System Analysis and Design

Scrum

Department of Computer Engineering
Sharif University of Technology
Maryam Ramezani
maryam.ramezani@sharif.edu

01 Introduction

Principles behind the Agile Manifesto

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

Extreme Programming

- A software methodology
- A type of agile programming
- Has short development cycles (called boxes) & releases
- Advocates: pair programming, ongoing unit testing, flat management structure, frequent communication among team & with customer

Possible drawbacks of extreme programming [Beck 1996]

- Lack of overall specification
- (In)ability to scale up to large projects

The game of rugby --- similar to American football

- **Scrum**. Packs of opposing players push against each other for possession of the ball
- Once a team gets the ball, it attempts to “sprint” to the goal line.

The rugby scrum

The rugby “sprint”

wesleying.blogspot.com/2009/03/mens-club-rugb..

Scrum Development [1986]

- **Scrum**: incremental framework for managing complex work (such as new product development) commonly used with agile software development.

Sprint

- A fixed period of X days to develop a deliverable product
- The Sprint includes design, coding, testing, and documentation
- Once a Sprint has started only the Scrum Team can add or remove items in Sprint backlog
- Abnormal termination of Sprint is called for when the Sprint Goal no longer makes sense

02

Scrum Roles

Scrum Roles: Chickens and Pigs

A chicken and a pig are walking down the road. The chicken says to the pig: “Do you want to open a restaurant with me?”

The pig considers the question and replies, “Yes, I’d like that. What do you want to call the restaurant?”

The chicken replies, “Ham and Eggs”.

The pig stops, pauses and replies, “On a second thought, I don’t think I want to open a restaurant with you. I’d be committed, but you’d only be involved.”

Three major roles in scrum

1) **Product Owner:** who represents the stakeholders

1) **ScrumMaster:** who maintains the processes (typically in lieu of a project manager)

2) **Team:** a cross-functional group of about 7 people who do the actual analysis, design, implementation, testing, etc.

Scrum Roles: Product Owner

- **Represents** the interests of all stakeholders in the project and resulting system

- **Tasks/Responsibilities:**
  - is the only person in charge of the Product Backlog
    - maintains and sustains content
    - prioritizes Product Backlog items
    - estimates Product Backlog effort
  - is responsible for initial and ongoing funding

Scrum Roles: Scrum Master

- **Represents** management and team to each other

- **Tasks/Responsibilities:**
  - ensures that Scrum values, practices and rules are enacted and enforced
  - plans and initiates Sprints together with the team
  - conducts Daily Scrums and ensures that impediments are promptly removed
  - controls progress and takes appropriate measures

Scrum Roles: Scrum Team

- **Tasks/Responsibilities:**
  - formulates a Sprint Goal at the Sprint Planning Meeting
  - commits to turn a set of Product Backlog into a working product
  - self-organizes (assigns, works on, modifies and (re-)estimates tasks)

  → responsible for doing all analysis, design, coding, testing and user documentation

  → has full authority to do whatever is necessary to meet the Sprint Goal

- **Team size:** seven people, plus or minus two

- **Team composition:** cross-functional → analysts, designers, coding engineers

03

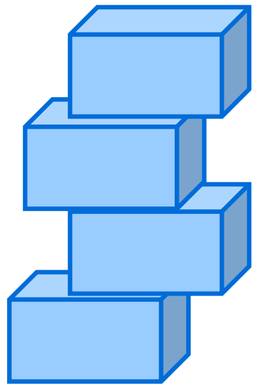
Scrum Artifacts

Scrum Artifacts

- Product backlog
- Burndown chart
- Sprint Backlog

Product backlog

- Ever changing
- Prioritized list
- Owned by product owner
- Spread sheet example
- Prioritized list of work to be performed on a product
- Anyone can contribute backlog items
- Product Owner responsible for prioritisation



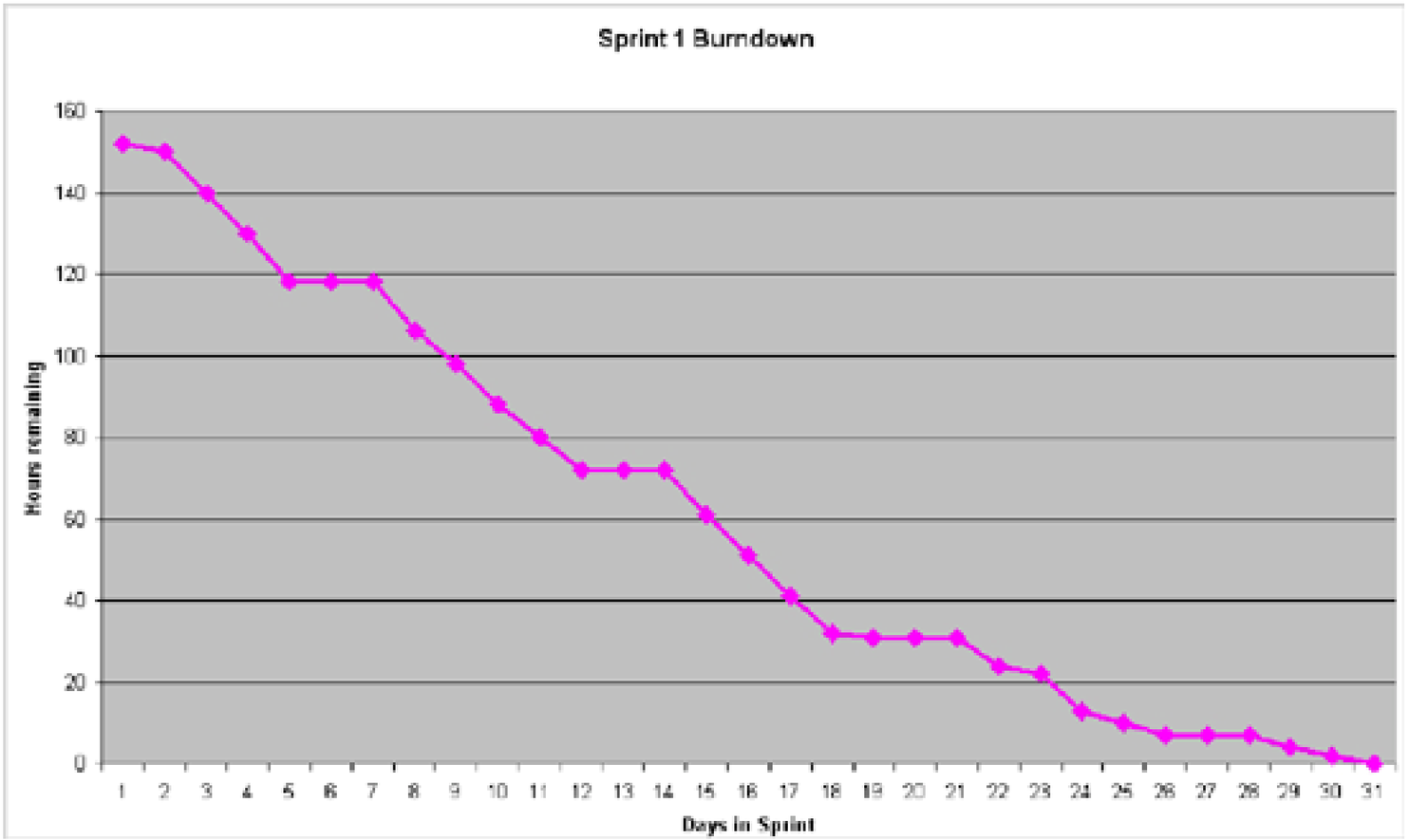
Sprint Backlog

- The sprint backlog defines the work, or tasks, that a team defines for turning the Product backlog it selects for that Spring into an increment of potentially shippable product functionality
- Task should be 4-16 hours each
- Highly visible, real-time picture of the work
- Owned by the team
- Maintained as spread sheet daily by a tracker or responsible individuals.

Burndown chart

- Visualize the correlation between the amount of work remaining and the progress in reducing the work
  - X: date
  - Y: hours of work remaining
- Updated according the Sprint backlog

Burndown Chart Example



04

Scrum Meetings

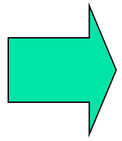
Scrum meetings

- Sprint planning meeting
- Daily standup meeting
- Sprint review meeting
- Sprint Retrospective

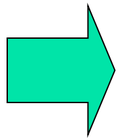
Sprint Planning

Meeting to set the next Sprint goal

- Product Backlog
- Team Capabilities
- Business Conditions
- Technology Stability
- Executable Product Increment

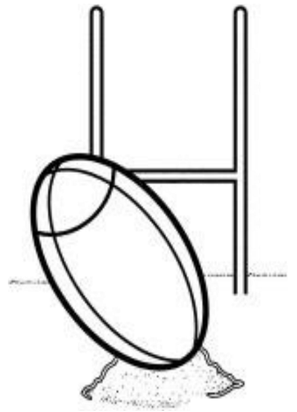


Review
Consider
Organize



Next Sprint Goal

Sprint Backlog



Sprint Planning Meeting

- Plan next X days
- Select what work is to be done
- Prepare the Sprint Backlog that details the time it will take to do that work, with the entire team
- Identify and communicate how much of the work is likely to be done during the current sprint
- Eight hour limit

Sprint Planning Meeting

- Stake-holders to refine and re-prioritize the Product Backlog and Release Backlog and to choose the goals for the next iteration, usually droved by the highest business value and risk.
- Scrum team and Product Owner meet to consider how to achieve the requests, and to create a sprint backlog of tasks to meet the goals.

Daily Standup

- Daily 15 minute status meeting
- Team **stands** in a circle facing each other
- Each team member answers 3 questions:
  - What have you done since yesterday?
  - What are you planning to do by today?
  - Do you have any problems preventing you from accomplishing your goal?

For synchronization not problem solving!

- The Scrum Master asks the questions. Same place and time every day.
- A short meeting may be held after the Scrum meeting to resolve issues (only attended by relevant team members).
- Why standup meeting?
  - Promote individual's commitment to the team
  - Promote close working relationship
  - Identify issues in timely fashion

Sprint Review

- During this meeting the team presents to management, customers, users and the Product Owner the product increment that has been built during the Sprint
- The team tells the story of its journey during the Sprint.
- Powerpoint presentations are forbidden!
- The Scrum Master is responsible for coordinating and conducting this meeting
- No more that 4 hours duration
- Review the work that was completed and not completed
- Present the completed work to the stakeholders (a.k.a. "the demo")
- Incomplete work cannot be demonstrated

Sprint Review Meeting

- Demo time
  - Informal
  - Anybody can attend
- Did the team achieved the sprint goal?

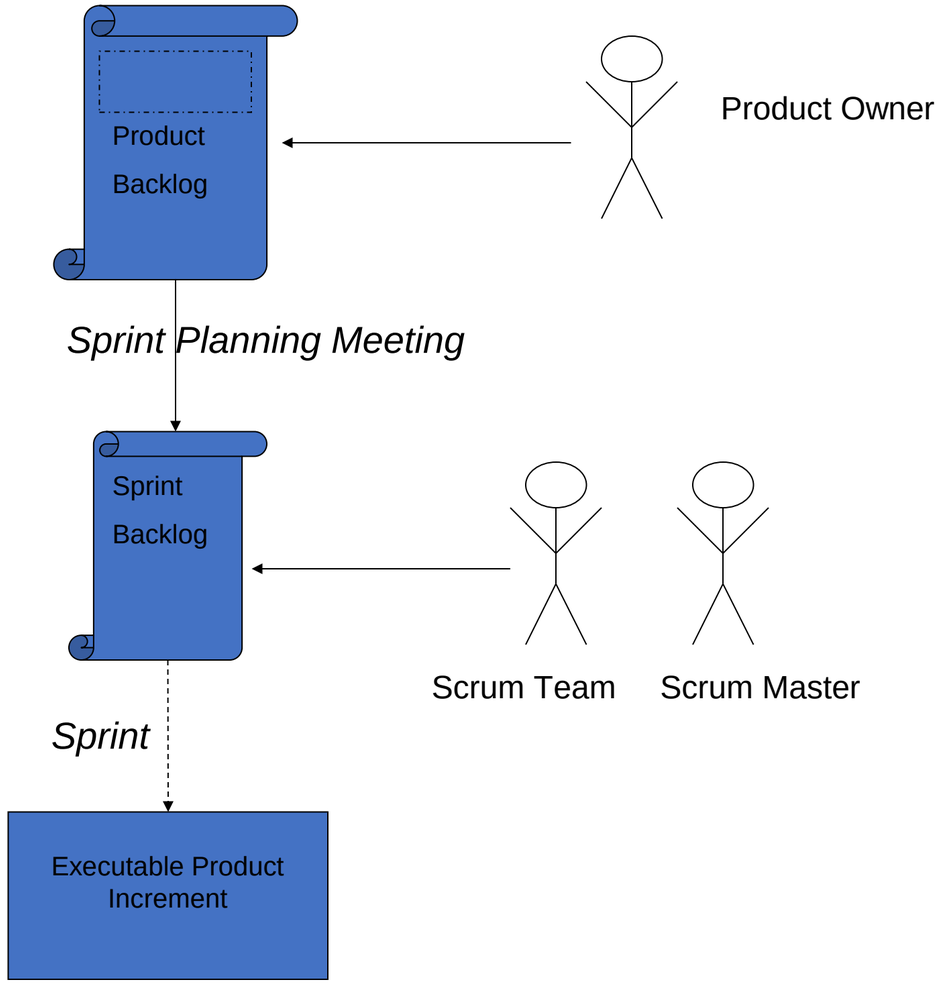
Sprint Retrospective

- All team members reflect on the past sprint.
- Make continuous process improvement.
- Find the ways to improve team's performance
  - Start doing
  - Stop doing
  - Continue doing
- Who can attend?
  - Team, product owner, scrum master
- Three hour time limit

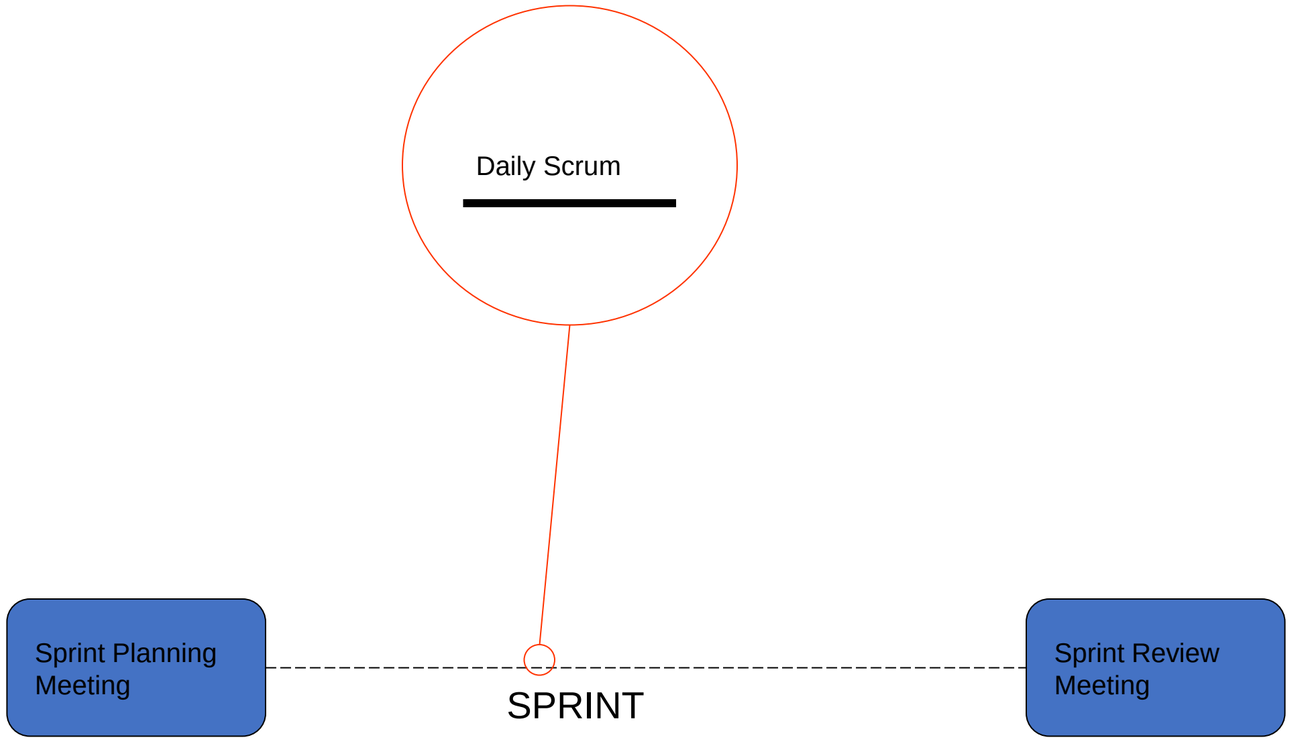
05

Scrum Overview

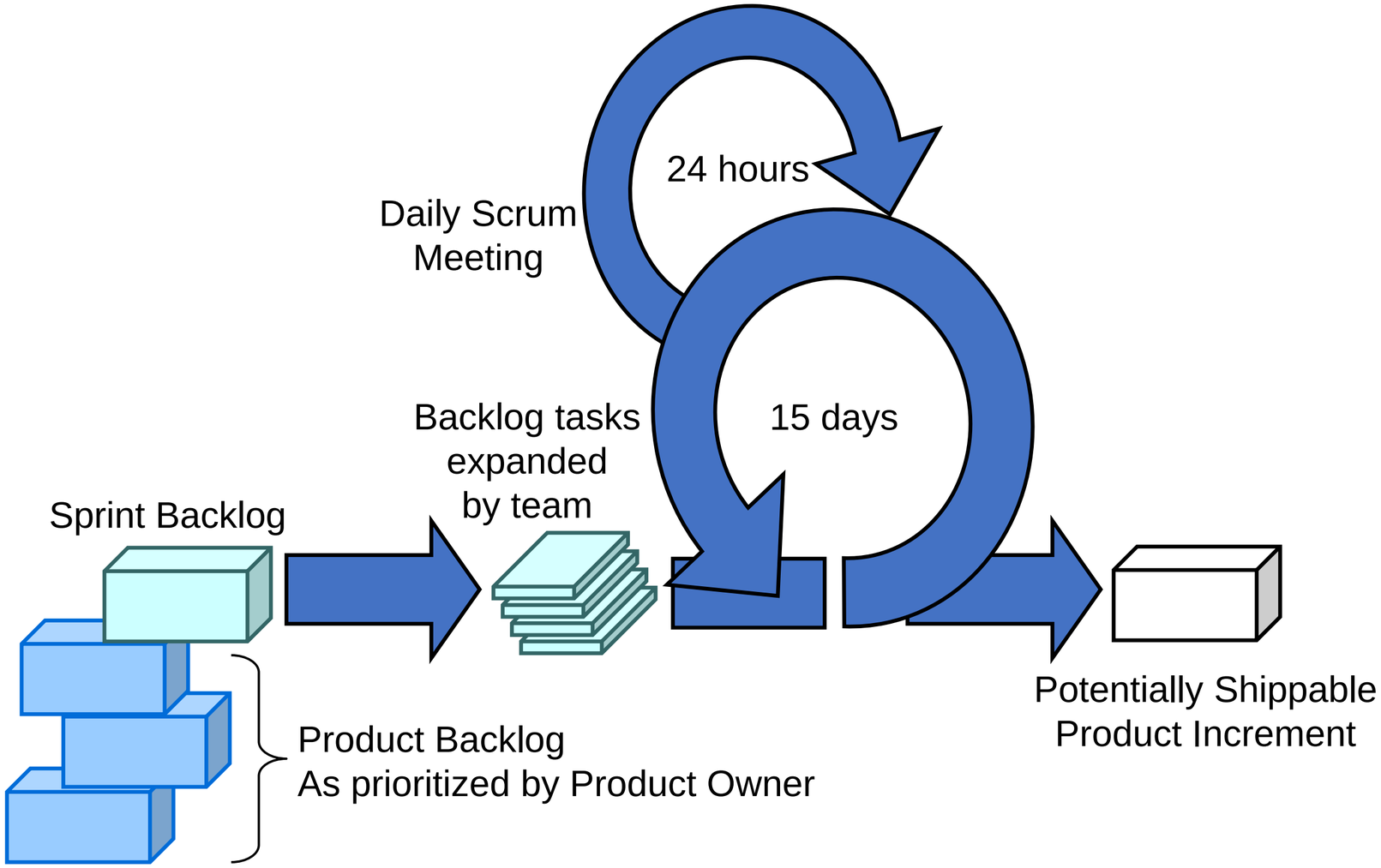
Quick Overview I



Quick Overview II



SCRUM Lifecycle



06

Product Owner

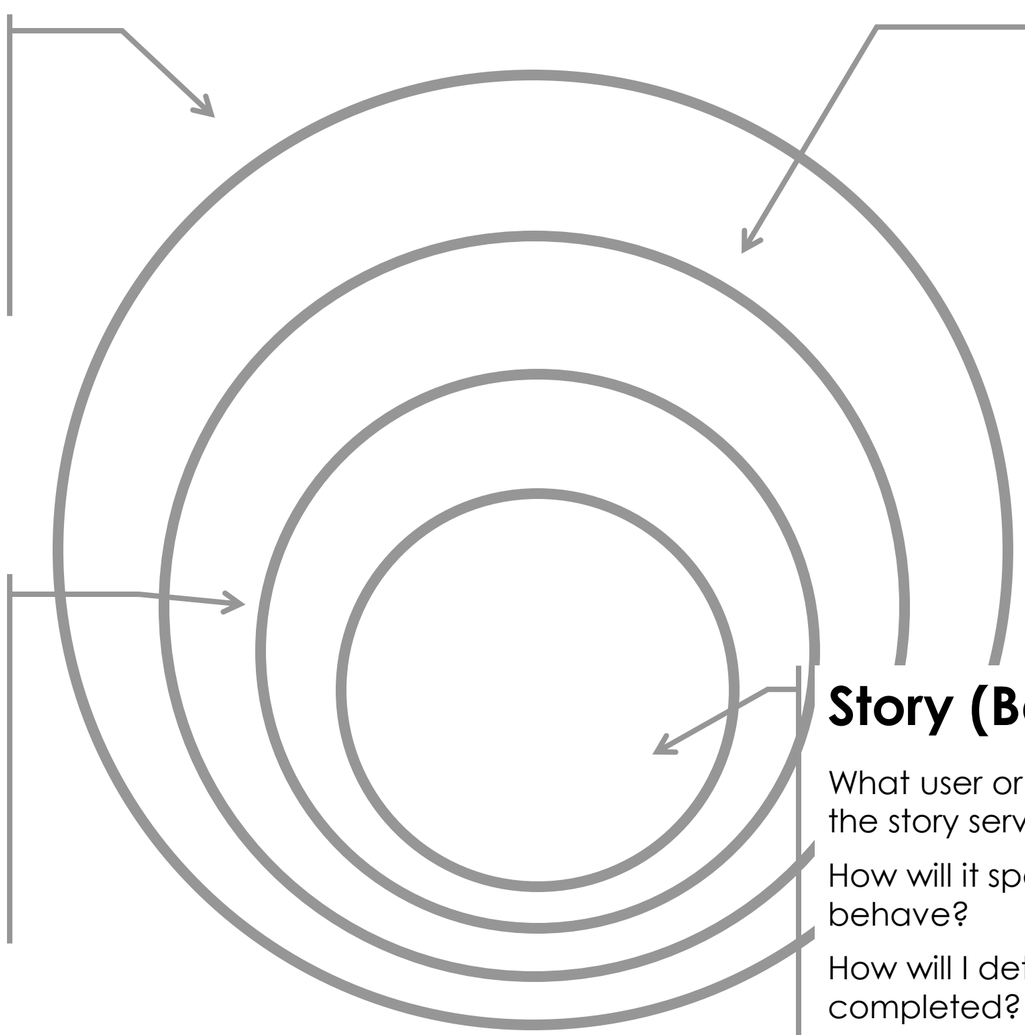
The product owner plans the product in layers

The product owner plans the product in layers

Product
or **Project**

What business objectives will the product fulfill?

Product Charter



Release

How can we release value incrementally?

What subset of business objectives will each release achieve?

What user constituencies will the release serve?

What general capabilities (big stories) will the release offer?

Release plan

Iteration

What specifically will we build? (user stories)

How will this iteration move us toward release objectives?

Iteration Plan

Story (Backlog Item)

What user or stakeholder need will the story serve?

How will it specifically look and behave?

How will I determine if it's completed?

Story Details

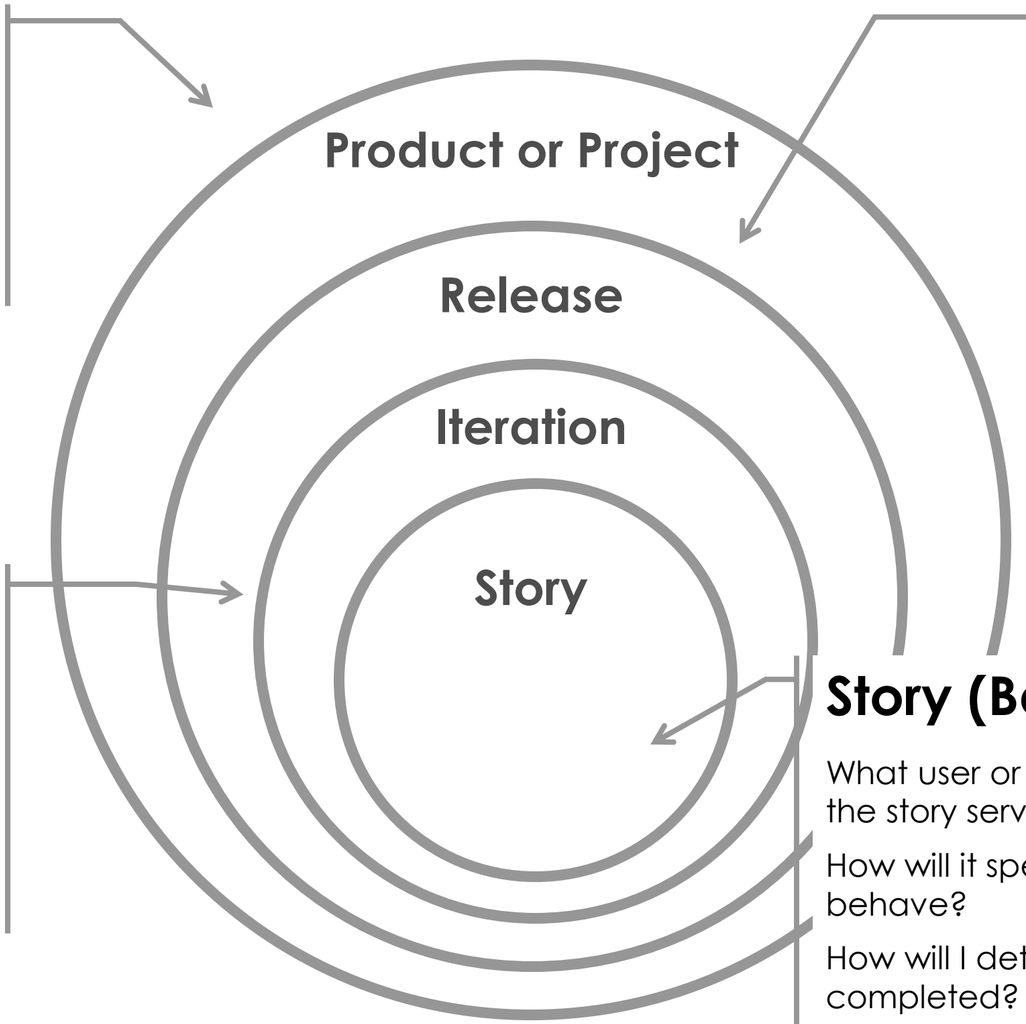
Acceptance Tests

The Planning Onion can grow to include product portfolios and business strategy

Product
or **Project**

What business objectives will the product fulfill?

Product Charter



Release

How can we release value incrementally?

What subset of business objectives will each release achieve?

What user constituencies will the release serve?

What general capabilities (big stories) will the release offer?

Release plan

Iteration

What specifically will we build? (user stories)

How will this iteration move us toward release objectives?

Iteration Plan

Story (Backlog Item)

What user or stakeholder need will the story serve?

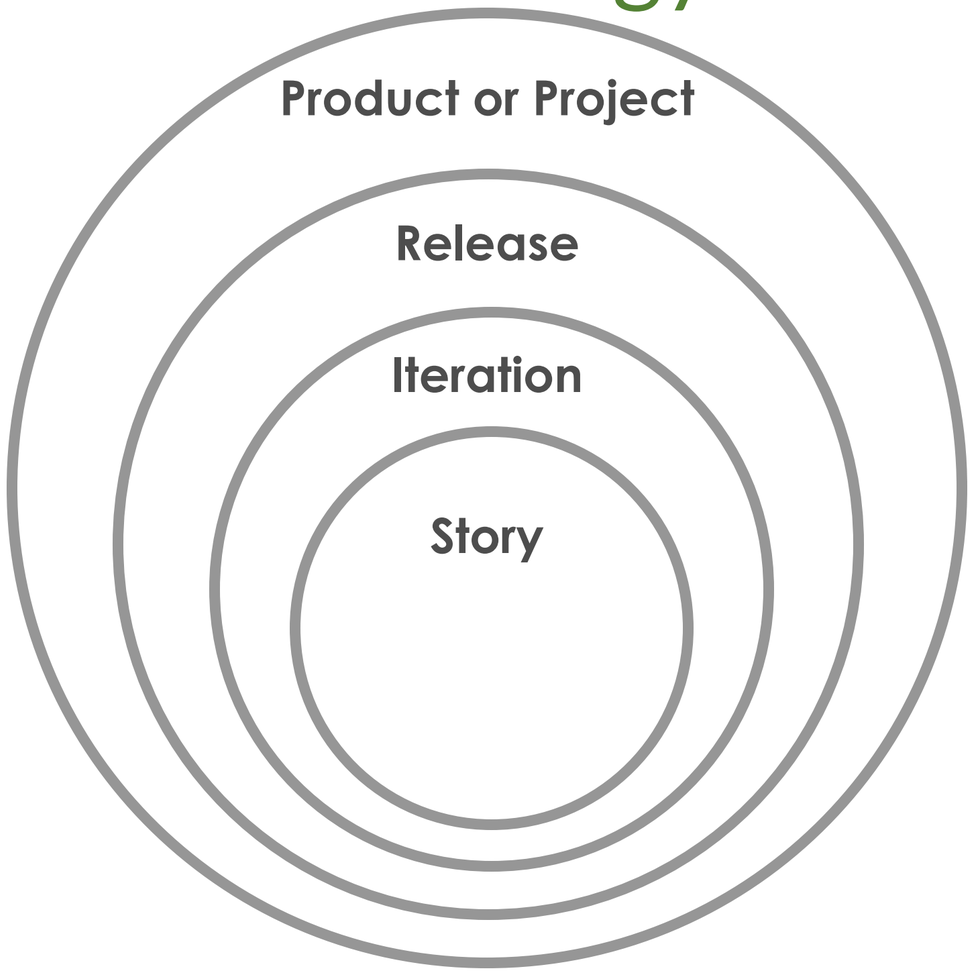
How will it specifically look and behave?

How will I determine if it's completed?

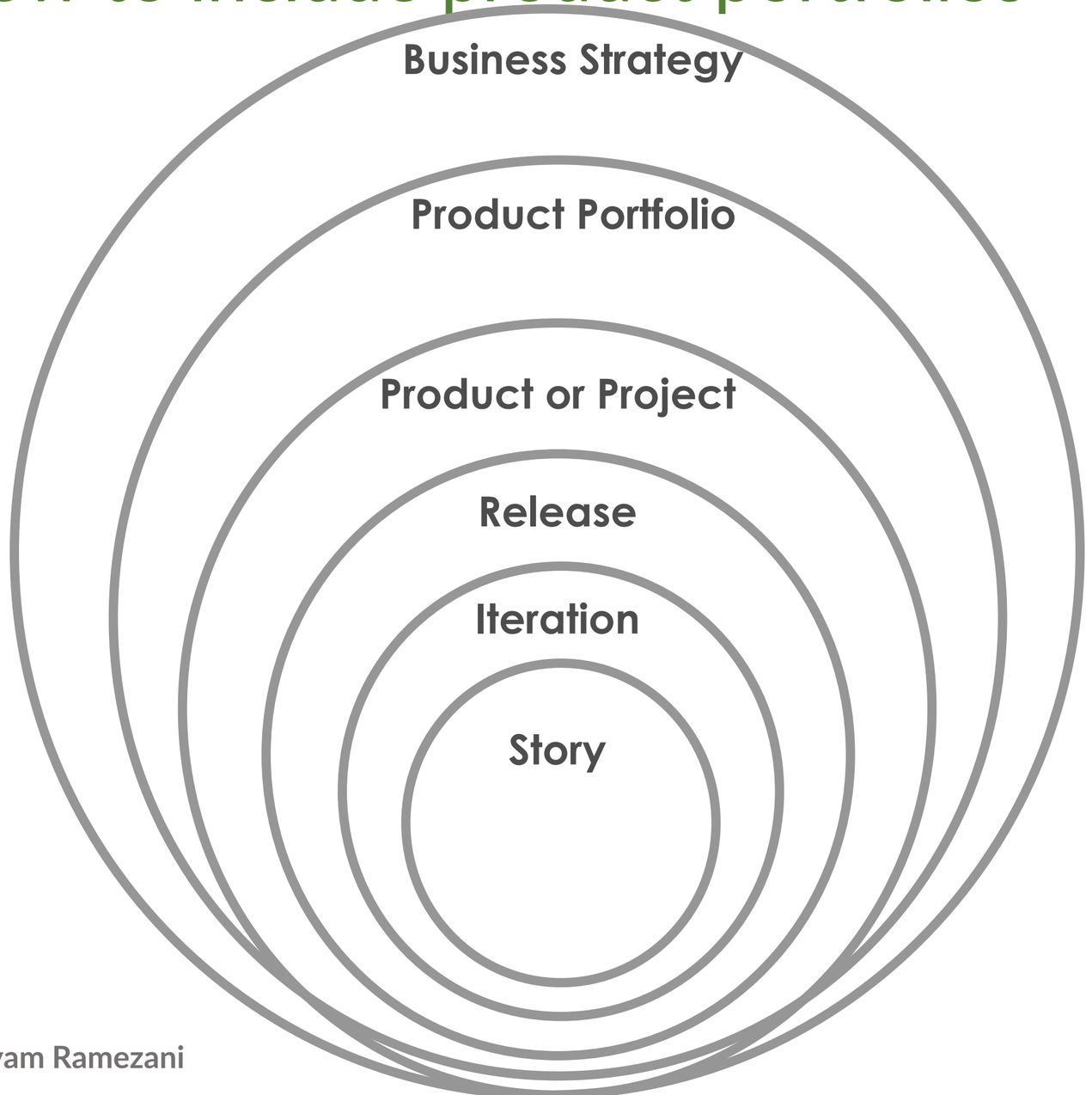
Story Details

Acceptance Tests

The Planning Onion can grow to include product portfolios and business strategy



The Planning Onion can grow to include product portfolios and business strategy



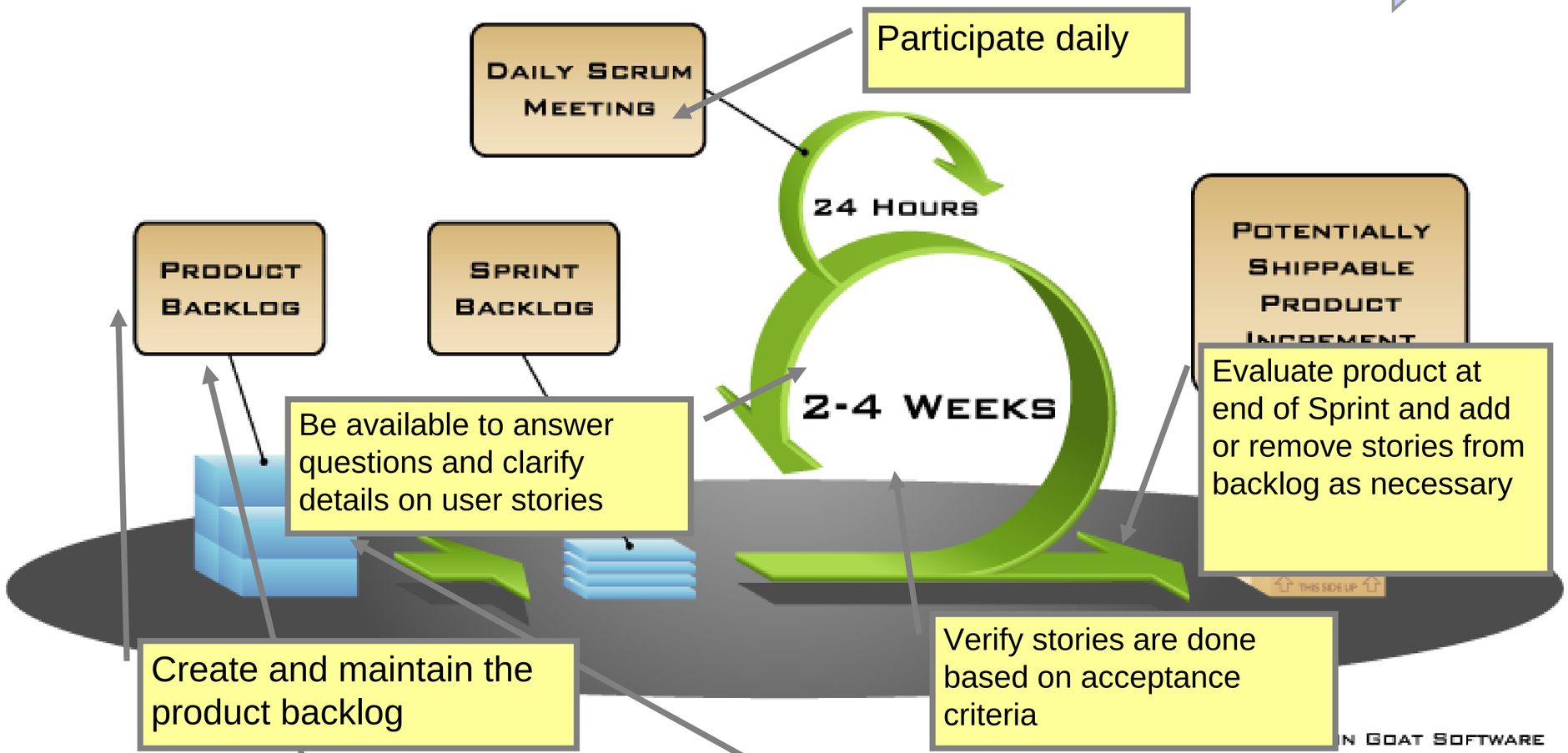
The Product Owner Is a:

- Subject Matter Expert
  - Understand the domain well enough to envision a product
  - Answer technical questions on the domain for those creating the product
- End User Advocate
  - Describe the product with understanding of users and use, and a product that best serves both
- Customer Advocate
  - Understand the needs of the business buying the product and select a mix of features valuable to the customer
- Business Advocate
  - Understand the needs of the organization paying for the software's construction and select a mix of features that serve their goals
- Communicator
  - Capable of communicating vision and intent – deferring detailed feature and design decisions to be made just in time
- Decision Maker
  - Given a variety of conflicting goals and opinions, be the final decision maker for hard product decisions

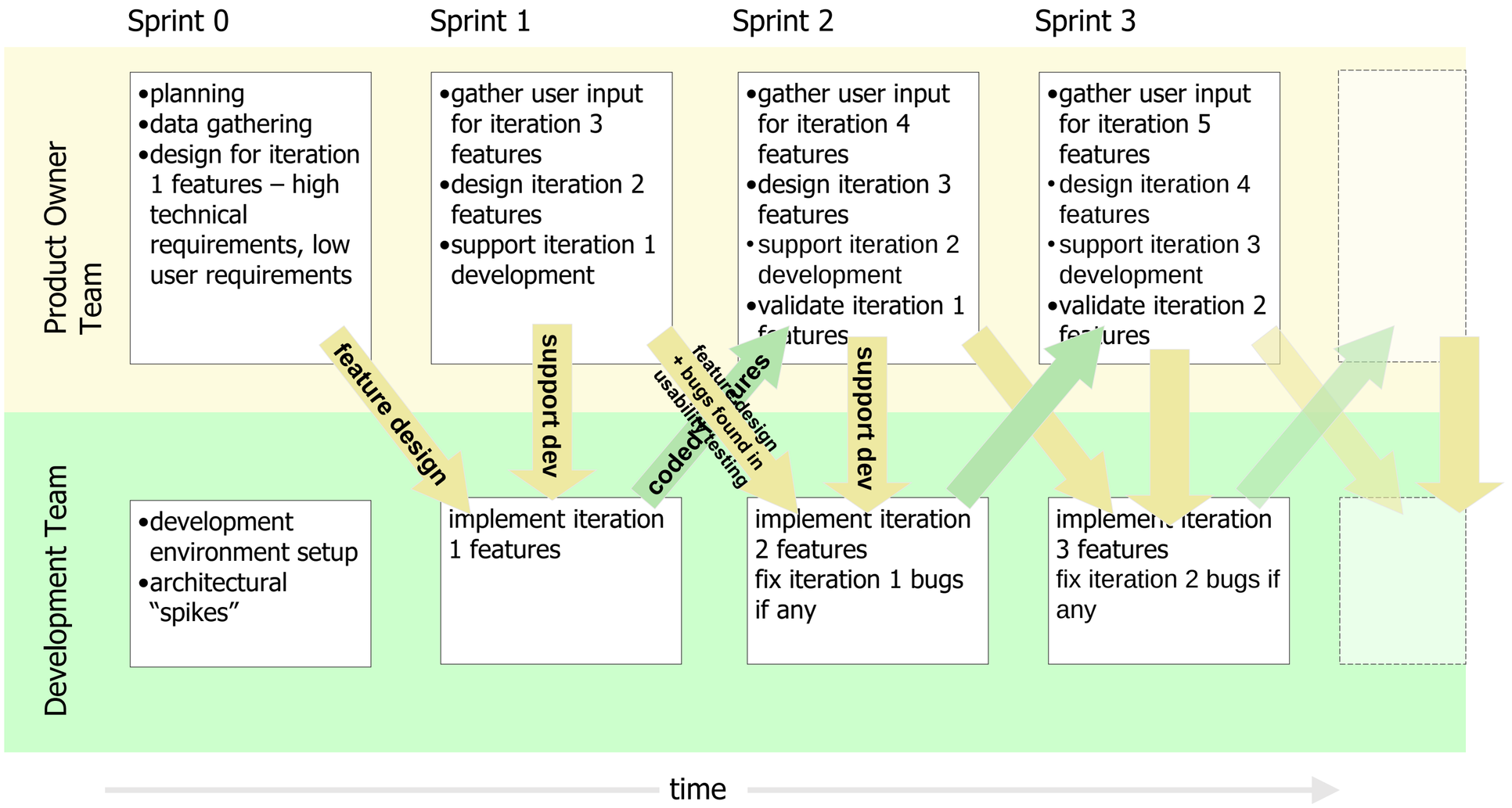
The Product Owner role is generally filled by a single person supported by a collaborative team

Prod ties

- Communicate Business Goals, Customer Goals, End User Goals
- Coordinate involvement of SMEs, users, and business stakeholders
- Coordinate with other product owners to insure coherence of product and releases



Design and Coded Features Pass Back and Forth Between Tracks



Key Differences Between

Product Owners & Project Managers

Product Manager (PM)

- Focus: Strategic vision and long-term roadmap of the product.
- Responsibilities:
  - Conduct market research and understand customer needs.
  - Define the product vision and overall strategy.
  - Collaborate with marketing, sales, and support teams.
- Level: Strategic
- Key Question: "What should we build and why?"

Product Owner (PO)

- Focus: Tactical execution within Agile teams.
- Responsibilities:
  - Manage and prioritize the product backlog.
  - Define user stories and acceptance criteria.
  - Work closely with the development team and participate in Scrum ceremonies.
- Level: Tactical
- Key Question: "How should we build it and when?"

PO vs PM vs PM!

Project Manager (PM)	Product Manager (PM)	Product Owner (PO)	Aspect
Short-term operational focus on project execution .and delivery	Long-term strategic focus on product vision and .market alignment	Day-to-day focus on managing the product backlog and delivering value .to the team	**Focus**
Plan, execute, and close projects according to scope, timeline, and .budget	Define the overall product strategy, roadmap, and .market needs	Prioritize and manage the product backlog, define user stories, and ensure the team .builds the right product	**Main Responsibility**

PO vs PM vs PM!

Project Manager (PM)	Product Manager (PM)	Product Owner (PO)	Aspect
Manage timelines, resources, and risks to .ensure project success	Set product vision, business goals, and work with various departments .to align strategies	Work closely with the development team to clarify requirements and ensure product increments meet .goals	**Key Tasks**
Short-term (focused on project completion within .time and budget)	Long-term (focused on the product's future growth and market .presence)	Short-term (focused on the current sprint and upcoming .features)	**Time Horizon**

PO vs PM vs PM!

Project Manager (PM)	Product Manager (PM)	Product Owner (PO)	Aspect
Works with project teams, stakeholders, and sometimes clients to .ensure project delivery	Works with stakeholders across the company (marketing, sales, support, .etc.)	Works closely with the development team to refine requirements and review .deliverables	**Interaction with Teams**
Responsible for delivering the project on time, within .scope, and within budget	Responsible for the product's success in the market and its strategic .direction	Responsible for ensuring the team is working on the highest priority items and .meeting customer needs	**Accountability**

PO vs PM vs PM!

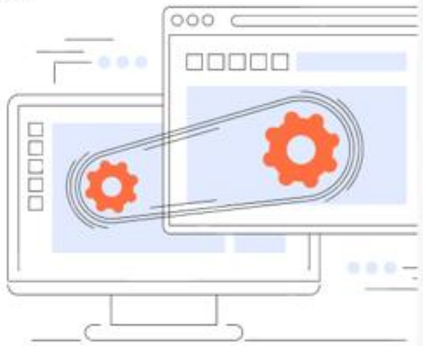
Project Manager (PM)	Product Manager (PM)	Product Owner (PO)	Aspect
Focused on the scope and execution of a specific .project	Oversees the entire product lifecycle, from .concept to market	Focused on specific features and iterations within the .product lifecycle	**Scope**
Project plans, timelines, resource allocation, and .project completion	Product roadmap, strategy, market research, .and business alignment	Product Backlog, user stories, sprint goals, and product .increments	**Key Deliverables**

Technical Product Owner (TPO)

- TPO is a specialist who acts as a bridge between the business side, which expresses customer needs, and the technical teams, which translate business requirements into understandable and feasible tasks.
- This role is that of a Product Owner's right-hand. He is responsible for the technical aspects of the product, as well as some functional and communication responsibilities.
- Having a Tech Product Owner in the team enables you to:
  - prioritize the roadmap, considering technical capabilities and constraints;
  - improve communication among the customer, development team, and product management team;
  - reduce technical debt;
  - avoid development delays and minimize risks associated with efficient resource use.

Product Owner vs. Technical Product Owner

Focus on product vs. technical aspects



Area of responsibility

Communication and stakeholders

Skills and experience

Product Owner vs. Technical Product Owner

- Focus on product vs. technical aspects
  - Product Owner's primary focus is on the value of the product to users and the business. PO is responsible for setting benchmarks and vision, backlog creation and management.
  - TPO focuses on the technical aspects. This is the choice of technologies, system architecture, ensuring the technical feasibility of functions, etc.
- Communication and stakeholders
  - Product Owner actively engages with customers, users and other stakeholders to understand their needs and product requirements.
  - TPO works closely with developers, engineers and other members of the technical team.

Product Owner vs. Technical Product Owner

- Area of responsibility
  - Product Owner: Responsible for delivering value at every stage of development and release.
  - Technical Product Owner: Responsible for technical strategy, technical leadership and ensuring technical feasibility of requirements.
- Skills and experience
  - Product Owner has a more business-oriented background: understanding the market, users, competitors. Often has experience in product management or analysis.

Some Other Roles

- Product Owner (PO) is focused on the product itself, ensuring that the team is working on the right features that deliver value.
- Scrum Master (SM) ensures that the Scrum process is followed and helps the team operate efficiently.
- Product Manager (PM) is responsible for the strategic direction of the product and its success in the market.
- Project Manager (PM) focuses on project execution, ensuring that individual projects are delivered on time and within scope.
- Program Manager (PM) manages multiple projects within a program, ensuring they align with organizational goals and are executed successfully.

Notes

- A Portfolio is a collection of projects, programs, and initiatives managed together to align with an organization's strategic objectives. It focuses on optimizing resource allocation, prioritizing efforts, and managing risks to maximize overall value.

Some Other Roles

- Release Manager (RM)
- Engineering Manager (EM)
- Quality Assurance Manager (QA Manager)
- DevOps Manager
- Business Analyst Manager (BA Manager)
- Change Manager
- Customer Success Manager (CSM)

What is the relation of these roles to scrum?

07

Scrum Master

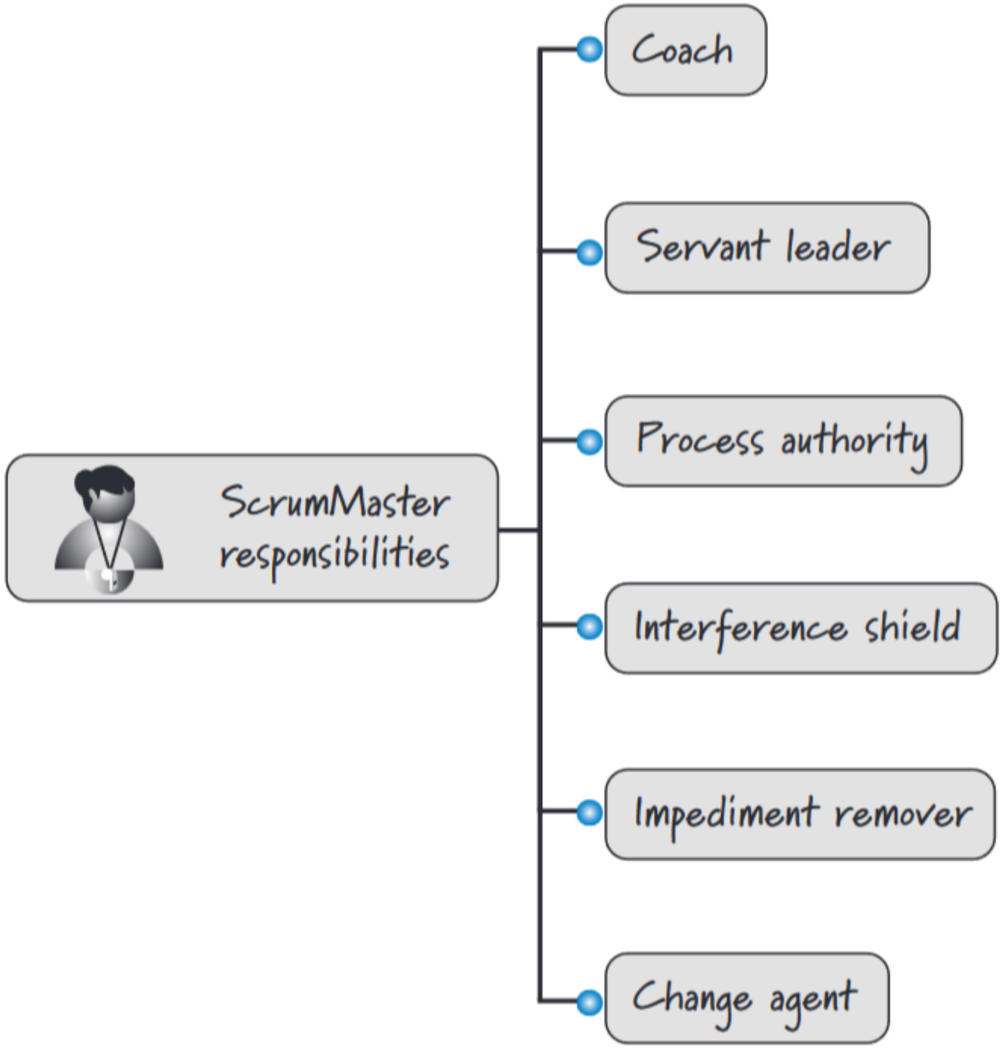

ستاوین
Website:
Setavin.com


ستاوین
Website:
Setavin.com

The Scrum Master

- Responsible for ensuring that Scrum values, practices and rules are enacted and enforced
- Nexus between management and the team
- Drives daily scrums comparing progress made vs. expected.
- Ensures impediment are quickly removed and decisions are promptly made.
- The Scrum Master and the management form Scrum teams
- The Scrum Master, the Product owner and the Scrum team create a Product backlog

Principal ScrumMaster responsibilities



1) Coach

- The ScrumMaster observes how the team is using Scrum and does anything possible to help it get to the next level of performance.
- When problems arise that the team can and should be able to solve,

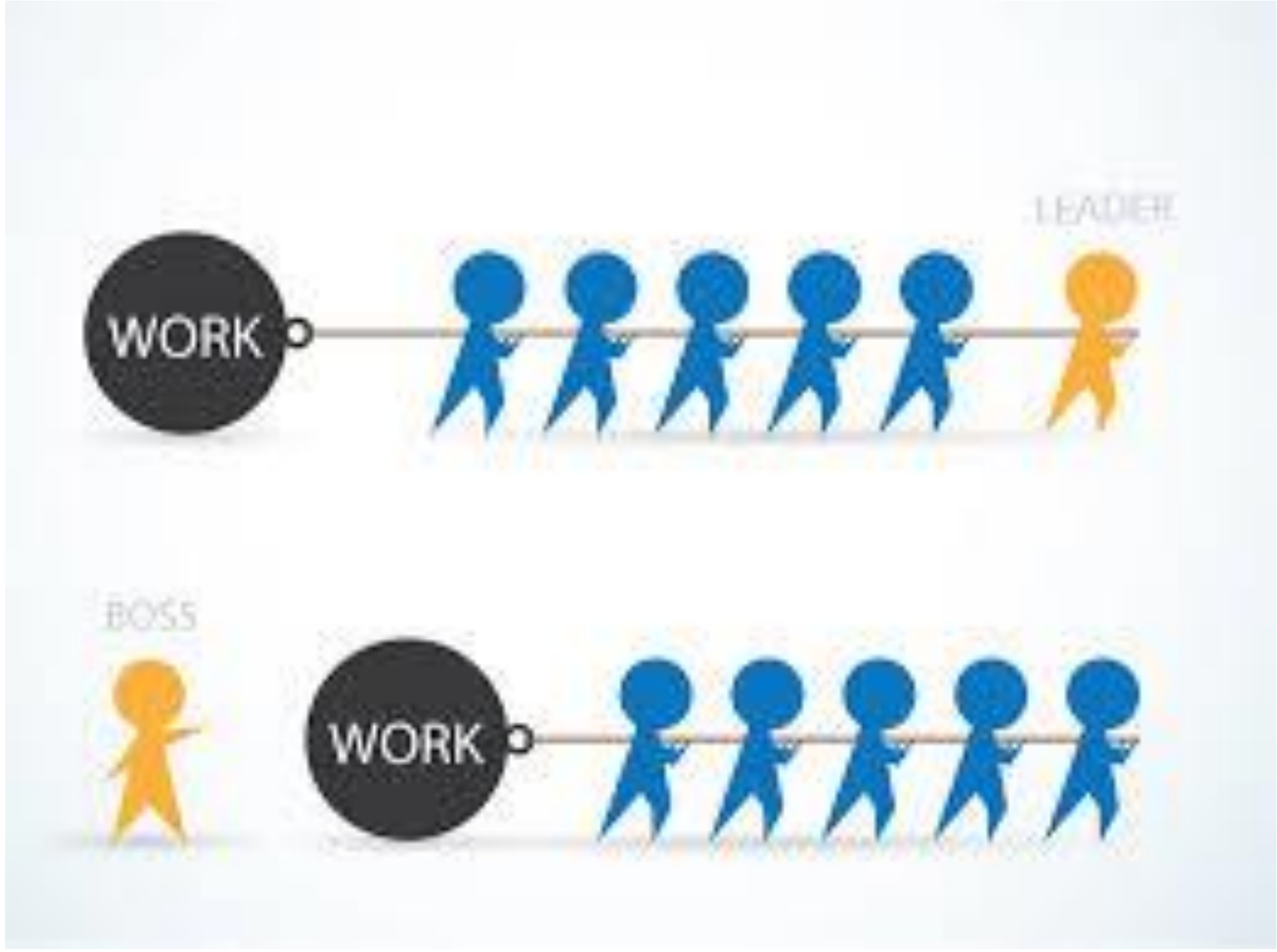
the ScrumMaster's attitude, like that of any good coach, is "I'm not here to solve your problems for you; instead, I'm here to help you solve your own problems." If the problem is an impediment that the team can't resolve, the ScrumMaster takes ownership of getting it resolved.

- The ScrumMaster coaches a new product owner by helping him understand and perform his product owner responsibilities. Once the ScrumMaster helps the product owner get established in his role, she provides him with ongoing assistance for activities such as grooming the product backlog

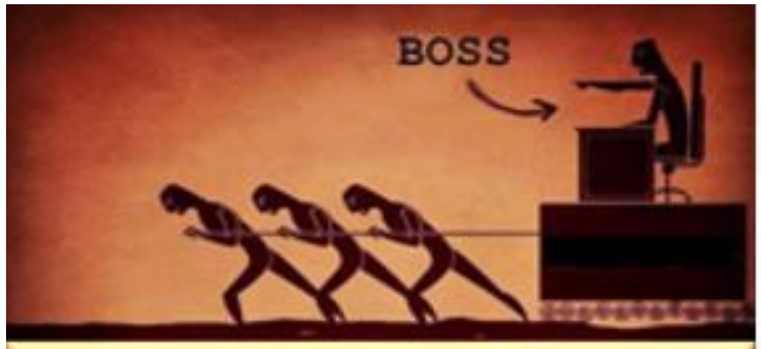
2) Servant Leader

- ScrumMaster is first and foremost a servant to the Scrum team, ensuring that its highest-priority needs are being met.
- A servant leader would never ask, “So, what are you going to do for me today?” Instead, a servant leader asks, “So, what can I do today to help you and the team be more effective?”

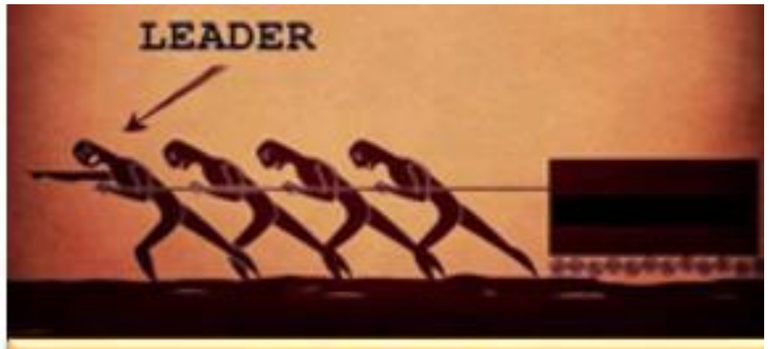
Note! Leader vs Boss



Note! Leader vs Boss



Demands
Relies on Authority
Issues Ultimatums
Uses People
Takes Credit
Places the Blame
Says “Go”
My way is the only way



Coaches
Role Models Behavior
Generates Enthusiasm
Develops People
Gives Credit
Accepts Blame
Says “Let’s Go”
I’ve got your back

3) Process Authority

- ScrumMaster is empowered to ensure that the Scrum team enacts and adheres to the Scrum values, principles, and practices along with the Scrum team's specific approaches.
- The ScrumMaster continuously helps the Scrum team improve the process, whenever possible, to maximize delivered business value.
- Authority in this context is not the same type of authority that a functional manager or project manager would have.
  - For example, the ScrumMaster doesn't hire and fire and cannot dictate to the team what tasks it should do or how to do them.
- The ScrumMaster also is not responsible for making sure the work gets done.

4) Interference Shield

- The ScrumMaster protects the development team from outside interference so that it can remain focused on delivering business value every sprint.
  - Interference can come from any number of sources, from managers who want to redirect team members in the middle of a sprint, to issues originating from other teams. No matter what the source of the interference, the ScrumMaster acts as an interceptor (fielding inquiries, addressing management, and arbitrating disputes) so that the team can focus on delivering value.

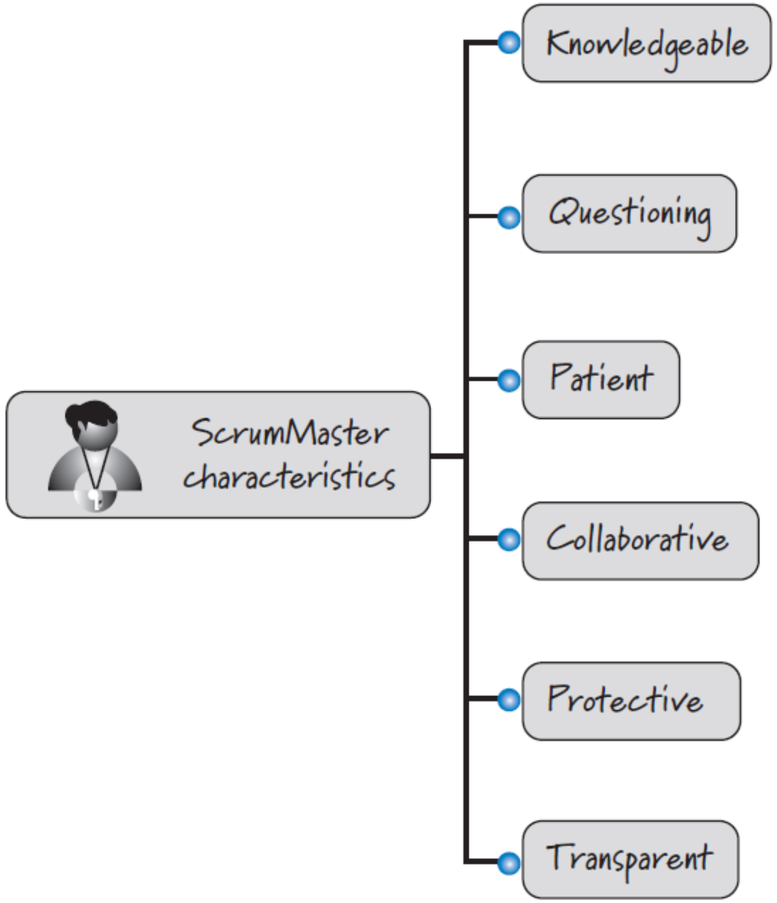
5) Impediment Remover

- The ScrumMaster also takes responsibility for removing impediments that inhibit the team's productivity (when the team members themselves cannot reasonably remove them).
  - What kind of impediments?

6) Change Agent

- Educates the organization.
- Transforms team and organizational culture.
- Removes behavioral and organizational impediments.
- Ensures everyone understands Scrum values and embraces necessary changes.

ScrumMaster characteristics



Knowledgeable

- To be an effective process coach: must be very knowledgeable about Scrum.
- The ScrumMaster should also understand the technical issues the team needs to address and technologies the team will use to create solutions.
- A ScrumMaster doesn't need to have tech-lead- or dev-lead-level knowledge, but reasonable technical knowledge is an asset.
- The ScrumMaster also doesn't need to be an expert in the business domain (the product owner does), but again, working knowledge of the business domain is very helpful.

Questioning

- ScrumMasters use their coaching skills in conjunction with their process, technical, and business knowledge to ask great questions.
- Great ScrumMasters almost never directly answer a question but instead reflexively answer with their own question

فرض کن یکی از اعضای تیم به اسکرام مستر می‌گوید:
"ما به دلیل کمبود زمان نمی‌توانیم این وظیفه را کامل کنیم. چه کار کنیم؟"

اگر اسکرام مستر مستقیم جواب دهد:
"از وظایف دیگر صرف‌نظر کنید."
این پاسخ باعث می‌شود که فرد فقط به جواب وابسته شود و خودش فکر نکند.

اما اگر اسکرام مستر بپرسد:
"آیا همه وظایف فعلی شما اولویت دارند؟ آیا چیزی هست که بتوانید برای تمرکز روی این وظیفه کنار بگذارید؟"
این سؤال باعث می‌شود که فرد به **اولویت‌بندی** فکر کند و خودش راه‌حل پیدا کند.

Patient

- Because ScrumMasters prefer not to give out answers, they need to be patient, giving teams time to arrive at appropriate answers on their own.
  - At times it is hard for me to be a ScrumMaster because I see the issue the team is dealing with and I “know” the answer. Well, at least I think I know the answer! It is arrogant for me (or any ScrumMaster) to believe that I am smarter than the collective intelligence of the team. So, at times I just have to bite my tongue and be patient, letting the team work out the solution, periodically asking probing questions to help guide things along.

Collaborative

- The ScrumMaster must have excellent collaboration skills to work with the product owner, development team, and all the other parties, even those who might not be directly involved with Scrum.

Protective

Protective

Protective

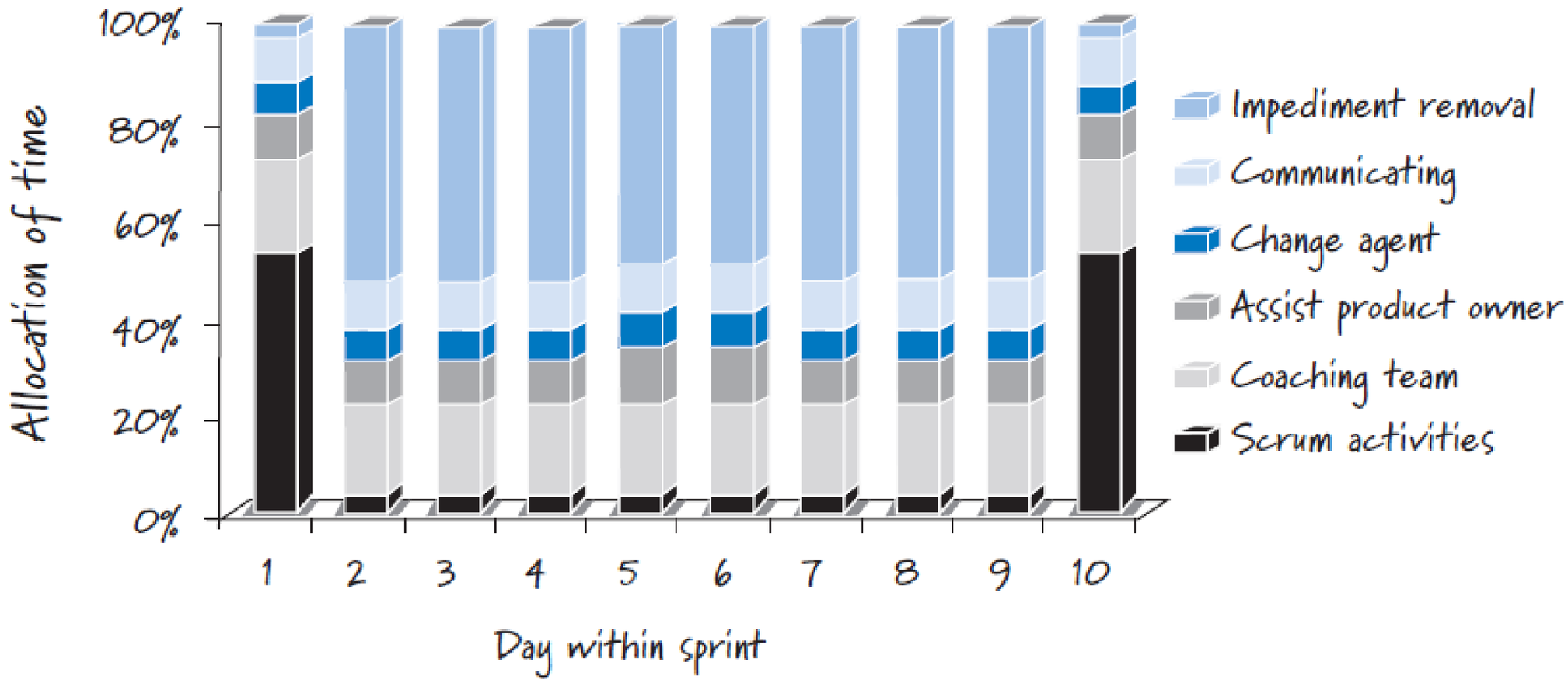
The ScrumMaster should be very protective of the team. The common analogy is that the ScrumMaster acts like a sheepdog, guarding the flock from wolves that might try to attack. In our context wolves could be organizational impediments or people with differing agendas. The ScrumMaster is adept at ensuring the protection of the team within the greater context of making economically sound business decisions. With acute sensitivity toward both team protection and business needs, the ScrumMaster helps the Scrum team achieve a healthy balance.
The ScrumMaster also helps team members who begin to wander away from the flock. When things get difficult, it is easy for people to fall back on familiar, nonagile approaches. In this case it is the ScrumMaster's job to help shepherd straying team members, helping them overcome their difficulties by reinforcing how to use Scrum more effectively.

Transparent

Finally, the ScrumMaster is transparent in all forms of communication. When working with team members,

- There is no room for hidden agendas;
- What you see and hear from the ScrumMaster must be what you get.
- People expect nothing less of a servant leader.
- The ScrumMaster also promotes transparent communication outside of the Scrum team. Without transparent access to information it is difficult for the organization to inspect and adapt to achieve its desired business results from using Scrum.

A Day in the Life



Is ScrumMaster a Full-Time Job?

- NO!!!!!
- Possibly not. A Scrum team that has been working together for an extended period of time and has become highly proficient with Scrum might require less coaching than a new team made up of people who have never worked together and are new to Scrum.

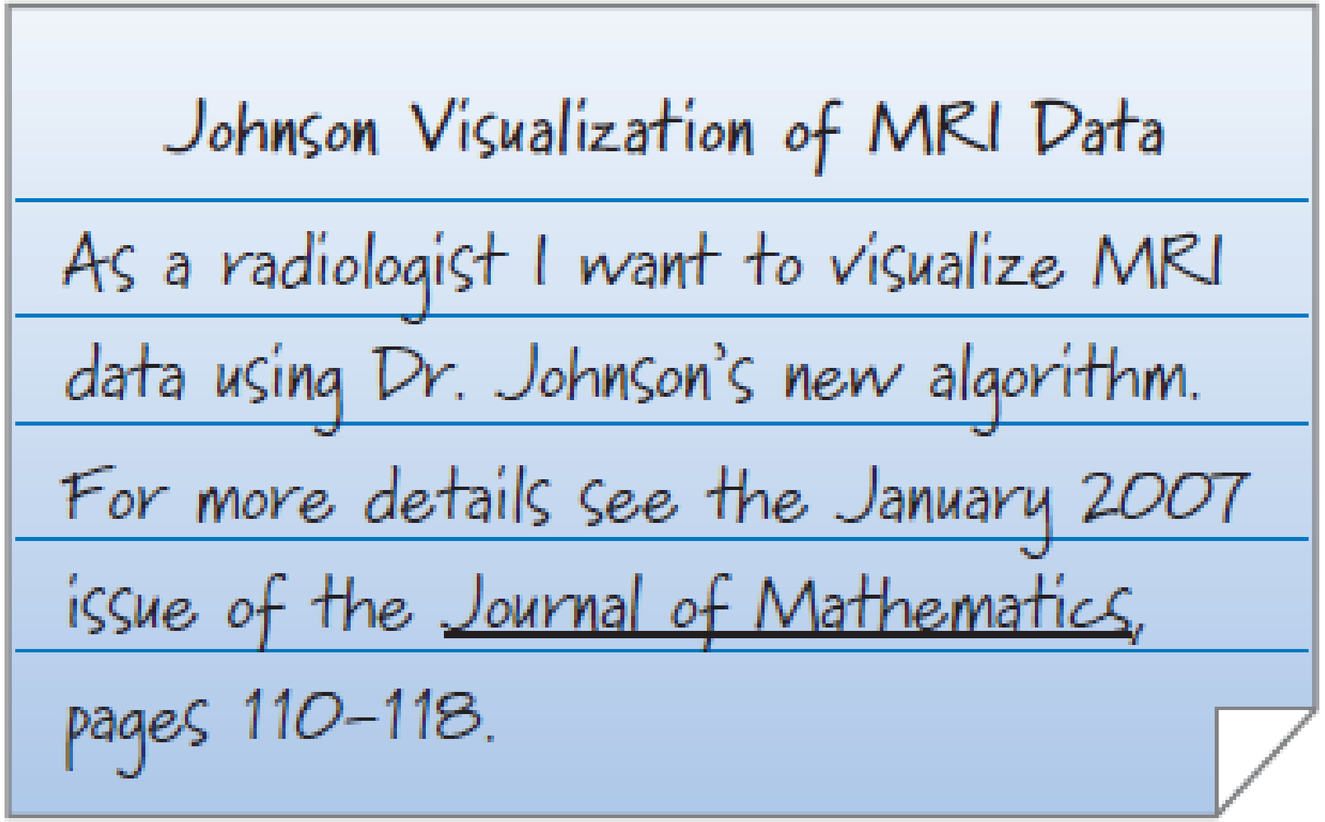
08

User Story

User Story


User Story Title
As a <user role> I want to <goal> so that <benefit>.
Template


Upload File
As a wiki user I want to upload a file to the wiki so that I can share it with my colleagues.



User story with additional data attached

User Story

- Conversation: The details of a requirement are exposed and communicated in a conversation among the development team, product owner, and stakeholders. The user story is simply a promise to have that conversation.

- Confirmation: A user story also contains confirmation information in the form of conditions of satisfaction. These are acceptance criteria that clarify the desired behavior. They are used by the development team to better understand what to build and test and by the product owner to confirm that the user story has been implemented to his satisfaction.

Level of Detail

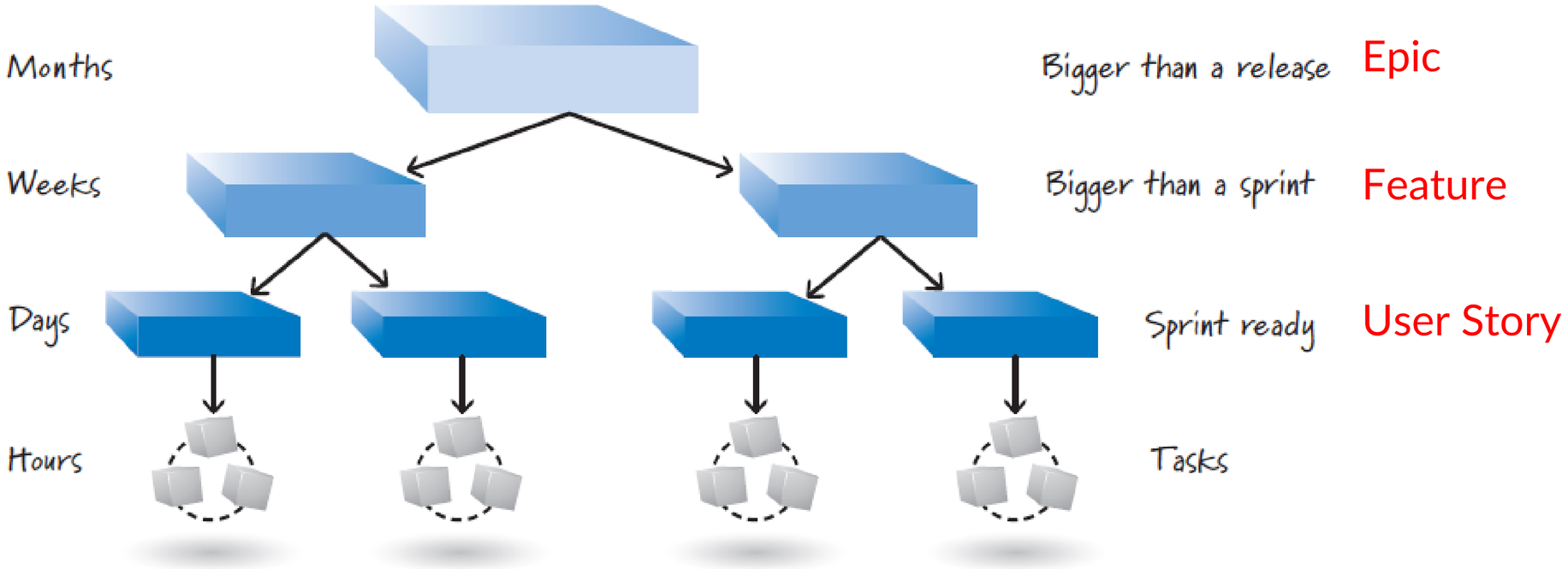
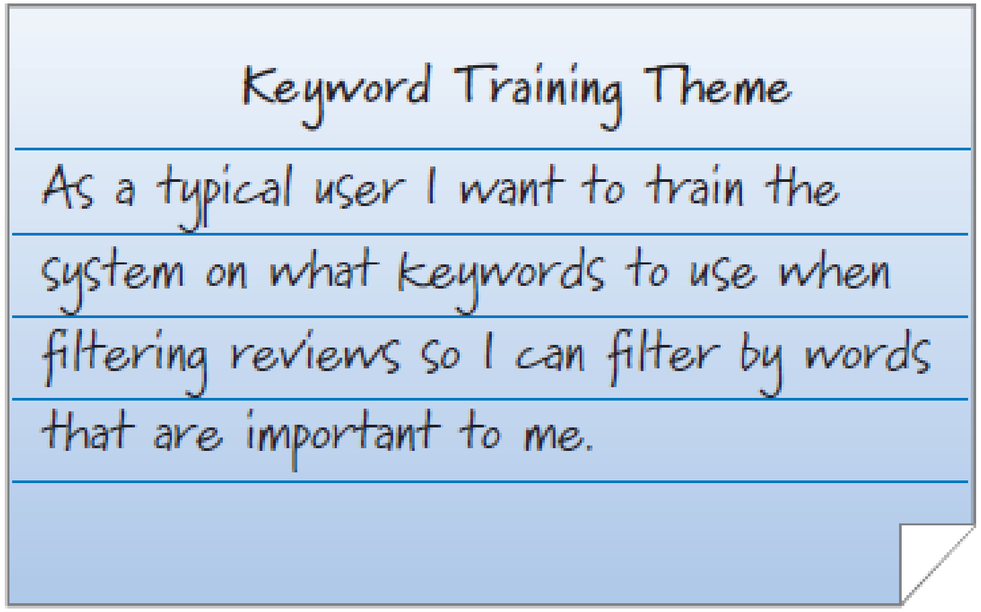


FIGURE 5.5 User story abstraction hierarchy

Preference Training Epic

As a typical user I want to train the system on what types of product and service reviews I prefer so it will know what characteristics to use when filtering reviews on my behalf.

- Some teams also use the term theme to refer to a collection of related stories.

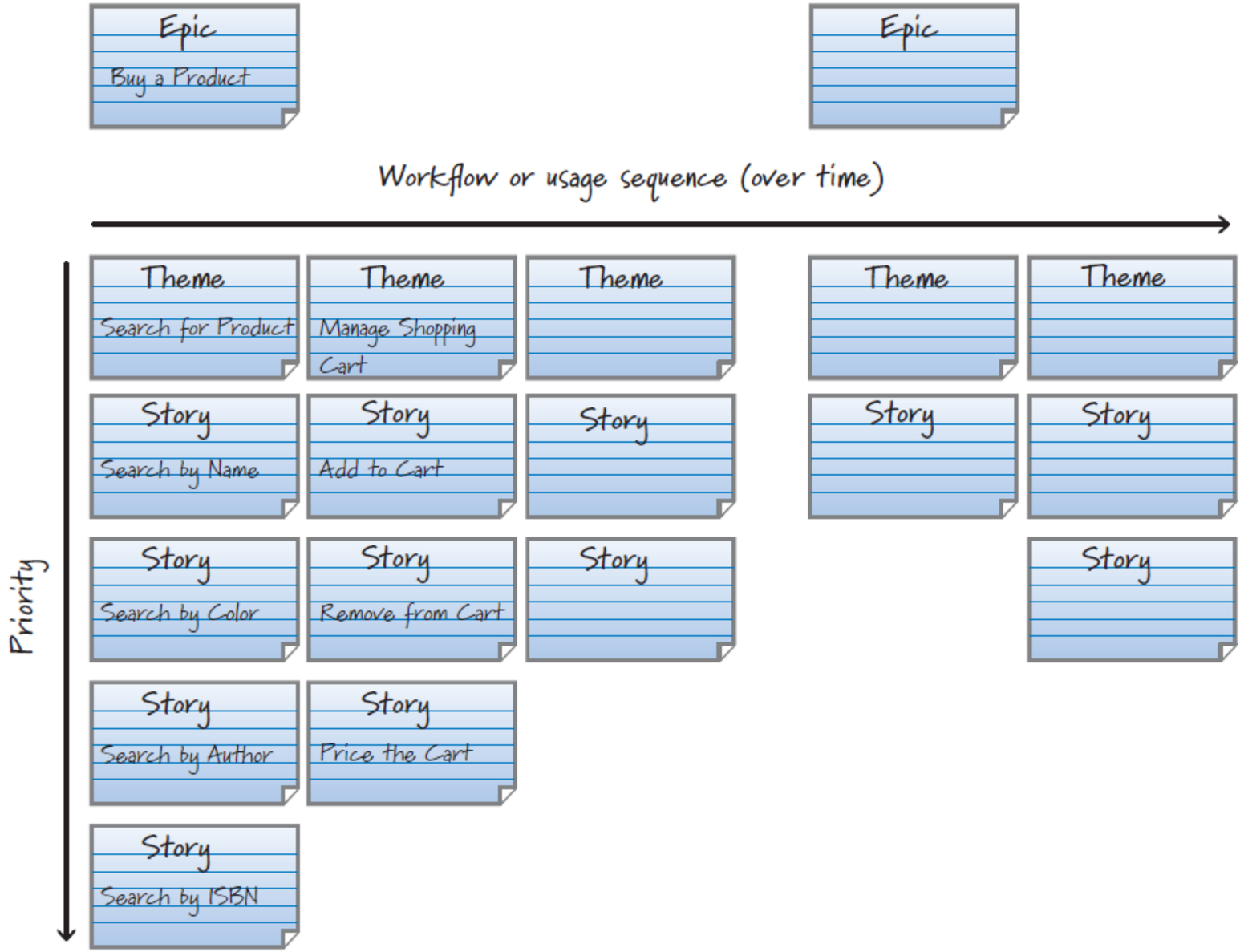


Example theme

INVEST in Good Stories

- The INVEST criteria are Independent, Negotiable, Valuable, Estimatable, Small (sized appropriately), and Testable.

Story map


Epic
Buy a Product
Epic
Workflow or usage sequence (over time)
Priority
Theme
Search for Product
Theme
Manage Shopping Cart
Theme
Theme
Theme
Story
Search by Name
Story
Add to Cart
Story
Story
Story
Story
Search by Color
Story
Remove from Cart
Story
Story
Story
Search by Author
Story
Price the Cart
Story
Search by ISBN

Theme, Epic, Feature, and User Story

Criterion	Theme	Epic	Feature	User Story
Definition	A broad topic or strategic vision	A large, actionable goal	A specific capability or part of an Epic	A small, detailed task from the user's perspective
Level	Highest	High	Medium	Lowest
Includes	Multiple Epics	Multiple Features	Multiple User Stories	Specific tasks to be done
Time Focus	Long-term (months to years)	Medium-term (a few months)	Short-term (a few Sprints)	Very short-term (a few days)
Outcome/Goal	Strategic organizational objective	Tangible, deliverable result	Specific functionality or capability	A task solving a user's need
Example	Increase digital sales	Improve online shopping experience	Add advanced search functionality	Sort results by price

Example

- Theme: Boost sales through better user experience
- └── Epic: Enhance the online shopping process
- └── Feature: Add advanced search functionality
- └── User Story: As a customer, I want to sort products by price so I can find affordable items easily.

08

Scrum Team

Scrum philosophy

"During a Scrum, the pack must work as a unit, not as 8 individuals. Everybody has a role to play. The important goal to bear in mind is that when you work well together as a unit, the whole is much greater than the sum of the parts."

The On-Line Rugby Coaching Manual

Product Backlog

PBI types (extended list)

- Feature (User Story)

Note Scrum does not mandate the User Story format!

- Change (not used at MSOE)
- **Defect**
- Internal (Technical) improvement
- Knowledge acquisition
- Epic

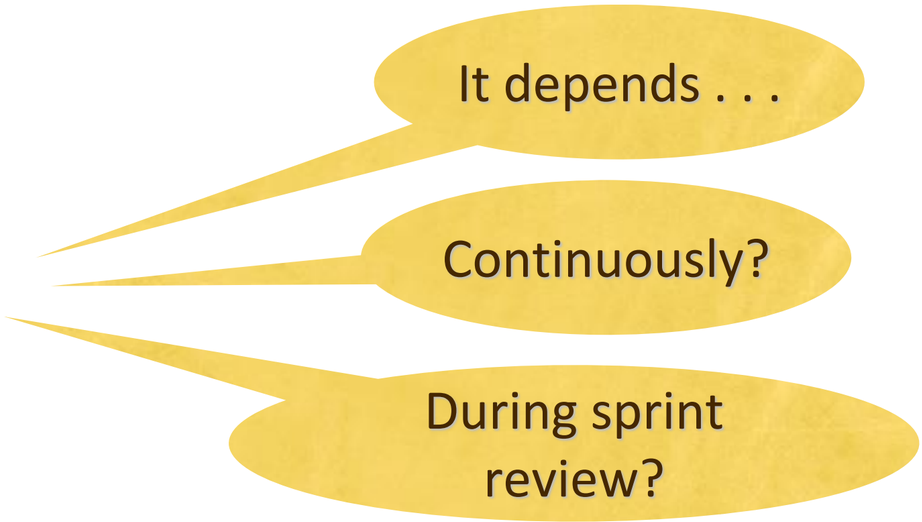
Briefly, define each type; how do **changes** differ from **defects**? Or **features**?

Good PB characteristics

- Detailed appropriately
- Emergent
- Estimated
- Prioritized

Grooming the PB

- Name/define three grooming activities
  - Hint: DEEP concepts
- Who makes the decisions?
  - Who else is involved?
  - When is it done?



Definition of Ready

Ensure top PBI's are good enough to take action on (to be worked on in a sprint)

- Business value articulated
- Details understood
- No blocking dependencies
- Small enough
- Acceptance criteria well-defined/testable

Estimation in Scrum

Estimation target	Size unit
Product backlog item (PBI)	
Task	

Fill in the blanks

More on task estimation later . . .

Estimation in Scrum

Estimation target	Size unit
Product backlog item (PBI)	Story Points (ideal days are sometimes used)
Task	Time

Reminder

For Tuesday:

- ❑ Read Ch 7
- ❑ Read through Activity 4-1
- ❑ Quiz on Ch 7

Story Points vs Time

- Why bother estimating PBI's?

  with time?

  this way…

- Why estimate PBI's with Story Points?

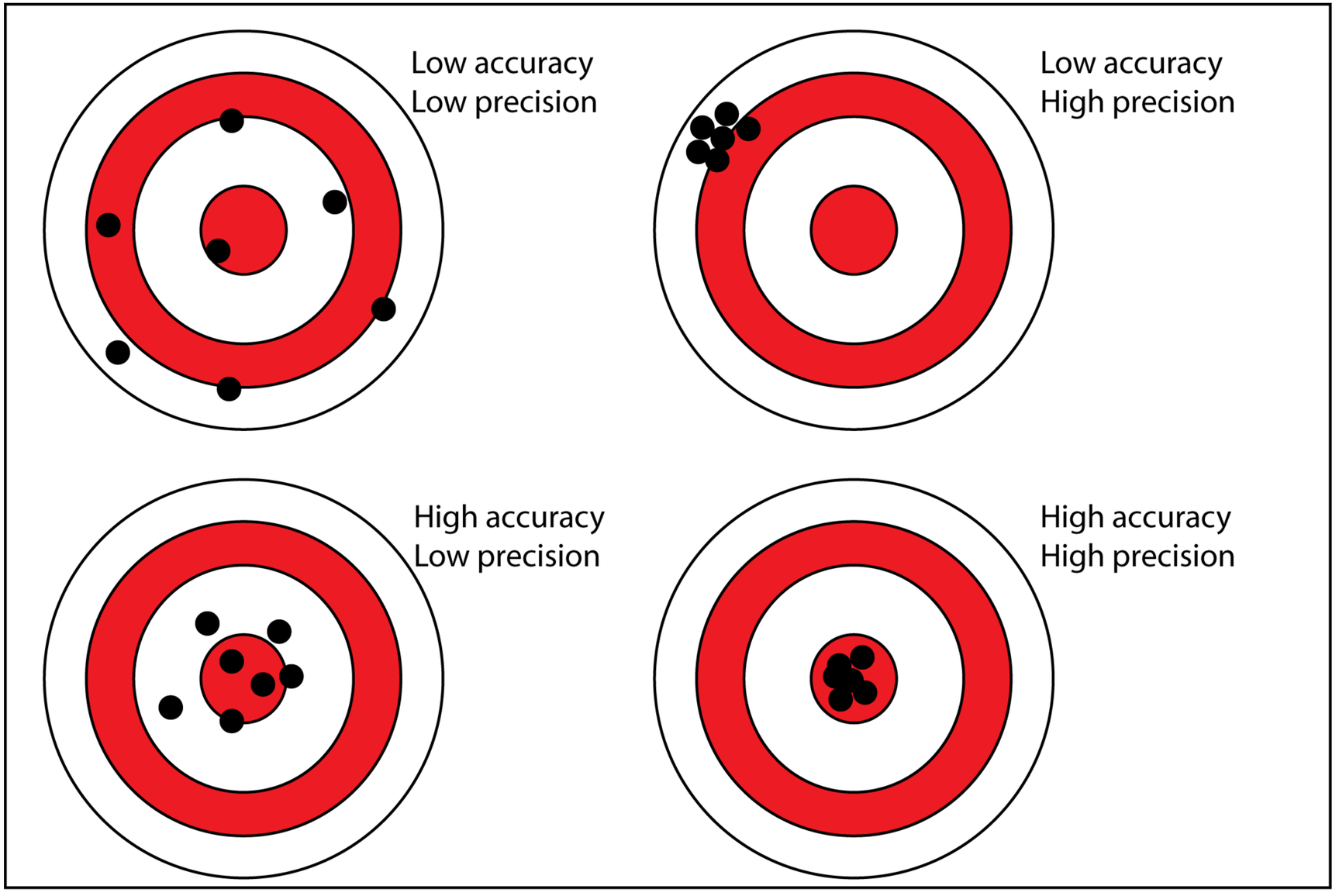
Can't we just estimate subtasks

We already get the total PBI time

PBI Estimation Concepts

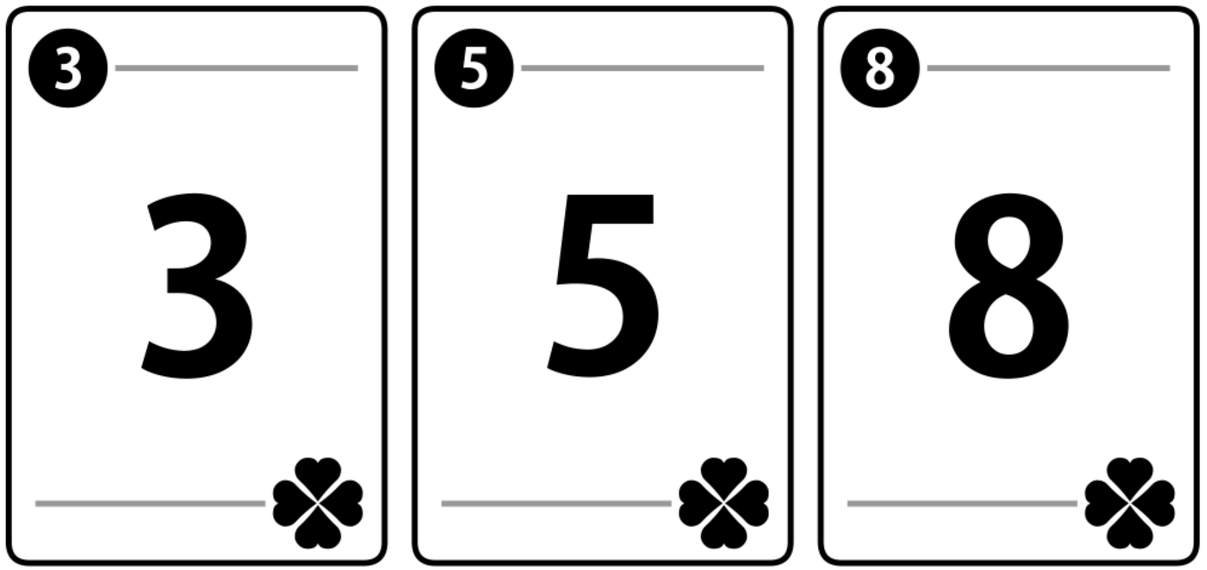
What is meant by these concepts?

- Estimate as a team
- Estimates are not commitments (or are they?)
- Focus on accuracy, not precision
- When using Story Points, use relative versus absolute sizes


Low accuracy
Low precision
Low accuracy
High precision
High accuracy
Low precision
High accuracy
High precision

- 0 - the item is already done or is so trivial (e.g. changing a class name) it doesn't warrant giving it a finite value.
- 1/2 - a tiny item - perhaps a minor defect fix, such as correcting a spelling error in a UI prompt.
- 1,2,3 - small defects, knowledge acquisitions, or stories that might take from 1 to several hours to fix, research, or implement.
- 5,8,13 - medium stories (not defects or knowledge acquisition, unless a significant re-write or amount of research is needed). A 13 is the largest single story that could be completed within a sprint. Anything larger must be broken down into smaller stories.
- 20, 40 - significant features that would take longer than a sprint to implement
- 100 - huge sets of features (comprising an entire product release)
- infinity - obviously too big to even guess at without further grooming discussion
- ? - the defect or story under consideration needs further discussion - not enough is known about the story or defect to make sense of it at all or provide an estimate. More discussion is needed.
- pi - not used in SE2800, since you won't be doing day-long or multi-day grooming. Used to indicate a break is needed during a multi-hour planning session.

Planning Poker



What is this all about?

- Consensus
- Expert opinion
- Discussion
- Relative size
- Accurate grouping
- Use of history

Why do we do it?

Why the funny values?

1/2, 1, 2, 3, 5, 8, 13, 20, 40, …

How do we play?

Playing Planning Poker

In your team:

- Choose a PBI to estimate
- Discuss the item to ensure everyone has a common understanding
- Each estimator privately chooses a virtual card

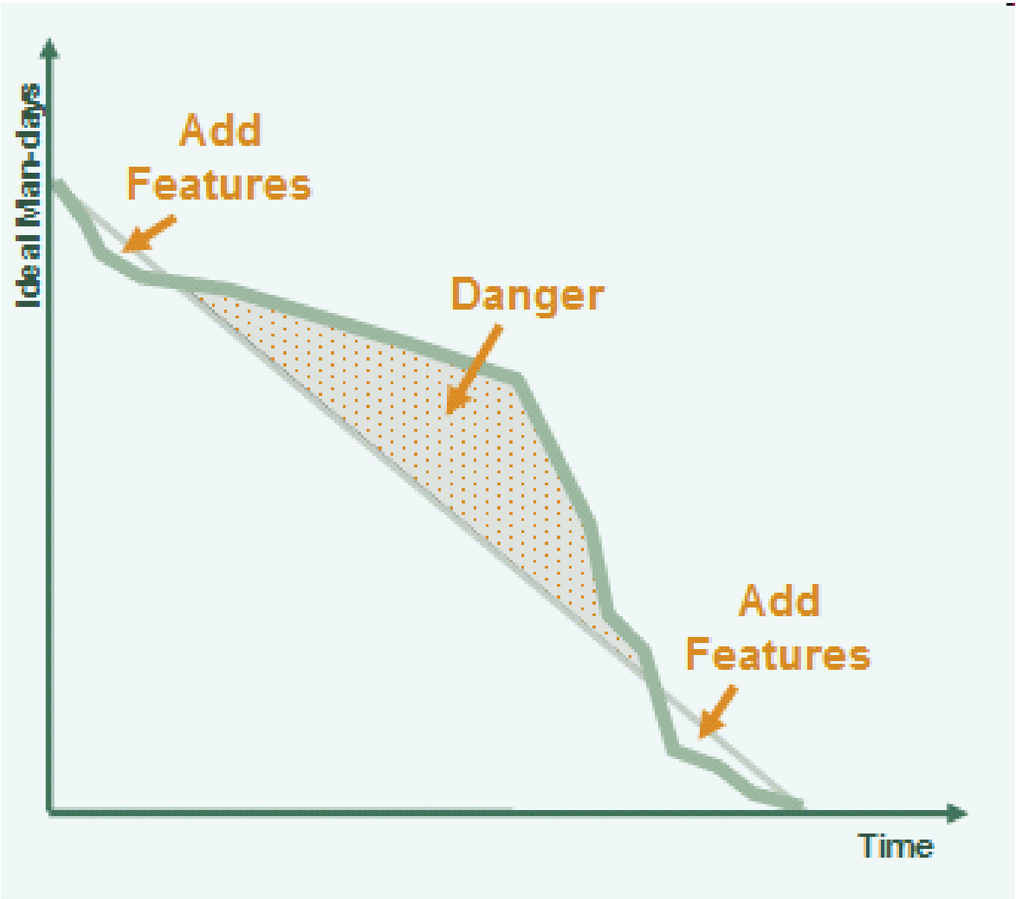
  Why?
- Show all cards at once

  Why?
- Check for consensus; discuss and repeat if none

Sprint Velocity

- What is it?
- How is it calculated?
- How do we use the velocity value?
  - For planning?
  - How about as a diagnostic?

Burndown chart



http://www.xqual.com/resources/images/scrum_burndown_chart.gif