Agile Methodologies (Scrum)

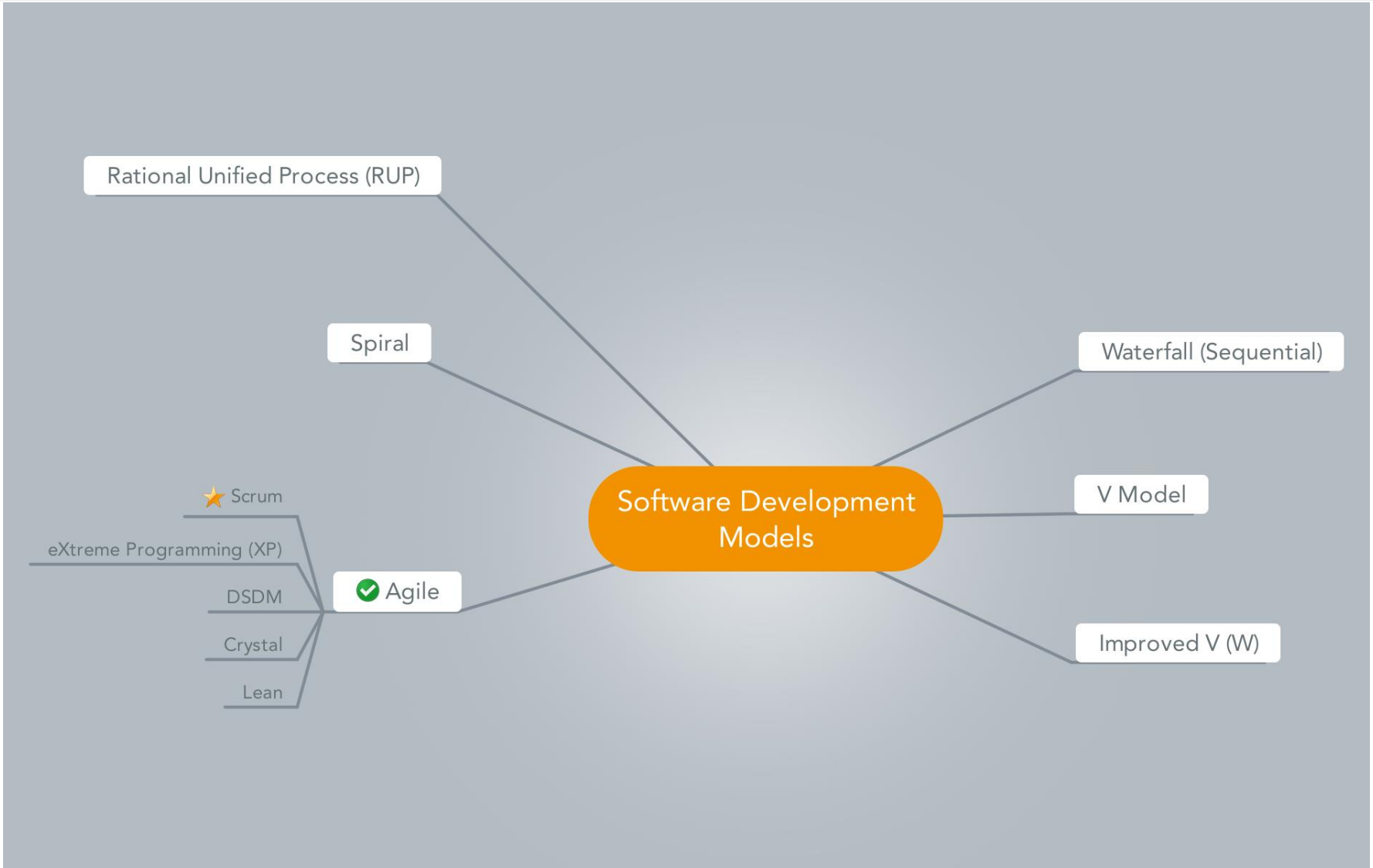
Software Testing

Lector: Aleksandar Karamfilov
Skype: sanders_kar
Email: alex@pragmatic.bg
Facebook: http://facebook.com/a.karamfilov

PRAGMATIC

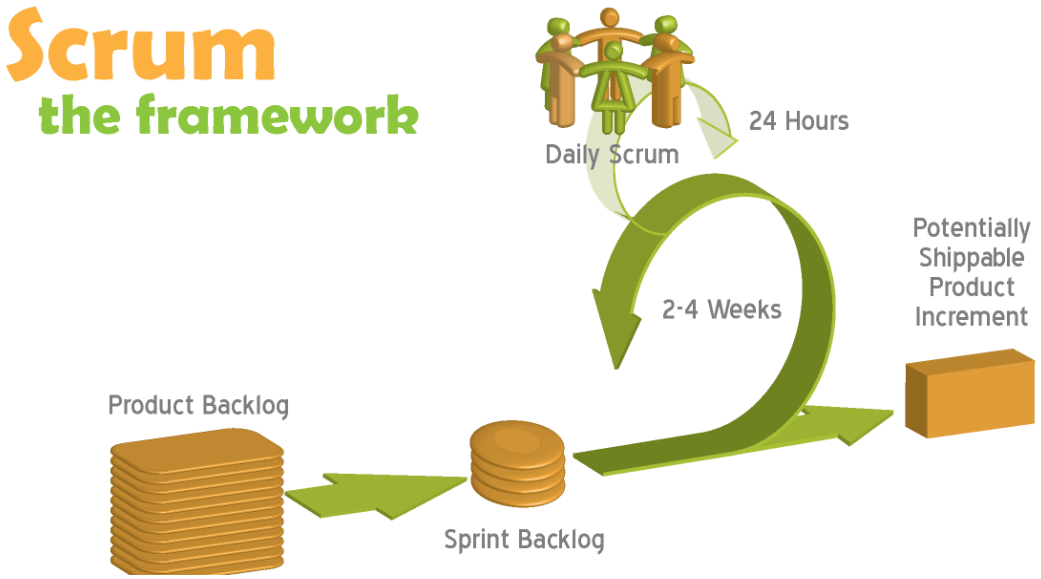
www.pragmatic.bg

Software Development Models



What is Scrum?

Scrum is a framework for developing and sustaining complex products.



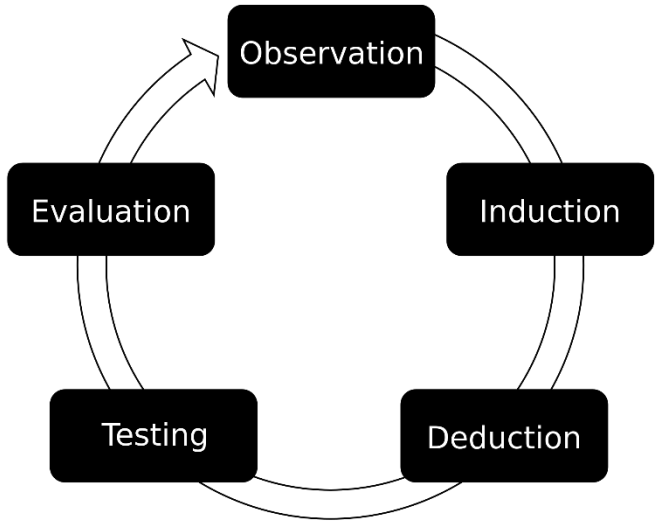
Cynefin Framework

Who invented Scrum?

Scrum framework is developed and sustained by Ken Schwaber and Jeff Sutherland.

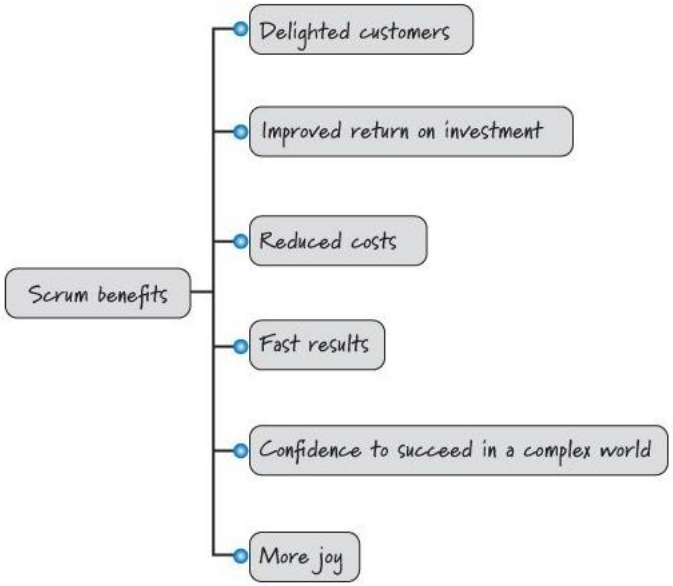
Scrum Theory

Scrum is founded on empirical process control theory, or empiricism. Empiricism asserts that knowledge comes from experience and making decisions based on what is known. Scrum employs an iterative, incremental approach to optimize predictability and control risk.



Scrum Benefits

- Delighted customers
- Improved return on investment
- Reduced costs
- Confidence to succeed in a complex world
- Fast results
- More joy



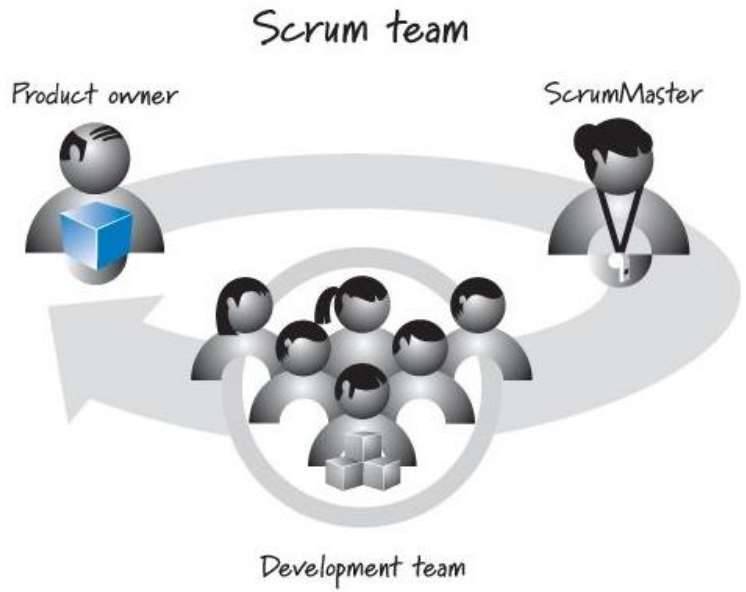
Scrum base pillars

- Transparency
- Inspection
- 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

The Scrum Team

- Product owner
- Scrum master
- Development team



Product Owner

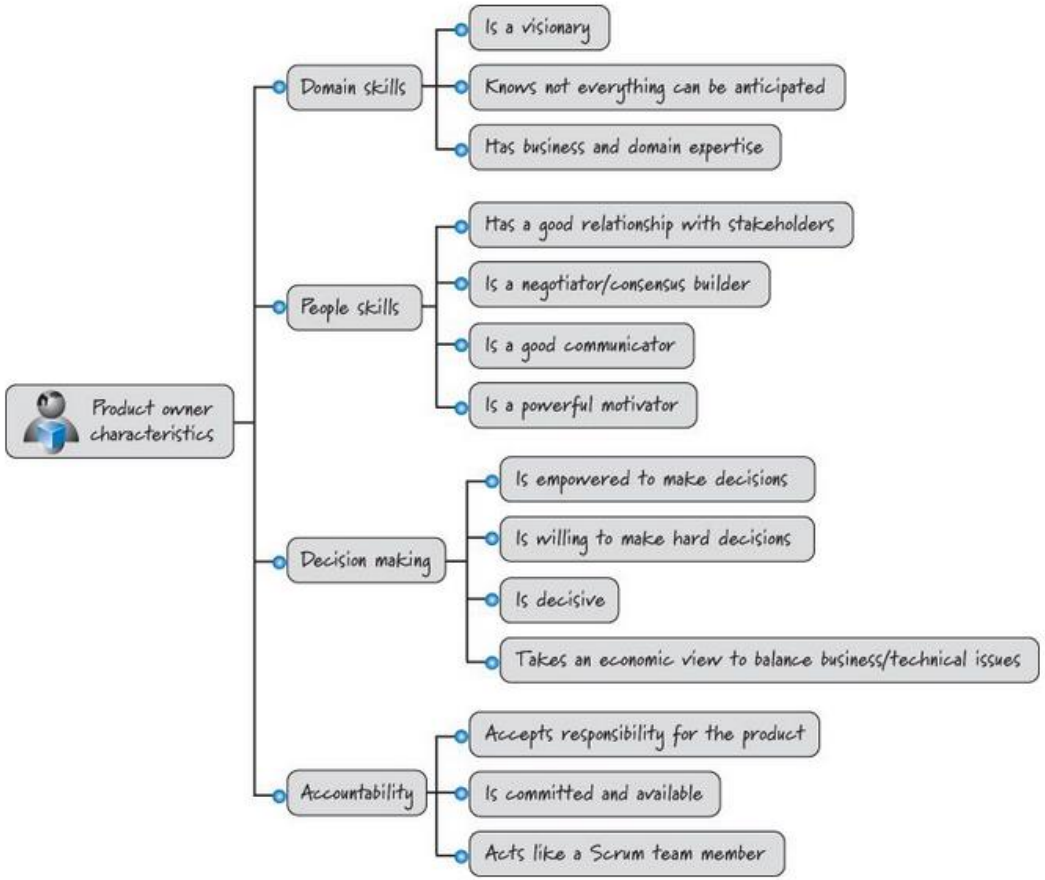
The product owner is the empowered central point of product leadership. He is the single authority responsible for deciding which features and functionality to build and the order in which to build them. The product owner maintains and communicates to all other participants a clear vision of what the Scrum team is trying to achieve.

Product Owner Responsibilities

- Manage economics
- Participate in planning meeting
- Groom the product backlog
- Define acceptance criteria and verify that they are met
- Collaborate with the development team
- Collaborate with stakeholders

Product Owner Characteristics

- Domain Skills
- People Skills
- Decision Making
- Accountability

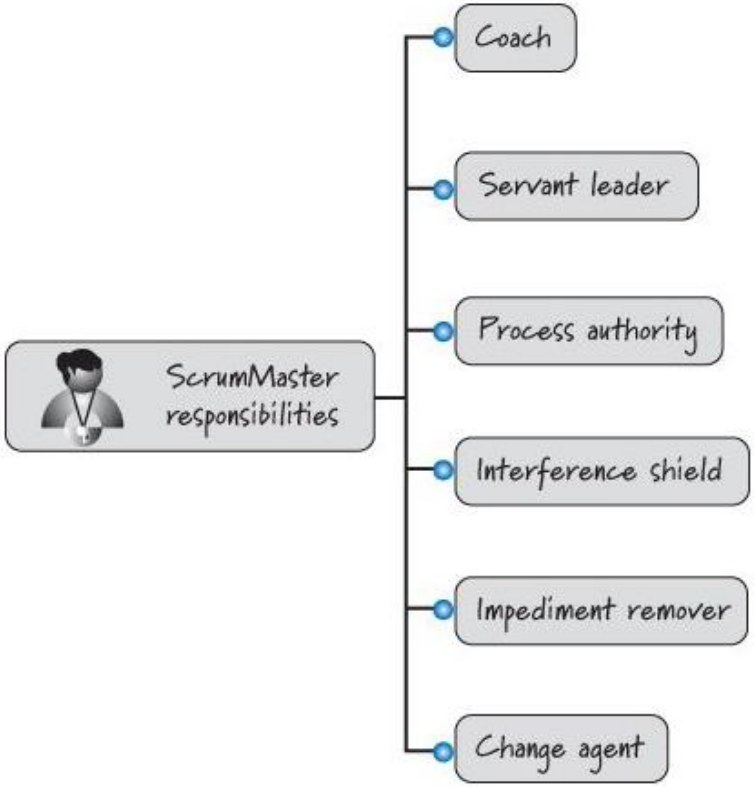


Scrum Master

The Scrum Master helps everyone involved understand and embrace the Scrum values, and practices. She acts as a coach, providing process leadership and helping the Scrum team and the rest of the organization develop their own high-performance, organization-specific Scrum approach. At the same time, the Scrum master helps the organization through the challenging change management process that can occur during a Scrum adoption.

Scrum Master Responsibilities

- Coach
- Servant leader
- Process authority
- Interference shield
- Impediment remover
- Change agent



Scrum Master Characteristics

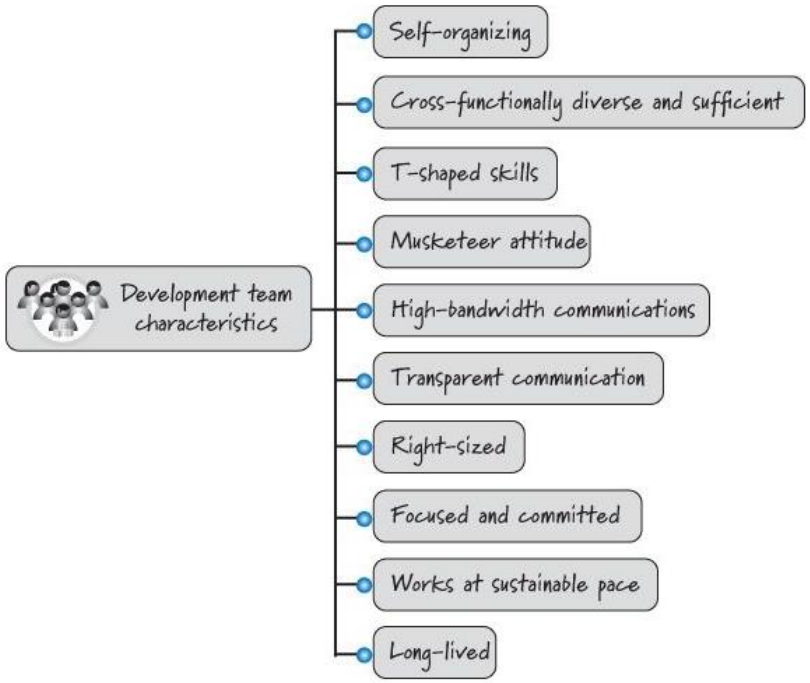
- Knowledgeable
- Questioning
- Patient
- Collaborative
- Protective
- Transparent

Development Team

Traditional software development approaches discuss various job types, such as architect, programmer, tester, database administrator, UI designer, and so on. Scrum defines the role of development team, which is simple a diverse, cross-functional collection of these types of people who are responsible for designing, building, and testing the described product.

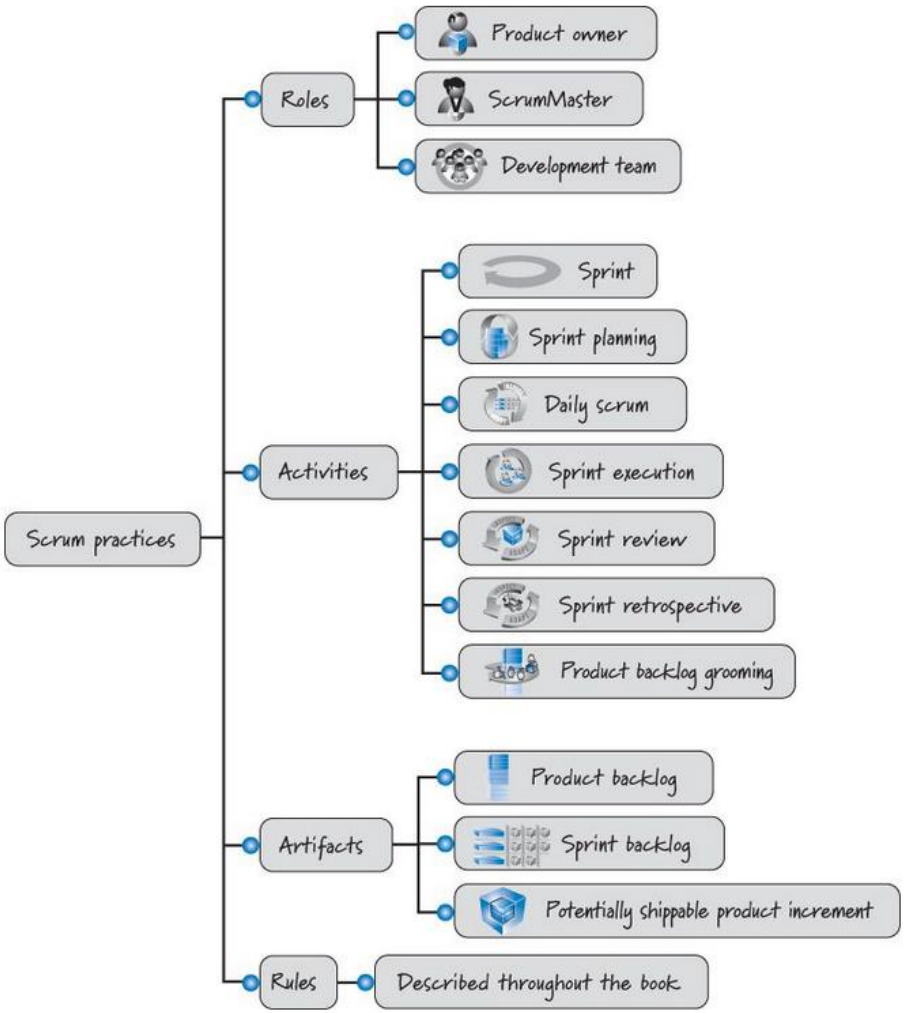
Development Team Characteristics

- Self-organizing
- Cross-functionally diverse and sufficient
- T-shaped skills
- Musketeer attitude
- High-bandwidth communication
- Transparent communication
- Right-sized
- Focused and committed
- Works at sustainable pace
- Long-lived



Scrum Framework

- Roles
- Activities
- Artifacts
- Rules

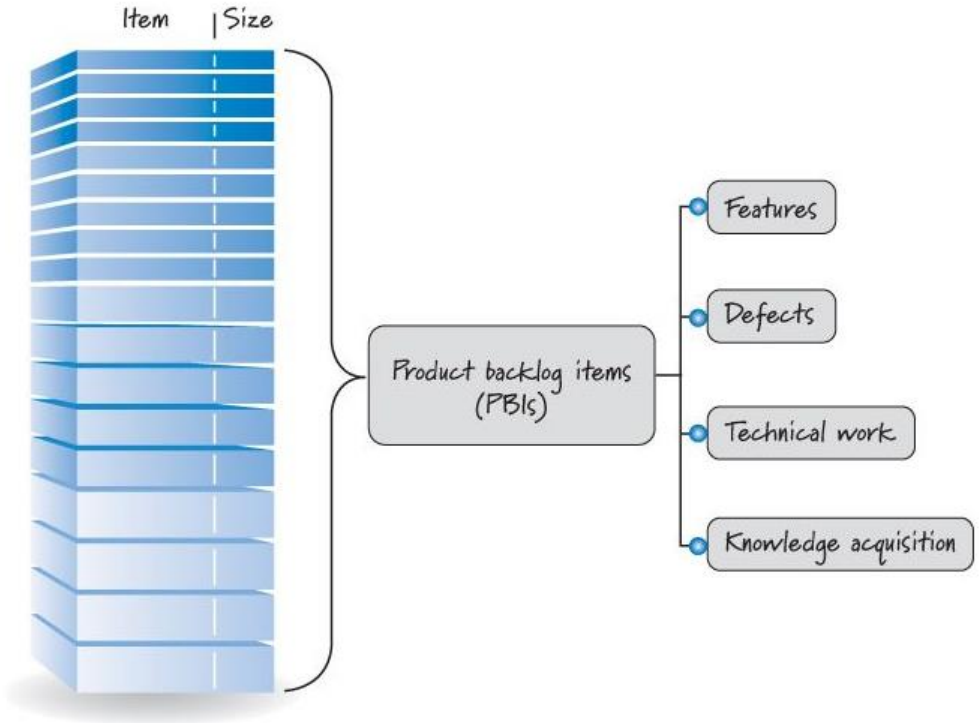


Product Backlog

The agile product backlog in Scrum is a prioritized features list, containing short descriptions of all functionality desired in the product. When applying Scrum, it's not necessary to start a project with a lengthy, upfront effort to document all requirements. Typically, a Scrum team and its product owner begin by writing down everything they can think of for agile backlog prioritization. This agile product backlog is almost always more than enough for a first sprint. The Scrum product backlog is then allowed to grow and change as more is learned about the product and its customers.

Product Backlog

- Features
- Bugs
- Technical Work
- Knowledge acquisition



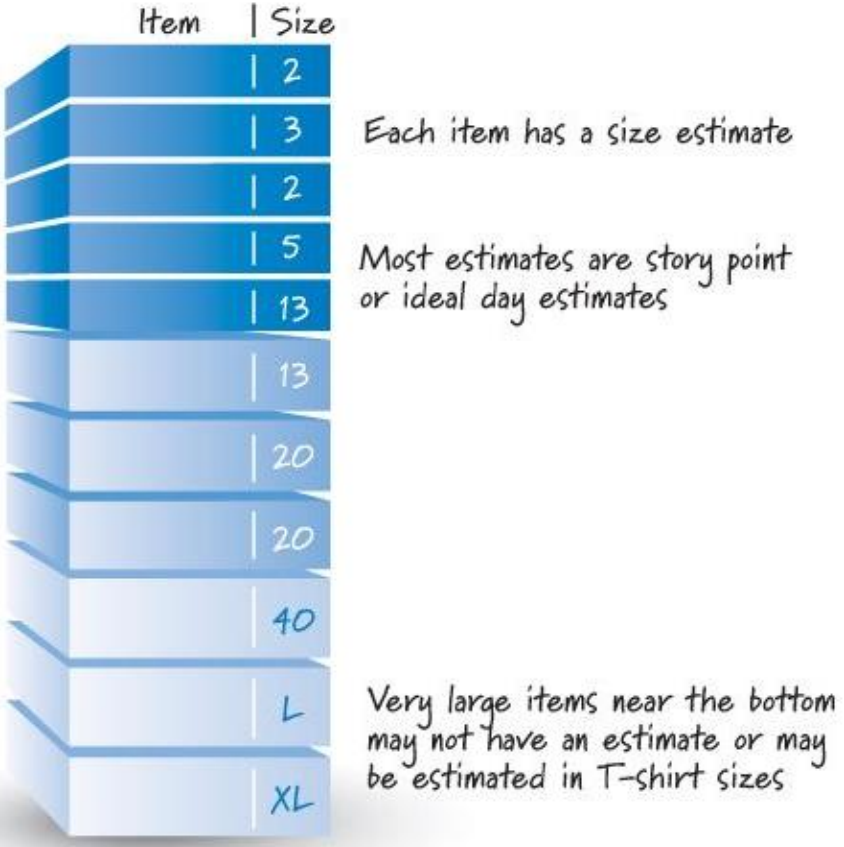
Product Backlog Samples

- Features
- Change
- Technical Improvement
- Defect
- Knowledge acquisition

PBI Type	Example
Feature	As a customer service representative I want to create a ticket for a customer support issue so that I can record and manage a customer's request for support.
Change	As a customer service representative I want the default ordering of search results to be by last name instead of ticket number so that it's easier to find a support ticket.
Defect	Fix defect #256 in the defect-tracking system so that special characters in search terms won't make customer searches crash.
Technical improvement	Move to the latest version of the Oracle DBMS.
Knowledge acquisition	Create a prototype or proof of concept of two architectures and run three tests to determine which would be a better approach for our product.

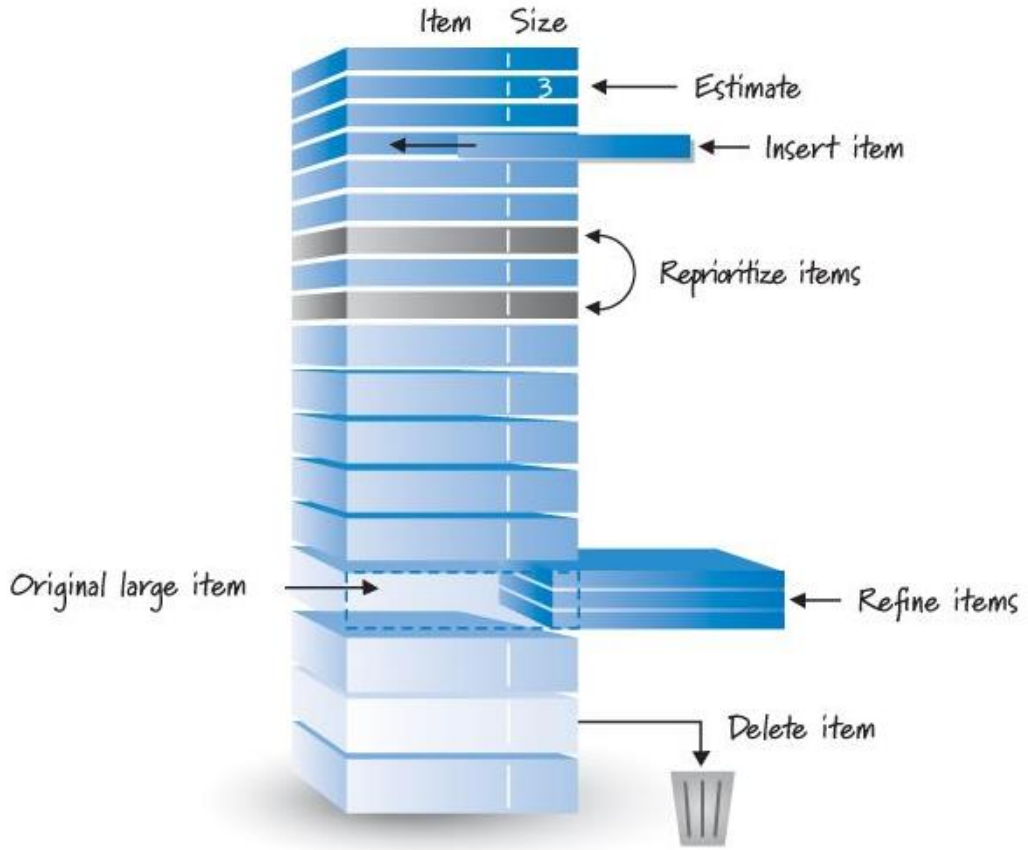
PBI Items

- Features
- Bugs
- Technical Work
- Knowledge acquisition



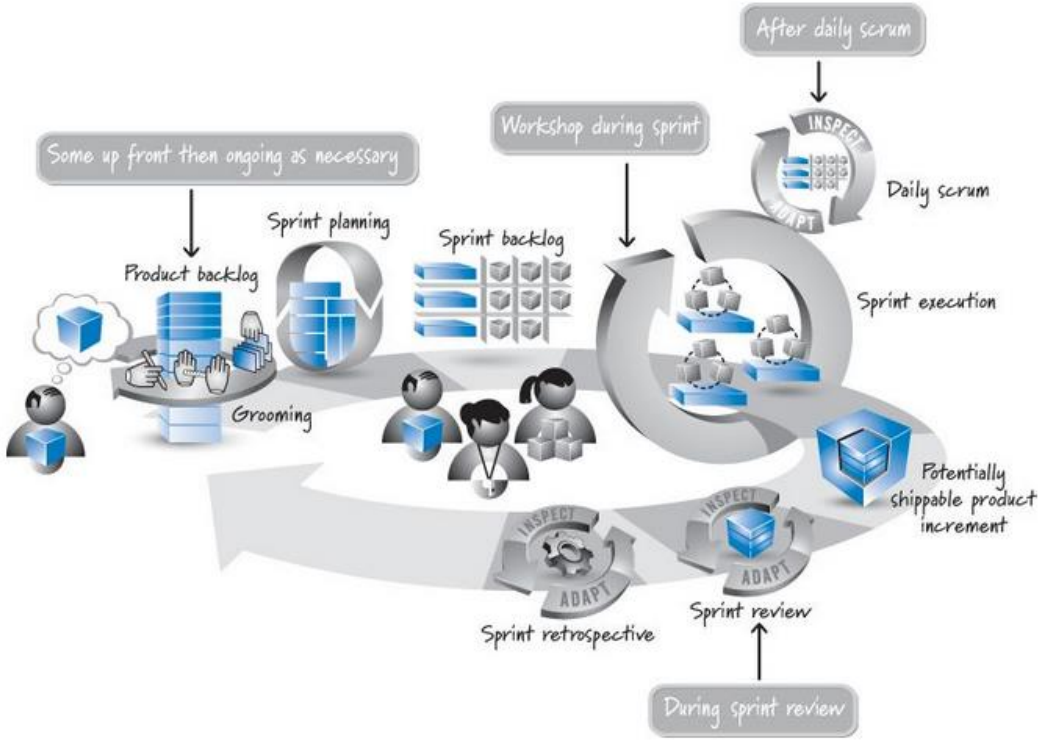
Grooming

Grooming refers to a set of three principal activities: creating and refining (adding details to) PBIs, estimating PBIs, and prioritizing PBIs.



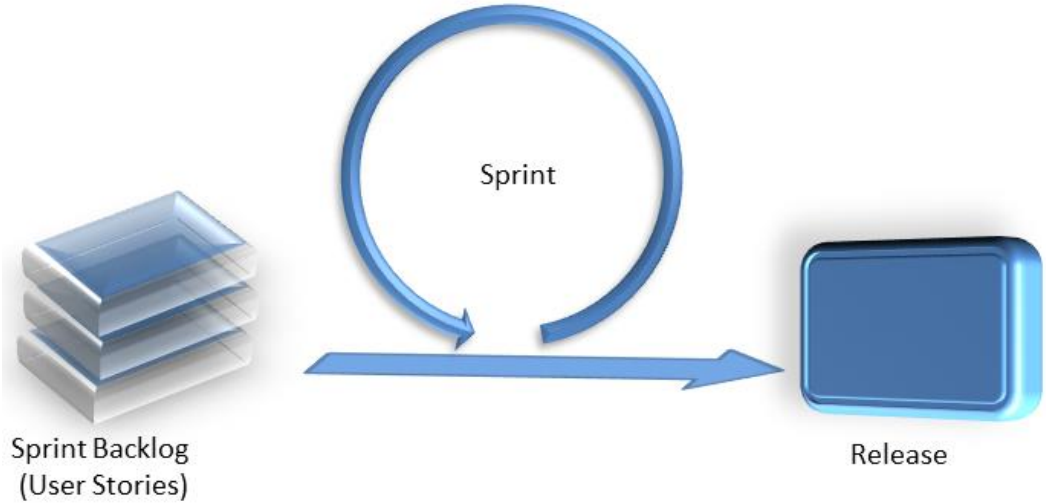
When Does Grooming Take Place?

The Scrum framework only indicates that grooming needs to happen; it doesn't specify when it should happen. So when does grooming actually take place?



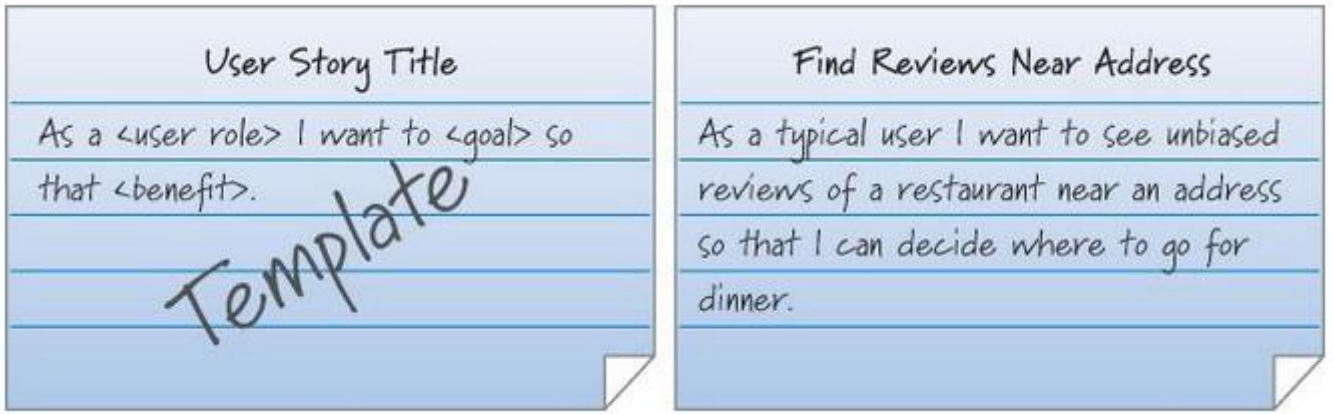
What is Sprint?

In product development, a sprint is a set period of time(1- 4 weeks) during which specific work has to be completed and made ready for review.
Each sprint begins with a planning meeting. During the meeting, the product owner (the person requesting the work) and the development team agree upon exactly what work will be accomplished during the sprint. The development team has the final say when it comes to determining how much work can realistically be accomplished during the sprint, and the product owner has the final say on what criteria needs to be met for the work to be approved and accepted.

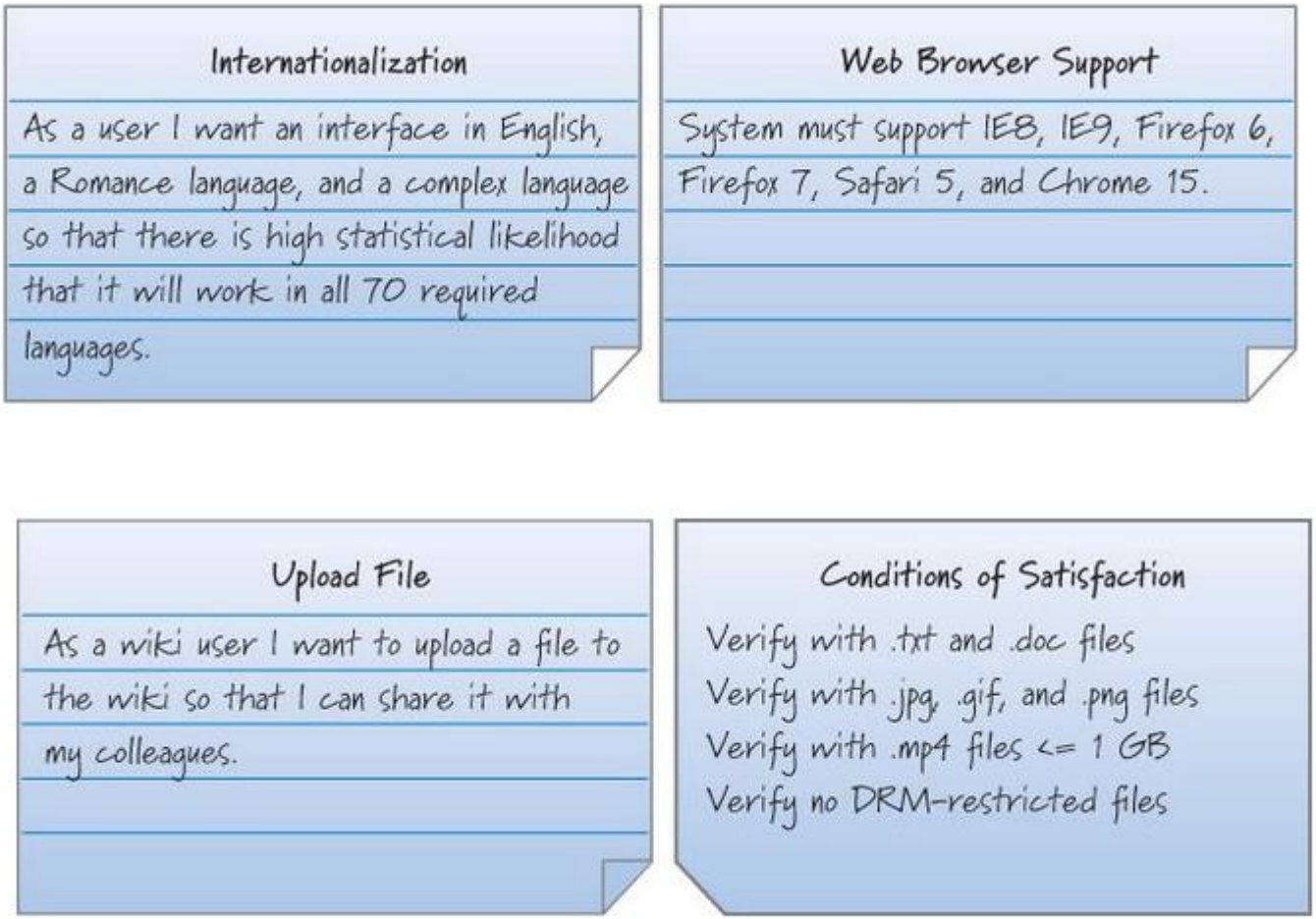


User Stories

User stories are a convenient format for expressing the desired business value for many types of product backlog items, especially features. User stories are crafted in a way that makes them understandable to both business people and technical people. They are structurally simple and provide a great placeholder for conversation. Additionally, they can be written at various levels of granularity and are easy to progressively refine.

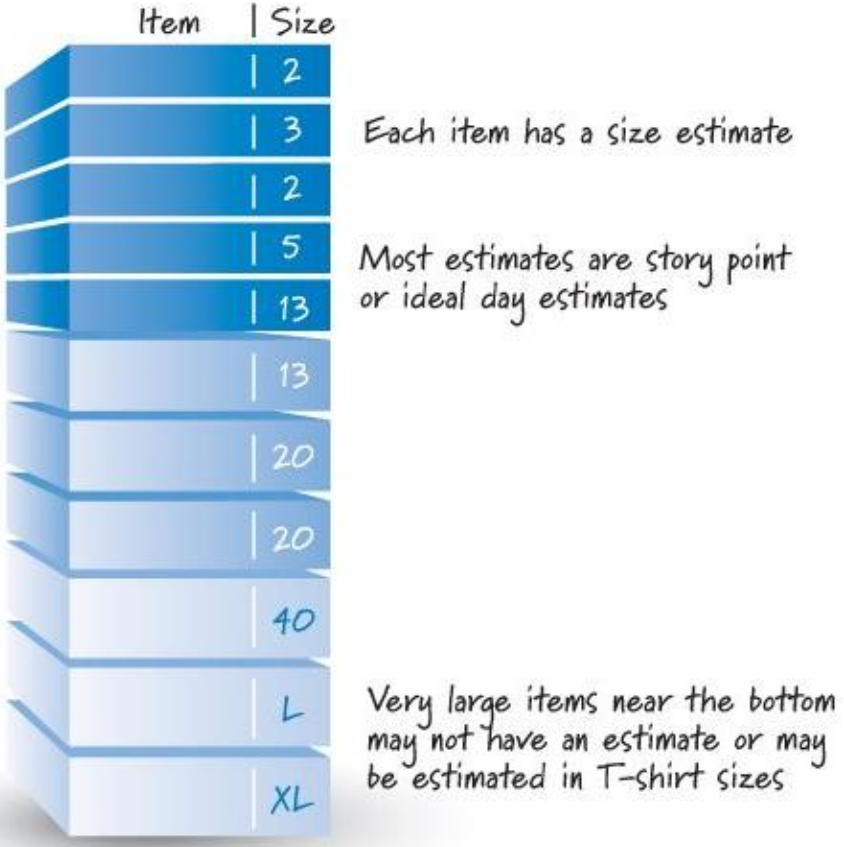


User Story Types



Scrum Events

- Sprint Planning Meeting
- Daily Scrum
- Sprint Review
- Sprint Retrospective



Scrum Planning Meeting

The work to be performed in the Sprint is planned at the Sprint Planning Meeting. This plan is created by the collaborative work of the entire Scrum Team. The Sprint Planning Meeting is time-boxed to eight hours for a one-month Sprint.

Daily Scrum Meeting

The Daily Scrum is a 15-minute time-boxed event for the Development Team to synchronize activities and create a plan for the next 24 hours. This is done by inspecting the work since the last Daily Scrum and forecasting the work that could be done before the next one.

A common approach to performing the daily scrum has the Scrum Master facilitating and each team member taking turns answering three questions for the benefit of the other team members:

- What did I accomplish since the last daily scrum?
- What do I plan to work on by the next daily scrum?
- What are the obstacles or impediments that are preventing my from making progress?

Sprint Review Meeting

A Sprint Review is held at the end of the Sprint to inspect the Increment and adapt the Product Backlog if needed. During the Sprint Review, the Scrum Team and stakeholders collaborate about what was done in the Sprint.
The conversation is focused on reviewing the just-completed features in the context of the overall development effort. Everyone in attendance gets clear visibility into what is occurring and has an opportunity to help guide the forthcoming development to ensure that the most business-appropriate solution is created.

Sprint Retrospective Meeting

The Sprint Retrospective occurs after the Sprint Review and prior to the next Sprint Planning Meeting. This is a three-hour time-boxed meeting for one-month Sprints. Proportionately less time is allocated for shorter Sprints.

Purpose of Sprint Retrospective Meeting

- Inspect how the last sprint went with regards to people, process, relationships and tools.
- Identify and order the major items that went well and potential improvements.
- Creating plan for implementing improvements for the sprints onwards.

Definition of Done

Conceptually the definition of done is a checklist of the types of work that the team is expected to successfully complete before it can declare its work to be potentially shippable.
Obviously the specific items on the checklist will depend on the number of variables:

- The nature of the product being built
- The technologies being used to build it
- The organization that is building it
- The current impediments that affect what is possible

Definition of Done

Most of the time, a bare-minimum definition of done should yield a complete slice of product functionality, one that has been designed, built, integrated, tested and documented and would deliver validated customer value.

	Definition of Done
❑	Design reviewed
❑	Code completed
❑	Code refactored
❑	Code in standard format
❑	Code is commented
❑	Code checked in
❑	Code inspected
❑	End-user documentation updated
❑	Tested
❑	Unit tested
❑	Integration tested
❑	Regression tested
❑	Platform tested
❑	Language tested
❑	Zero known defects
❑	Acceptance tested
❑	Live on production servers

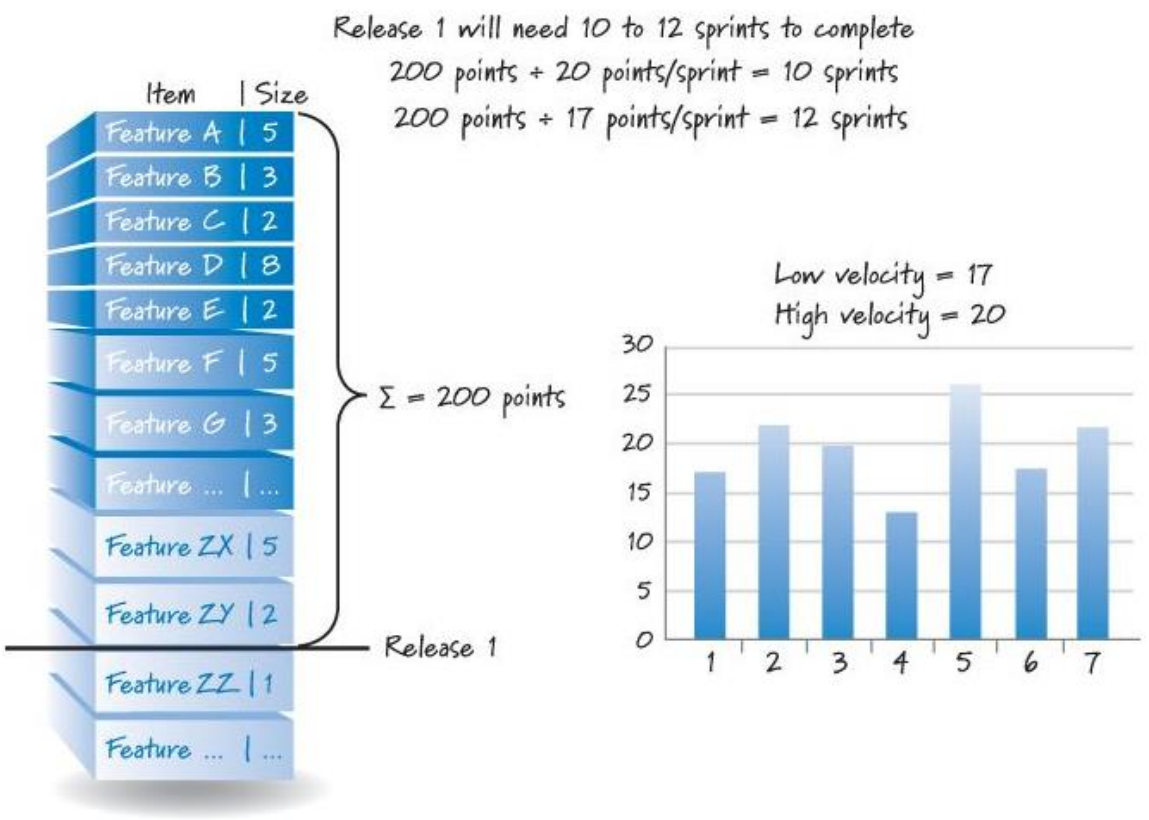
What is Velocity?

Velocity is the amount of work completed each sprint. It is measured by adding the sizes of the PBIs that are completed by the end of the sprint. A PBI is either done or its not done. The product owner does not get any value from undone items, so velocity does not include the size numbers of partially completed PBIs.

Velocity measures output(the size of what was delivered), not outcome (the value of what was delivered.

Average Velocity

Using some simple math(like high and low averages, 90% confidence intervals, and so on), we can easily get two velocity numbers from our team's historical velocity data.

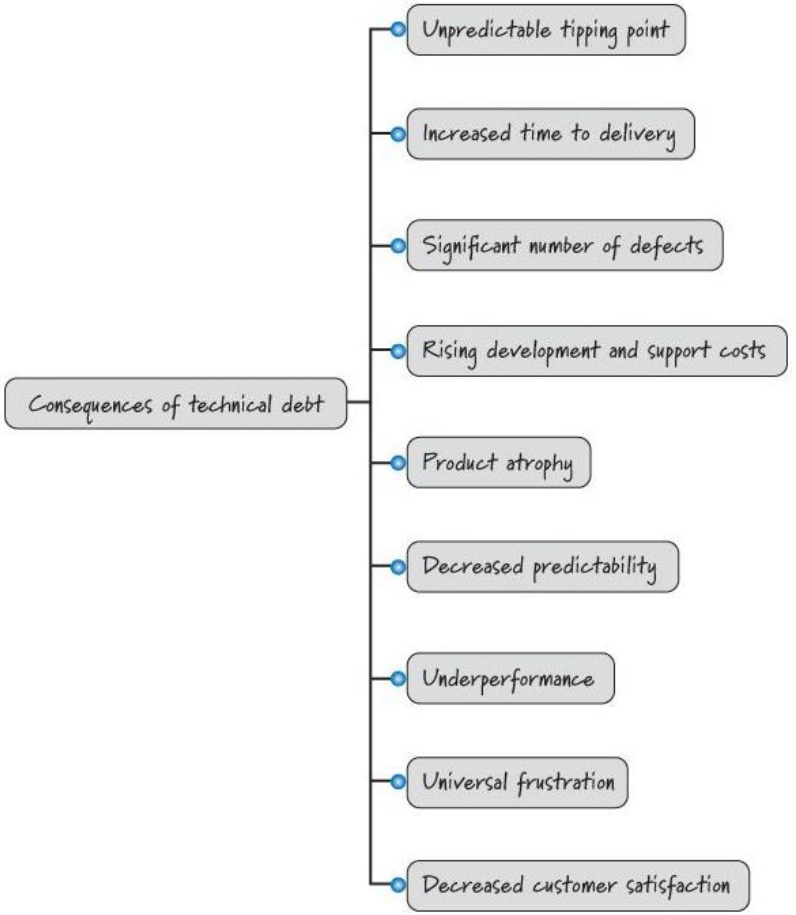


Technical Debt

Technical Debt is a wonderful metaphor developed by Ward Cunningham to help us think about this problem. In this metaphor, doing things the quick and dirty way sets us up with a technical debt, which is similar to a financial debt. Like a financial debt, the technical debt incurs interest payments, which come in the form of the extra effort that we have to do in future development because of the quick and dirty design choice. We can choose to continue paying the interest, or we can pay down the principal by refactoring the quick and dirty design into the better design. Although it costs to pay down the principal, we gain by reduced interest payments in the future.

Consequences of Technical Debt

As the level of technical debt rises, so does the severity of the consequences.



Questions