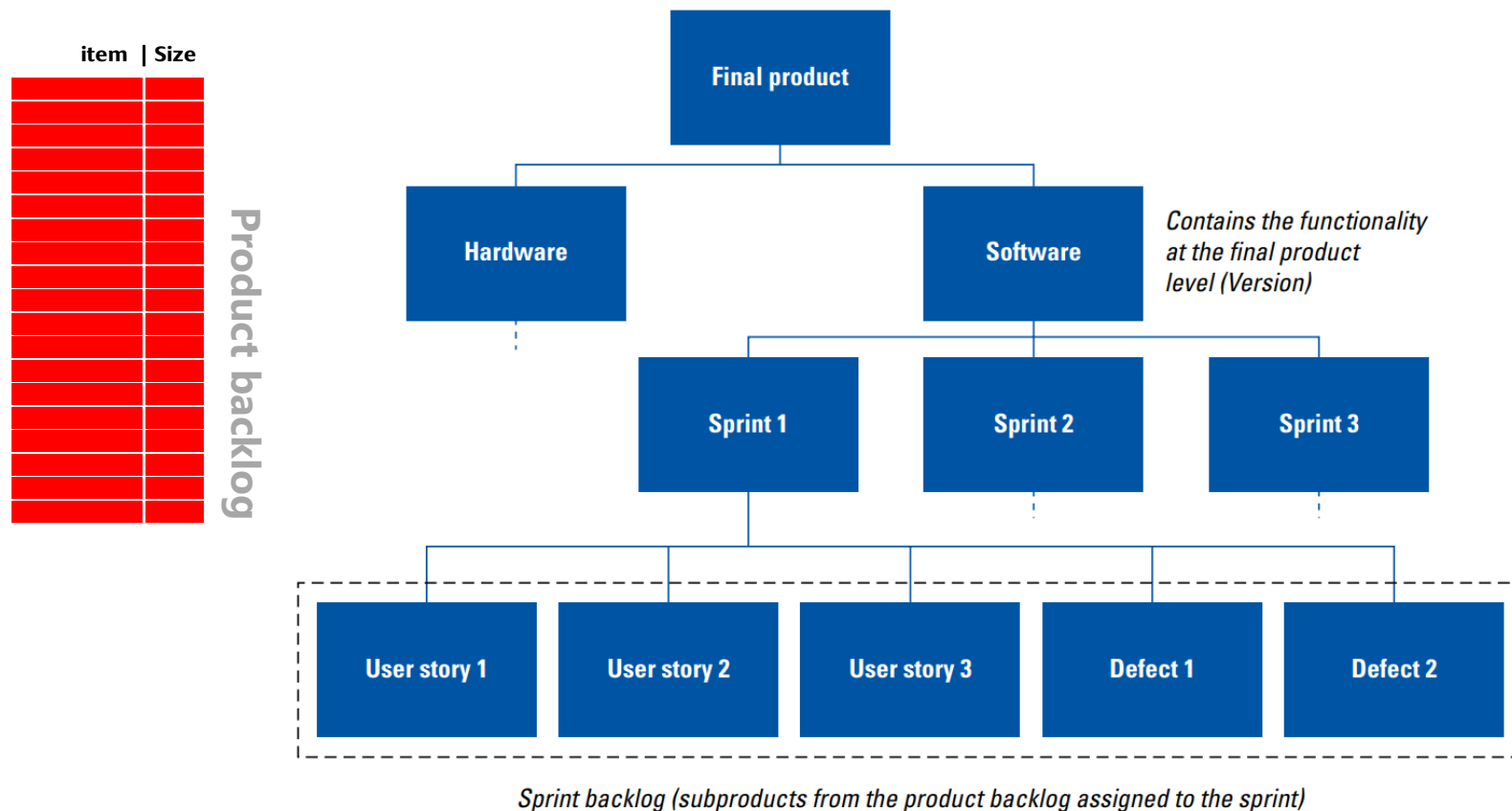
Scrum implementation and execution

- Characteristics of Scrum
- Team and organization
- **Product backlog**
- Sprint planning
- Sprint execution
- Reporting tools and metrics
- Multiple (multidisciplinary) teams



User story and product backlog

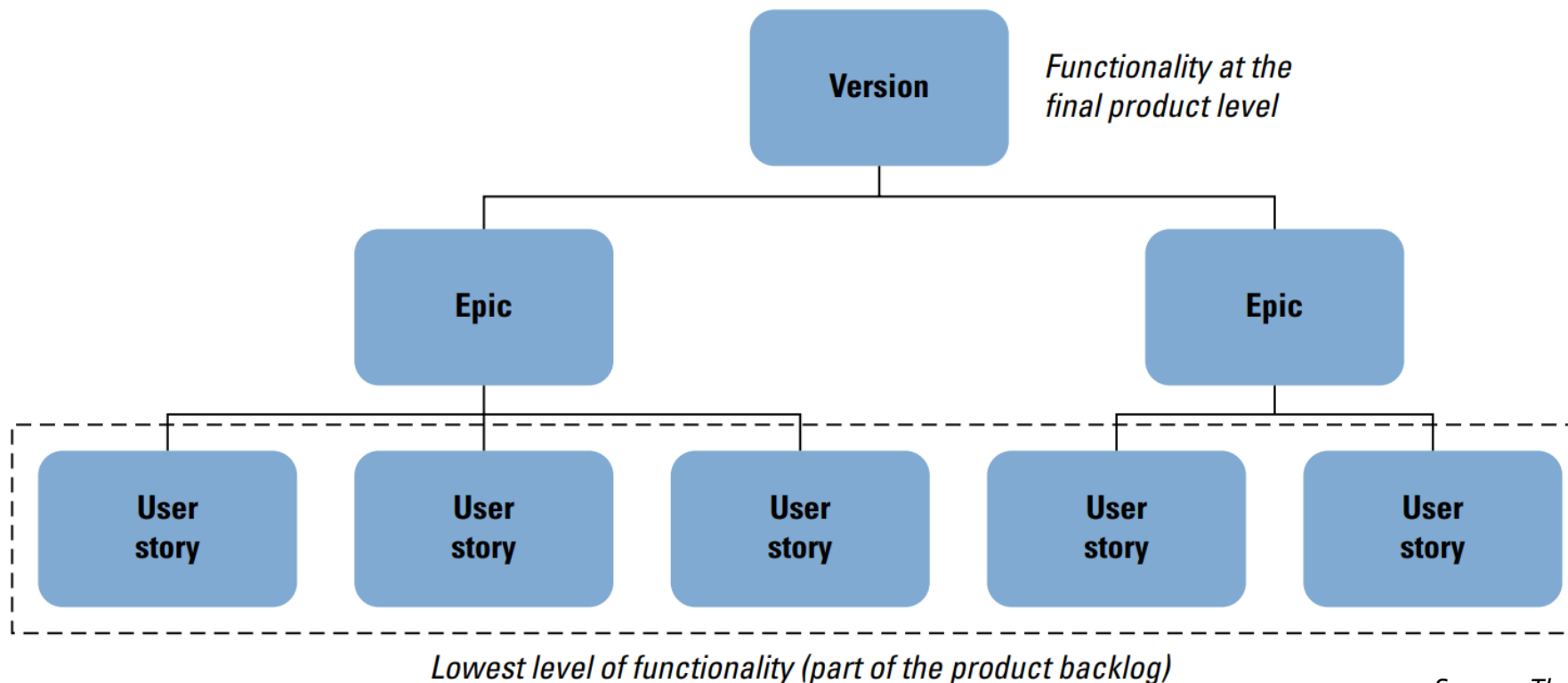
So you can consider the user stories on the product backlog as the smallest (bottom) elements of the PBS.



Source: *The complete project manager, 2016*

User story and product backlog

User stories are also the lowest level of the **requirements breakdown** (the specification structure) of the product. At a higher level, they combine into **epics**. Epics are functionalities that are of interest to the stakeholders and which are usually delivered during a sequence of multiple sprints. In Scrum, there is another level above the epics, known as **versions**. These are the official releases delivered to the client.



Source: *The complete project manager, 2016*

Scrum implementation and execution

- Characteristics of Scrum
- Team and organization
- Product backlog
- **Sprint planning**
- Sprint execution
- Reporting tools and metrics
- Multiple (multidisciplinary) teams



Size of user stories: story points

Because the duration of a Scrum sprint is set, the lead time per sprint is known in advance, as is the available effort from the team (depending on the number of team members, of course). For that reason, there is no need to determine how much effort the implementation of a user story will take when developing your plan; it is enough to know **how many user stories you can fit into a sprint**. You can find this out by determining **the size of the user stories** (and the capacity of a sprint).

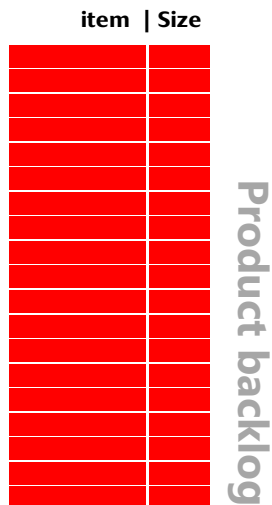
This size is related to:

- the **amount of** work
- the **complexity** of the work
- the **risks and uncertainties** associated with the work

The size of the user stories is usually expressed in **story points**. A story point is an abstract and relative unit with which to relate the size of the user story to a known piece of work. This means story points are not directly related to hours and are **dependent on the team**. The main goal is to “fill” the sprints.

Velocity is the amount of work the team has proven it can deliver in one sprint and is also measured in story points.

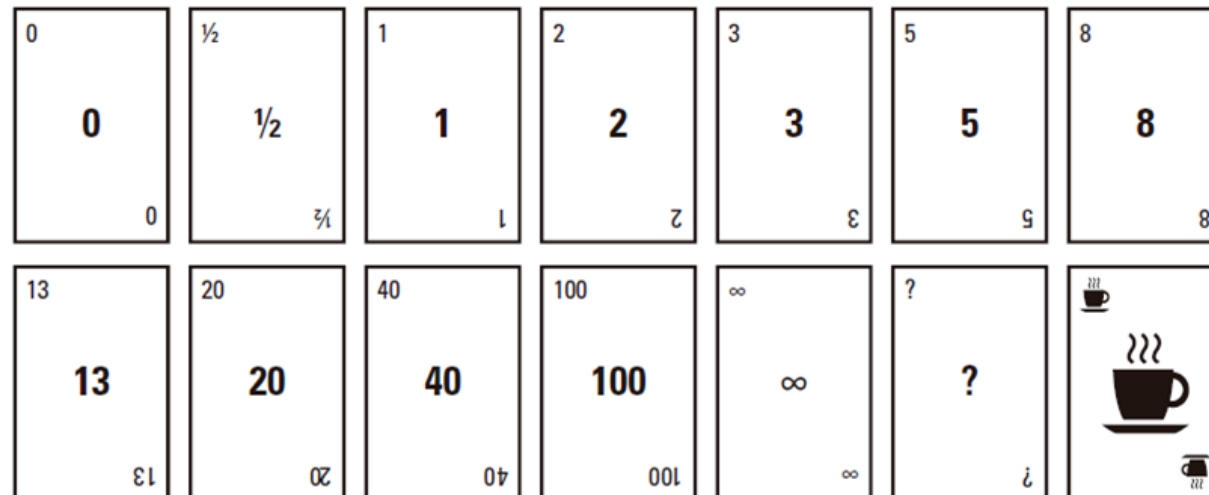
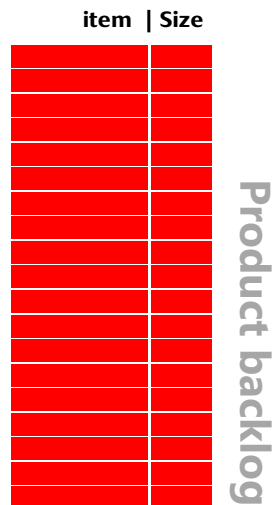
Source: PULZ, 2022



Size of user stories: Planning Poker

Estimating is done with **Planning Poker**. With this, during project initiation, the team estimates the number of story points per user story in a game-like manner. This is done with a set of cards or with software/apps.

The cards show the following numbers: 0,0.5, 1, 2, 3, 5, 8, 13, 20, 40 en 100. These numbers are based on the **Fibonacci sequence**, which represents the fact that uncertainty grows in proportion to the size of the user story.

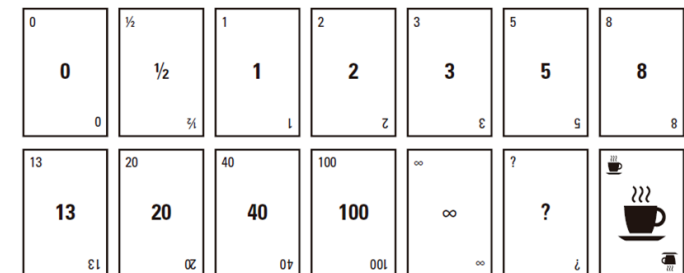
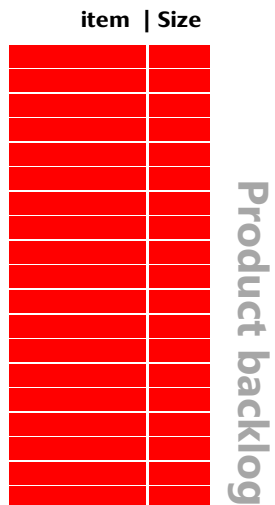


Source: *The complete project manager, 2016*

Size of user stories: Planning Poker

The implementation of Planning Poker:

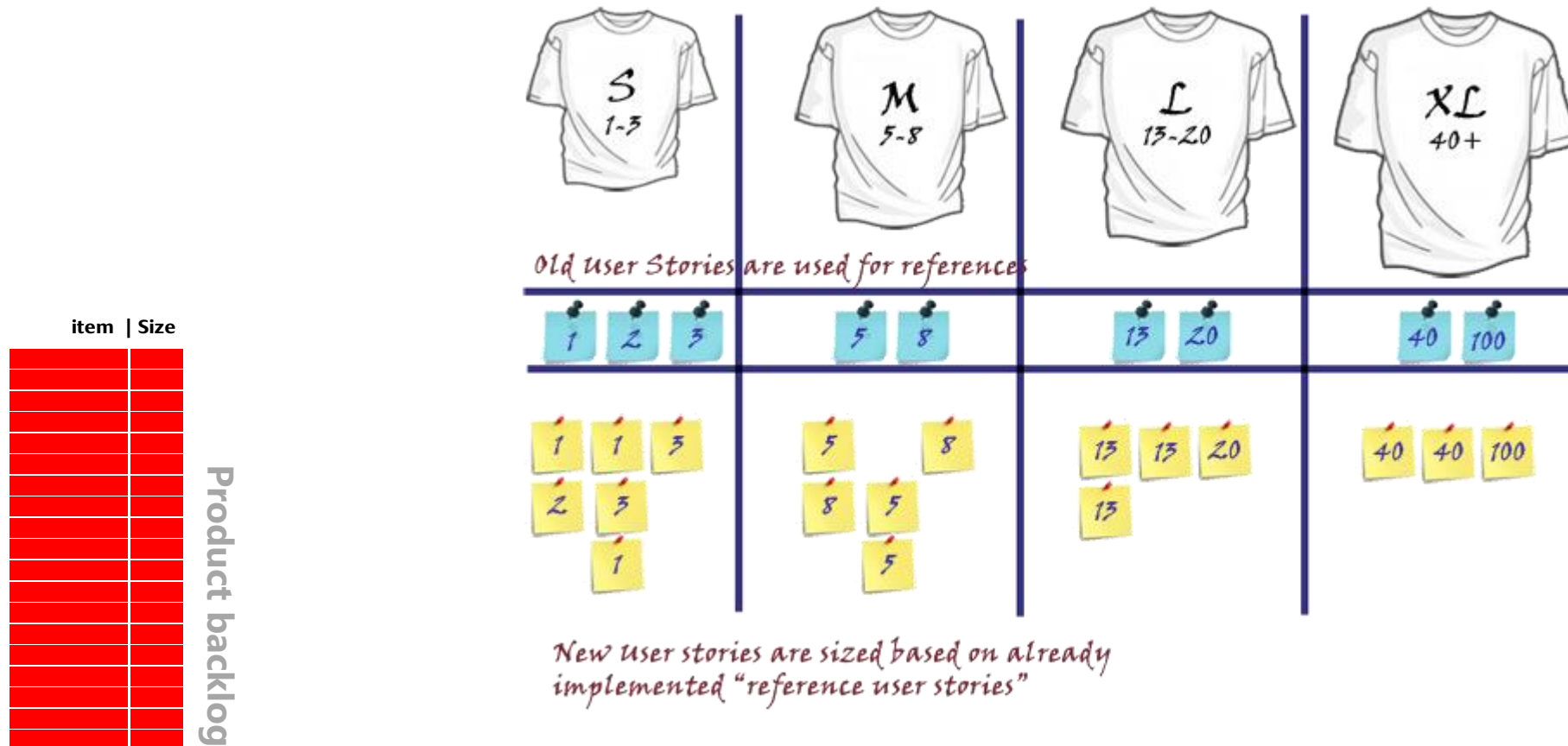
1. The product owner explains the user story.
2. The team members ask questions, discuss assumptions and risks, and determine the necessary tasks (so they are thinking about the WBS...).
3. The scrum master asks all team members to make an individual estimate of the work and pick a card from their own set.
4. The team members reveal their cards at the same time and the persons with the highest and lowest estimates explain their choice.
5. Steps 3 and 4 are repeated until a consensus is reached.



Source: *The complete project manager, 2016*

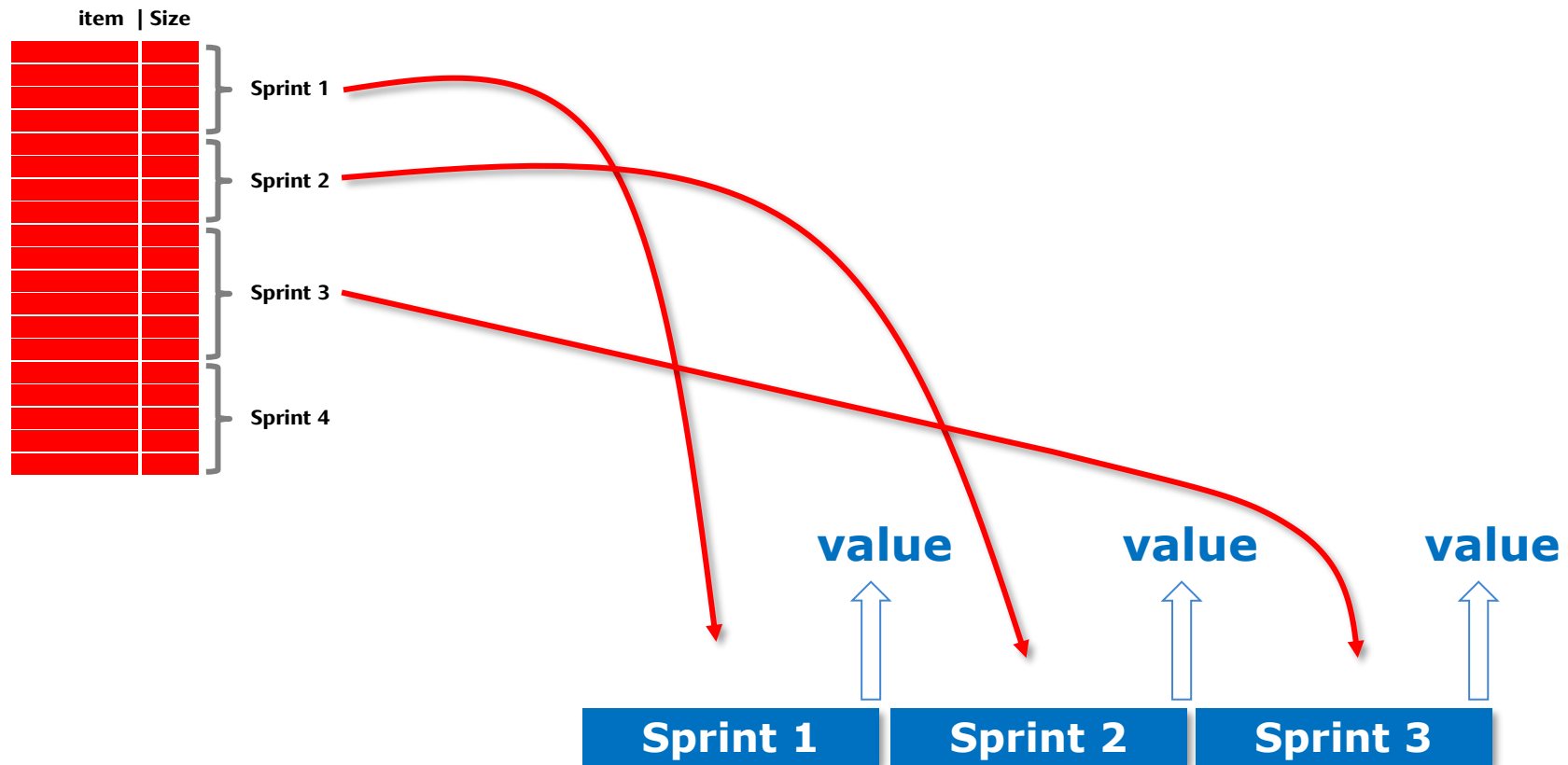
Size of user stories: T-shirt sizing

Alternatively, **T-shirt sizing** can be used to assign size/complexity to the user stories:



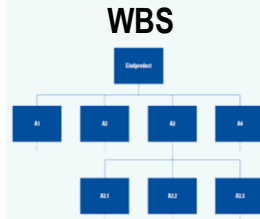
Source: www.deltamatrix.com, 2022

Sprint planning



Sprint planning: substantially different than traditional

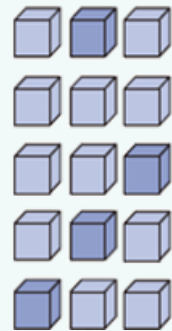
Traditional: How do I assign the resources to realize the work?



- What is the size and amount of work of the desired deliverables?
- What variable capacity do I need to deliver the deliverables on time?

Scrum: How do I use the available resources to do the maximum?

Product backlog

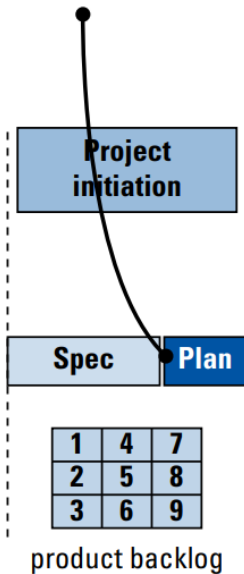


- What is the size and priority of the items on the product backlog?
- How many items can fit into a sprint with fixed capacity and fixed duration?

Sprint planning: only detail that is needed

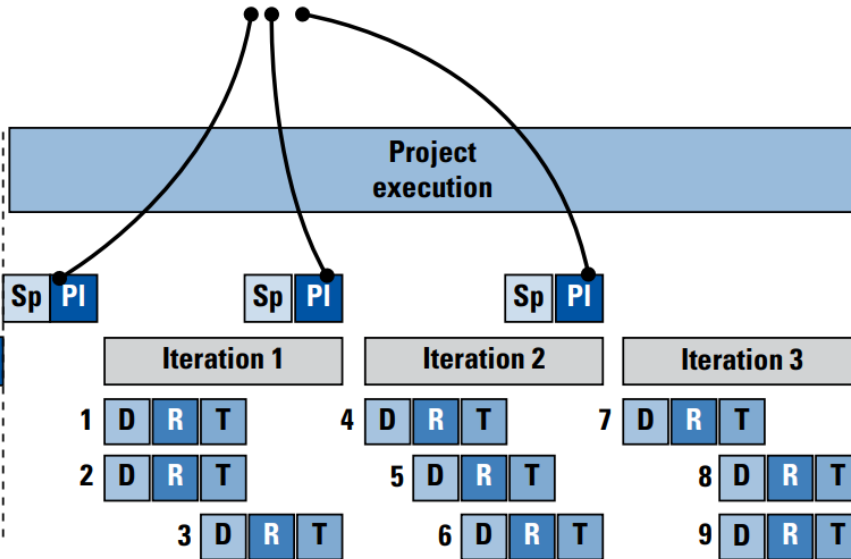
Understanding project size

User stories on product backlog (PBS) → **size** based on story points (to allocate user stories to sprints)



Directing project execution

Tasks on sprint backlog (WBS) → **effort** based on hours (to facilitate sprint detailed plan)



Source: *The complete project manager*, 2016



With Scrum, in the initiation phase you limit yourself to understanding the project size and planning the sprints. Further detail regarding execution (tasks on the scrum board) is not needed until the sprint starts.



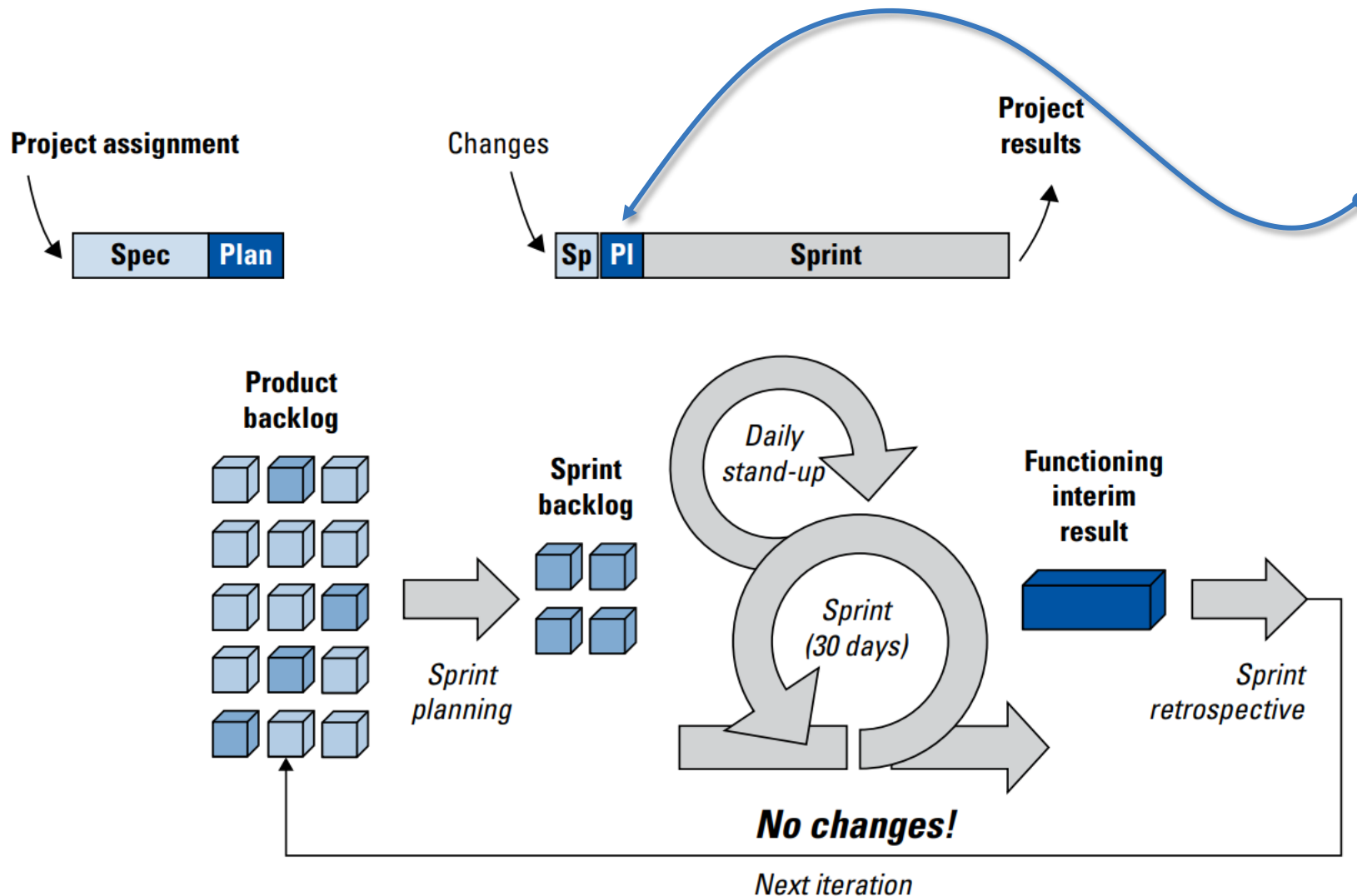
This way you avoid recording detail too early that almost certainly will have to be adjusted in the time before the execution actually starts.

Scrum implementation and execution

- Characteristics of Scrum
- Team and organization
- Product backlog
- Sprint planning
- **Sprint execution**
- Reporting tools and metrics
- Multiple (multidisciplinary) teams



Sprint execution



During project execution the sprints are carried out. Each sprint is started with a **sprint planning meeting** and completed with a **sprint review** and **sprint retrospective**.

During the sprints, team members communicate daily in the **daily stand-up meeting**.

Sprint planning meeting

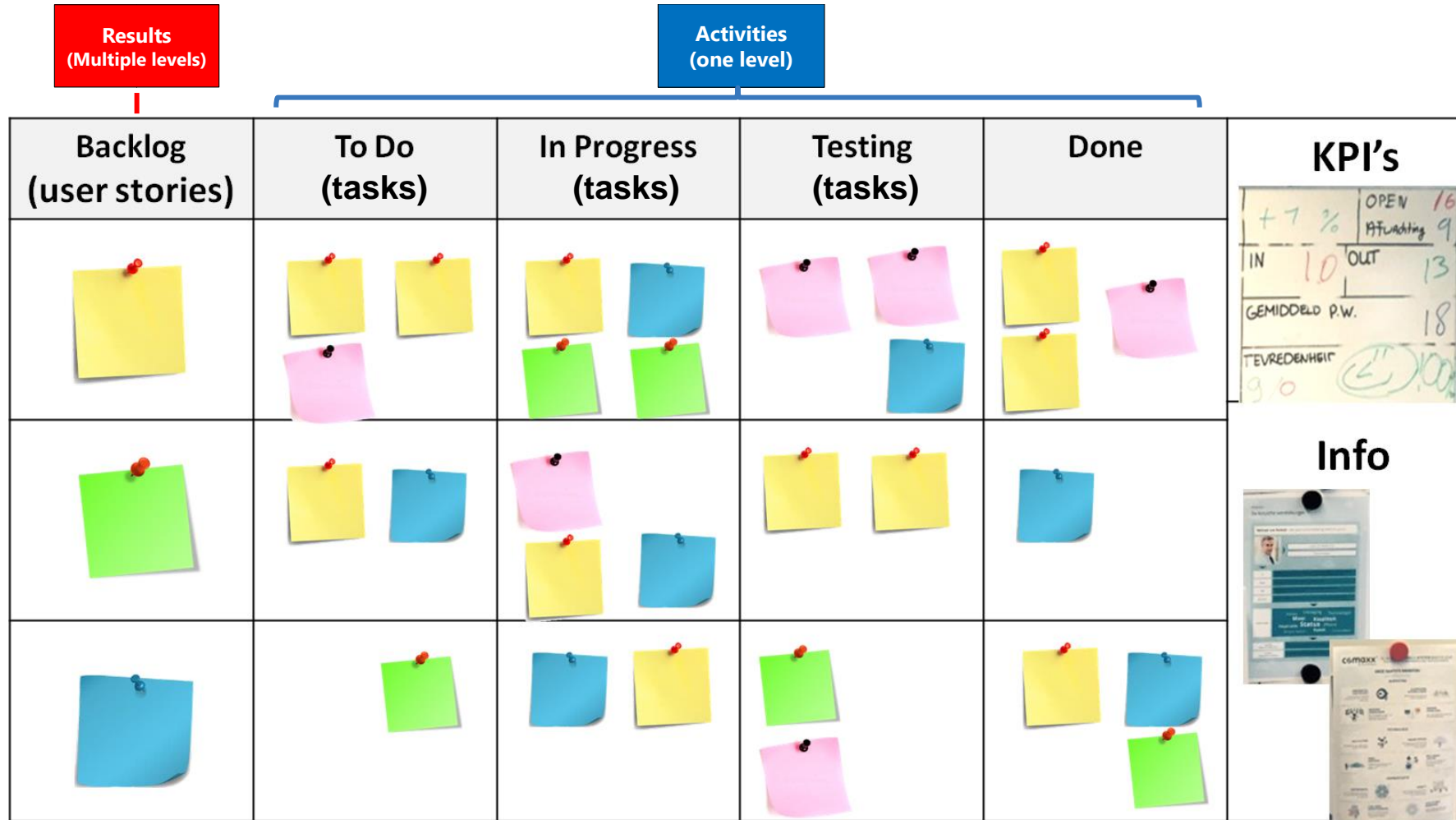
in the **sprint planning meeting** prior to the sprint, the sprint goal is finalized (which user stories need to be delivered) and the team makes a plan for the work to be carried out, the **sprint backlog**. This means that the user stories are **broken down into tasks** (activities) of up to one day's duration. These tasks appear on the **scrum board** under 'to do'.

At the sprint planning meeting, the entire team is present (plus a few additional experts if necessary). The **scrum master** facilitates the meeting and checks whether the appropriate level of detail is achieved. The **product owner** explains the items on the product backlog in such a way that everyone understands them.



Source: PULZ, 2022

Sprint planning meeting: Scrum board



Results
(Multiple levels)

Activities
(one level)

End result

Results
(Multiple levels)

Activities
(one level)

Source: PULZ, 2022

Sprint planning meeting: Sprint backlog

Sprint backlog

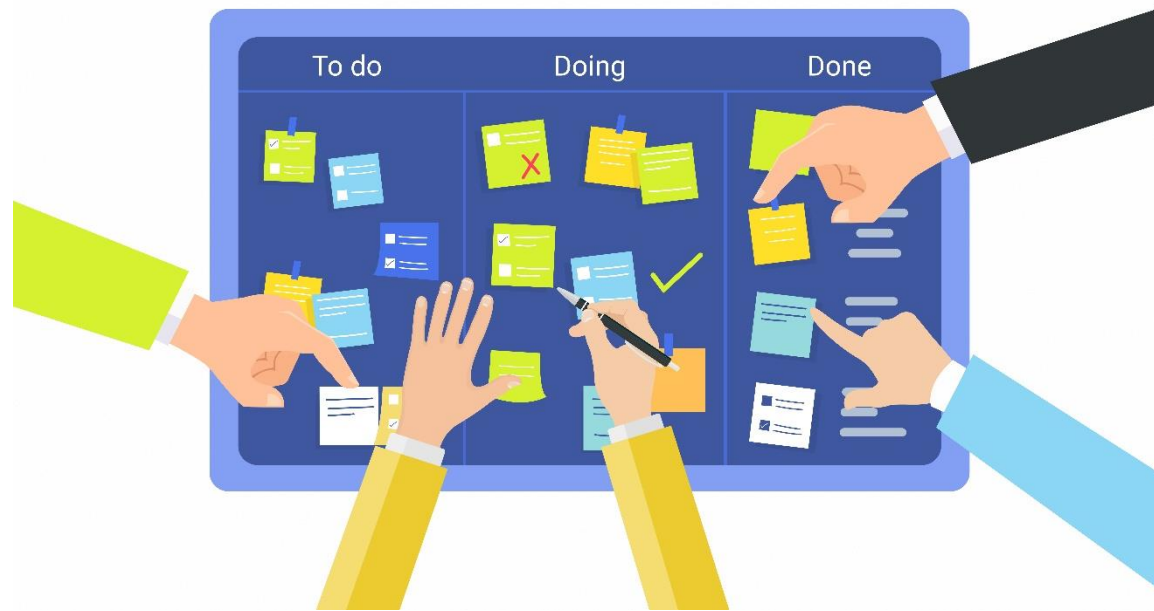
							hours spent per sprint day																				
Id	Theme	Backlog item	Status	Responsible	Story points (user story)	Hours scheduled (task)	Hours spent (task)	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
		User story #1					0																				
		task					0																				
		task					0																				
		task					0																				
		task					0																				
		User story #2					0																				
		task					0																				
		task					0																				
		task					0																				
		User story #3					0																				
		task					0																				
		task					0																				
		task					0																				
		task					0																				
		User story #4					0																				
		task					0																				
		task					0																				
		task					0																				
		task					0																				

[Download this template](#)

Daily stand-up meeting

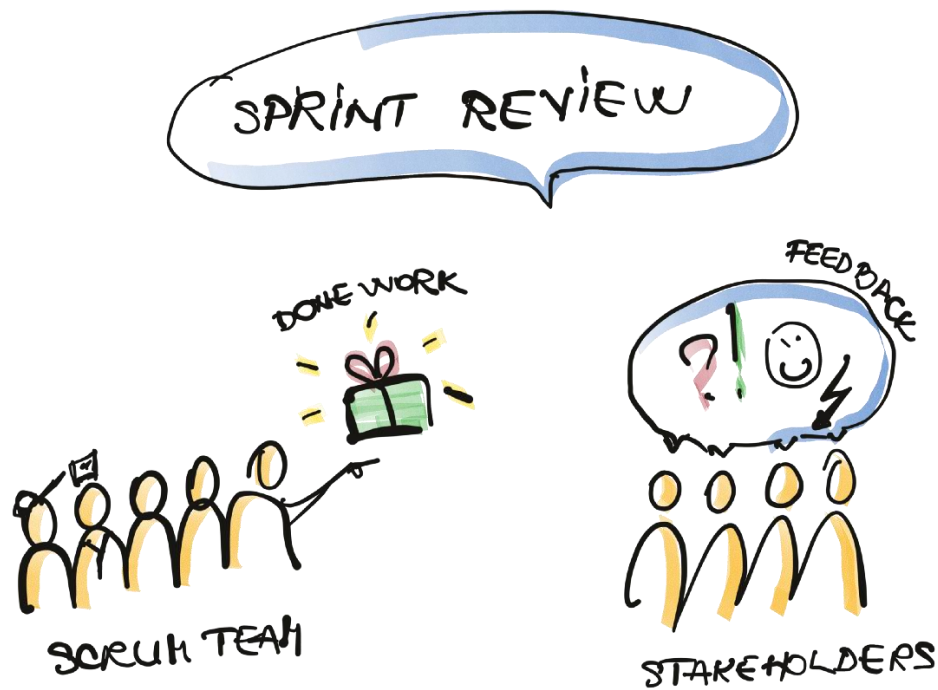
In the **daily stand-up meeting** (same place, same time, maximum 15 minutes), team members answer the following questions at the scrum board daily:

- What did I do **yesterday** to help the team finish the sprint?
- What will I do **today** to help the team finish the sprint?
- What **obstacles** prevent the team or me from completing the sprint?



It is continuous work and improvement to ensure that the daily stand-up meeting does not exceed 15 minutes.

Sprint review en retrospective



Source: A. Onuta, 2022

After the completion of the sprint, there are two meetings that take place prior to the next sprint.

Sprint review: in this meeting, the achieved **results** are reviewed and - if necessary - the product backlog is adjusted. Project stakeholders are also present at this meeting. A **demo** of the working product is an excellent way to obtain feedback and test whether the development team and stakeholders (think of the end user!) have a shared vision about the added value. The feedback is used in the continuation of the project.

Sprint retrospective: during the sprint retrospective, the functioning of the **scrum team itself** is reviewed. Areas for improvement are raised, which are taken into account in the next sprint.

Source: PULZ, 2022

Scrum implementation and execution

- Characteristics of Scrum
- Team and organization
- Product backlog
- Sprint planning
- Sprint execution
- **Reporting tools and metrics**
- Multiple (multidisciplinary) teams



Tools: overview metrics and reporting

Metrics (indicators, KPIs) help to monitor progress. The following applies here: make sure it is **really about helping and not about checking (afterwards)**. In other words: choose indicators that make it possible to adjust in time (**leading indicators**) and not only tell you in retrospect that it is not good enough (**lagging indicators**).

Possible reports:

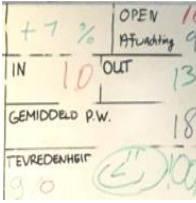








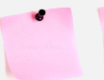
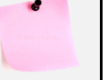
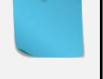

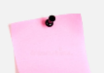





















- Scrumboard
- Sprint burndown reporting
- Status sprint backlog
- Product backlog status
- Velocity chart
- Progress sprints
- Test track progress



Source: PULZ, 2022

Hulpmiddelen: Scrum bord

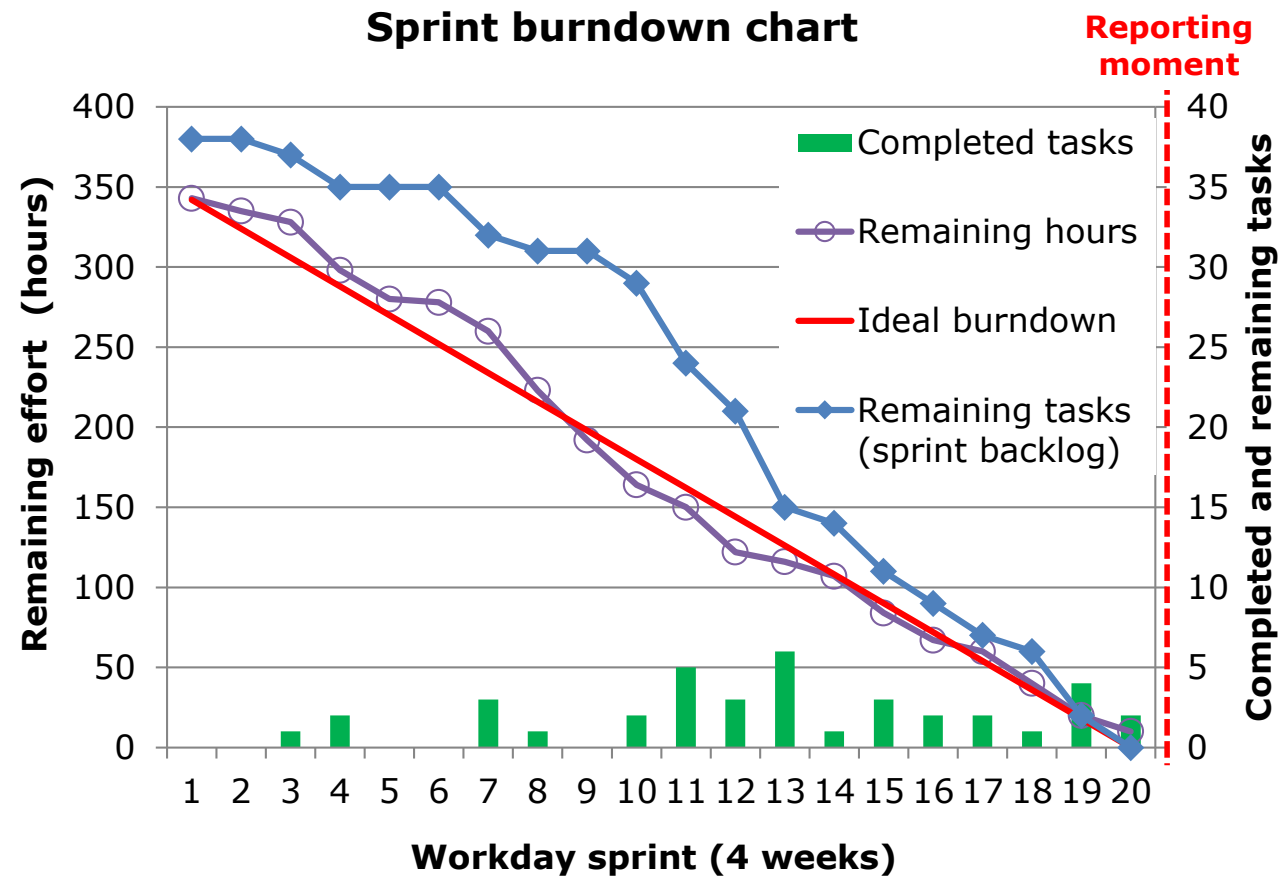
Scrum board (sprint status daily stand-up meeting)

Backlog (user stories)	To Do (tasks)	In Progress (tasks)	Testing (tasks)	Done	KPI's 
	  	   	  	  	Info  
	 	  	 		
		 	 	  	



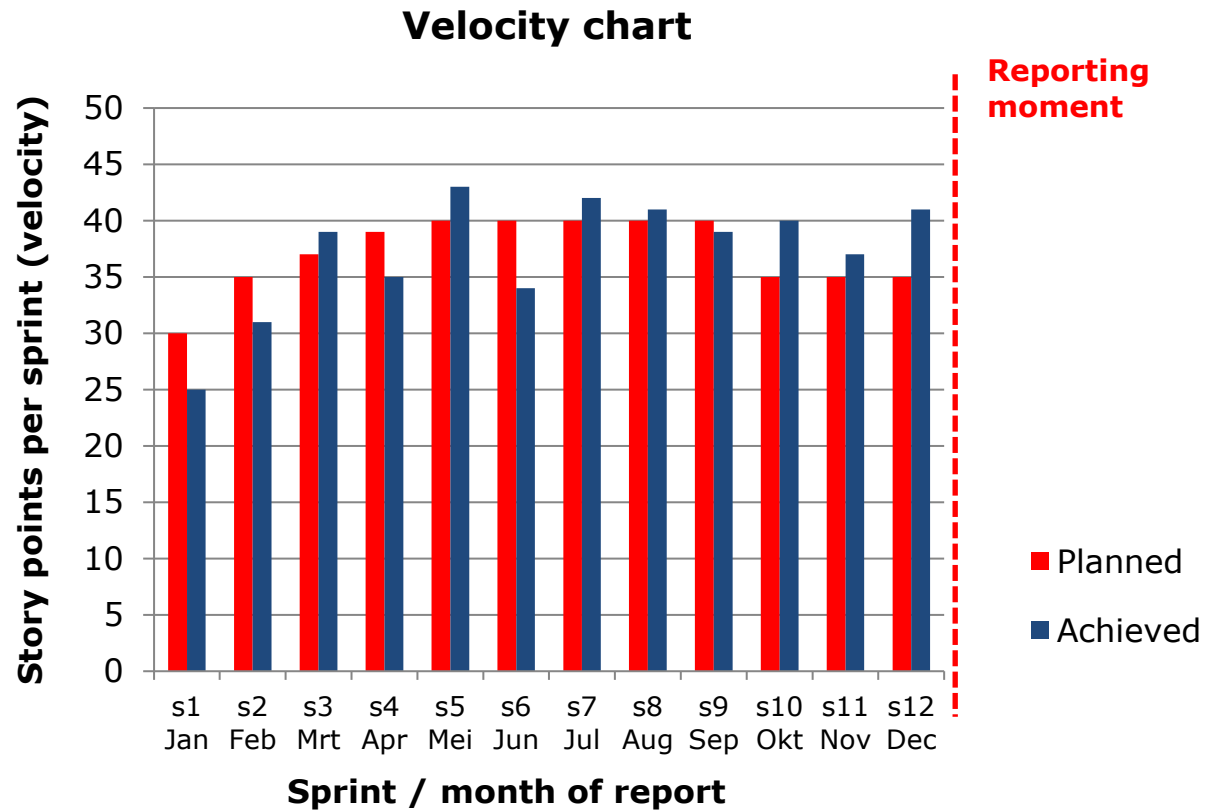
Choose your own strategy for the colors. E.g. type of backlog item (new feature, found issue), which project, which system part, which discipline, etcetera.

Tools: sprint burndown report



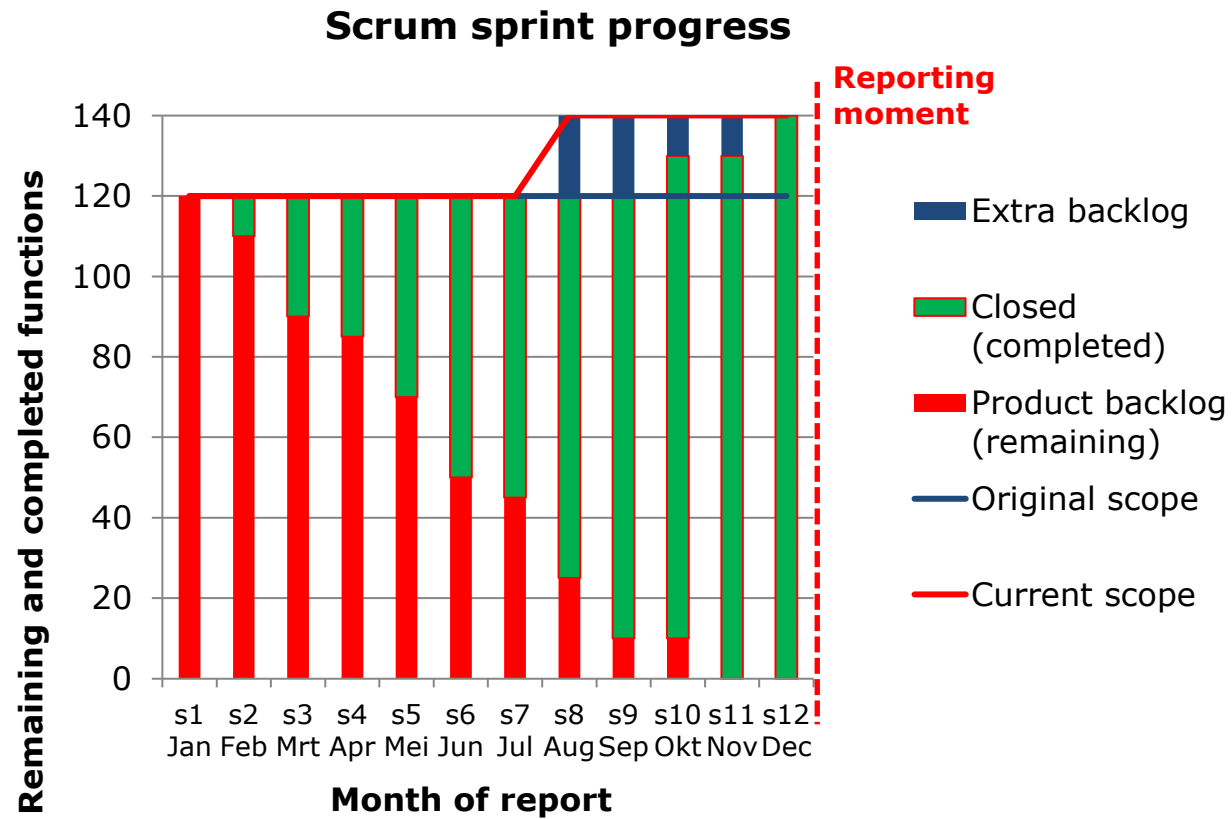
Source: Wessels, 2006

Tools: velocity chart



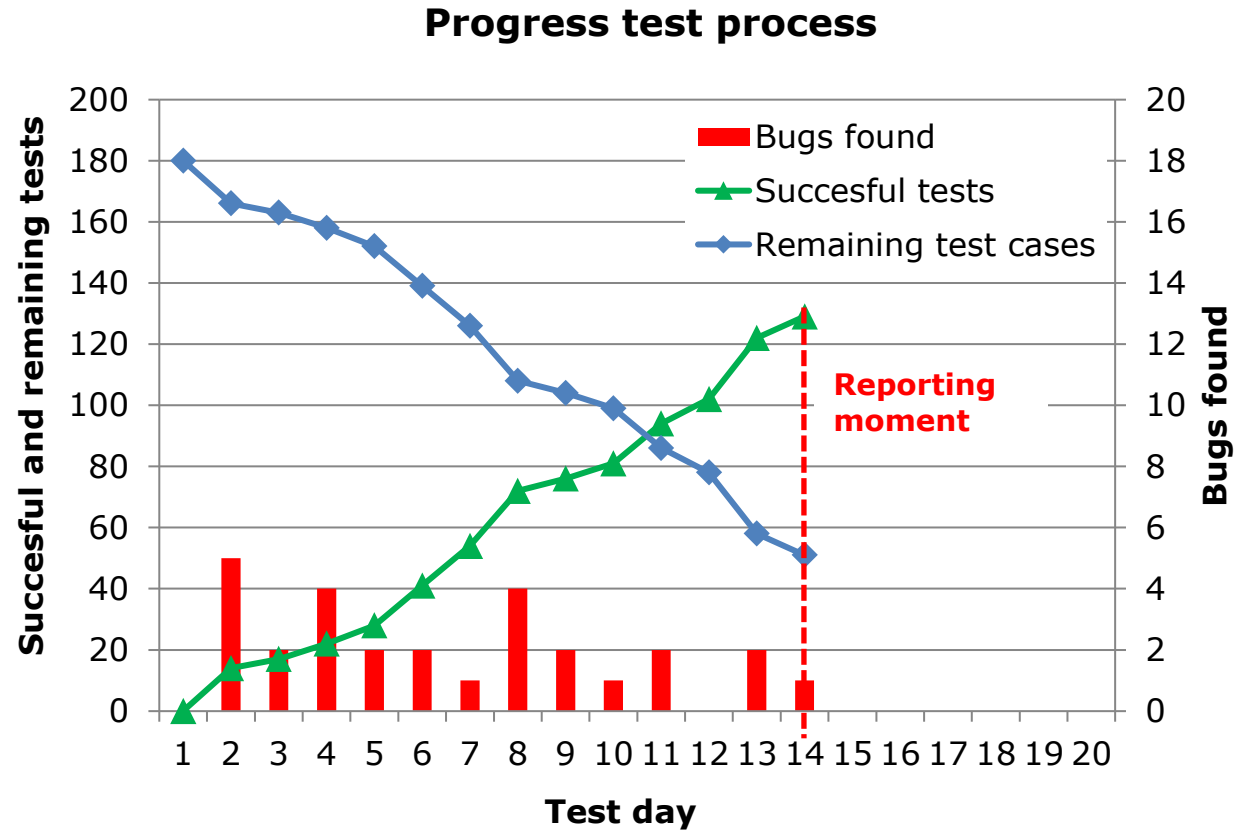
Source: Wessels, 2006

Tools: sprint progress



Source: Wessels, 2006

Tools: progress test process



Source: Wessels, 2002

Scrum implementation and execution

- Characteristics of Scrum
- Team and organization
- Product backlog
- Sprint planning
- Sprint execution
- Reporting tools and metrics
- **Multiple (multidisciplinary) te**



Often it does not work out with one team

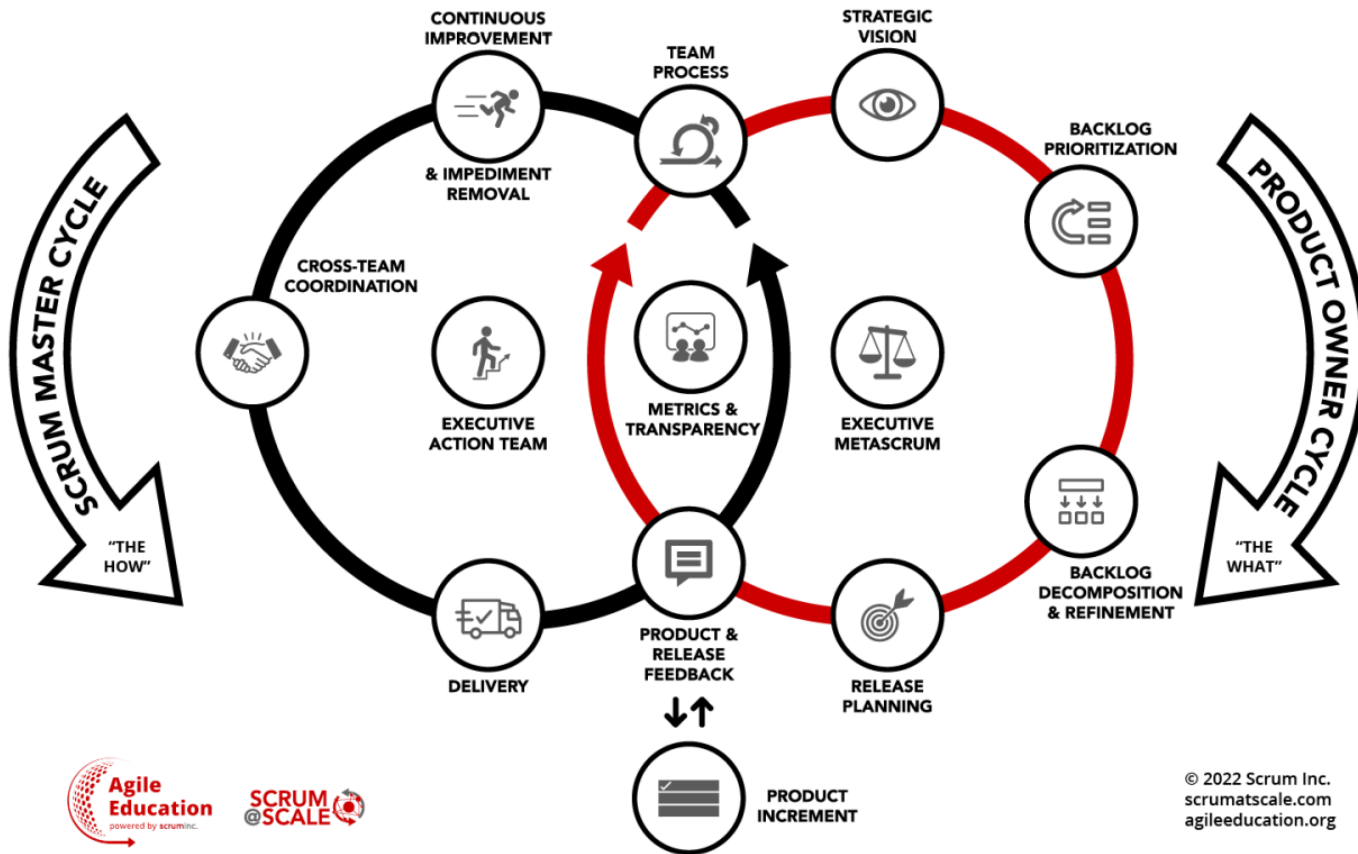
Scrum is all about one team and products developed by one team. In practice, you quickly run into limitations with this, because you often need more people than the prescribed team size of 7 ± 2 team members.

This means that the work has to be divided between **several teams**. The **collaboration of these teams is not trivial, because it involves** a combination of planning&communicating as well as how to make it technically possible. The latter means making the right decomposition into subsystems and modules from the perspective of **systems thinking and architecture**, so that the teams know what they need to make, the sprint results can be tested relatively independently and the integration of subresults is facilitated.

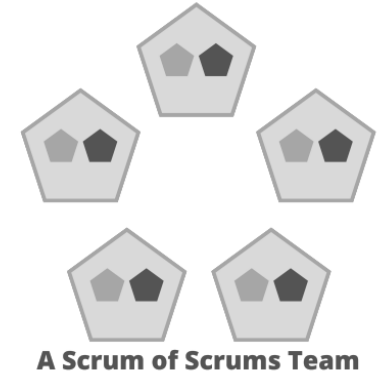
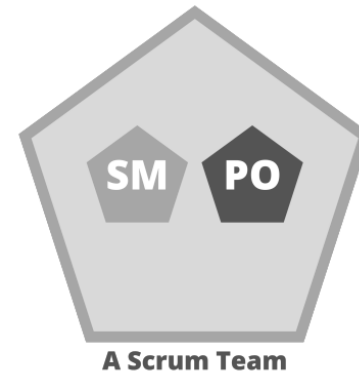
Several parties have responded to this shortcoming. For example:

- **Scrum@Scale (by Jeff Sutherland)**
- **SAFe (Scaled Agile Framework)**
- **LeSS (Large Scale Scrum)**
- **Spotify Model**

Scrum@Scale Framework



© 2022 Scrum Inc. scrumatscale.com agileeducation.org

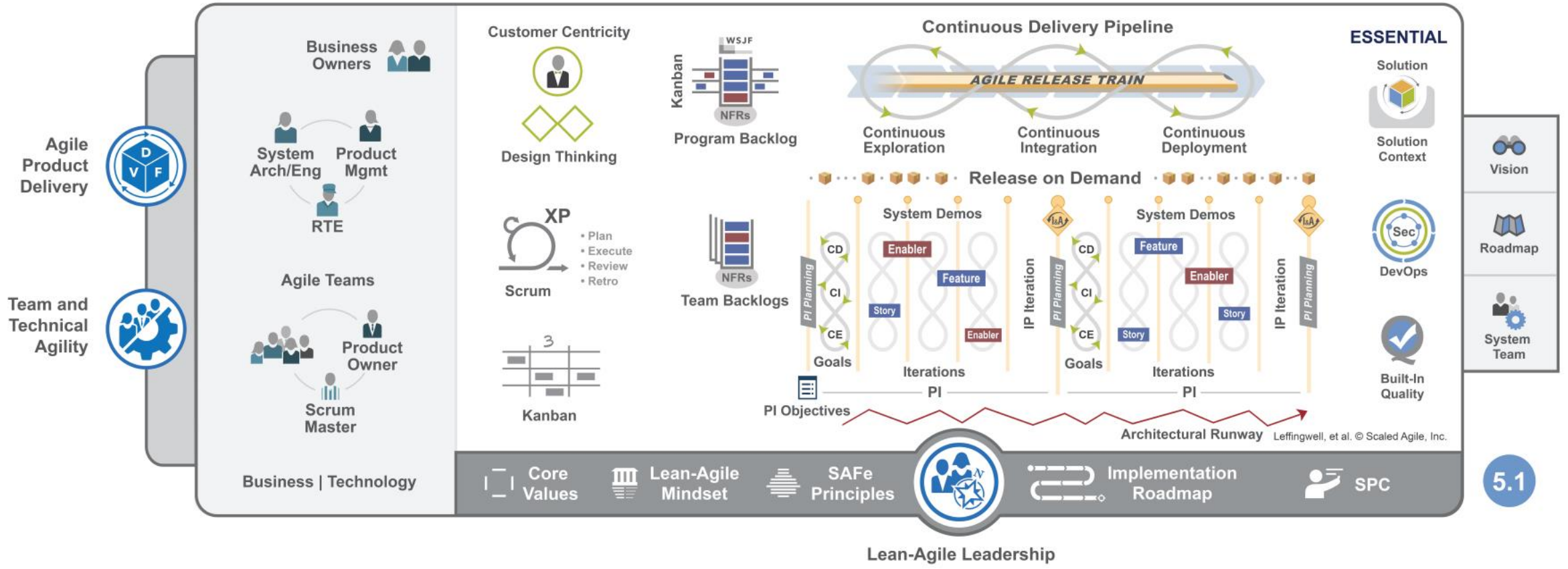


Scrum: www.scrumatscale.com, 2023

SAFe (Scaled Agile Framework)

SAFe online resources

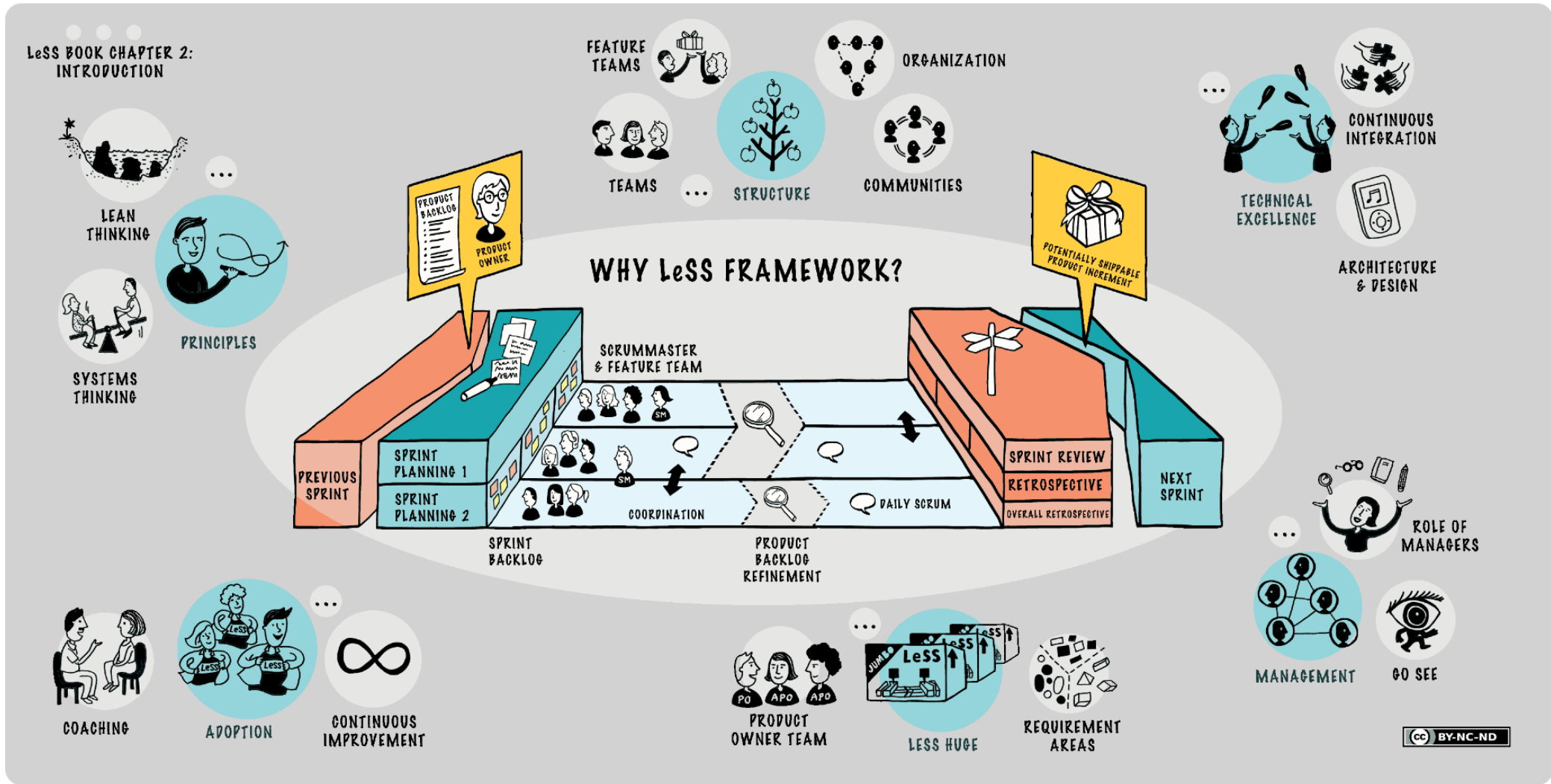
SAFe® for Lean Enterprises



Source: www.scaledagileframework.com, 2023

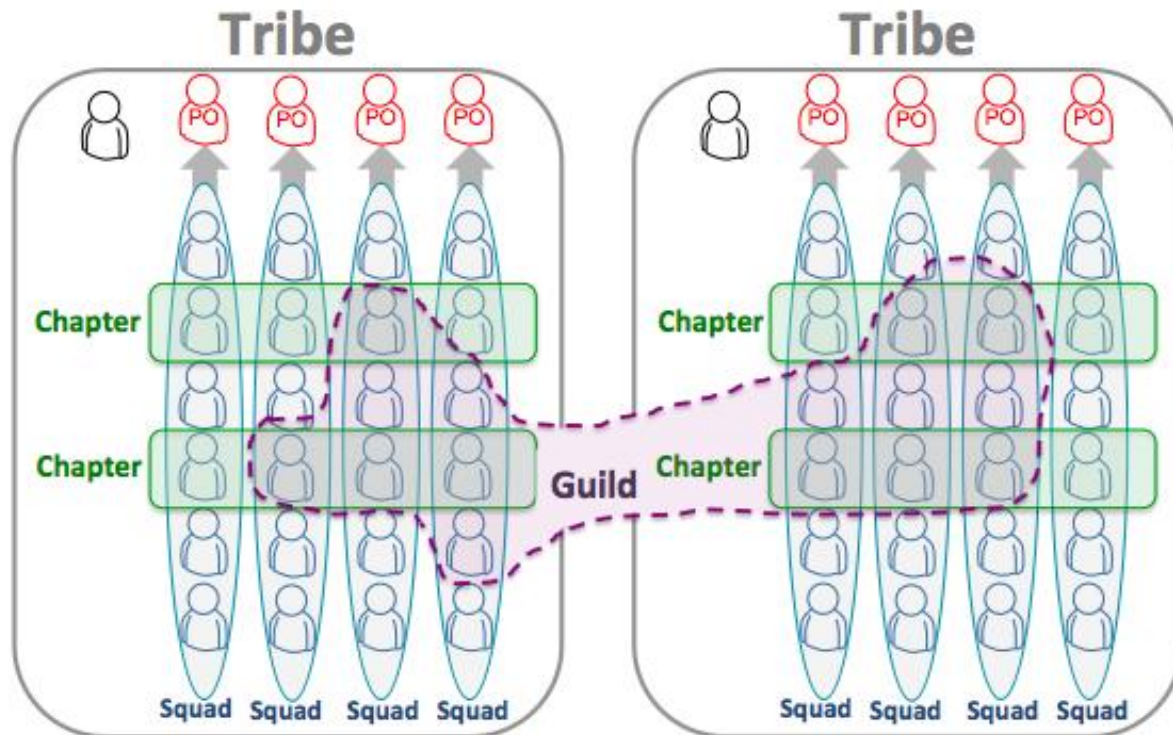
LeSS (Large Scale Scrum)

LeSS online resources



Source: www.less.works, 2023

Spotify Model



A **Squad** is similar to a Scrum team. A Squad is a **self-managing** and **multidisciplinary** team that has all the knowledge and competences to deliver a product or service from design to final product.

A **Tribe** is a **group of Squads working on** the same or related products or services.

A **Chapter** consists of employees with **the same expertise within a Tribe**.

A **Guild** is a group of employees with **the same expertise and/or interests** that are not limited to the Tribe but scattered throughout the **organisation**.

Challenges of Scaled Agile Frameworks



If you delve into the various Scaled Agile Frameworks, you will see many logical elements and similarities. At the same time, they are very **complex** and **require uniform application organization-wide**. This makes successful implementation difficult.

But more importantly, **you cannot implement an agile framework 1:1 as a copy in an organization**. The **framework needs to be tuned** and **users need to learn to apply it**. That is pretty demanding. Agile frameworks could be considered a *"car that you can only use if you have your driving licence AND if you understand all the technology it contains AND you can even adapt it."* Logically, there is often a lack of specialists who master this combination, or they are available but do not get support for the transition process within the organization. As a result, there are more people discussing what is wrong with the framework than there are people who really want to make a success of it in a practical way.

By the way, you also see the opposite; that organizations without hesitation push a new way of working. For example, large organizations (including ING) have jumped into implementing the Spotify Model at once. The question is whether that is brave or naive, since **Spotify itself no longer used it at all** (and, according to some insiders, never did). The Spotify Model seems less powerful in practice than it looks on paper. You might conclude that it is more of a relabelling of a traditional matrix organisation, rather than really addressing its drawbacks. Old wine in new bottles, and a fairly naive push for large-scale autonomy without creating the right preconditions. **Self-organization without structure and boundaries, unfortunately, usually leads to chaos....**

Challenges of Scaled Agile Frameworks



Fortunately, there is also a **bright side**. If you study the Scaled Agile Frameworks, **you will see the essences, which we named in the Masterclass and the book *The Complete Project Manager*, all reflected.**

To make the transition more likely to succeed, by implementing these essences, you can also **scale up in smaller steps**. Steps that fit the knowledge level of your organization and support the learning curve.

Once you have mastered the essences in the basics, you can start applying them in a more complex situation and combine them. An example is the **Multidisciplinary Heartbeat**, that I explain on the following slides. The Multidisciplinary Heartbeat is suitable for synchronizing multiple teams (or disciplines) and integrating their interim results into one end result (system).



Multidisciplinary Heartbeat: characteristics

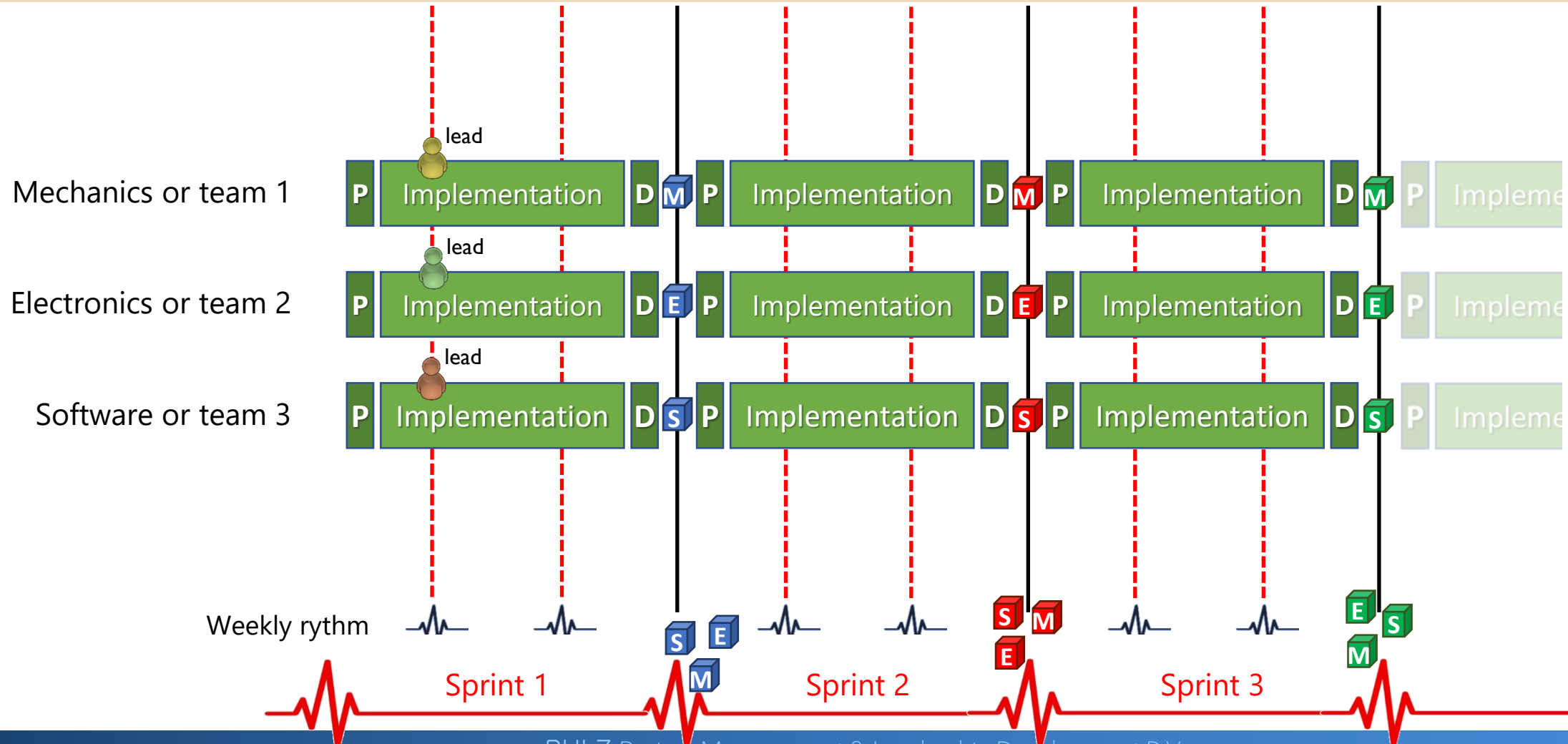
- One synchronised rhythm (heartbeat) for all teams.
- Two types of teams:
 - Subsystem/module/discipline team: delivery of module and subsystem results (including module test).
 - System teams: 'Architecture & PM' creates the PBS, system design, subsystem specifications and sprint planning. 'Integration' integrates the module results into a working system.
- The aim of the system teams is to create maximum independence between modules and early integration of modules (and testing at system level).
- Each subsystem/discipline has a responsible lead.
- The system is divided into releases, subsystems, modules and deliverables (or user stories) using the product breakdown structure (PBS).
- These deliverables are assigned to sprints (sprint planning). This is the interface/contract between the (system) project manager and the teams.
- The PBS and sprint planning are a co-creation of the (system) project manager, the (system) architect and the team leaders.
- The team converts the requested sprint results into its own activity plan. This can be a scrum board, a gantt chart, or another planning tool (depending on the team's preference).
- Heartbeat levels:
 - Sprint: delivery of module and integration results
 - Weekly: project progress, PBS update and preparation for next sprints (PM, architect and team leaders)
 - Daily (or lower rhythm): coordination of work and sprint progress (team members with team leader/lead)

Source: PULZ, 2022

Multidisciplinary Heartbeat: three independent teams

P Sprint planning: determine assigned product backlog items, define tasks, check dependencies (if more than one team).

D Sprint demo: check sprint results (of modules **M** and system integration if more than one team).

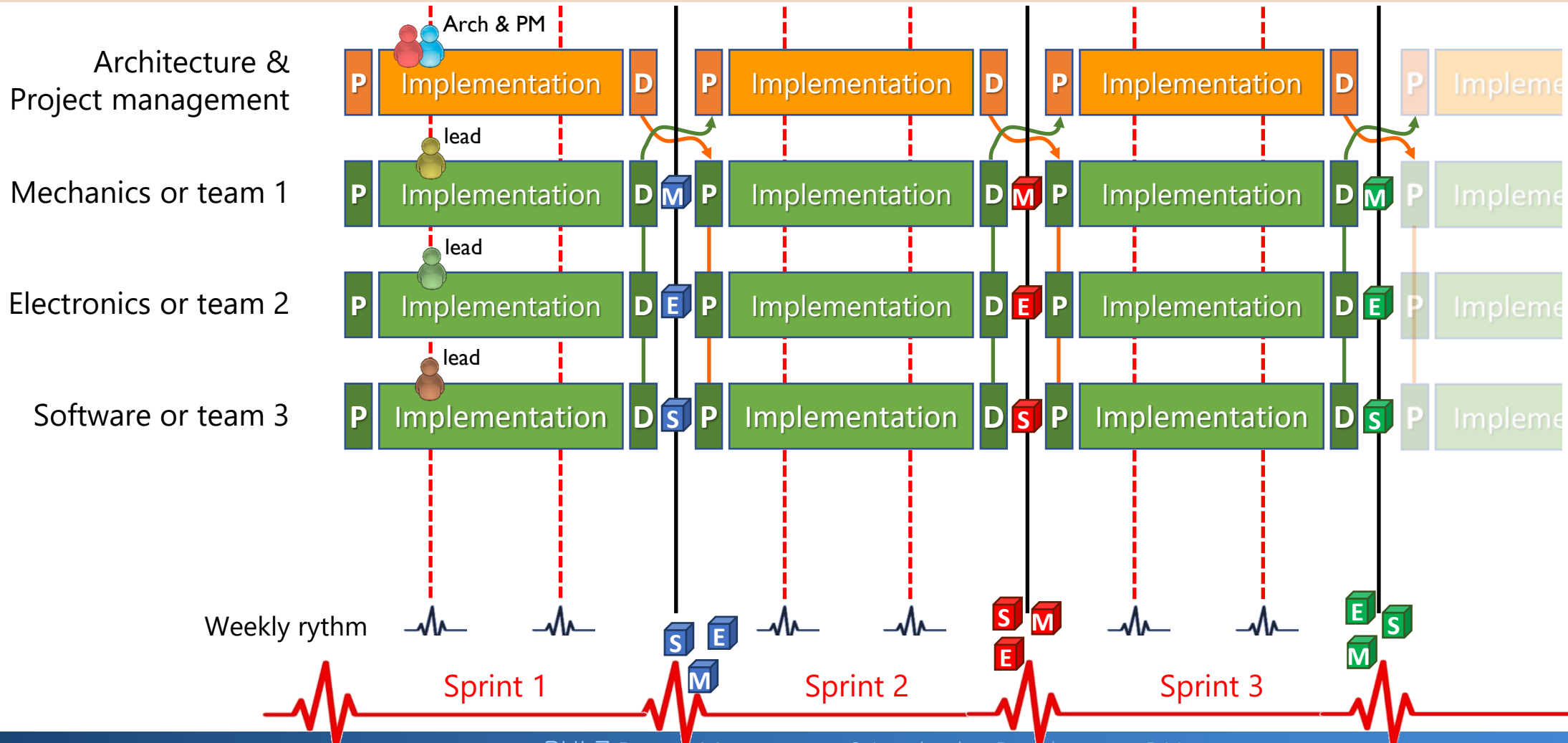


Source: PULZ, 2022

Multidisciplinary Heartbeat: three synchronized teams

P Sprint planning: determine assigned product backlog items, define tasks, check dependencies (if more than one team).

D Sprint demo: check sprint results (of modules **M** and system integration if more than one team).

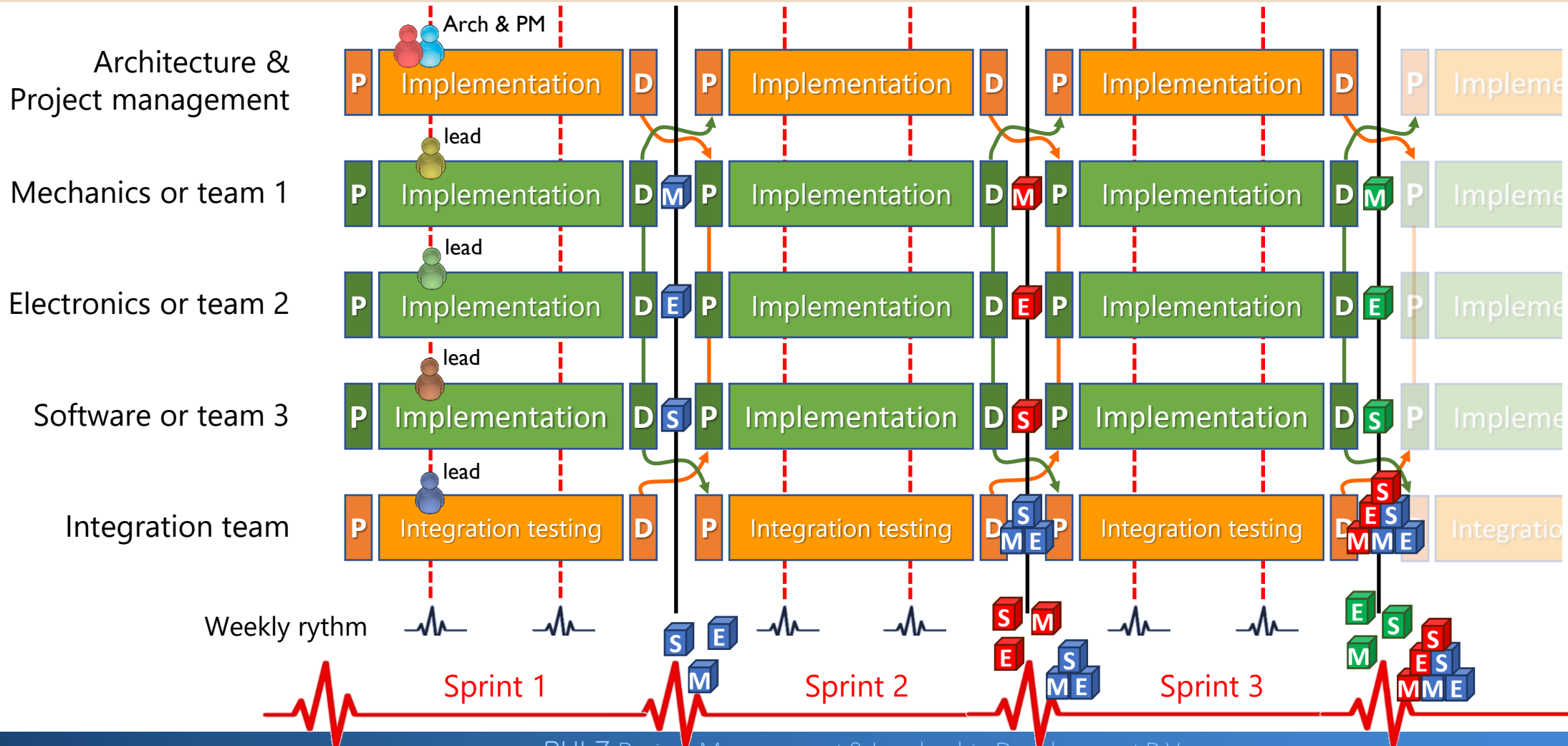


Source: PULZ, 2022

Multidisciplinary Heartbeat: including integration

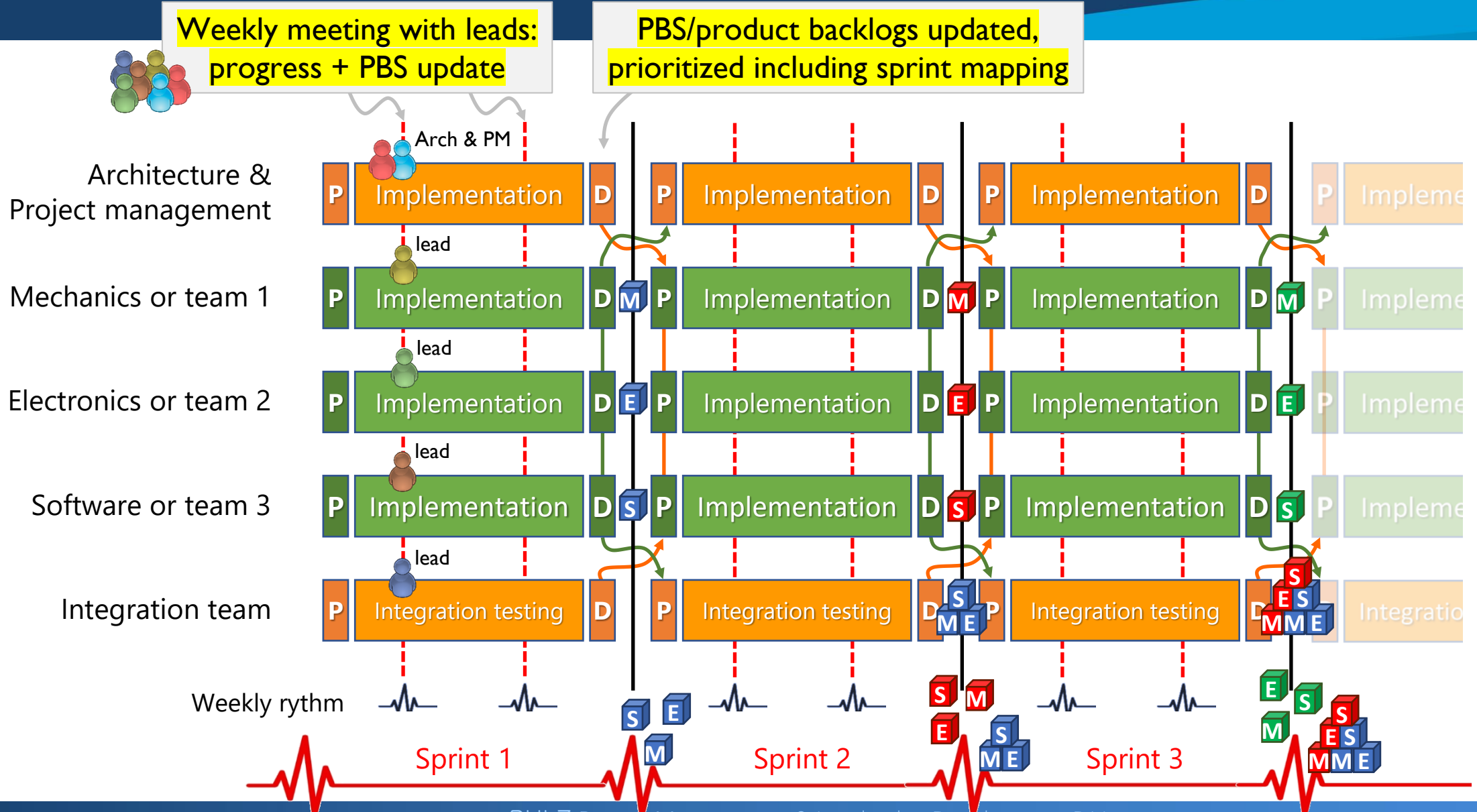
P Sprint planning: determine assigned product backlog items, define tasks, check dependencies (if more than one team).

D Sprint demo: check sprint results (of modules **M** and system integration if more than one team).



Source: PULZ, 2022

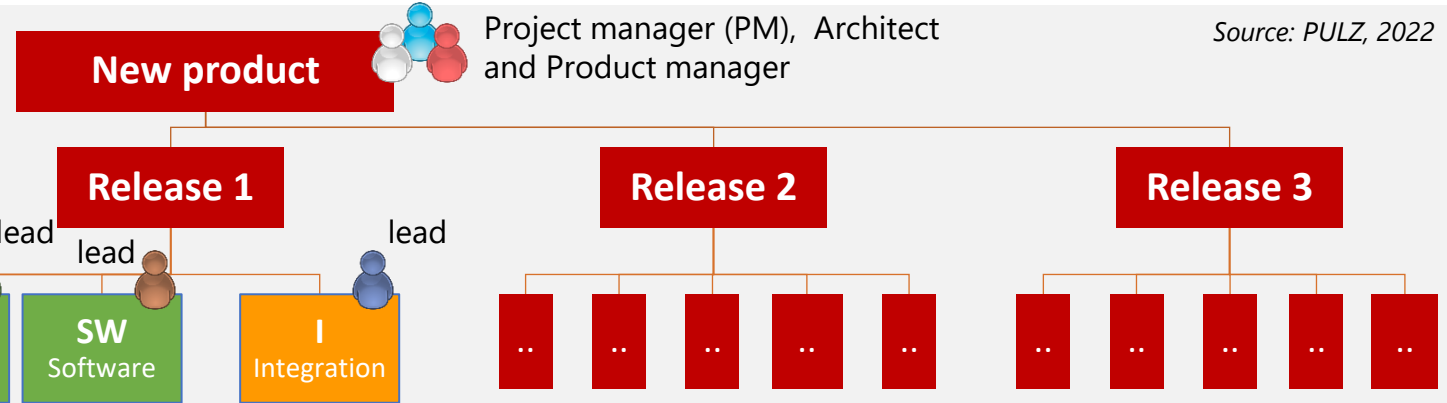
Multidisciplinary Heartbeat: coordinating meetings



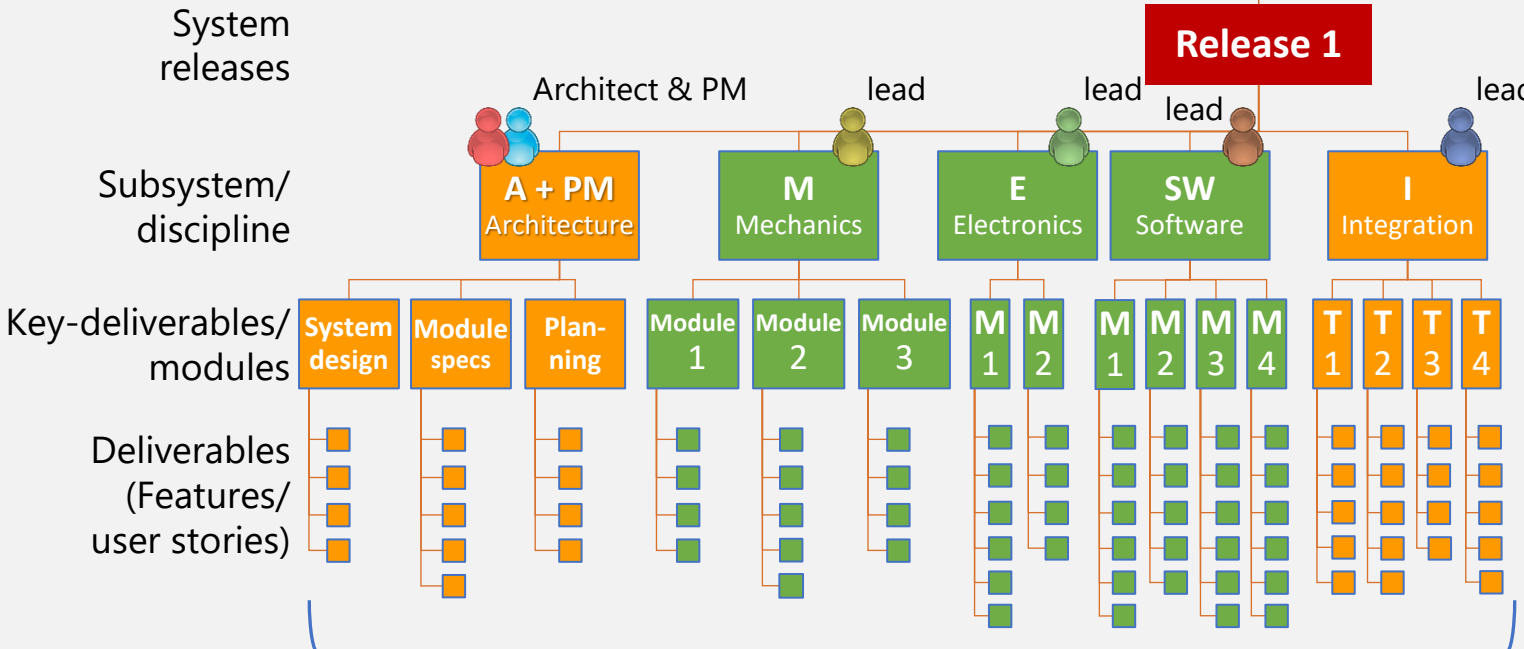
Source: PULZ, 2022

Multidisciplinary Heartbeat: associated PBS

Product Breakdown Structure (PBS)

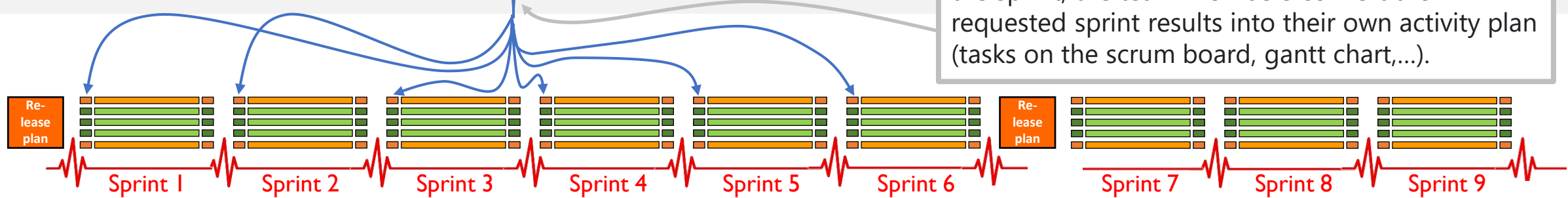


Source: PULZ, 2022



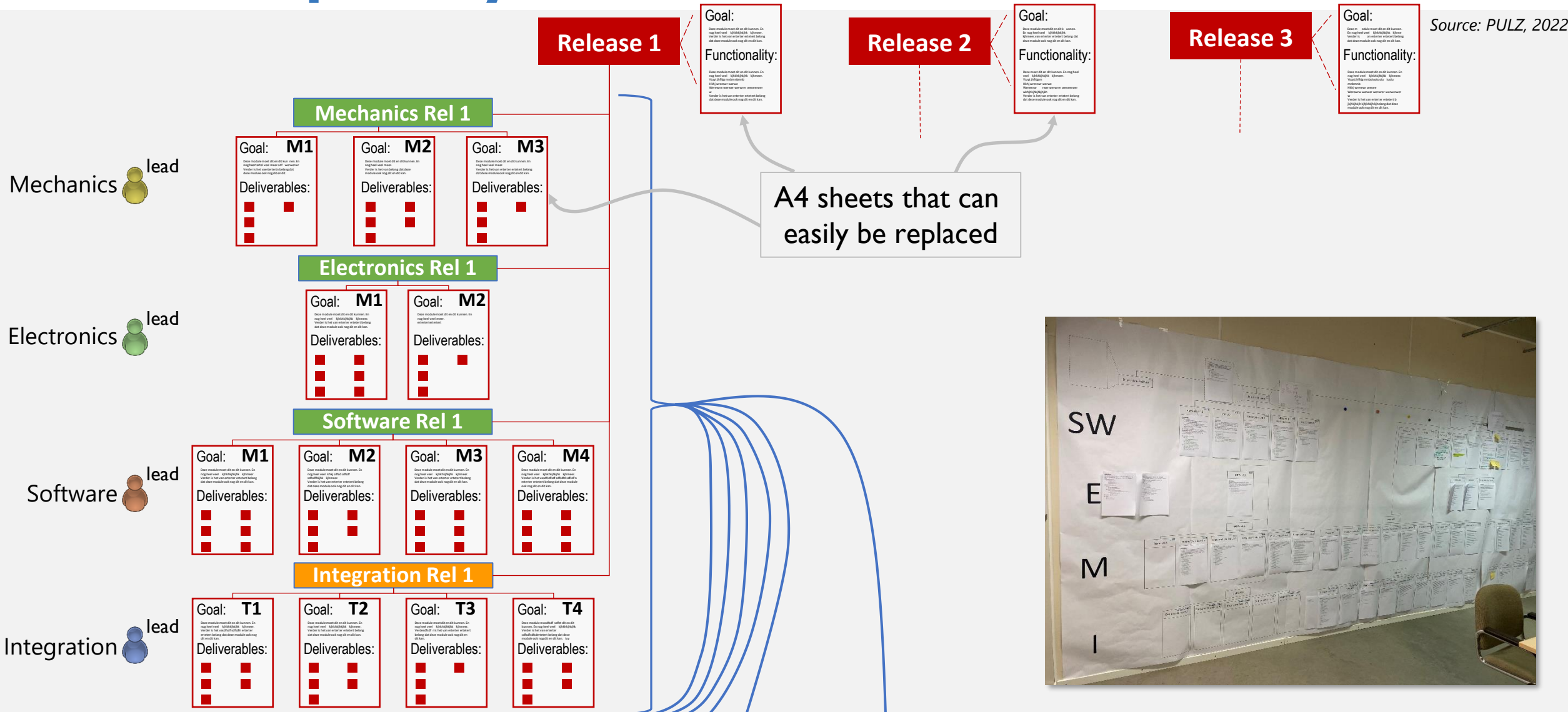
During the release planning week (**Re-release plan**), the PBS for the next release is refined and the features/user stories/deliverables are assigned to the sprints.

During the sprint planning meeting (**P**) prior to the sprint, the team members convert the requested sprint results into their own activity plan (tasks on the scrum board, gantt chart,...).



Multidisciplinary Heartbeat: 'Plan on the wall'

Source: PULZ, 2022





Nothing from this publication may be reproduced, recorded in an automated database or published on or via any medium, either electronically, mechanically, through photocopying or any other method, without prior written permission from the author.

This publication was produced with the utmost care and attention. Nevertheless, the text may contain errors. PULZ Project Management & Leadership Development B.V. and the author are not liable for any errors and/or inaccuracies in this text.

www.roelwessels.nl