The Professional Scrum Competencies

https://www.scrum.org/professional-scrum-competencies

Contents:

Scrum.org has created these Professional Scrum Competencies to help guide an individual's personal development with Scrum. Building proficiency with Scrum starts with the fundamentals, *Understanding and Applying the Scrum Framework*, and it is the foundation for personal growth. The competencies and underlying focus areas apply to the Scrum Team (Product Owner, Scrum Master, and Developers) and to other roles in the organization such as Agile Leaders.

Organizations can benefit from the use of a common understanding of the competencies and focus areas that they use to evaluate and balance their team's proficiencies based on their unique needs. View how Scrum.org Professional Scrum Training Courses cover the Focus Areas.

1/ Understanding and Applying the Scrum Framework

https://www.scrum.org/professional-scrum-competencies/understanding-and-applying-scrum-framework

Understanding and Applying the Scrum Framework allows teams and organizations to iteratively and incrementally deliver valuable products of "**Done**" working releasable software in 30 days or less. Successful use of the Scrum framework requires an understanding and application of the **Scrum Values** and the tenets of **Empiricism** to professionally deliver value to the organization while addressing the inherent complexity of product delivery. The Scrum framework consists of Scrum Teams and their associated **Roles**, **Events**, and **Artifacts**. Each of these components within the framework serves a specific purpose and are essential to Scrum's success and usage. The rules of Scrum bind together the Roles, Events, and Artifacts, governing the relationships and interaction between them.

In the case of **Scaling** and scaled implementations of Scrum, minimizing cross-team dependencies and resolving integration issues are unique and critical challenges when multiple Scrum Teams are collaborating to deliver a product.

The Scrum Guide is the foundational body of knowledge for the Scrum framework and the Nexus Guide builds upon that foundation as the body of knowledge for the Nexus scaling framework.

Understanding and Applying the Scrum Framework provides a necessary foundation for building proficiency within the four additional Professional Scrum Competencies. Proficiency in this competency is relevant and required to effectively practice Scrum in any organizational role.

Within each competency, a number of Focus Areas provide a more detailed view of the knowledge and skills you require to master that competency.

- Empiricism
- Scrum Values
- Scrum Team
- Events
- Artifacts
- Done
- Scaling

empiricism /ɪmˈpɪrəˌsɪzəm/

A cornerstone to Scrum and Agile. A practitioner will be able to apply the concepts of the empirical process to the problems they encounter. That means they can describe problems in terms of learning, break problems down into the smallest increments that will generate valuable evidence, and execute in an empirical way. By learning and practicing the skills in this Focus Area, a practitioner will become an expert in the application of scientific methods to complex problems, understanding why and how to apply an empirical process.

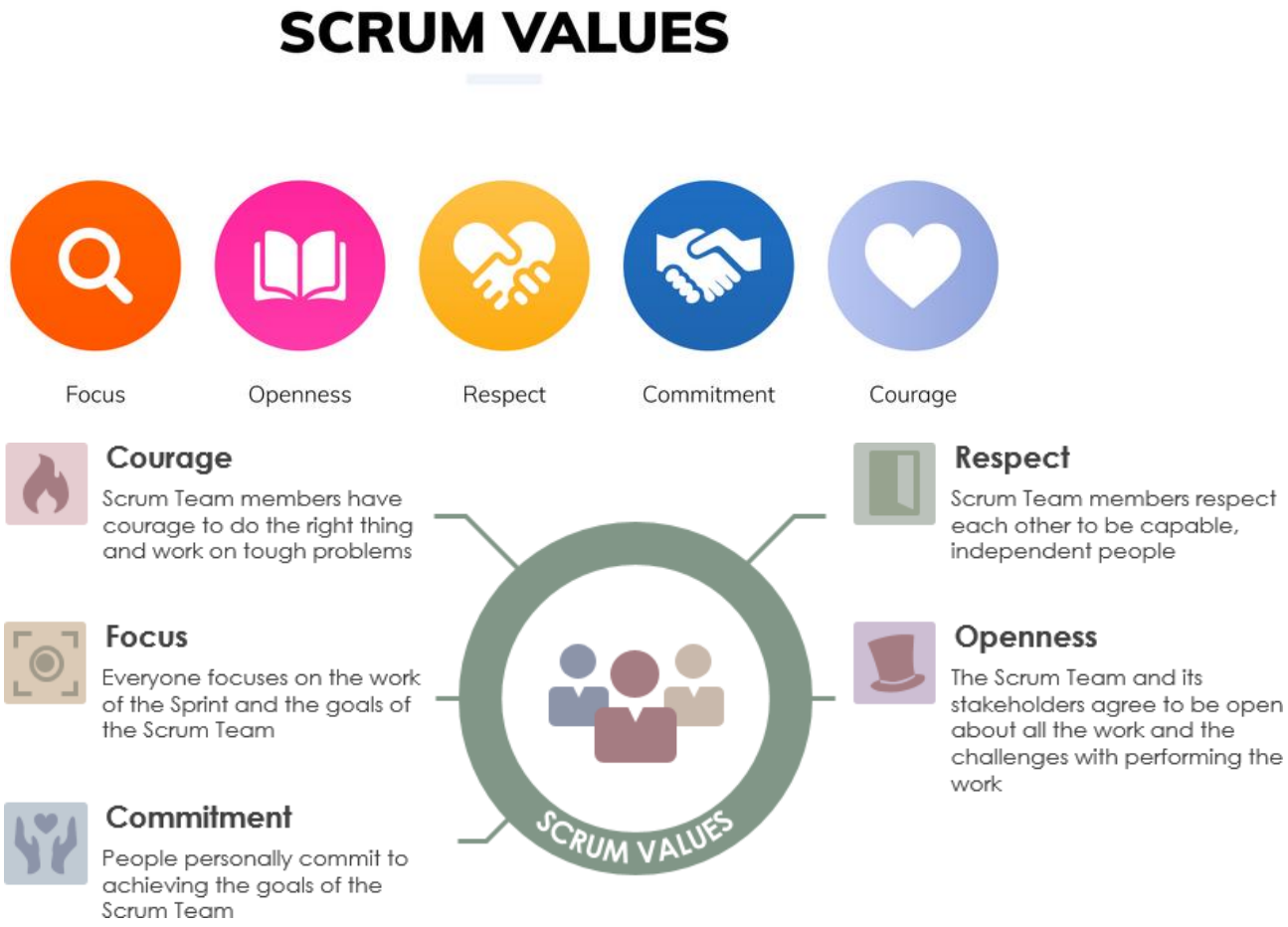
Scrum Values

For agility /əˈdʒɪləţi/ to thrive /θraɪv/, the culture of the organization must support the fundamental concepts of agility. A practitioner will understand both

the Scrum Values:

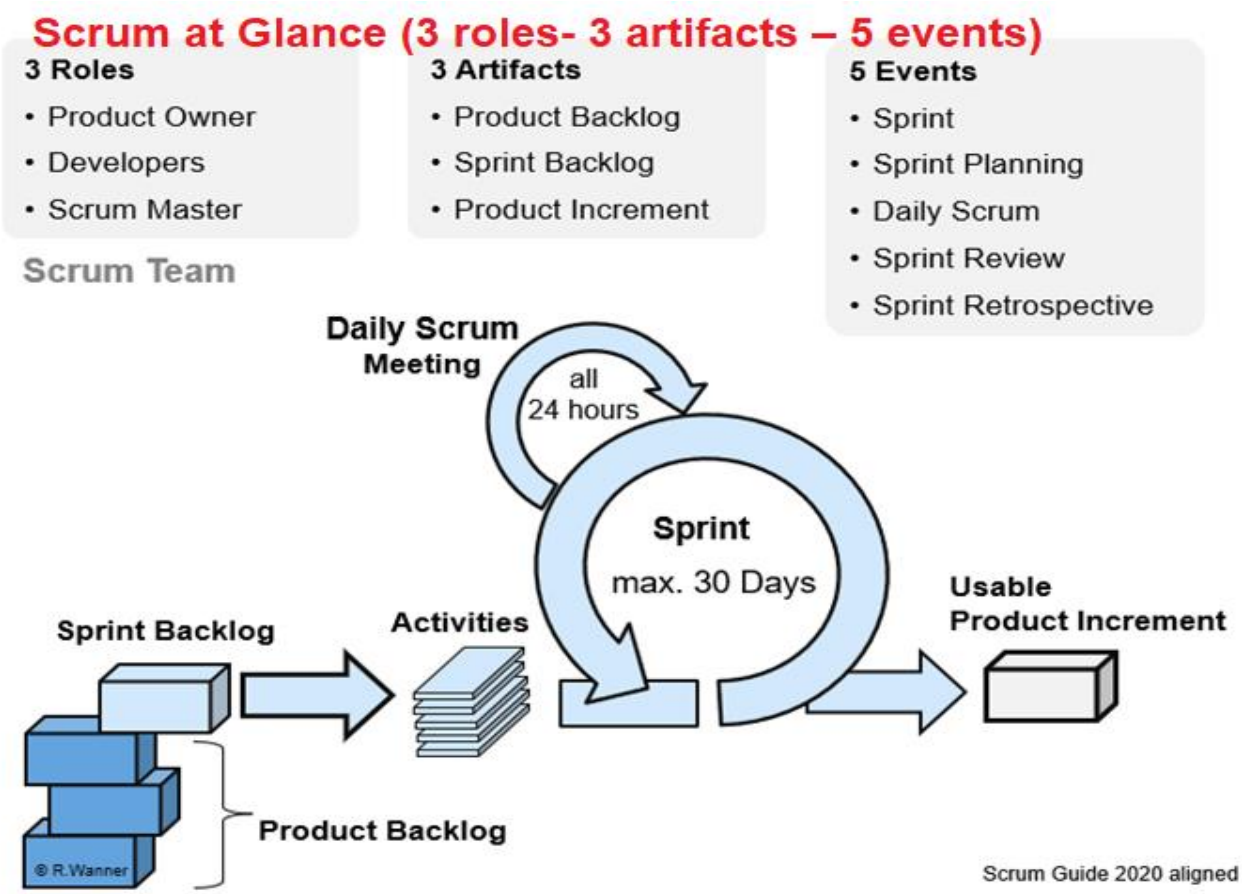
- *Courage →to do the right thing*
- *Focus →the work of the Sprint, goals*
- *Commitment →achieve a goal*
- *Respect, →each other to be capable, independent people*
- *Openness, →Scrum Team and its stakeholders*

and demonstrate that they can apply them in the reality of organizations whose values do not match those of Scrum.



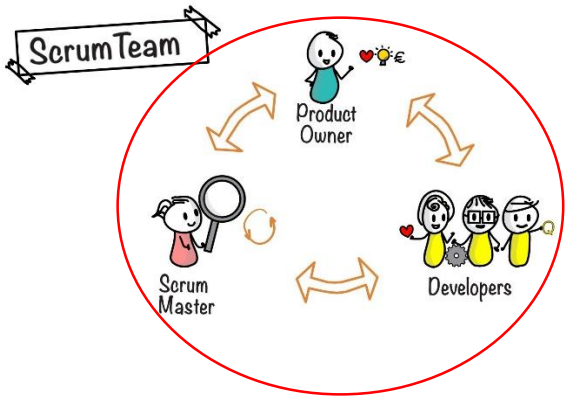
By living the Scrum Values and helping others to apply them, learners will create an environment where empirical process, self- organization, and continual improvement will be more successful.

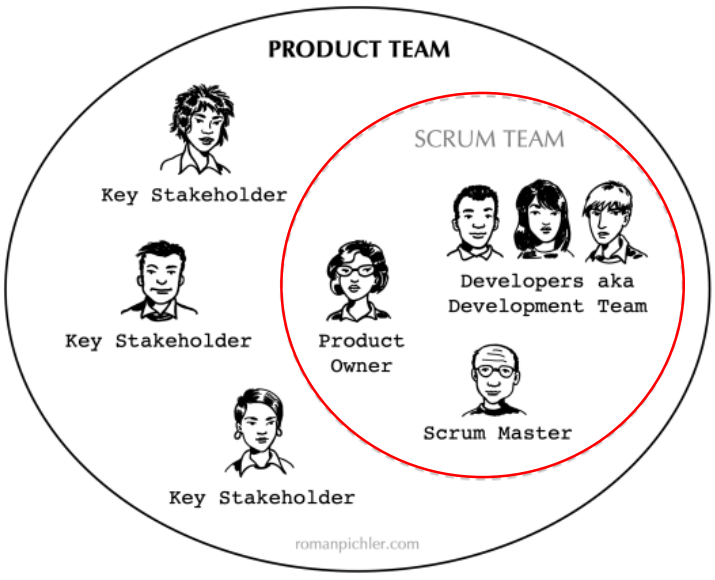
Scrum at Glance (3 roles- 3 artifacts – 5 events)

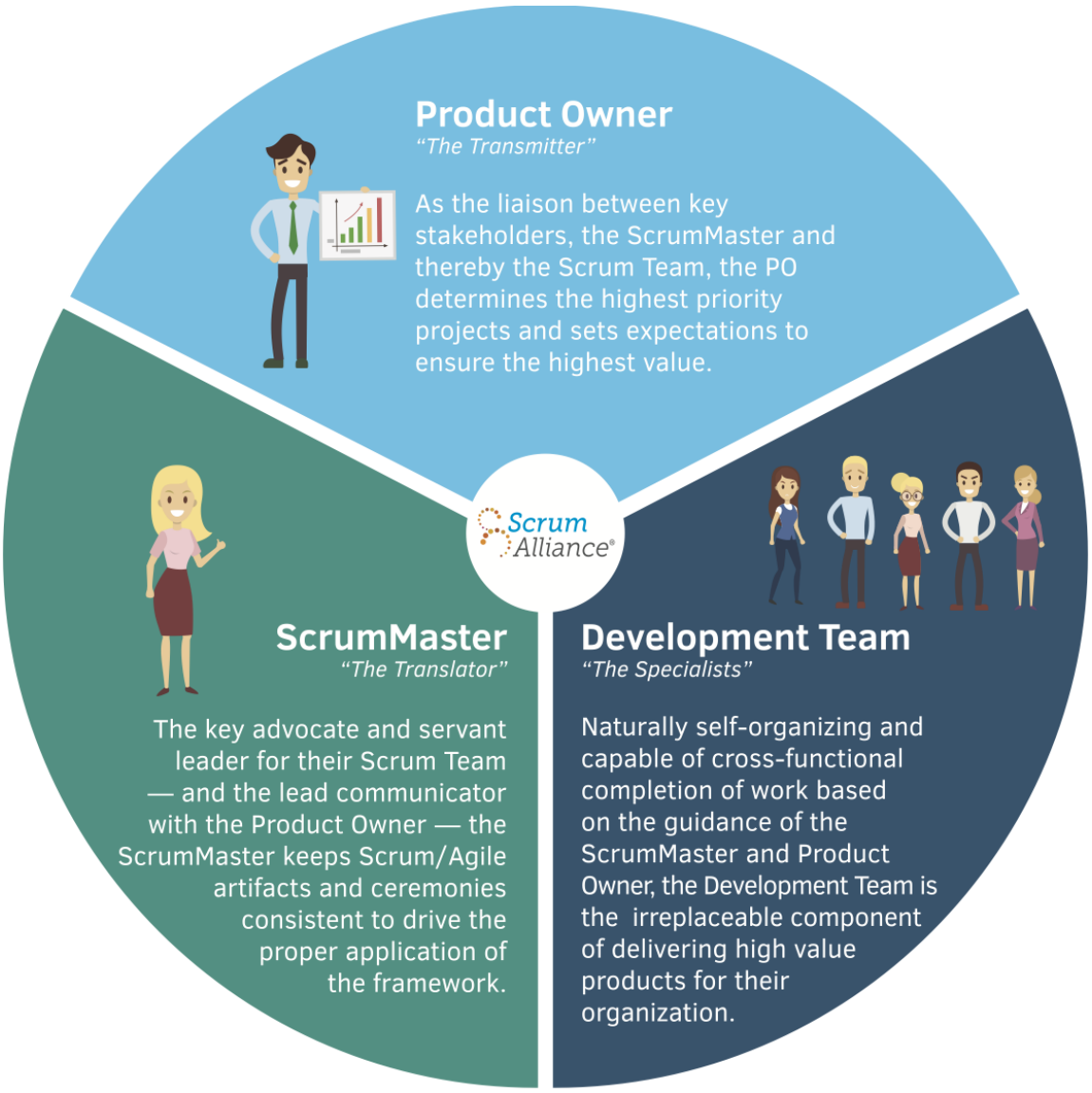


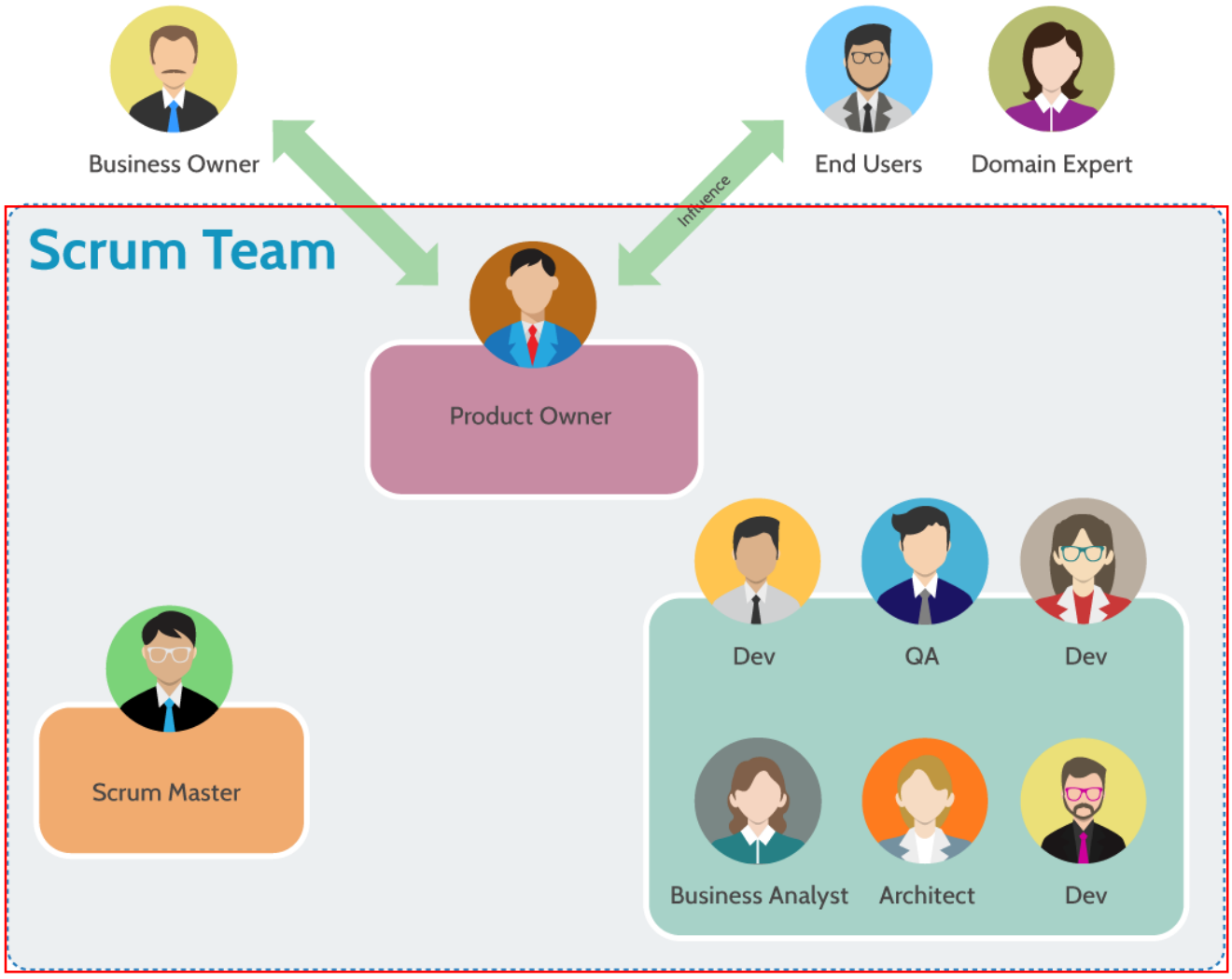
Scrum Team

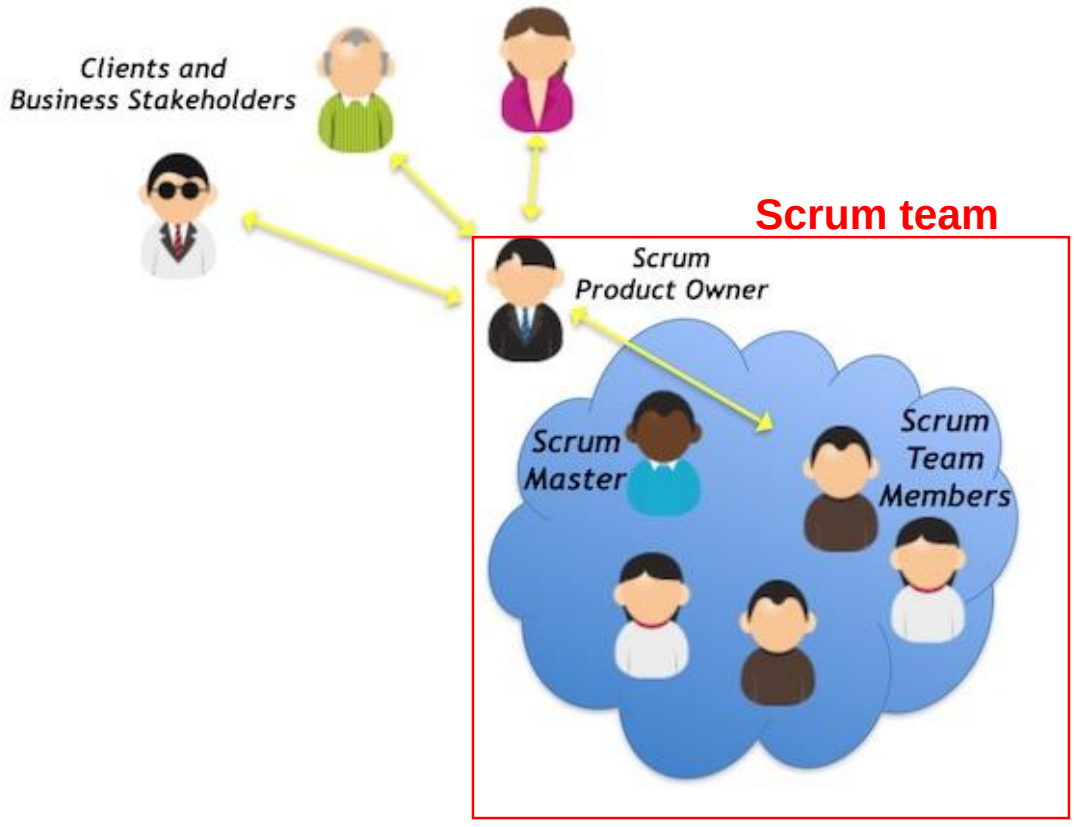
The Scrum Team consists of one Product Owner, one Scrum Master, and Developers (= Development team).



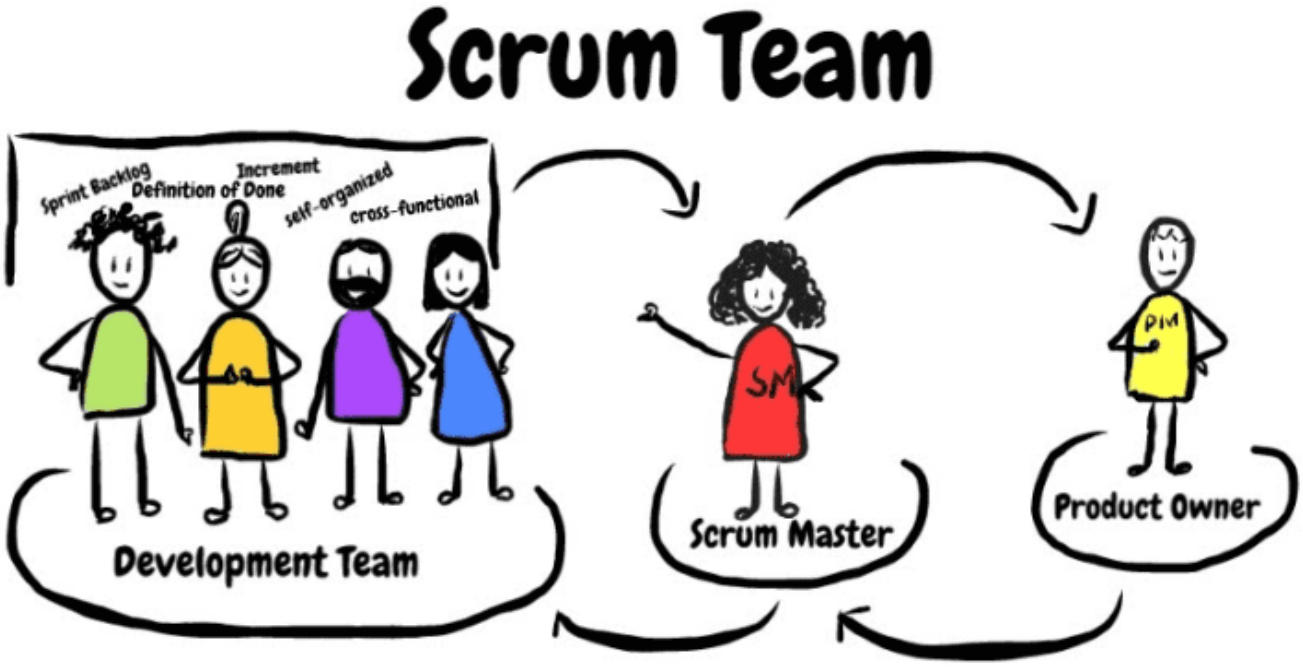


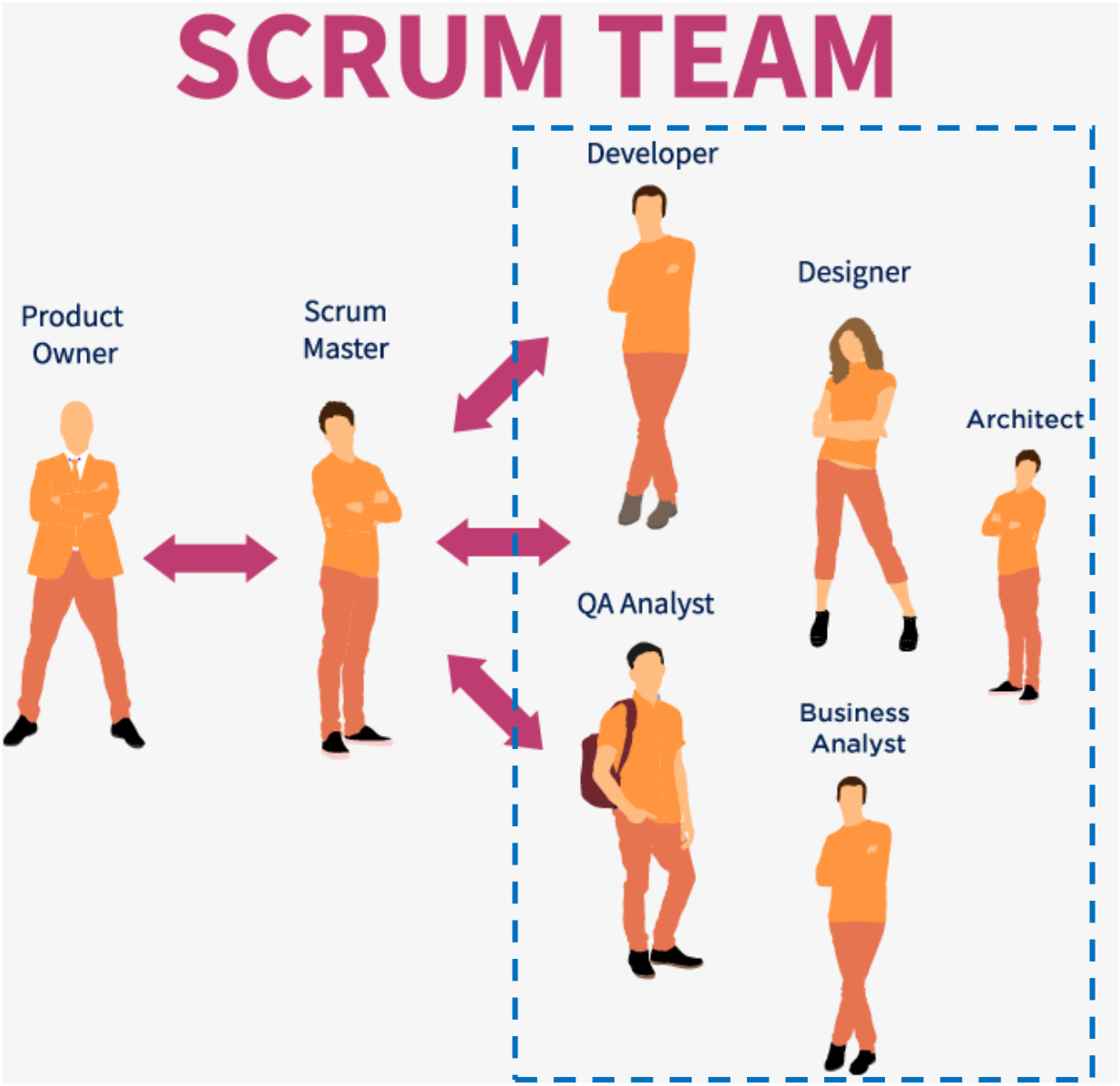

Product Owner
"The Transmitter"
As the liaison between key stakeholders, the ScrumMaster and thereby the Scrum Team, the PO determines the highest priority projects and sets expectations to ensure the highest value.
Scrum Alliance®
ScrumMaster
"The Translator"
The key advocate and servant leader for their Scrum Team — and the lead communicator with the Product Owner — the ScrumMaster keeps Scrum/Agile artifacts and ceremonies consistent to drive the proper application of the framework.
Development Team
"The Specialists"
Naturally self-organizing and capable of cross-functional completion of work based on the guidance of the ScrumMaster and Product Owner, the Development Team is the irreplaceable component of delivering high value products for their organization.

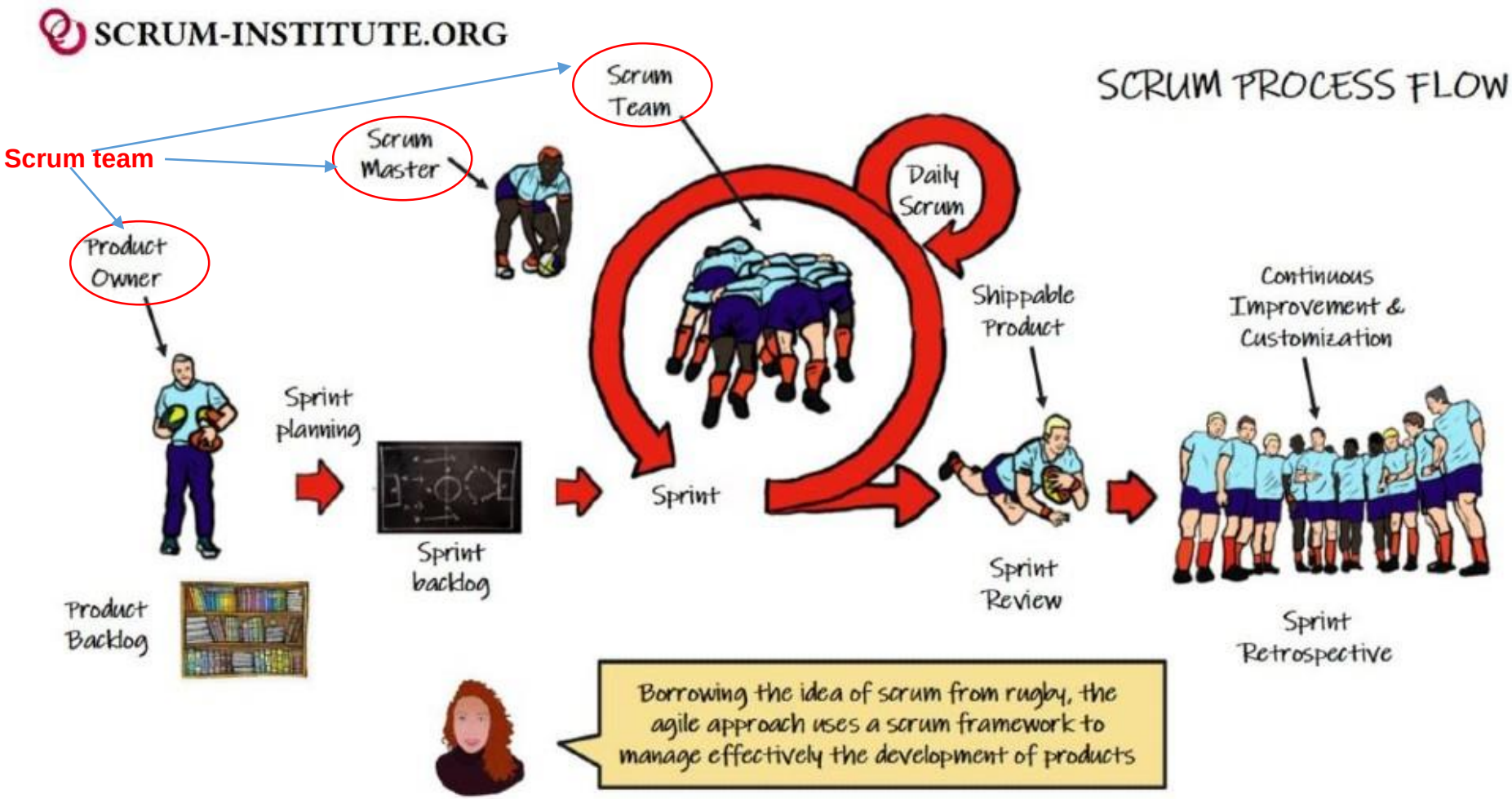

Business Owner
End Users
Domain Expert
Influence
Scrum Team
Product Owner
Scrum Master
Dev
QA
Dev
Business Analyst
Architect
Dev


Clients and
Business Stakeholders
Scrum team
Scrum
Product Owner
Scrum
Master
Scrum
Team
Members


Product Owner
- Decides what to do and why
Scrum Master
- Focus on how to work better
SCRUM TEAM ROLES
Dev Team
- Do the work development


Scrum Team
Sprint Backlog
Definition of Done
Increment
self-organized
cross-functional
Development Team
SM
Scrum Master
PM
Product Owner


SCRUM TEAM
Developer
Designer
Product Owner
Scrum Master
Architect
QA Analyst
Business Analyst

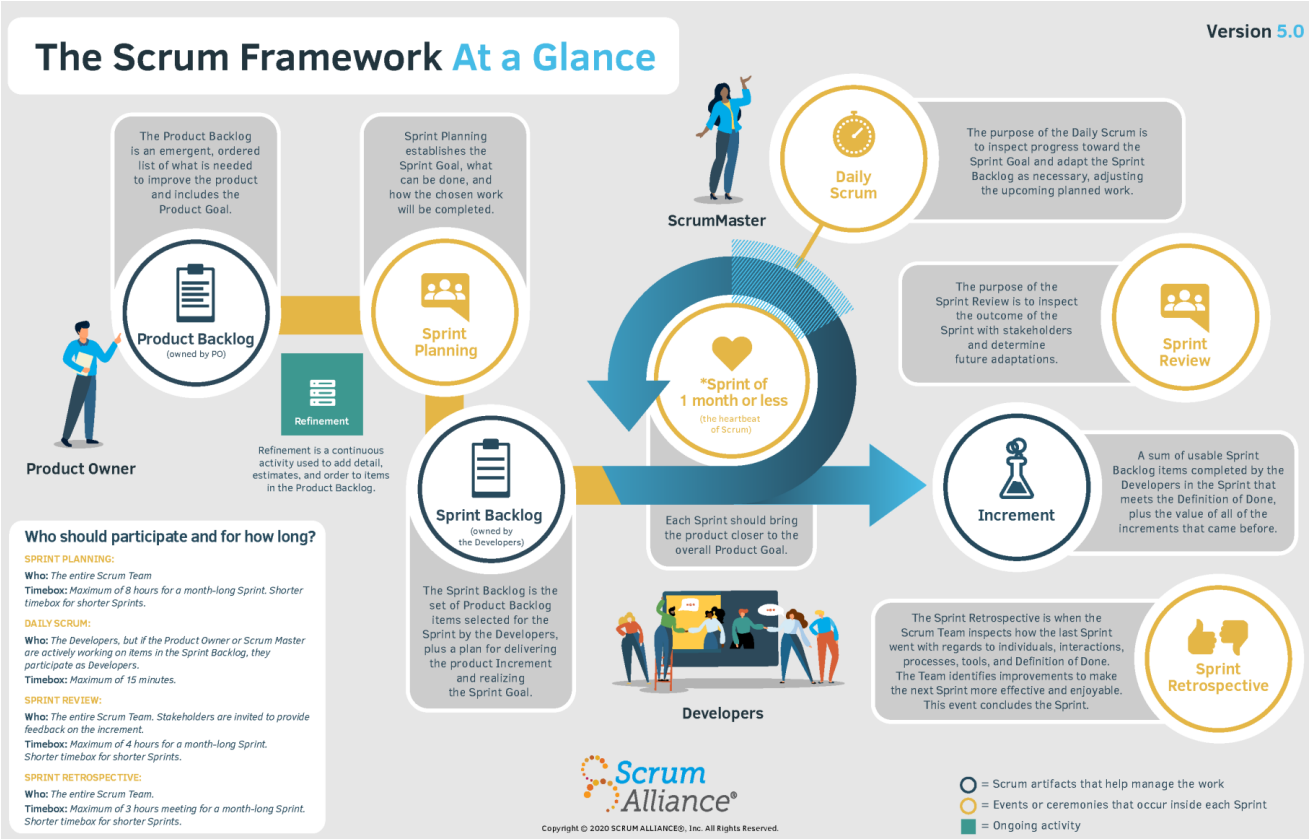

SCRUM-INSTITUTE.ORG
SCRUM PROCESS FLOW
Scrum team
Scrum Team
Scrum Master
Product Owner
Daily Scrum
Shippable Product
Continuous Improvement & Customization
Sprint planning
Sprint backlog
Sprint
Sprint Review
Sprint Retrospective
Product Backlog
Borrowing the idea of scrum from rugby, the agile approach uses a scrum framework to manage effectively the development of products

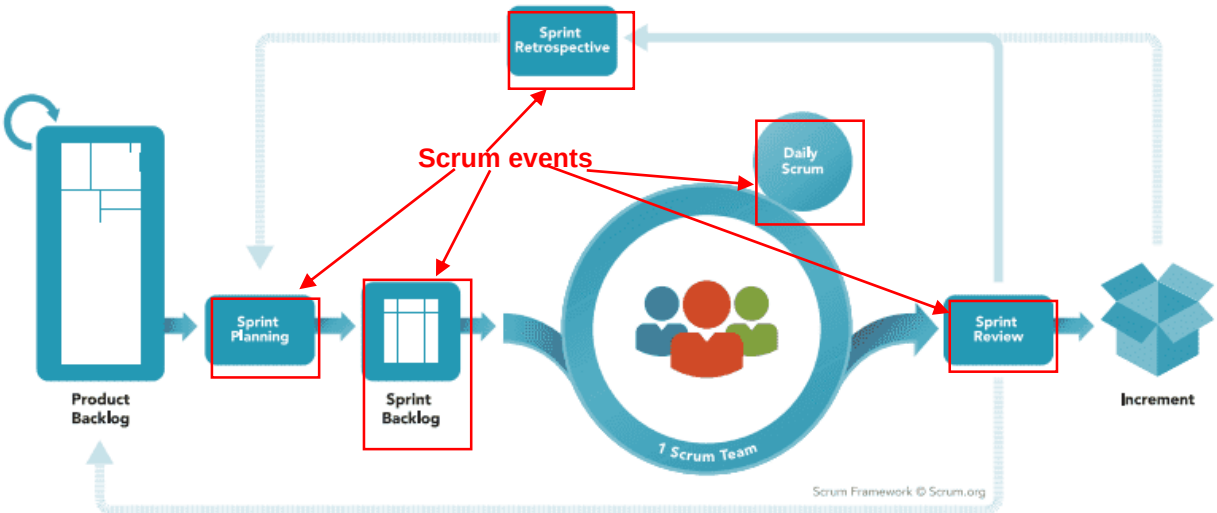
The skilled practitioner will understand how accountability /əˌkaʊntəˈbɪləţi/ is shared amongst team members and how they take on work in the context of their Product Goal.

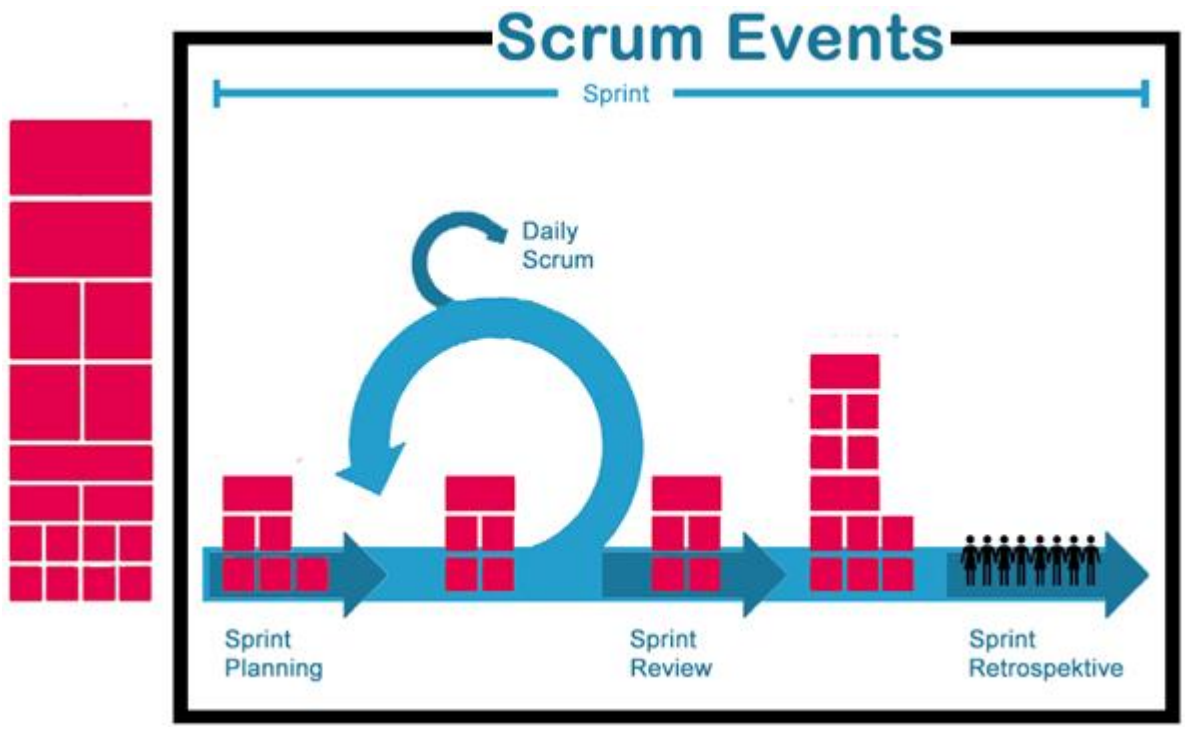
Events

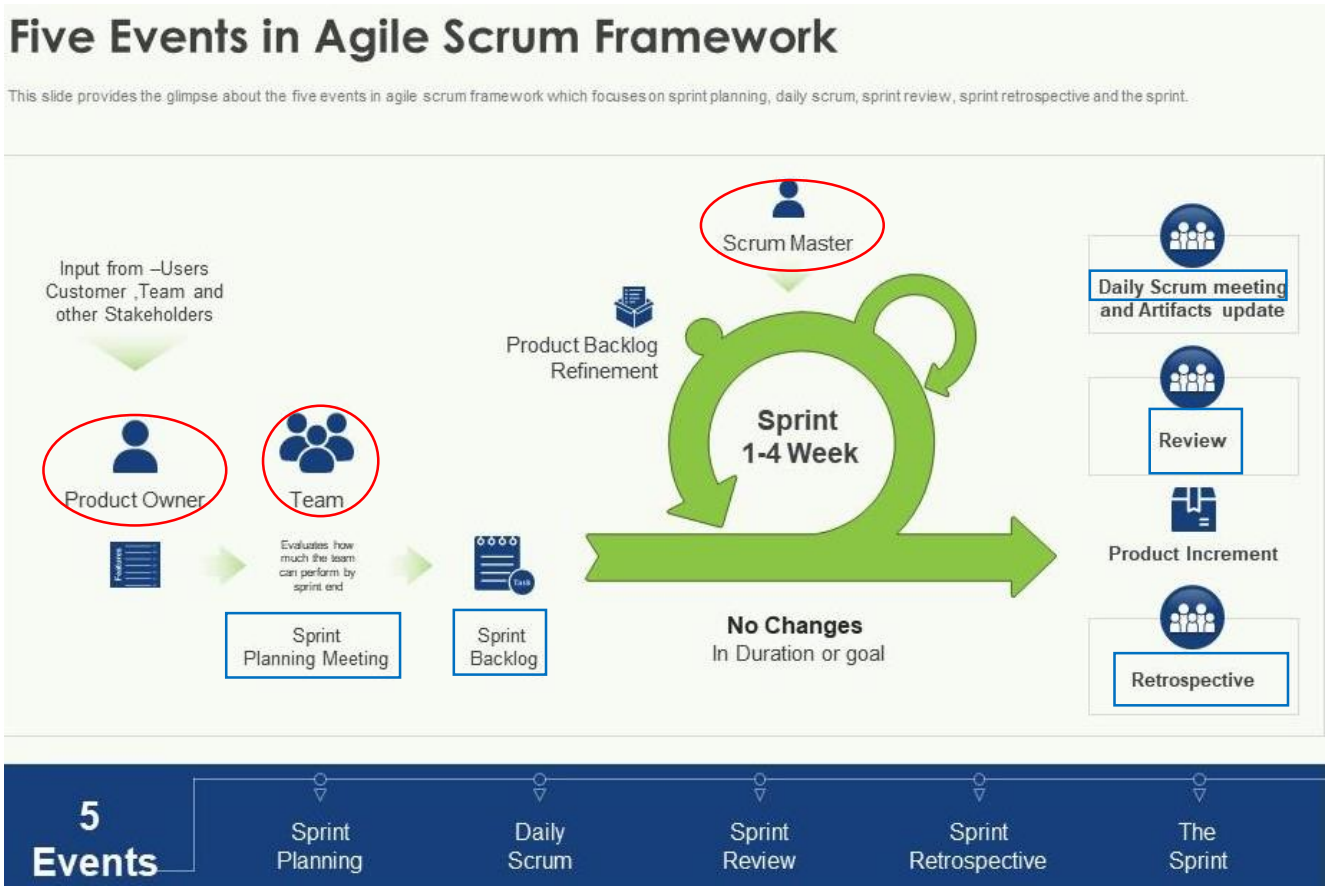
The Scrum framework describes 5 events:

1. Sprint Planning,
2. The Sprint,
3. Daily Scrum,
4. Sprint Review,
5. and the Sprint retrospective /ˌrɛtrəˈspɛktɪv/.







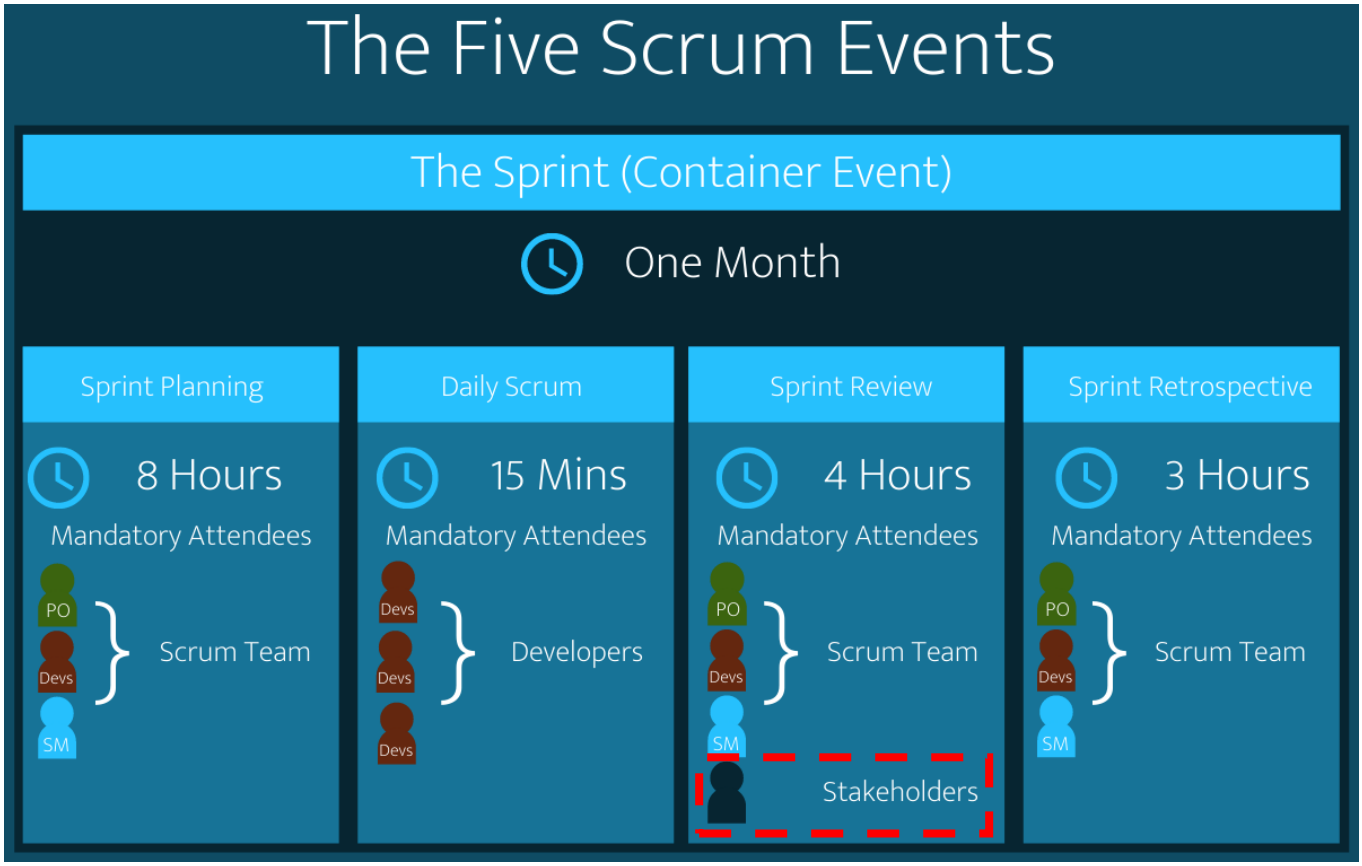


All events are time-boxed and enable progress through adaptation and transparency.

SCRUM EVENTS

Scrum Events Quick Reference

Event	Inspection	Adaption	Who Attends	Time-Box For 1 Month
Sprint Planning	Product Backlog	Sprint Goal, Forecast, Sprint Backlog	Scrum Team	8 Hours
Daily Scrum	Progress Toward Sprint Goal	Sprint Backlog	Development Team	15 Minutes (Always)
Sprint Review	Increment, Sprint, Product Backlog	Product Backlog	Scrum Team Stakeholders	4 Hours
Sprint Retrospective	Sprint	Actionable & Committed Improvements	Scrum Team	3 Hours



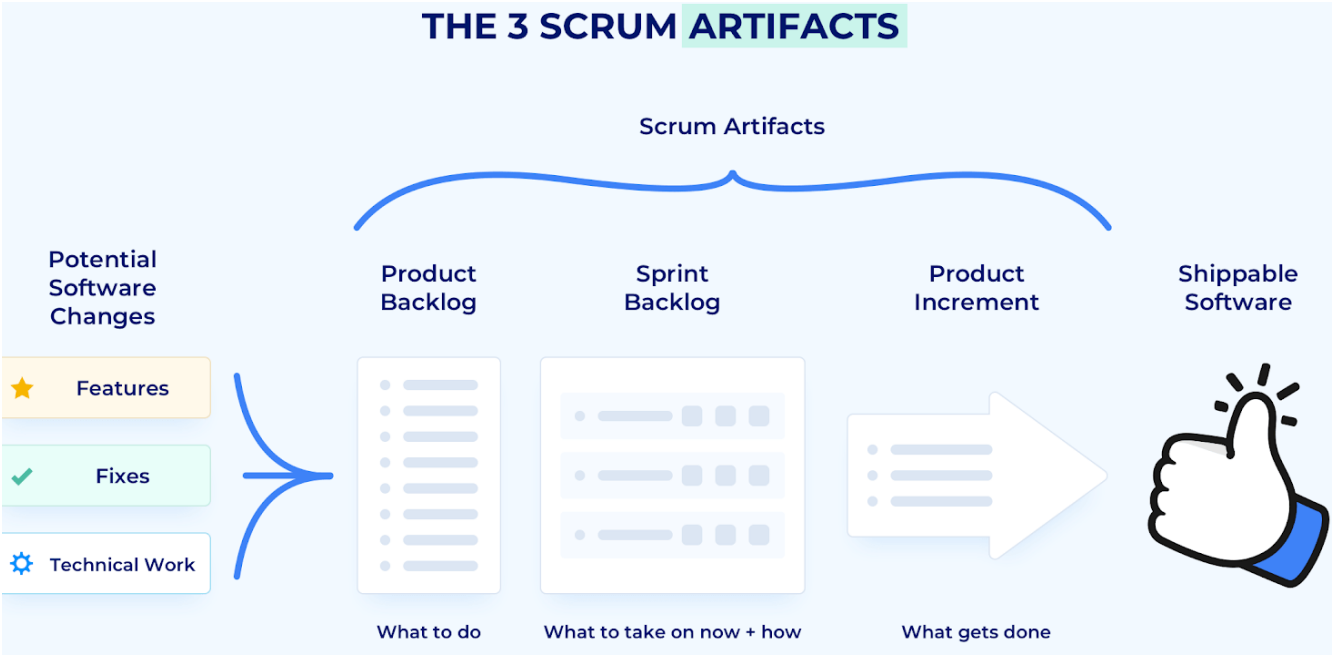
([link])

The practitioner will understand the events and be able to practice each event, but more importantly be able to apply these events in complex situations and at scale. The events are used to uphold empirical process control, through the three pillars of Scrum: transparency, inspection, and adaptation.

Event	Inspection	Adaptation
Sprint Planning	• Product Backlog • (Commitments Retrospective) • (Definition of Done)	• Sprint Goal • Forecast • Sprint Backlog
Daily Scrum	• Progress toward Sprint Goal	• Sprint Backlog • Daily Plan
Sprint Review	• Product Increment • Product Backlog (Release) • Market-business conditions	• Product Backlog
Sprint Retrospective	• Team & collaboration • Technology & engineering • Definition of Done	• Actionable improvements

Artifacts (Product backlog, Sprint backlog, Product increment)

The Scrum framework describes 3 artifacts. 1. Product Backlog, 2. Sprint Backlog, 3. Product Increment.	



These artifacts provide the team with a minimal set of materials to plan, execute, and review the Sprint. The Practitioner will understand these artifacts and how to implement them in complex, real-world situations. They will also understand the relationship of these artifacts relative to other practices and techniques and how to integrate them into an organization's own process.

Done

The objective of each Sprint is to deliver an Increment. The **Definition of Done (DoD)** provides a way for the team to make what done means transparent.

What Does in Mean to be DONE?

1. "Definition of Done" (DoD) decided on beforehand – along with acceptance tests
   - DoD can be standard across a group of common stories, or defined specifically for unique ones
2. Done means the feature has been developed, tested AND meets all required acceptance tests
3. Ideally, Done means the feature could be shipped to a customer
4. Product Owner officially "Accepts" Done features back from Team at the Sprint Review meeting

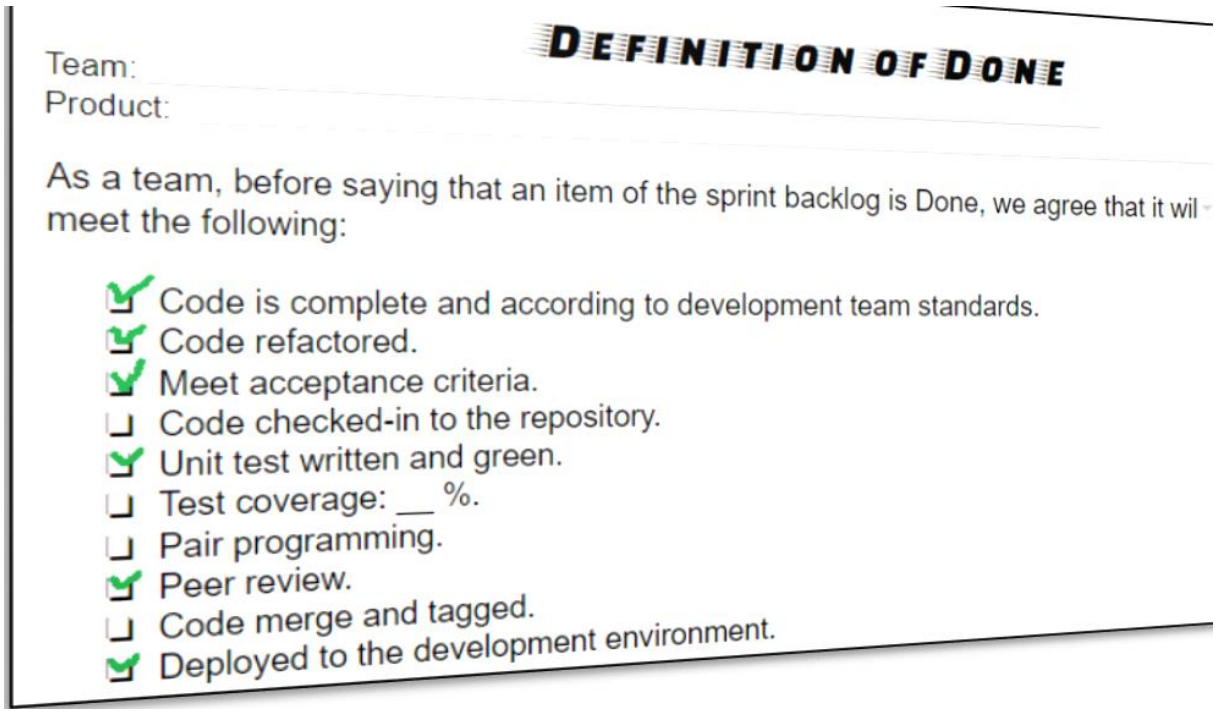
Sample Definition of Done (DoD)

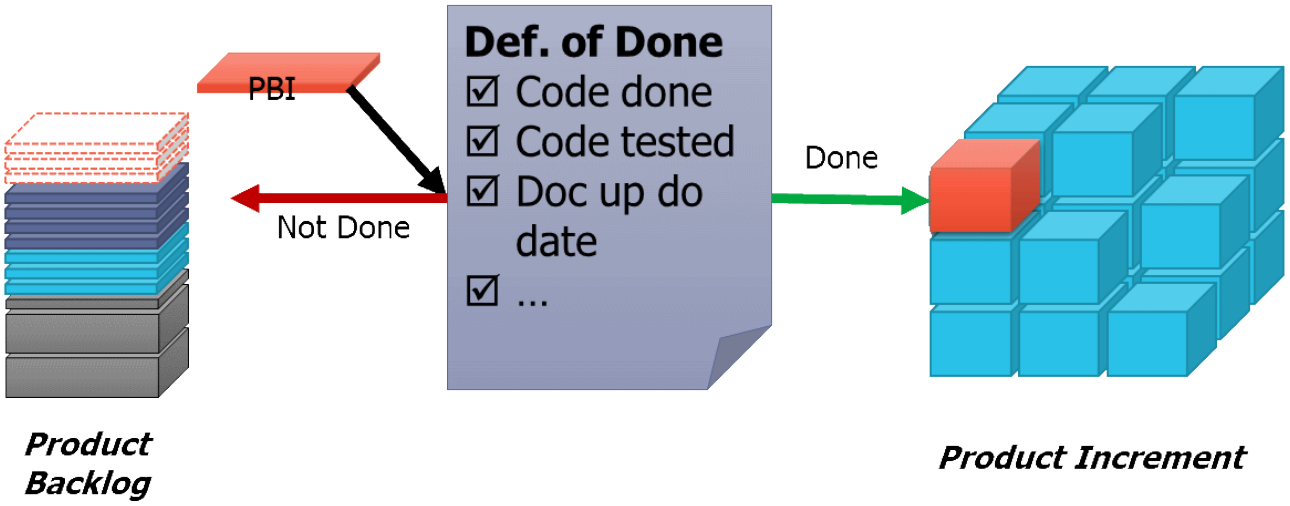
Development / Coder

- Code is written with unit tests
- Unit tests have a minimum of 75% code coverage
- Code has been merged to Main
- Code compiles and unit tests pass when run as part of an automated build
- Database schema objects are under source control
- Database upgrade script is under control
- Code reviewed by someone other than the original author

Testing, Deployment, Ops

- Written QA test plan
- Tested with QA test plan by someone other than the original author
- Deployed and tested in Staging environment
- Automated UI tests are written and pass
- No Severity 1 or 2 bugs
- Reviewed by Product Owner
- Passes acceptance criteria for the PBI
- Known deployment & rollback plan
- Deployment plan reviewed by Ops
- Database changes reviewed by DBAs
- Load tested
- Deployed to Production





In this Focus Area, the practitioner will be able to describe what a DoD is, apply it to their particular context, and understand how the DoD enables the benefits of agile. They will also be able to describe the implications of the necessary trade-offs and compromises required to deliver Increments within their organization.

Scaling

Scrum is designed to work at the team, product, and organization level. The practitioner will be able to apply Scrum in increasing levels of complexity and scale. They will be able to demonstrate when to scale and when not to scale and appreciate scaling practices and complementary frameworks that help organizations scale Scrum. The ultimate level of proficiency within this Focus Area is the ability to know what, and what not, to compromise in pursuit of a scaling approach by understanding the trade-offs and benefits of particular concepts and practices. Ultimately, the practitioner will demonstrate that they can scale Scrum and still keep its essential qualities of empiricism, self-organization, and continuous improvement. The practitioner should also be able to demonstrate the results of good scaling practices from both an organization and business perspective.

2/ Developing People and Teams

- Self-Managing Teams
- Facilitation
- Leadership Styles
- Coaching & Mentoring
- Teaching

Traditional management models consider the work of people and team development to be the job of the team's manager. Scrum puts specific responsibility on Scrum Masters to support and guide Scrum Team members (as well as other members of the organization).

However, team development is not only the responsibility of the Scrum Master. Since Scrum Teams are self-managing, all team members are responsible for helping the team continually improve through techniques that "**develop people and teams**."

This competency has six key focus areas, the last four of which are usually discussed together since they embody specific skills that are used in continuous self and team improvement:

- Self-Managing Teams
- Leadership Styles
- Facilitation
- Coaching
- Teaching
- Mentoring

Key Focus Areas

The Focus Areas provide a more detailed view of the knowledge and skills required to master the competency of Developing People and Teams.

Self-Managing Teams

A fundamental element to Scrum; cross-functional, self-managing and empowered teams are the engine to delivering value. Practitioners need to understand what self-management is and how to apply it to their context. They should also understand how to incrementally introduce self-management, the practices that can help it thrive, and the measures that help one determine if a team is able to be empowered to self-manage.

Leadership Styles

There are many different leadership styles ranging from traditional "command and control" to more collaborative approaches. Understanding the right style to use at a given time and how different styles can influence - in a positive or negative way - the agile agenda of empiricism, empowerment, and improvement is a key Focus Area. Practitioners should understand the concepts of leadership styles and be able to apply a particular style when the situation calls for it. They should also be able to demonstrate their ability to decide on the right style and understand its impact on the organization.

→Skills for Developing People & Teams: 4 skills

Each Scrum Team member brings a unique set of skills, knowledge and experiences to the team. The team, and the individuals in it, are made stronger when these points of view are brought together and spread throughout the team. Facilitation, coaching, mentoring and teaching, are all techniques for making this possible and they are key focus areas in the competency of Developing People and Teams.

For example:

- a Product Owner may facilitate a session with stakeholders to help unlock product innovations
- a stakeholder may coach a Developer to help them understand their career aspirations and how to achieve them
- a Scrum Master may teach Developers about the Scrum Framework
- a highly experienced Product Owner may mentor a less experienced Product Owner in the impact of ineffective stakeholder management.

By introducing themselves to these skills, learning the differences among them and developing proficiency with a few of their competencies and tools, Scrum Team members can help advance the way the team works together, as well as helping individual colleagues improve their ability to contribute to the team's success.

More info:

When to use Facilitation, Coaching, Mentoring or Teaching

We often use the words “coaching,” “mentoring,” “teaching,” and “facilitating” informally. We may say that we're being *coached* on how to become a better golfer, or that the person managing the agenda in a meeting is the meeting's *facilitator*. But once we investigate these practices, we discover that each has characteristics that are formally defined by professional organizations related to them, and that our casual uses of those words were probably incorrect in their eyes.

However, it’s not necessary for every Scrum Team member to become a certified facilitator, for example, in order to adopt a facilitator’s stance during Scrum Events. They should introduce themselves to each of these practices, understand the differences among them and learn a few of their techniques. With sufficient proficiency, Scrum Team members will find themselves identifying situations where they can utilize these techniques and fluidly adopt the appropriate stance*.

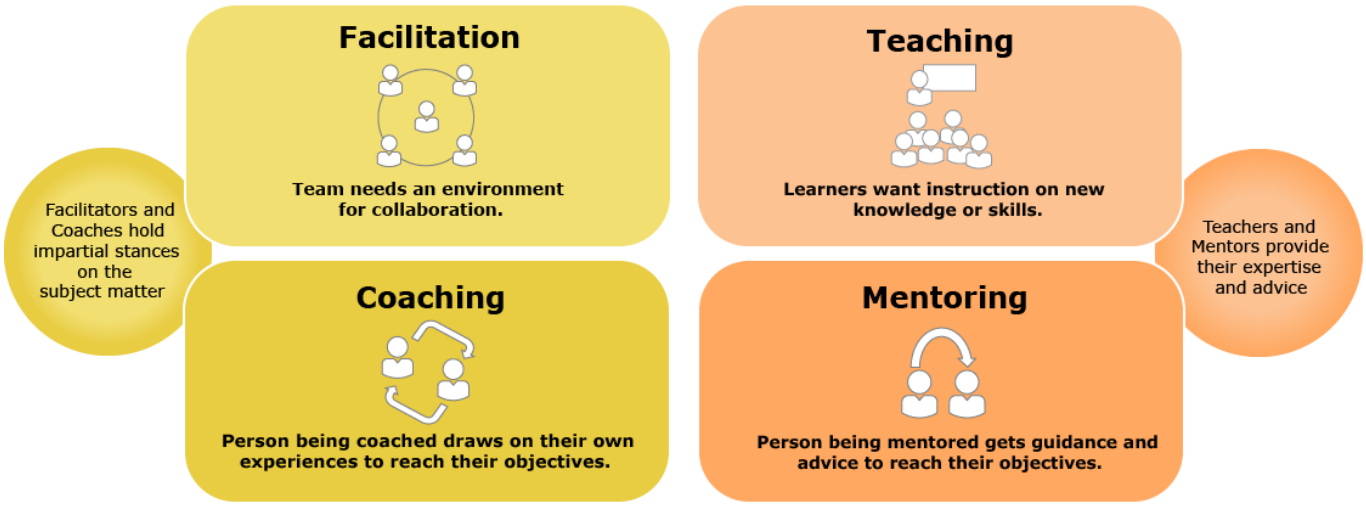
When should you either provide or seek out Facilitation, Coaching, Mentoring or Teaching? When there is one or more of the following:

- a challenge, goal or opportunity
- a gap in knowledge, skills or confidence in a particular area
- a lack of clarity when a decision must be made
- an unhealthy conflict among team members

There are countless other situations in which one team member can help others. Deciding which of these skills to use depends on:

- the situation and the needs of the person or team needing help
- the expertise and knowledge of the person providing the help.

The Situation and Needs of the People Needing Help



Facilitation /fəˌsɪlə'teɪʃn/ and coaching differ from teaching and mentoring in the neutrality of the provider. A teacher or mentor provides their expertise and advice to their student or mentee. In contrast, coaches and facilitators should remain neutral. That is, they hold an impartial /ɪm'pɑrʃl/ stance and do not take a position on the subject matter. Their job is to be a process expert, enabling those they are coaching or facilitating to achieve their goals.

The Expertise and Knowledge of the Person Providing the Help

While it's useful for Agile practitioners to learn about each of these skill sets, it's possible that they are not well-suited for all of them. For example, if someone cannot prevent themselves from injecting their points of view in the discussion, it may be that they are not well-suited to be facilitators or coaches. Similarly, individuals that are eager to impart their knowledge in a structured setting, may be more suited to teaching, rather than mentoring.

* *A note about switching stances: It is important that the individual discusses and is explicit about the stance that they are taking with the recipient(s) so that they know they are being coached, for example. If the situation requires a change in stances, such as the person coaching switches to mentoring, they must make that change clear as it occurs so as not to confuse the* recipient(s) /rɪˈsɪpiənt/.

Facilitation

It's easy to agree that we should make decisions, share ideas and work together to find new solutions. In reality, these are hard to do. Facilitation is a set of practices that help support the collaboration, communication, and creativity /ˌkrieɪˈtɪvəʧi/ of teams and individuals. The practitioner should understand the value of facilitation, and have a collection of techniques they can apply. They should also have experience applying them in different situations.

The facilitator creates a safe environment for the team to flourish. They hold an impartial stance and do not take a position on the subject matter. Their job is to be a process expert, enabling those they are facilitating to achieve their goals.

More info:

What is Facilitation?

Facilitation can be used to lead people toward agreed-upon objectives in a manner that encourages participation, ownership and creativity by all involved. A well-facilitated session can unlock collective intelligence and play an important role in providing opportunities for people to progress and succeed. Good facilitation enables transparency and collaboration, creates synergy /ˈsɪnərdʒi/ and leads to achieving a collective objective.

A facilitator plays an important role in helping people to understand and achieve their shared goals and objectives. They do this while remaining neutral and impartial. Facilitators enable a purposeful and participative environment in which people feel safe to engage, learn and collaborate. They encourage people to explore different perspectives, harness diversity and leverage collective wisdom.

Facilitation Principles

The Scrum Values are at the heart of a Professional Scrum Team, guiding them in their work, actions and behavior. Complementary to the Scrum Values are the facilitation principles of participa,tory /pɑrˈtɪsəpəˌtɔri/, healthy, transparency, process and purposeful /ˈpərpəsfl/. Falling back on these core principles can help facilitators work with teams to achieve objectives collaboratively in different situations. These principles can also help facilitators decide which facilitation skills and techniques might be appropriate and useful. This holds true not only when creating an energetic environment where the team is engaged and focused on achieving the objective together, but also when interactions do not go as expected.

	**Participatory** - Core to effective facilitation is full participation and engagement, which enables shared responsibility in a team
	***Healthy*** - A safe environment means creating a healthy space where people feel safe to raise differences and even conflicting perspectives while respectfully learning from each other
	***Transparency*** - Transparency only exists when there is shared understanding
	***Process*** - Facilitation should enable a team to progress toward the desired objective of the interaction in a way that is collaborative, inclusive and leverages diverse perspectives
	***Purposeful*** /ˈpərpəsfl/ - Well-facilitated sessions should have a clear objective that everyone is aligned with and works toward

Skills and Traits of a Facilitator

Facilitators can come from many backgrounds and have varying levels of experience. Great facilitators, however, demonstrate the following skills and traits:

Active Listening:	A facilitator has the ability to listen actively, and focus completely on what is said and what is not said. They lead by example, inspiring participants to both fully express themselves and engage in active listening when others are speaking.
Encouraging Curiosity:	A facilitator encourages curiosity and different viewpoints. They are skilled in asking powerful, often open-ended questions, in order to stimulate reflection and discussion.
Problem Solving:	A facilitator is skilled at applying group problem-solving techniques. They can help a group define a problem, reframe it as a clear problem statement and encourage the group to consider a range of solutions to the problem.
Resolving Conflict:	A facilitator recognizes that conflict among group members is natural and, as long as it's expressed appropriately, does not need to be suppressed. Conflict should be expected and dealt with constructively and respectfully.
Using a Participative Style:	A facilitator encourages all participants to actively engage and contribute in activities and discussions, depending on their individual comfort levels. This includes creating a safe and comfortable atmosphere, in which group members are willing to share their thoughts and ideas.
Encouraging Openness:	A facilitator encourages the group to be open to other people's ideas, suggestions and perspectives.
Empathizing and Showing Compassion:	A facilitator /fəˈsɪləˌteɪtər/ is understanding, aware and respectful of the feelings, perspectives or actions of others.
Demonstrating Leadership:	A facilitator leads a group of people to reach their collective goals and objectives.
Building Consensus:	A facilitator is skilled in helping groups to achieve general agreement.
Managing Time Effectively:	A facilitator keeps things on course while allowing flexibility. They focus on achieving the outcome within a timeframe instead of a strict agenda. Overly restrictive time management can stifle good, purposeful conversations and reflection, whereas a lack of time management can limit focus and progress.

Setting Objectives:	A facilitator communicates the purpose of a meeting in a clear and concise manner. This can be done by setting a strong overarching /ˌoʊvərˈɑrtʃɪŋ/ objective (often done in collaboration with the team) instead of focusing on a strict agenda.
Communicating Adequately:	A facilitator communicates effectively, using clear and concise language.
Being Organized:	Facilitation does not start or end with the act of facilitating a group of people. It includes preparation and following-up on decisions that were made.

Why is Facilitation Beneficial for Scrum Teams?

Open and respectful communication will help a Scrum Team thrive as a self-managing team. While members on a Scrum Team should talk to each other whenever they need to, Scrum assures communication points for the team in the Scrum events. Every event has a specific purpose and the team benefits from having these events facilitated with the desired outcome in mind.

Any person on the Scrum Team can facilitate the Scrum events. For example, Sprint Planning is more effective and exploratory /ɪkˈsplɔrəˌtɔri/ when someone on the team, acting as an objective facilitator, knows how to frame problems to understand how Product Backlog Items may be valuable for customers. A Developer may be a great person to do that, given their fa,miliarity /fəˌmɪliˈærəţi/ with the product.

Often, Scrum events don't go as planned. Good, lightweight facilitation can help the Scrum Teams get back on track.

For example, if the Scrum Master observes that the team continually uses the Daily Scrum as a status update instead of an inspection of progress toward the Sprint Goal, then the Scrum Master could help team members to focus by reminding them of the purpose of the event.

This will encourage team members to shift their focus from tasks to how they can collaborate toward achieving the Sprint Goal.

Taking on a facilitator stance is also valuable for a Product Owner, especially at the Sprint Review when the Scrum Team and stakeholders inspect progress toward the Product Goal, gather stakeholder feedback and adapt the Product Backlog accordingly. When done well, the Product Owner and Developers can learn and hear different opinions from the stakeholders. When not done as well, the Product Owner risks anchoring /ˈæŋkər/ or limiting the information gathered, reducing the effectiveness of the Sprint Review.

Learn more about **facilitation techniques for Scrum events**

Learn about **facilitating diverse perspectives** in Scrum Teams

Explore more **blogs and resources about facilitation** from Scrum.org

Coaching

The ability to unlock new ways of thinking is important for any agile practitioner who aims to enable sustaining change and transformation within teams and organizations. Coaching is a non-directive way of meeting a person where they are, helping them connect with their existing talent and wisdom and leading them to find within themselves what it takes to try new things. Agile practitioners should understand the range of skills and capabilities needed to create value when coaching and be able to discern /dɪˈsərn/ when a coaching stance is the best way to support people and teams.

The coach unlocks a thought-provoking process that helps individuals and groups draw on their own experiences to reach their objectives. The coach holds an impartial stance and does not take a position on the subject matter. Their job is to be a process expert, enabling those they are coaching to achieve their goals.

More info:

What is Coaching?

In an agile context, the word "coaching" is used in several ways:

1. **Agile Coach** - Among agilists, the word "coach" is frequently used to describe the role of an "Agile Coach." This is someone who uses agile approaches to help people and teams reach their objectives or overcome challenges. An Agile Coach doesn't just coach per se /pər ˈseɪ/; they also advise, lead, facilitate and teach the teams they are helping.
   Learn more about Agile Coaching in Professional Scrum
2. **To describe the discipline of "coaching."** Informally speaking, someone who coaches provides guidance to help others achieve their professional or personal goals. There are also professional organizations that provide prescribed /prɪˈskraɪb/ courses of study to become a certified coach. These organizations have formal definitions of what coaching entails /ɪnˈteɪl/.

When we refer to coaching as a Professional Scrum competency, we're generally discussing the discipline of coaching (not the role of an Agile Coach) and how coaching skills can be used to increase effectiveness and outcomes of a Scrum Team and its members.

The coach's job is to be a process expert, enabling those they are coaching to achieve their goals using skills such as developmental conversations, active listening and asking thought-provoking questions. Strictly speaking, coaches take a neutral stance with regard to how the person being coached archives their goals; they do not share their experience, advice and opinions. (This is in contrast to "mentorship," where the mentor DOES share their experience, advice and opinions).

We're providing the following information so that you can learn more about the discipline of coaching and determine if further study is interesting to you. However, even if your interest is less formal, the following may provide inspiration /ˌɪnspəˈreɪʃn/ for improvement of your coaching skills.

Coaching Principles

The various certifying bodies for coaching have strict guidance on how coaching is done and how a coach interacts with those they are coaching. The following are some of the elements of successful coaching:

Intentional -	When a practitioner is coaching they are deeply involved and interested in a person or team's ultimate success. However, coaches do not have their own agenda, they act on the goals set by the person or group they are coaching.
Neutral -	The coach remains unbiased and non-judgemental about the subject matter. They help people achieve their goals without steering /stɪr/ them. Those being coached are guided to draw on their own experiences and capabilities to overcome challenges, rather than learning directly from the coach's experiences. (This is a key differentiator from mentoring where the mentor actively provides advice based on their own experiences).
Agreed -	Formal coaching requires permission or consent in the form of written agreements. These agreements include the goals of the engagement and coaching approaches, clearly distinguishing between what coaching is and what it is not.

Systemic -	Coaches listen actively to what is being communicated. They seek to find and allow for the voice of the system (or the whole group, or the unsaid) to be heard. Coaches notice trends and patterns in behaviors and interactions, then reflect them back to enable richer communication and more effective decision-making processes.
Ethical -	Coaches create and manage a unique, safe and inclusive space. For this reason, it's imperative that they maintain integrity and confidentiality. For more information, we encourage you to explore the International Coaching Federation's Code of Ethical Conduct as well as the Agile Alliance's Code of Ethics for Agile Coaching.

Traits of a Coach

Successful coaches demonstrate capabilities from both agile and coaching areas of expertise. In order to have a holistic approach to coaching, we suggest agilists develop proficiency in the following areas:

- **Supports the Team's Self-Management** - Self-management is founded on the assumption that everyone on the team has valuable ideas and is responsible for their own outcomes. When coaching, you have a responsibility to help uncover ways of developing self-management, including your own.

- **Models the Scrum Values** - In the context of Scrum, sharing common values is central to building an environment of trust where people feel encouraged to inspect and adapt toward shared goals. When coaching, you have a unique opportunity to model the Scrum Values and bring awareness of their merit to the team. Here are some examples to draw on:
    - *Courage*: Challenge new ways of thinking and offer support in exploring them.
    - *Commitment*: Encourage commitment to action and help build the right plan for accountability.
    - *Focus*: Focus the team on the topic, and bring their attention to emerging thoughts, and inquire how they want to proceed.
    - *Openness*: Model and promote openness, creating a safe and open space where everyone feels comfortable to fully express themselves.
    - *Respect*: View your colleagues as creative and resourceful. Allow their solutions, ideas, and thoughts to emerge throughout the conversation.
- **Navigates Complexity… in Human Relationships** - People are complex. Navigating this complexity requires emotional/social skills like listening, empathy, and building shared understanding.

- **Encourages the Team's Growth through Empiricism** - Empiricism is the growth of knowledge through experience. Empiricism is foundational to Scrum. When we help the team focus on a defined problem, apply small changes and use the evidence of what happened to build their understanding, they generate new solutions and options for themselves.

Coaching Skills

Coaching requires you to leverage a wide range of skills. Successful coaching requires you to work with what is present and make choices in the moment. This means developing proficiency in many skills:

- **Listening Actively** - Recognize that people communicate with more than their words. Their context, identity, environment, experiences, values and beliefs must be taken into account. Pay attention to

their body language, mood, emotion and hear what is unsaid through their words and actions.

- **Asking Powerful Questions** - Ask questions that are open-ended (cannot be answered by “yes” or “no”), neutral and short. The key is to choose questions that provide the opportunity for the person you are coaching to gain new insight or reframe their perspective. Powerful questions are best applied when you are also listening actively.

- **Reframing** - Invite the person you are coaching to take an alternate point of view to broaden their range of solutions and consider multiple perspectives.

- **Reading and Working with the Emotional Field** - Coaching requires that you actively monitor the atmosphere, energy, or mood of the coaching space. As it changes during the coaching session, try to bring curiosity and reflection on it into the conversation to deepen an understanding of what's going on. The emotional field is a phrase commonly used by practitioners who have studied Organization and Relationship Systems Coaching (ORSC)™.

- **Normalizing** - Create a feeling that the person you are coaching is not alone in the challenges they face. When used well, normalizing has the ability to reduce tension or frustration and open people up to new ways of thinking.

- **Supporting with Silence** - Sometimes called “awkward silence,” this is a situation where you and the person you are coaching are both quiet and present, waiting for the wisdom and intelligence of the person being coached to emerge.

- **Taking a Meta-View** - Seek to observe the situation from the highest perspective to see the whole picture, uncover new information and connect what previously seemed to be disparate /ˈdɪspərət/ information.

- **Holding Things Lightly** - When you hold something lightly you acknowledge its presence and avoid allowing the topic or information to consume the conversation or overshadow the topic at hand.

- **Bottom-lining** - Be clear and precise with your words in order to create a “mic-drop” moment. Avoid over-explaining or asking long questions that interweave /ˌɪntərˈwiv/ too much context.

- **Forwarding the Action** - Be intentional about how the person you’re coaching will move forward in action as a result of the coaching conversation.

Why is Coaching Beneficial for Scrum Teams?

Knowledge workers, particularly Scrum Teams, who are constantly navigating change and being challenged to innovate, can often benefit from the guidance of an external, professional coach.

Scrum Masters often take the role of a coach with their teams. However, any member of the Scrum Team can use coaching skills informally or organically to help the team improve its effectiveness.

Examples include:

- Using coaching skills during the Scrum Events - any team member can use coaching skills to help the team use their knowledge and experience to determine next steps
    - *Sprint Planning* - ask **powerful questions** to help the team determine how the work for the Sprint is going to be done.
    - *Daily Scrum* - use **active listening** to uncover what challenges may not be expressly articu,lated /ɑrˈtɪkyəˌleɪʈəd/ during the Daily
    - *Sprint Retrospective* - use **normalizing** when the team is facing challenges in working together. It may be important for them to understand that they are not the only team to face these challenges.
    - *Sprint Review* - use **reframing** to encourage stakeholders to consider a new perspective

- Using a combination of coaching skills to manage the complexity around collaborating or communicating within and outside of the team, particularly with building relationships with stakeholders

Teaching

The ability to inspire others to learn and share information in an effective, repeatable, and efficient manner is a key aspect to any agile practitioners' skills. The practitioner should understand the value of teaching and appreciate the means of measuring the success of their teaching. They should understand different learning approaches and understand when to apply different techniques in different contexts.

The teacher shares knowledge, provides information and helps learners develop new skills. A teacher does not remain neutral, they provide their expertise and advice to the learner.

More info:

There are many ways for a team member to learn from another team member. They can:

- they can simply observe a colleague,
- ask a more senior member of the team to *mentor* them,
- they can engage with someone who acts as a *coach*,
- or they can ask a colleague to *teach* them something that they'd like to learn.

What is Teaching?

At the highest level, teaching is about transferring knowledge from the teacher to the learner so that the learner acquires new knowledge or skills.

You, or anyone on the team, can act as a teacher, helping your colleagues obtain new knowledge or learn new skills. However, if you want to become a very effective teacher, it's best if you learn a few of the principles of the teaching profession.

Scrum.org recognizes that those studying to become professional teachers or trainers go through an extensive learning and certification journey. Our goal in these pages isn't to replicate those learning journeys, but instead to make Agile practitioners aware of what the practice entails so they can incrementally improve their teaching ability and seek out further resources.

Teaching Principles

It's easy to fall into the trap of thinking that if you want to teach something to someone, you can simply stand before them and lecture them on the topic. Particularly in a professional setting, that's neither very effective nor well-received. So, how can you become a better teacher?

Scrum.org suggests that at a minimum, you become familiar with how people learn, how to effectively convey information and how to assess that the learner's objectives have been achieved.

Learners learn differently

Certified teachers and trainers delve /dɛlv/ deeply (= *to try hard to find out more information about something*) into cognitive /ˈkɑgnəʇɪv/ science and learning theories. As someone who just wants to do a good job teaching something to a colleague /ˈkɑlig/, you should be aware that the way you may initially want to teach something may not be the way your colleagues will best learn it. There are many, sometimes conflicting, theories about learning styles. Whatever the differences may be, there seems to be general agreement that using different teaching methods is important for helping learners be successful.

The following is a short list of teaching methods and models. It's not meant to be a formal education on teaching methods, but rather to spark your creativity:

- **Lecture:** There are learners who benefit most from a more traditional style like lecturing. This involves speaking to the learner(s) about the topic while they possibly take notes to reference later. Learners can then ask clarifying questions at the end of the lecture or at some point afterwards.
- **Self-study:** Learners who benefit more from watching videos and reading any materials that are provided to them are likely to absorb the information better while engaging with it on their own. It allows them the time and space to understand the topic at their own speed.
- **Hands-on:** Many people prefer to learn by doing. Having a learner interaction with the concept through activities may prove useful to them and make it easier for them to learn the material.
- **Assignment-based:** This method helps people who learn by doing but prefer to do it on their own or in a small group. The method involves teaching the subject to the learner(s) before either giving them an assignment that requires them to apply the concepts or asking them to teach the subject back to you. It demonstrates that they were able to properly understand what was taught to them and how it is applied.
- **Experiential** /ɪkˌspɪriˈɛnʃl/ **Learning:** In this model, learning is a cycle that starts with hands-on learning; followed by reflection and thinking about the experience; which leads to experimenting/ɪkˈspɛrəmənt/ with what was learned. This style is particularly well-suited to Agilists for whom experimentation/ɪkˌspɛrəmɛnˈteɪʃn/, reflection and adaptation are second nature.

Instruction should be designed around learning outcomes

As Agilists, we feel strongly that the work we do should be based on the outcomes we seek to create. In teaching, we start by determining what learning outcomes the learners need to achieve. Once that's established, we design the content of the instruction in a way that achieves each of those goals.

Assessment of the impact of the lesson helps the learner and the teacher

Frequent feedback is a key tenet /ˈtɛnət/ of Scrum. It helps the Scrum Team either continue on their current course, or adapt and change their approach. The same concepts hold true for teaching. When a learner is provided the results of an assessment, they can uncover whether they have achieved their learning goals, or if they need to consider another approach to learning.

Just as Scrum includes the Sprint Retrospective to assess whether the team used methods that were effective in achieving the Sprint Goal, a teacher should spend time reflecting on whether their methods were effective in helping the learners achieve their learning objectives.

Skills and Traits of a Teacher

Teachers come from varying levels of experience and different backgrounds. However, great teachers often have the following qualities:

- **Humility** /hyuˈmɪlətɨ/: Someone who is teaching should have a humble attitude in order to make themselves more approachable and provide a safe environment for learning. →*khiêm tốn*
- **Subject Matter Knowledge:** Teachers must have confidence in the subject they're teaching. Being able to answer questions without preparation will help teach as well as give learners confidence in the knowledge they are receiving.
- **Patience:** Teachers should have patience with their learners. Creating an environment that is not welcoming to those who may need more help will discourage them from learning.

- **Empathy** /ˈɛmpəθi/: Being understanding of those who are learning is an important quality when teaching. A teacher should be able to put themselves in the position of the learner in order to assess whether or not they are teaching effectively /ɪˈfɛktɪvli/.
- **Adaptability:** When teaching, one should be able to change their teaching style when needed. If a concept they are teaching is understood faster than expected, move on to the next portion. If the concept is not well understood, shift to another style of teaching to accommodate /əˈkɑməˌdeɪt/ the learners or material. --> *to help someone by doing what they want*

When Can Teaching Be Helpful for a Scrum Team?

There are many ways that teaching plays a role in helping a Scrum Team grow, improve and have a shared understanding. Examples may include:

- A *Scrum Master can teach a Scrum Team* about complexity theory and its connection to the fundamentals of the Scrum framework
- A *Scrum Master can teach a new Product Owner* how to create and order a Product Backlog based on factors such as priority, risk, value and dependencies
- A *Developer teaches their team* about concepts they learned in their UX course that they think will be helpful for his/her team to consider when building the product
- A *Product Owner teaches other Product Owners* in a community of practice on how to use a lean canvas when pursuing new initiatives
- A *Product Owner teaches customers and stakeholders* about the product

Mentoring

There are many ways to support people in their personal and professional growth. Mentoring is particularly useful because it brings forward personal stories and experiences specifically to help another person uncover their own way to accomplish something. Mentors share their experience with a given topic or technique, answer questions and find ways to guide someone who is less experienced. The practitioner should understand the value of mentoring and how it is different from other approaches for supporting people in their development.

The mentor shares their expertise, skills and knowledge with the person being mentored, through developmental conversations, sharing of experiences and by being a role model.

3/ Managing Products with Agility

- Forecasting & Release Planning
- Product Vision
- Product Value
- Product Backlog Management
- Business Strategy
- Stakeholders & Customers

4/ Developing & Delivering Products Professionally

- Emergent Software Development
- Managing Technical Risk
- Continuous Quality
- Continuous Integration
- Continuous Delivery
- Optimizing Flow

5/ Evolving the Agile Organization

- Organizational Design & Culture
- Portfolio Planning
- Evidence Based Management™