CIZOTI NIGERIA LIMITED
With us, nothing is lost!

maps
nigeria
initiativ

Software Projects according to Agile!

For the IUFMP Stakeholders

29th – 30th June, 2020

DAY 1

Introduction to Projects

Introduction

- The ability to deliver projects on schedule, on budget and to align with business goals is key to gaining an edge in today’s highly competitive global business environment. This is where project managers come in.

- Project managers have an incredibly complex assignment, one that blends organizational skills, an analytical mind, and adept interpersonal abilities.

- In this section, we’ll walk you through the basics of project management and what it means to be a project manager.

What is a Project?

A temporary endeavor undertaken to create a unique product, service or result.

What is a Project?

- A project in somewhat looser terms is “a large or major undertaking, especially one involving considerable money, personnel and equipment.

- An individual or collaborative enterprise that is carefully planned to achieve a particular aim.

What is a Project? (Cont.)

Temporary

- A project is temporary because it has a defined beginning and end in time, and therefore defined scope and resources.

Unique

- A project is unique because it does not involve routine operations, but a specific set of operations designed to accomplish a singular goal.

So a project team often includes people who don’t usually work together – sometimes from different organizations and across multiple geographies.

Project Management Institute

Project Components

Every project <u>must</u> have the following components:

- Goal: What are you trying to achieve?
- Timeline: When are you trying to achieve it?
- Budget: How much will it cost to achieve?
- Stakeholders: Who are the major players interested in the project?
- Project Manager: Who ensures that every task gets completed?

A project is not something routine. Day-to-day operations or maintenance is not considered a project because it does not have a definitive start and end.

Project Examples

- Development of Software to Improve Business Process
- Relief Effort after a Flood
- Construction of a Bridge
- Construction of a Building
- Sales Expansion into a New Geographical Market
- A Wedding Ceremony
- House Movement

All are Projects!

Projects must be expertly managed to deliver the on-time, on-budget results, learning and integration that organizations need.

What is Project Management?

Project Management is the application of knowledge, processes, skills, tools, and techniques to project activities to meet the project requirements.

PMI®
Project Management Institute

10 Knowledge Areas

Project Management Knowledge Areas include:

- Integration
- Scope
- Time
- Cost
- Quality
- Procurement
- Human resources
- Communications
- Risk management
- Stakeholder management

Project Management Institute

10 Knowledge Areas

Project Management Knowledge Areas include:

- **Project Communication Management**:
  - Disseminates information among team members and external stakeholders
  - Ensures that information is exchanged continuously and understood by all concerned.

- **Project Cost Management**:
  - Involves processes regarding budgets, funding, spending allocation and timing.
  - Cost management is dependent on activity estimates from time management.

- **Project Human Resources Management**:
  - Involves managing the project team, like sourcing, hiring and assigning roles.
  - Also involves professional development and fostering team spirit.

10 Knowledge Areas

Project Management Knowledge Areas include:

- **Project Integration Management**:
    - These are processes necessary to define, consolidate, and coordinate all the other processes and project management activities.
    - The above processes are key to setting expectations and keeping communication lines open.

- **Project Procurement Management:**
    - This involves planning, budgeting and purchasing resources; could be physical or informational, in order to complete a task.

- **Project Quality Management:**
    - This defines the success of a project or criteria for the completion of a project.
    - Quality is managed at every stage of the project from planning to the continuous performance improvement.

10 Knowledge Areas

Project Management Knowledge Areas include:

- **Project Risk Management:**
  - Processes involved with preparing for and managing unexpected risks.

- **Project Scope Management:**
  - Involves managing the scope or parameters of a project.
  - These processes ensure that the scope is well-defined and that all requirements remain within the scope limits.

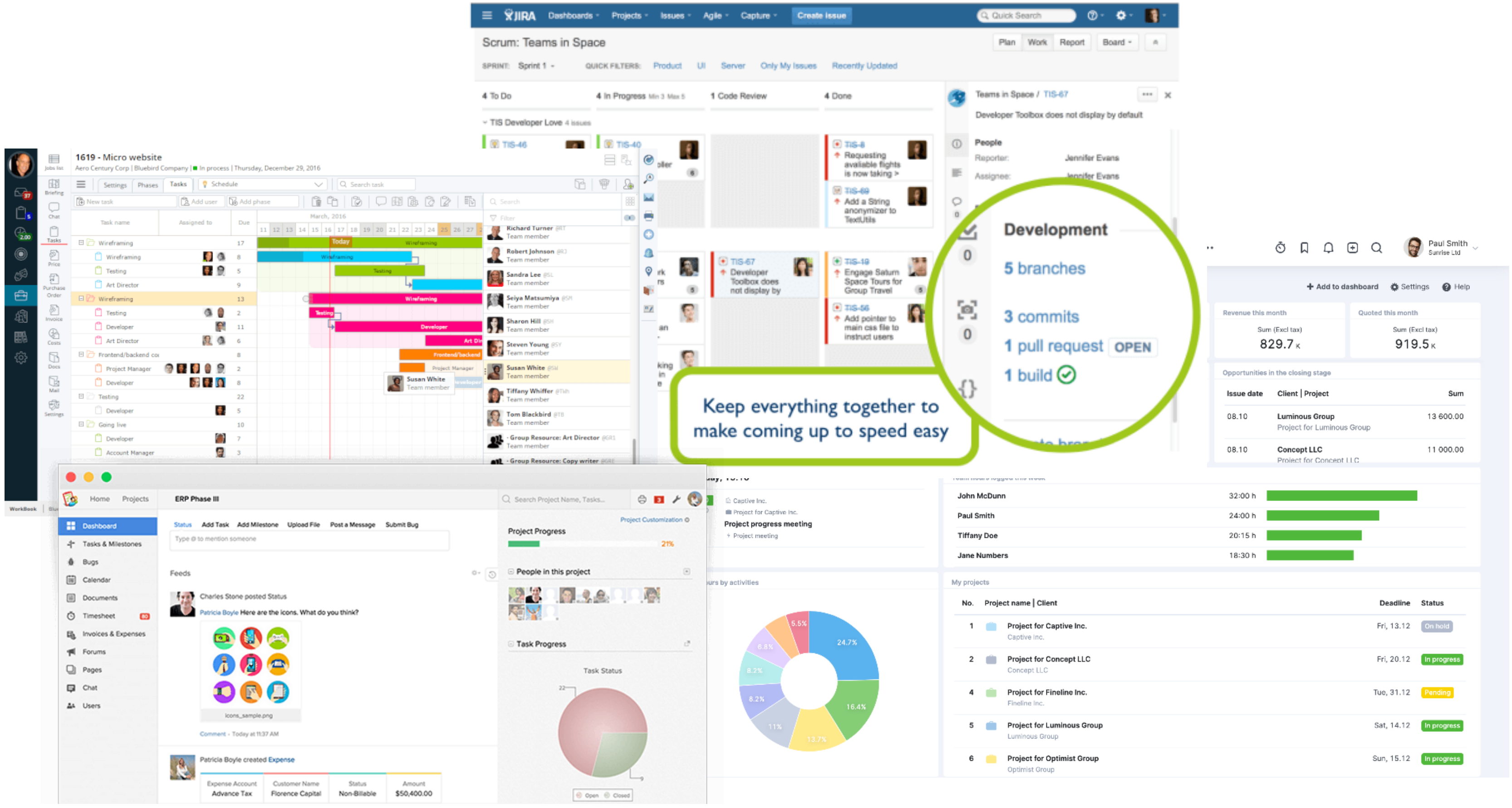
- **Project Stakeholder Management:**
  - **I**dentifying who will be impacted by the project and managing relationships with them, including strategies for collaborating with stakeholders on project direction and execution.

- **Project Time Management:**
  - This is needed to ensure the project is completed before the specified deadline.

Tools

Examples of Project Management Software Tools include:

- Wrike
- Microsoft Projects
- Zoho
- Asana
- Btrix24
- Jira
- Scoro
- Advantage
- Proworkflow
- Celoxis
- Clarizen
- Integrify
- FunctionFox



https://www.scoro.com/blog/best-project-management-software-list/

How to select the best tool?

There are a variety of project management software, but choosing the right one for you and your team is critical. Here are some questions to consider:

- Will my team actually use it?
- Can multiple departments use it?
- Does it support transparency and clear communication?
- Is it flexible?
- Can you generate custom reports?
- Does it easily (and securely) allow for external communication/users?
- Does it integrate with other tools we use?

- Think about the long-term strategy when selecting your software.

- Ensure that your team can grow and adapt to it.

20 Project Management Skills

Examples of Project Management Skills:

- Communication
- Leadership
- Organization
- Negotiation
- Team management
- Time management
- Risk management
- Problem-solving
- Budget/Finance/Accounts management
- Motivation
- Technical writing
- Adaptability
- Technologically savvy
- Reporting skills
- Active listening
- Research skills
- Interpersonal skills
- Project management methodologies
- Policy knowledge
- Conflict management

Project Management Techniques

Project Management Techniques include:

- Classic Technique
- Waterfall Technique
- Agile Project Management
- Rational Unified Process
- Program Evaluation and Review Technique
- Critical Path Technique
- Critical Chain Technique
- Extreme Project Management
- Adaptive Project Framework
- Process Based (Lean)
- Six Sigma
- Prince 2

Project Management Techniques in Focus

For the purpose of this training, Agile will be the methodology of focus.

We will also highlight the benefits of Agile frameworks which include techniques such as Scrum and Kanban

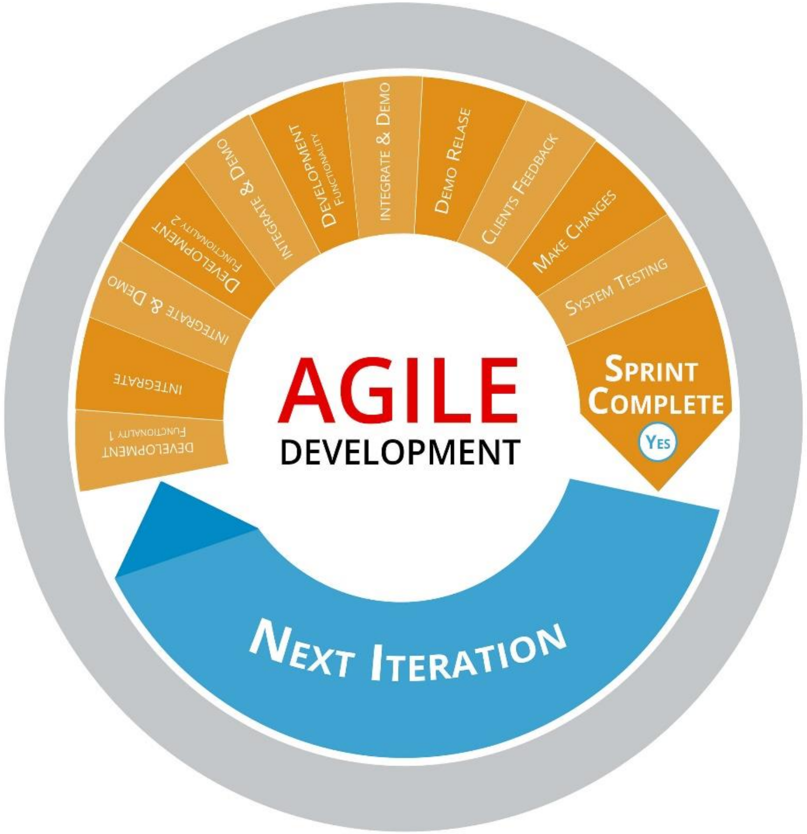
Agile Technique

- Agile project management method is a set of principles based on the value-centered approach.

- It is used in software development projects that involve frequent iterations and are performed by small and highly collaborative teams.

- It prescribes dividing project work into:
  - short sprints
  - using adaptive planning and continual improvement,
  - fostering teams’ self-organization
  - collaboration targeted at producing maximum value.

- Agile frameworks include techniques such as Scrum, Kanban, DSDM, FDD, etc.



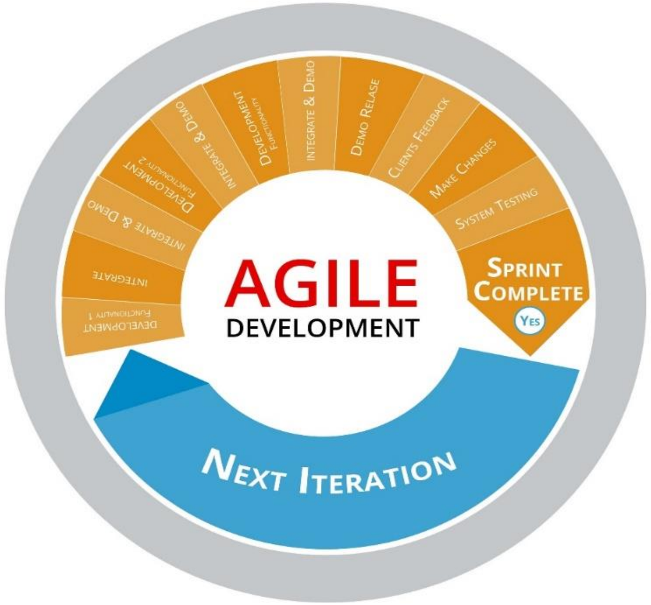
Scrum Technique

- Scrum is the most popular agile development framework

- It is relatively simple to implement

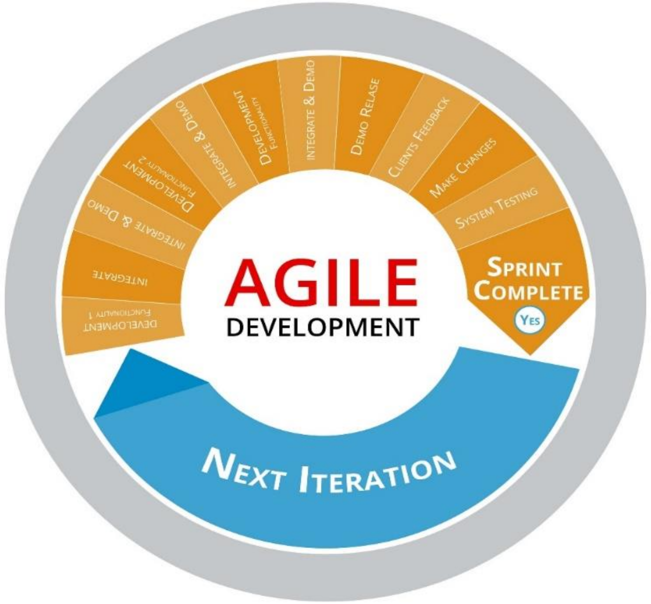
- It also solves the problems of:
  - convoluted development cycles
  - inflexible project plans
  - shifting production schedules.

- The Scrum methodology allows for quick development and testing, especially within small teams.



Kanban Technique

- Kanban is another framework for implementing agile
- It is based on a team's capacity to do work
- It requires project teams to create visual representations of tasks, often using sticky notes and whiteboards (or virtual versions that can be used online)
- These notes or tasks are moved through predetermined stages to see progress as it happens and identify where roadblocks occur



How do I choose the right technique?

A. Start With the End in Mind

B. Assess What’s Already Working

Importance of Project Management

Project Management enables your organization to:

- have a more predictable project planning and execution process
- adhere to project budgets, schedules, and scope guidelines
- resolve project roadblocks and escalate issues quicker and easier
- identify and terminate projects that do not have relevant business value
- become more efficient
- improve collaboration across and within teams
- identify and plan for risks

Your role as a Project Manager

Project Manager roles are as follows:

- **Build the plan**:
  - In charge of plotting out the most realistic course for the project
  - The plan must include the project scope, timeline, budget and identifying the right tools for the job.

- **Assemble the team**:
  - Identifying the proper team is critical to project success.
  - Every project team will vary depending on the scope of the the project.
  - Finding specialists and experts for each of the necessary tasks is ideal.

- **Assign tasks**:
  - Project managers must provide their team with a clear definition of specific tasks and timeline for every part of the project.

Your role as a Project Manager

Project Manager roles are as follows:

- **Leading the team**:
    - This will include checking in on individuals for status updates
    - Identifying and clearing roadblocks
    - Negotiating disagreements and keeping team morale high
    - Providing training and mentoring.

- **Managing budget**:
    - Requires understanding of a project budget and managing cost is critical for success.
    - Involves comparing real-life expenditures to estimates
    - Involves adjusting the project plan if necessary.

- **Managing timelines**:
    - Requires setting realistic deadlines throughout the lifecycle of the project
    - Communicating consistently with team members for status updates
    - Maintaining a detailed schedule.

Your role as a Project Manager

Project Manager roles are as follows:

- **Engaging stakeholders**:
    - Project managers need to maintain a good relationship and an open line of communication with stakeholders
    - This will clear roadblocks and empower the project team

- **Document the process:**
    - Identifying and documenting “lessons learned” for personal project manager growth
    - Helps relay experience to other teams around the organization for future use.

- **Handover the project:**
    - The project manager must now deliver the project to the team who will be managing, maintaining, and operating it moving forward. At this point, the project manager will no longer be the “go to” person, and will be assigned to a new project.

Project Life Cycle

5 Process Groups/Stages

Project Management Process Groups include:

- Initiating:
  - The goal is to define the project.

- Planning:
  - Includes developing a roadmap for everyone to follow.

- Executing & Monitoring:
  - The project team is built and deliverables are created.
  - Project managers will monitor and measure project performance to ensure it stays on track.

- Closing:
  - The project is completed
  - A post mortem is held
  - The project is transferred to another team who will maintain it.

Initiation Phase

- The initiation phase is the first phase of the entire project management life cycle.

- The goal is to define the project, develop a business case for it, and get it approved.

During this time, the project manager may do any of the following:

  - Perform a feasibility study
  - Create a project charter
  - Identify key stakeholders
  - Select project management tools

By the end of this phase, the project manager should have a high-level understanding of the project purpose, goals, requirements and risks.

Planning Phase

- The planning phase is critical to creating a project roadmap for the entire team to follow
- This is where all of the details are outlined and goals are defined in order to meet the requirements laid out by the organization.

During this phase, project managers will typically:

- Create a project plan
- Develop a resource plan
- Define goals and performance measures
- Communicate roles and responsibilities to team members
- Build out workflows
- Anticipate risks and create contingency plans

Execution Phase

The phase typically begins with a project kickoff meeting where the project manager outlines the project objectives to all stakeholders involved.

Before that meeting happens, it is crucial for the project manager to do the following:

- Establish vision and deliverables
- Identify team and set roles
- Develop initial project plan
- Define metrics for success
- Identify potential risks and bottlenecks
- Establish logistics for team communication
- Choose work process or project management methodology
- Decide which tools you’ll use
- Schedule the kickoff meeting
- Set the agenda and prepare the slides for the meeting

Monitoring and Control Phase

- This phase happens in tandem with the execution phase
- During the controlling and monitoring phase, project managers may have to do any of the following:
    - Manage resources
    - Monitor project performance
    - Risk management
    - Perform status meetings and reports
    - Update project schedule
    - Modify project plans

At the end of this phase, all of the agreed project deliverables should be completed and accepted by the client.

Project Closure Phase

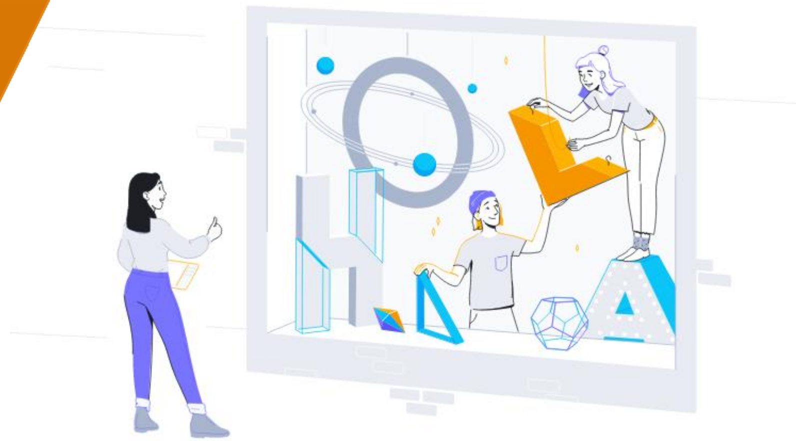
- The closing phase is a critical step in the project management life cycle.
- It signals the official end of the project
- Provides a period for reflection, wrap-up and organization of materials.

Project managers can:

- Take inventory of all deliverables
- Tie up any loose ends
- Hand the project off to the client or the team that will be managing the project's day-to-day operations
- Perform a post-mortem to discuss and document any learnings from the project
- Organize all project documents in a centralized location
- Communicate the success of the project to stakeholders and executives
- Celebrate project completion and acknowledge team members

Task 1

Project Team



Project Team

- Assemble several best practices for setting up and managing your project team to encourage collaboration for optimal performance.

- Work with individual team members to function as a cohesive unit throughout the entire project life cycle.

- Tackle different work habits, communication styles, short-term and long-term goals that could throw the entire team (and the project) off track.

- Establish the values for the team from the beginning.

Importance of Collaboration

- Internal team collaboration increases productivity

- Collaborating with external stakeholders increases innovation

What makes a Successful Team?

- **Strong team leadership**
- **Clear goals and purpose**
- **Standardized operating procedures**
- **Diversity**
- **Bonding time**

How to Setup a Project Team

- A project team is a group of people who are all working towards a common goal by bringing valuable and unique skills to the table.

So what should you consider when assembling your project team?

Project Needs:

  - Understand the scope of the project first
  - This allows you to strategically choose who needs to be on the team.

Skill set:

  - Select team members who can offer a diverse set of unique and relevant skills .
  - Avoid too many people with the same skills on a single project

Capacity:

- Select team members who will be available for upcoming project schedules.

Work styles:

- Understand how different work styles and personalities may affect individual team members the project.

Task 2

Software Projects Management

Software Project Management

- It is an art and science of planning and leading software projects.

- It is a sub-discipline of project management in which software projects are planned, implemented, monitored and controlled.

- It involves the planning, scheduling, resource allocation, execution, tracking and delivery of software and web projects.

Software Project Management

- It is distinct from traditional project management

- It has a unique lifecycle process that requires multiple rounds of testing, updating and customer feedback.

Most IT-related projects are managed in the agile style in order to:

  - keep up with the increasing pace of business
  - iterate based on customer and stakeholder feedback.

Software Development Life Cycle (SDLC)

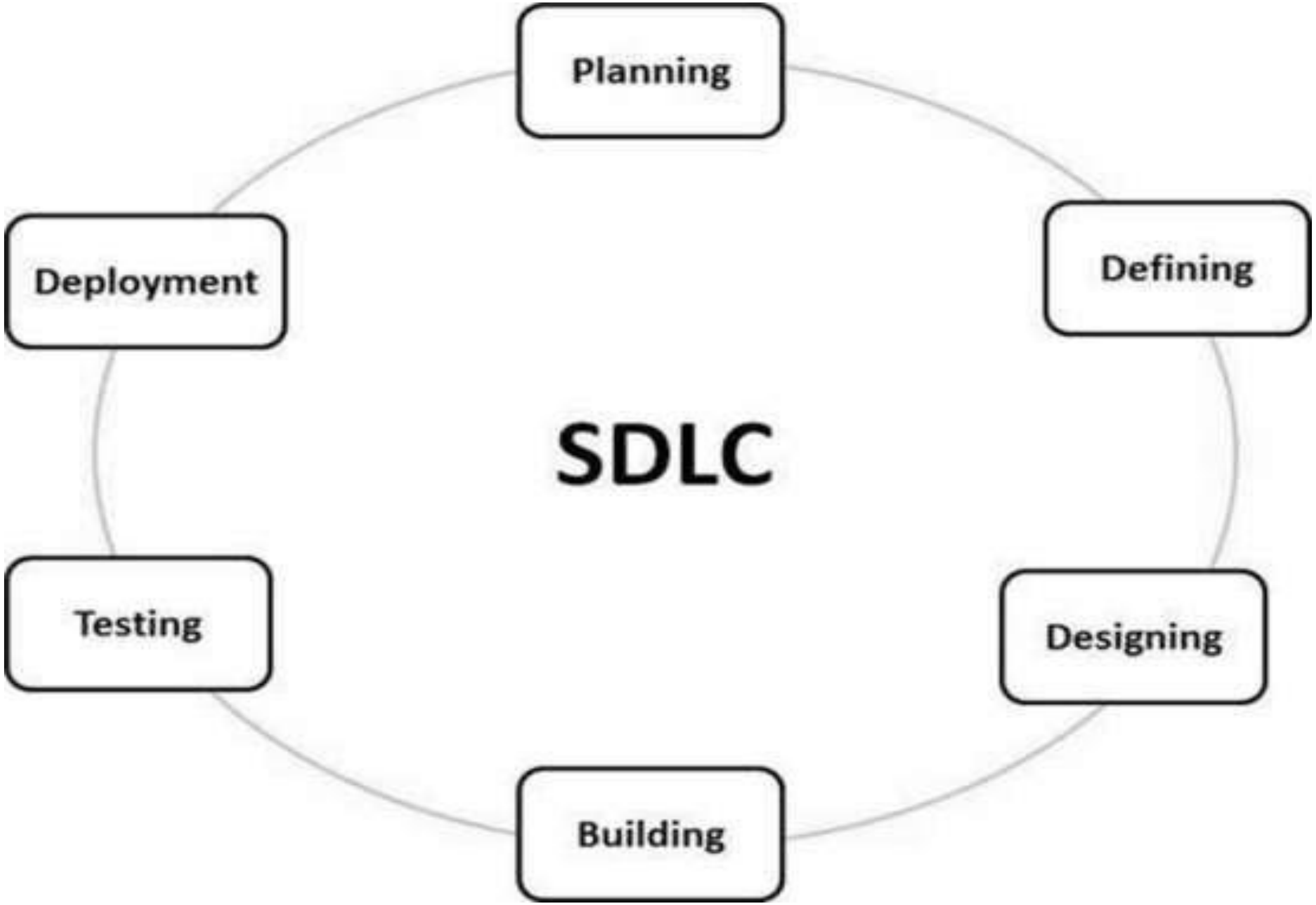
- ISO/IEC 12207 is the international standard for software life-cycle processes.

- SDLC is a framework for defining tasks performed at each step in the software development process.

- It is also called a Software Development Process

- It is a process used by the software industry to
    - design
    - develop
    - test high quality software

Software Development Life Cycle



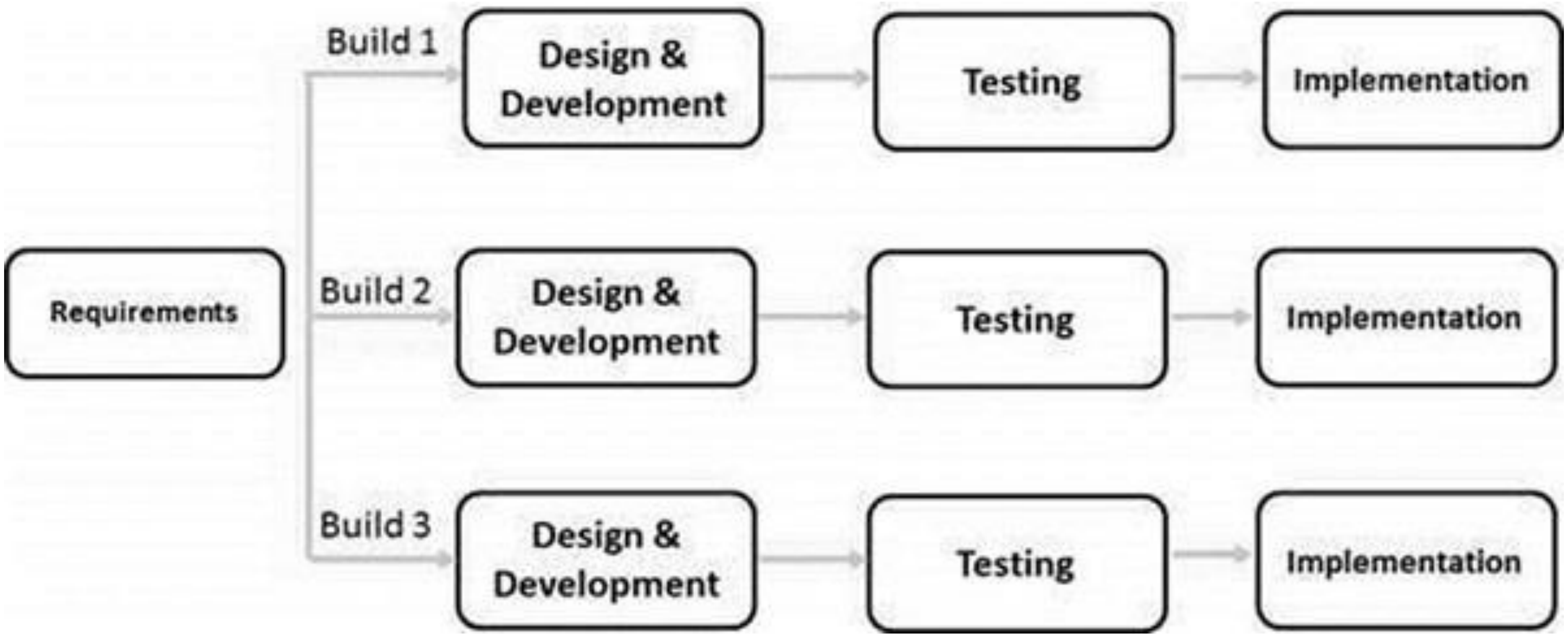
Software Development Stages

- **Planning and Requirement Analysis**
- **Defining Requirements**
- **Designing the Product Architecture**
- **Building or Developing the Product**
- **Testing the Product**
- **Deployment in the Market and Maintenance**

SDLC Models

- **Iterative Model**
- **Agile Model**
- **Waterfall Model**
- **Industry**
- **Spiral Model**
- **V-Model**
- **Big Bang Model**
- **RAD Model**

Iterative Model



Iterative Application

This model is most often used in the following scenarios −

- Requirements of the complete system are clearly defined and understood.
- Major requirements must be defined; however, some functionalities or requested enhancements may evolve with time.
- There is a time to the market constraint.
- A new technology is being used and is being learnt by the development team while working on the project.
- Resources with needed skill sets are not available and are planned to be used on contract basis for specific iterations.
- There are some high-risk features and goals which may change in the future.

Iterative Advantages

- **Some working functionality can be developed quickly and early in the life cycle.**
- **Results are obtained early and periodically.**
- **Parallel development can be planned.**
- **Progress can be measured.**
- **Less costly to change the scope/requirements.**
- **Testing and debugging during smaller iteration is easy.**
- **Risks are identified and resolved during iteration; and each iteration is an easily managed milestone.**
- **Easier to manage risk - High risk part is done first.**
- **With every increment, operational product is delivered.**
- **Issues, challenges and risks identified from each increment can be utilized/applied to the next increment.**
- **Risk analysis is better.**
- **It supports changing requirements.**
- **Initial Operating time is less.**
- **Better suited for large and mission-critical projects.**
- **During the life cycle, software is produced early which facilitates customer evaluation and feedback.**

Iterative Disadvantages

- **More resources may be required.**
- **Although cost of change is lesser, but it is not very suitable for changing requirements.**
- **More management attention is required.**
- **System architecture or design issues may arise because not all requirements are gathered in the beginning of the entire life cycle.**
- **Defining increments may require definition of the complete system.**
- **Not suitable for smaller projects.**
- **Management complexity is more.**
- **End of project may not be known which is a risk.**
- **Highly skilled resources are required for risk analysis.**
- **Projects progress is highly dependent upon the risk analysis phase.**

Task 3

Working Agreement

Working Agreement

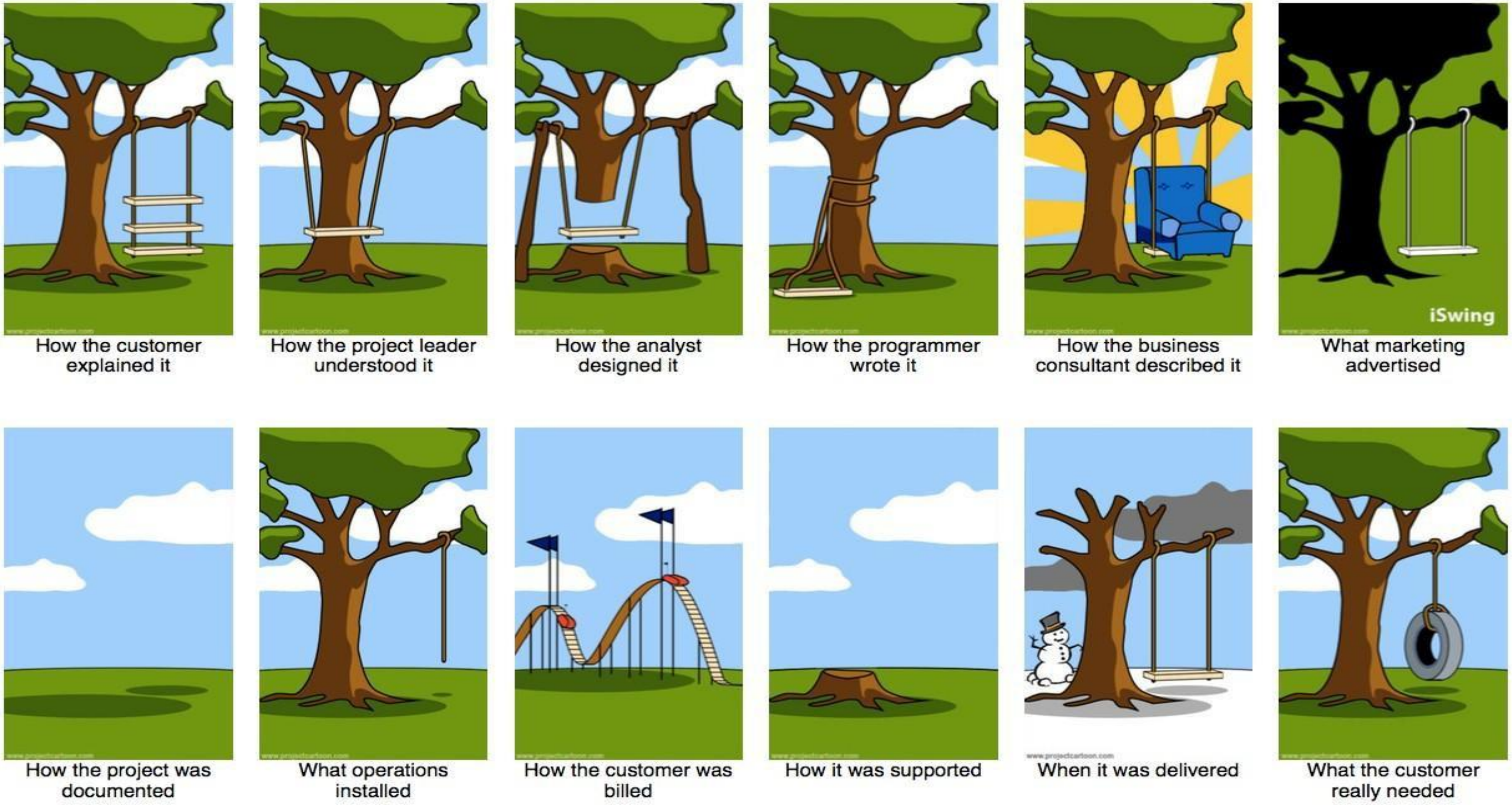
- What do we need to create the best working environment together?

- What is the value of creating a working agreement with the teams or individual you are working with?

Roman Voting

Agile: Modern Iterative Approach

Typical Project



How the customer explained it

How the project leader understood it

How the analyst designed it

How the programmer wrote it

How the business consultant described it

What marketing advertised

How the project was documented

What operations installed

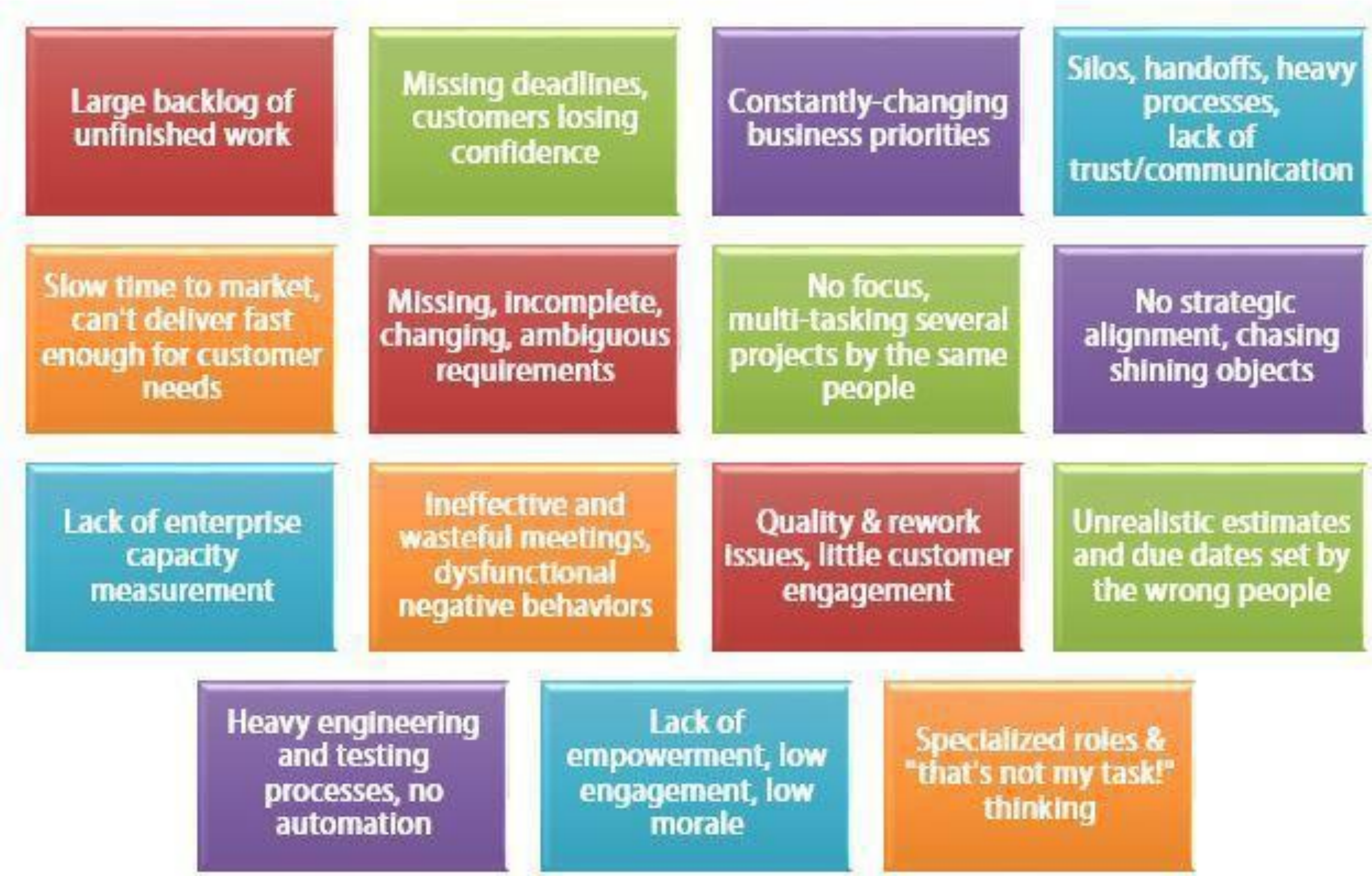
How the customer was billed

How it was supported

When it was delivered

What the customer really needed

Common Problem - Looks Familiar



Large backlog of unfinished work

Missing deadlines, customers losing confidence

Constantly-changing business priorities

Silos, handoffs, heavy processes, lack of trust/communication

Slow time to market, can't deliver fast enough for customer needs

Missing, incomplete, changing, ambiguous requirements

No focus, multi-tasking several projects by the same people

No strategic alignment, chasing shining objects

Lack of enterprise capacity measurement

Ineffective and wasteful meetings, dysfunctional negative behaviors

Quality & rework issues, little customer engagement

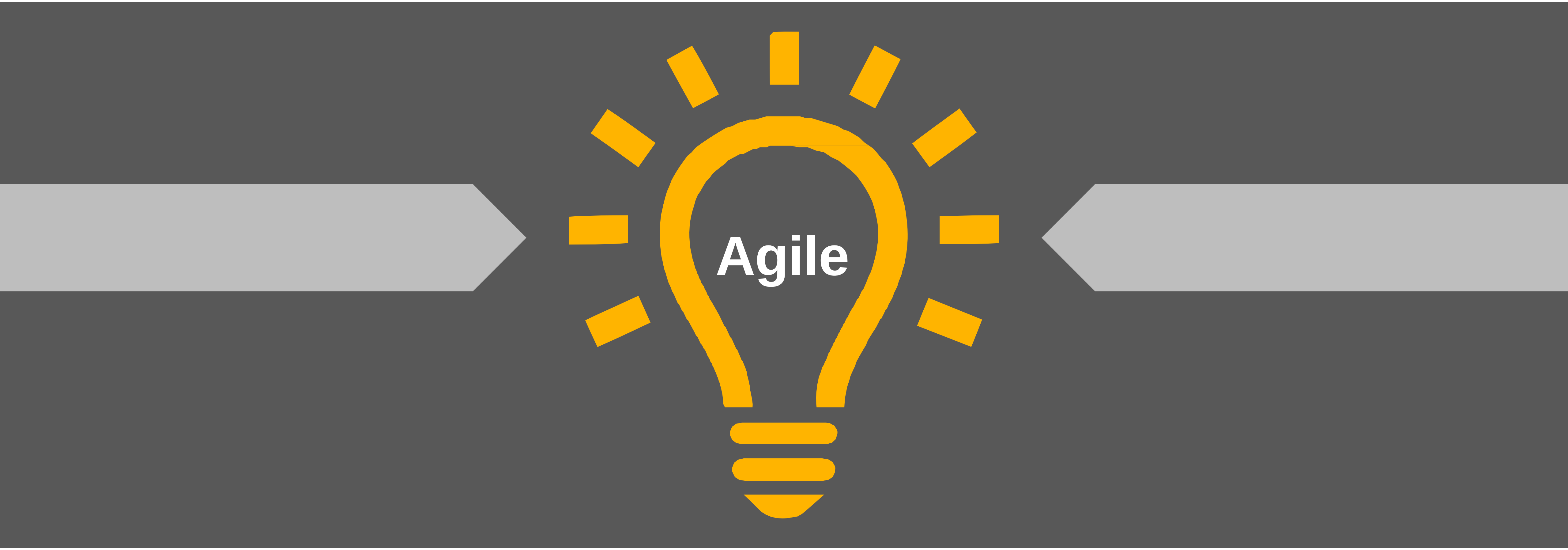
Unrealistic estimates and due dates set by the wrong people

Heavy engineering and testing processes, no automation

Lack of empowerment, low engagement, low morale

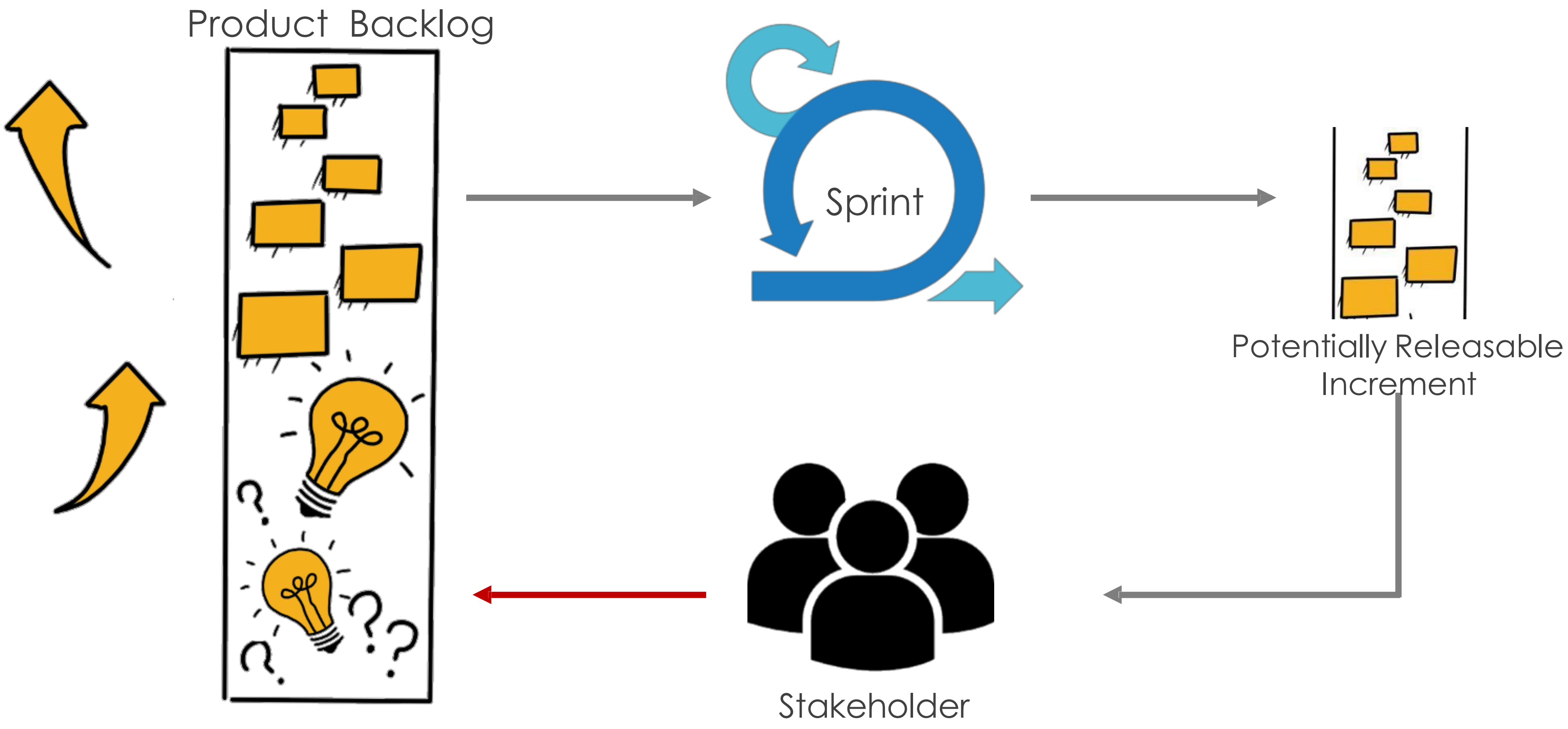
Specialized roles & "that's not my task!" thinking

Agile is a Mindset



Empowerment | Embrace Change | Transparency

Agile Way of Working



- Feedback loops now much shorter
- Customer engagement much more frequent and regular

What Agility Means

Speed to Market

Deliver Quality Product

Lower Cost

Reduce Complexity

Faster Decisions

Competitive Advantage

Agile Manifesto: Statement of Value

10:00

The twelve principles of agile include:

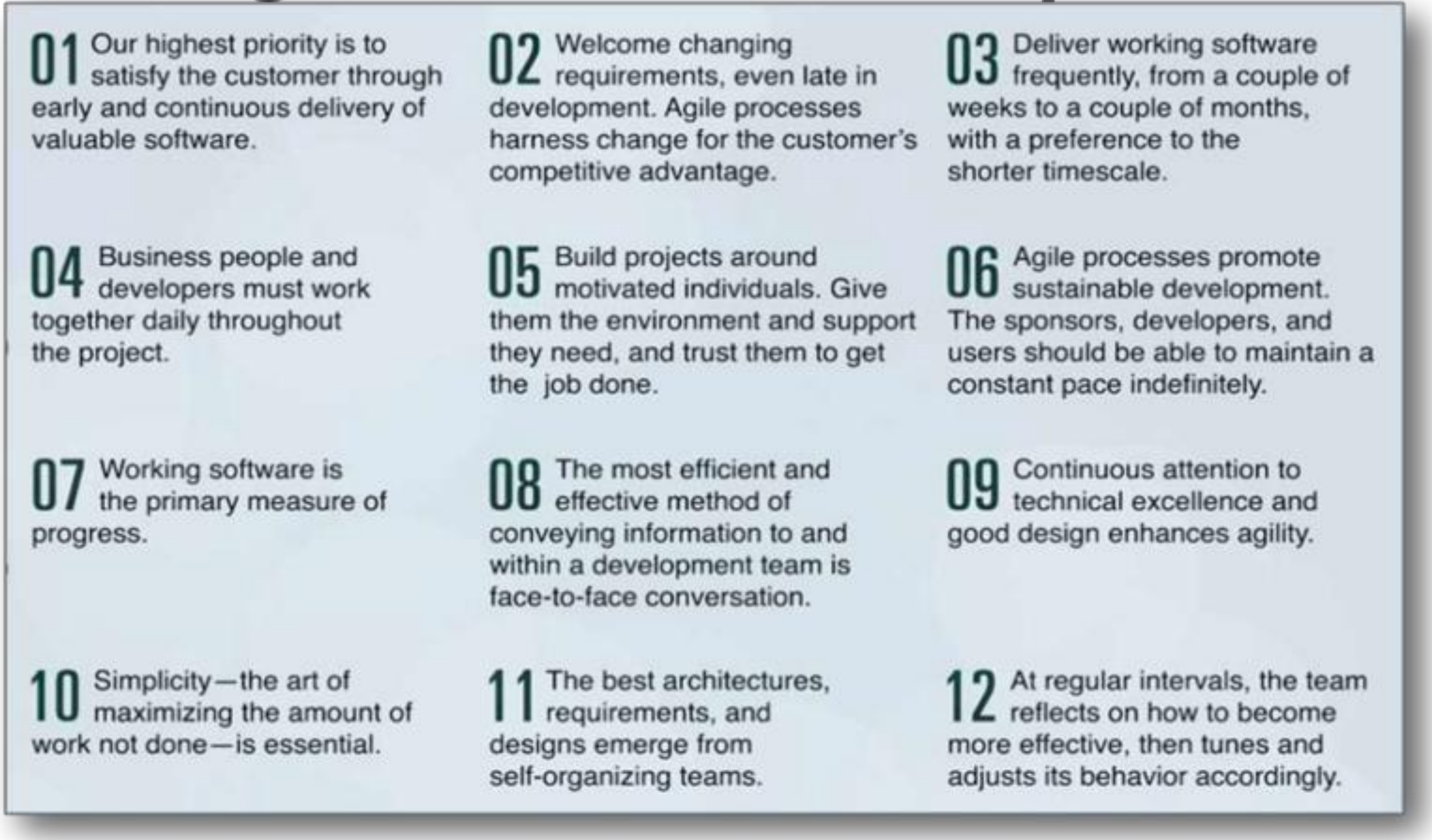
1.Customer satisfaction through early and continuous delivery
2.Accommodate changing requirements throughout the delivery process
3. Frequent delivery of working solution
4. Collaboration between the business stakeholders and delivery team throughout the project
5. Support, trust, and motivate the people involved.
6. Enable face-to-face interactions
7. Working product is the primary measure of progress
8. Agile processes to support a consistent delivery pace
9.Attention to technical detail and design enhances agility
10, Simplicity
11. Self-organizing teams encourage great architectures, requirements, and designs
12.Regular reflections on how to become more effective

Agile Manifesto & Values

Individuals & Interactions	Over	Processes & Tools
Working Software	Over	Comprehensive Documentation
Customer Collaboration	Over	Contract Negotiation
Responding to change	Over	Following a plan

Source: www.agilemanifesto.org

Agile Manifesto Principles



01 Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.

02 Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.

03 Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.

04 Business people and developers must work together daily throughout the project.

05 Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.

06 Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.

07 Working software is the primary measure of progress.

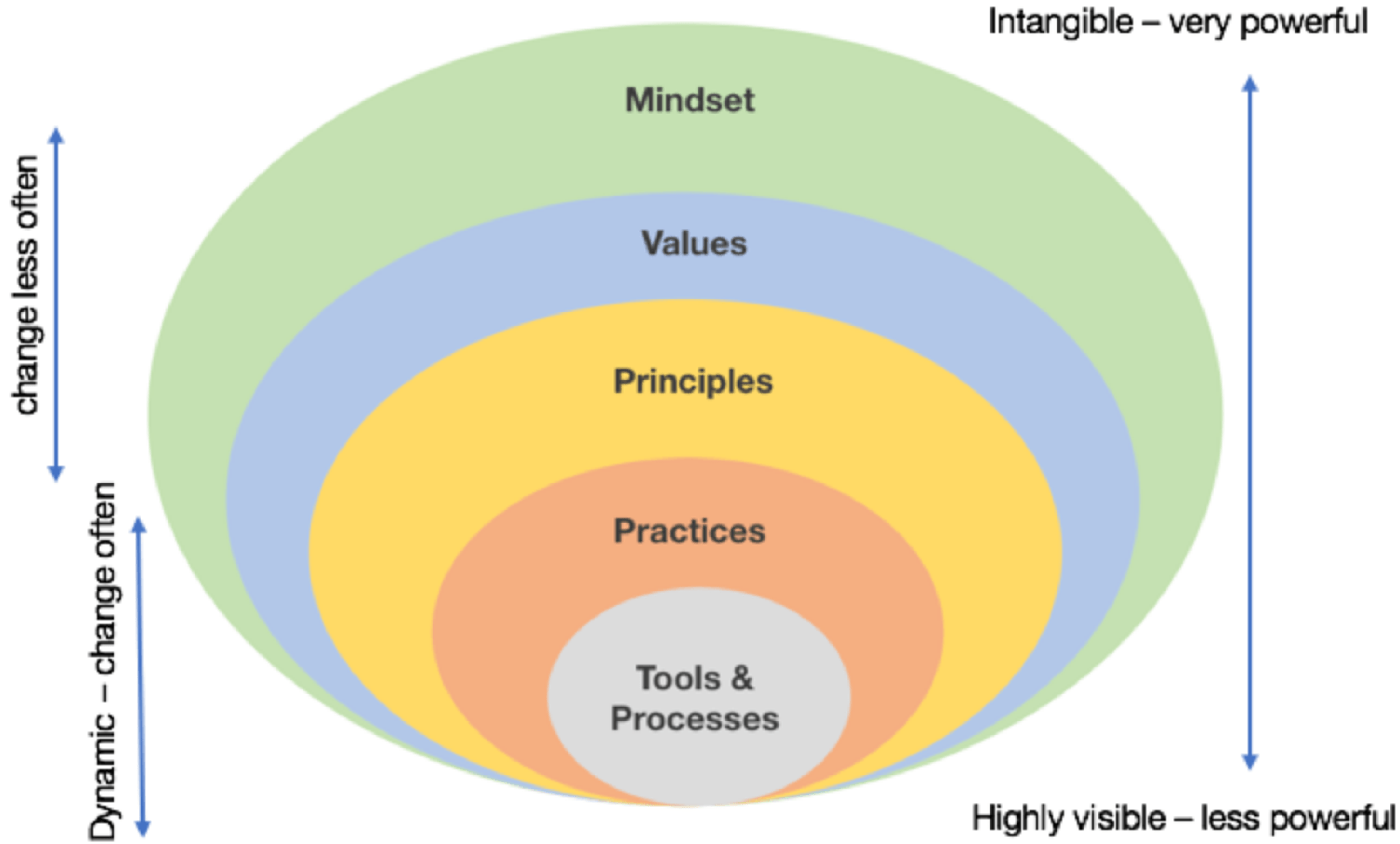
08 The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.

09 Continuous attention to technical excellence and good design enhances agility.

10 Simplicity—the art of maximizing the amount of work not done—is essential.

11 The best architectures, requirements, and designs emerge from self-organizing teams.

12 At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.



Intangible – very powerful

Mindset

Values

Principles

Practices

Tools & Processes

Highly visible – less powerful

change less often

Dynamic – change often

Diagram ©Adventures with Agile

Benefits of Agile

- **Continuous customer contact**
- **The ability to adapt**
- **Faster delivery**
- **Lower project risk**
- **Ongoing innovation**

When not to use Agile

- **The outcome of your project is stable and well understood**

- **Your project must produce a repeatable deliverable**

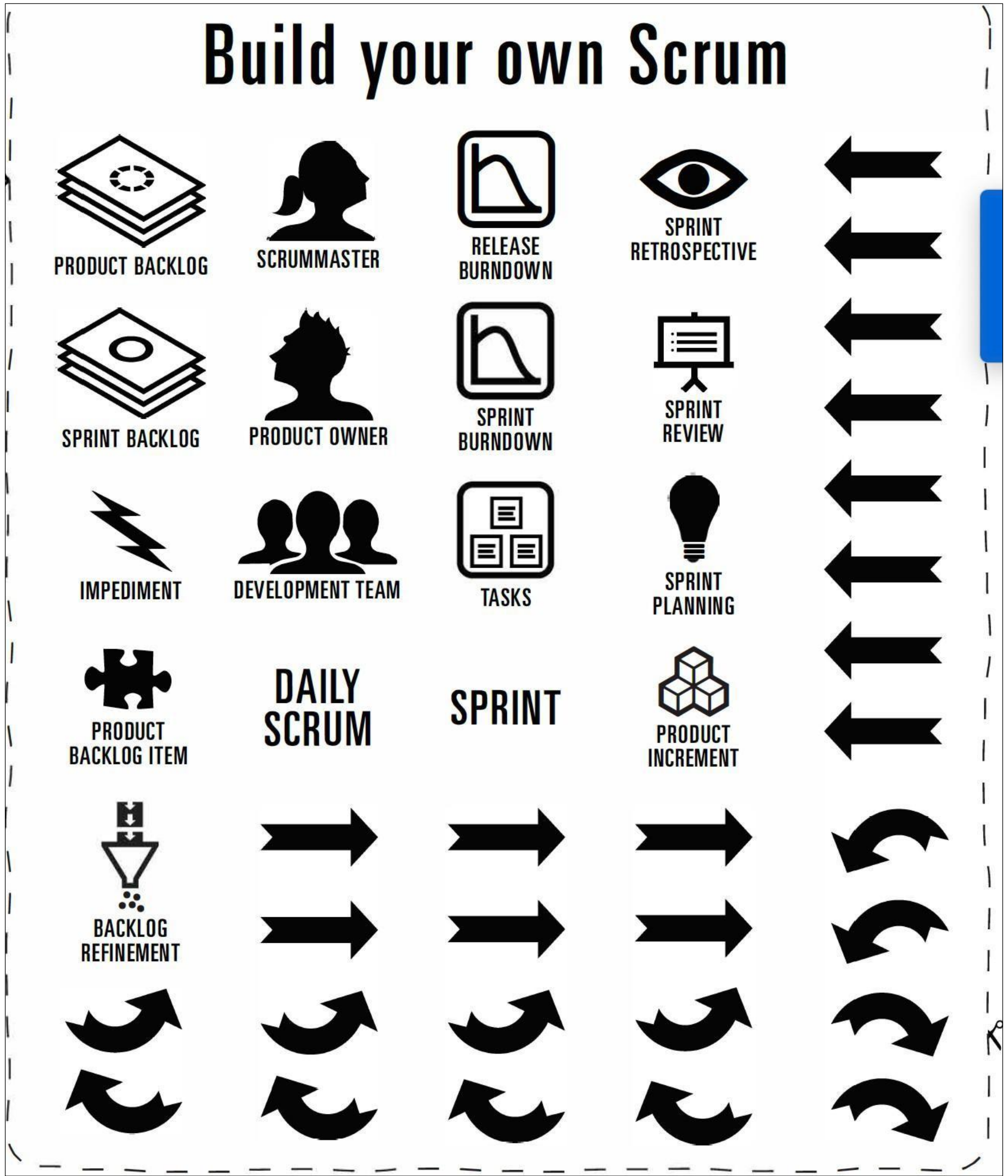
- **Your customer doesn’t want Agile**

- **Your company cannot support Agile.**

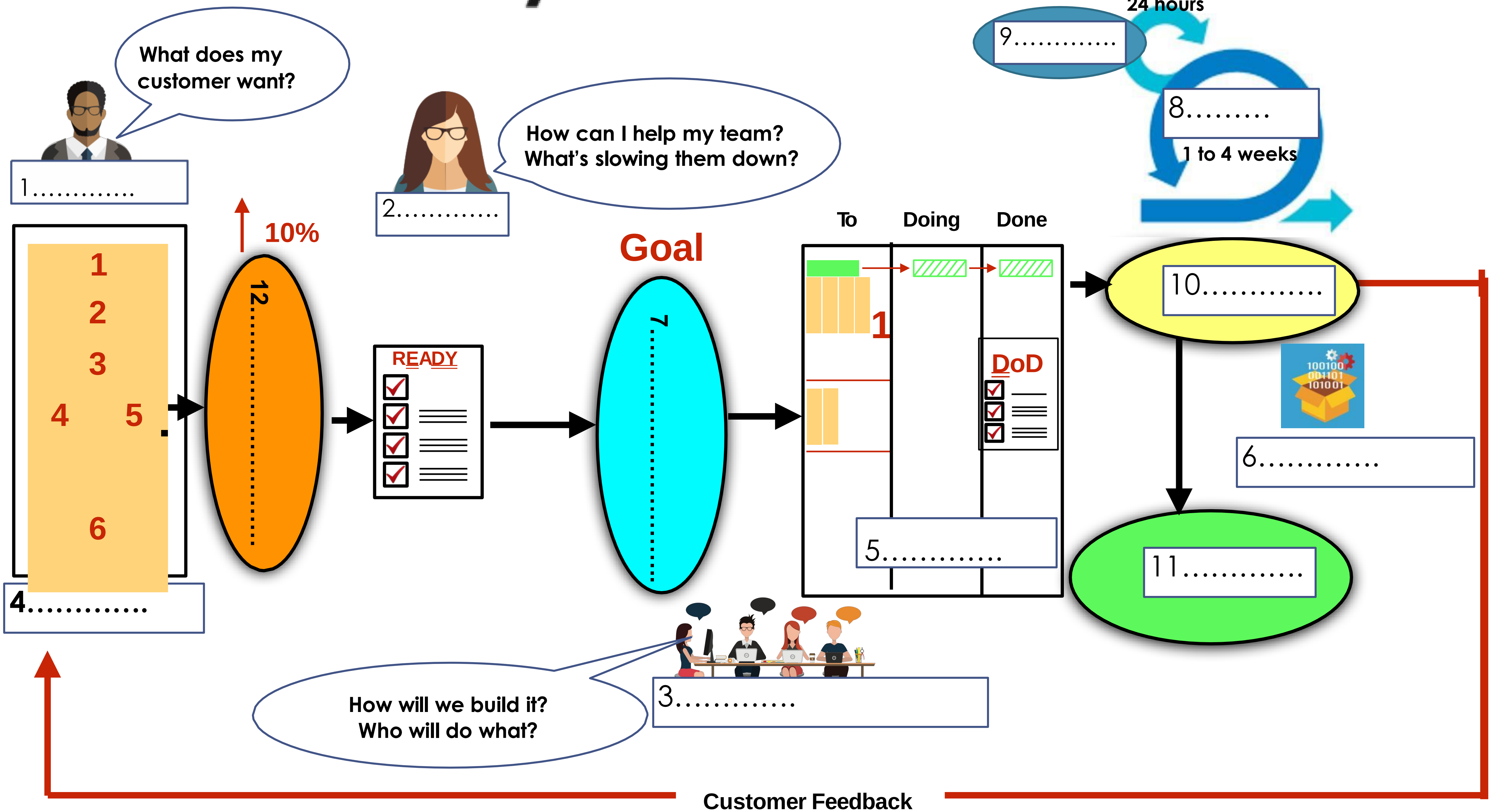
DAY 2

Scrum Framework

Build Your Own Scrum



Build your Scrum



What does my customer want?

1.............

How can I help my team? What’s slowing them down?

2...........

24 hours

9............

8........

1 to 4 weeks

4............

10............

6............

5............

11............

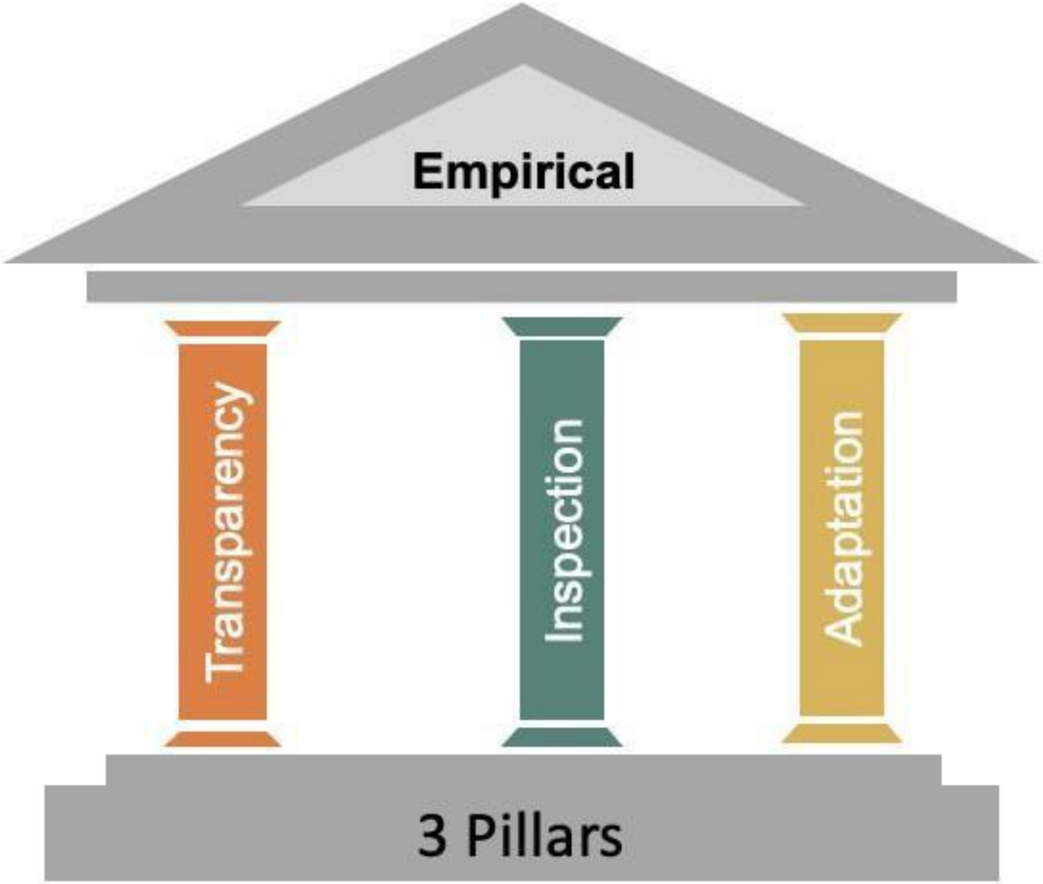
How will we build it? Who will do what?

3...........

Customer Feedback

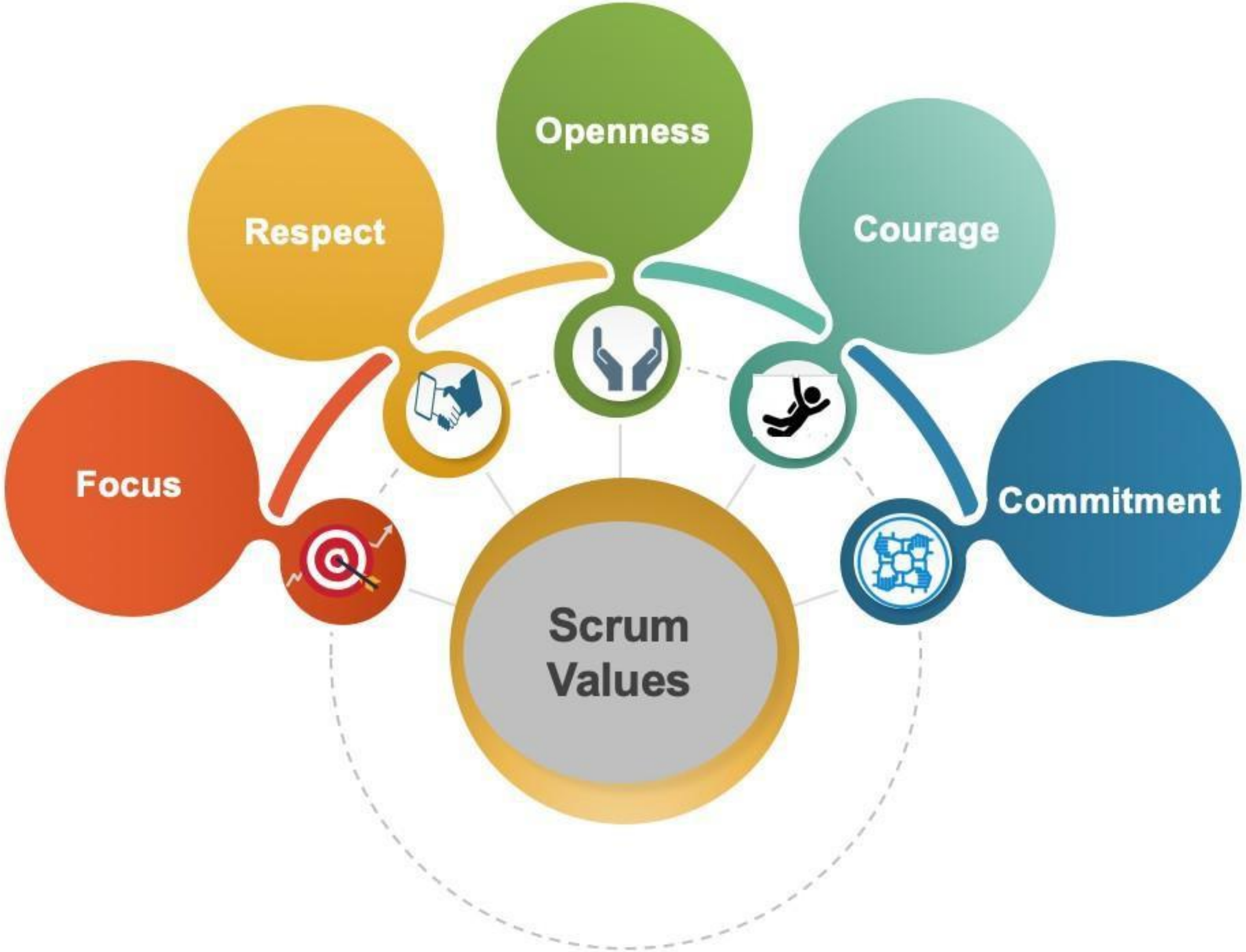
Empiricism is the foundation of Scrum

Empiricism: Solution evolved as knowledge is gained



Empiricism is a fundamental part of the Modern Scientific Process

Scrum VALUES

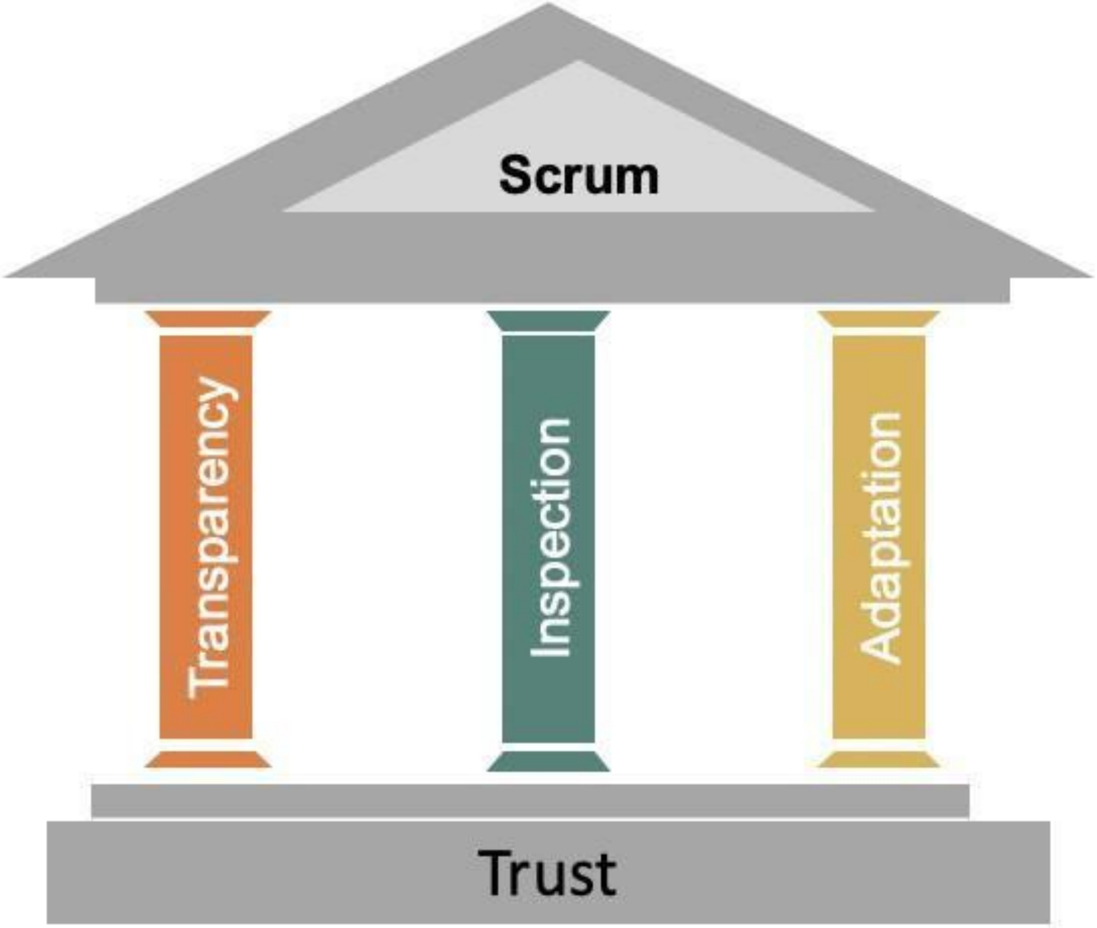


House of Scrum

OPENESS

RESPECT

FOCUS



Scrum

Transparency

Inspection

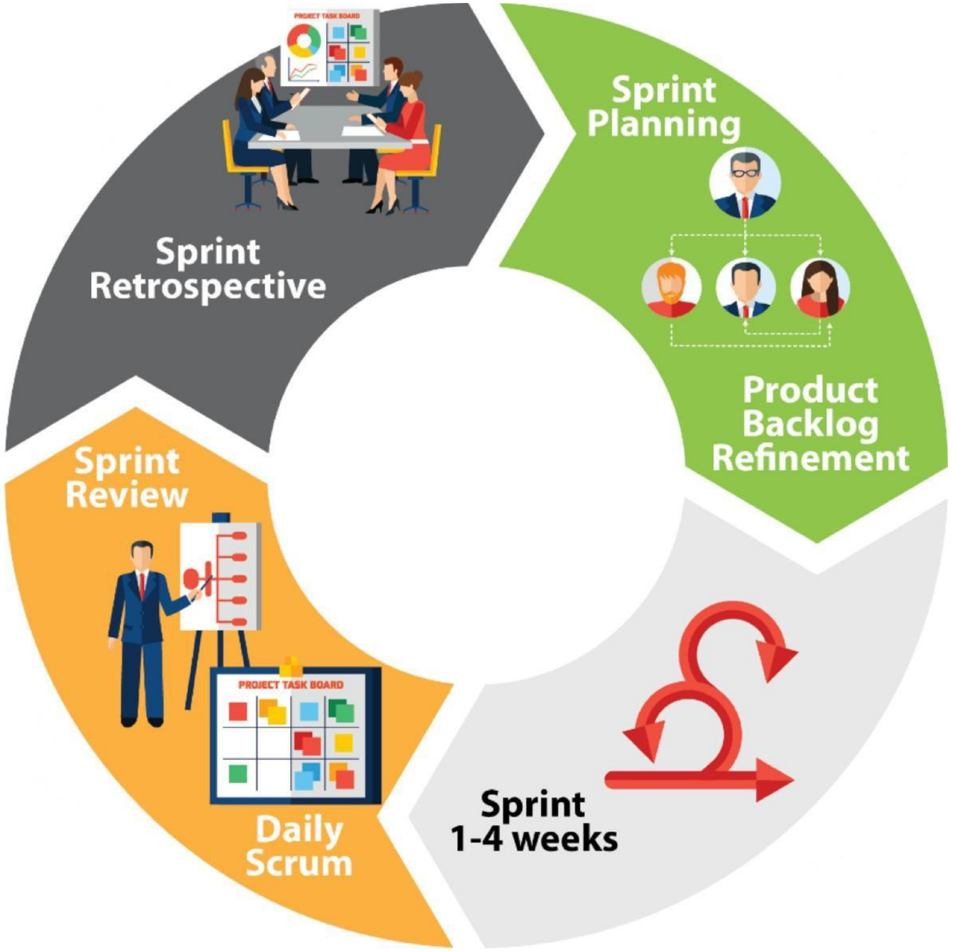
Adaptation

Trust

COURAGE

COMMITMENT

Scrum Events



Sprint Planning

Product Backlog Refinement

Sprint 1-4 weeks

Daily Scrum

Sprint Review

Sprint Retrospective

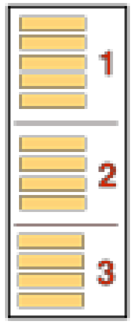
Scrum Roles

Scrum Master

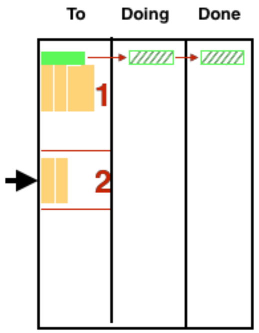
Development Team

Product Owner

Scrum Artifacts



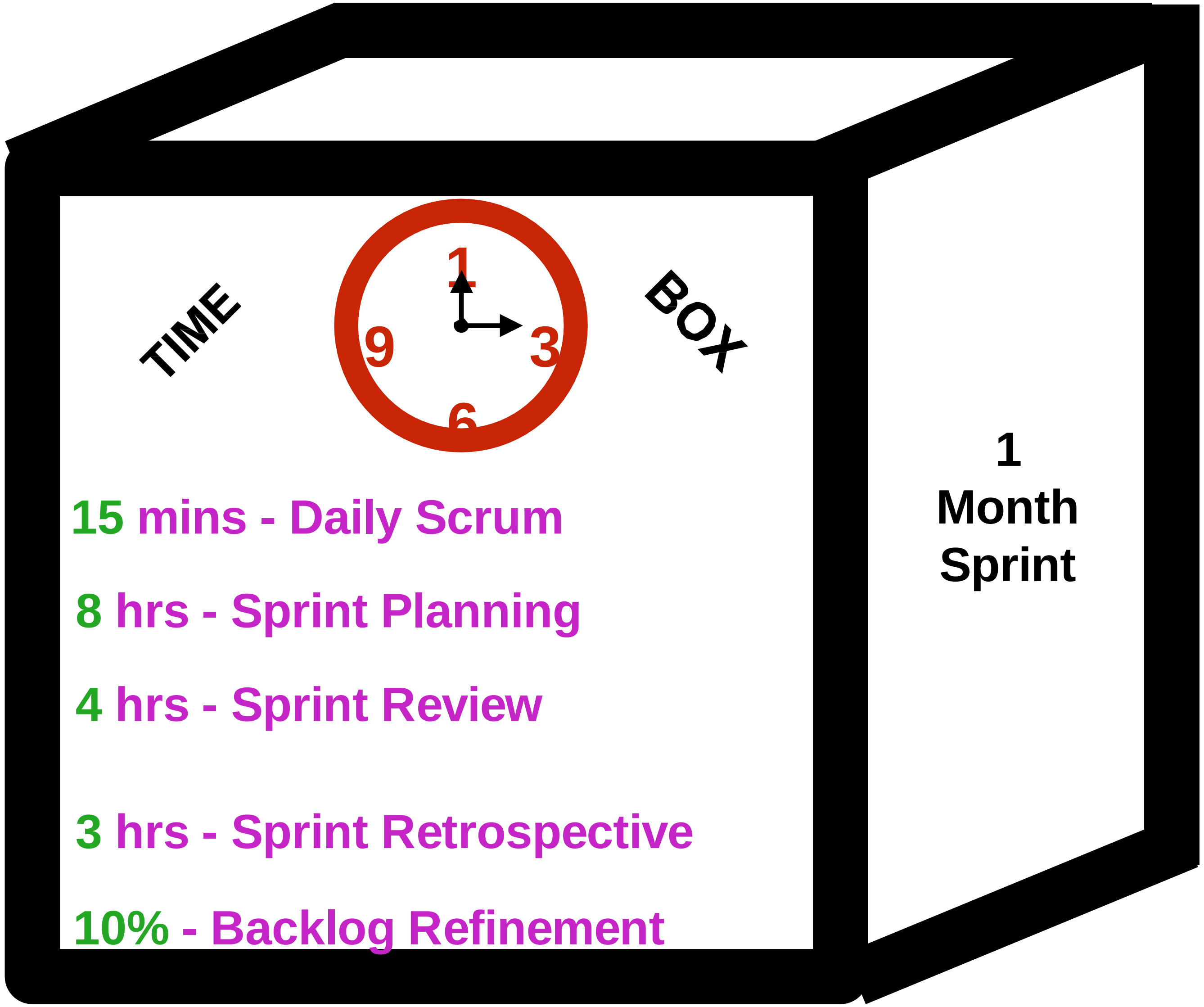
Product Backlog



Sprint Backlog

Increment

The Scrum Time Boxes



15 mins - Daily Scrum

8 hrs - Sprint Planning

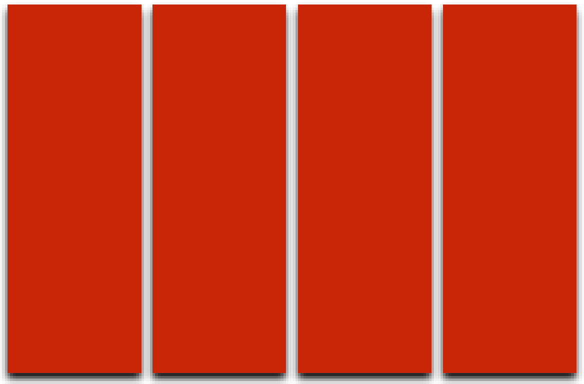
4 hrs - Sprint Review

3 hrs - Sprint Retrospective

10% - Backlog Refinement

1
Month
Sprint

Shorter Sprints? Proportionate for Planning, Review and Retro

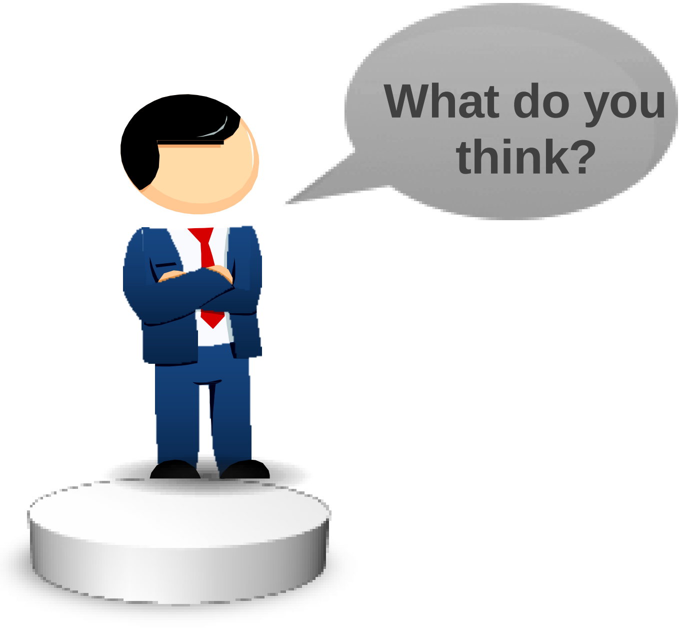


Planning 2hrs/wk/sp

Review 1hr/wk/sp

Retro 45min/wk/sp

Scrum and Agile



Your Answers

Scrum Roles

The Scrum Team

The "Guard"
The "Guide"

Scrum Master
(SM)

The "How"
The "How long"

Development Team

The "What"
The "Why"

Product Owner
(PO)

The Product Owner

The Product Owner owns the WHAT

Prerogatives

- Sole authority over the product backlog
- Decides what to build
- Decides when to release
- Make Product-related decisions

Responsibilities

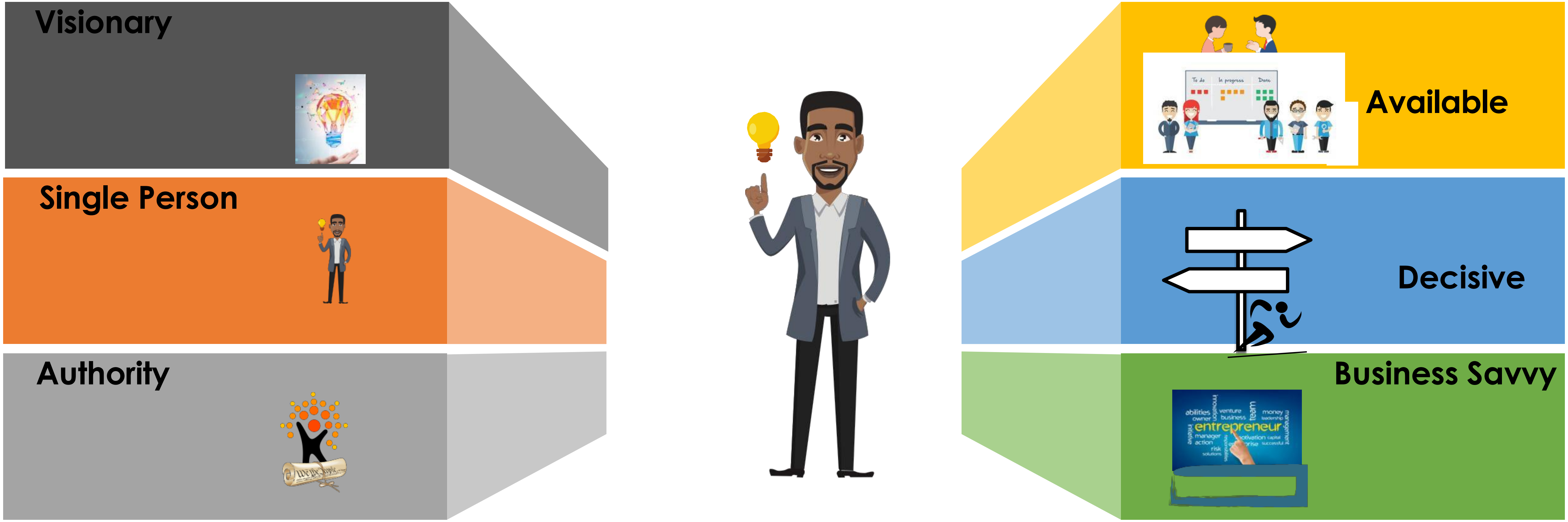
- Ongoing communication of a compelling and executable product vision
- Ordering the product backlog by value
- Clearly expressing PBIs to the development team
- Decides when to release
- Budget and release management

Discuss..

Key things you need to know about PO

- The PO is the single voice of the stakeholders to the scrum team – a one man band not a committee
- Has a compelling product vision and an executable roadmap for achieving the vision
- Owns and manages the backlog – and order them by value
- His Primary focus is product backlog management - happens every day in every sprint. SM coaches the PO on effective techniques for managing the product backlog
- During the sprint - he is available for the stakeholder - and the scrum team
- Communicate with the team what to do next
- Manages the budget and releasing it is the responsibility of the PO - NOT the SM

The Product Owner Attributes



Visionary

Single Person

Authority

Available

Decisive

Business Savvy

The Development/ Delivery Team

The Development Team owns the **HOW**

Prerogatives

- Produce quality product
- Provides their own estimates
- Manages their work
- Decides how to turn PBIs into Potential Releasable Product Increments
- Assign work to themselves

Responsibilities

- Collaborate to achieve the sprint goal(s)
- Delivers quality product increment at the end of each sprint
- Organize and manages their own work
- Do the work to deliver a potential releasable product increment
- Assist the Product Owner in refining the Product Backlog

Development Team Attributes

Cross-functional

Self Organizing/
Management

Mutually Accountable

Team size (3-9)

Generalized Specialist

Focussed

The Scrum Master

The Scrum Master Owns the PROCESS

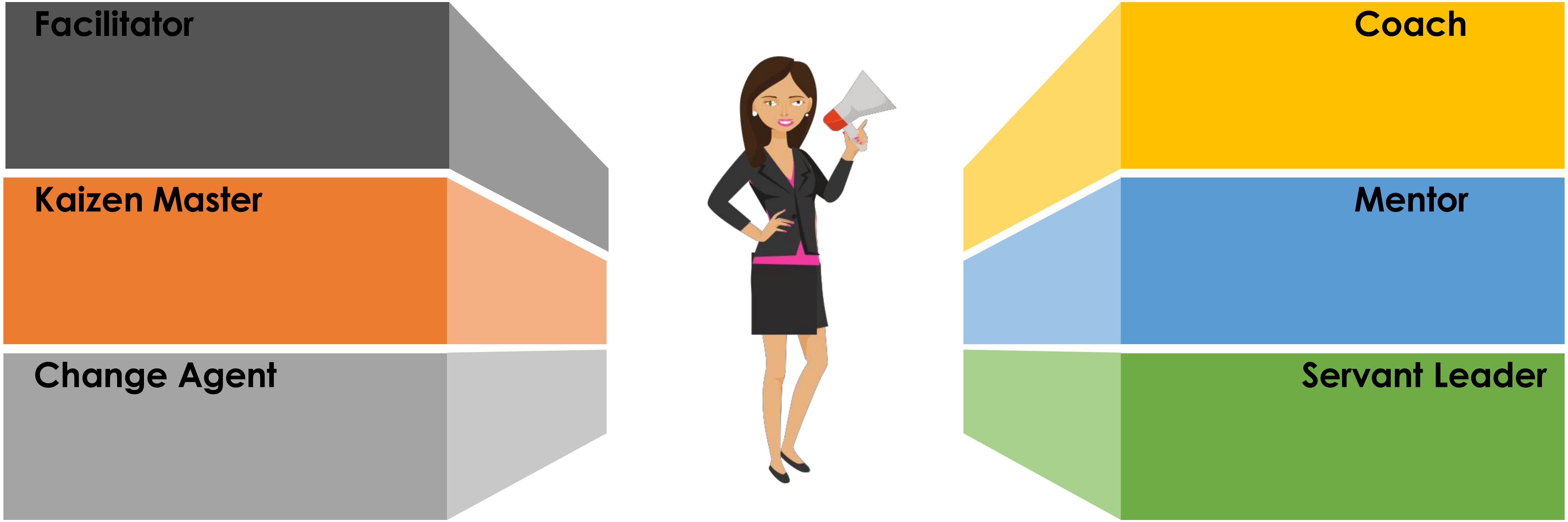
Prerogatives

- Own the scrum process
- Experiment with new ideas
- Address issues openly and in a safe manner

Responsibilities

- Coaches the Team and Product Owner in Scrum techniques
- Understands and implements the values of the Agile Manifesto
- Facilitates Scrum Events and team decisions
- Foster the culture of continuous improvement and waste reduction
- Identifies and ensures impediments are resolved.
- Allowing the team to figure out their own solutions

The Scrum Master Attributes



Facilitator

Kaizen Master

Change Agent

Coach

Mentor

Servant Leader

The Scrum Master is not.....

Pipe

Scrum Mom

Enabler

Secretary

Dictator

Fixer

T-Shape Team

Broad
can turn their hand to many things

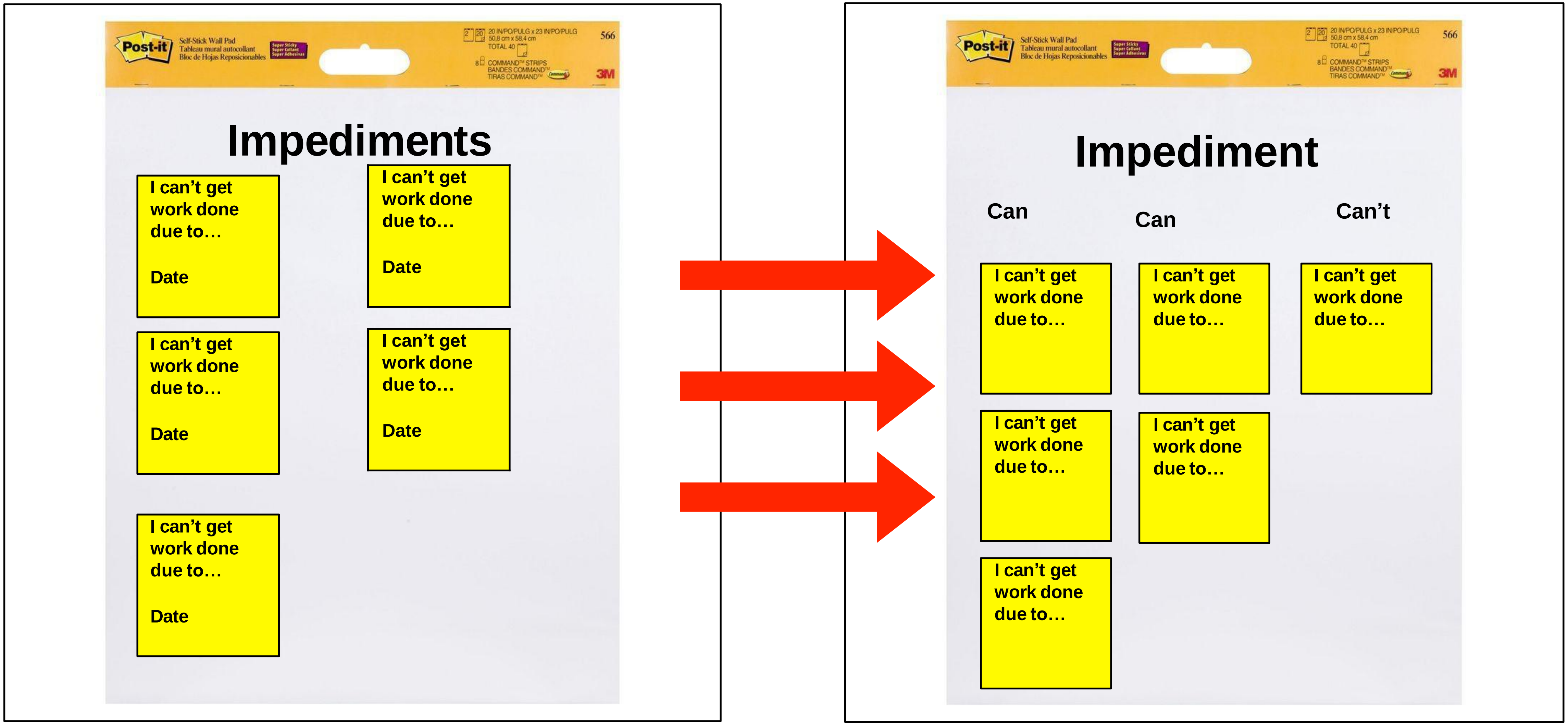
Deep specialization in one area

Multi - Specialization

	Charli	Bob	Alice
Java coding	1	3	4
Writing selenium tests	3	3	2
Web page design	2	4	2
....			
Skills map:	1 – Would like to learn 2 – Can do this with help 3 – Can do this unaided 4 – Can teach others		

Impediment

Impediment Backlog



Impediments

I can't get work done due to…

Date

I can't get work done due to…

Date

I can't get work done due to…

Date

I can't get work done due to…

Date

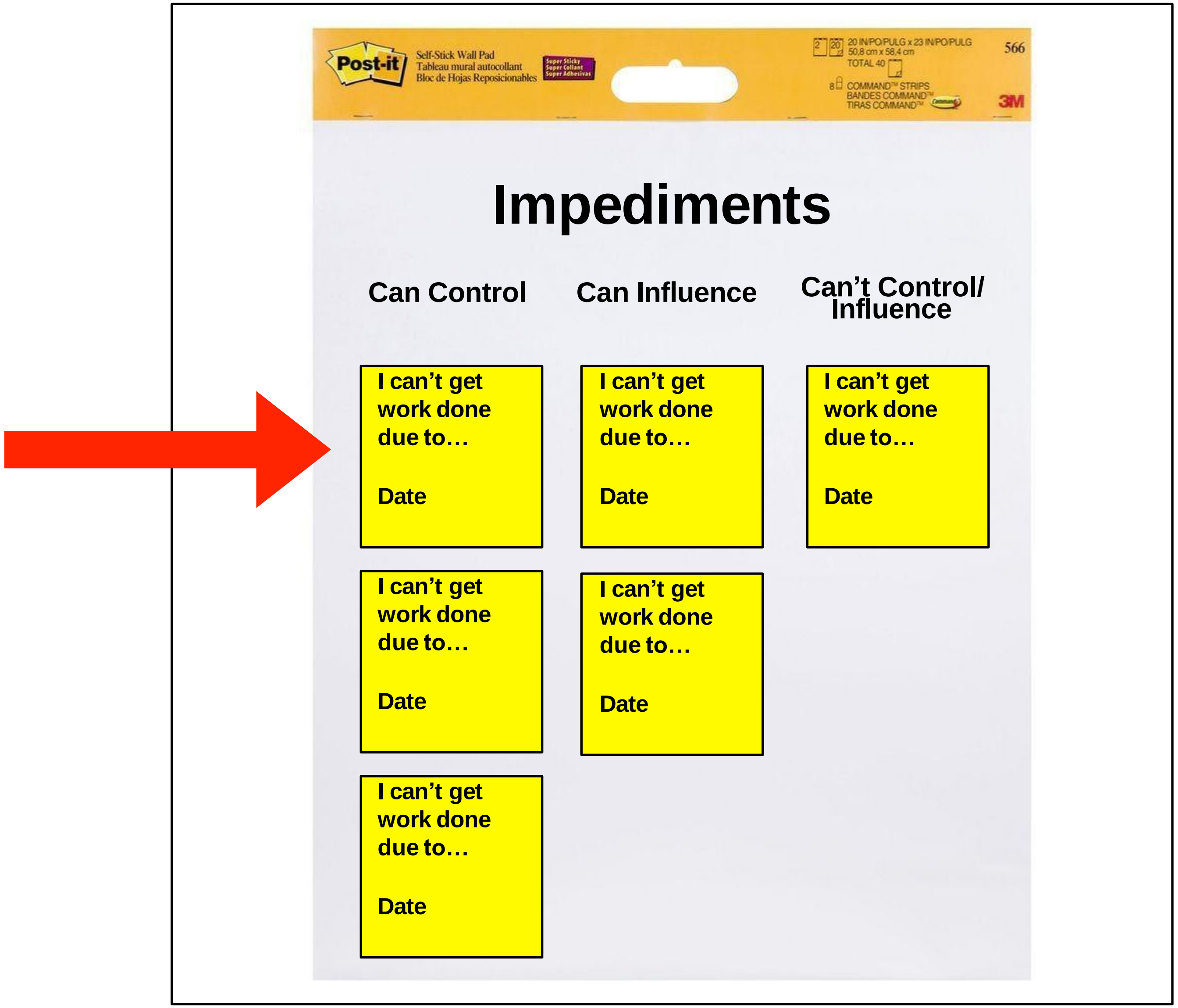
I can't get work done due to…

Date

Impediment

Can	Can	Can't
I can't get work done due to…	I can't get work done due to…	I can't get work done due to…
I can't get work done due to…	I can't get work done due to…	
I can't get work done due to…		

Share Plan to Resolve 1 Impediment

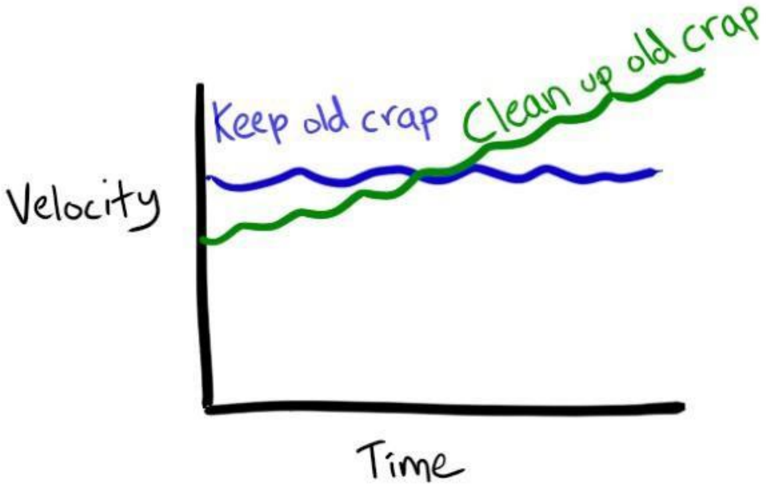


Impediments

Can Control	Can Influence	Can't Control/ Influence
I can't get work done due to… Date	I can't get work done due to… Date	I can't get work done due to… Date
I can't get work done due to… Date	I can't get work done due to… Date	
I can't get work done due to… Date		

Hmm.. Impediment?

Technical Debt



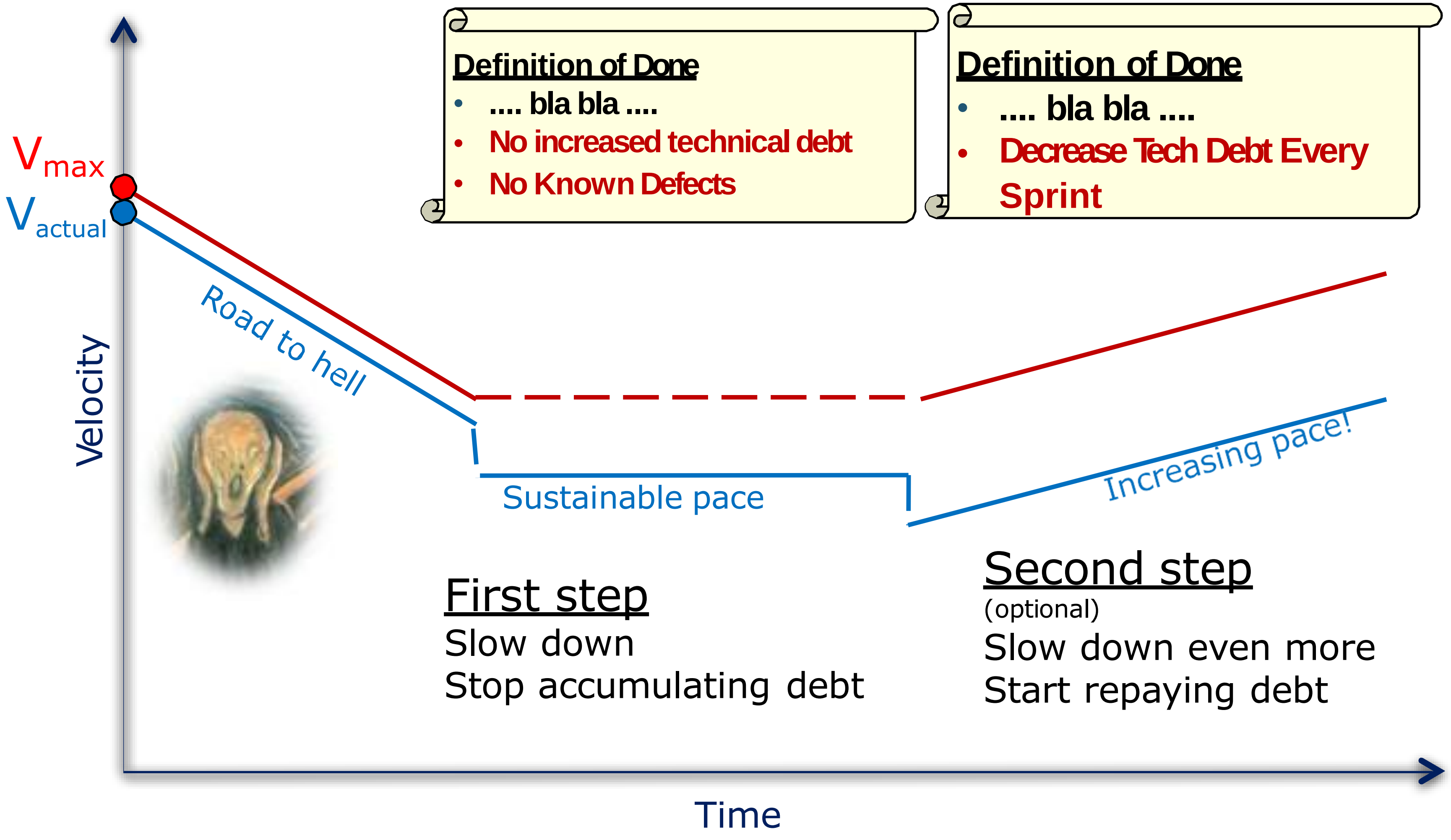
What is Technical Debt?

The deferred, necessary work not done that has impacted present work or will impact future work

Technical debt quadrant

http://martinfowler.com/bliki/TechnicalDebtQuadrant.html

Dealing with Technical Debt



Definition of Done

- .... bla bla ....
- No increased technical debt
- No Known Defects

Definition of Done

- .... bla bla ....
- Decrease Tech Debt Every Sprint

First step

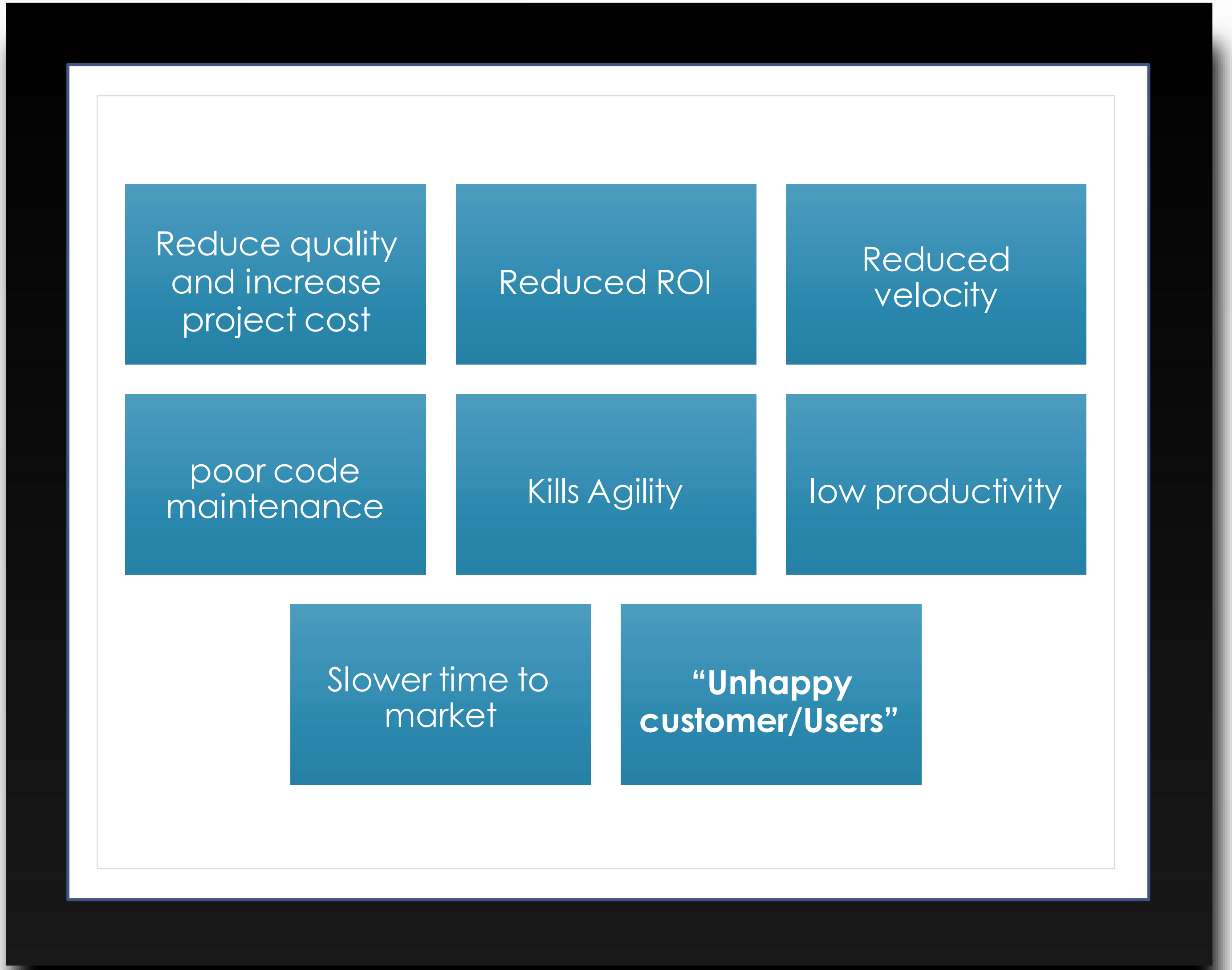
Slow down
Stop accumulating debt

Second step
(optional)

Slow down even more
Start repaying debt

Source: Henrik Kniberg

Impacts of Accumulating Technical Debt



Reduce quality and increase project cost

Reduced ROI

Reduced velocity

poor code maintenance

Kills Agility

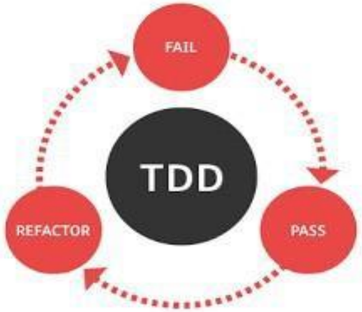
low productivity

Slower time to market

“Unhappy customer/Users”

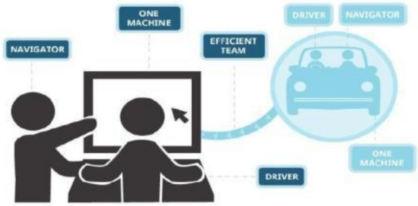
Reducing Technical Debts

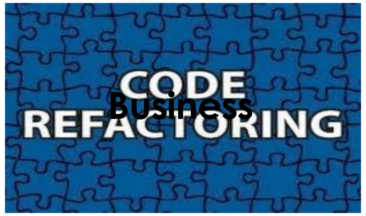
Product Backlog

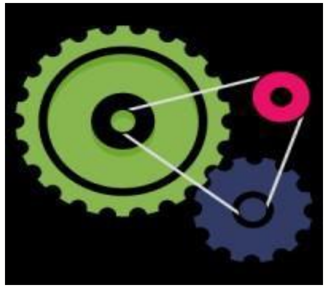


EMERGENT
ARCHITECTURE

PAIR PROGRAMMING







REQUIREMENTS

Product Vision

Understanding Vision

Goals

- What is our ultimate goal(s)?
- How can we measure progress towards them?

Markets

- What do we believe the market wants or needs?
- How can we test these beliefs?

Competitive Position

- What are our strengths and weaknesses relative to other competitors?
- How can we test these beliefs?

Guiding Principles

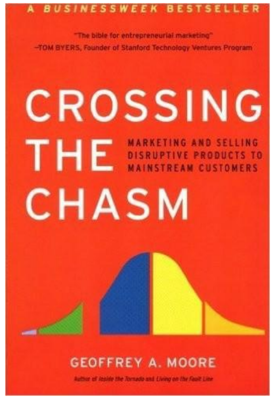
- What does this imply about what we should or should not do?

Vision Drives Backlog

- Backlogs should represent work to achieve the company's vision.
- Leadership must set clear vision to focus on delivering the right value.
- Product Owners should know how backlog aligns to the vision.

Example: Creating a Vision Statement

- FOR <target customers>
- WHO <statement of need>
- THE <product name> IS A <product category>
- THAT <key benefit, compelling reason to buy and use>
- UNLIKE <competition/alternative>
- OUR PRODUCT <differentiating statement>


A BUSINESSWEEK BESTSELLER
"The bible for entrepreneurial marketing"
–TOM BYERS, Founder of Stanford Technology Ventures Program
CROSSING THE CHASM
MARKETING AND SELLING DISRUPTIVE PRODUCTS TO MAINSTREAM CUSTOMERS
GEOFFREY A. MOORE
Author of Inside the Tornado and Living on the Fault Line

Asuccessful vision statement is compelling enough to be broadly shared, yet concise and easily remembered

Source: Crossing the Chasm by Geoffrey Moore

Customer Personas

Customer Personas Make Data Real

Personas are archetypes…not real people

- Describes the “centroid” of a customer segment

Provides context for a user and what he/ she wishes to accomplish

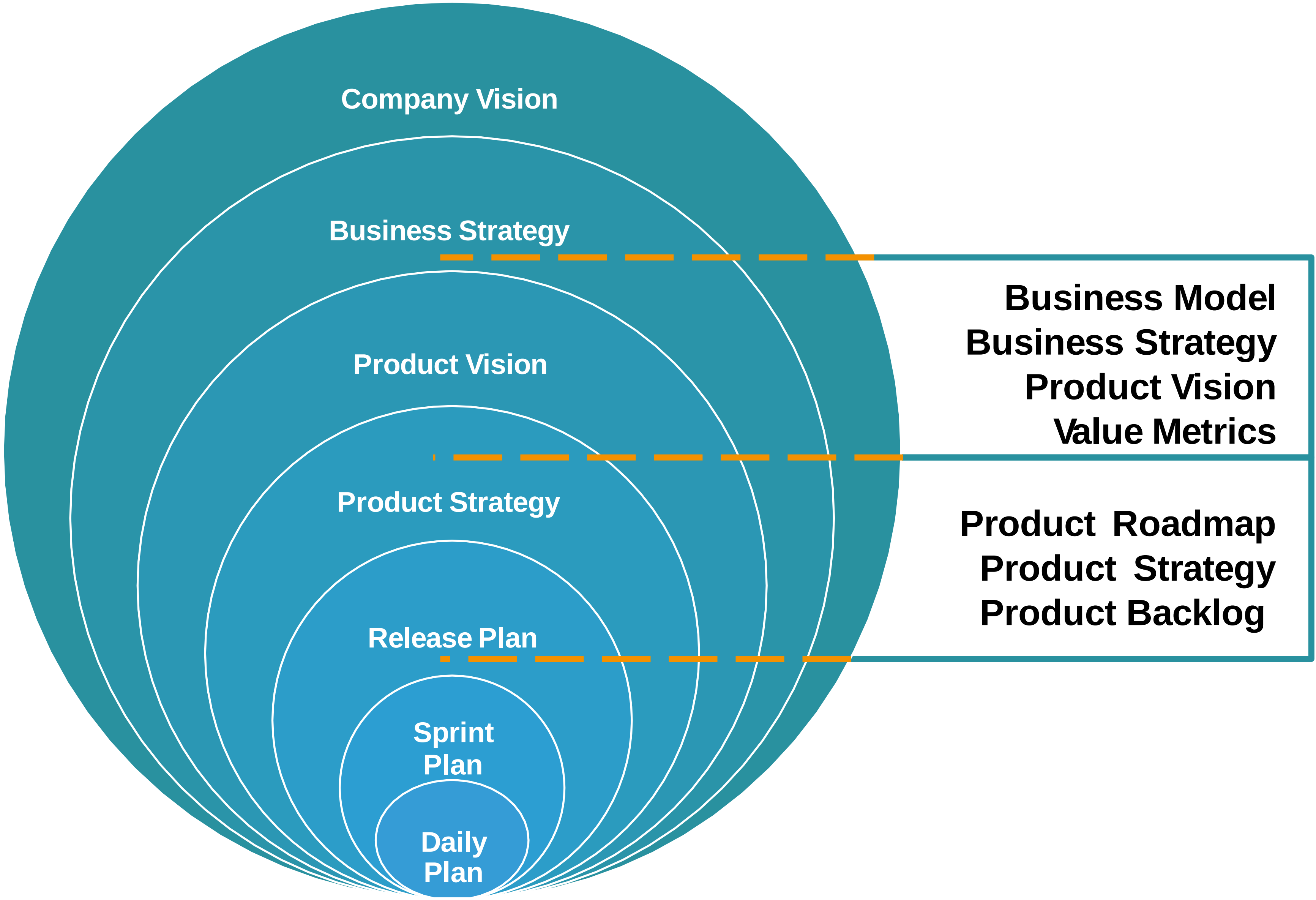
- Focuses the Team on the desired outcomes
- Personas often end feature debates

Need a persona for each targeted customer segment and/or product

Product Backlog

Product Backlog
(Features)

Layers of Product Delivery



Company Vision

Business Strategy

Product Vision

Product Strategy

Release Plan

Sprint Plan

Daily Plan

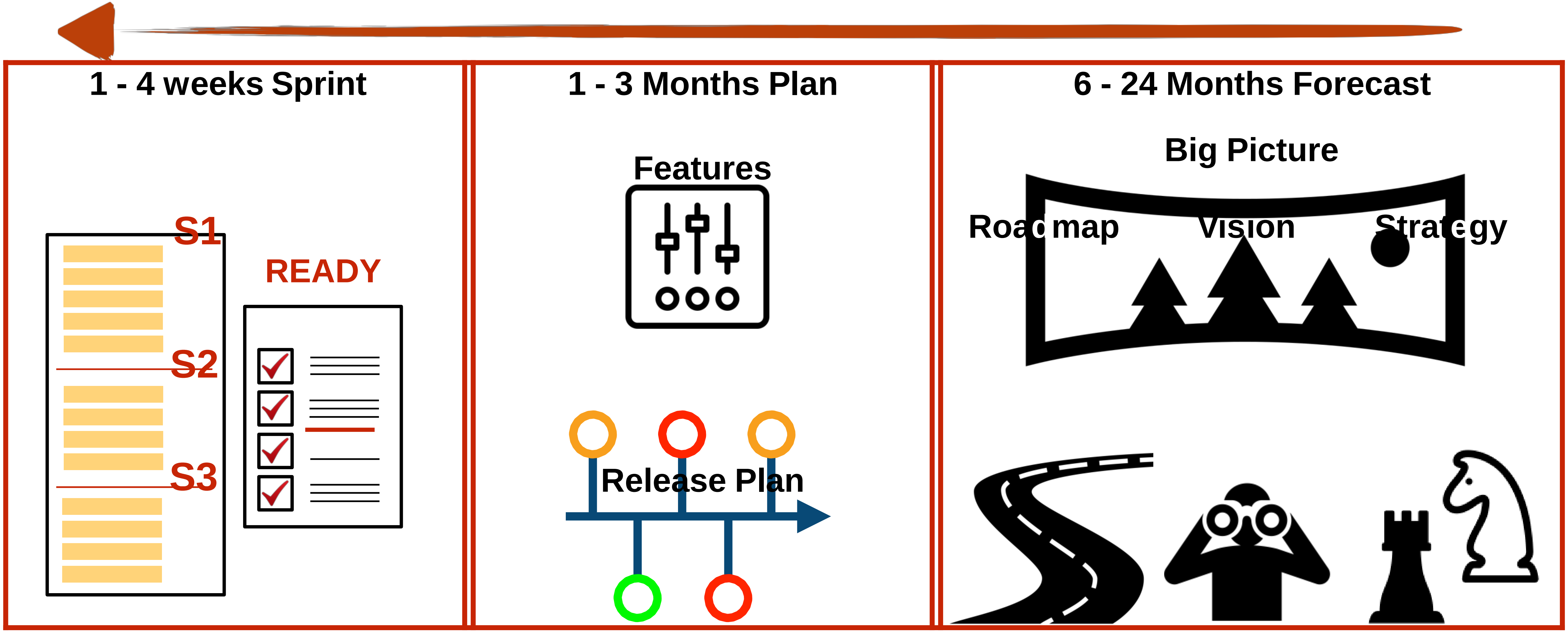
Business Model
Business Strategy
Product Vision
Value Metrics

Product Roadmap
Product Strategy
Product Backlog

Product Owner

c

Product Planning Horizon



1 - 4 weeks Sprint

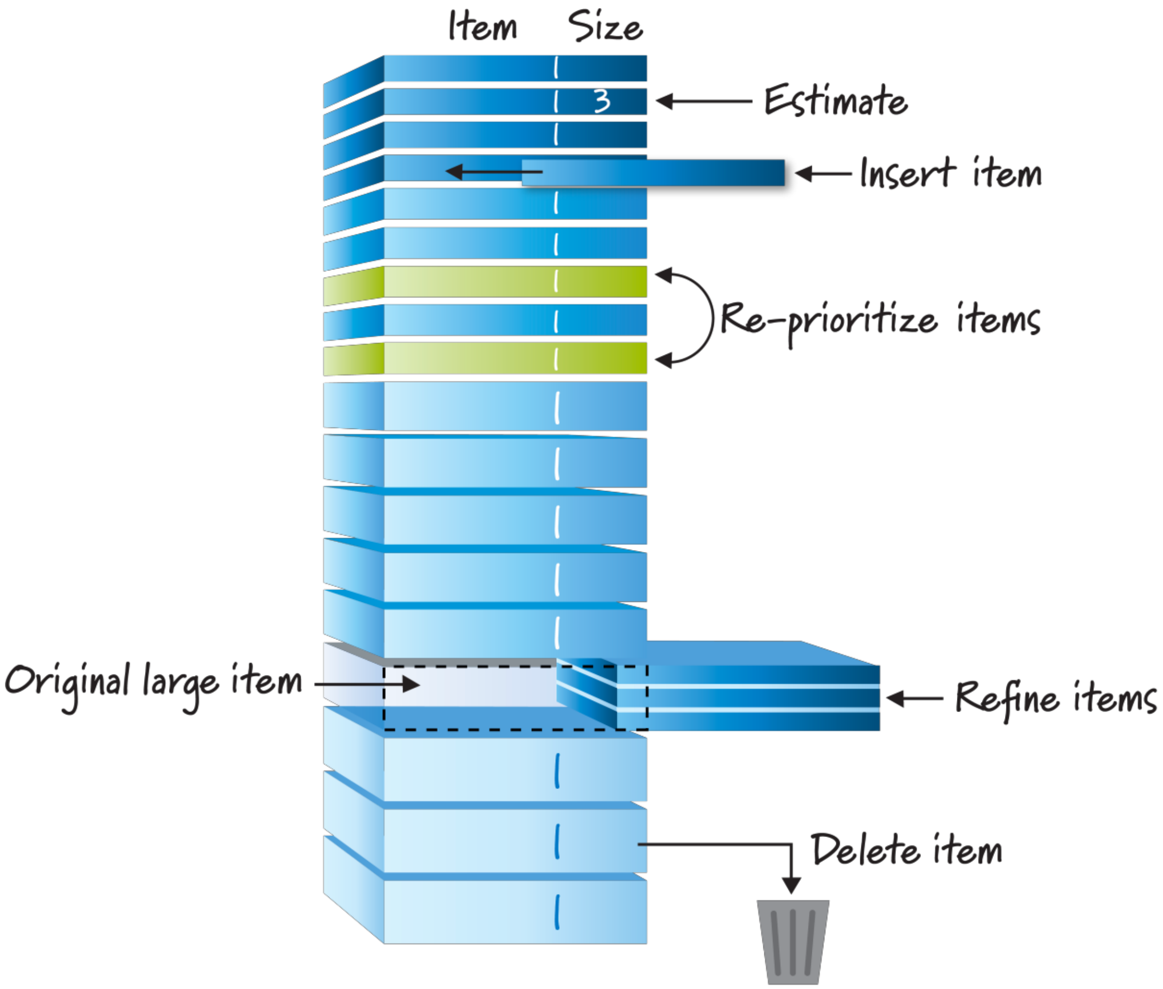
1 - 3 Months Plan

6 - 24 Months Forecast

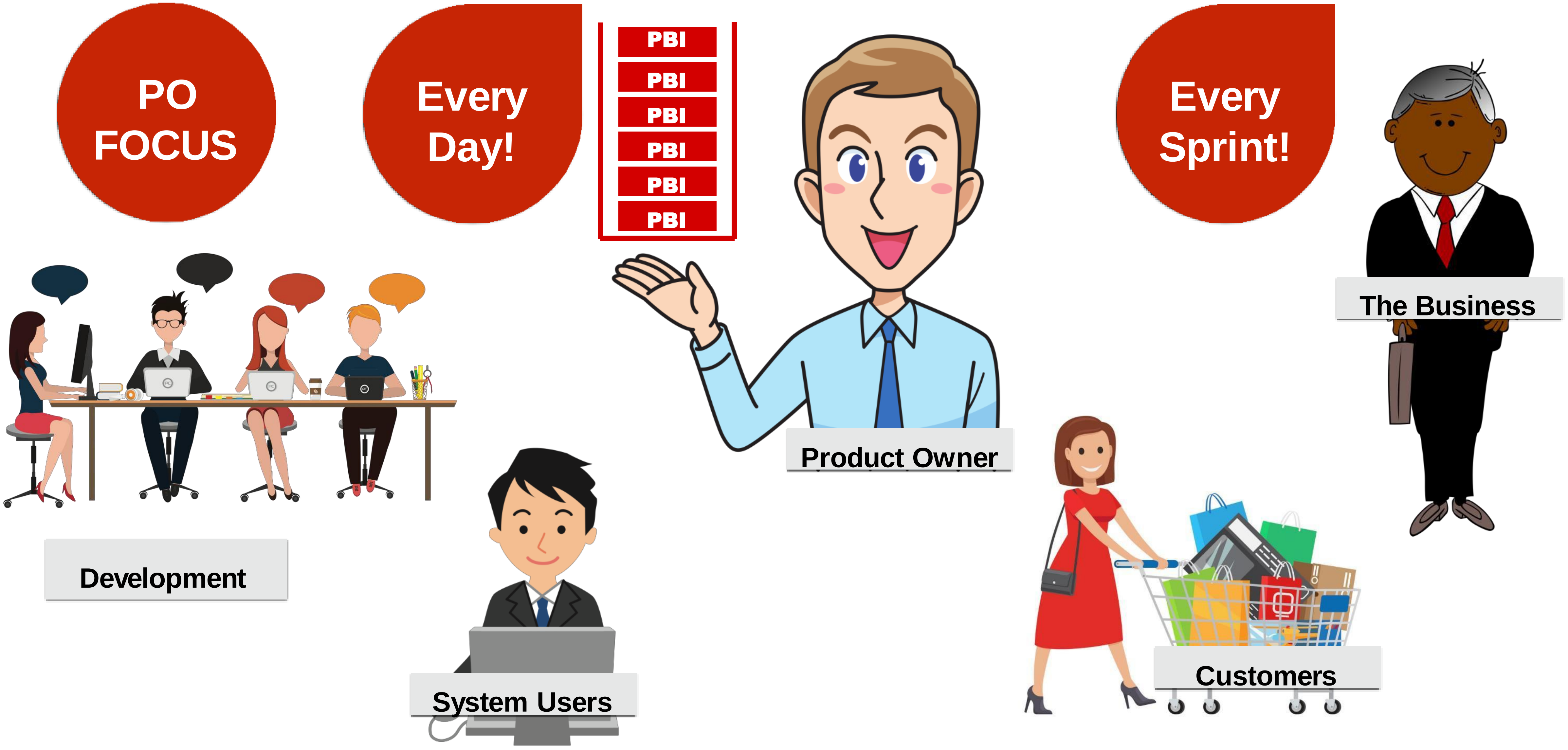
Based on the work of Ellen Gottesdiener www.ebgconsulting.com

The Product Backlog

- Consists of **work to be done** ordered by customer value.
- Only one Product Backlog which is shared across teams working on the same product.
- In Scrum, the Product Owner is the final authority on ordering the backlog.
- The Backlog consists of Product Backlog Items (PBIs).
- Most Scrum teams use User Stories as PBIs.



Product Backlog Development



PO FOCUS

Every Day!

PBI
PBI
PBI
PBI
PBI
PBI

Every Sprint!

The Business

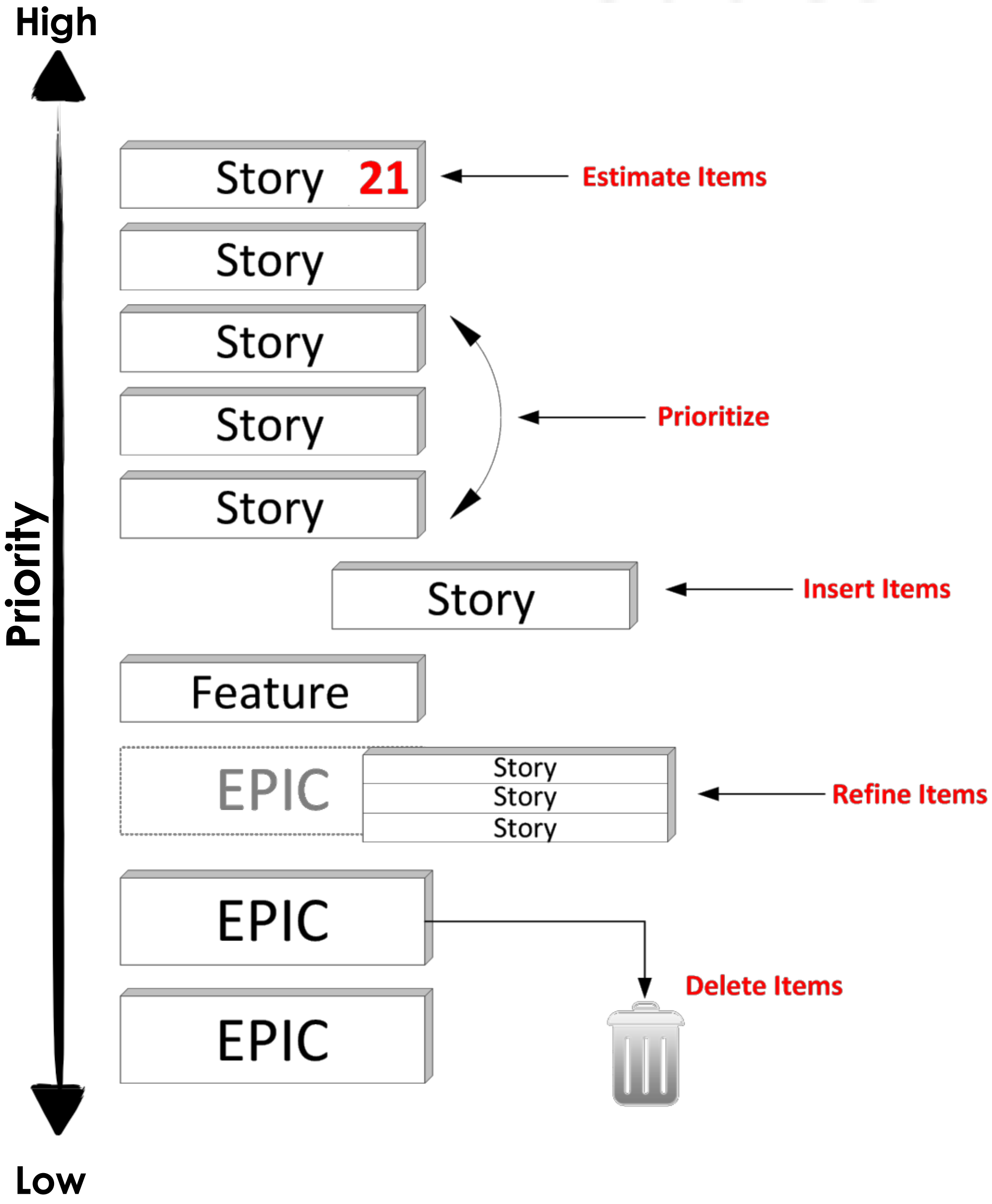
Product Owner

Development

System Users

Customers

Product Backlog Refinement



High

Priority

Low

Story 21

Estimate Items

Story

Story

Story

Story

Prioritize

Story

Insert Items

Feature

EPIC

Story

Story

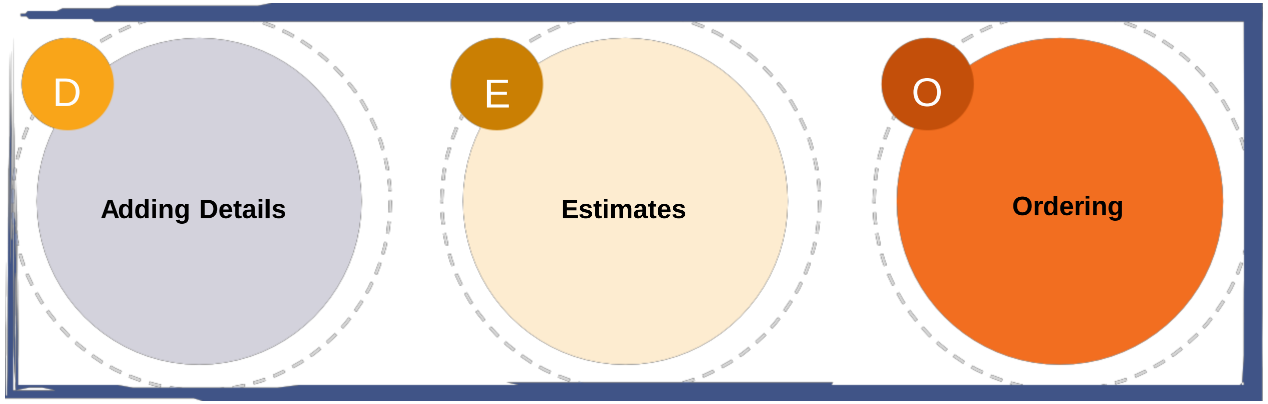
Story

Refine Items

EPIC

EPIC

Delete Items



D

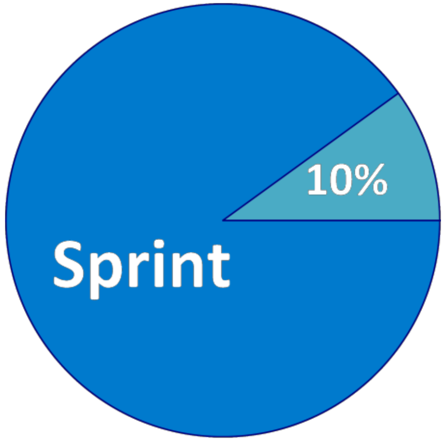
Adding Details

E

Estimates

O

Ordering



10%

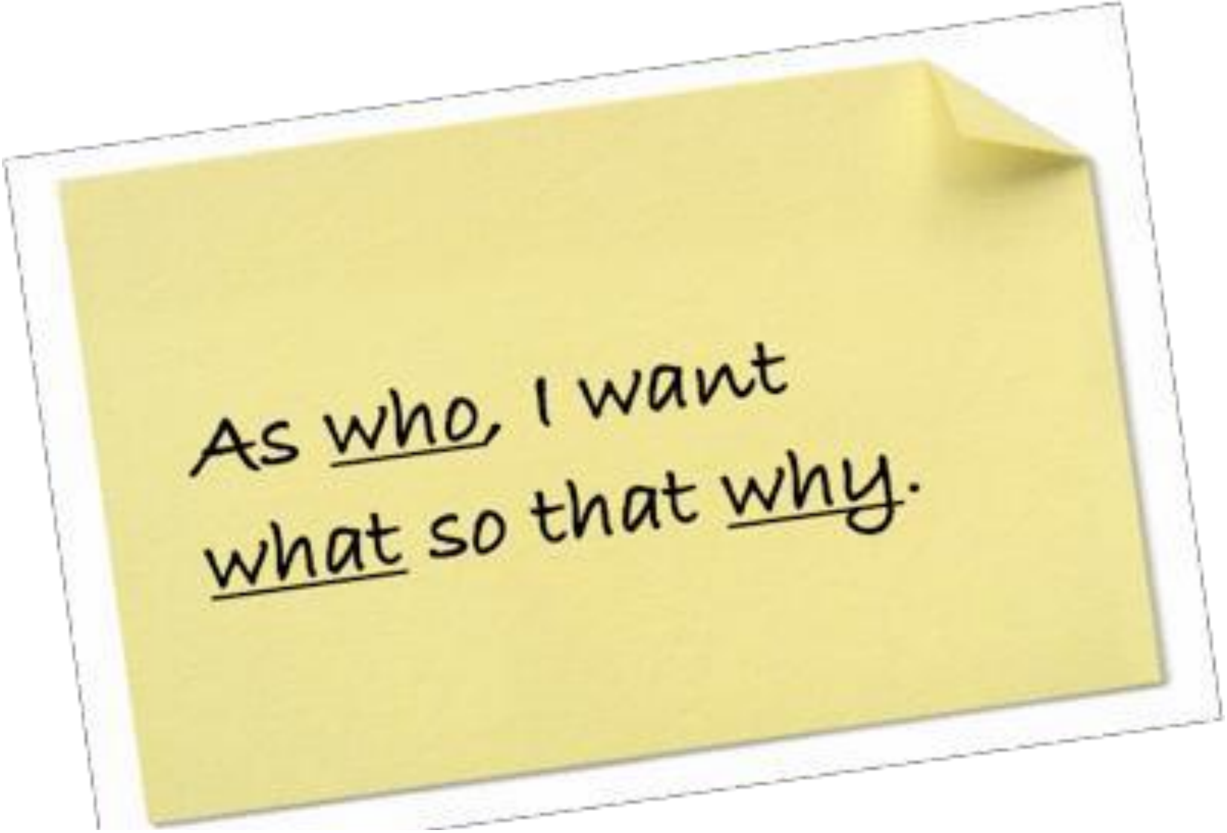
Sprint

Dev Team Capacity

User Stories

Product Backlog Items (PBIs) - User Story format

As <someone> I want <something> so that <I get some value>



“User stories are not required in scrum, they are only a type of product backlog item”

PO Should Write User Stories, Not Tasks

User Story

User Story

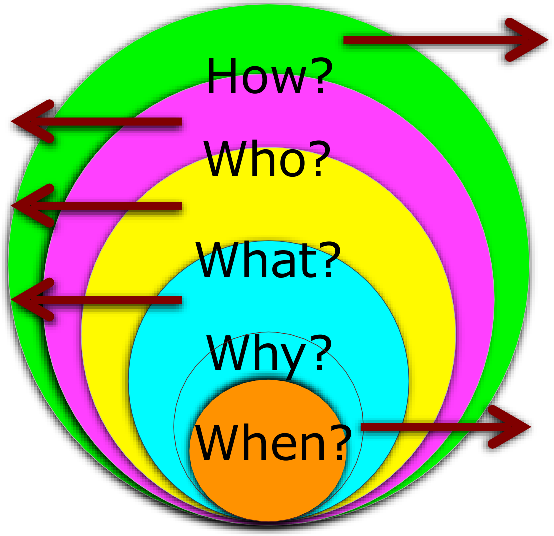
User Story

Task

Task

Task

Product Owner



How?

Who?

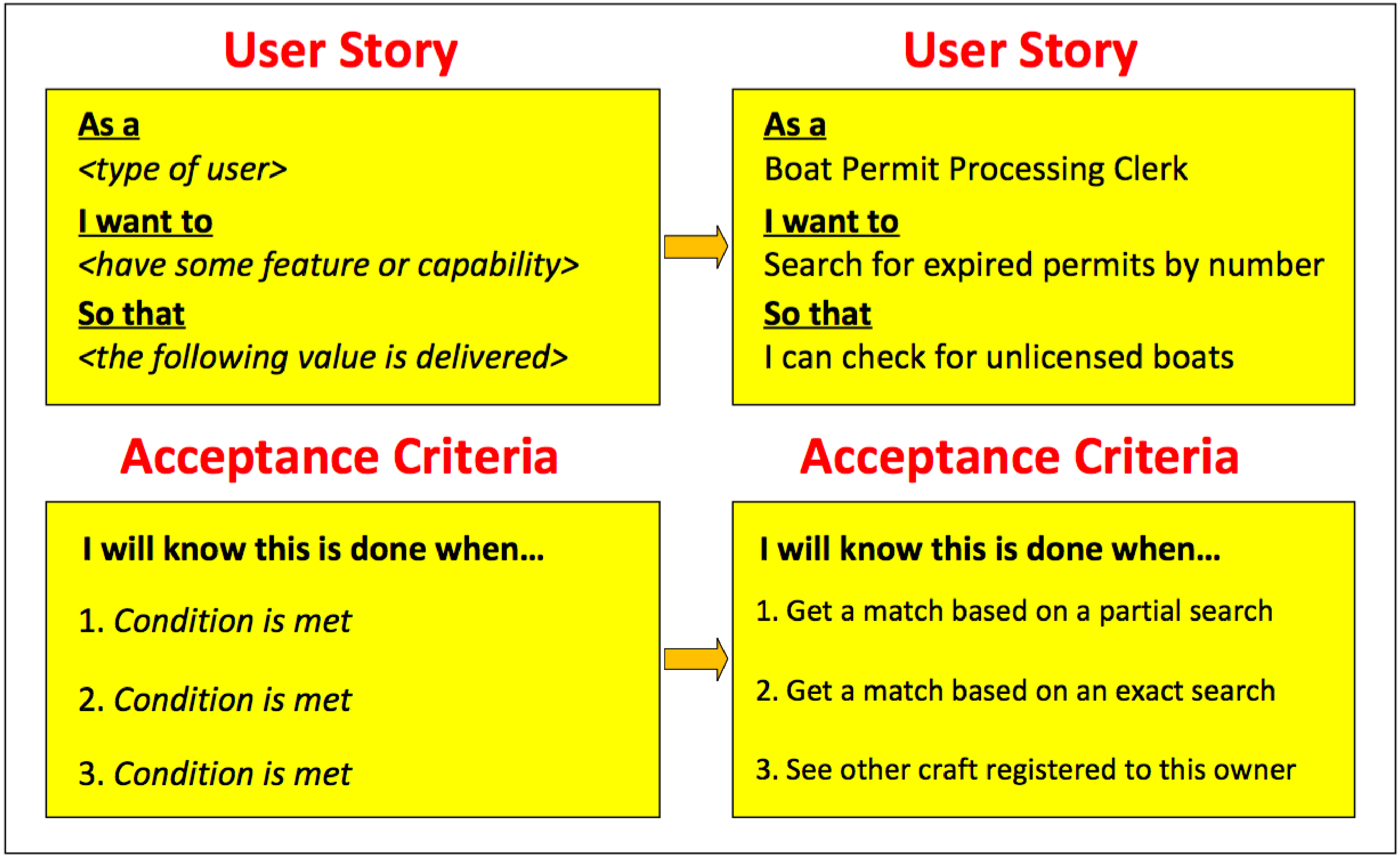
What?

Why?

When?

Development Team

Creating Good User Stories



User Story

As a

<type of user>

I want to

<have some feature or capability>

So that

<the following value is delivered>

User Story

As a

Boat Permit Processing Clerk

I want to

Search for expired permits by number

So that

I can check for unlicensed boats

Acceptance Criteria

I will know this is done when…

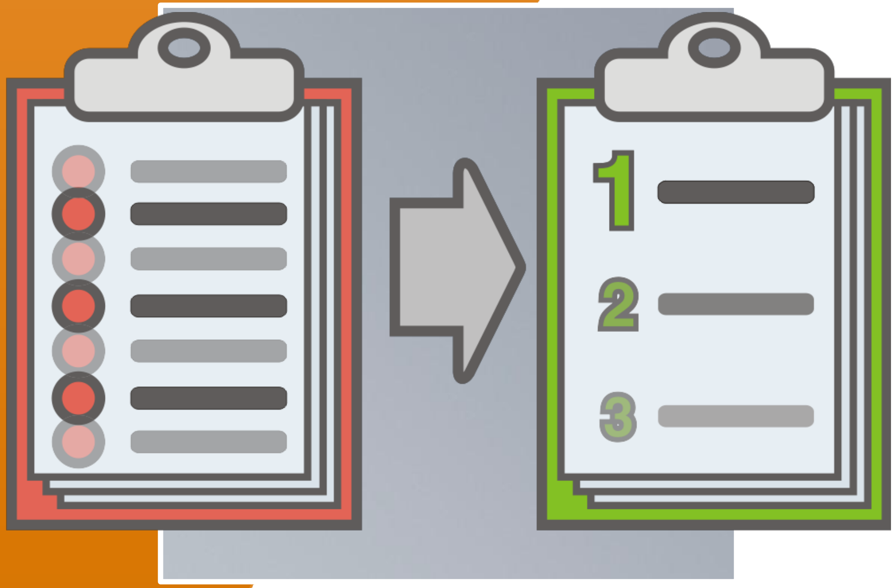
1. *Condition is met*
2. *Condition is met*
3. *Condition is met*

Acceptance Criteria

I will know this is done when…

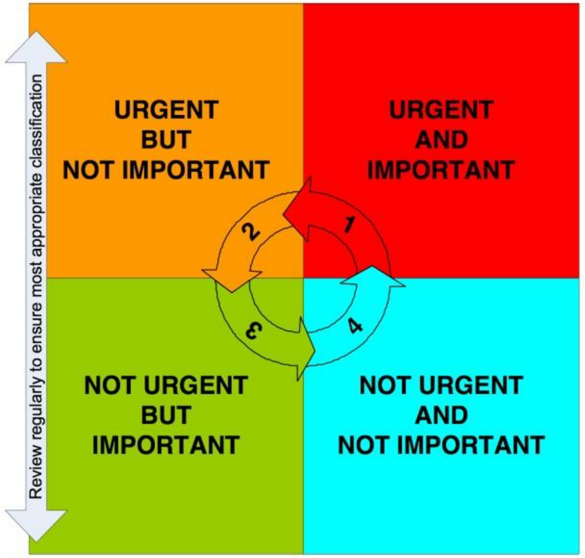
1. Get a match based on a partial search
2. Get a match based on an exact search
3. See other craft registered to this owner

Prioritization Techniques



The Eisenhower Box

NOT SCRUM



Review regularly to ensure most appropriate classification

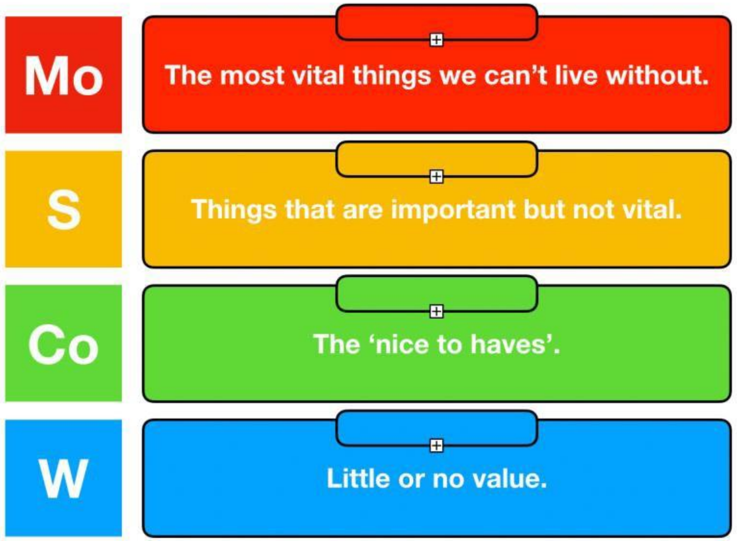
URGENT BUT NOT IMPORTANT

URGENT AND IMPORTANT

NOT URGENT BUT IMPORTANT

NOT URGENT AND NOT IMPORTANT

The Eisenhower Box



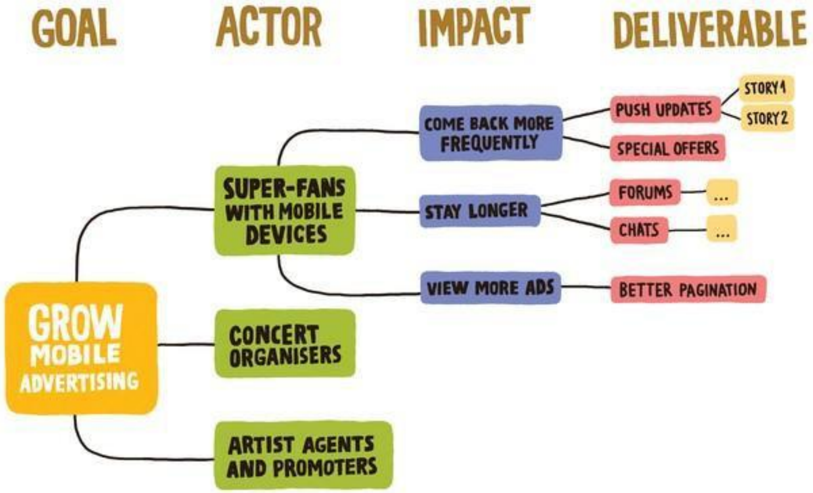
Mo — The most vital things we can't live without.

S — Things that are important but not vital.

Co — The 'nice to haves'.

W — Little or no value.

MoSCoW



GOAL — ACTOR — IMPACT — DELIVERABLE

GROW MOBILE ADVERTISING

- SUPER-FANS WITH MOBILE DEVICES
  - COME BACK MORE FREQUENTLY
    - PUSH UPDATES
      - STORY1
      - STORY2
    - SPECIAL OFFERS
  - STAY LONGER
    - FORUMS
      - ...
    - CHATS
      - ...
  - VIEW MORE ADS
    - BETTER PAGINATION
- CONCERT ORGANISERS
- ARTIST AGENTS AND PROMOTERS

Impact Mapping

Buy a Feature

Highest Priority

Medium Priority

Lowest Priority

Dot Voting

- "You ain't gonna need it"
- YAGNI is a principle of extreme programming(XP) that states programmer should not add functionality until deemed necessary.
- XP co-founder Ron Jeffries has written:

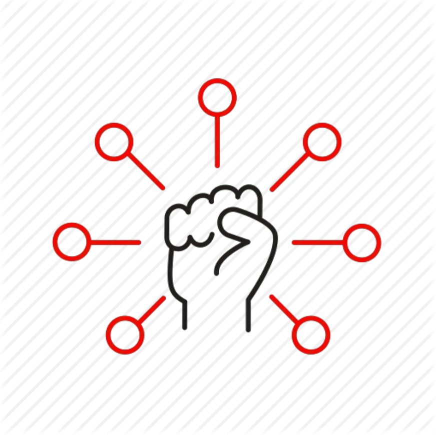
  "Always implement things when you actually need them, never when you just foresee that you need them."

- Sometimes called '**The Last Responsible Moment**'.

YAGNI

Slicing Techniques

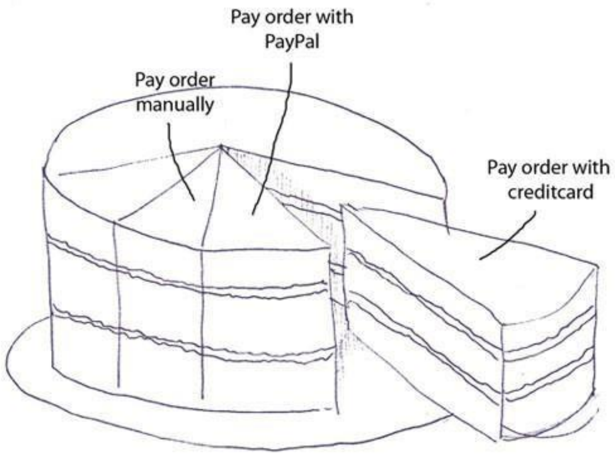
Some Splitting Techniques



Capability Offered

User Roles

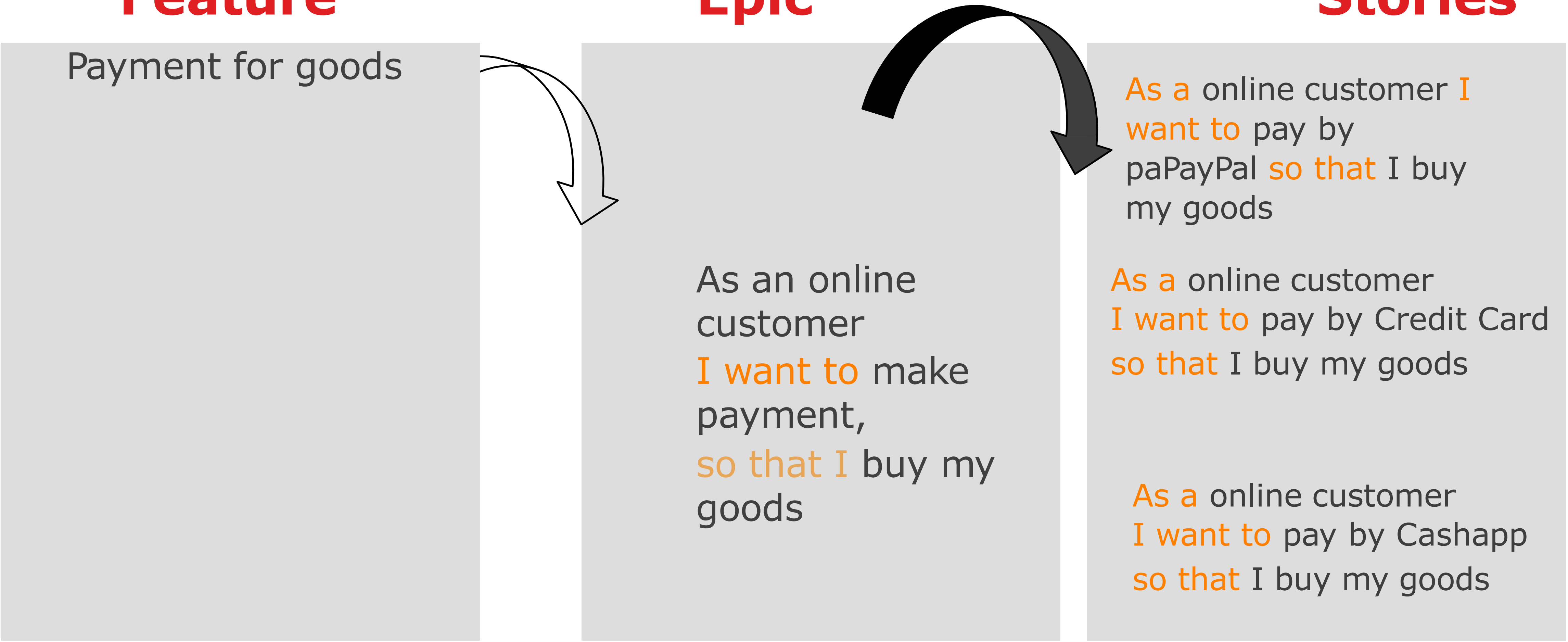
User Persona



Vertical Slicing

User Currency

Sampling Splitting



Feature

Payment for goods

Epic

As an online customer
I want to make payment,
so that I buy my goods

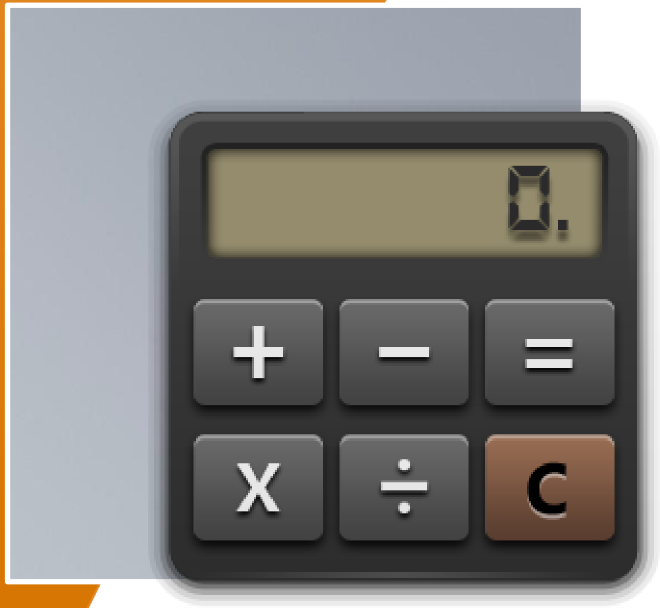
Stories

As a online customer I want to pay by paPayPal so that I buy my goods

As a online customer
I want to pay by Credit Card
so that I buy my goods

As a online customer
I want to pay by Cashapp
so that I buy my goods

Estimation



Choosing Estimation Techniques

Dephi popularly know as Planning Poker

- No Empirical Data available, or Known or Unknown Unknowns
- Higher Accuracy driven by Greater Team Discussion

T Shirt Sizing

•XS, L, M, S, XS

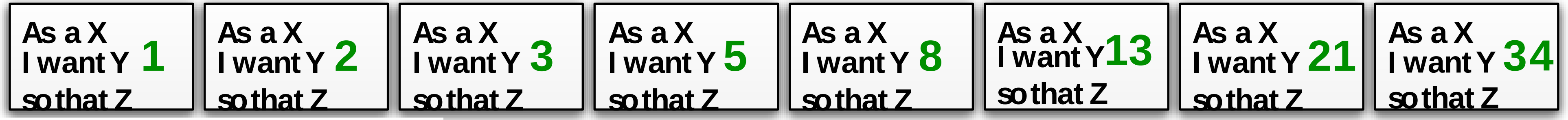
•Hard to calibrate but good to communicate uncertainty

Affinity Estimation

- Significant Empirical Data available
- Need to estimate large quantity of stories rapidly
- Need to estimate large Epics for Budgetary Purposes

Planning Poker Technique

- Known as the ‘Estimate - Talk - Estimate’ method.
- Team agrees on the smallest story and baselines it as a size 2.
- Product owner explains a story until the team is ready to size it.
- Team - keeping your thoughts to yourself, estimate relative size of other stories by comparing the size of the work including effort, complexity, risk, as well as skills available.
- Discuss outliers and vote again until all numbers are within 3 values, then average.
- The Maximum Likelihood equation for most distributions is the average.
- **Do not try to converge.**
- **The best estimate will almost never be a Fibonacci number!**



As a X
I want Y 1
so that Z

As a X
I want Y 2
so that Z

As a X
I want Y 3
so that Z

As a X
I want Y 5
so that Z

As a X
I want Y 8
so that Z

As a X
I want Y 13
so that Z

As a X
I want Y 21
so that Z

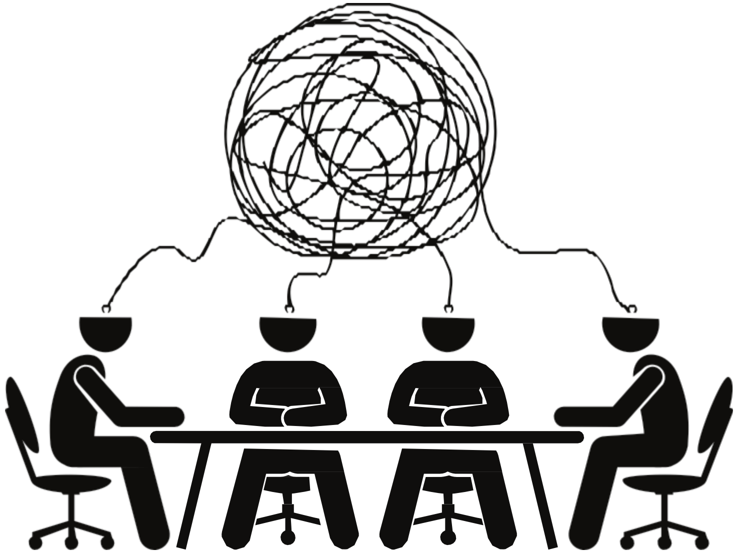
As a X
I want Y 34
so that Z

Team Speed = Velocity

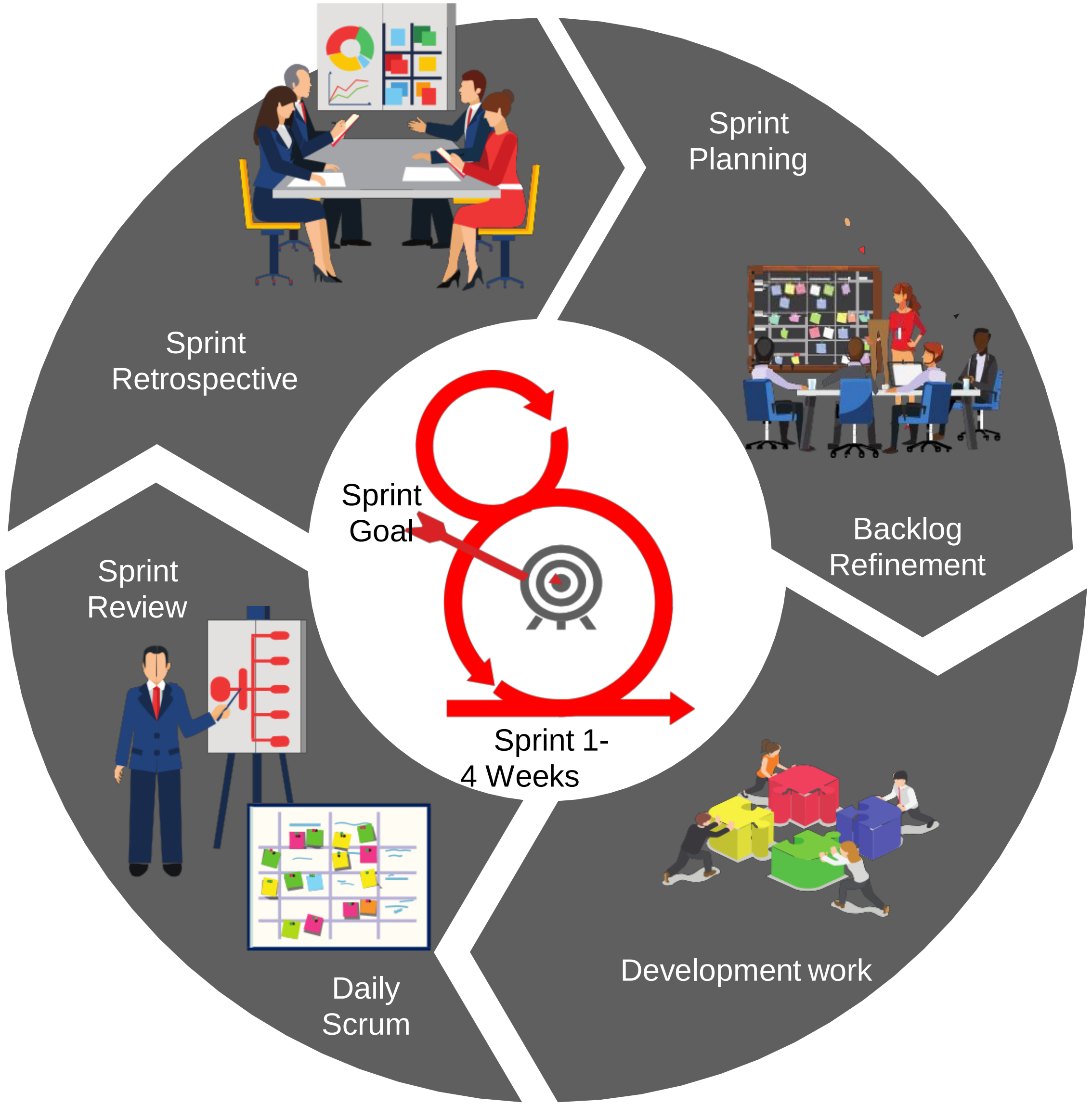
Velocity = Total of Points for all Product Backlog Items Completed.

Scrum Events

The Sprint

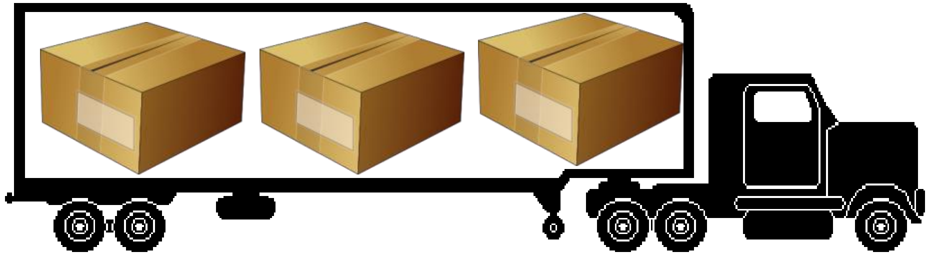


The Sprint

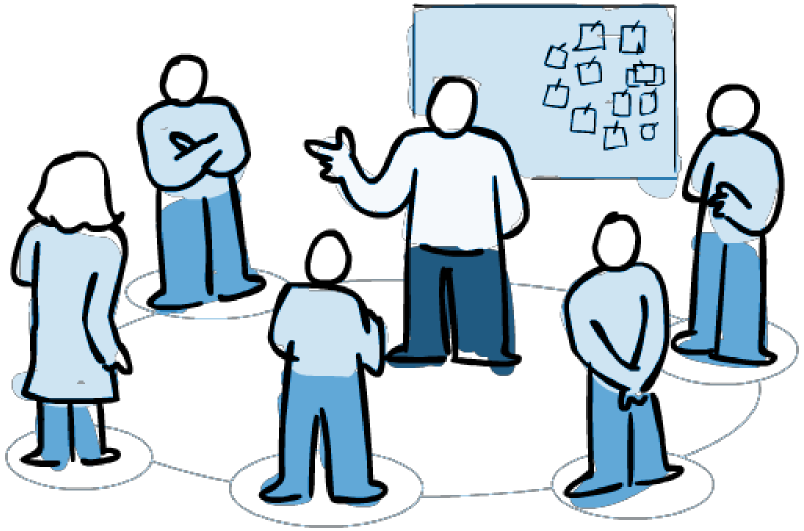

Sprint Planning
Backlog Refinement
Development work
Daily Scrum
Sprint Review
Sprint Retrospective
Sprint Goal
Sprint 1-
4 Weeks

Sprint Outcome

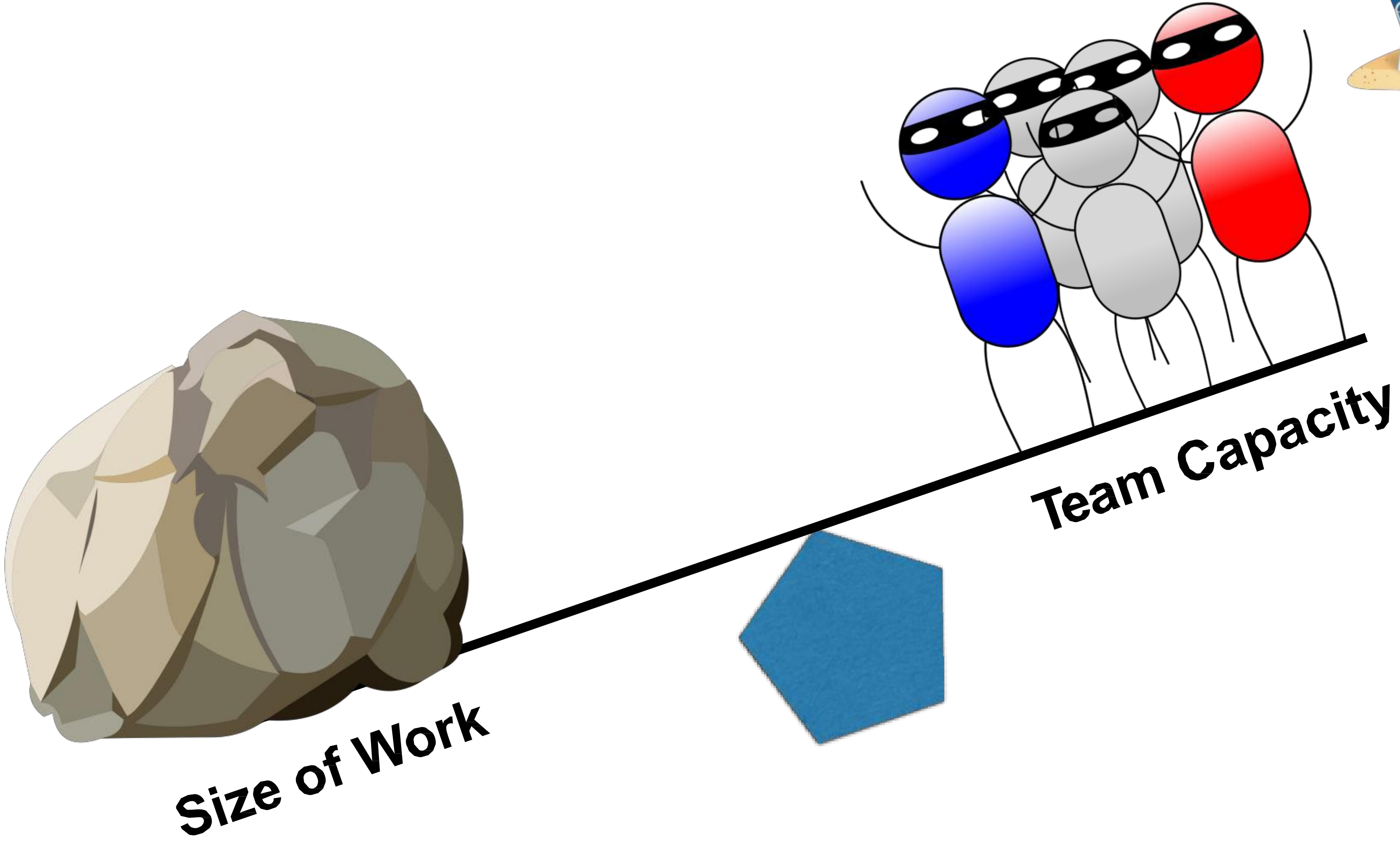
Potentially Releasable Product Increment



Sprint Planning

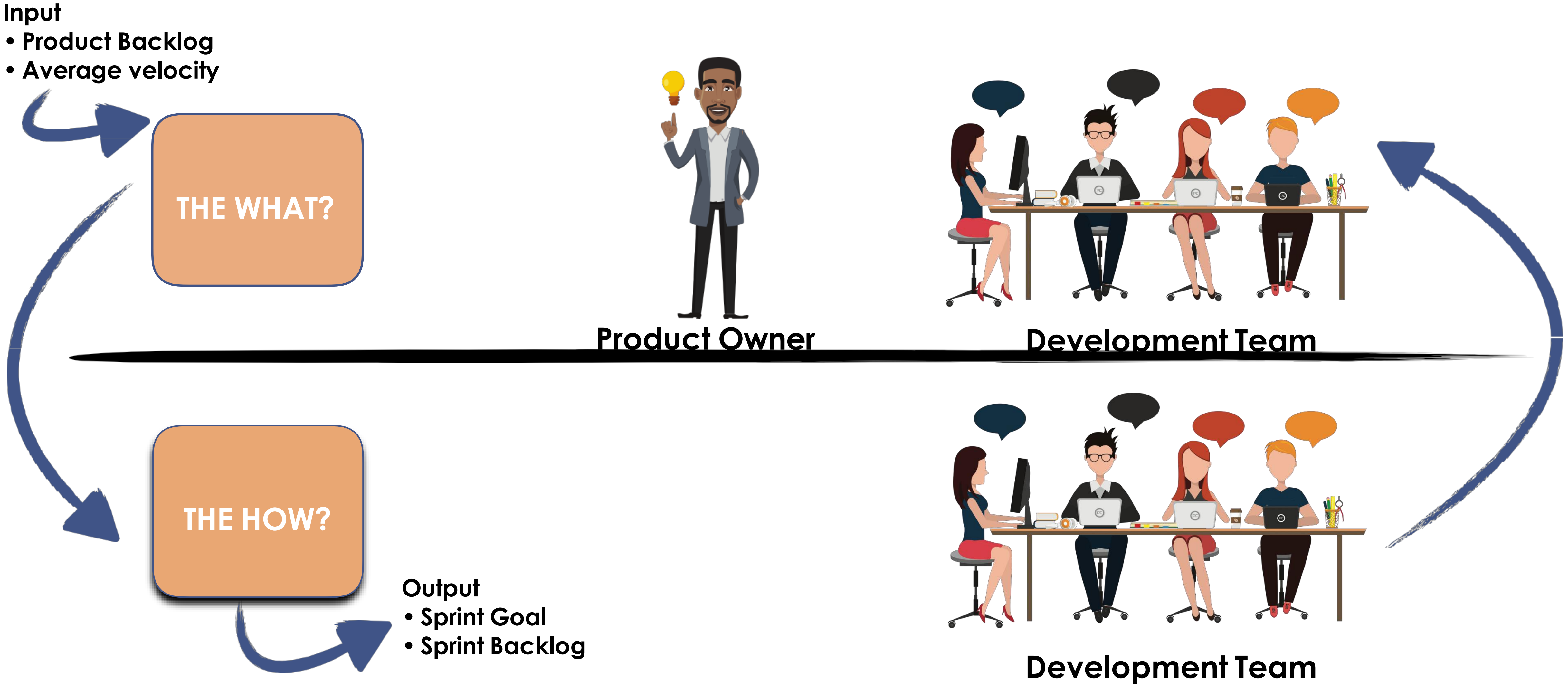


Capacity Planning



The Two Topics

This part.... is negotiated by ....



Input
- Product Backlog
- Average velocity

THE WHAT?

Product Owner

Development Team

THE HOW?

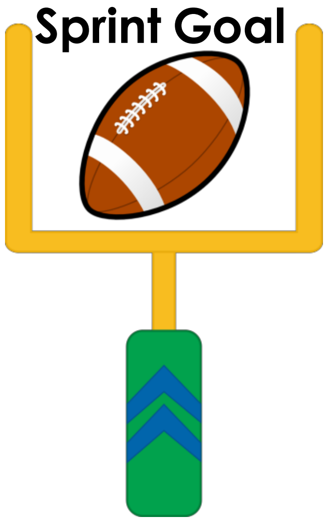
Output
- Sprint Goal
- Sprint Backlog

Development Team

Iterate between the two parts until agreement is reached

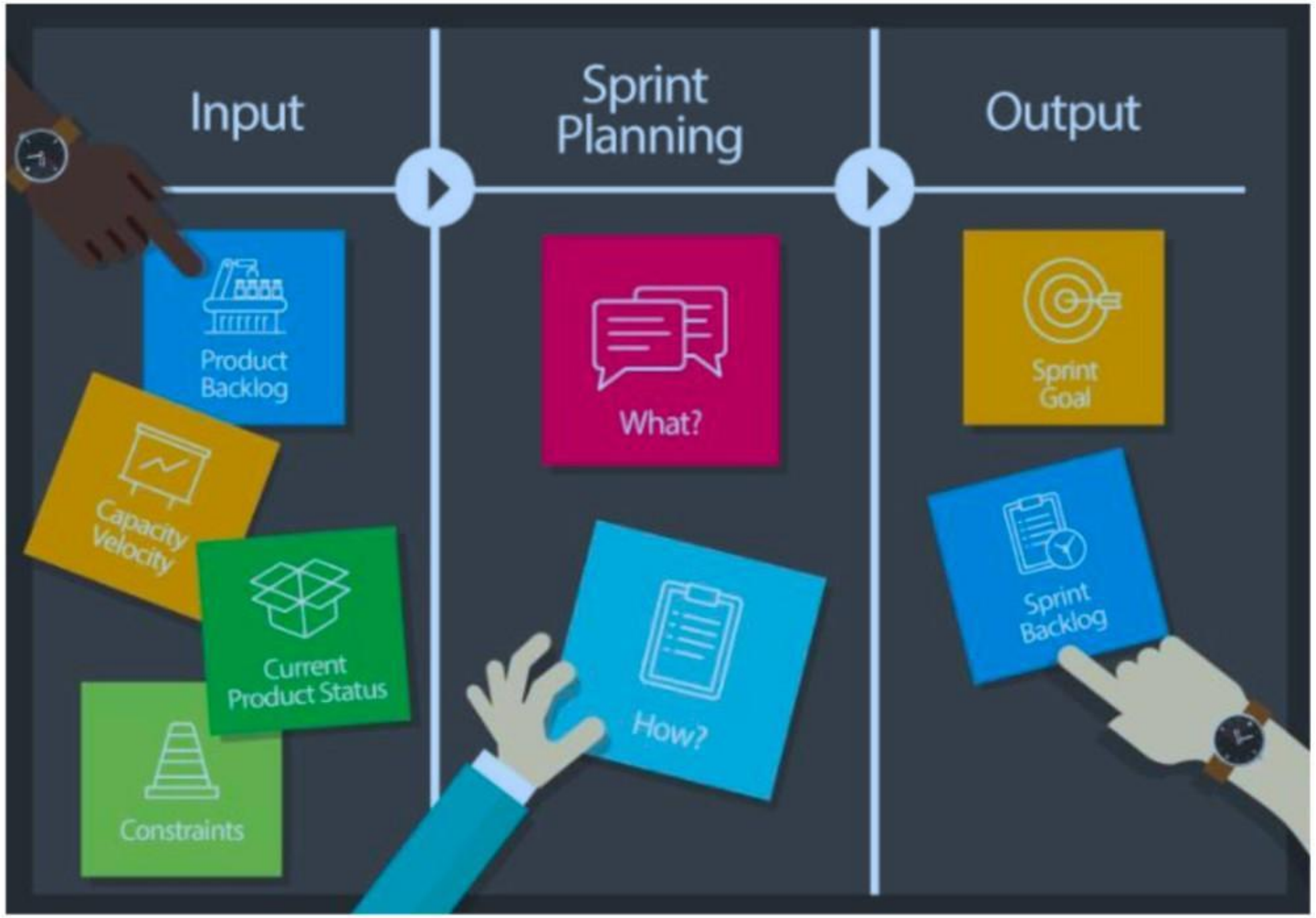
Sprint Goal

- The Sprint Goal is an objective that will be met within the Sprint through the implementation of the Product Backlog Items.
- It provides guidance to the Development Team on why it is building the Increment.
- Set during the Sprint Planning Event, and agreed between the PO & Team.
- The selected PBIs deliver one coherent feature.



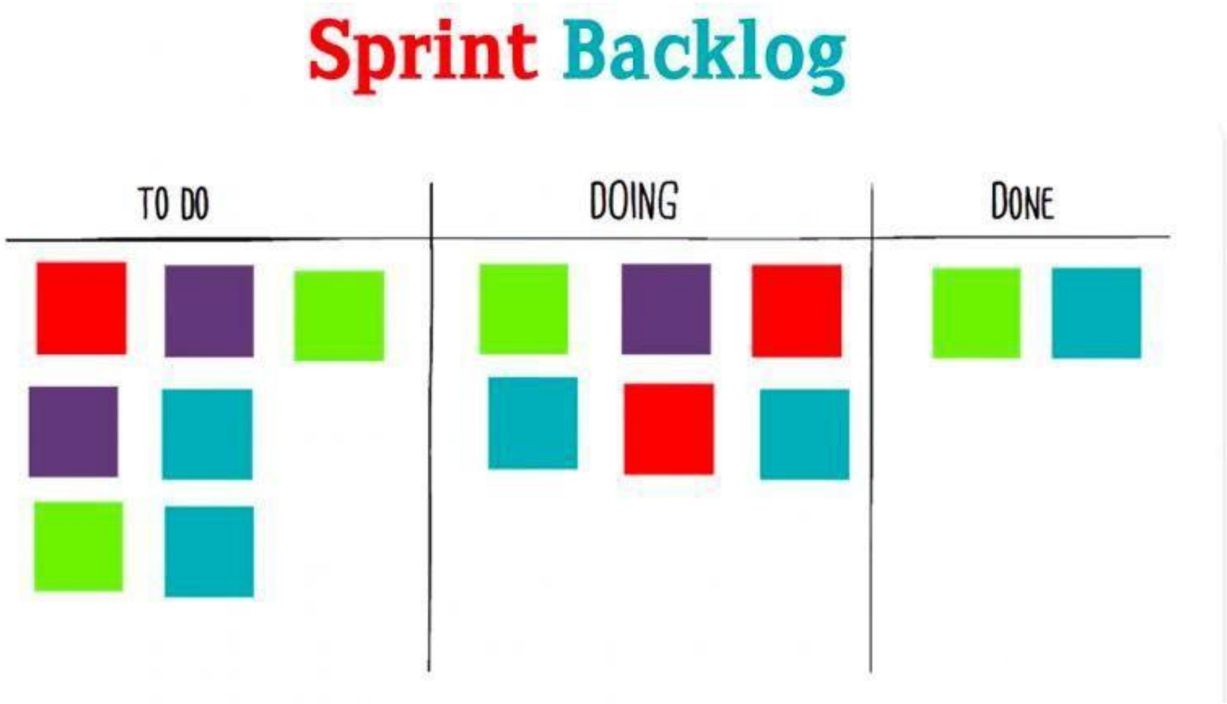
Why does the sprint goal do not change during the sprint?

Sprint Planning input & Output



Sprint Backlog

Sprint Backlog Characteristics...



What are they?

pcl.

Definition of Done

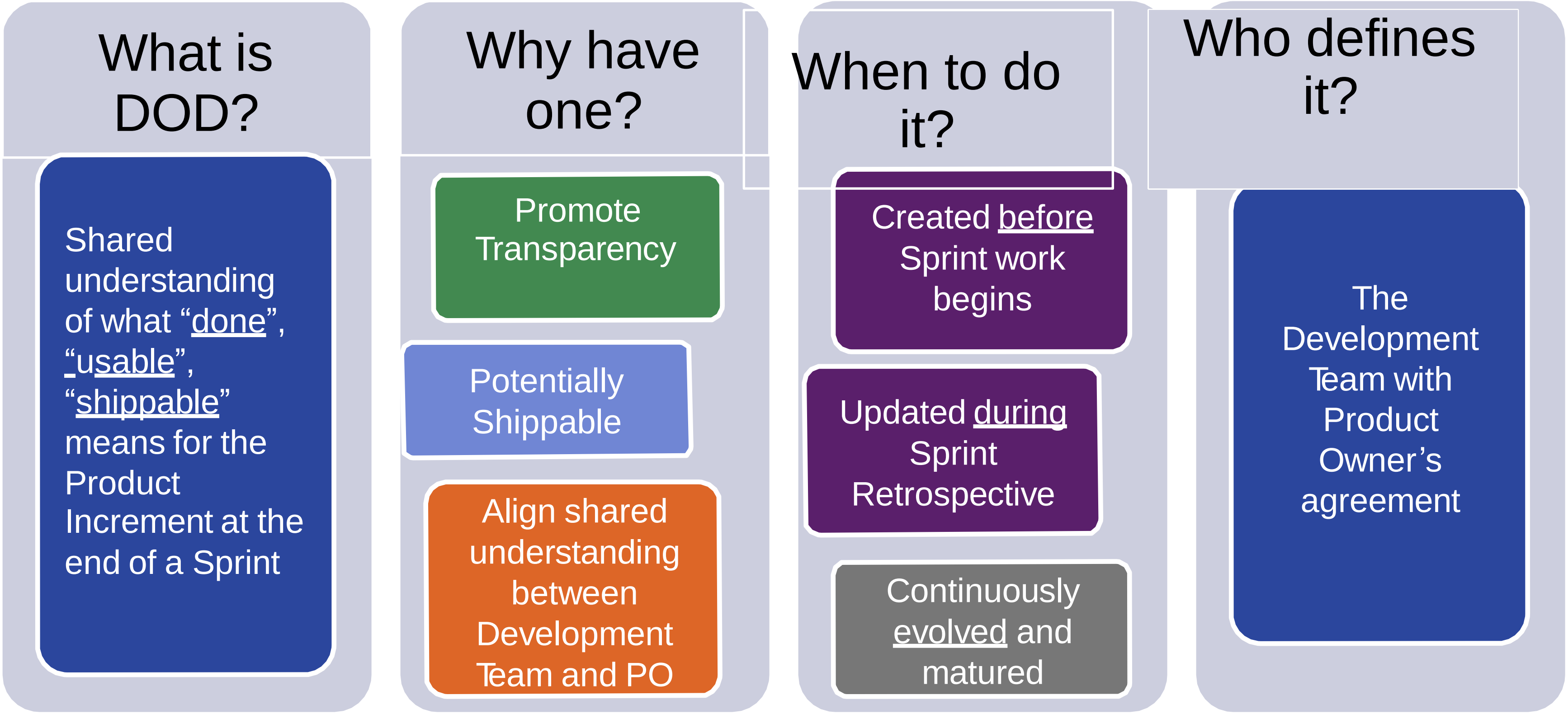
What DONE Means

- The purpose of each Sprint is to deliver Increments of potentially releasable functionality that adhere to the Scrum Team’s current definition of “Done.”

- Development Teams deliver an Increment of product functionality every Sprint. This Increment is useable, so a Product Owner may choose to immediately release it.

DONE MEANS DONE EVERY SPRINT! THAT MEANS RELEASABLE!

Definition of Done



What is DOD?

Shared understanding of what "done", "usable", "shippable" means for the Product Increment at the end of a Sprint

Why have one?

Promote Transparency

Potentially Shippable

Align shared understanding between Development Team and PO

When to do it?

Created before Sprint work begins

Updated during Sprint Retrospective

Continuously evolved and matured

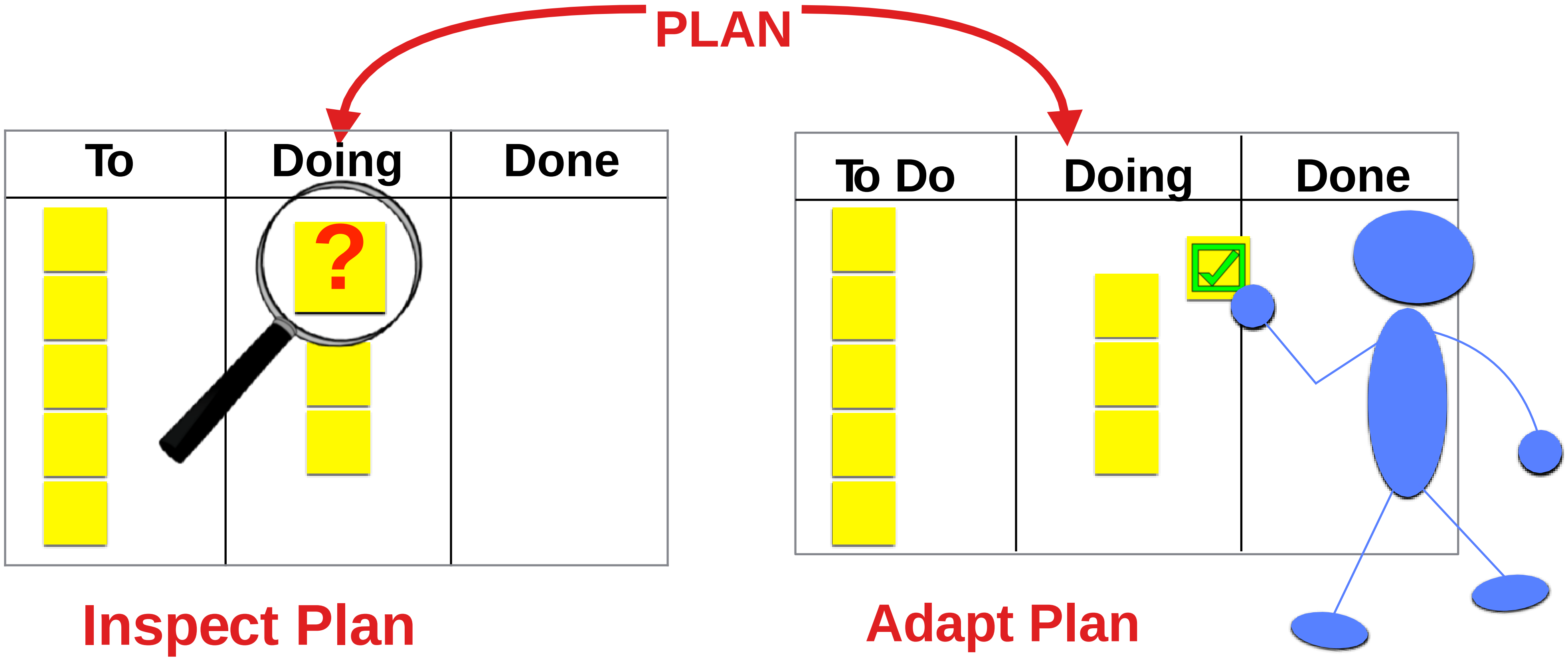
Who defines it?

The Development Team with Product Owner's agreement

DONE MEANS DONE EVERY SPRINT! THAT MEANS RELEASABLE!

Daily Scrum

Purpose of The Daily Scrum



Inspect Plan

Adapt Plan

Daily Scrum

- 15 minute
- Development Team plans work for the next 24 hours
- Optimizes team collaboration and performance by inspecting the work since the last Daily Scrum and forecasting upcoming Sprint work.
- Held at the same time and place each day to reduce complexity

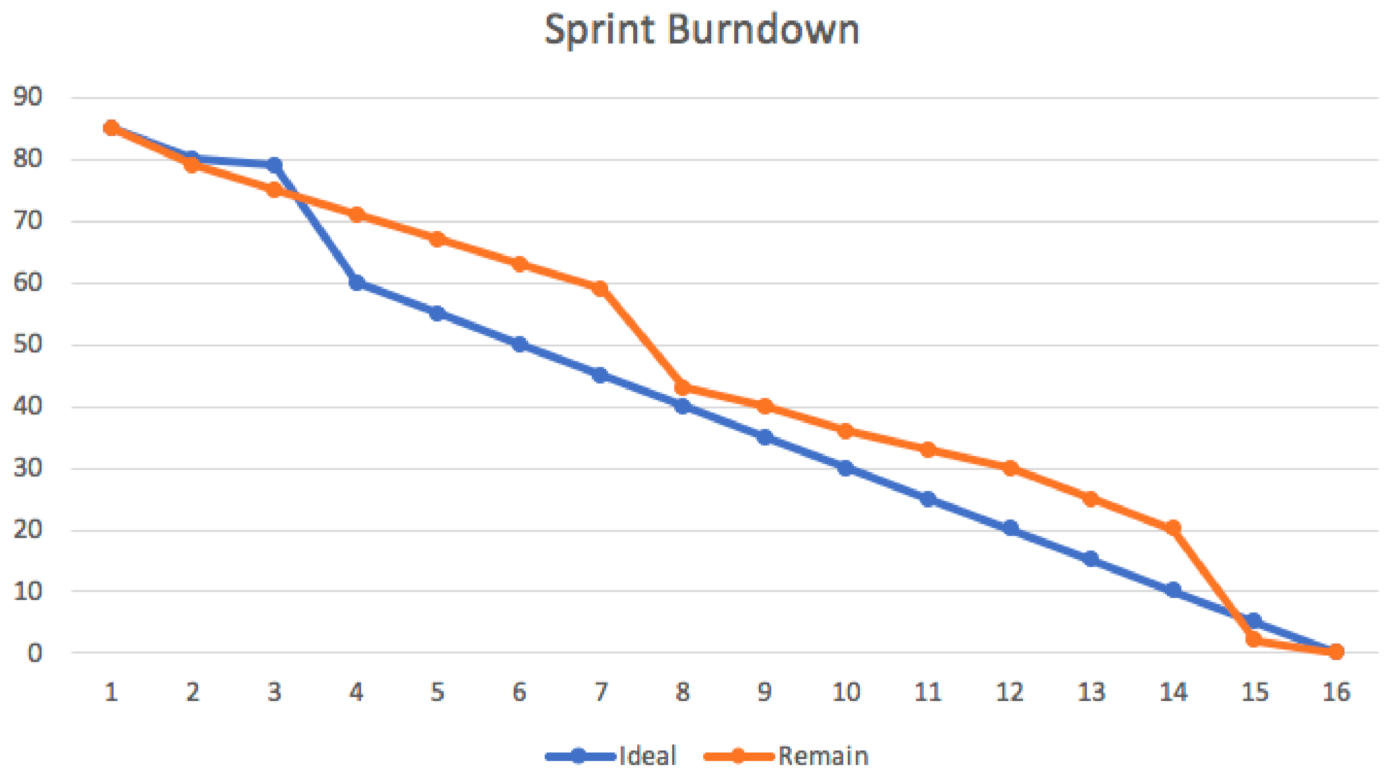
Common mistakes:

- Status reporting to management
- Focus on activity not output
- Going over time

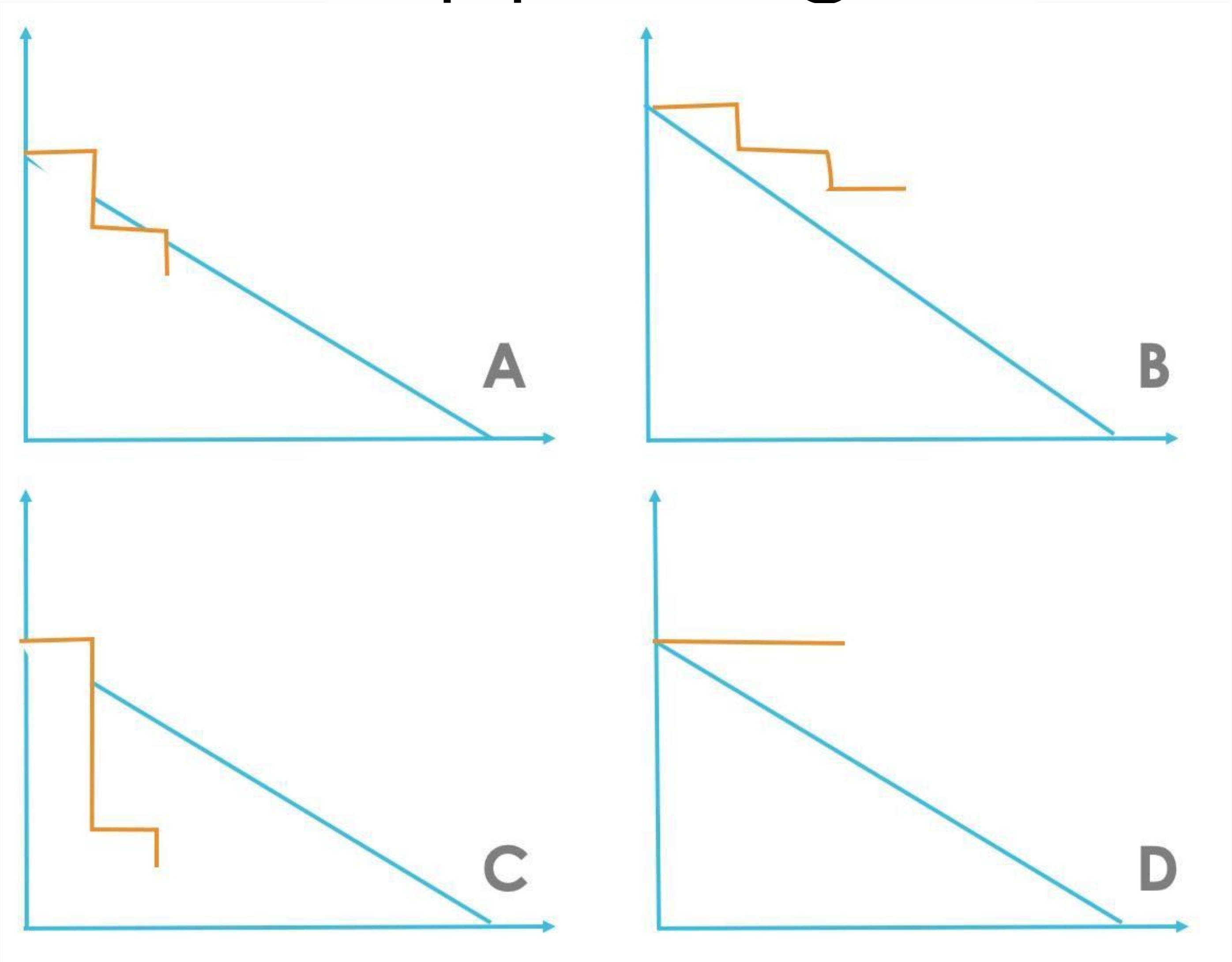
Burndown Chart

SPRINT BURNDOWN CHART

- Show the amount of work remaining for a sprint.
- Maintained by the development team.
- Effective tool for communicating to stakeholders how the extra features they are requesting will affect deadline.
- Also for reassuring the stakeholders that progress is being made
- Act as early warning indicators: can be used to highlight lack of progress.
- Highlight areas where there is redundancy.

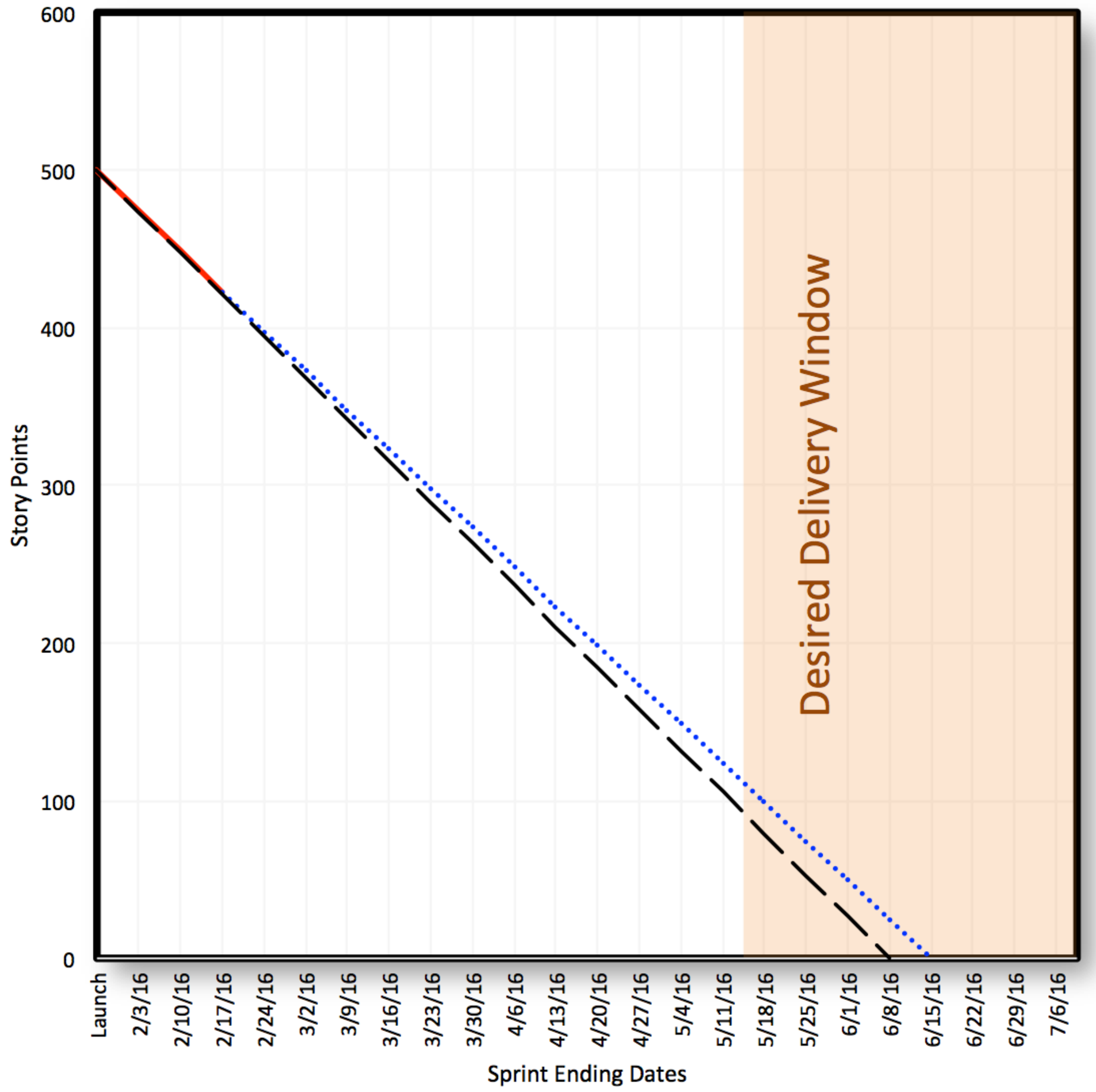


What is happening here?

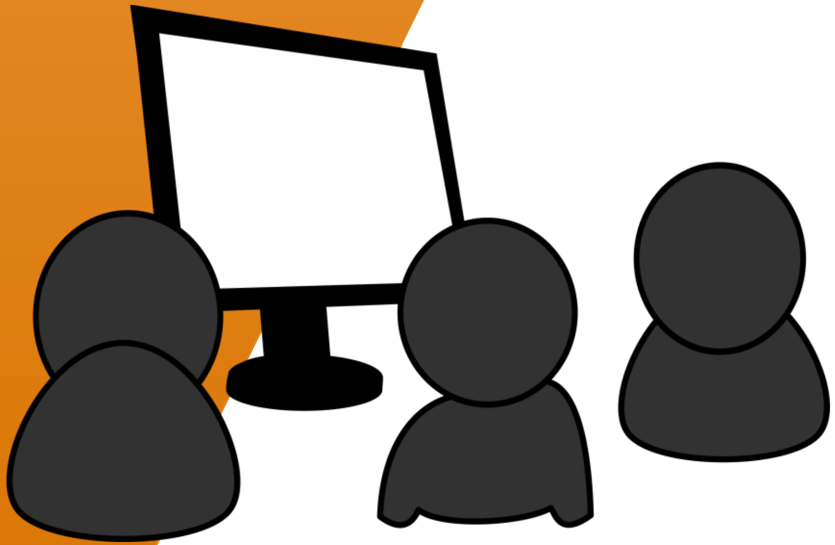


Release Burndown Chart

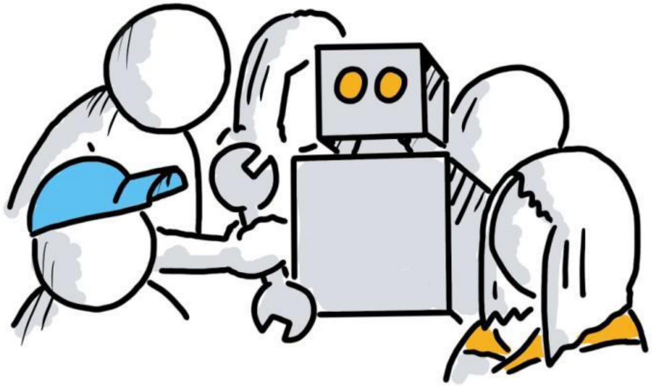
- ✓Show the amount of work remaining for the release.
- ✓It could span across multiple sprints for a team that does not release every sprint.
- ✓Maintained by the Scrum Team.
- ✓Effective tool for communicating to stakeholders how the extra features they are requesting will affect deadline.
- ✓Also for reassuring the stakeholders that progress is being made
- ✓Act as early warning indicators: can be used to highlight lack of progress.
- ✓Highlight areas where there is redundancy.



Sprint Review



Sprint Review

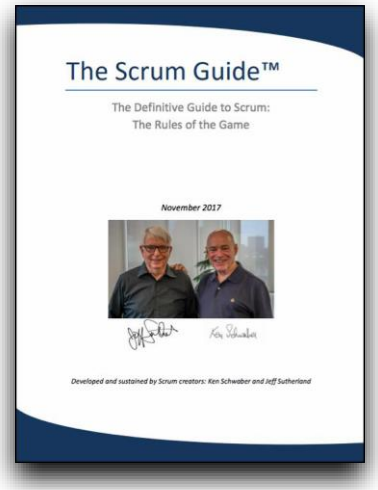


Scrum guide says…

- Attendees include the Scrum Team and key stakeholders invited by the Product Owner;
- The Product Owner explains what Product Backlog items have been “Done” and what has not been “Done”;
- The Development Team demonstrates the work that it has “Done” and answers questions about the Increment;
- The entire group collaborates on what to do next, so that the Sprint Review provides valuable input to subsequent Sprint Planning;
- Review of how the marketplace or potential use of the product might have changed and what is the most valuable thing to do next;
- Review of the timeline, budget, potential capabilities, and marketplace for the next anticipated release of the product;
- The Product Owner discusses the Product Backlog as it stands. He or she projects likely completion dates based on progress to date (if needed)

The Increment

The Increment is the sum of all the Product Backlog items completed during a Sprint and the value of the increments of all previous Sprints



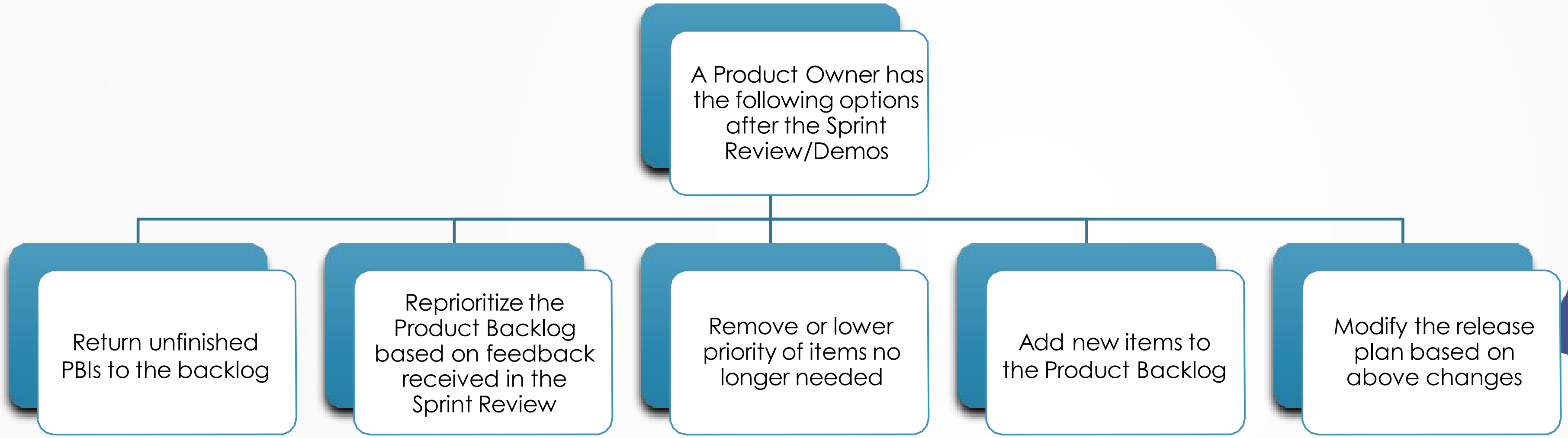
The Scrum Guide™

The Definitive Guide to Scrum:
The Rules of the Game

November 2017

Developed and sustained by Scrum creators: Ken Schwaber and Jeff Sutherland

Product Owner’s Feedback After Sprint Review



A Product Owner has the following options after the Sprint Review/Demos

- Return unfinished PBIs to the backlog
- Reprioritize the Product Backlog based on feedback received in the Sprint Review
- Remove or lower priority of items no longer needed
- Add new items to the Product Backlog
- Modify the release plan based on above changes

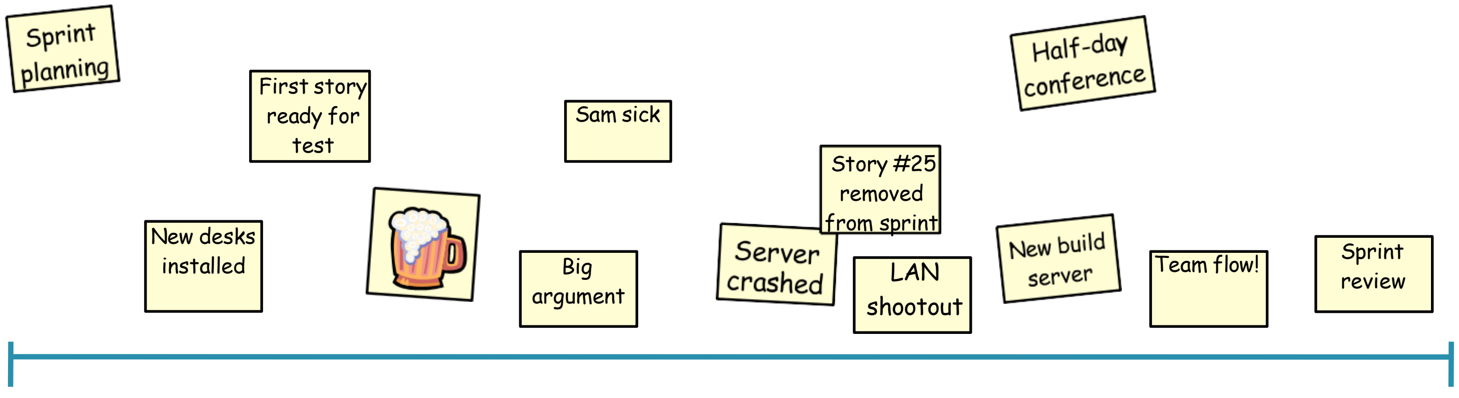
Sprint Retrospective

- Purpose is for the Scrum Team to inspect itself and create a plan for improvements to be enacted during the next Sprint.

"Which of the scrum pillars shows up in this event"

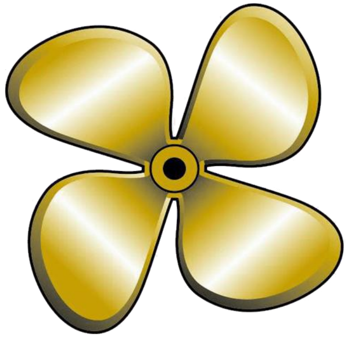
Sprint Retrospective

The Timeline Retrospective

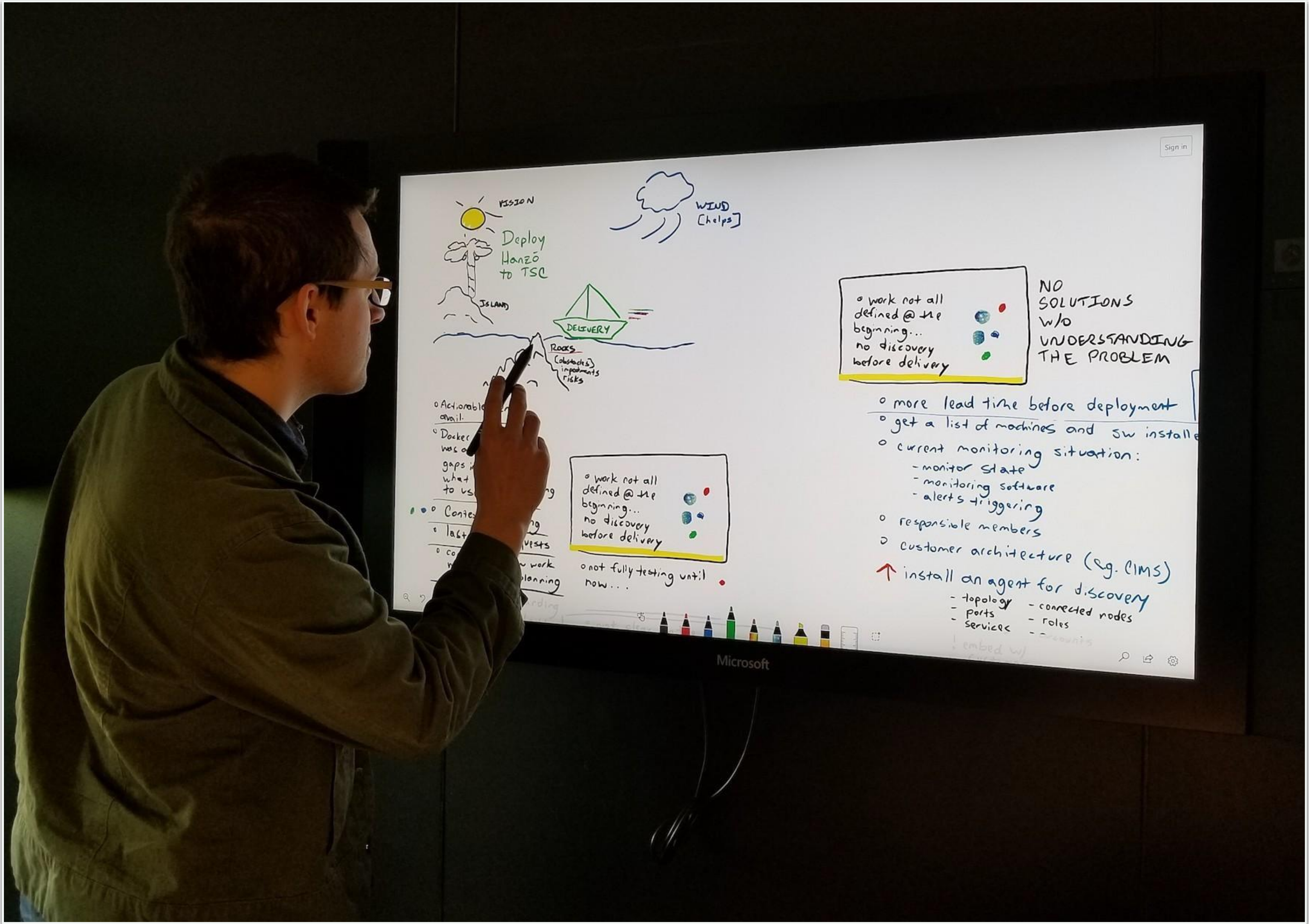


Sprint

Speed Boat Retro



Propellers
What moves us forward?



- Actionabl… avail.
- Docker was o… gaps i… what … to us… …ng
- Conte…
- last… …ng …uests
- co… n… …w work …planning

- work not all defined @ the beginning… no discovery before delivery
- not fully testing until now…

- work not all defined @ the beginning… no discovery before delivery

NO SOLUTIONS w/o UNDERSTANDING THE PROBLEM

- more lead time before deployment
- get a list of machines and sw installe…
- current monitoring situation:
  - monitor state
  - monitoring software
  - alerts triggering
- responsible members
- customer architecture (eg. CIMS)

↑ install an agent for discovery
- topology
- ports
- services
- connected nodes
- roles
- --

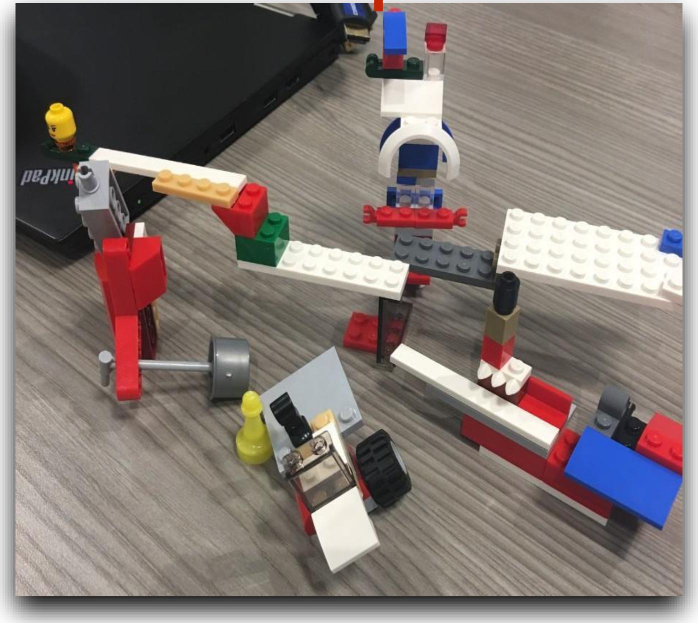
Life Belt
What can save or help us?

Anchors
What holds us back?

Rocks
Where can we crash?

Lego Serious Play Retro

Last Sprint



- Unstable bases
- Striking balance
- Lot of scope for improvement
- Lot of unknowns
- Better planning
- More insight to process
- Need to adapt

How It Works

- Create a model that reflects the last Sprint.
- Each person describes their model.
- Scrum Master records key observations.
- Team members then create a model to reflect improvements for the next Sprint.
- Scrum Master records insight to what improvements are needed.

Next Sprint

- Want Stable Bases
- More clear vision
- More refined structures
- Missiles and boats to tackle impediments
- Similar structure shows team collaboration and shared vision
- Establish Trust
- Kaizen

Sprint Retrospective

Scrum guide says..

- The purpose of the Sprint Retrospective is to:
- Inspect how the last Sprint went with regards to people, relationships, process, and tools;

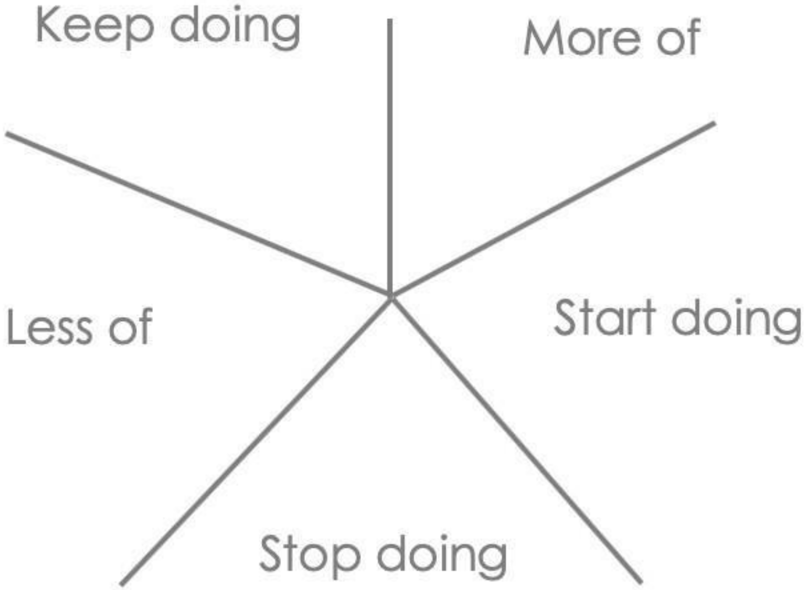
- Identify and order the major items that went well and potential improvements; and,

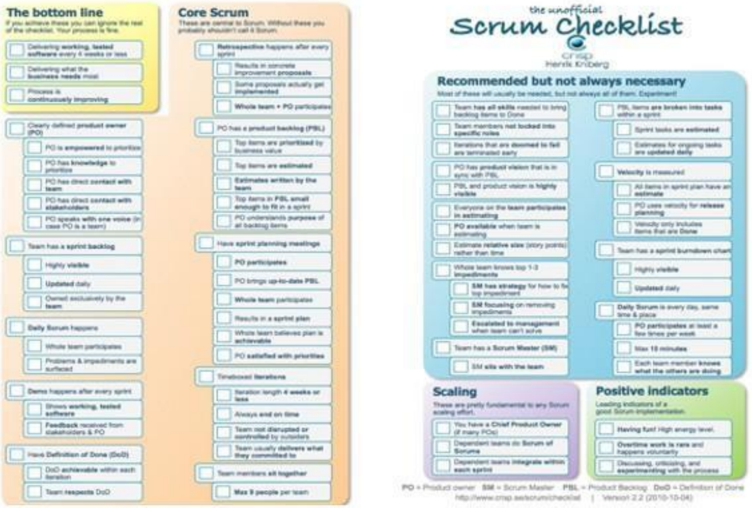
- Create a plan for implementing improvements to the way the Scrum Team does its work.

This is at most a three-hour meeting for one-month Sprints. For shorter Sprints, the event is usually shorter

Some Techniques

What went well?	Would have been even better if…





What went well?	What didn't go well?
What did we learn?	What still puzzles us?

Task 4

(

Are you Agile Ready?

5 Reasons you aren’t ready

- **Agile isn’t well understood**
- **Key stakeholders are resistant**
- **Your organization cannot support daily collaboration**
- **The company structure cannot support cross-functional teams**
- **Your organization requires heavy documentation**

(

Use Cases relevant to the masterplan

- Use case 1: Disaster Risk Forecasting and Planning

- Use case 2: Application for Building Permit

- Use case 3: City Planning, Siting of Infrastructures/Facilities in the Communities

- Use case 4: Emergency Response and Relieve Material Distribution

The End