

# CS-SDF Agile Lifecycle Training

## Scrum Master



Enabling Practices & Tools

Introduction to this course

## Course Etiquette

- Please turn your mobile phones off
- No laptops
- We start on time after breaks
- Please leave the seminar room clean and tidy
- Questions are encouraged at any time

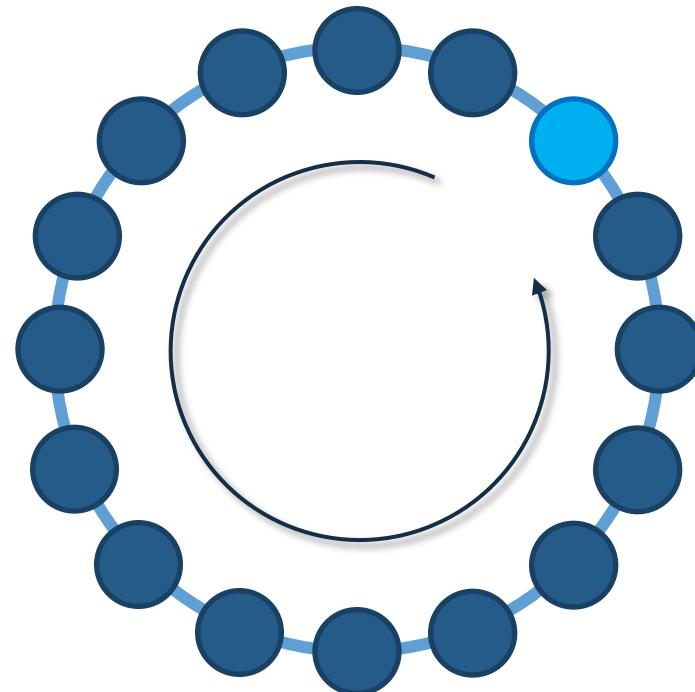
Introduction to this course

## Get to Know Each Other

- Please introduce yourself
- What do you want to take with you after this course

### Framework

- Your Name
- Your current Role
- What would you like to get out of this course?



# Agile learning map

Leadership

**Lean Leadership  
for  
Agile Managers**  
0.5 days  
Techniques and behaviors for Lean/Agile Leaders

Specialization

**Scrum Master**  
1 day

SM role introduction

**Product Owner**  
1 day

PO role for Scrum and Kanban

**Odyssey for a Scrum Team**  
1 day

Foundation

**Scrum Intro**  
1 day

Intro to Scrum

**Kanban**  
1 day

Intro to Kanban

**Lean Practices**  
1 day

Lean mini-bootcamp

**SAFe ScrumXP**  
2 days

Intro to SAFe (practitioners)

**Leading SAFe**  
2 days

Intro to SAFe (management)

**Odyssey Intro**  
WBT,  
self-study

CS-SDF ALCC

EPT Tooling



Certification included



Certification optional

Introduction to this course

# Training Objectives

As an outcome of this training,  
participants should:

- Introduce the role of the Scrum Master: behaviors and techniques
- Understand the borders of the role of the Scrum Master
- Recreate typical scenarios a Scrum Master may be faced with



# Agenda

1

Scrum Introduction

2

Role Scrum Master

3

Performing Team

4

Sprint

5

Backlog refinement

6

Sprint Planning

7

Motivation

8

Daily Business

9

Sprint Review and Demonstration

10

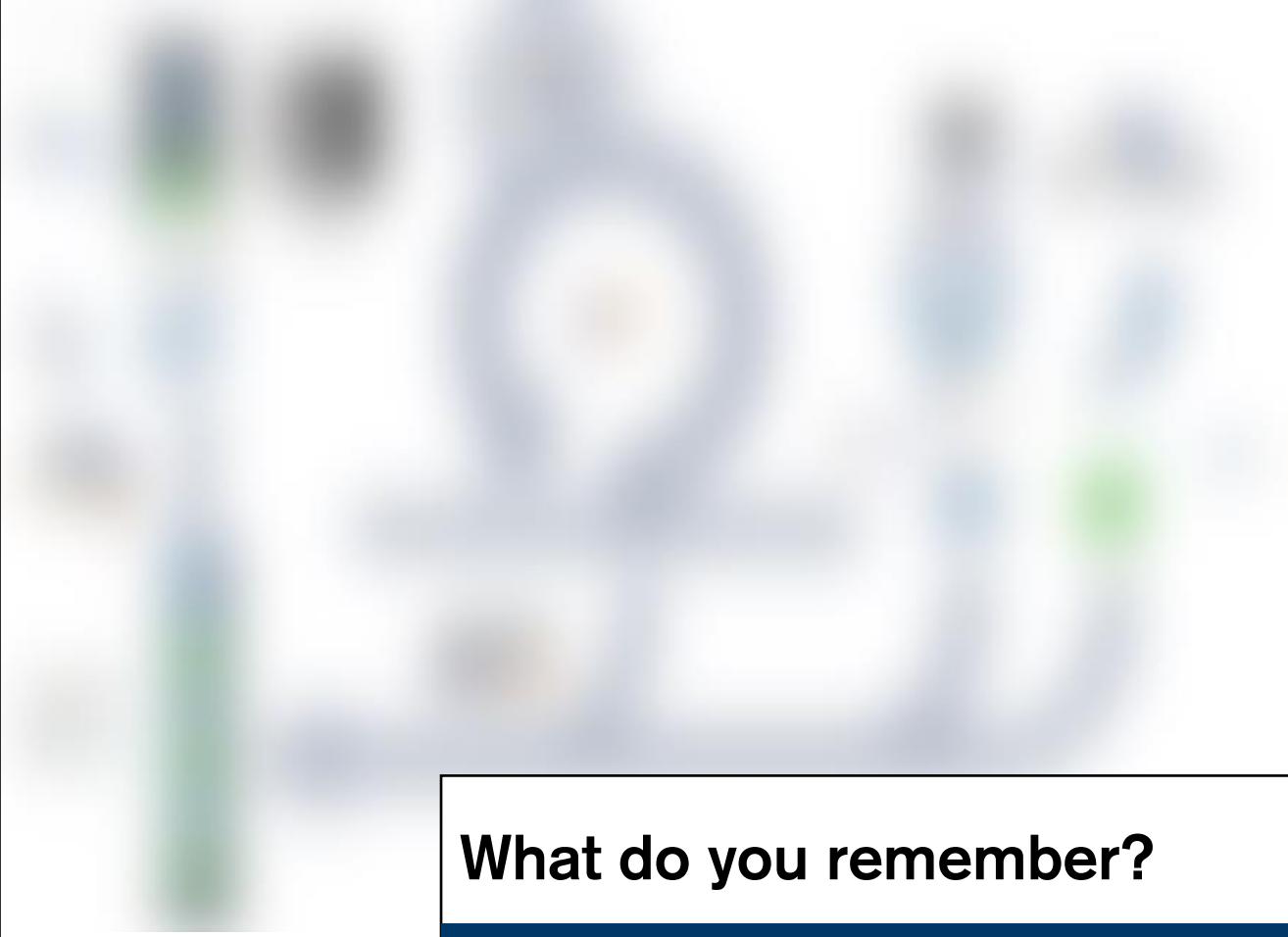
Retrospective

11

Track Progress

# Scrum Introduction

# Exercise

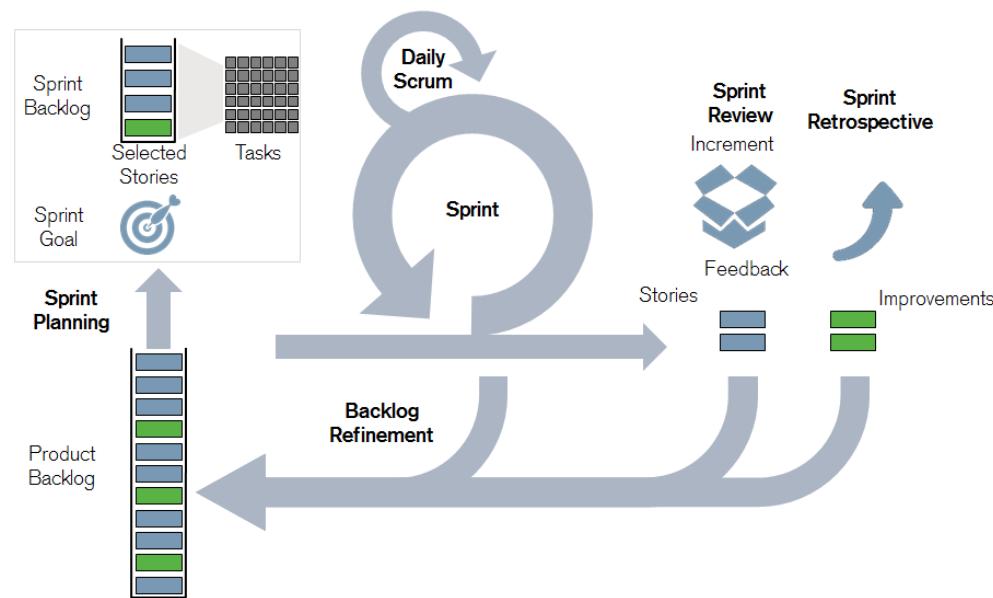


**What do you remember?**

# Scrum Introduction

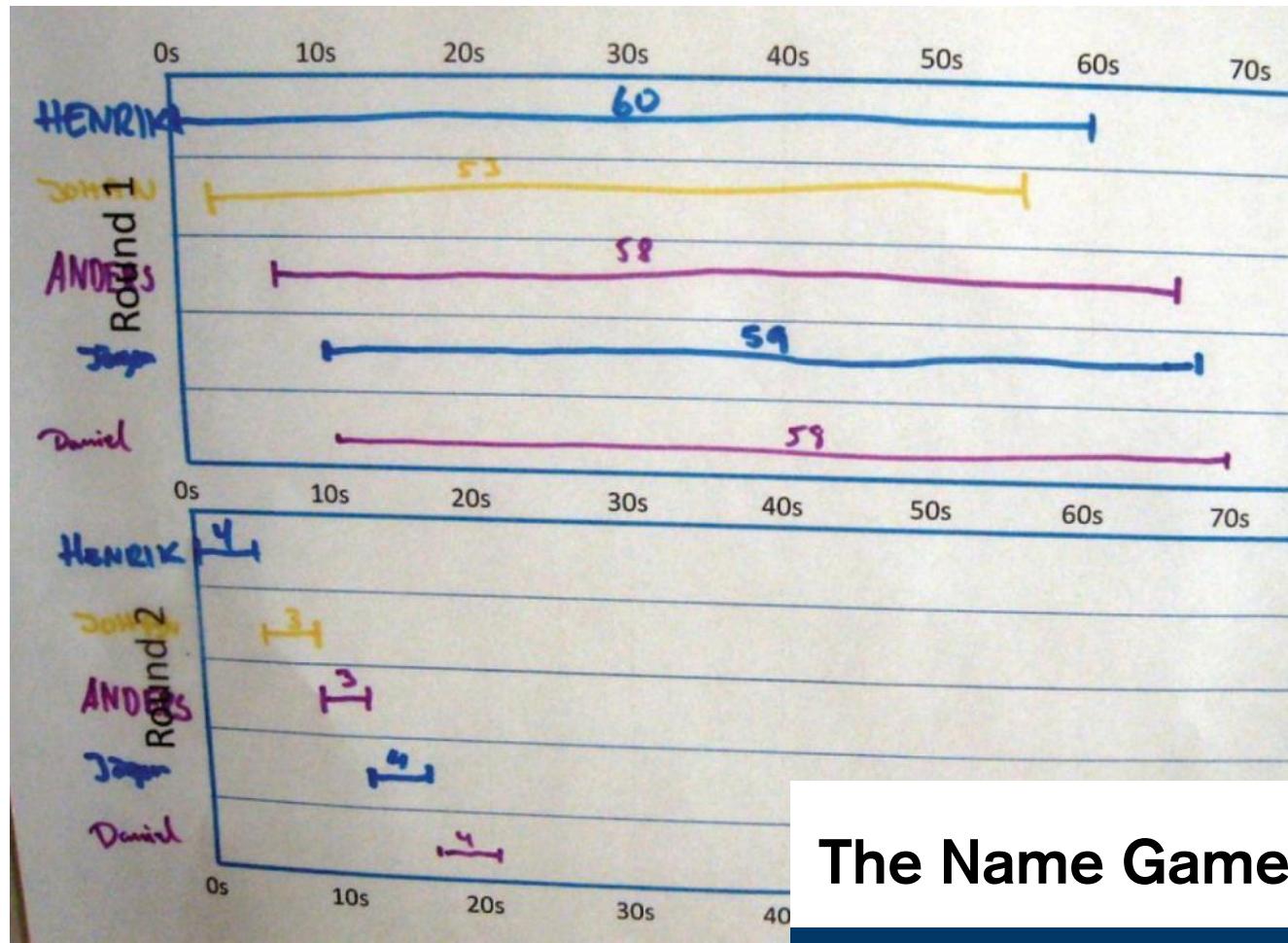
## Recap

- An empirical process, for both Product Development and Process Improvement
- Timeboxed set of activities in “Sprint”, helps team focus only on planned Stories
- Frequent team alignment to focus on work to be done, in “Daily Standup Meetings”
- Sprint work is reviewed and a shippable increment of Product shown to customer
- Built on core themes like
  - Value
  - Focus
  - Rhythm
  - Collaboration
  - Self-organization



# Role Scrum Master

## Exercise



### The Name Game

Source: Henrik Kniberg

Version: 2011-12-07

Creative Commons

Role Scrum Master

## Keeps the Big Picture in perspective



Enable the team to reach its potential...

... while working in a stable environment...

... by eliminating any obstacles the team may face.

# To be and not to be

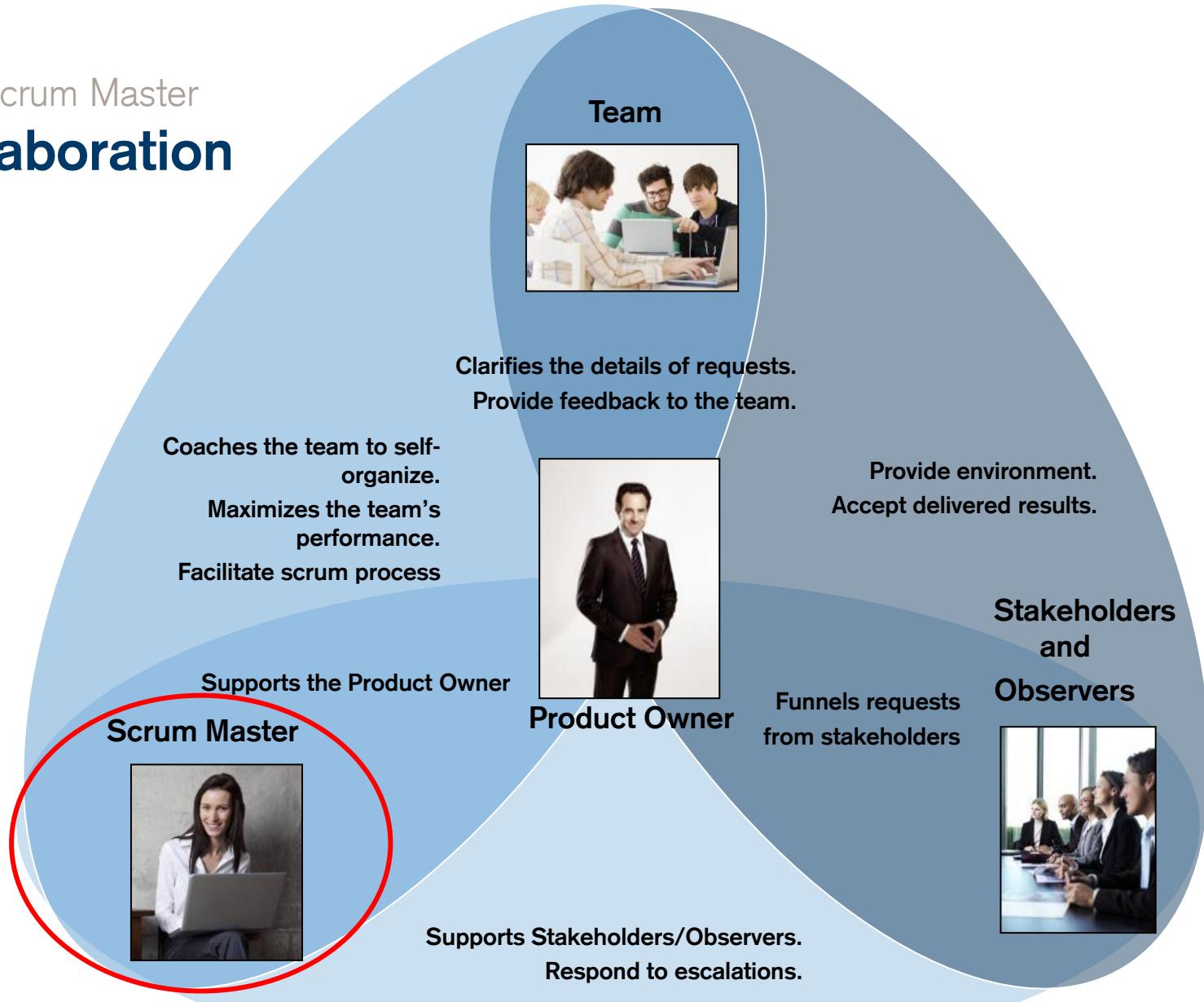
## The Scrum Master is a

- Coach
- Trainer
- Challenger
- Facilitator
- Enabler
- Defender
- Motivator

## Rather than a

- Leader
- Controller
- Task dispatcher
- Organizer
- Product expert
- Manager
- Boss

# Role Scrum Master **Collaboration**



## Working with the Team

- Challenge the team
- Support the team
- Resolve impediments the team may have
- Coach the team
- Bring focus

Team



Coaches the team to self-organize.

Maximizes the team's performance.

Facilitate scrum process

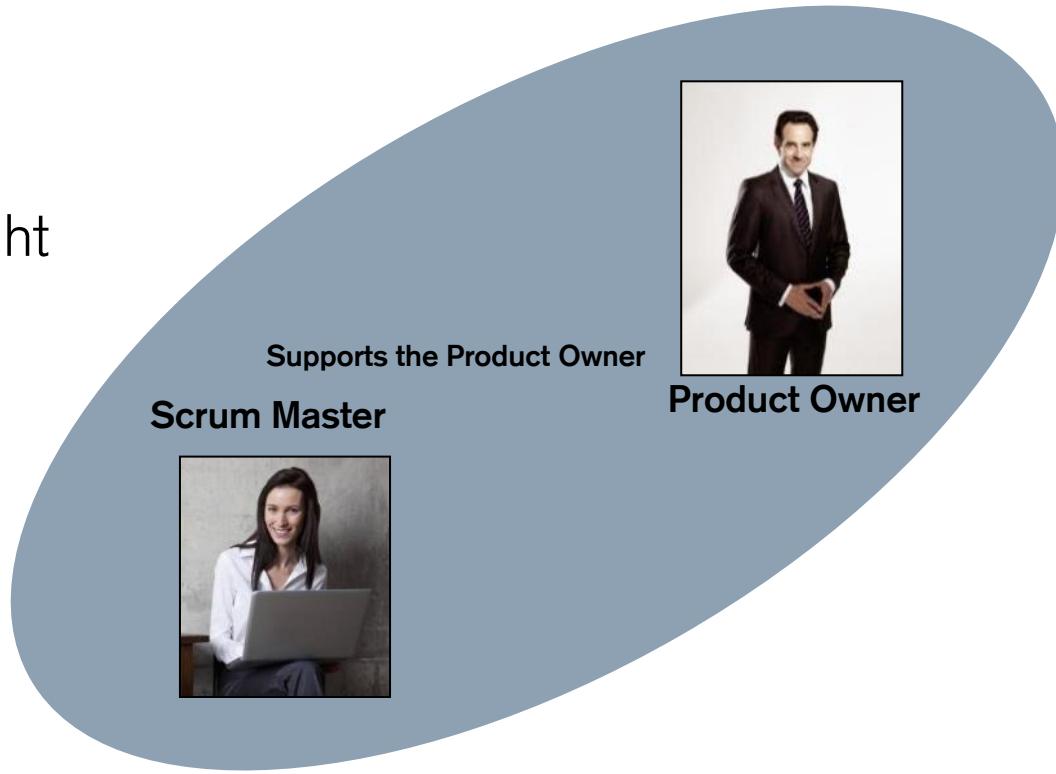
Supports the Product Owner

**Scrum Master**



## Working with the Product Owner

- Challenge the Product Owner
- Coach the Product Owner
- Support the Product Owner
- Communicate with the Product Owner
- Give the Product Owner the right information



## Working with Stakeholders and Observers

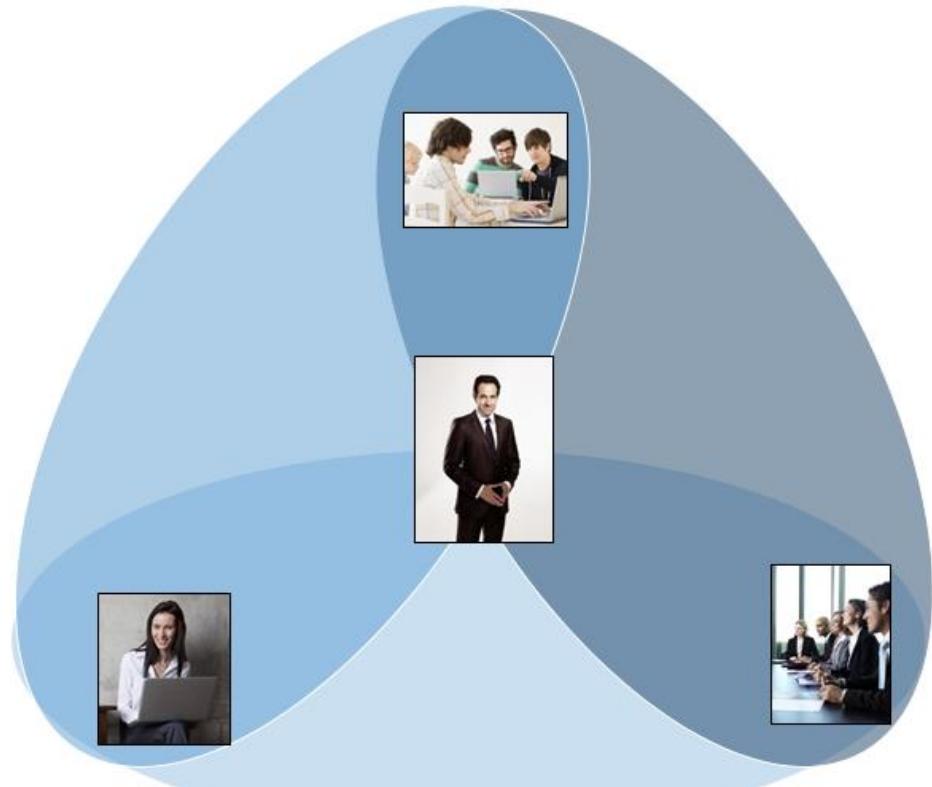
- Help and guide to set up the necessary environment
- Report status of current Sprints
- Monitor and manage escalations



## Working as Facilitator

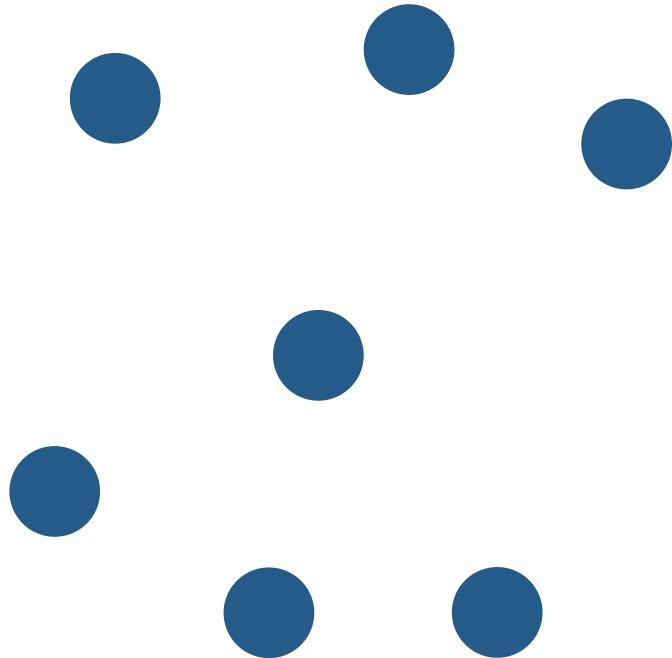
Act as a facilitator of communication during:

- Backlog refinement
- Sprint Planning
- Review & Demo Meeting
- Daily Standup
- Team Issues



# Performing Team

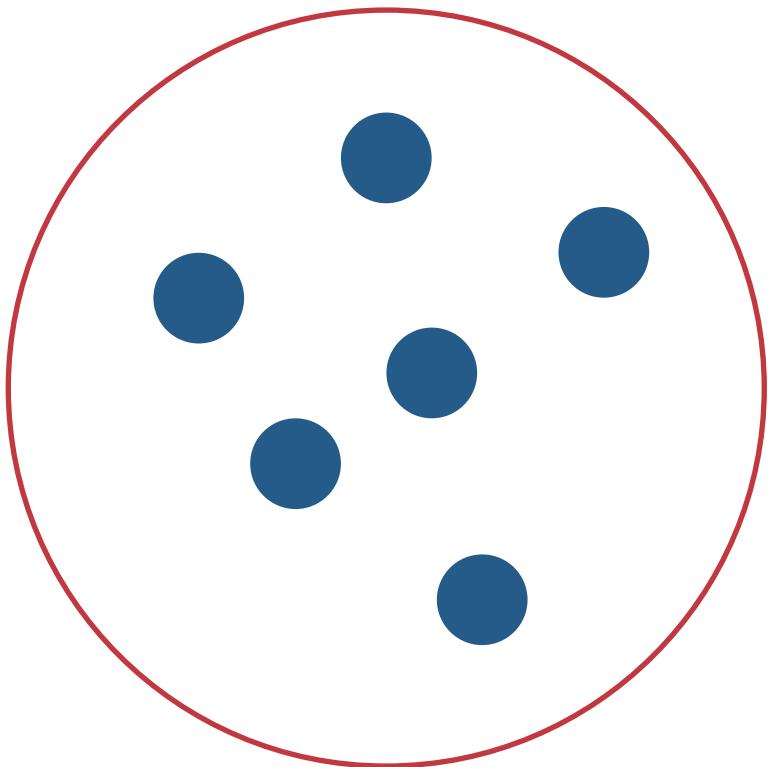
# What makes a group into a team?



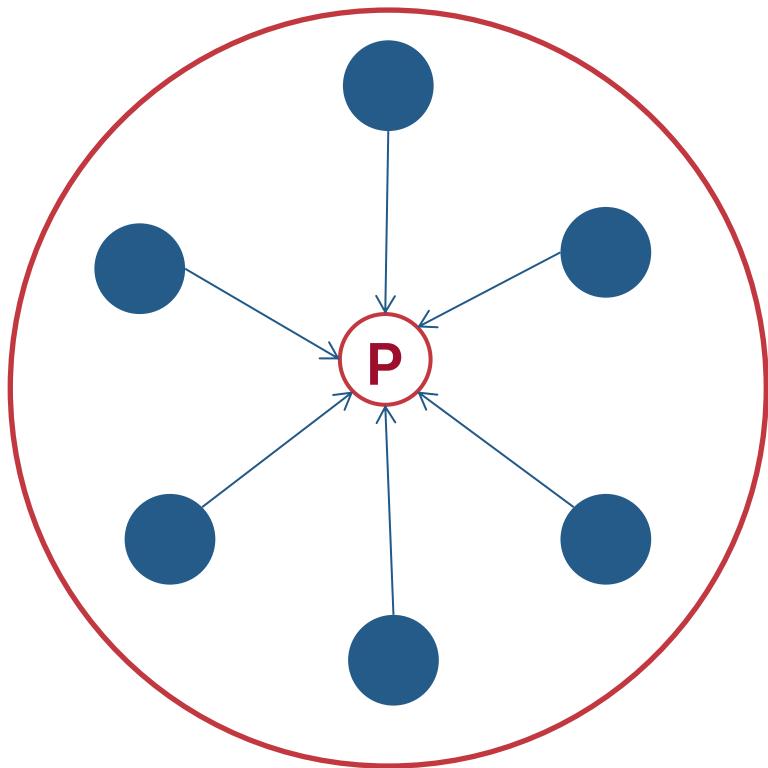
# What makes a group into a team?

## 1. Bounded membership

- All team members know who is on the team and who is not

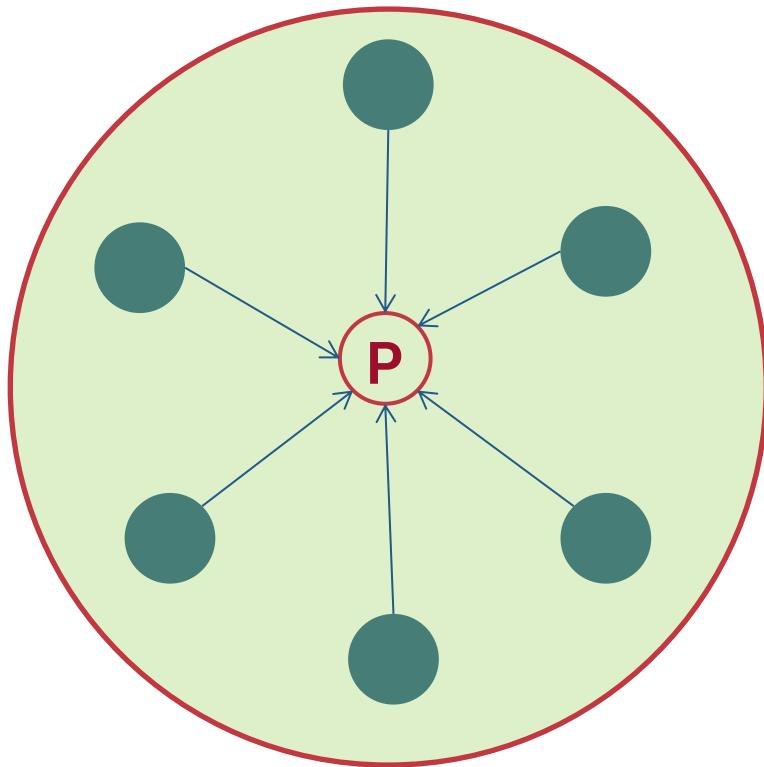


# What makes a group into a team?



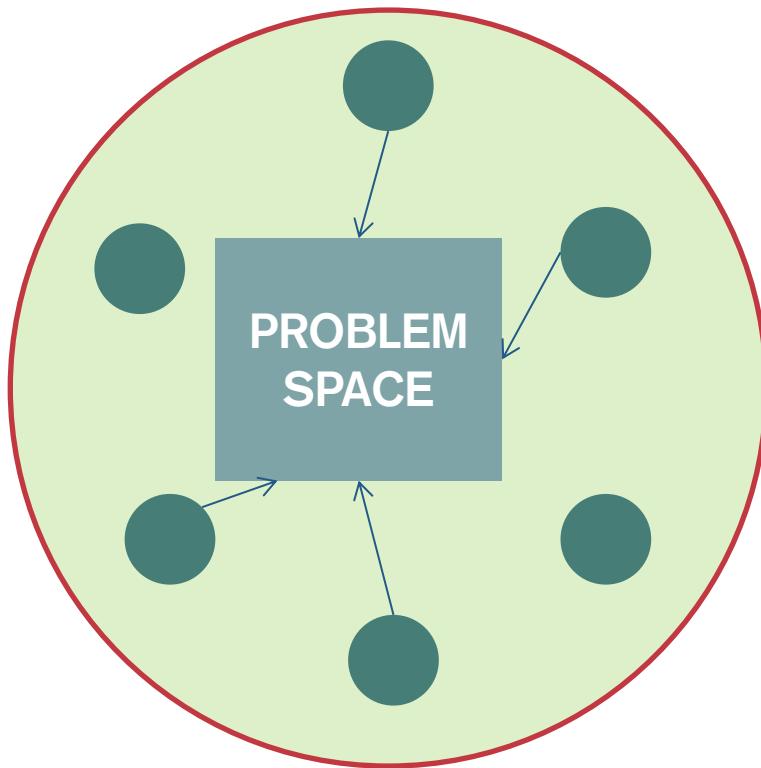
- 1. Bounded membership**
  - All team members know who is on the team and who is not
- 2. A shared 'core' purpose**
  - This goal requires joint contributions of the team and their mix of skills

# What makes a group into a team?



- 1. Bounded membership**
  - All team members know who is on the team and who is not
- 2. A shared 'core' purpose**
  - This goal requires joint contributions of the team and their mix of skills
- 3. Delineated authority**
  - Team must explicitly know breadth and depth of their authority

# What makes a group into a team?



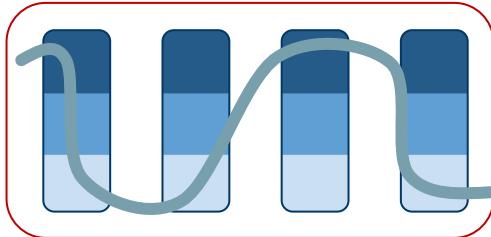
- 1. Bounded membership**
  - All team members know who is on the team and who is not
- 2. A shared 'core' purpose**
  - This goal requires joint contributions of the team and their mix of skills
- 3. Delineated authority**
  - Team must explicitly know breadth and depth of their authority
- 4. Team 'work'**
  - The work taken on by the team must be work that can only be done by 'joint contribution'. For *teamwork* do not give individual work

## Team Facts

- Teams are far more productive than the same number of individuals
- Face-to-face communication is extremely efficient
- Teams work best when not interrupted
- When teams themselves make a commitment, they will probably figure out how to meet it
- Peer pressure is far more efficient than manager pressure

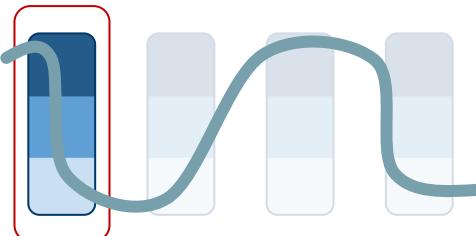


# Optimal Team Setups



## Feature team

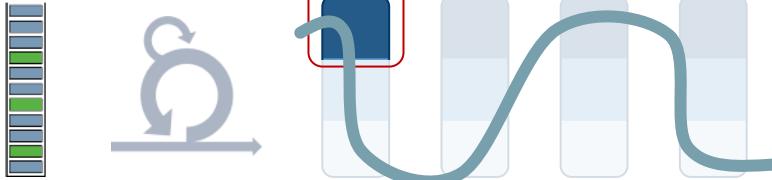
- The team has the skills and is enabled to implement changes across several applications
- e2e delivery of functionality across applications



## Application team

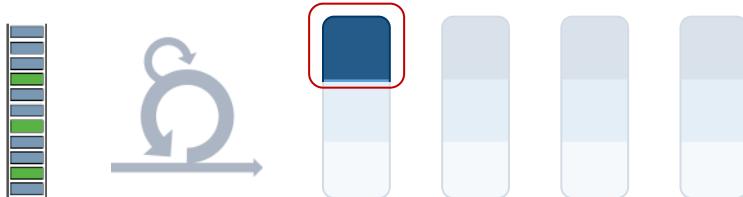
- The team can do the changes across all the layers in one application
- e2e delivery of functionality inside one application
- Contribution to e2e delivery of functionality across applications

## Non-optimal Team Setups



### Layer/component team

- The team can deliver parts of the functionality belonging to one layer or component only

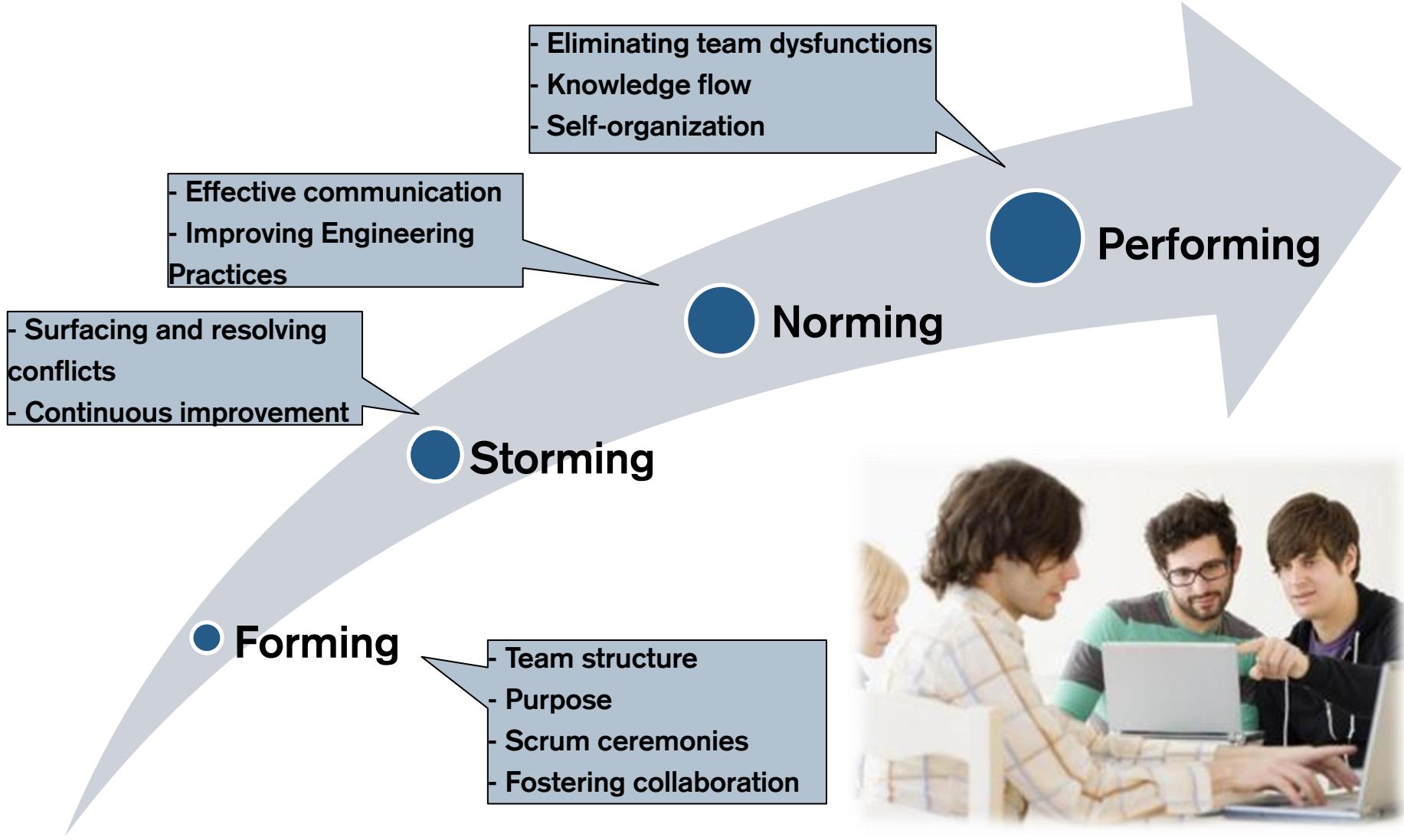


### Single discipline team

- The team can only achieve part of the work (i.e. Analysis team, QA team)

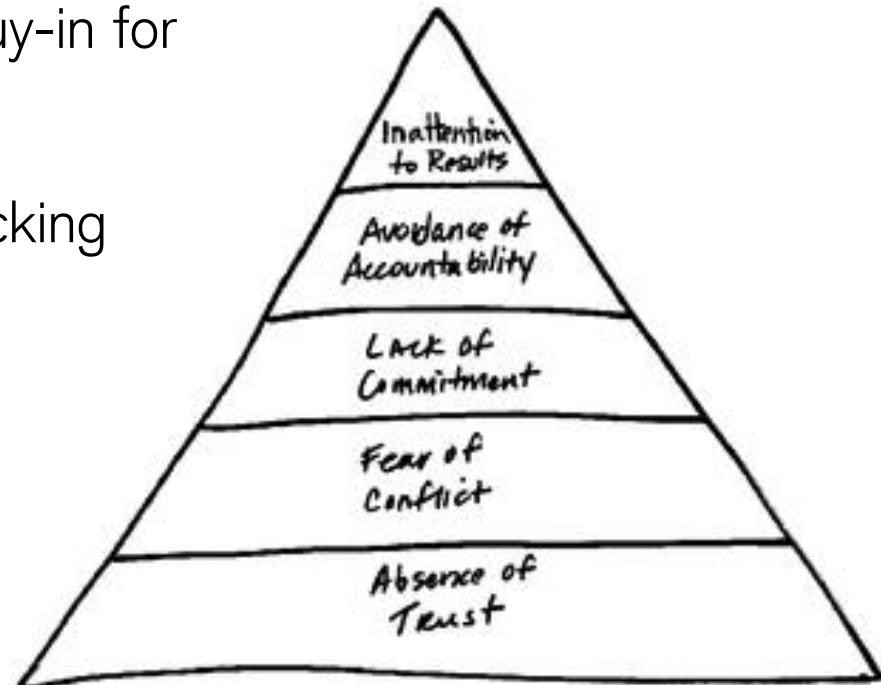
**In these configurations, each piece of work is/has a dependency on another team**

# Developing a High-Performance Team



## Team Dysfunctions

- **Absence of trust** – Unwilling to be vulnerable within the group
- **Fear of conflict** – Seeking artificial harmony over constructive passionate debate
- **Lack of commitment** – Feigning buy-in for group decisions creates ambiguity
- **Avoidance of accountability** – Ducking responsibility to call peers on counterproductive behavior
- **Inattention to results** – Focusing on personal success, status and ego

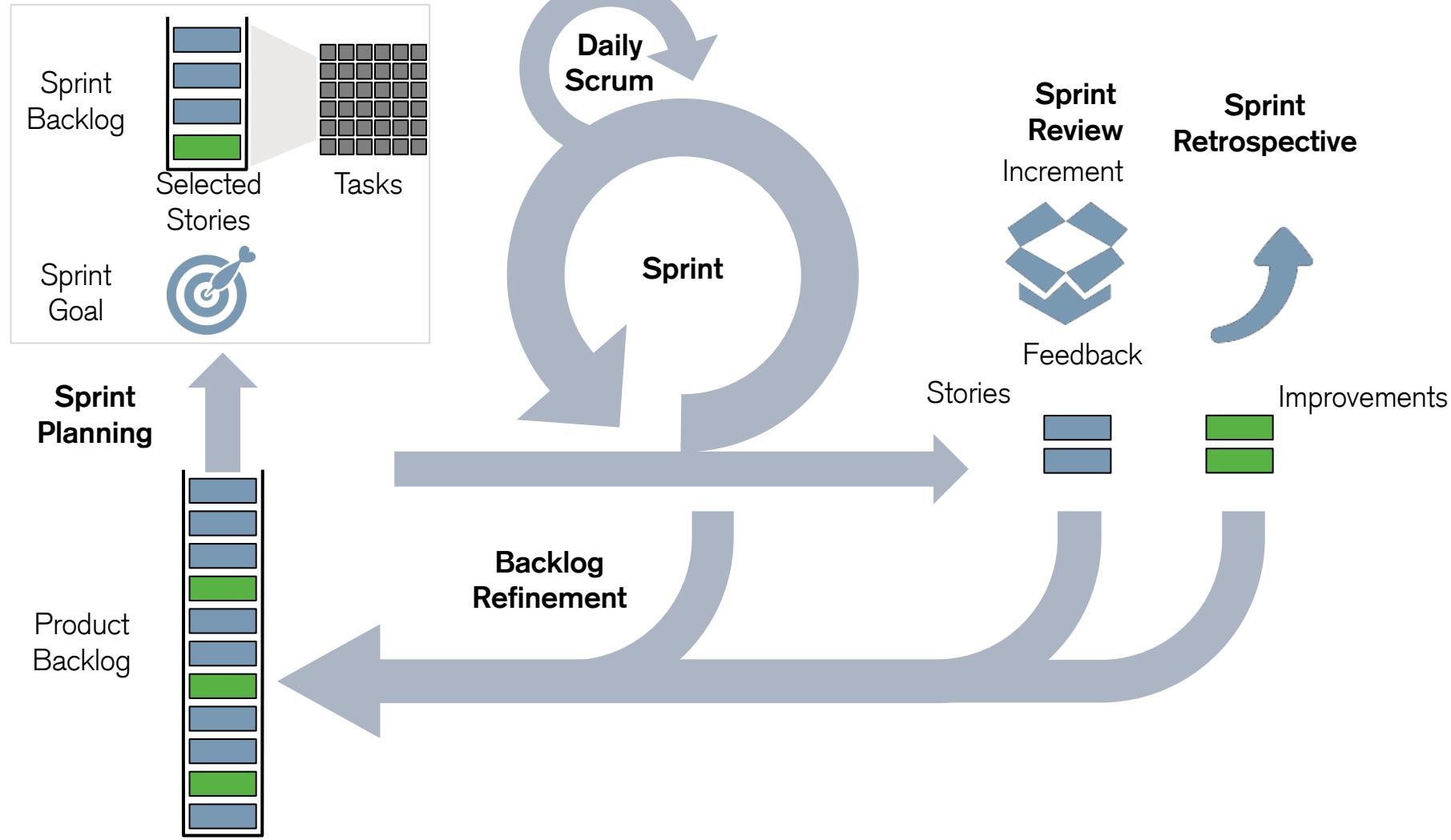




Sprint

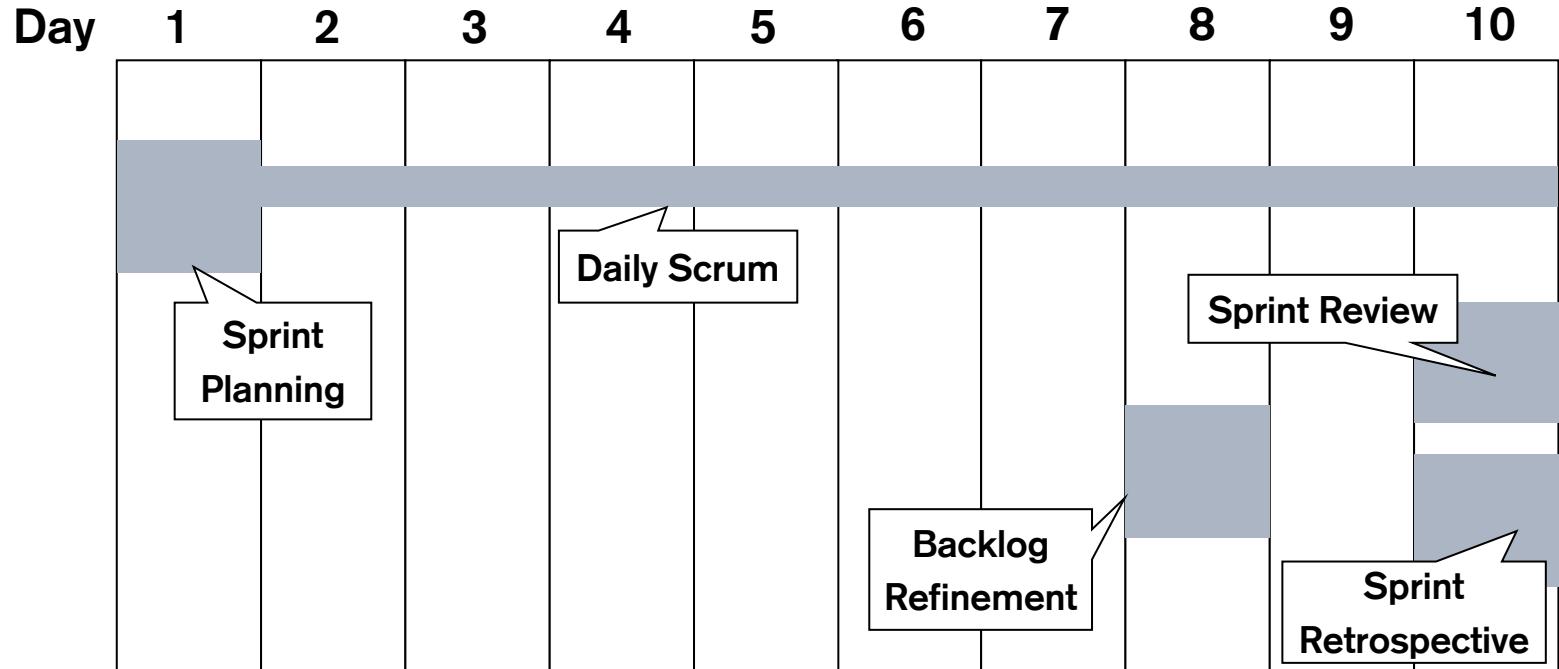
Sprint

# The Complete Cycle



Sprint

# Meetings in a calendar view

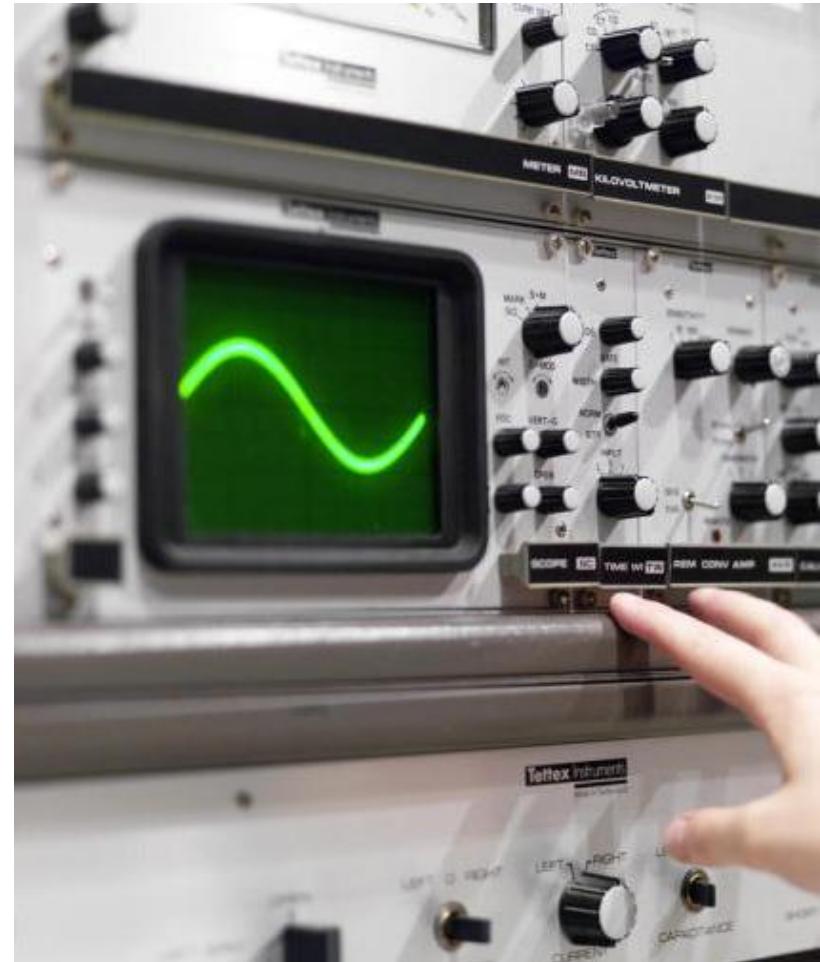


**Responsible  
to set up these  
meetings**

## Rhythm - The Heartbeat of Scrum

The Scrum Master establishes a regular rhythm within the team

- Establish strategic and tactical rhythms
- Keep rhythms regular and at a healthy pace
- Aim team rhythms to sustain focus over the long term
- Add predictability to activities



# Exercise



## Simulation Setup

## Setting the context

- Sprint structure – Ground rules
  - Sprint length = 15 min
  - Each Day = 4 min
  - Each Standup = 1 min
- Three days being simulated, each starting with a Standup meeting
- Decide your Scrum Master, rotate role every Sprint
- Scrum Master facilitates planning exercise and clarifications from PO
- Team selects a set of stories based on
  - a) Estimates (done earlier)
  - b) Team capacity (indicated by the 15' Sprint length)
- As acting PO, the trainer clarifies any story-level details

# Product: Scrum Quick Introduction Guide – A booklet

## Product Vision

For my project stakeholders who do not know CS-SDF Scrum;

the CS Scrum Quick Introduction Guide will be a source of information that explains the basic elements of CS-SDF Scrum such as *benefits, behaviors, roles, ceremonies and artifacts* in an easy-to-understand way;

So that I get more buy-in from our project stakeholders to adopt Scrum.



# Product: Airport

## Product Vision

For movie makers who have to shoot different airport scenes;

the Movie Maker's Airport Model is a 3D model which enables shooting different scenes like runway traffic, getting to and from the airport, war scenes, airport logistics at one place;

So that traveling costs can be lowered and scenes can be shoot faster;



## Product: Biotope

### Product Vision

For my biology teachers who teach lessons about co-habitation of different life-forms;

the Teacher's Biotope Model is a 3D training material which can be used to explain evolution, food chain, co-existence and collaboration between life forms;

So that a more interactive, interesting and memorable learning experience can be provided



## Product: Medieval castle

### Product Vision

For kids who have interested in Medieval history;

the Medieval 3D Castle is a 3D toy supporting learning about fortress defense, food and water supply, social life and trading;

So that they can learn about the society and engineering skills of medieval period as they are playing.

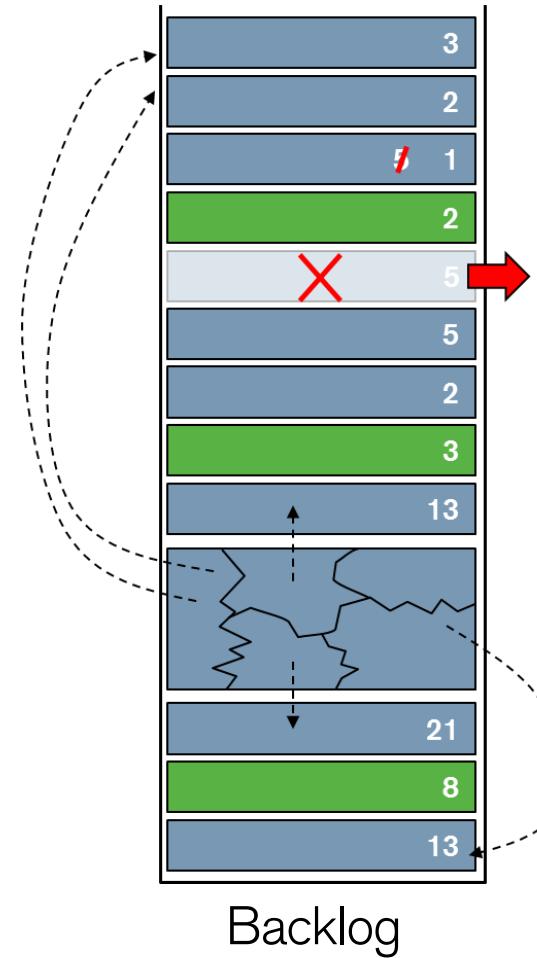


## Backlog refinement

Regular meeting

## Backlog Refinement

- Regular session to prepare items to be worked on
- PO + Team
- Support the Product Owner in creating high quality User Stories
- Scrum Master Facilitates the Backlog refinement sessions
- Ensures the quality of the Product Backlog, so the team is able to work on the User Stories

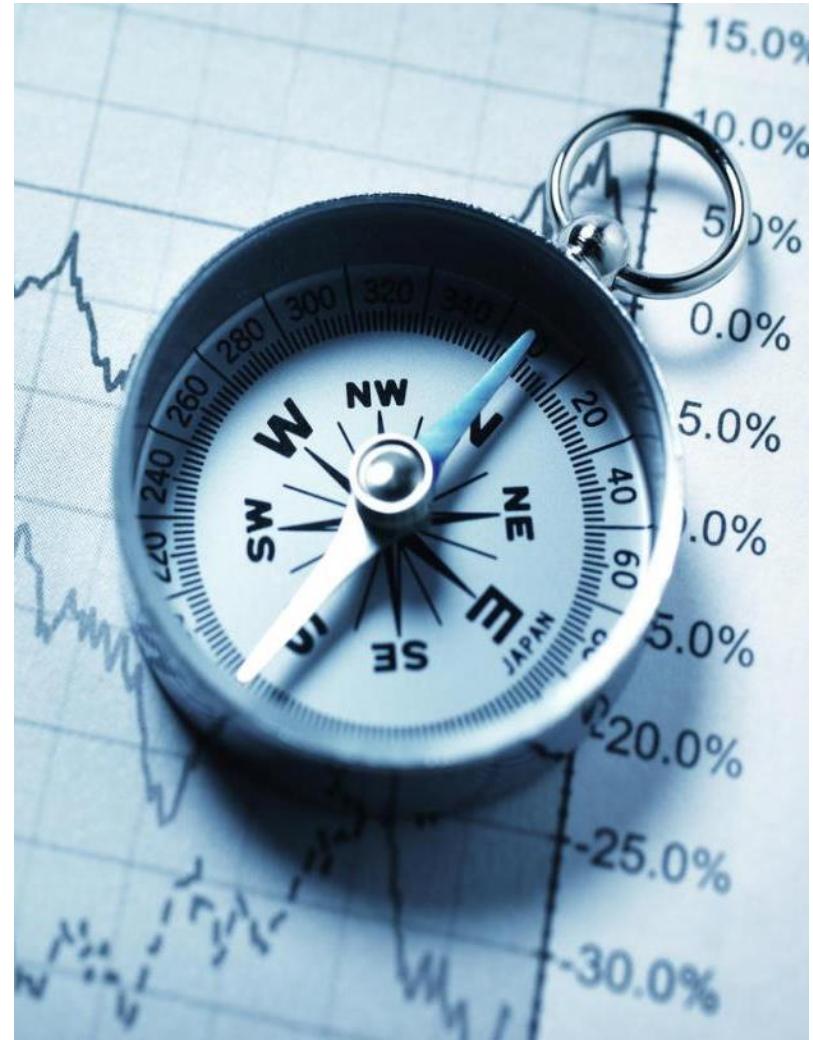


Regular meeting

## Discuss Backlog Content – Backlog Refinement

### ■ The Scrum Master

- Should ensure that the User Stories are ready for planning by facilitating regular Backlog Refinement sessions
- That all work is sized in story points before Sprint planning – This should be done during Backlog Refinement with the Team and Product Owner
- Ensures that the teams velocity is understood as this is a fundamentally important metric for planning



# Exercise



**Story Explanation**

# Exercise



**Story Estimation**

## Exercise – Estimation with Planning Poker Method

### Instructions:

- Split into teams of 4-5 people
- Utilize the User Stories provided
- Assume the smallest Story as 1 Story Point (split Stories as needed)
- Estimate User Stories using Planning Poker



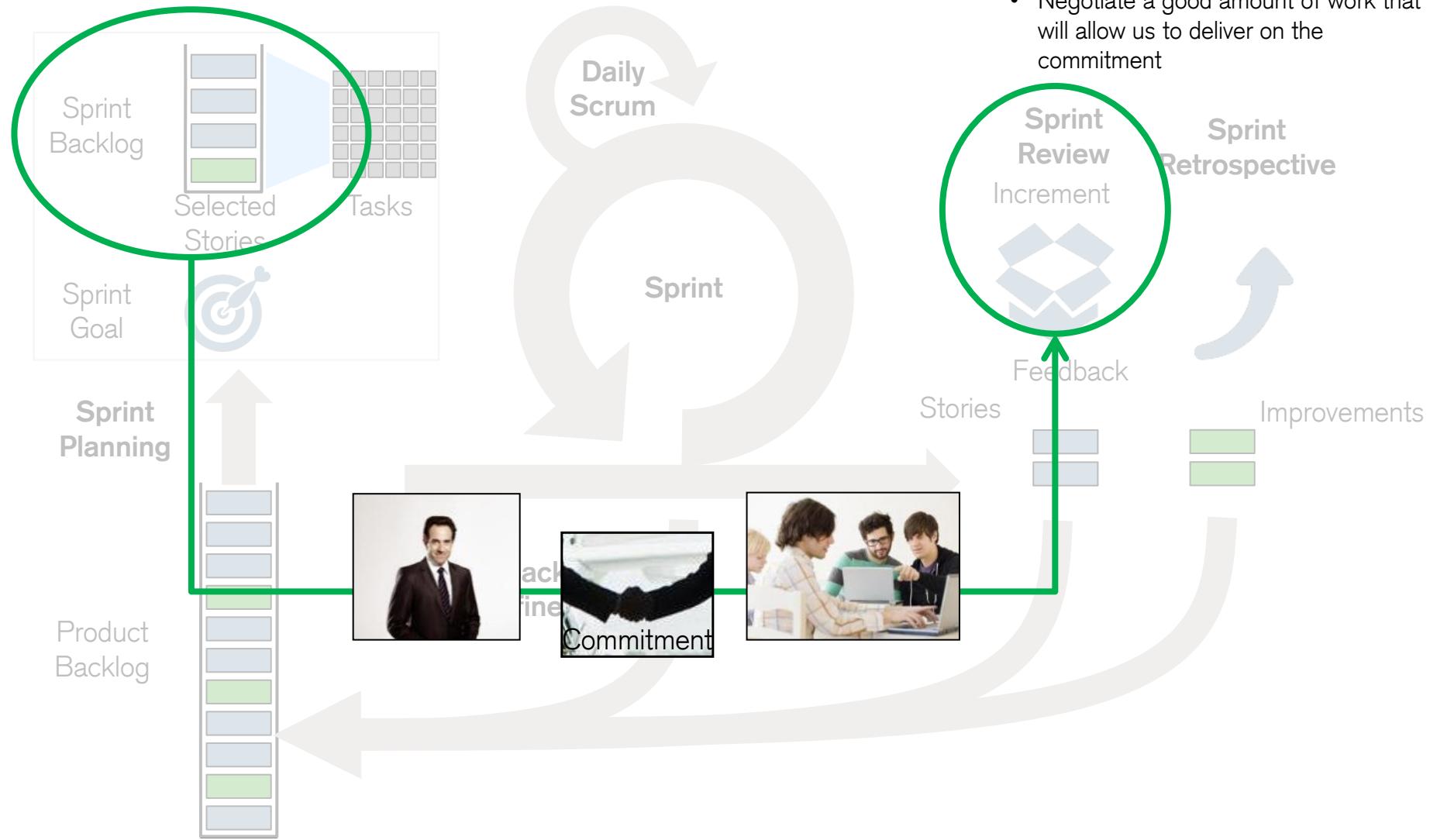
# Sprint Planning

## The desired Outcomes are

- The team understands what is requested by the Product Owner
- The team and Product Owner reach agreement on the Sprint content
- The team breaks the work down into tasks with estimates
- The team understands its capacity
- The team make a commitment to the Product Owner on the Sprint deliverables
- The Sprint goal is defined and agreed with the Product Owner

## Sprint Planning

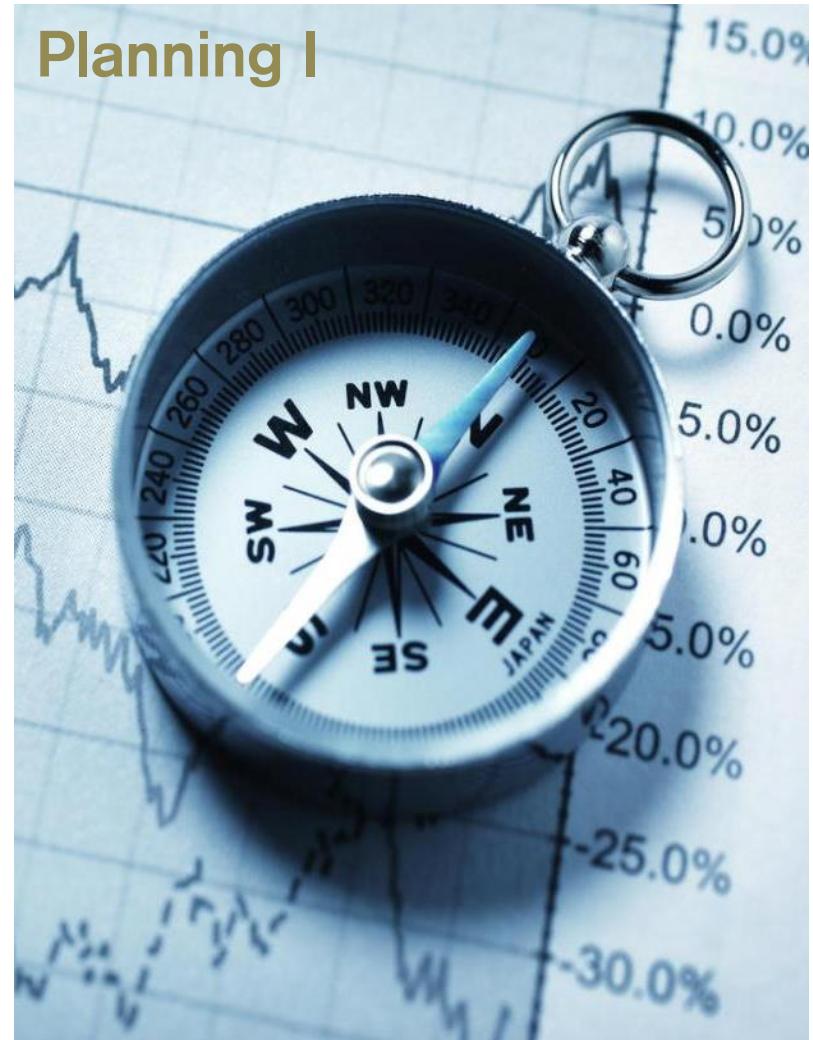
# Goal of Sprint Planning



## Sprint Planning

# Discuss Sprint Content – Meeting

- The Scrum Master
  - Organizes and facilitates the meeting
  - Moderates the meeting
  - Challenges the team
  - Encourages individual team members to raise questions
  - Mediates between the team and the Product Owner



Scrum Simulation

## Exercise

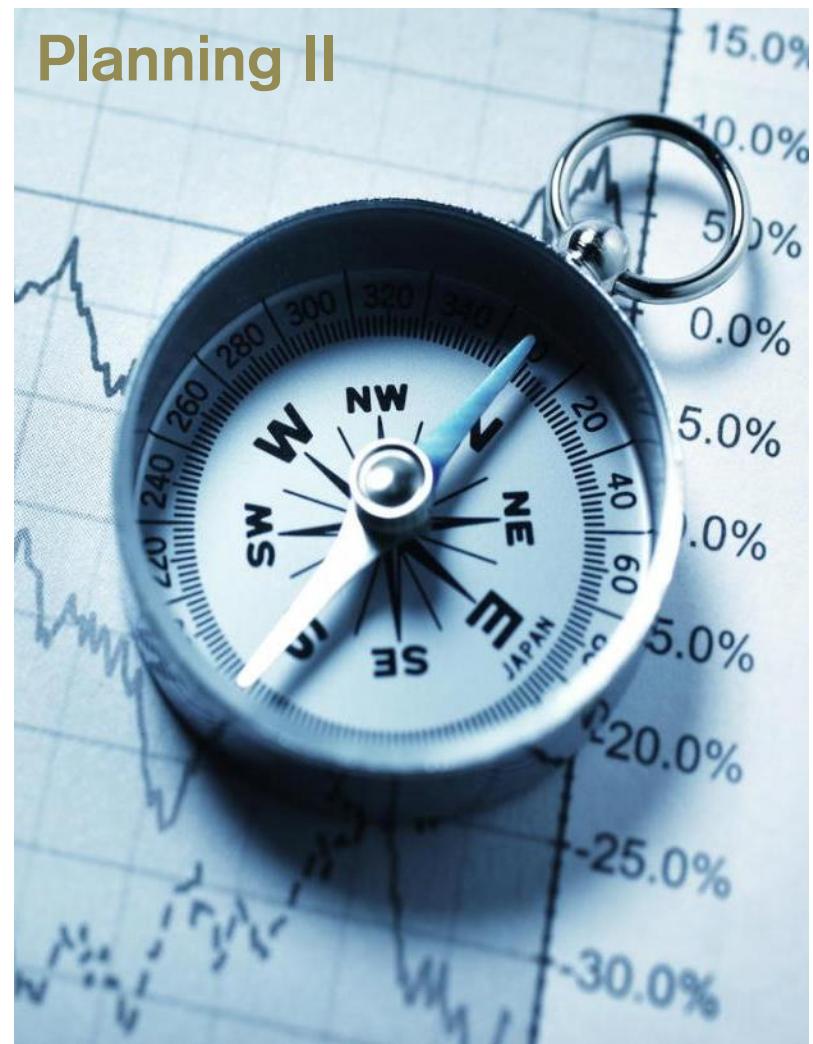


**Discuss Sprint Content**

## Sprint Planning

# Breakdown Stories for Sprint – Meeting

- The Scrum Master
  - Balances the “voices” of individual team members
  - Ensures the Stories are correctly broken down into tasks
  - Ensures the tasks cover all work, end-to-end
  - Encourages each team member to offer an opinion



# Commitment

- The Scrum Master
  - Ensures the team accepts the Sprint Backlog and that they reach consensus that they can deliver the Sprint
  - Validates that the team has a full understanding of the workload
  - Keeps the Product Owner from including additional workload
  - Ensures that a Sprint Goal is in place



## Exercise

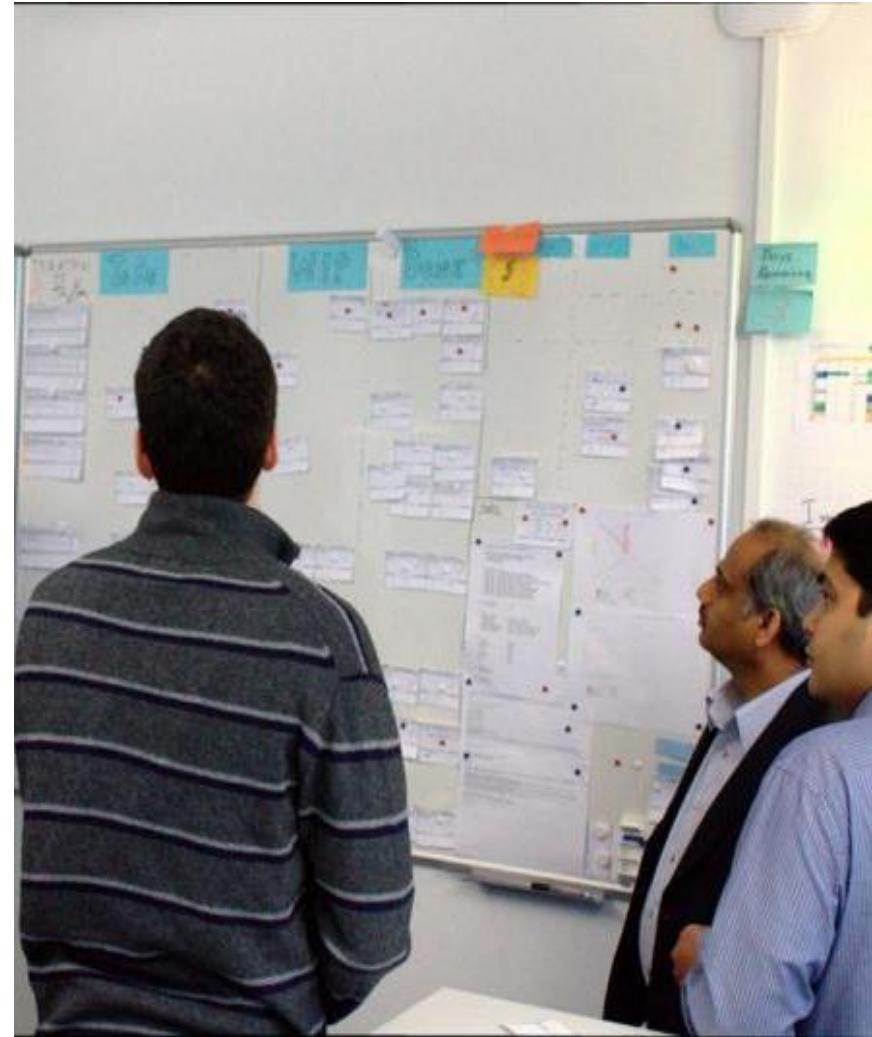


**Breakdown Stories for Sprint**

## Set up the Task Board

### ■ The Scrum Master

- Ensures the task board is available
- Sets up the basic structure
- Trains the team
- Ensures the team uses the task board as intended
- Helps maintain the task board utility
- Cleans up the task board as needed

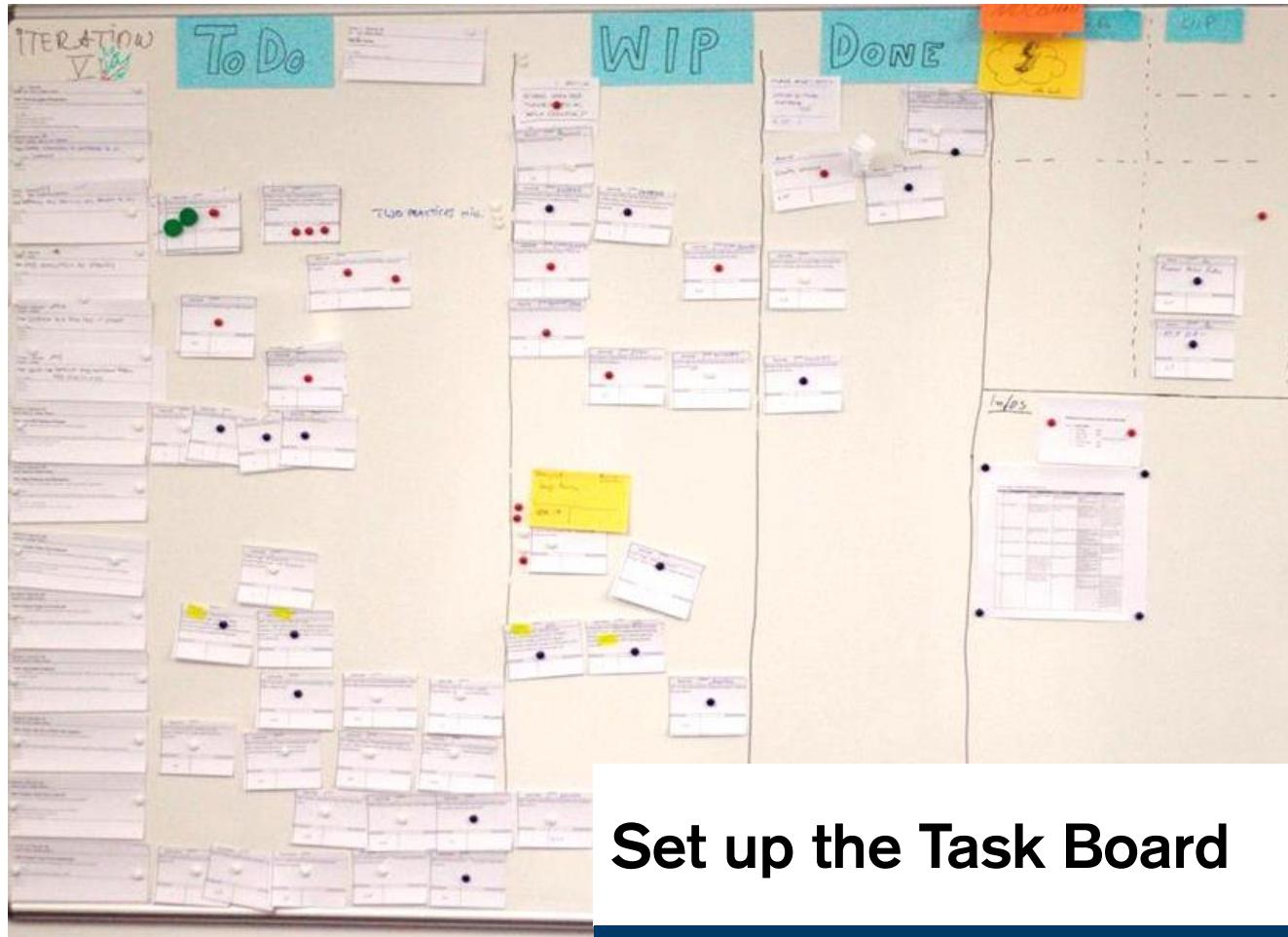


## Sprint Planning

# Example: Task Board

User Stories	Task Cards		
	To Do	WIP (Work In Progress)	Done
<p><b>As a [Some role]</b> <b>I want to [Something]</b> <b>so that [Some benefit]</b></p>			
<p><b>As a [Some role]</b> <b>I want to [Something]</b> <b>so that [Some benefit]</b></p>			
<p><b>As a [Some role]</b> <b>I want to [Something]</b> <b>so that [Some benefit]</b></p>			

# Exercise



## Set up the Task Board

# Motivation

## What motivates us – the video



**The surprising truth about what motivates us**

<http://www.youtube.com/watch?v=u6XAPnuFjJc>

# Purpose, autonomy and mastery motivate us

## ■ Purpose

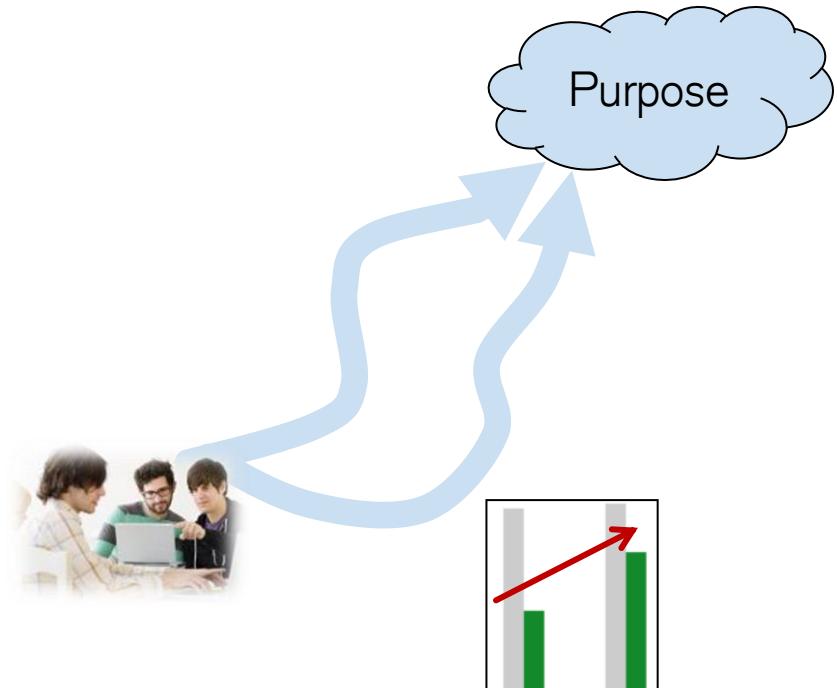
- Understanding the big picture,  
what's the aim and why

## ■ Autonomy

- Having the freedom to figure out  
the way to the goal, how to do  
things

## ■ Mastery

- Developing, improving skills all the  
time



# The right mindset, skillset and environment is needed

## ■ Willing

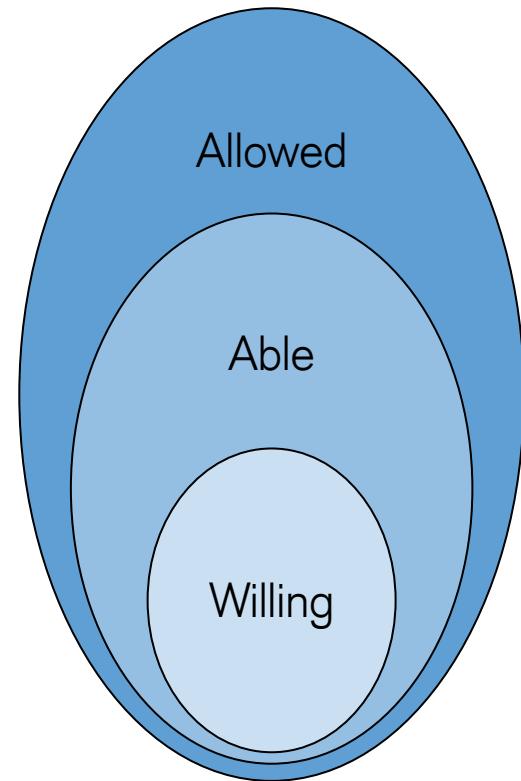
- Willingness to do more, do something different, ...
- Comes from the individual

## ■ Able

- Skill gaps have been closed to enable performing
- Measures are worked out together between the individual and coach/mentor/manager

## ■ Allowed

- An environment is needed where one can, is allowed to perform
- Task of the coach/mentor/manager to ensure one



Motivation

## Exercise



**Scrum & motivation**



**Daily Business**

## The desired Outcomes are

- Identify new impediments that need to be resolved
- Enable alignment of the team members
- Create a frequent opportunity for team members to identify where they can support each other in executing their tasks
- Reduce the risk of team members working on the wrong tasks
- Manage Risks and Issues in an integrated way with other work

# Daily Standup

## ■ The Scrum Master

- Sets the schedule of the daily routine
- Facilitates the Daily Standup
- Ensures contribution by everyone during the meeting
- Interrupts lengthy discussions and/or high emotions
- Collects new impediments and gives an update about the current status

1

What did you do **yesterday** to help the team meet the Sprint Goal?

2

What will you do **today** to help the team meet the Sprint Goal?

3

What's in your way?

Scrum Simulation

## Exercise



**Day 1: Standup Meeting**

# Exercise



**Day 1: Work the Stories**

## Daily focus points

- Develop collaborative culture, manage conflicts
- Promote swarming
- Develop multi-skilled people
- Enhance soft and hard skills
- Ensure the same information is available to everyone
- Encourage experimenting and learning with short feedback loops



## Exercise



**Day 2: Work the Stories and Daily Standup**

## Exercise



**Day 3: Work the Stories and Daily Standup**

## Sprint Demo and Review

## The desired Outcomes are

- Reduce uncertainty in what is being built
- Confirm and agree whether a work item is "done"
- Customer acceptance is achieved
- Delivery is more likely to be aligned with business needs
- Supplier agreements more likely to be met
- Business and project team are aligned, and trust each other

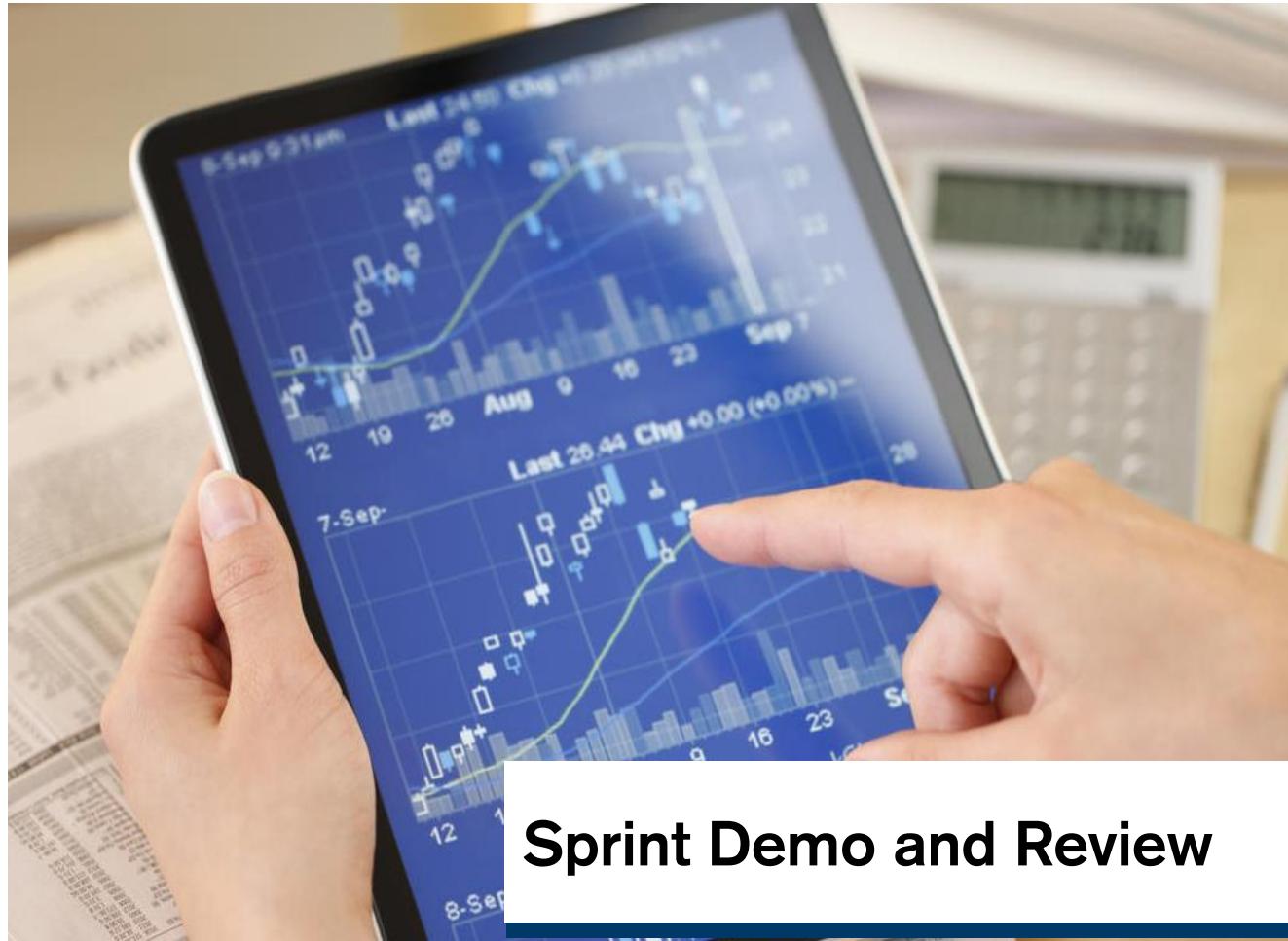
## Review & Demonstration

### ■ The Scrum Master

- Ensures availability of the Product Owner and team members
- Schedules and prepares for the review meeting
- Ensures the team generate the agenda
- Ensures items are time-boxed
- Keeps focus on key areas
- Facilitates on-going reviews



## Exercise



**Sprint Demo and Review**

# Retrospective



# Open Communication is Critical Closed Communication is Destructive

Even bad news is good news

## The desired Outcomes are

- Continuous improvement by identifying actions a team can take to improve
- Common understanding amongst the team on what happened
- Identify team impediments
- Identify organizational impediments
- Acknowledge what is going well
- Identify strengths
- Retain project experience

# Continuous improvement is change management

## Enabling change requires

- **Leadership** – commitment & coaching to make it happen
- **Sense, need to change** – everyone must understand why to change, leave the comfort zone
- **Vision** – the future state has to be shaped and accepted by those affected
- **Short term wins** – help to prove that the selected direction, path is the right one
- **Role models** – the new behavior, mindset has to be established from inside the group
- **Resistance** – must be understood and dealt with
- **Support** – training, communication, assessment help to get to the future state

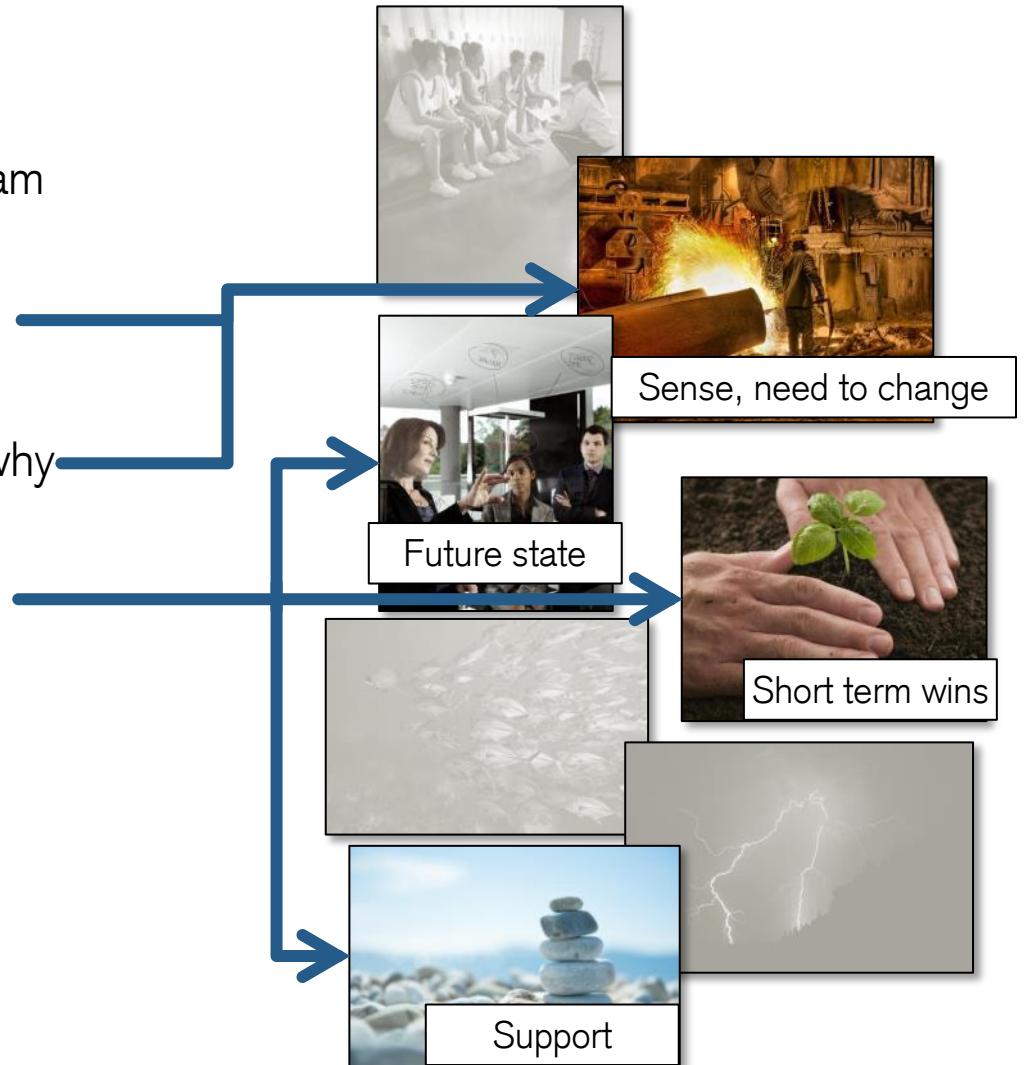


# Well structured retrospective enables change

## Structure of the retrospective

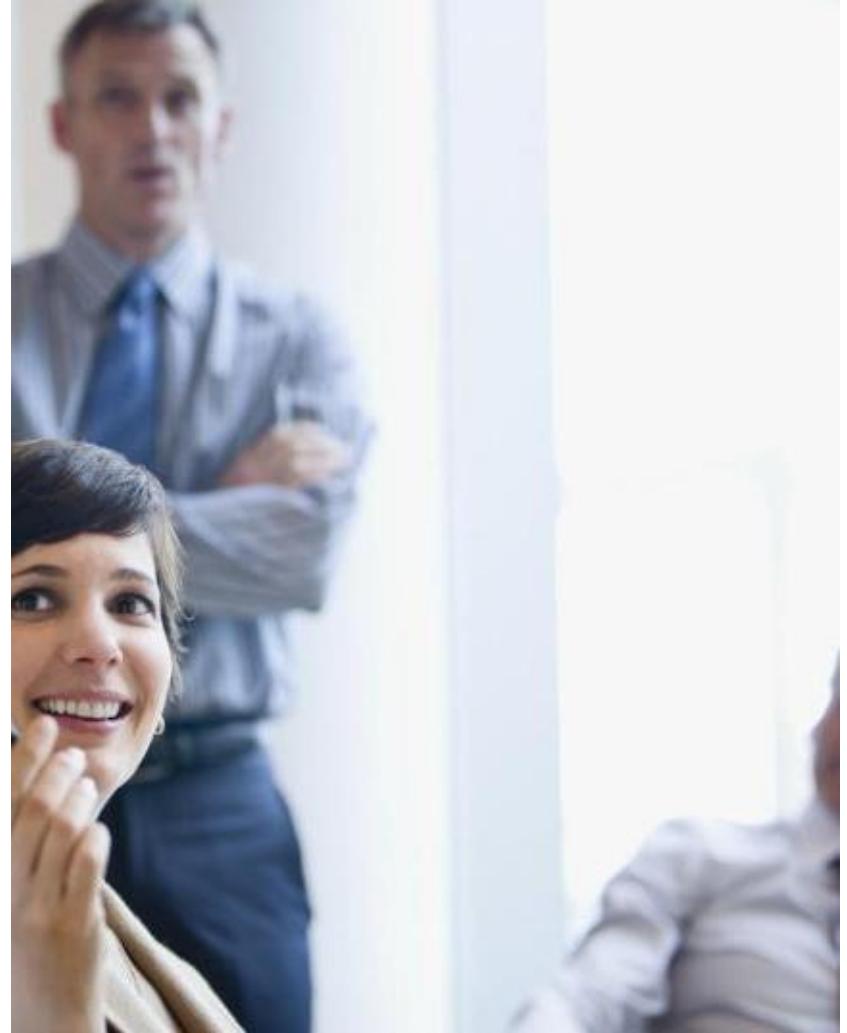
- **Setting the stage** – preparing the team for the retrospective
- **Gather data** – understanding what happened
- **Generate insights** – understanding why it happened
- **Decide what to do** – understanding where to go
- **Closing**

A Scrum Master should know different techniques, activities for every phase



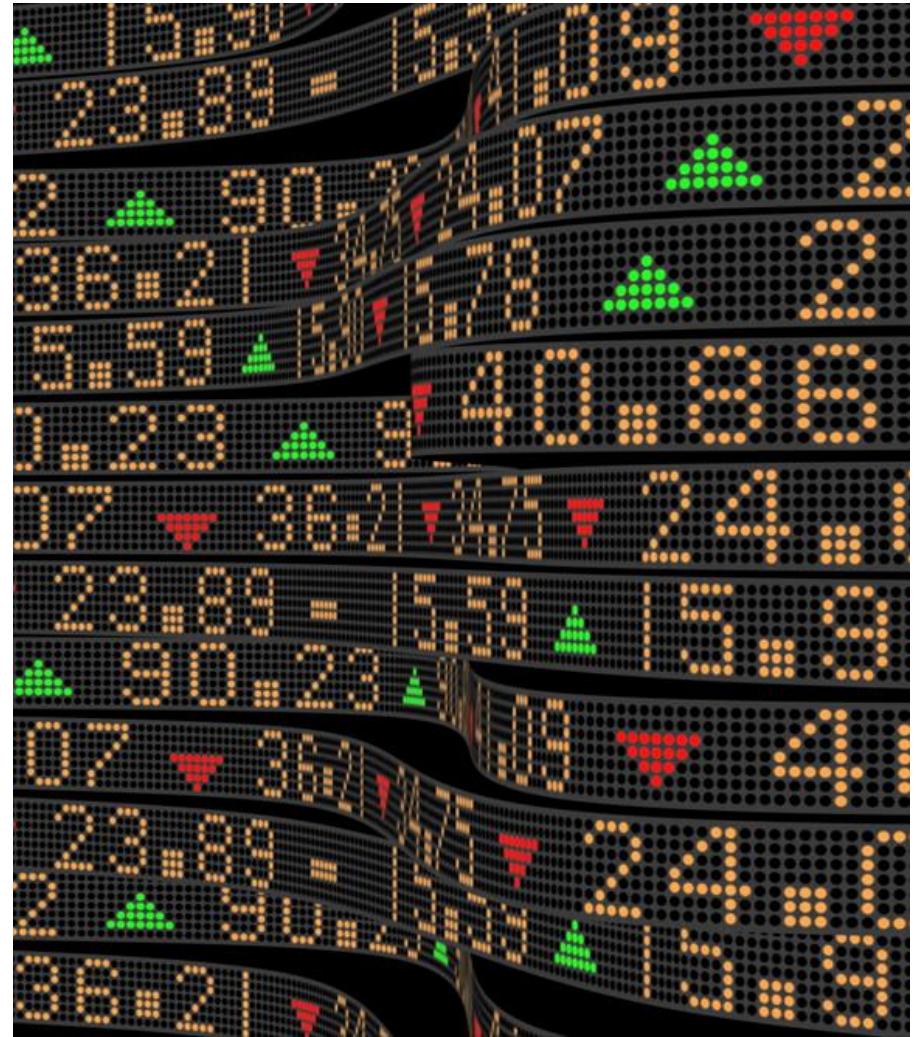
# Setting the stage ensures that everyone is “present”

- **Welcoming** participants
- **Check in** (increase participation, i.e. by asking everyone to say something)
- Explaining the **agenda**
- **Verify** the **status** of previous **improvements**
- **Provide sprint data** (i.e. burn-down chart, velocity chart,...)



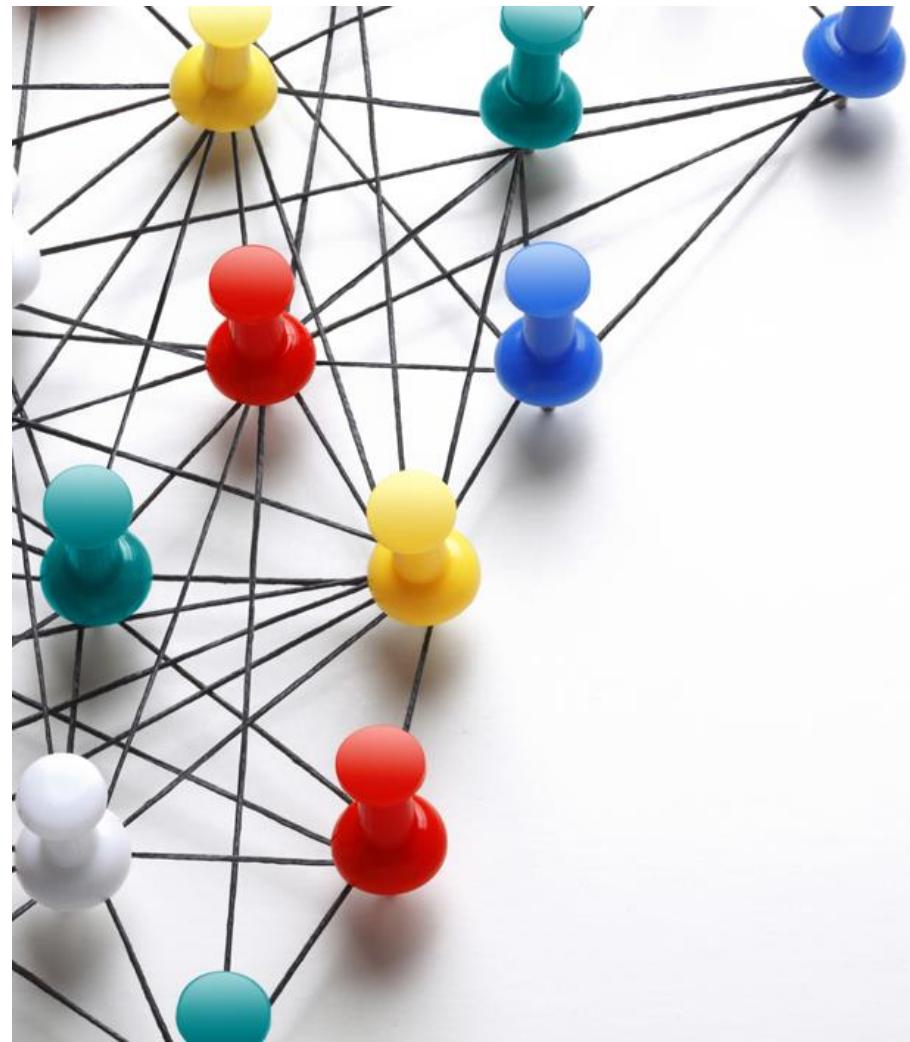
## Gathering data creates a common starting point

- Collecting data helps to understand what and how big the problem(s) is
- The whole team becomes aware of the need to change
- Time line, Team Radar, Triple Nickels, Satisfaction Histogram



## Root causes are identified during insights generation

- Problems cannot be solved without the root causes being identified
- The activity helps the team to further increase the understanding for the needed change
- Five whys, Fishbone, Patterns and Shifts



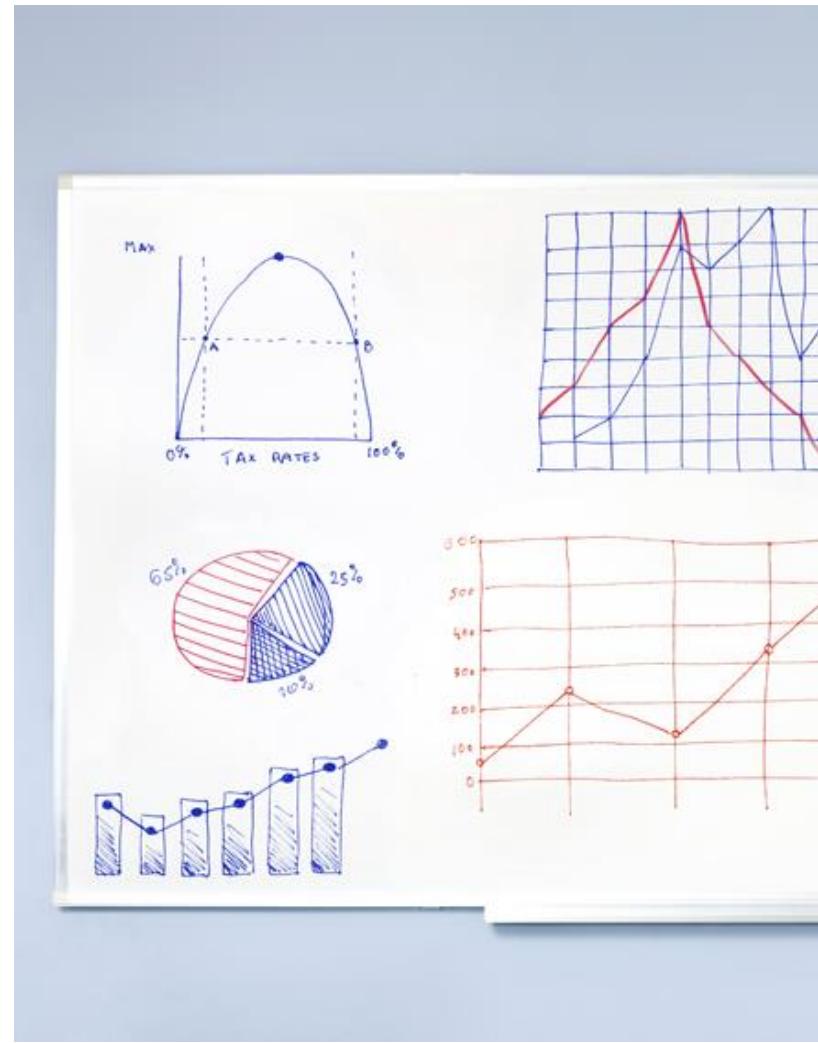
## Deciding what to do lines up the team for the future state

- The needed improvements are identified
- Improvements are described and put into the backlog
- By identifying improvements the team owns the future state
- Stories describing the future state, SMART goals, Short subjects



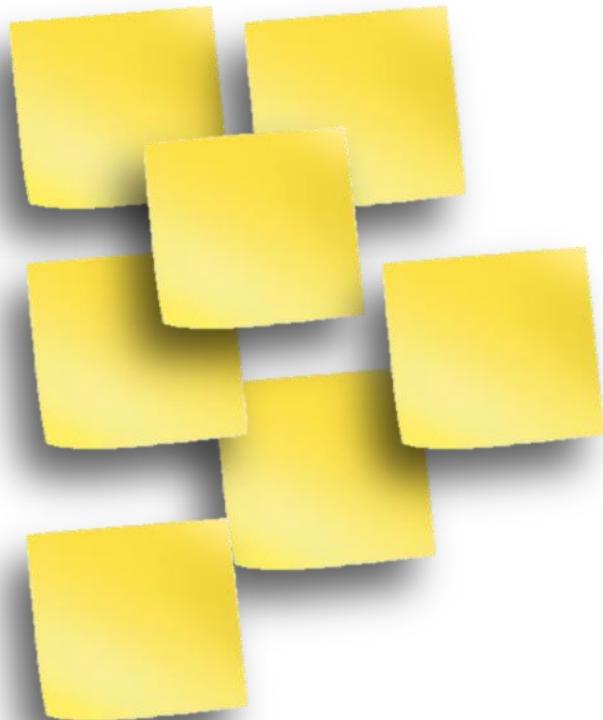
# Retrospective **Closing**

- Retrospective on the retrospective
- Thanking for participation
- Return on Time Invested, +/Delta, Appreciation

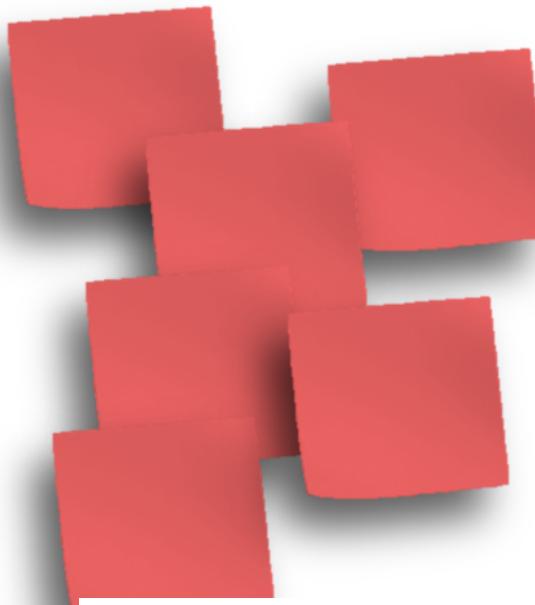


## Exercise

What went well?



What could be improved?



**Retrospective**



Track Progress

## Daily Business

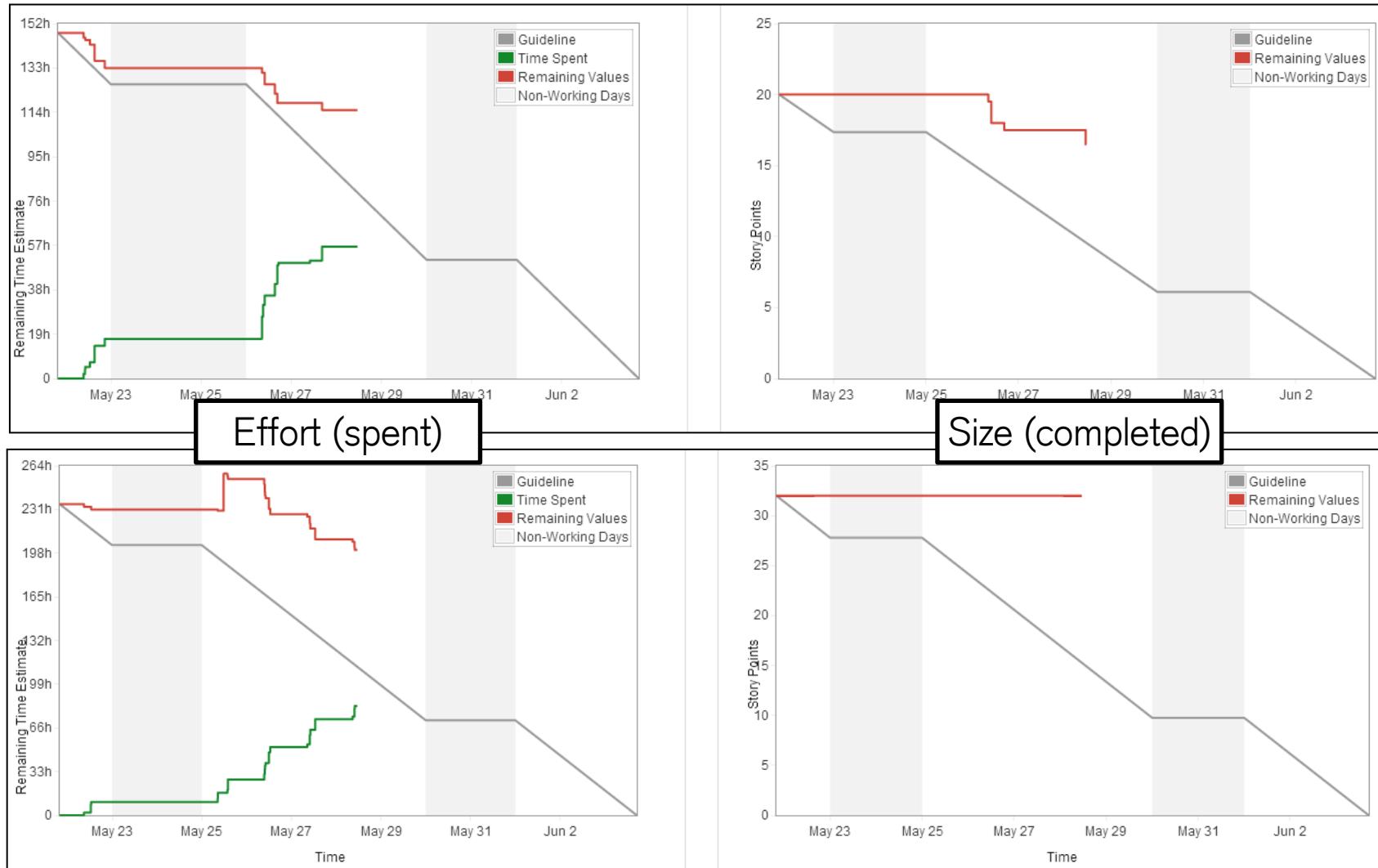
### ■ The Scrum Master

- Creates and updates the burn-down chart
- Posts the burn-down chart
- Shows open/closed impediments
- Monitors if the team is committed to the Sprint backlog
- Ensures the team joins the Daily Standups
- Checks that retrospective activities are worked on



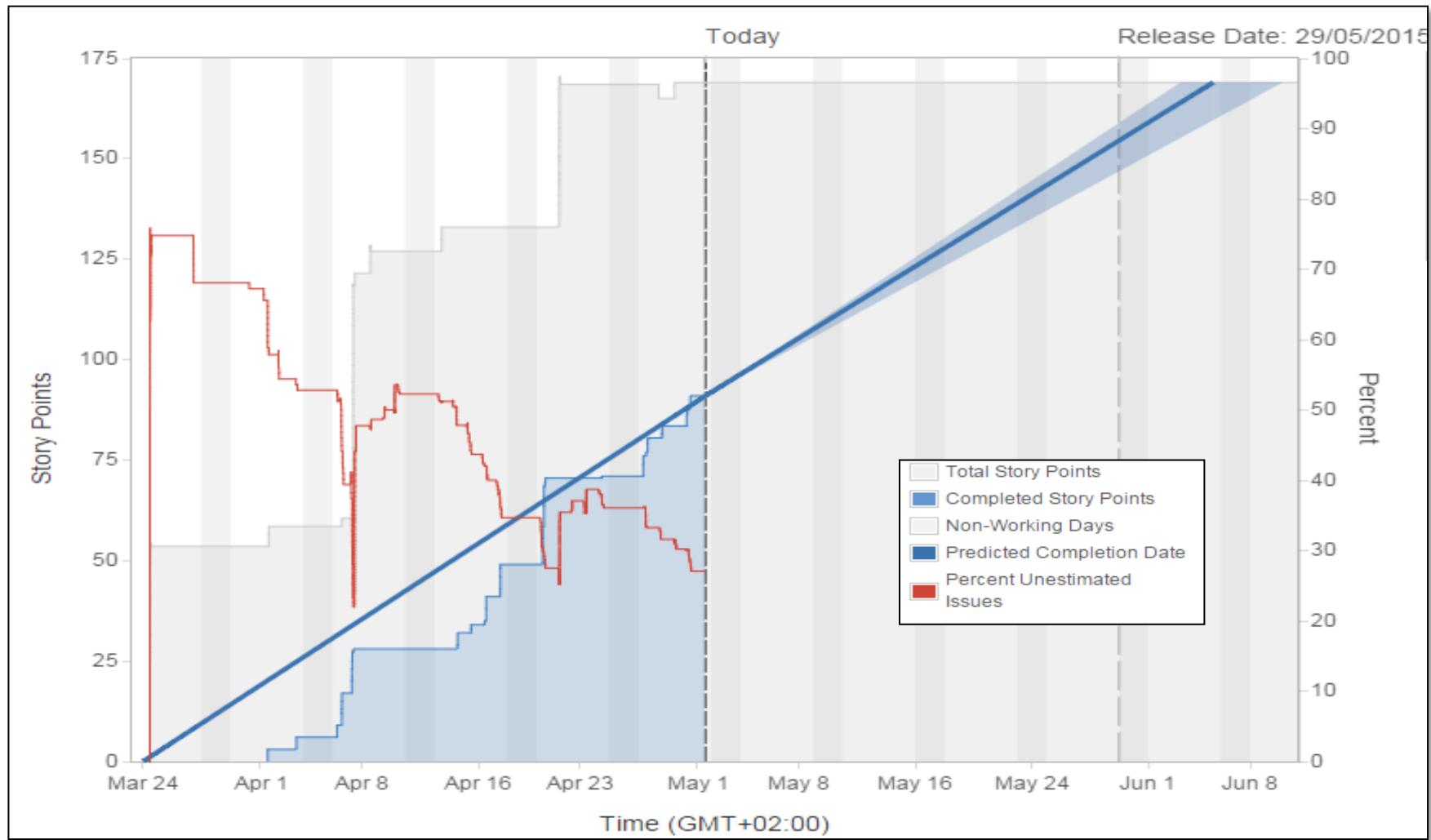
## Track Progress

# Sprint Burn-Down Charts (JIRA)



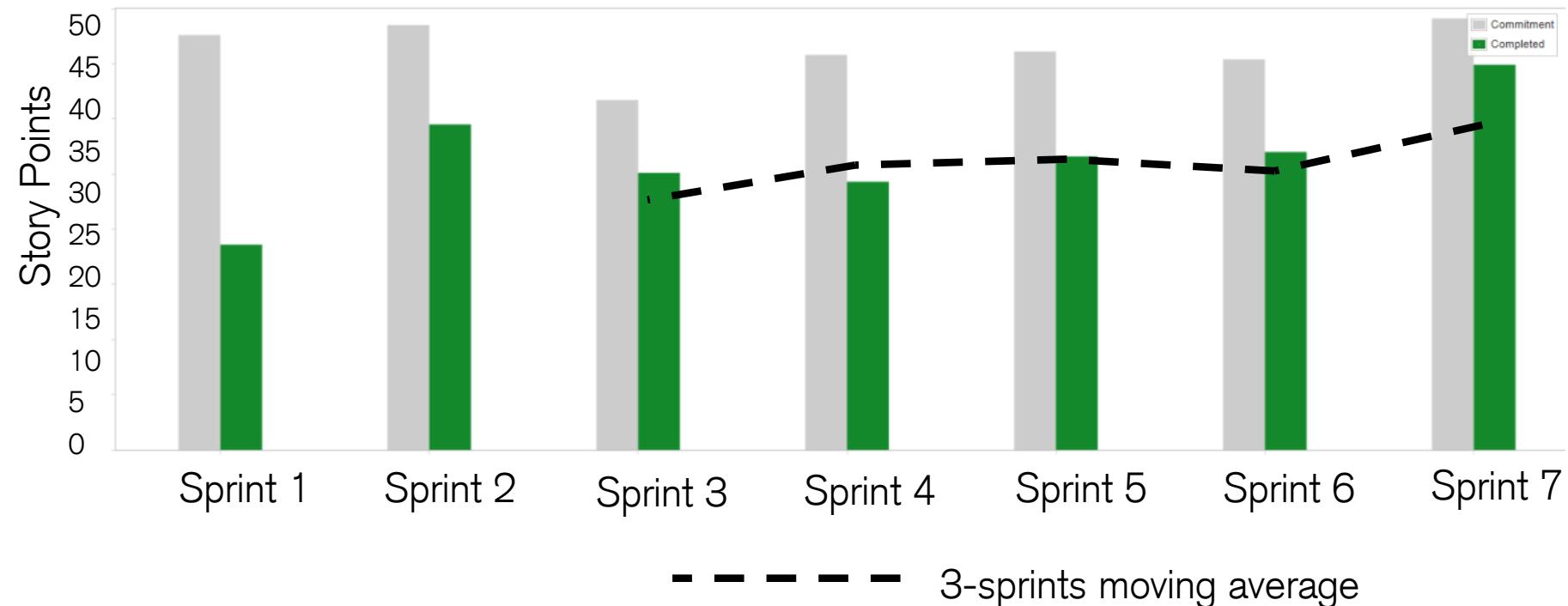
Track Progress

# Release Burn-up Chart (JIRA Version Report)



Track Progress

## Velocity (JIRA)



Scrum Simulation

## Exercise



**2<sup>nd</sup> Sprint Start**

# Exercise



**2<sup>nd</sup> Sprint End**

# Why is Velocity Important?



- Velocity helps to identify how much work we can plan to accomplish within a time-box
- Velocity allows us to make commitments based on our past performance

# Back to the Principles

## Agile Manifesto - Principles

1. Our **highest priority is to satisfy the customer** through **early and continuous delivery of valuable software**.
2. **Welcome changing requirements**, even late in development. Agile processes harness change for the customer's competitive advantage.
3. **Deliver working software frequently**, from a couple of weeks to a couple of months, with a preference to the shorter timescale.
4. **Business people and developers must work together daily** throughout the project.
5. Build projects around **motivated individuals**. Give them the environment and support they need, and **trust them** to get the job done.
6. The most efficient and effective method of conveying information to and within a development team is **face-to-face conversation**.
7. **Working software is the primary measure of progress**.
8. Agile processes promote **sustainable development**. The sponsors, developers, and users should be able to **maintain a constant pace indefinitely**.
9. 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.

Back to the Principles

## Exercise

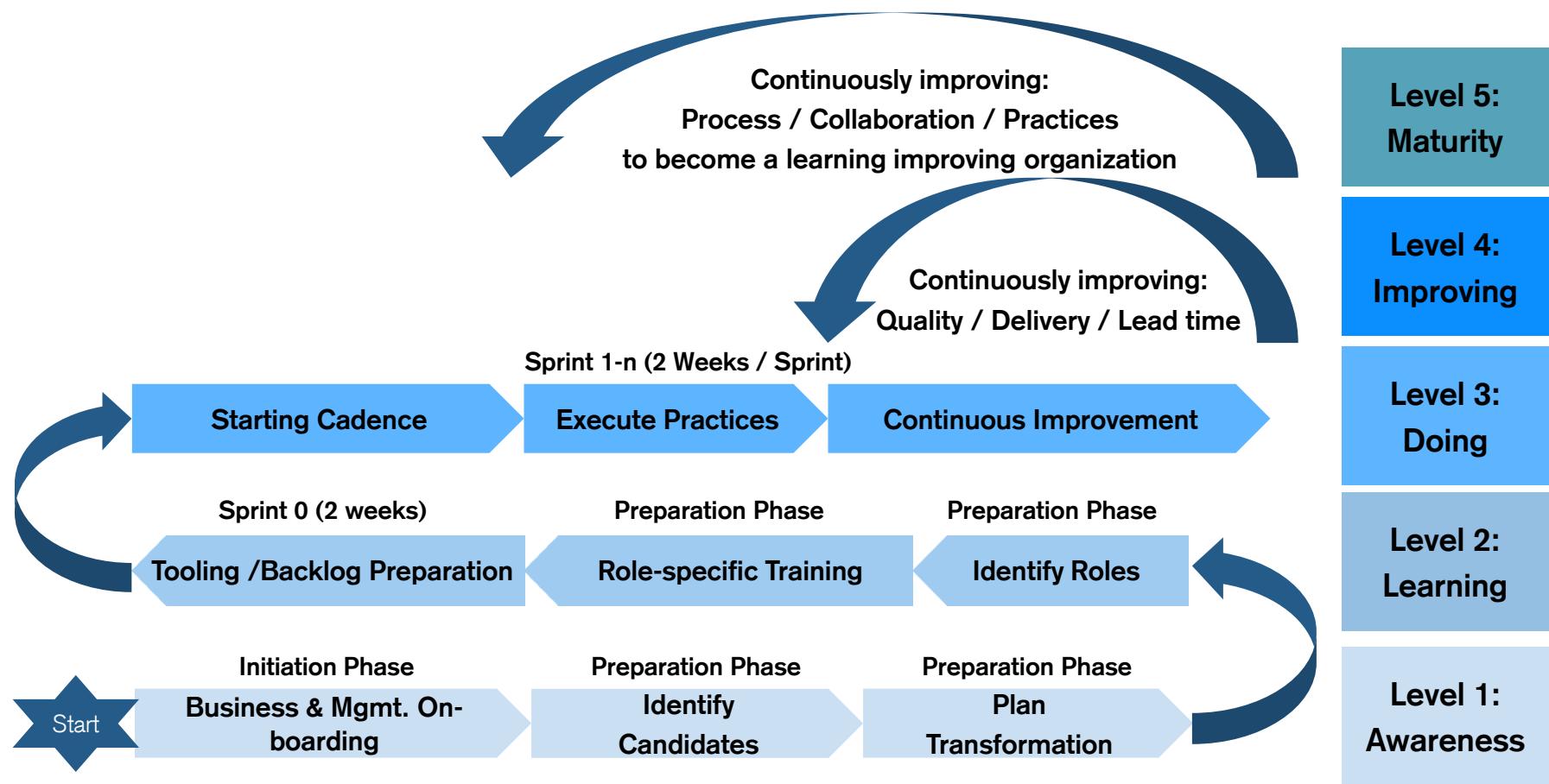


**How is a Scrum team  
exposing these principles ?**

## Q&A

# Agile Implementation

## Agile Transformation Roadmap



# Learn more about Agile

Agile Library Home

Created and last modified by Carlo Criniti just a moment ago



Agile Learning Map



Agile Events



Agile Quick Start



Agile in Practice

Available on:

<https://mylink.csintra.net/AgileL>

# Course Retrospective



## Course Retrospective

- What went well in the training course?
  - *Content and delivery*
  - *Relevance to work and examples*
  - *Clarifications and Q&A*
- What could be improved?
- Any suggestions?

# Further Information on CS-SDF



For general questions, please email the EPT Communication Team  
[communications.ept@credit-suisse.com](mailto:communications.ept@credit-suisse.com)

CS-SDF and EPT Intranet containing overviews  
<http://cs-sdf.csintra.net/>

CS-SDF Practice Library  
<https://practices.csintra.net/>

Odysse in BaSZ  
<https://odyssey.apps.csintra.net/odysseus/>

Odyssey in DvSZ  
<https://odyssey.rowini.net/odysseus/>



**Got a question? Please ask us!**

Now is the right time