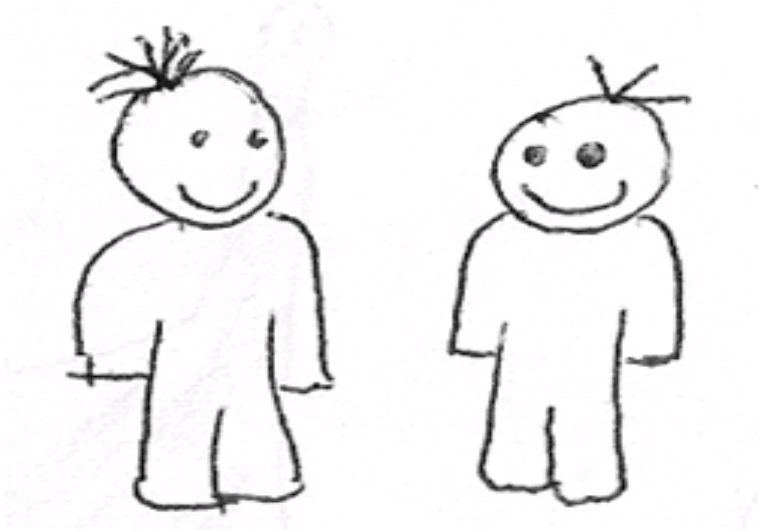
Introduction to SCRUM

M. Mizanur Rahman
CSM, CSP - ScrumAlliance

Setting up the Agile Mindset

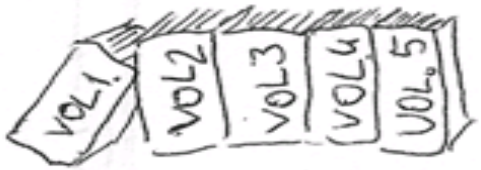
Empirical Process

Agile Manifesto



<u>Individuals and interactions</u> over processes and tools

Agile Manifesto



Working software over comprehensive documentation

Agile Manifesto

Customer collaboration over contract negotiation

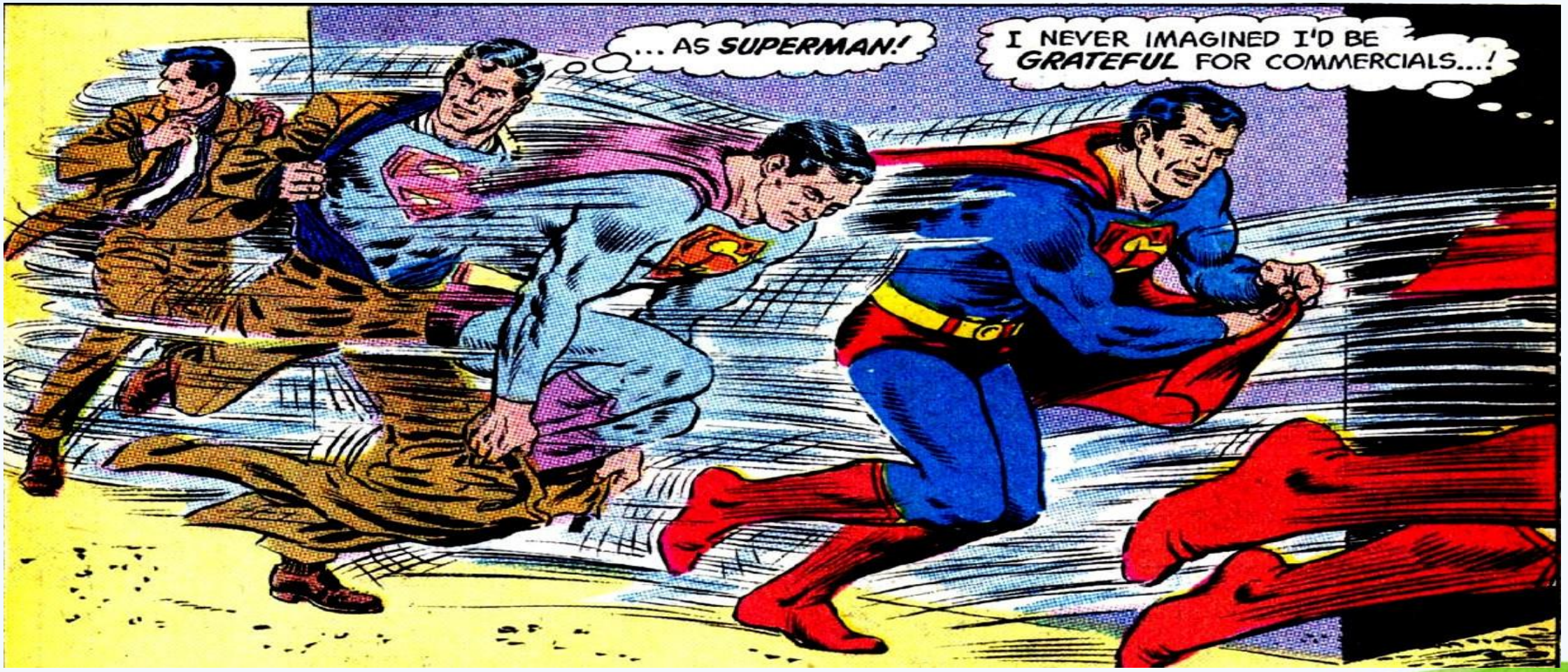
Agile Manifesto

Responding to change over following a plan

12 Agile Principles

#1: Customer satisfaction by rapid delivery of useful software

#2: Welcome changing requirements, even late in development

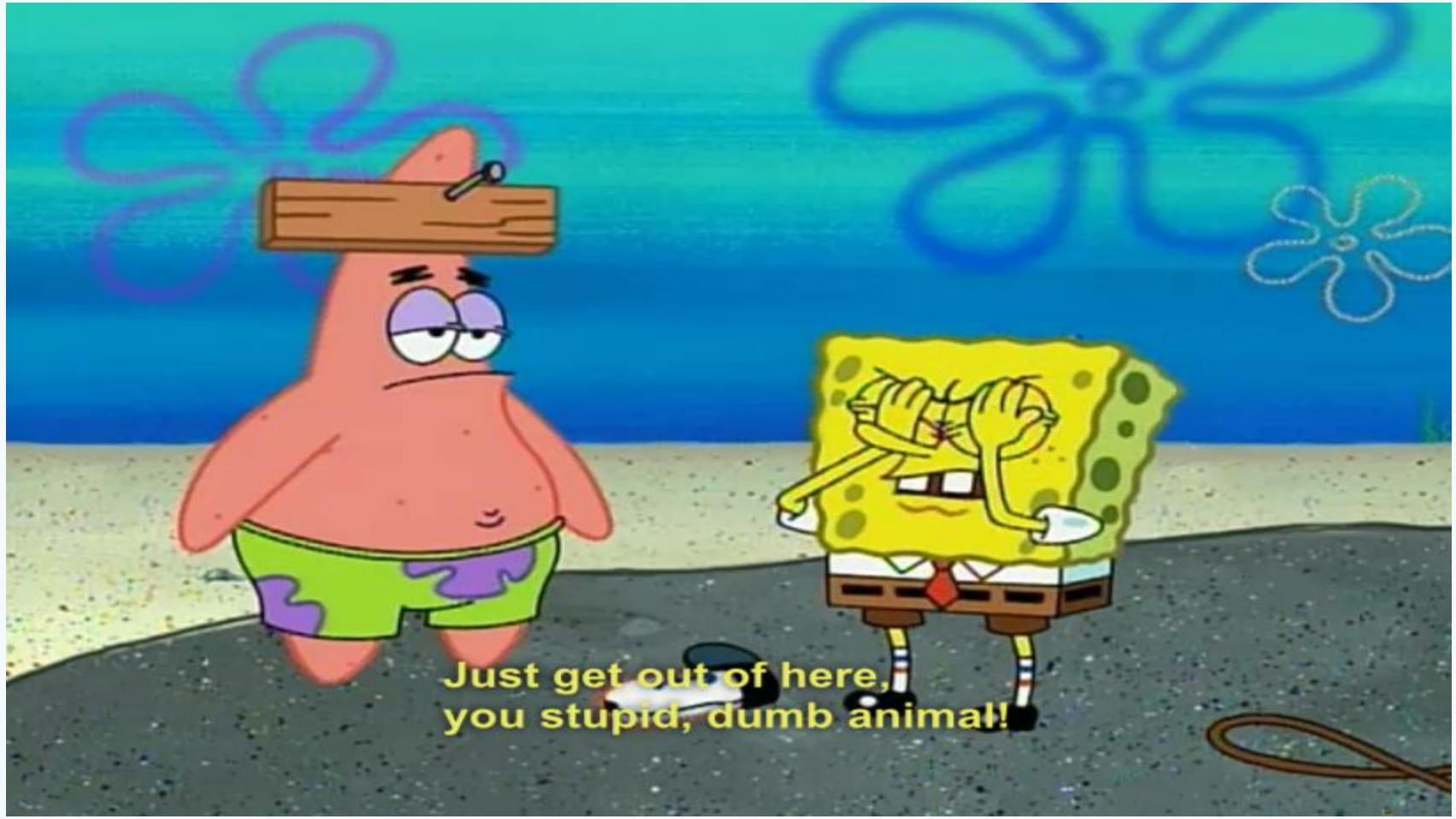


#3: Working software is delivered frequently (weeks rather than months)

#4: Close, daily cooperation between business people and developers

#5: Projects are built around motivated individuals, who should be trusted

#6: Face-to-face conversation is the best form of communication (co-location)



#7: Working software is the principal measure of progress

#8: Sustainable development, able to maintain a constant pace

#9: Continuous attention to technical excellence and good design

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

#11: Self-organizing teams

#12: Regular adaptation to changing circumstance

Agile Manifesto

- **Individuals and interactions** over processes and tools
- **Working software** over comprehensive documentation
- **Customer collaboration** over contract negotiation
- **Responding to change** over following a plan

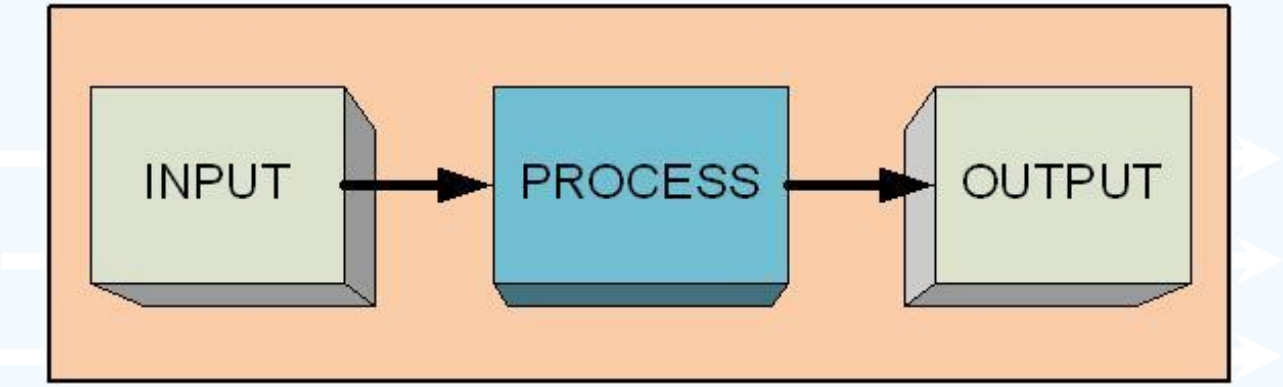
The 12 Agile Principles

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.

The 12 Agile Principles - continued

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.

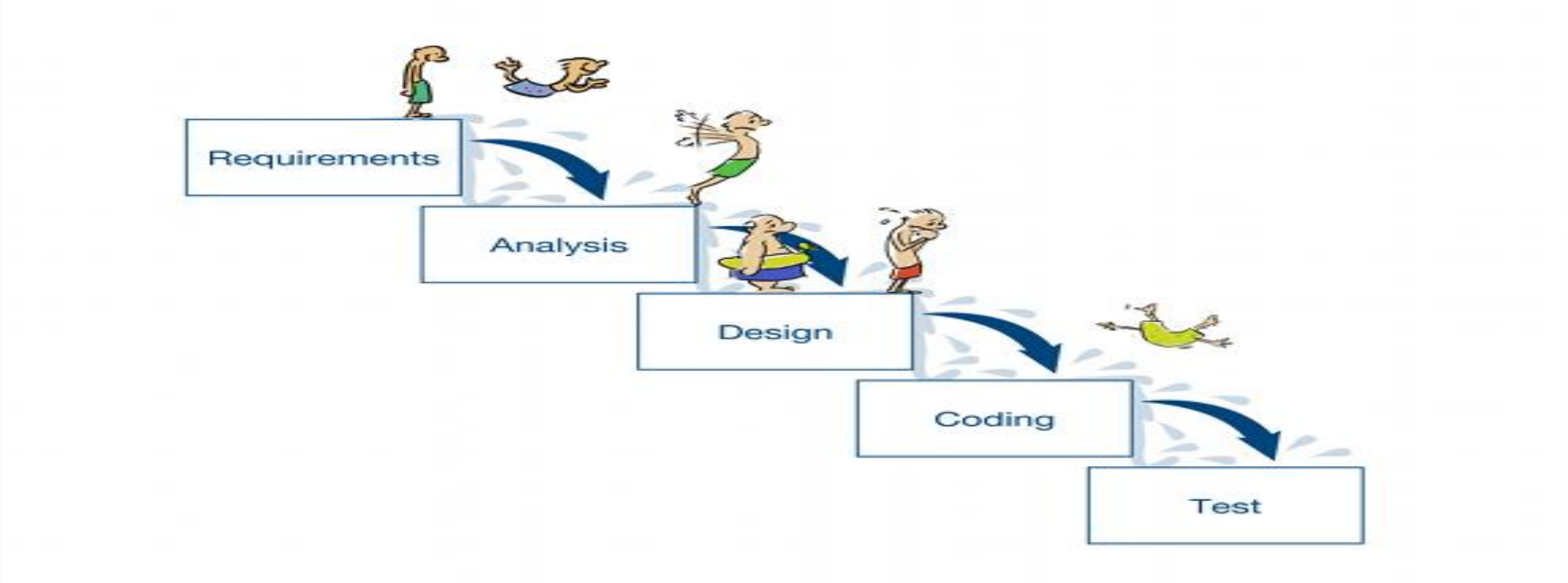
Defined Processes



Assuming every step is understood, a given well-defined set of inputs produces the same set of outputs every time.

Predictability is key.

The Waterfall



Advantage: Highly Logical

Disadvantage: People are involved

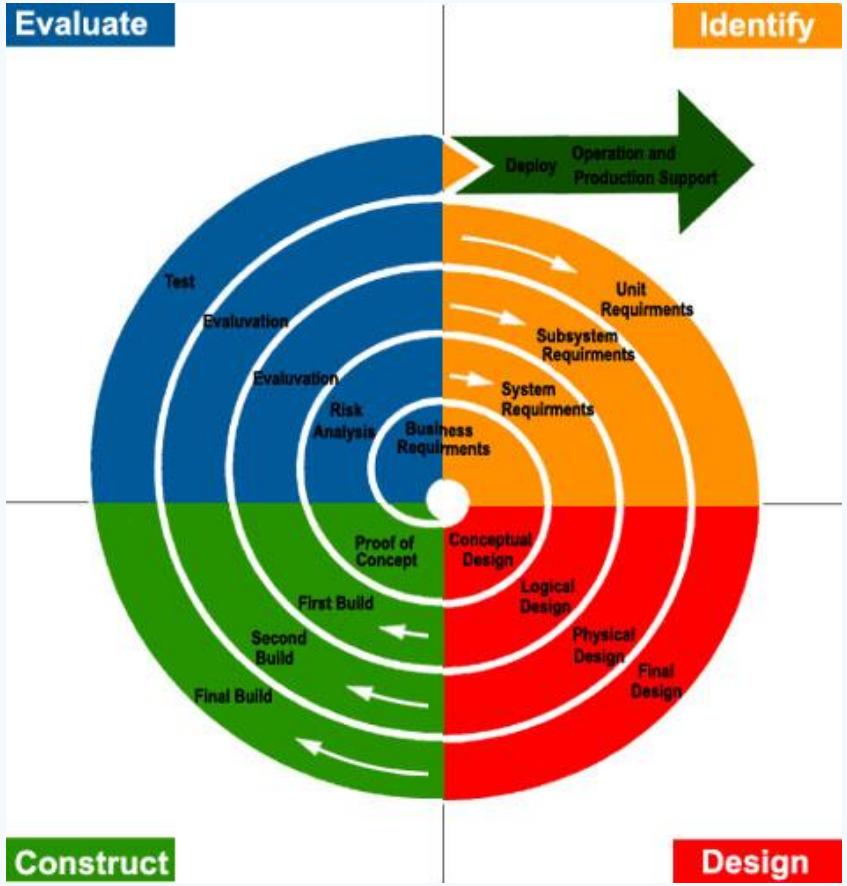
Iterative Incremental Development

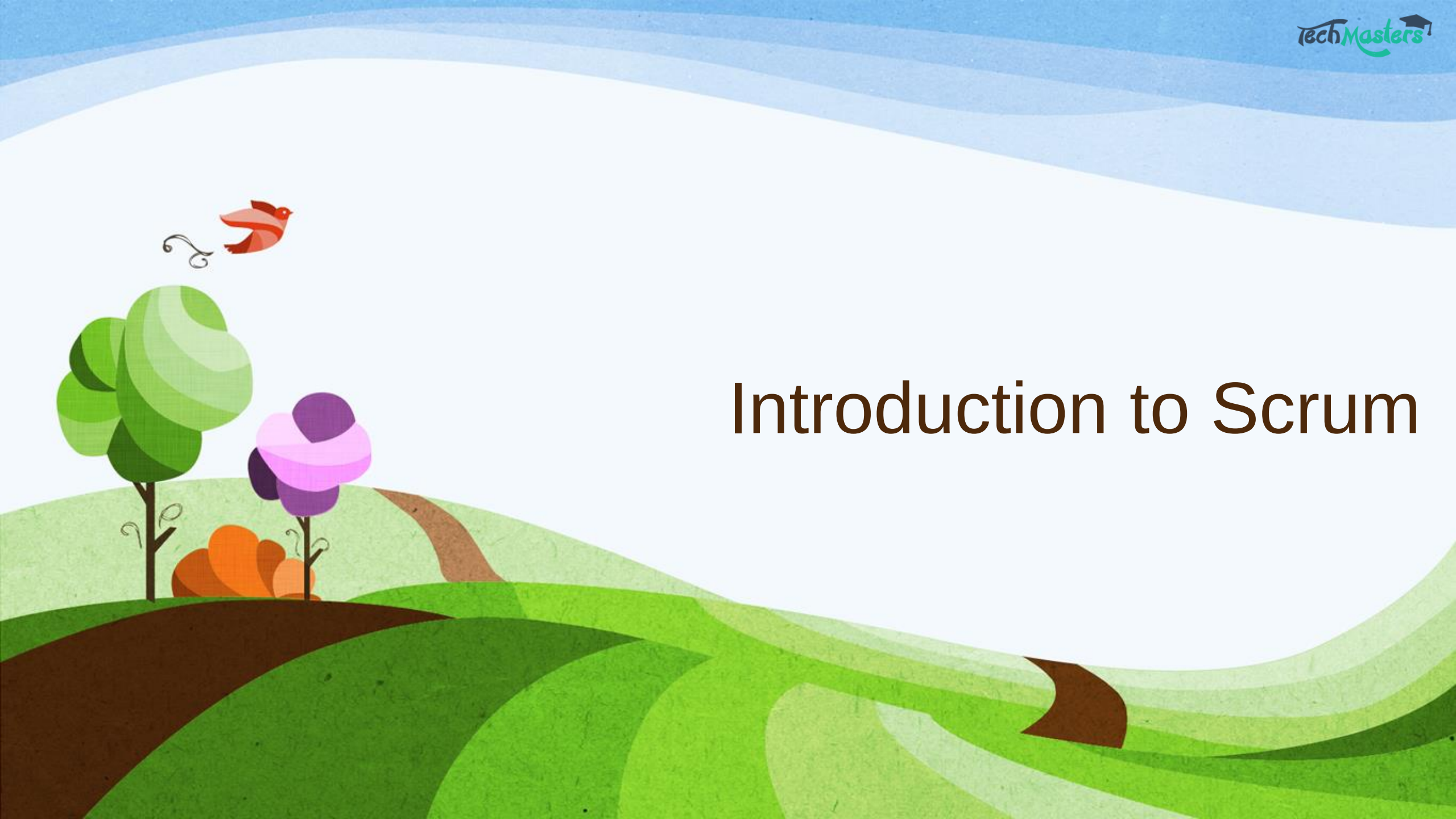
Iterative Development

- Project is structured as multiple short cycles in sequence, versus one long cycle

Incremental Development

- Each iteration produces a usable increment of finished product, with actual customer value



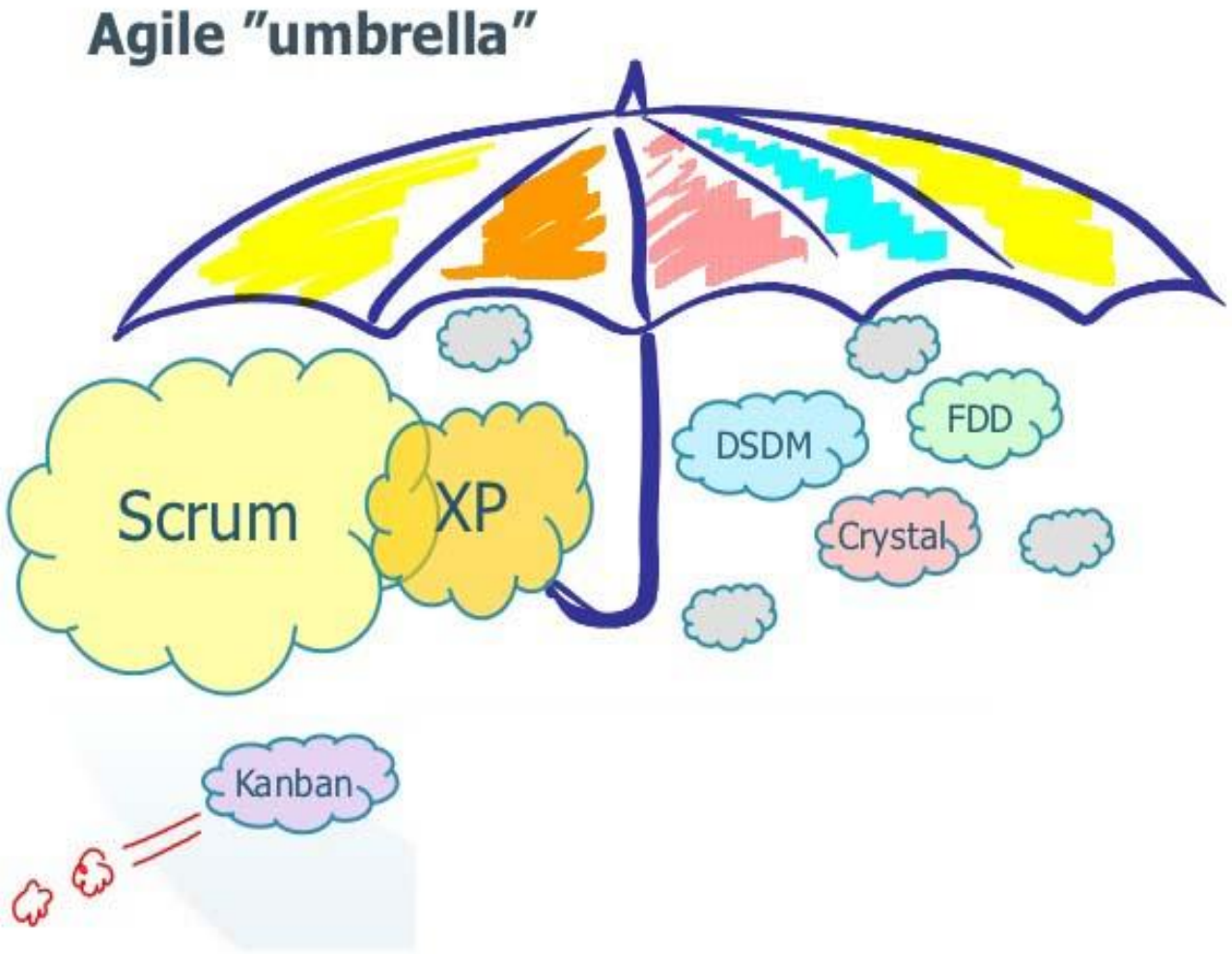


Introduction to Scrum

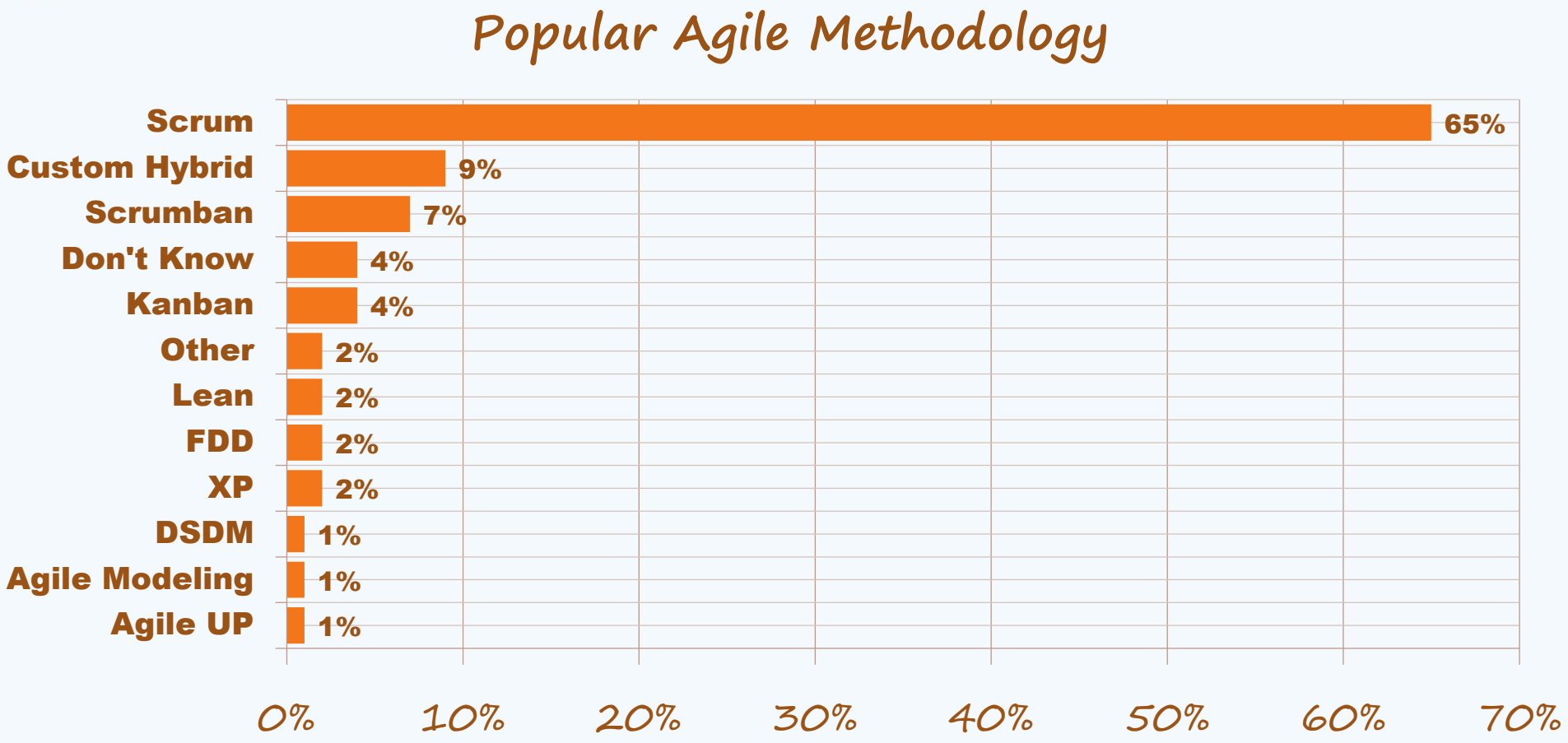
What is Scrum?

- Scrum is an Agile framework for completing complex projects.
- Scrum originally was formalized for software development projects, but it works well for any complex, innovative scope of work. The possibilities are endless. The Scrum framework is deceptively simple.
- An Iterative and Incremental method from Agile Family

What are they doing?


Agile "umbrella"
Scrum
XP
DSDM
FDD
Crystal
Kanban

The Agile Family



The Emergence of Scrum

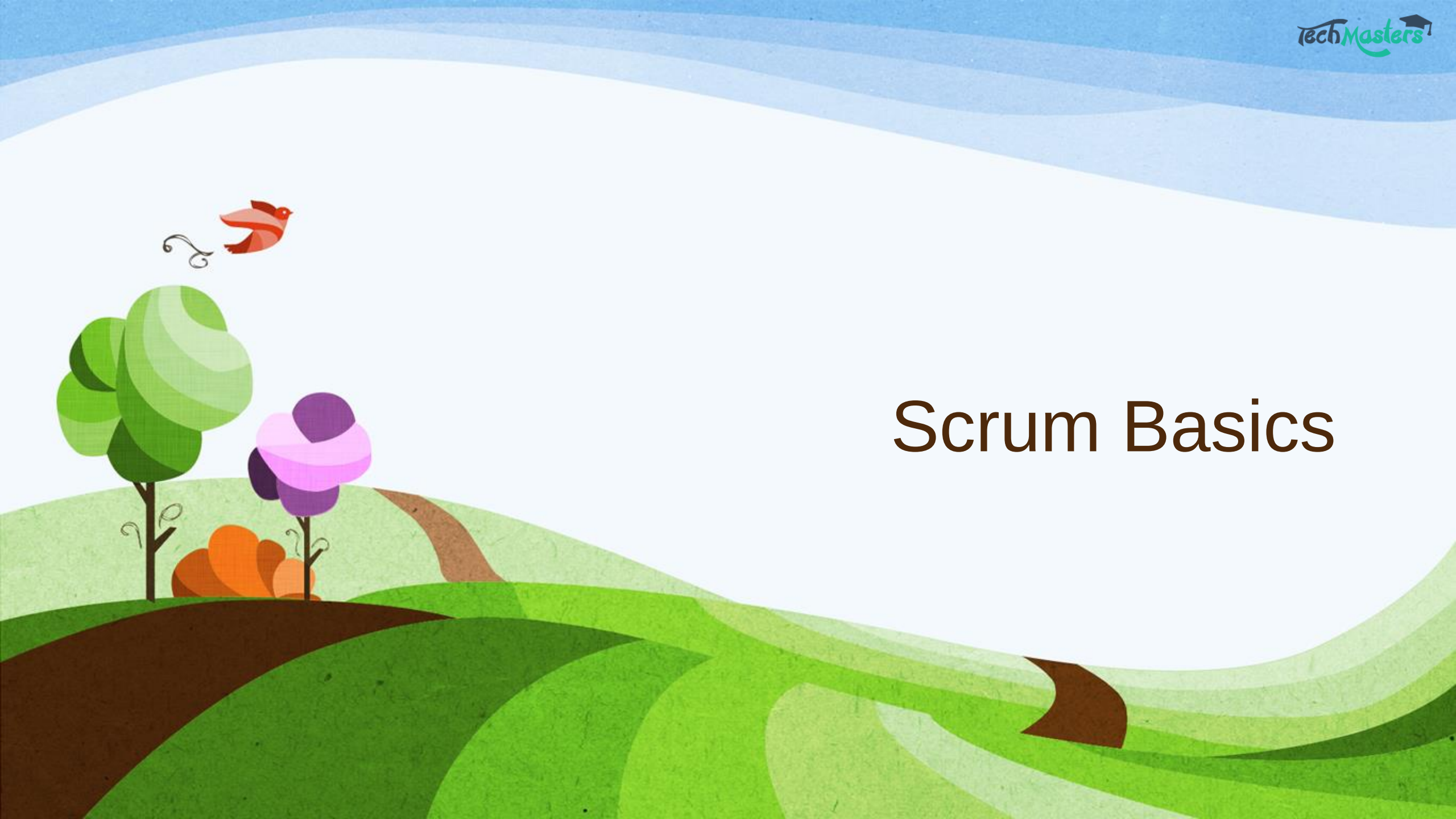
- Scrum formalized in 1996 by **Ken Schwaber**
  - “The Origins of Scrum”, OOPSLA 1996
  - **ScrumAlliance** is now taking the lead role in scrum growth worldwide
  - Ken Schwaber runs the scrum.org since 2009 after he left ScrumAlliance
  - There are several scrum certifications
    - CSM – Certified Scrum Master (Most Popular One)
    - CSPO – Certified Scrum Product Owner
    - CSD – Certified Scrum Developer
    - CSP – Certified Scrum Professional
    - CST – Certified Scrum Trainer
    - CTC – Certified Team Coach
    - CEC – Certified Enterprise Coach

What Is Scrum Being Used For?

- US FDA-approved software for X-Rays, MRIs
- High availability systems (99.9999% uptime)
- Financial payment applications
- Large database applications
- Embedded systems
- CMMi organizations
- Multi-location development
- Non-software projects

Scrum Disadvantages

- It’s hard!
- Makes all dysfunction visible
  - Scrum doesn’t fix anything: the team has to do it
  - Feels like things are worse at the beginning
- Bad products will be delivered sooner, and doomed projects will fail faster
- Some teams and organizations are not right or ready for it
  - Team willingness, capabilities
  - Management buy-in
- Risk of turnover during adoption
  - Some people will refuse to stay on a Scrum team
  - Some people will refuse to stay if Scrum is abandoned
- Partial adoption may be worse than none at all
- If adoption fails, time will have been wasted, and some people may leave

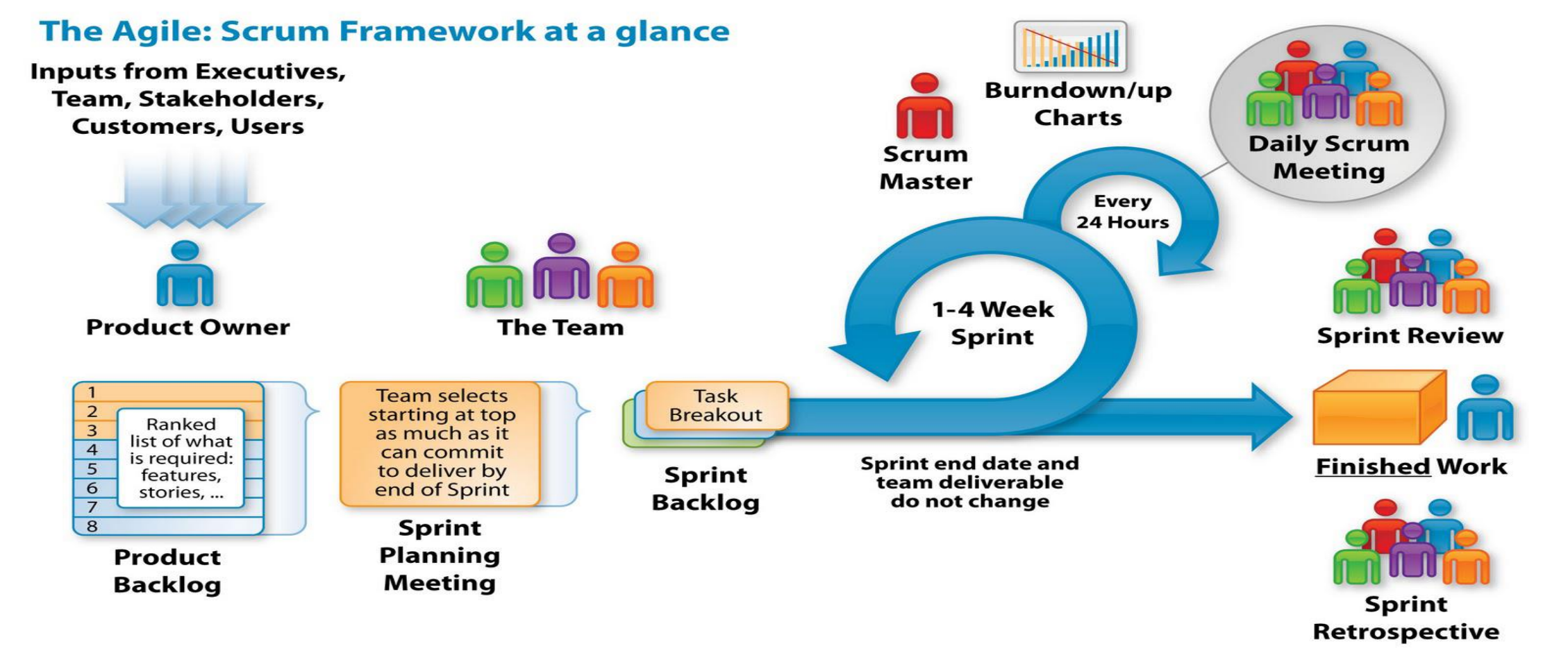


Scrum Basics

The Foundations of Scrum

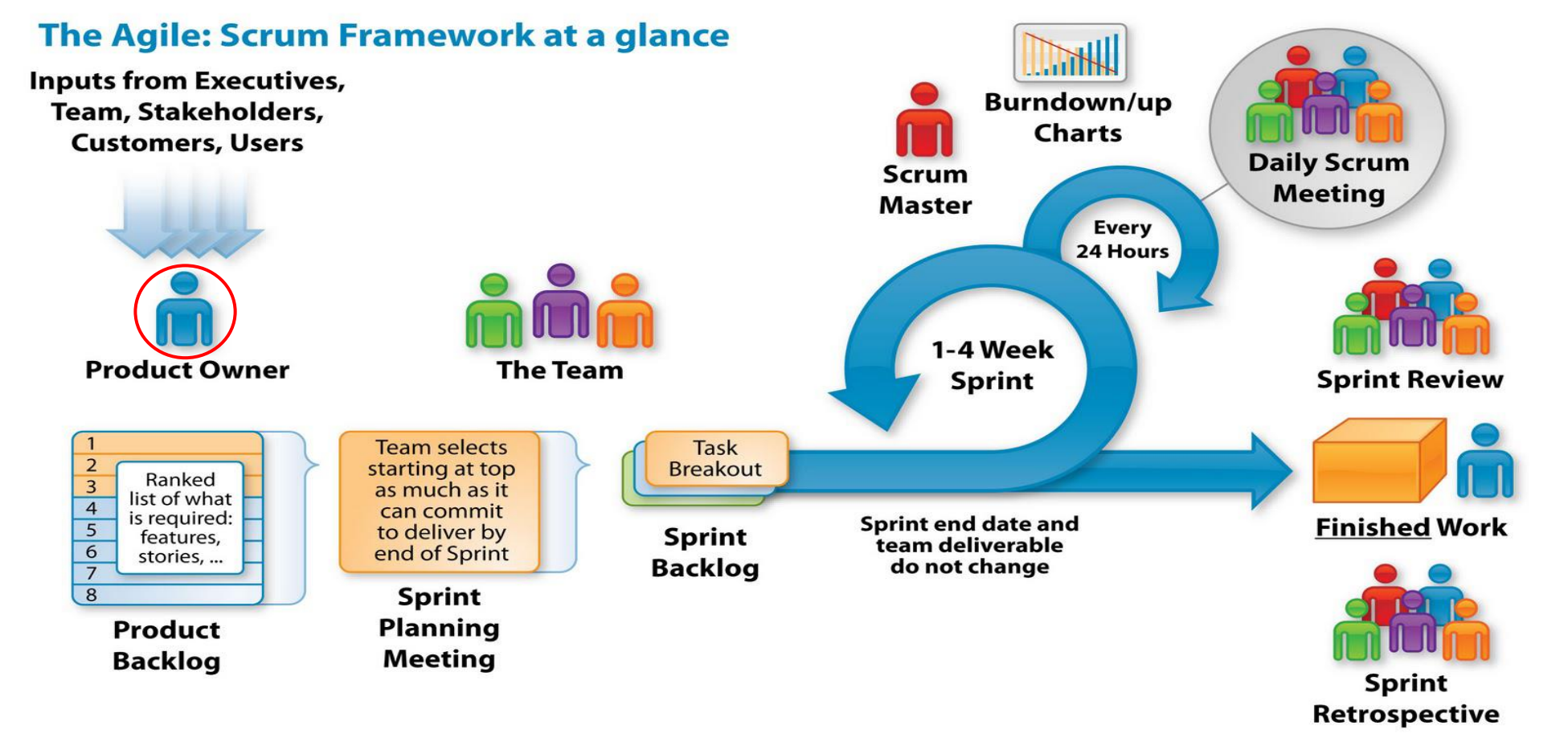
- **There is a cross-functional Development Team – as a team, they have the skills needed to produce “done” product in a Sprint.**
- **The Development Team is self-organizing – the team organizes itself to get the work done.**
- **The Dev Team plans its Sprints one at a time, at the start of the Sprint.**
- **The Product Owner decides what needs to be produced.**
- **The Dev Team decides how much to target in each Sprint.**
- **The Dev Team’s goal is shared, clear, and does not change in Sprint.**
- **The Dev Team tries to hit its Sprint goal, but in some Sprints it will under-deliver, and in other Sprints it will over-deliver.**
- **Each Sprint is made up of timeboxes, which are not extended.**
- **The Dev Team aims for “done” each Sprint = tested and defect-free.**
- **There is a “Definition of Done,” that tells us what “done” means.**
- **At the end of the Sprint, we inspect and adapt product and process.**

Snapshot


The Agile: Scrum Framework at a glance
Inputs from Executives, Team, Stakeholders, Customers, Users
Product Owner
The Team
Scrum Master
Burndown/up Charts
Daily Scrum Meeting
Every 24 Hours
1-4 Week Sprint
Sprint Review
1
2
3
4
5
6
7
8
Ranked list of what is required: features, stories, ...
Product Backlog
Team selects starting at top as much as it can commit to deliver by end of Sprint
Sprint Planning Meeting
Task Breakout
Sprint Backlog
Sprint end date and team deliverable do not change
Finished Work
Sprint Retrospective

Scrum Roles

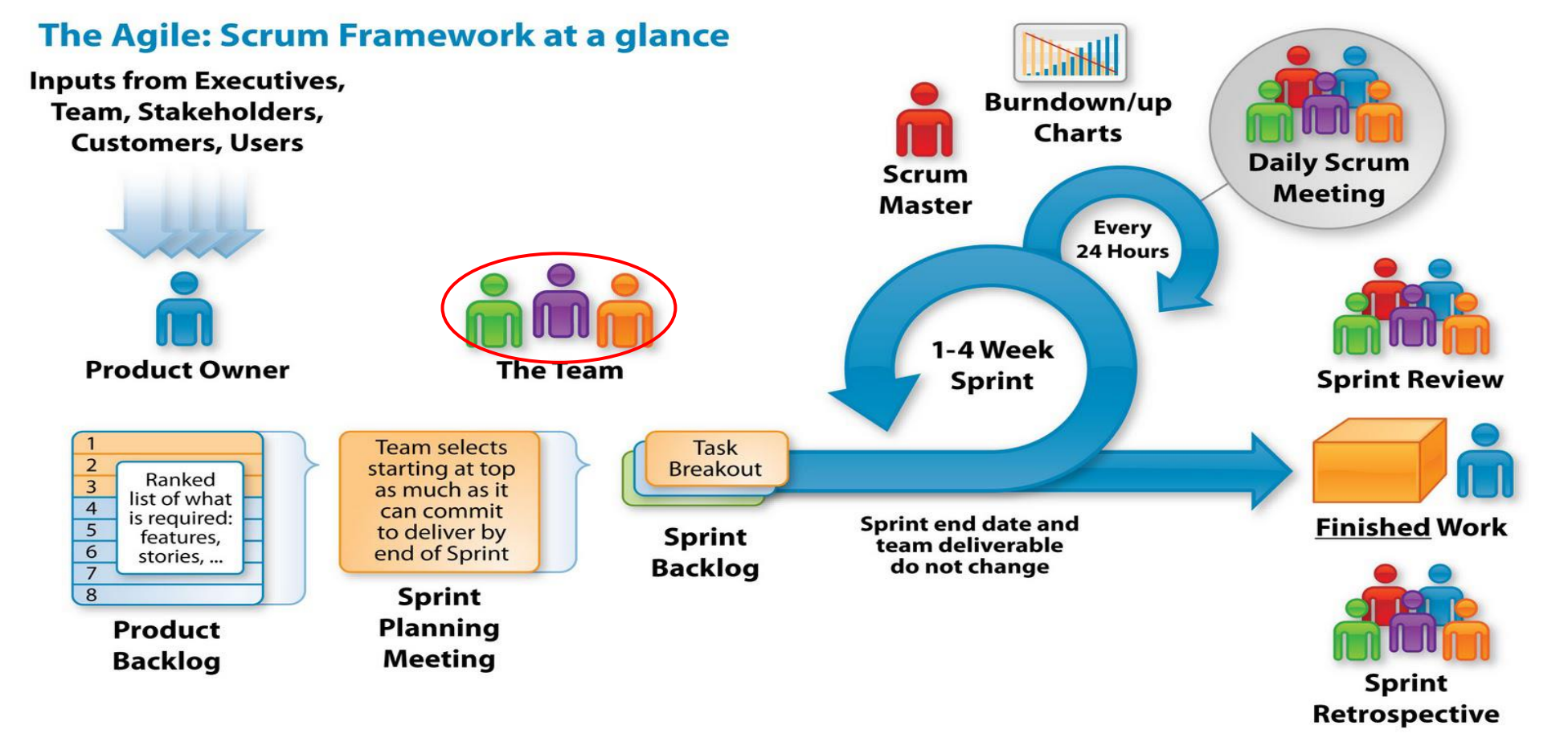
The Basics of Scrum



Product Owner

- Responsible for the overall project vision and goals
- Responsible for managing project ROI vs. risk
- Responsible for taking all inputs into what the team should produce, and turning it into a prioritized list (the Product Backlog)
- Participates actively in Sprint Planning and Sprint Review meetings, and is available to team throughout the Sprint
- Determines release plan and communicates it to upper management and the customer

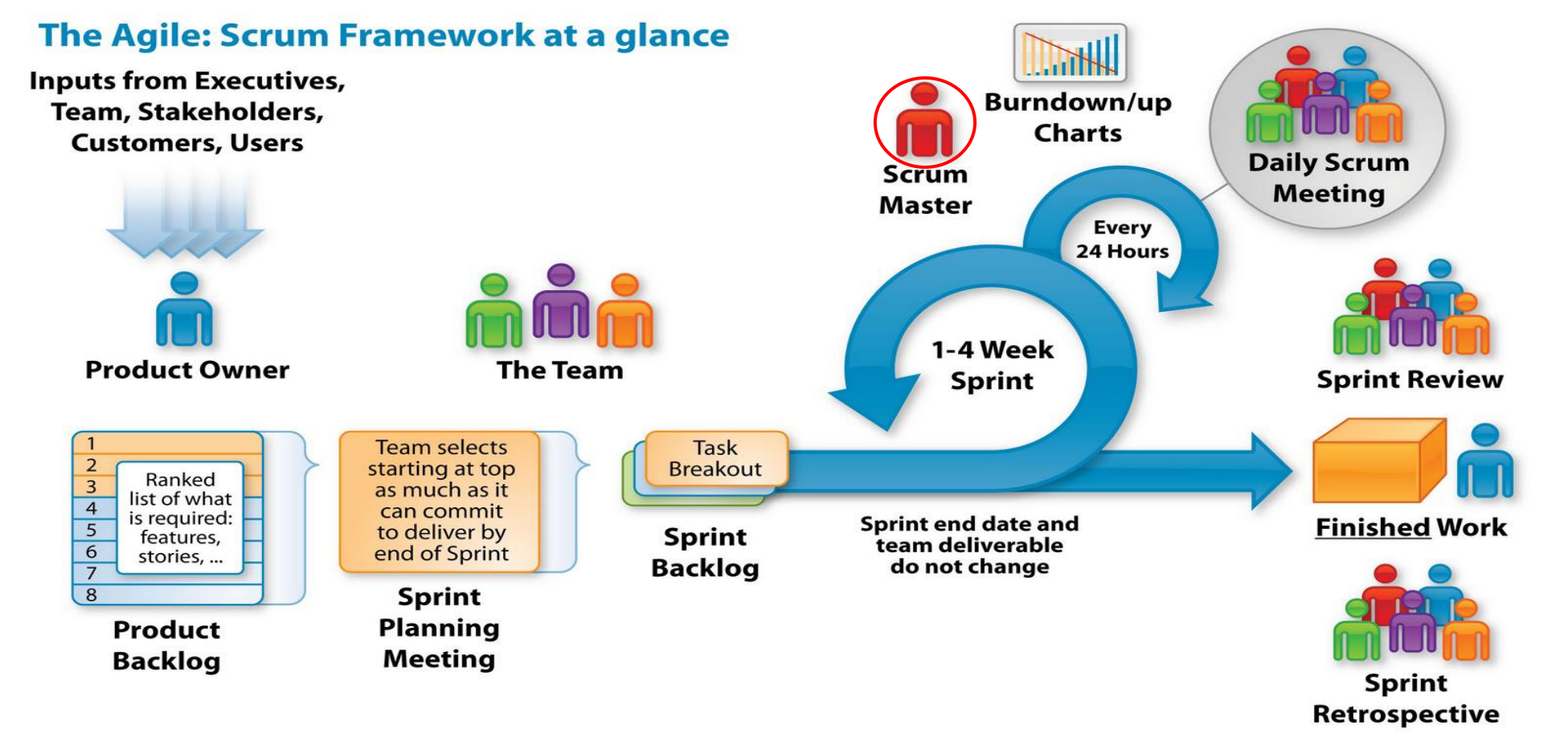
The Basics of Scrum



Team

- 7 people, + or – 2
  - Has worked with as high as 15, as few as 3
  - Can be shared with other teams (but better when not)
  - Can change between Sprints (but better when they don't)
  - Can be distributed (but better when colocated)
- Cross-functional
  - Possesses all the skills necessary to produce an increment of potentially shippable product
  - Team takes on tasks based on skills, not just official "role"
- Self-managing
  - Team manages itself to achieve the Sprint commitment

The Basics of Scrum



The Role of the ScrumMaster

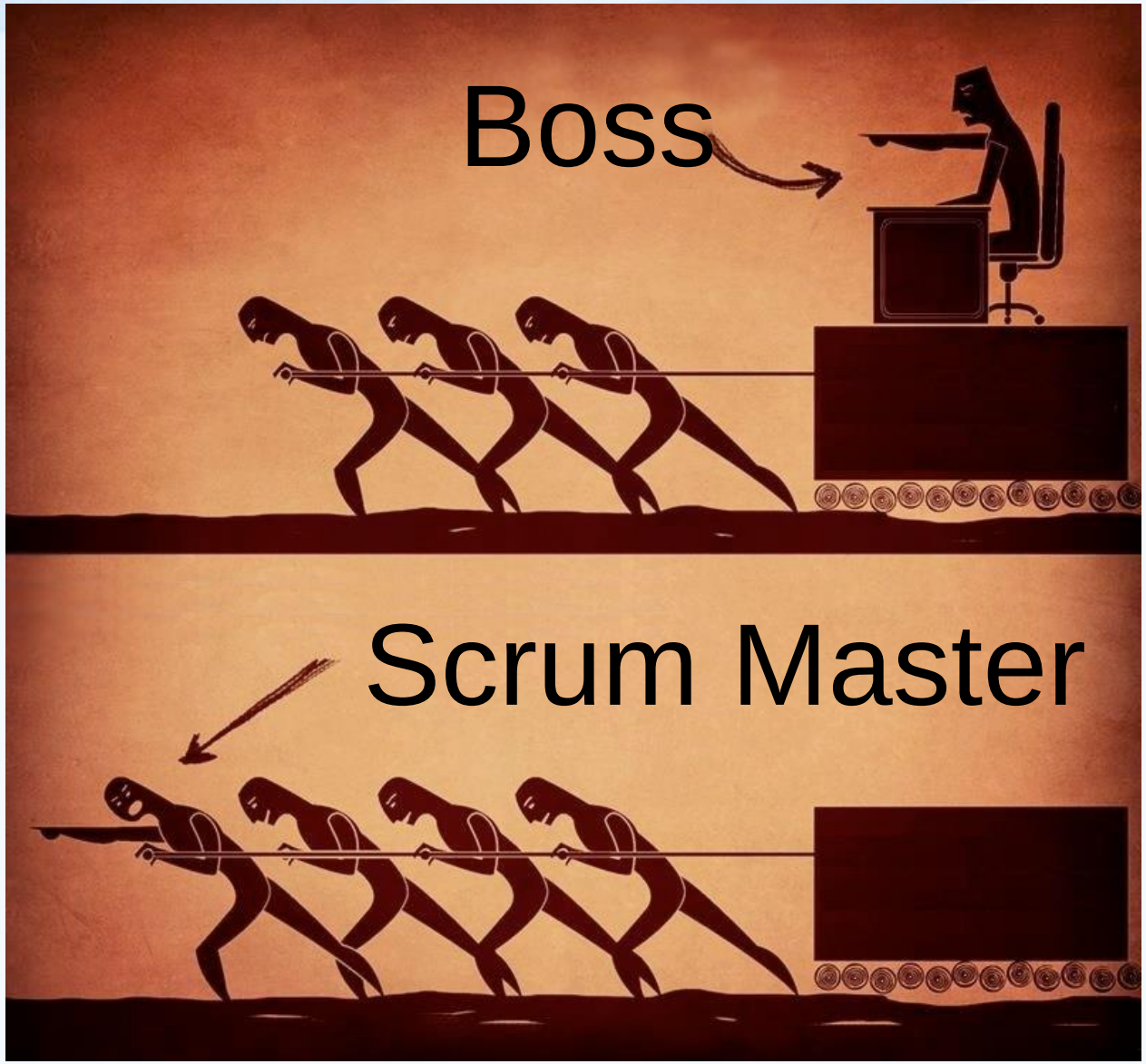
- The ScrumMaster does everything in their power to help the team achieve success
- This includes:
  - Serving the team
  - Protecting the team
  - Guiding the team’s use of Scrum

What the ScrumMaster Does

- Serves the team
  - The ScrumMaster takes action to help remove impediments to the team's effectiveness
  - The ScrumMaster facilitates the team's group interactions, to help the team achieve its full potential
  - The ScrumMaster coaches the team, to help them improve their practices and effectiveness
- Protects the team
  - The ScrumMaster protects the team from anything that threatens its effectiveness, such as outside interference or disruption
  - The ScrumMaster will need to confront uncomfortable issues, both inside and outside the team
- Guiding the team's use of Scrum
  - The ScrumMaster teaches Scrum to the team and organization
  - The ScrumMaster ensures that all standard Scrum rules and practices are followed
  - The ScrumMaster organizes all Scrum-related practices

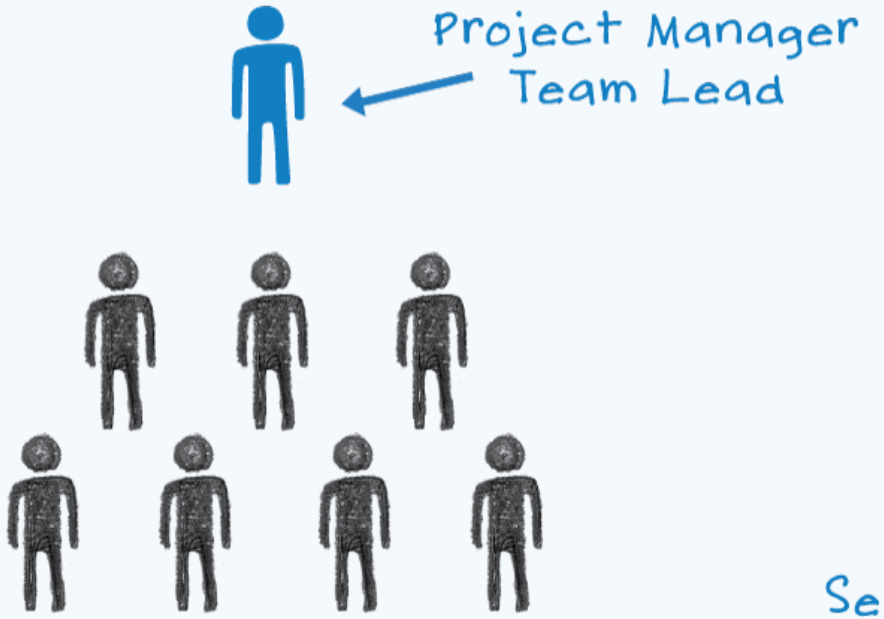
What Does the ScrumMaster NOT Do?

- The ScrumMaster does not manage the team
- The ScrumMaster does not direct team-members
- The ScrumMaster does not assign tasks
- The ScrumMaster does not “drive the team” to hit its goals
- The ScrumMaster does not make decisions for the team
- The ScrumMaster does not overrule team-members
- The ScrumMaster does not direct product strategy, decide technical issues, etc.

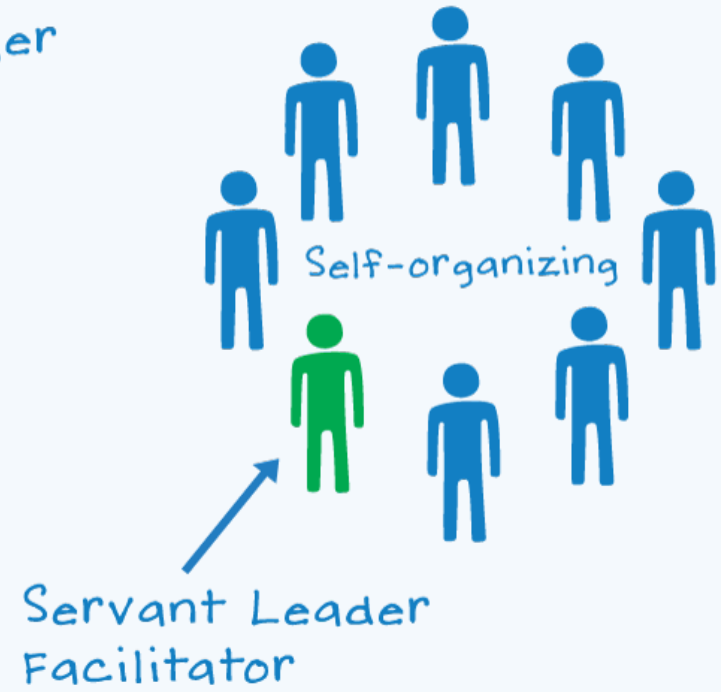

Boss
Scrum Master

The Servant Leader

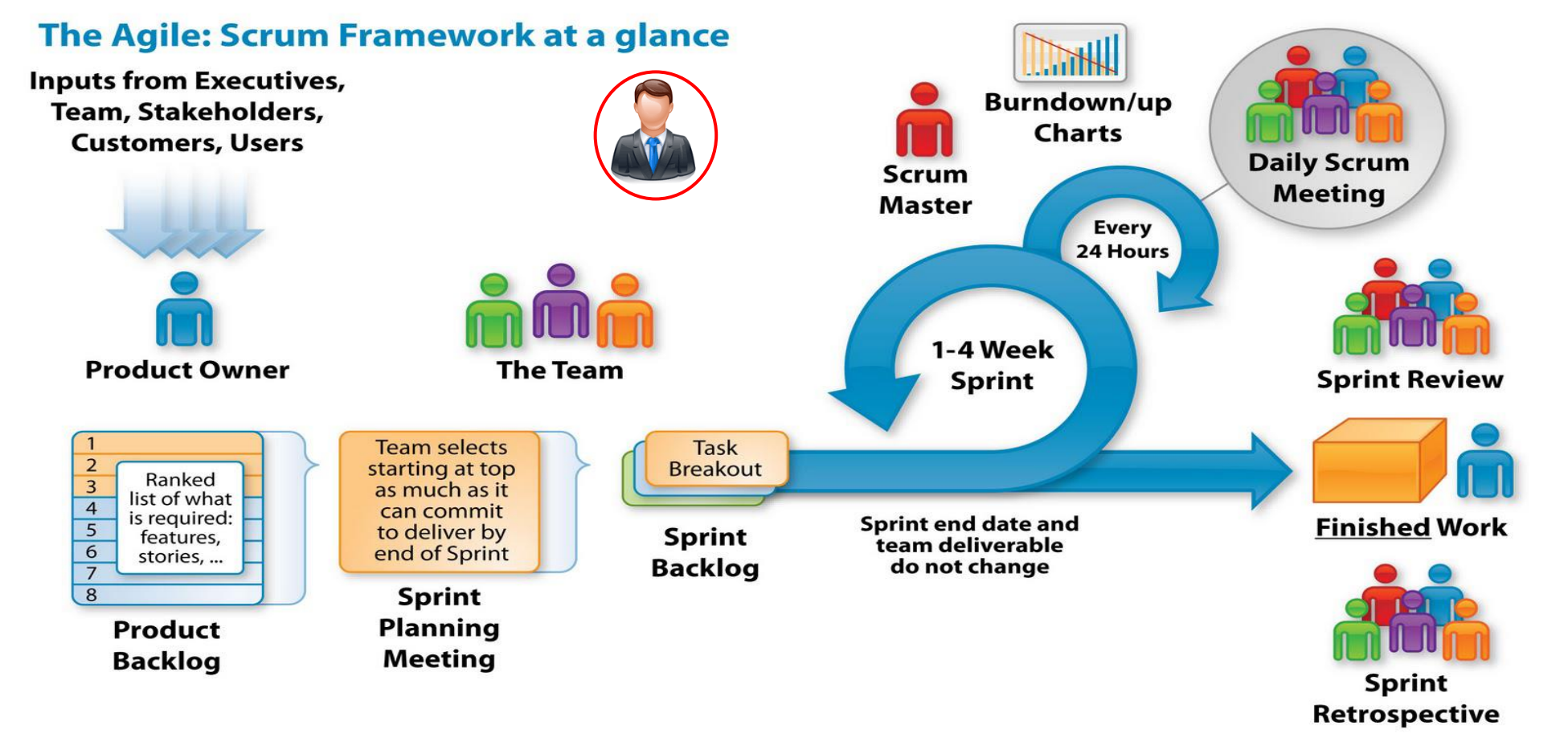
Traditional Teams



Agile Teams



What Happens to the “Project Manager” in Scrum?

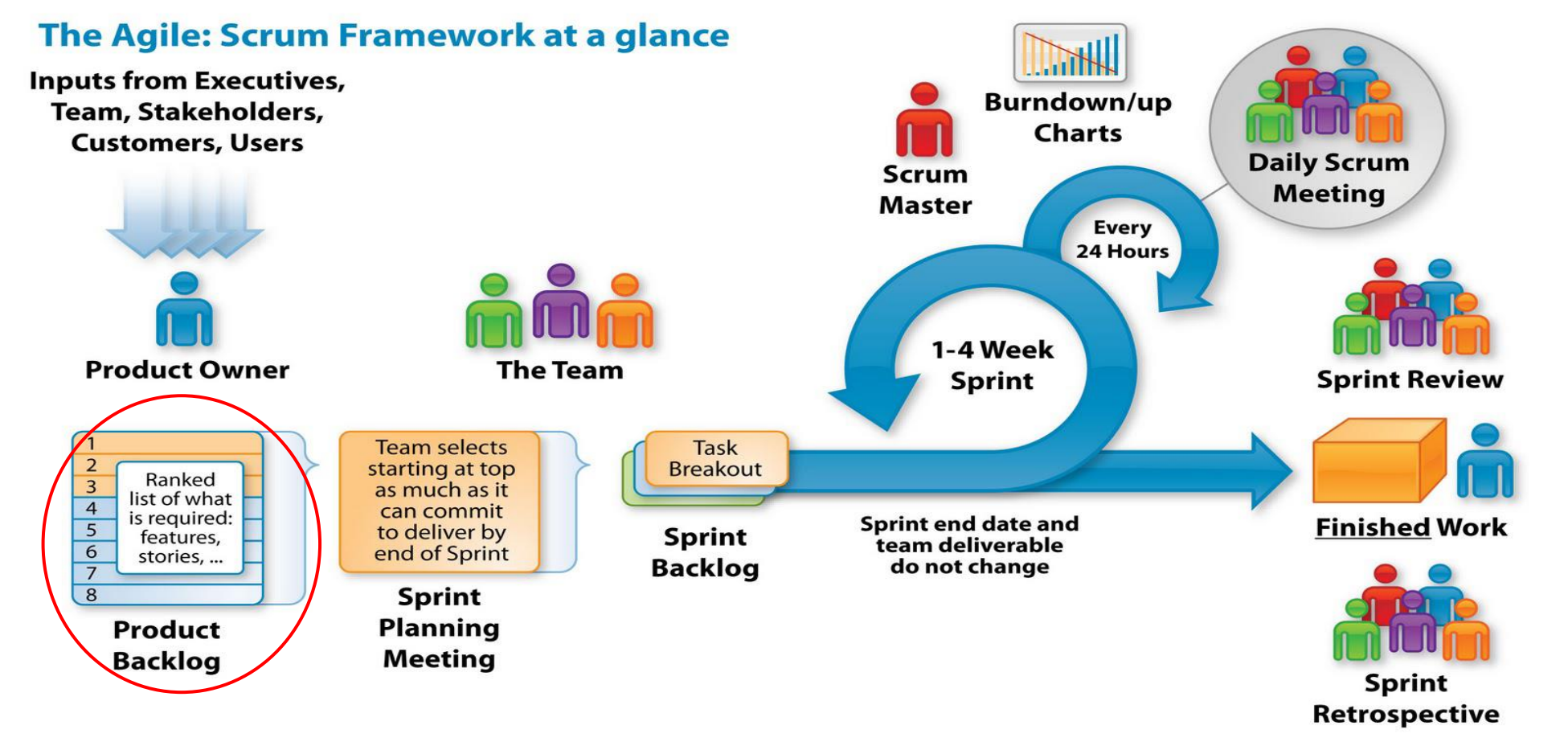


Transforming “Project Manager” to …..

- There is no such role is scrum where we can have a project manager.
- We need to transform our traditional project managers to
  - ScrumMaster (Highly risk involved as the intention of managing the team will be there)
  - Product Owner
  - Team Member (Hard to adjust as a member of the team)
  - More of a management role outside Scrum environment (Product Release manager, VP Software development, Analyst etc)

Starting Scrum

The Basics of Scrum



Product Backlog

Description	Rough Est. Size
Enable all users to place book in shopping cart (mocks and additional details are located here)	20
Upgrade transaction processing module (must be able to support minimum 500 transactions per second)	10
Investigate solutions for speeding up credit card validation (see target performance metrics located here)	20
Upgrade all servers to Apache 2.2.3	40
Diagnose and fix the order processing script errors (bugzilla ID 14823)	20
Enable all users to create / save wishlist	40
Enable all users to add and delete items on their wishlist	10

Product Owner lists items in descending order of priority (highest priority item is listed first, next-highest is second, etc.)

Size estimates are rough estimates (can either be arbitrary "points", or "ideal days")

Product Backlog

- List of everything that could ever be of value to the business for the team to produce
- Ranked in order of priority
  - Priority is a function of business value versus risk
- Product Owner can make any changes they want before the start of a Sprint Planning Meeting
  - Items added, changed, removed, reordered
- How much documentation is up to the team and Product Owner to decide
- The farther down the list, the bigger and less defined the items become
  - ~2 Sprints worth are defined in detail

User Stories

- User Stories are a good approach for writing Product Backlog Items
- User Stories are a short, plain-language description of the functionality, in terms of the customer benefit and need

Good Format

As a…

I want…

So that…

“As a customer, I can place an item on my wishlist, so that I can decide later whether or not I want to buy it”

“As a frequent flyer member, I can see the number of miles I have earned in my frequent flyer account, so that I can decide whether to redeem them for a ticket”

“As a new user, I can set up a profile so that potential employers can find out more about my skills and qualifications”

Epic & Themes

- Larger user stories are referred as Epic.
- Epics generally take more than one or two sprints to develop and test. They are usually broad in scope, short on details, and will commonly need to be split into multiple, smaller stories before the team can work on them.
- A theme is a collection of related user stories.
- A user Story can belong to one or more themes.
  Often times, an Epic can be sliced into several stories. One way to remember that all of these sliced stories is related is to say that they belong to a particular theme.

INVEST in Good Stories and SMART Tasks

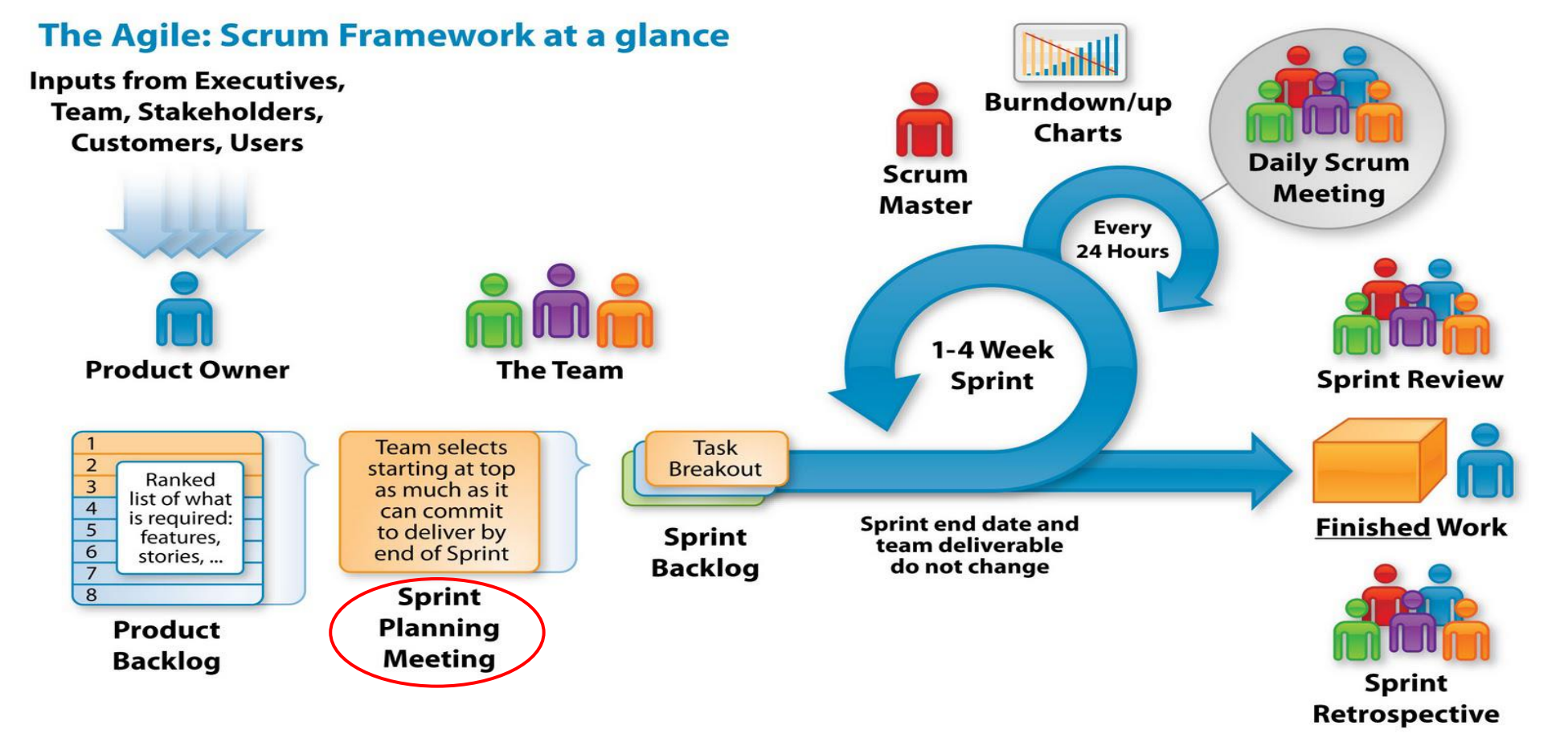
Stories should be

- I – Independent
- N – Negotiable
- V – Valuable
- E – Estimable
- S – Small
- T – Testable

Tasks should be

- S – Specific
- M – Measurable
- A – Achievable
- R – Relevant
- T – Time-boxed

The Basics of Scrum



Sprint Planning Meeting

- Takes place before the start of every Sprint
- Team decides how much Product Backlog it will commit to complete by the end of the Sprint, and comes up with a plan and list of tasks for how to achieve it
- What's a good commitment?
  - Clearly understood by all
  - Shared among the team
  - Achievable without sacrificing quality
  - Achievable without sacrificing sustainable pace
- Attended by Team, Product Owner, ScrumMaster, Stakeholders
- May require 1-2 hours for each week of Sprint duration
  - 2 week Sprint = 2-4 hours, 4 week Sprint = 4-8 hours

Sprint Planning

- Team understands the details of what the Product Owner has prioritized on the Product Backlog
- Team decides how much productive time it has available during the Sprint
- Team decides how many Product Backlog items it can commit to complete during the Sprint

Sprint Planning

- Team understands the details of what the Product Owner has prioritized on the Product Backlog
- Team decides how much productive time it has available during the Sprint
- Team decides how many Product Backlog items it can commit to complete during the Sprint

Sprint Pre-Planning Meeting

Sprint Planning Meeting

Pre-planning meeting / Backlog grooming

- Should take place 2-3 days before the actual sprint planning meeting
- Team must understand the prioritized items by product owner and ask for clarification if required
- Provide rough estimation for new items
- Calculate team's available days, hours etc.
- During each Sprint, the Dev Team, Product Owner, and ScrumMaster look ahead to upcoming Product Backlog Items (or User Stories) coming in the next 2-3 Sprints
- They split larger items into smaller slices – ideally, small enough that 1-2 people could completely finish them in 3-4 days ("1-2-3-4")
- They start to get a more detailed understanding of the requirements for these upcoming Product Backlog Items
- No fixed format, timing, or timebox: try a 2-3 hour session for Dev Team, Product Owner and ScrumMaster

Sprint Cycle: 2-Week Sprint

Mon	Tues	Weds	Thurs	Fri
		1	2	3
6 Sprint Planning Meeting	7 1	8 2	9 3	10 4
13 5	14 6	15 7	16 8	17 Sprint Review & Retrospective
20	21	22	23	24
27	28	29	30	31

Sprint Cycle: 4-Week Sprint

Mon	Tues	Weds	Thurs	Fri
		1	2	3
6 Sprint Planning Meeting	7 1	8 2	9 3	10 4
13 5	14 6	15 7	16 8	17 9
20 10	21 11	22 12	23 13	24 14
27 15	28 16	29 17	30 18	31 Sprint Review & Retrospective

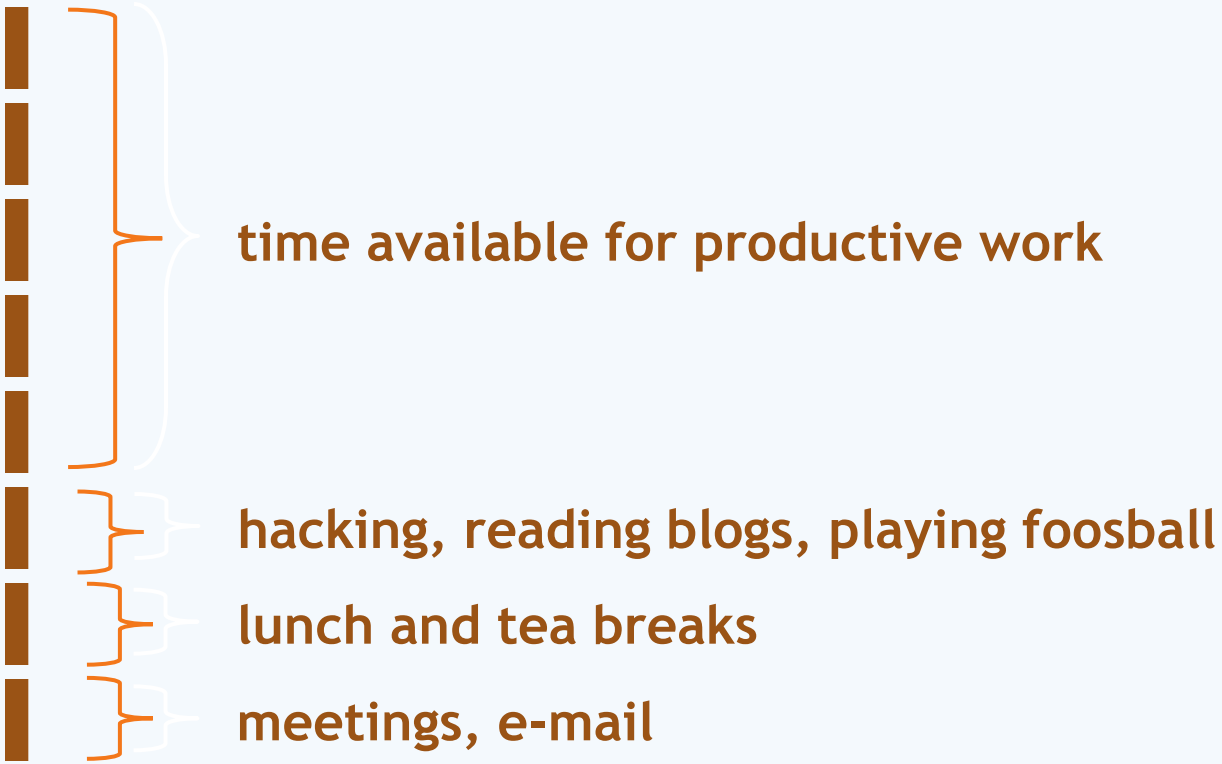
Available Time During Sprint

Sprint Length			
Workdays During Sprint			
Team Member	**Avail Days During Sprint***	**Avail Hours Per Day**	**Total Avail Hrs in Sprint**

*Net of holidays and other days out of the office

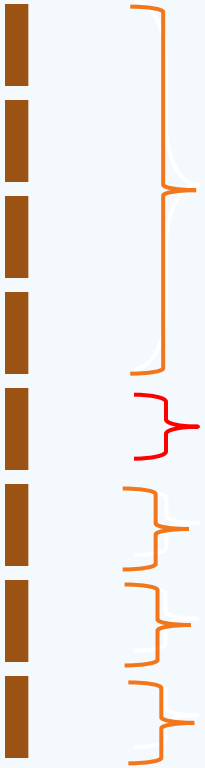
Hours Per Day

“8-Hour Day”



Hours Per Day

"8-Hour Day"



time available for productive work

operational responsibilities

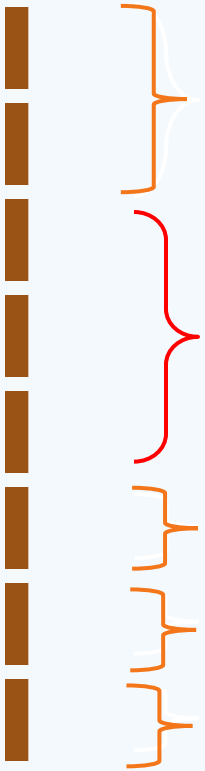
hacking, reading blogs, playing foosball

lunch and tea breaks

meetings, e-mail

Hours Per Day

“8-Hour Day”



time available for productive work

operational responsibilities

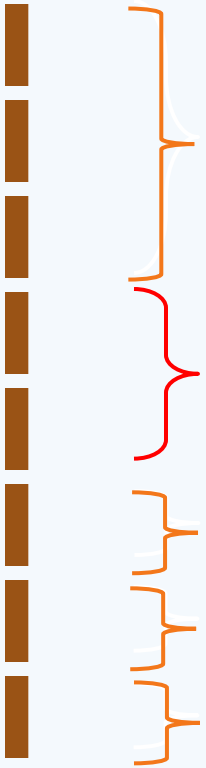
hacking, reading blogs, playing foosball

lunch and tea breaks

meetings, e-mail

Hours Per Day

"8-Hour Day"



time available for productive work

time committed to another team

hacking, reading blogs, playing foosball

lunch and tea breaks

meetings, e-mail

Available Time During Sprint

Sprint Length	**2 weeks**		
Workdays During Sprint	**8 days**		
Team Member	**Avail Days During Sprint***	**Avail Hours Per Day**	**Total Avail Hrs in Sprint**
Tracy	**8**		
Sanjay	**7**		
Phillip	**8**		
Jing	**5**		

*Net of holidays and other days out of the office

Sprint Planning

- Team understands the details of what the Product Owner has prioritized on the Product Backlog
- Team decides how much productive time it has available during the Sprint
- Team decides how many Product Backlog items it can commit to complete during the Sprint

Sprint Pre-Planning Meeting

Sprint Planning Meeting

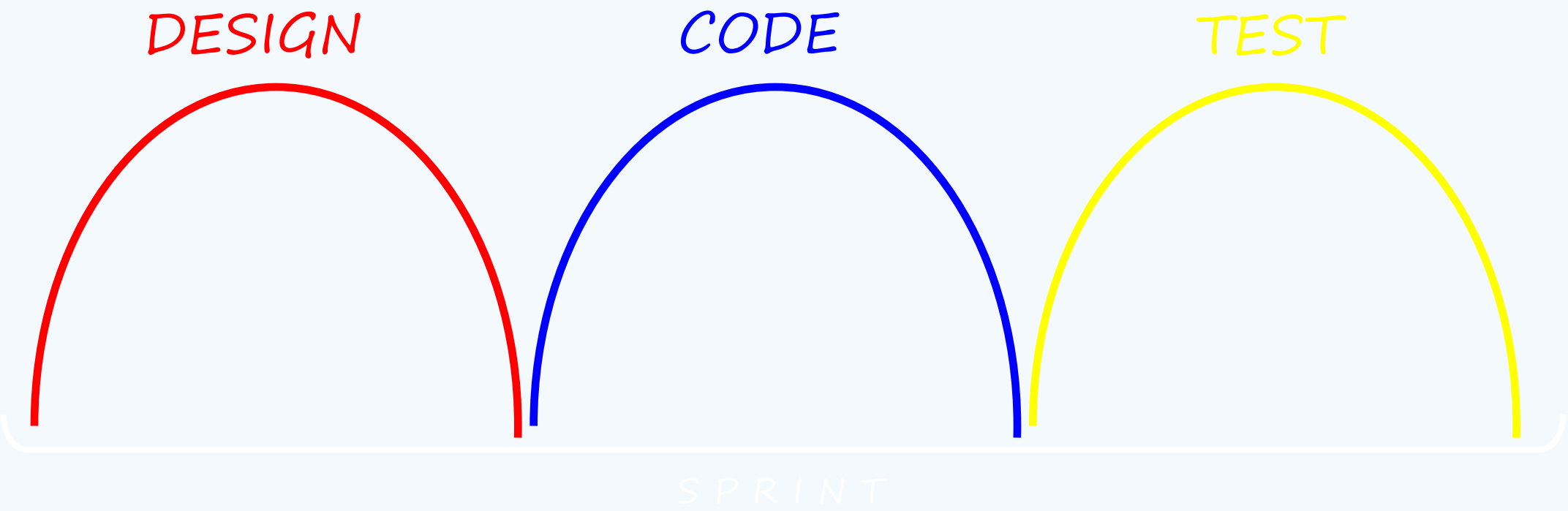
Product Backlog

Description	Rough Est. Size
Enable all users to place book in shopping cart (mocks and additional details are located here)	20
Upgrade transaction processing module (must be able to support minimum 500 transactions per second)	10
Investigate solutions for speeding up credit card validation (see target performance metrics located here)	20
Upgrade all servers to Apache 2.2.3	40
Diagnose and fix the order processing script errors (bugzilla ID 14823)	20
Enable all users to create / save wishlist	40
Enable all users to add and delete items on their wishlist	10

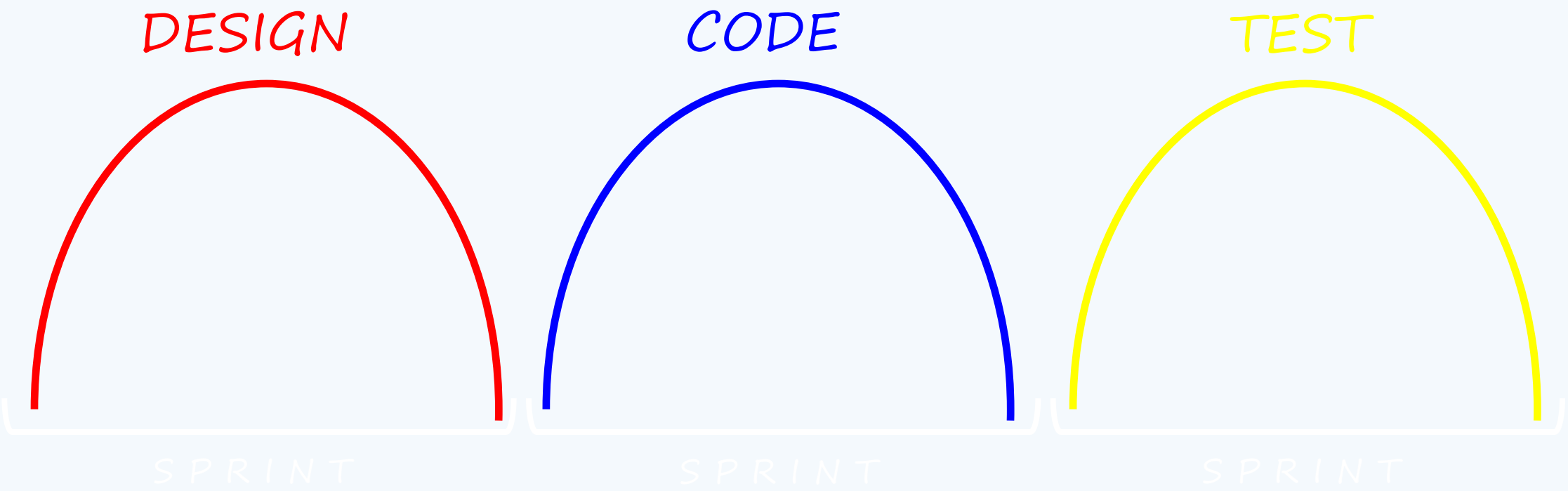
Getting to “Done”

Backlog Item	Task	Owner	Estimate

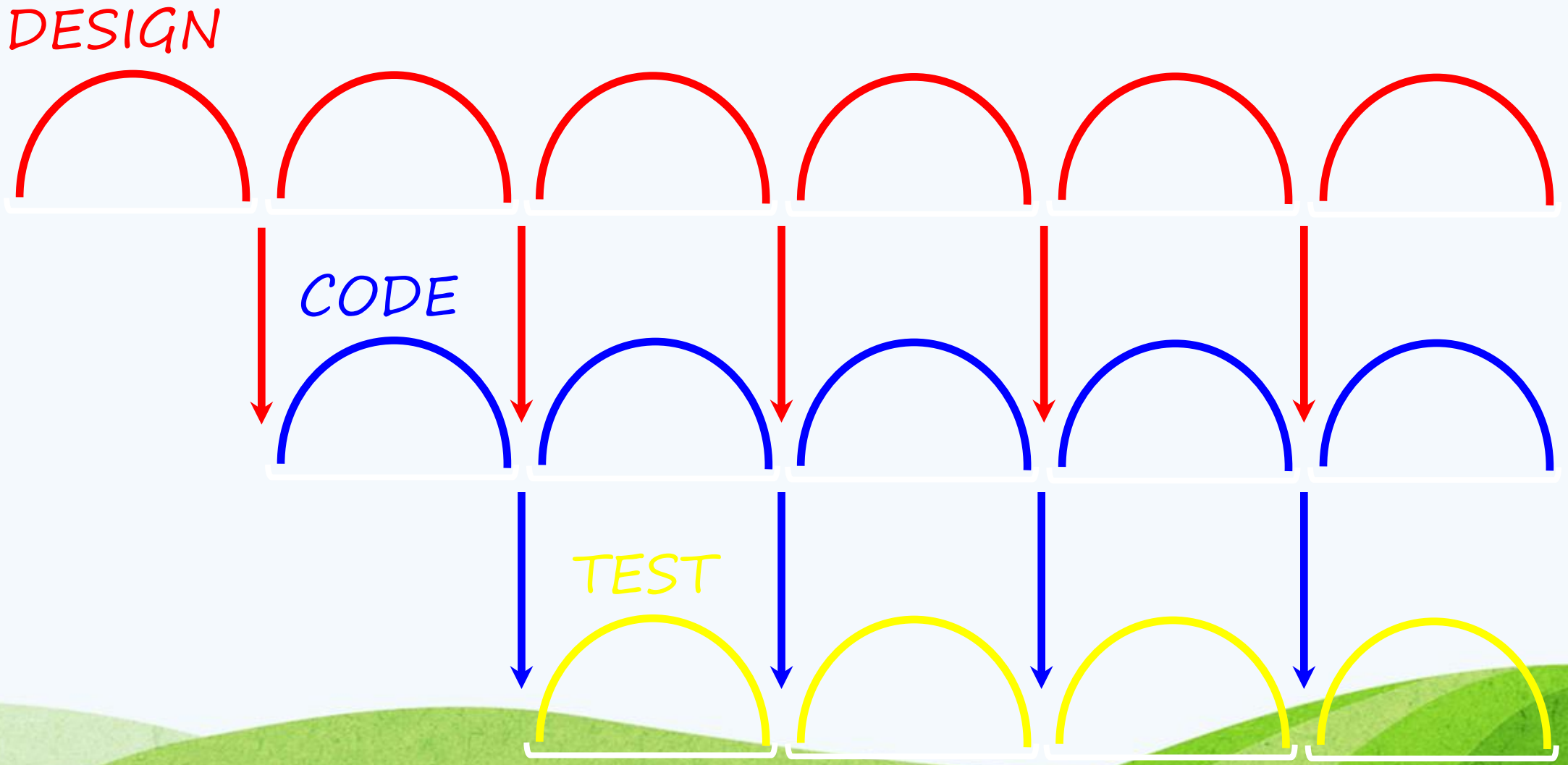
Getting To “Done”



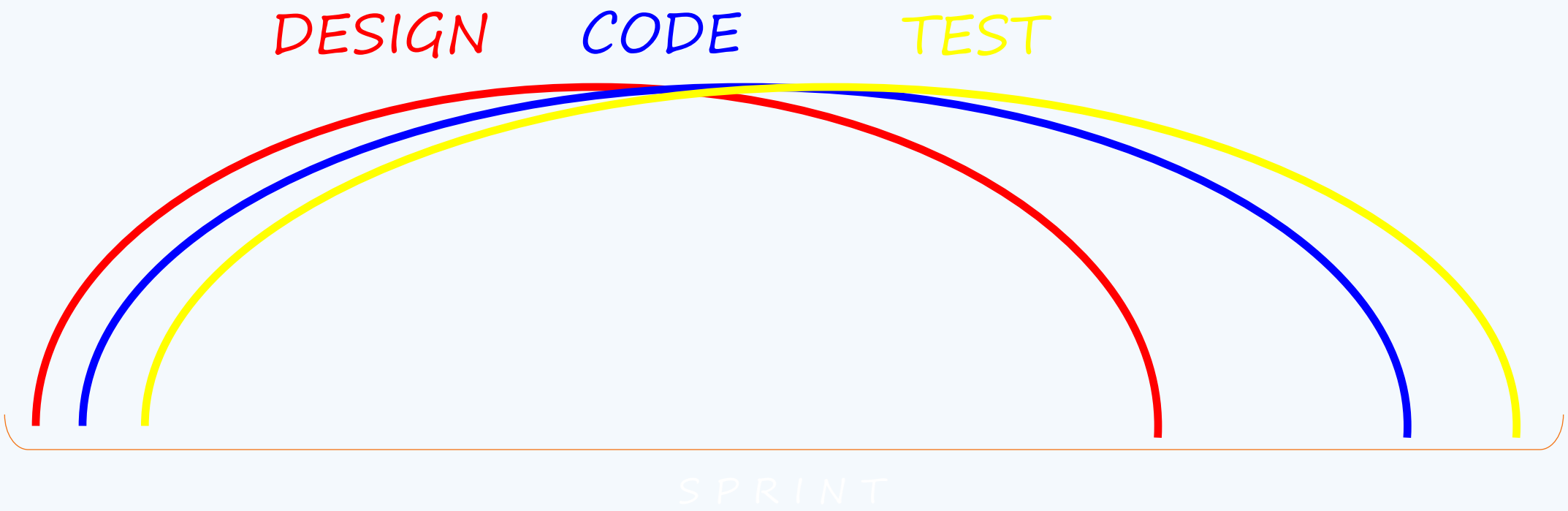
Getting To “Done”



Getting To “Done”



Getting To “Done”



The Definition of Done (DoD)

- Definition of Done (DoD) has been described as a tool for bringing transparency to the work a Scrum Team is performing.
- It is related more to the *quality* of a product, rather than its functionality.
- The DoD is usually a clear and concise list of requirements that a software Increment must adhere to for the team to call it complete. Until this list is satisfied, a product Increment is not done.
- During the Sprint Planning meeting, the Scrum Team develops or reconfirms its DoD, which enables the Development Team to know how much work to select for a given Sprint.
- The Definition of Done is not changed during a Sprint, but should change periodically between Sprints to reflect improvements the Development Team has made in its processes and capabilities to deliver software.

Creating the Sprint Backlog

Backlog Item	Task	Owner	Estimate
Enable all users to place book in shopping cart	Design business logic	Sanjay	4
	Design user interface		
	Set up shopping cart module		
	Implement back-end code		
	Implement front-end code		
	Unit testing		
	Regression testing		
	Documentation		

Available Time During Sprint

Sprint Length	**2 weeks**		
Workdays During Sprint	**8 days**		
Team Member	**Avail Days During Sprint***	**Avail Hours Per Day**	**Total Avail Hrs in Sprint**
Tracy	**8**	**4**	**32**
Sanjay	**7**	**5**	**35**
Phillip	**8**	**3**	**24**
Jing	**5**	**5**	**25**

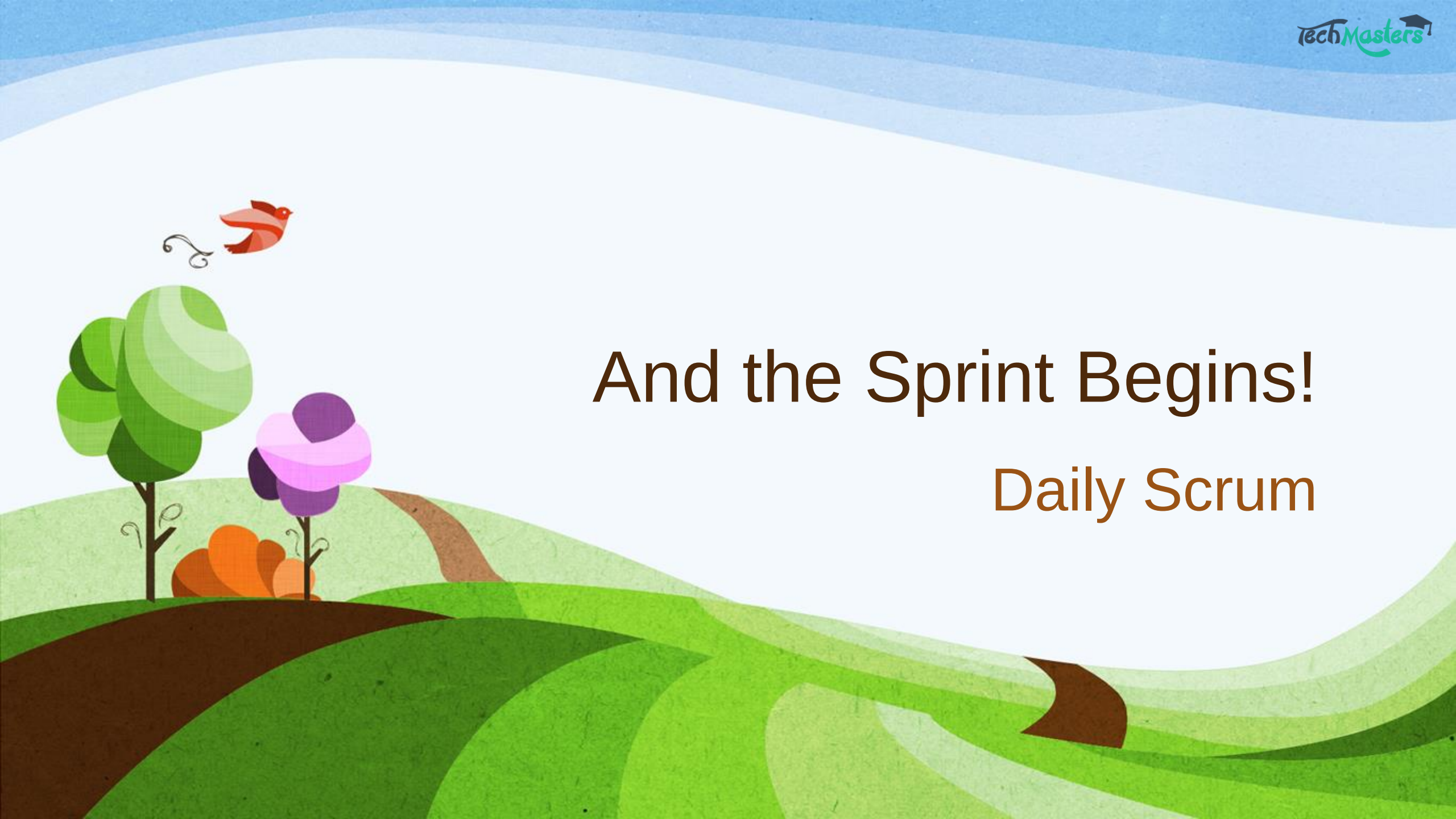
*Net of holidays and other days out of the office

Sprint Planning Ends when…

- Team’s available time is mostly committed
- Good idea to go through and make sure there’s full agreement on the commitment
- After the meeting, ScrumMaster turns the task list into the Sprint Backlog

The Sprint Backlog

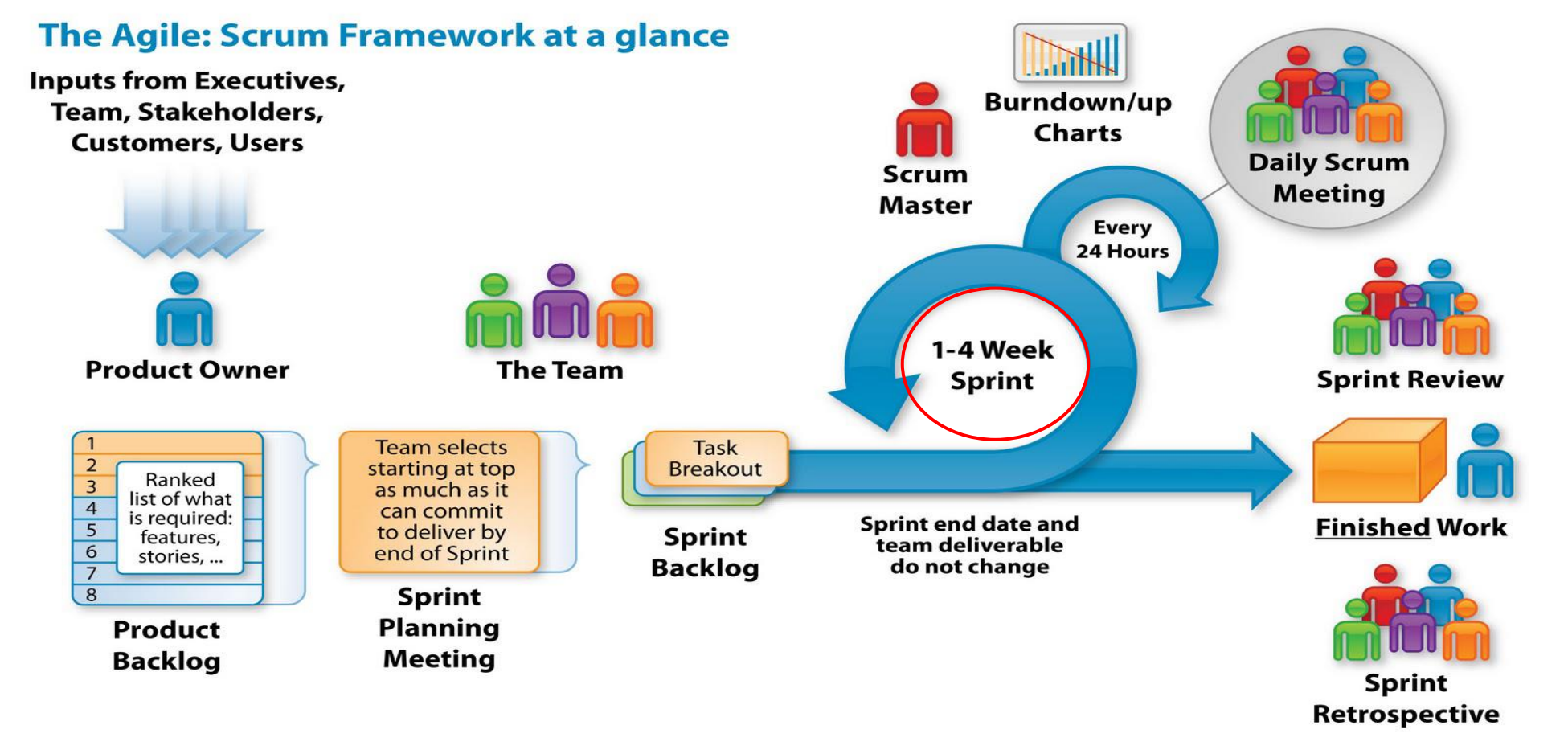
			Day of Sprint						
Backlog Item	Task	Owner	Initial Est.	1	2	3	4	5	6
Enable all users to place book in shopping cart	Design business logic	Sanjay	4						
	Design user interface	Jing	2						
	Implement back-end code	Tracy	2						
	Implement front-end code	Tracy	6						
	Complete documentation	Joe	8						
	Unit testing	Philip	4						
	Regression testing	Philip	2						
Upgrade transaction processing module	Implement back-end code	Tracy	5						
	Complete documentation	Joe	6						
	Unit testing	Philip	3						
	Regression testing	Philip	3						
	Total		214						



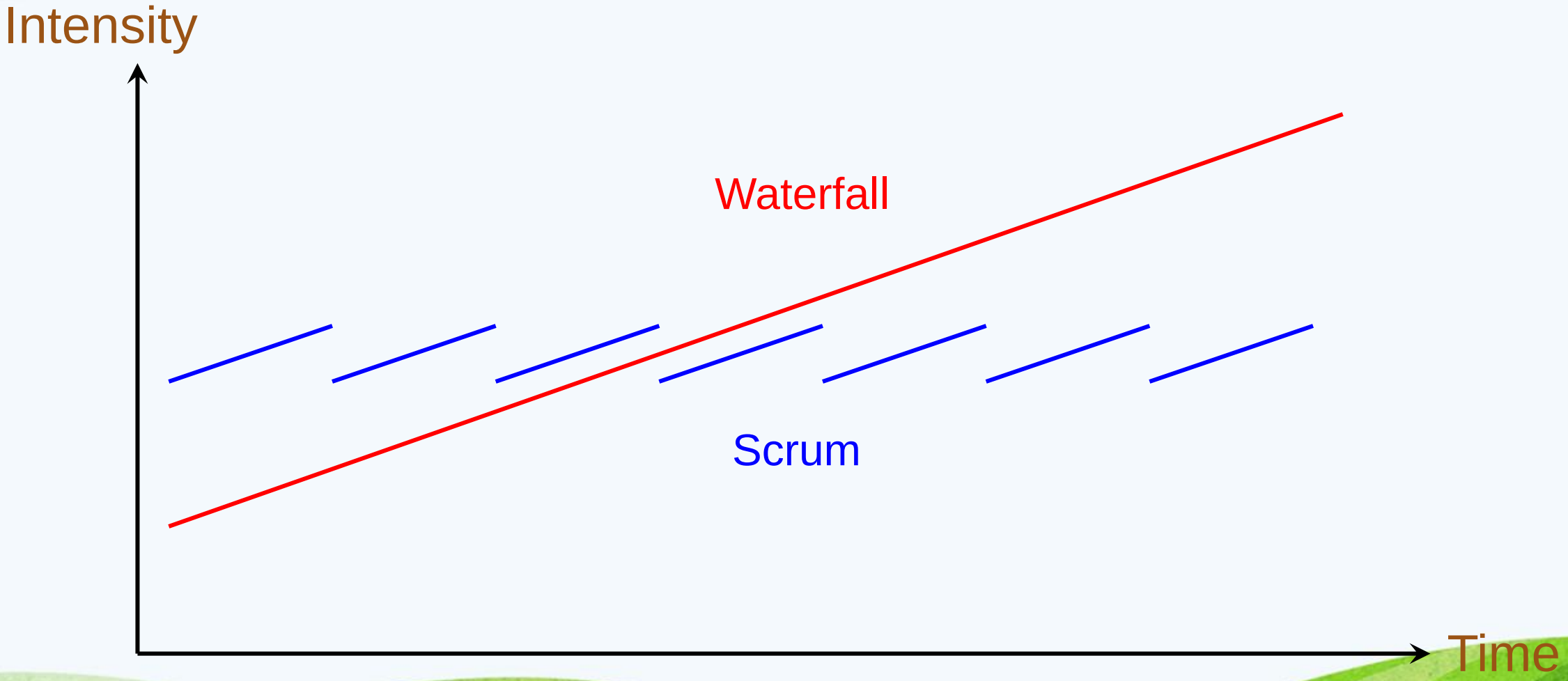
And the Sprint Begins!

Daily Scrum

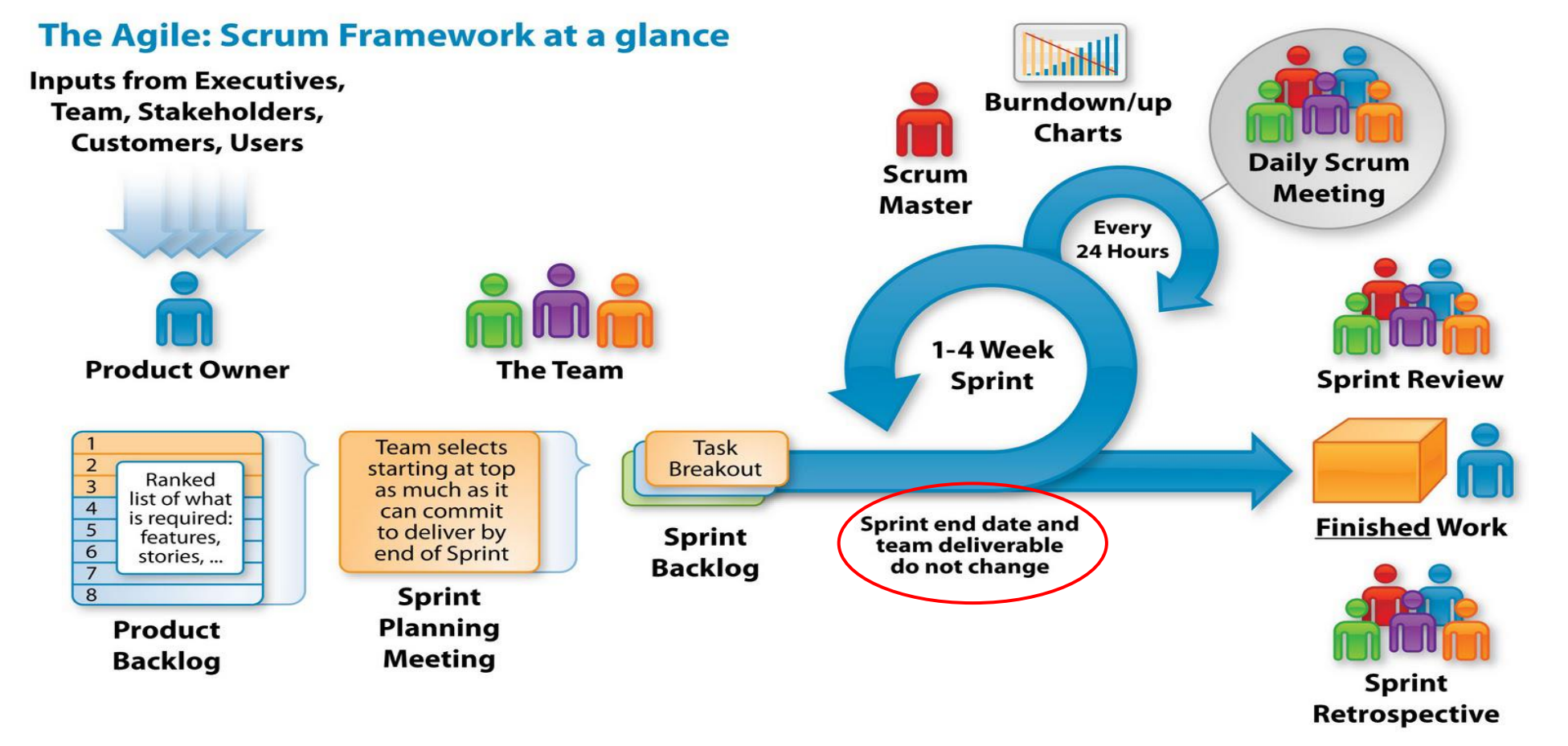
The Basics of Scrum



Intensity



The Basics of Scrum



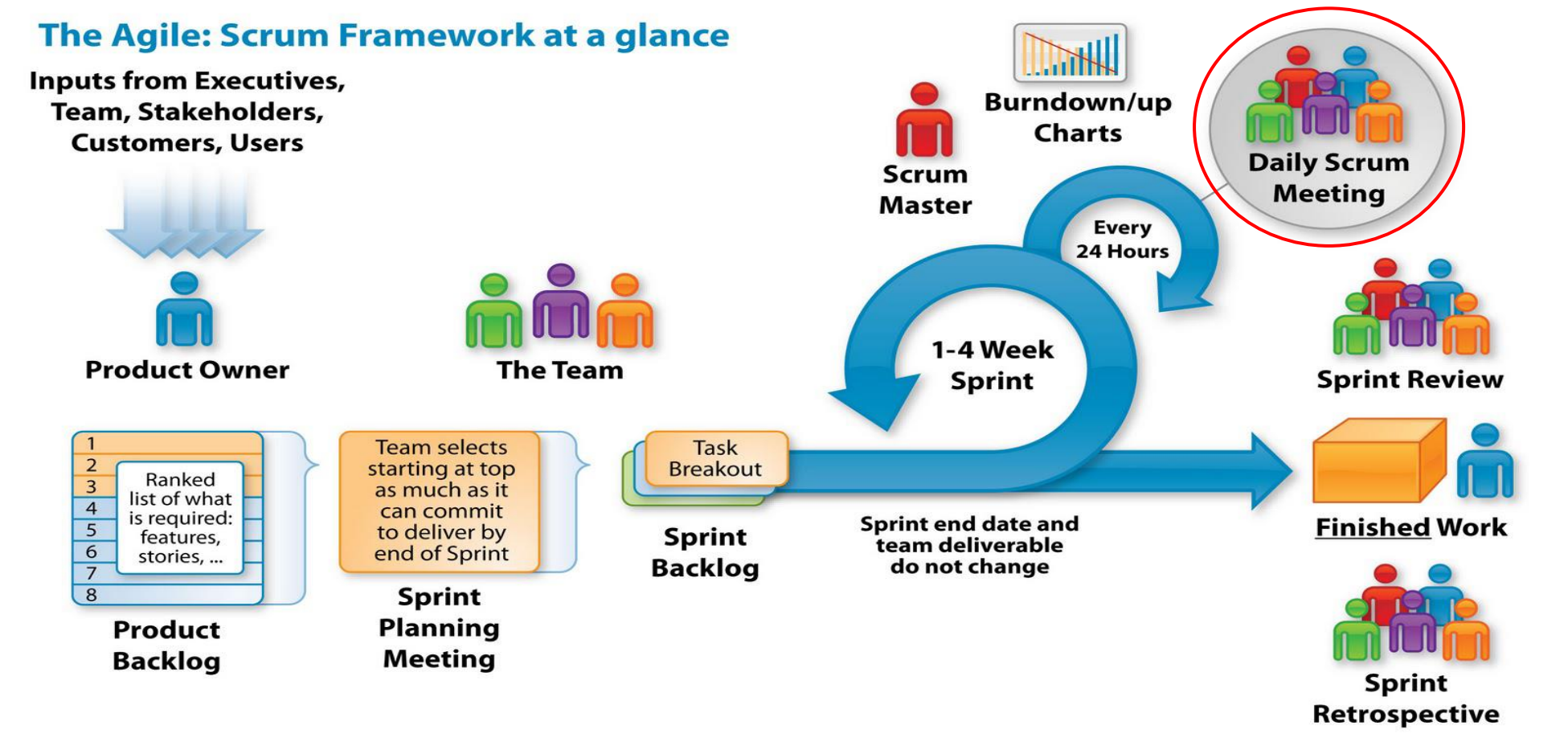
No Changes During Sprint

- No Changes to the Deliverable
  - Once team has committed, no changes to the deliverable
  - If something major comes up, Product Owner can terminate the Sprint and start new one
  - Details and clarifications will emerge during Sprint, but no new work or substantially changed work
- Difference between “change” and “clarification”
  - “If there’s any doubt, then it’s a change”

What if a change is required?

- Do not allow any changes in current sprint
- Move the item for next sprint
- Break the sprint and start a new sprint with modified backlog
- Product owner should be responsible for breaking the sprint or removing items from the sprint

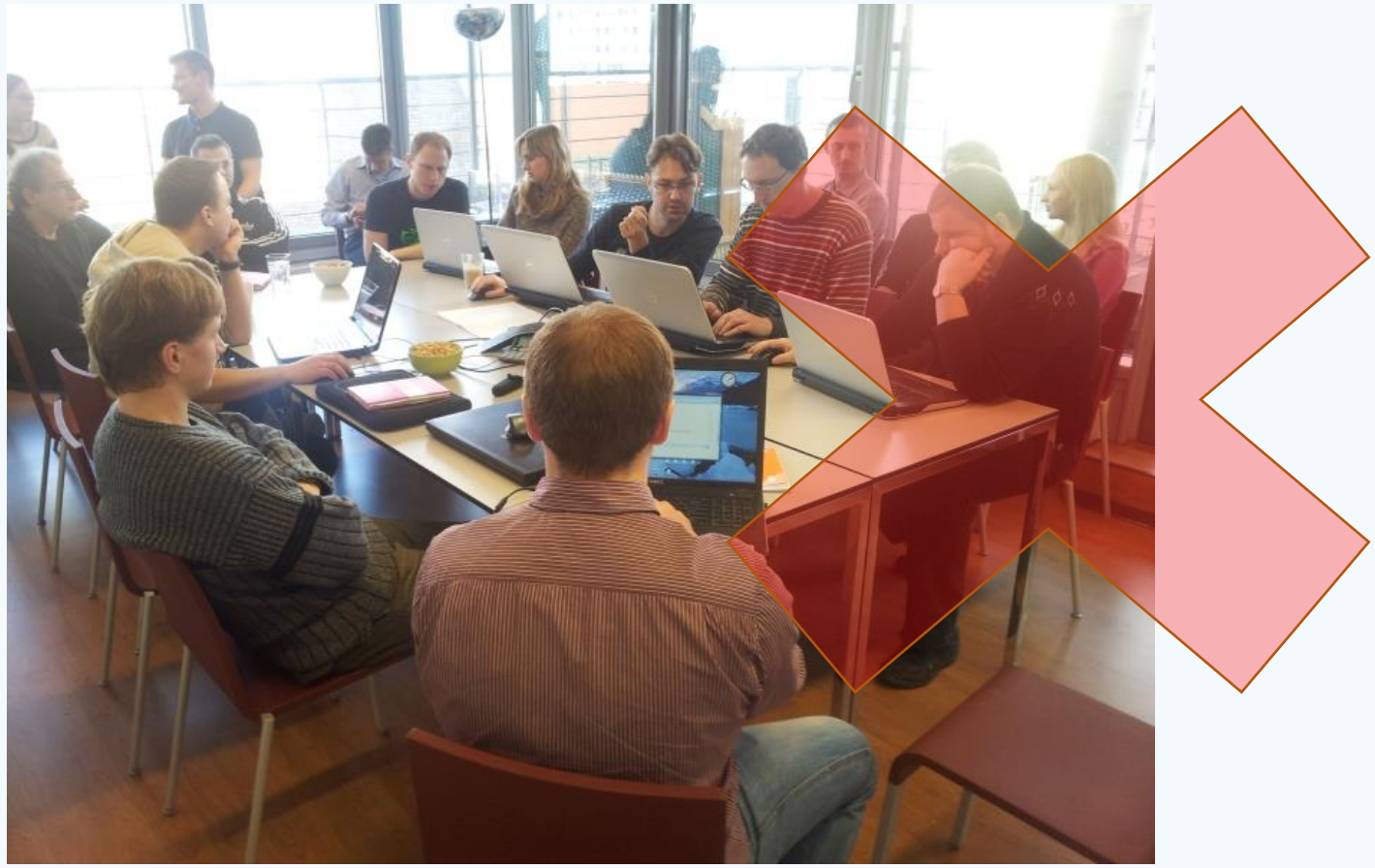
The Basics of Scrum



Daily Scrum Meeting

- Purpose of Daily Scrum Meeting
  - Keep team coordinated and up-to-date with each other
  - Surface impediments daily
- How it works
  - Every weekday
  - Whole team attends
  - Team chooses a time that works for everyone
  - Product Owner can attend, but doesn’t speak
  - Everyone stands in a circle, facing each other (not facing the SM)
  - Lasts 15 minutes or less
- Everyone reports 3 things only to each other
  - What was I able to accomplish since last meeting
  - What will I try to accomplish by next meeting
  - What are my blocks / problems / difficulties
- No discussion or conversation until meeting ends

Not allowed



Perfect Stand up

Updating the Sprint Backlog

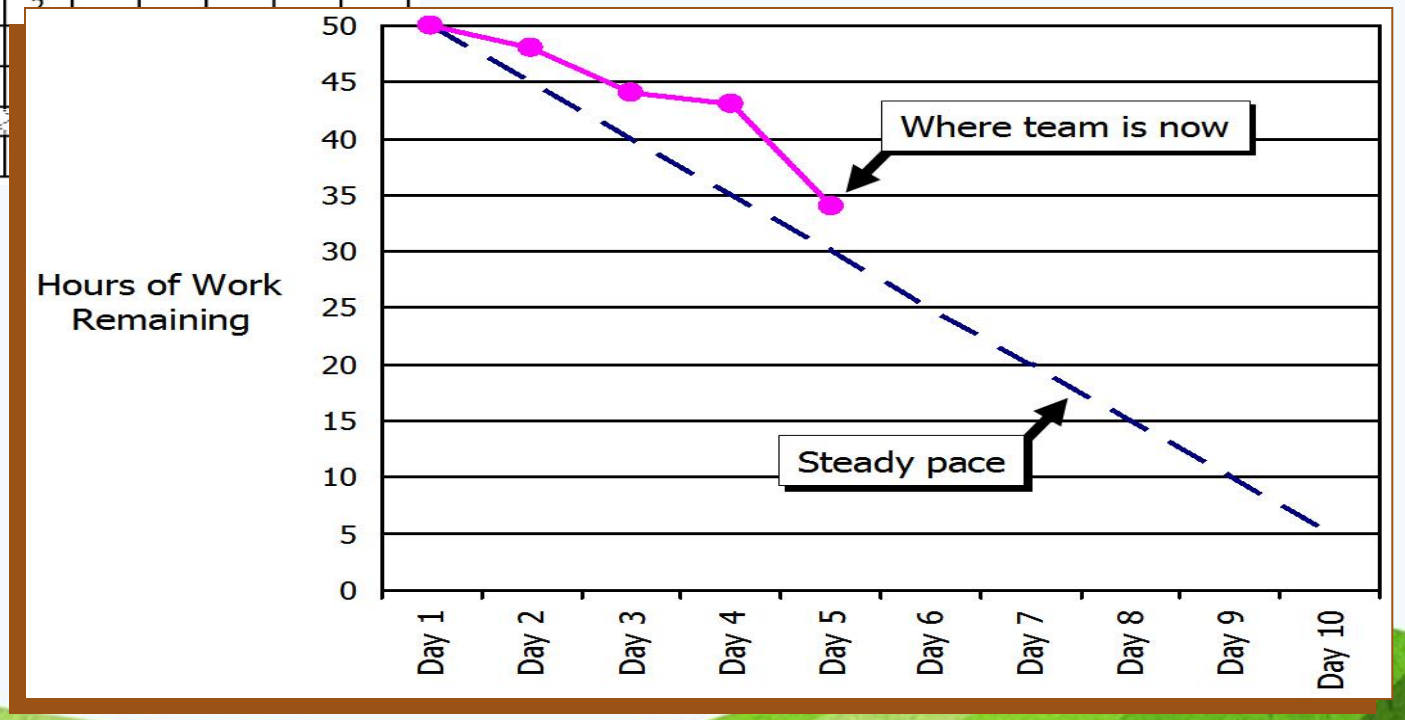
- After the Daily Scrum, team members update the hours remaining on the Sprint Backlog
- Also moves cards on the visual task board

Updating the Sprint Backlog

			Day of Sprint						
Backlog Item	Task	Owner	Initial Est.	1	2	3	4	5	6
Enable all users to place book in shopping cart	Design business logic	Sanjay	4						
	Design user interface	Jing	2	0					
	Implement back-end code	Tracy	2	2					
	Implement front-end code	Tracy	6	6					
	Complete documentation	Joe	8	6					
	Unit testing	Philip	4	3					
	Regression testing	Philip	2	2					
Upgrade transaction processing module	Implement back-end code	Tracy	5	5					
	Complete documentation	Joe	6	6					
	Unit testing	Philip	3	3					
	Regression testing	Philip	3	2					
	Total		214	210					

Burndown Chart

Task	Task Owner	Hours of Work Remaining on Each Day of the Sprint									
		Day 1	Day 2	Day 3	Day 4	Day 5	Day 6	Day 7	Day 8	Day 9	Day 10
Configure database and space IDs for Trac	Sanjay	4	4	3	1	0					
Use test data to tune the learning and action model	Jing	2	2	2	2	1					
Setup a cart server code to run as apache server	Philip	3	3	5	2	0					
Implement pre-Login Handler	Tracy	3	3	3	3	3					
Merge DCP code and complete layer-level tests	Jing	5	5	2	2	2					
Complete machine order for pRank	Jing	4	4	3	3						
Change DCP and reader to use pRank http API	Tracy	3	3	0	0						
	Total	**50**	**48**	**44**	**43**						



Task Board

TO DO	IN PROGRESS	DONE

Task Board

TO DO	IN PROGRESS	DONE
Task: Configure database and prepare DFD **Owner: Steve**		

Task Board

TO DO	IN PROGRESS	DONE
Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs		
Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs		
Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs		
Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs		
Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs		
Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs		
Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs		
Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs		
Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs		
Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs		
Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs		
Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs		
Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs		
Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs		
Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs		
Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs		
Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs		
Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs		
Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs		
Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs		
Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs		
Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs		
Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs		
Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs		

Task Board

TO DO	IN PROGRESS	DONE
Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	
Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	
Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	
Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	
Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	
Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	
Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	
Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	
Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	
Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	
Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	
Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	
Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	
Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	

Task Board

TO DO	IN PROGRESS	DONE
Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs
Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	
Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs
Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	
Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	
Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	
Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	
Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	
Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	

Task Board

TO DO	IN PROGRESS	DONE
Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs
Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs
Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs
Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs
Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs
Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs
Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs
Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs
Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs
Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs
Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs
Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs
Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs
Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs

Task Board

TO DO	IN PROGRESS	DONE
	Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs	Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs

Task Board

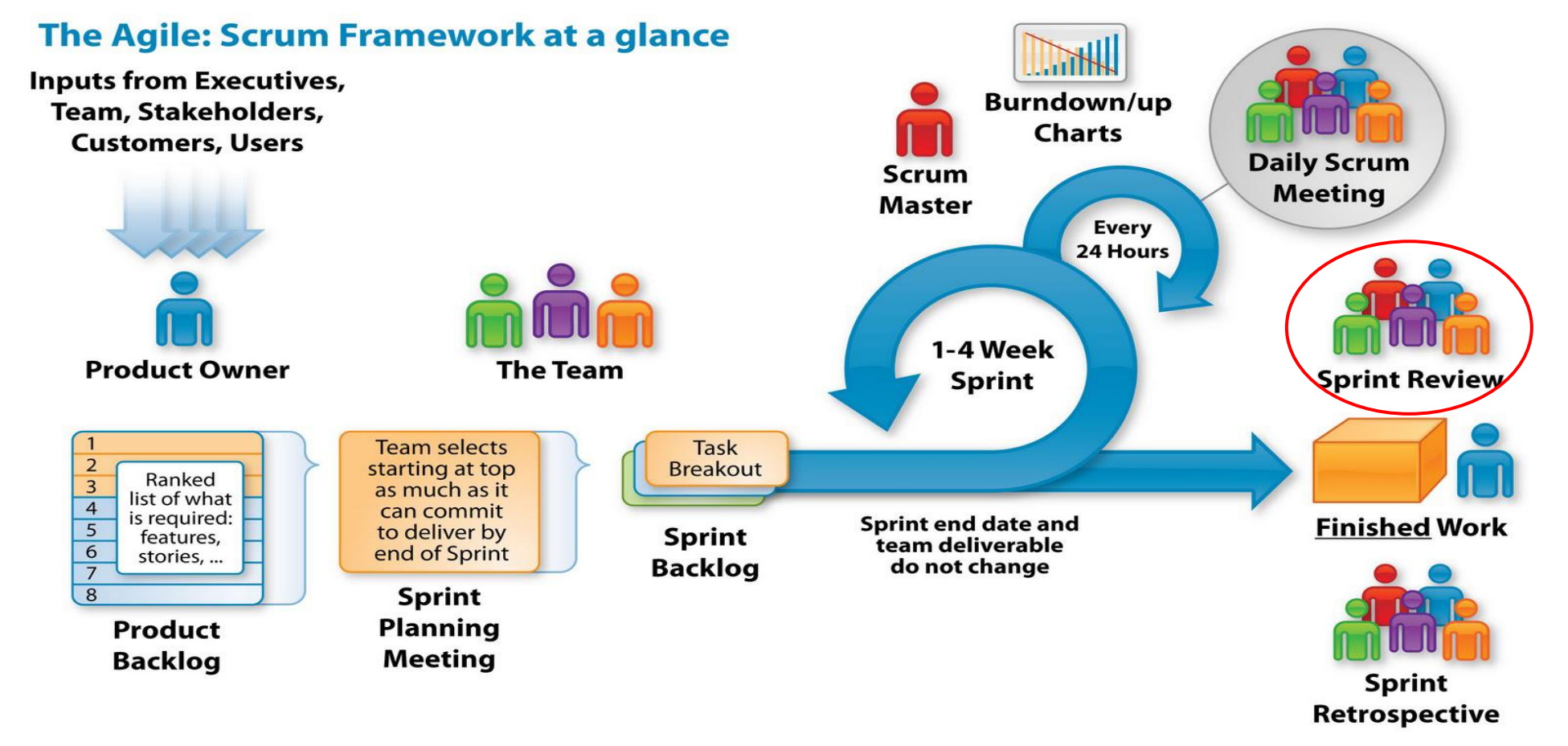
TO DO	IN PROGRESS	DONE
		Task: Configure database and SpaceIDs for Trac Owner: Sanjay Time Remaining: 4 hrs

Some important notes

- Use sticky paper on a board or use pin and paper on a soft board so that items are easy to move
- Can assign different color for team members, task type or themes
- Should be visible from a sitting position (for everyone)
- Item can move forward or backward
- Can be customized to have more than 2 columns

End of Sprint

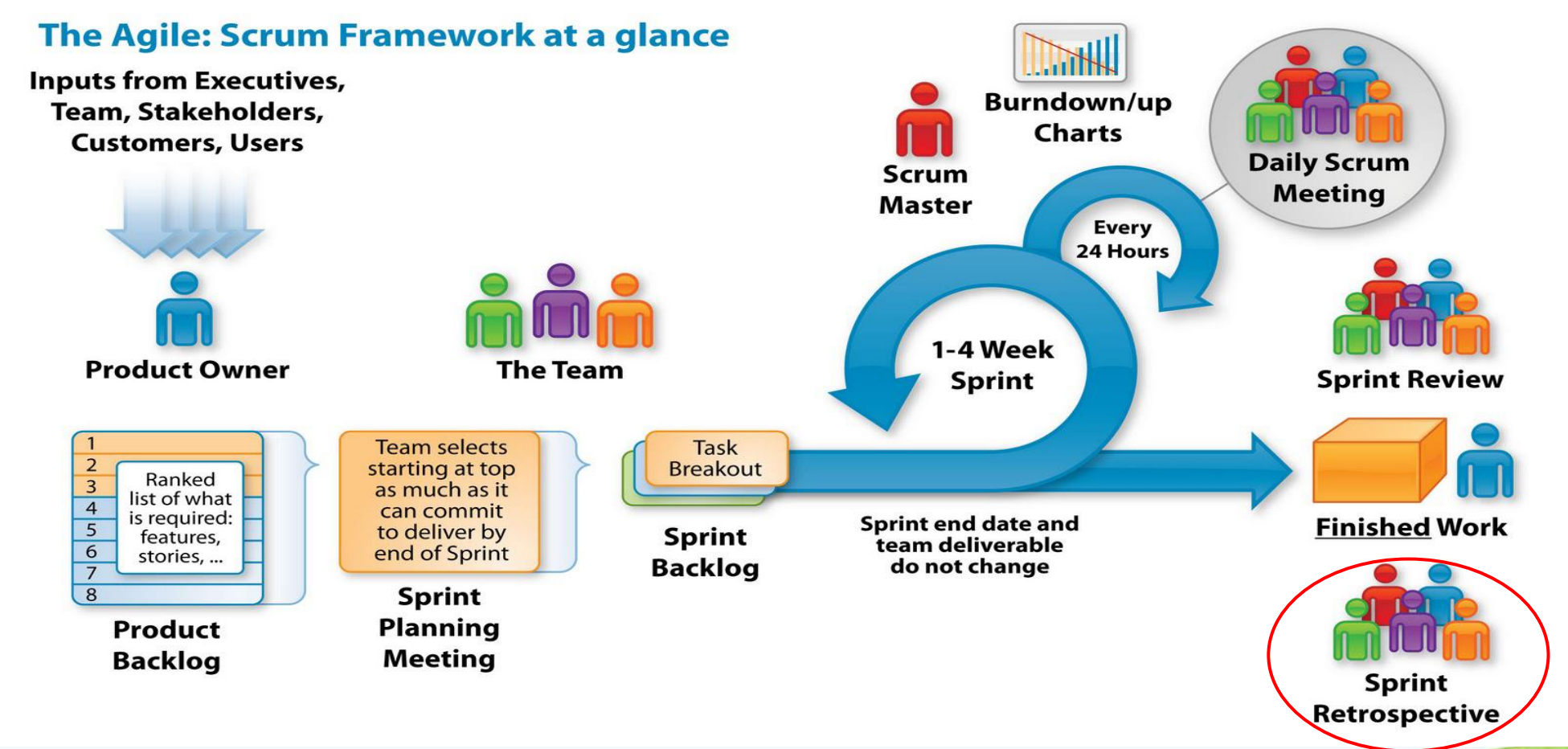
The Basics of Scrum



Sprint Review

- Purpose of the Sprint Review
  - Demo what the team has built
  - Make visible whether the team completed what they set out to
  - Generate feedback, which the Product Owner can incorporate in the Product Backlog
- Attended by Team, Product Owner, ScrumMaster, functional managers, and any other stakeholders
- A demo of what's been built, not a presentation about what's been built
  - no Powerpoints allowed!
- Usually lasts 1-2 hours
- Followed by Sprint Retrospective

The Basics of Scrum



Sprint Retrospective

- What is it?
  - 1-2 hour meeting following each Sprint Demo
  - Attended by Product Owner, Team, ScrumMaster
  - Usually a neutral person will be invited in to facilitate
  - What’s working and what could work better
- Why does the Retrospective matter?
  - Accelerates visibility
  - Accelerates action to improve

Sprint Retrospective – A Good Approach

- Create 3 large lists (whiteboard or flipchart)
  - What's working
  - What's could work better
  - Things to try in the next Sprint
- Go around the room, and give each person an opportunity to add 1 or more items to the 3 lists
- If people agree with something already on the lists, put a tick mark next to them
- Select a subset of the "Things to try..." list to try in the next Sprint (ScrumMaster responsible for tracking this)

WHAT WORKED

- Team felt more focused than before on hitting its goals
- Sense of commitment for the team was higher
- Team had a better sense of where it was in the Sprint because of the burndown chart
- The Daily Scrum Meeting improved Team communication during the Sprint
- Good idea came out during the Sprint Review

WHAT DIDN’T WORK

- We didn’t finish what we committed to in the time available
- Our estimations were way off
- We didn’t think through the dependencies between tasks
- The coordination between developers and testers wasn’t very good
- Our daily Scrum meeting didn’t start on time
- The food in the cafeteria was really bad
- There were a lot of outside distractions during the Sprint
- Blocks didn’t get removed quickly enough by the ScrumMaster
- There were too many bugs yet to be fixed at the end of the Sprint
- Team didn’t really work together as a team – still felt like a group of individuals

Roles-Rituals/Ceremonies-Artifacts

Roles	Rituals	Artifacts
•Product Owner •Scrum Master •Team	•Pre-planning meeting •Sprint planning meeting •Daily stand up meeting •Sprint review •Sprint retrospective	•Product backlog •Sprint backlog •Burn down chart •Visual task board

Agile Estimation & planning

Agile Estimation Games

Planning Poker

Getting Started

1. Everyone creates cards with following values on them: ?, ½, 1, 2, 3, 5, 8, 13, 20, 50 (WRITE LARGE USING MARKER!)

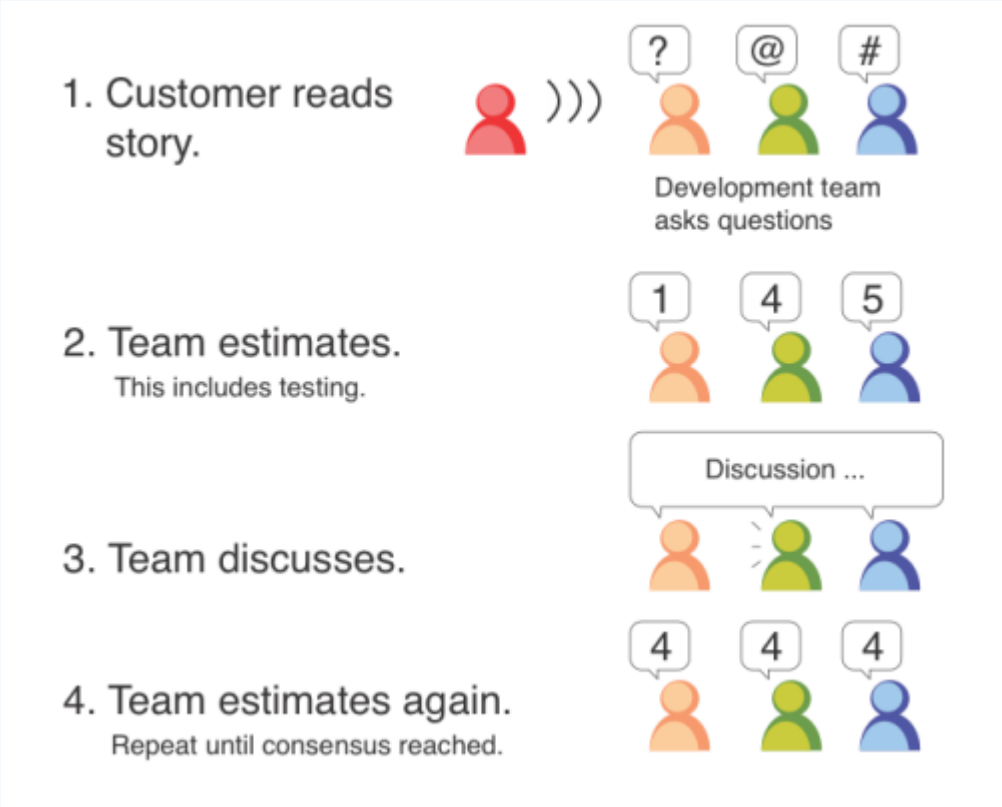
2. Team agrees on a backlog item to use as a “baseline” or “benchmark”, and gives it a size of “5” (item should be neither biggest nor smallest on list)

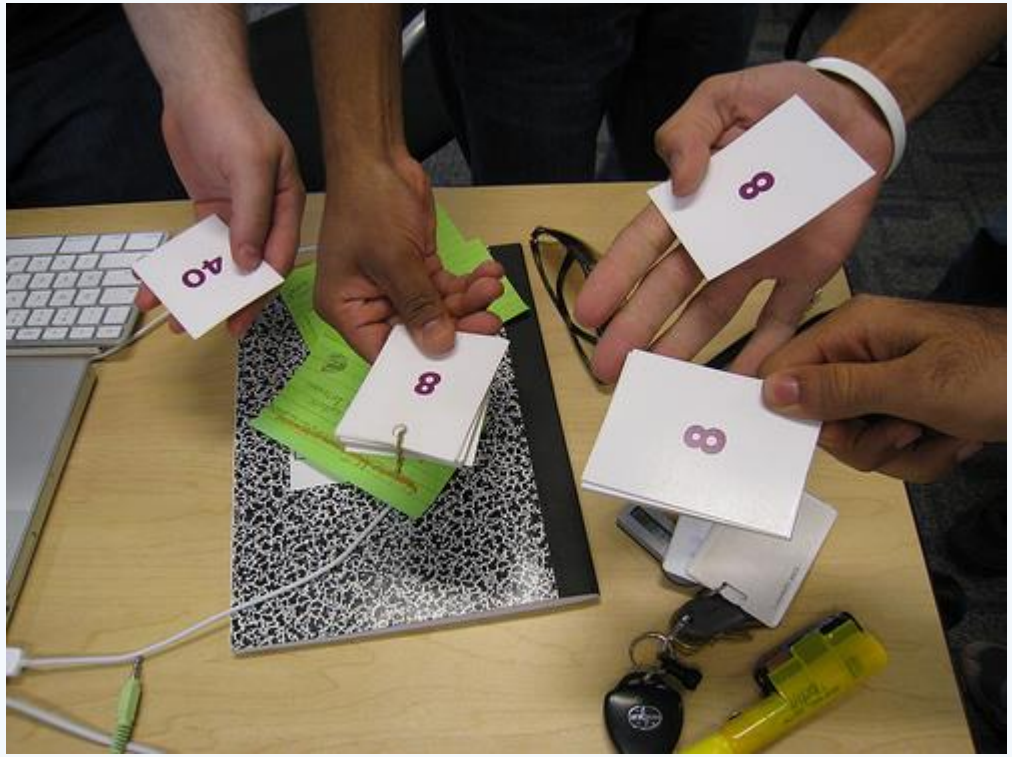
3. Size = Effort x Complexity x Uncertainty

Planning Poker

For each Product Backlog item:

1. Each person decides their size estimate relative to the items that have already been estimated, and privately selects a card
2. Everyone shows their cards at the same time (ScrumMaster says “1-2-3-Show”)
3. If estimates vary significantly, high and low estimators briefly explain
4. Repeat steps 3-5 until estimates stop converging
5. Decide estimate for backlog item
6. Move to next backlog item


1. Customer reads story.
?
@
#
Development team asks questions
2. Team estimates.
This includes testing.
1
4
5
Discussion ...
3. Team discusses.
4. Team estimates again.
Repeat until consensus reached.
4
4
4



Product Backlog

Description	Rough Est. Size
Enable all users to place book in shopping cart (mocks and additional details are located here)	20
Upgrade transaction processing module (must be able to support minimum 500 transactions per second)	10
Investigate solutions for speeding up credit card validation (see target performance metrics located here)	20
Upgrade all servers to Apache 2.2.3	40
Diagnose and fix the order processing script errors (bugzilla ID 14823)	20
Enable all users to create / save wishlist	40
Enable all users to add and delete items on their wishlist	10

Velocity – Based on Historical Data

Team's Recent Sprints

	backlog item	*rough est size*
~~Sprint 1~~	~~implement the chqhvxd functionality~~	20
	~~fix the bcmitla bug~~	50
	~~rearchitect the kicirpf module~~	20
~~Sprint 2~~	~~fix the gsbcicj bug~~	60
	~~rearchitect the rndbjtv module~~	20
	~~investigate the vbiihdv solution~~	20
	~~upgrade the aaninmn servers~~	10
	~~implement the bqeqoeu functionality~~	10
~~Sprint 3~~	~~upgrade the etkfiwj servers~~	40
	~~implement the aahmiqr functionality~~	40
	~~fix the cmafepj bug~~	20

90 points

120 points

100 points

~105 size points per Sprint

Velocity Calculation

SPRINT 1 | SPRINT 2 | SPRINT 3 | SPRINT 4 | SPRINT 5

Initial Estimate Of Velocity at Project Start

Refinement of Estimate based on 1 Sprint of Actual Data

Recalculation based on 2 Sprints of Actual Data

Recalculation based on 3 Sprints of Actual Data

Recalculation based on 4 Sprints of Actual Data

How to Plan a Release in Scrum

Calculate velocity then apply to the Product Backlog:

Team's velocity is ~105 points per Sprint

Therefore, in 6 Sprints, the team should be able to complete

6 x 105 = 630 points worth of Product Backlog

id	backlog item	rough est size
1	rearchitect the ahkivgb module	10
2	investigate the xxattjw solution	10
3	upgrade the eihfotj servers	60
4	rearchitect the okgdcps module	60
5	investigate the ehgvehf solution	10
6	upgrade the jktqfkx servers	30
7	implement the ukjqens functionality	60
8	fix the aorquve bug	20
9	fix the jsvdpwj bug	50
10	rearchitect the blijwic module	30
11	investigate the hhnkfit solution	60
12	upgrade the cqexrdj servers	10
13	implement the uusfguk functionality	30
14	fix the ifkwxtu bug	50
15	rearchitect the pmcqchw module	60
16	investigate the kppjevi solution	20
17	upgrade the vgloxmg servers	50
18	implement the xjcqafm functionality	60
19	fix the xvuheoi bug	10
20	rearchitect the pqgwboi module	50
21	investigate the ittdeme solution	60
22	upgrade the uljfpmc servers	50
23	implement the jqqxkuq functionality	30
24	fix the inoqmvt bug	20
25	rearchitect the ocbwhxk module	10
26	investigate the tutvqil solution	60
27	upgrade the runlbhq servers	30
28	investigate the kppjevi solution	20
29	upgrade the vgloxmg servers	50
30	implement the xjcqafm functionality	60

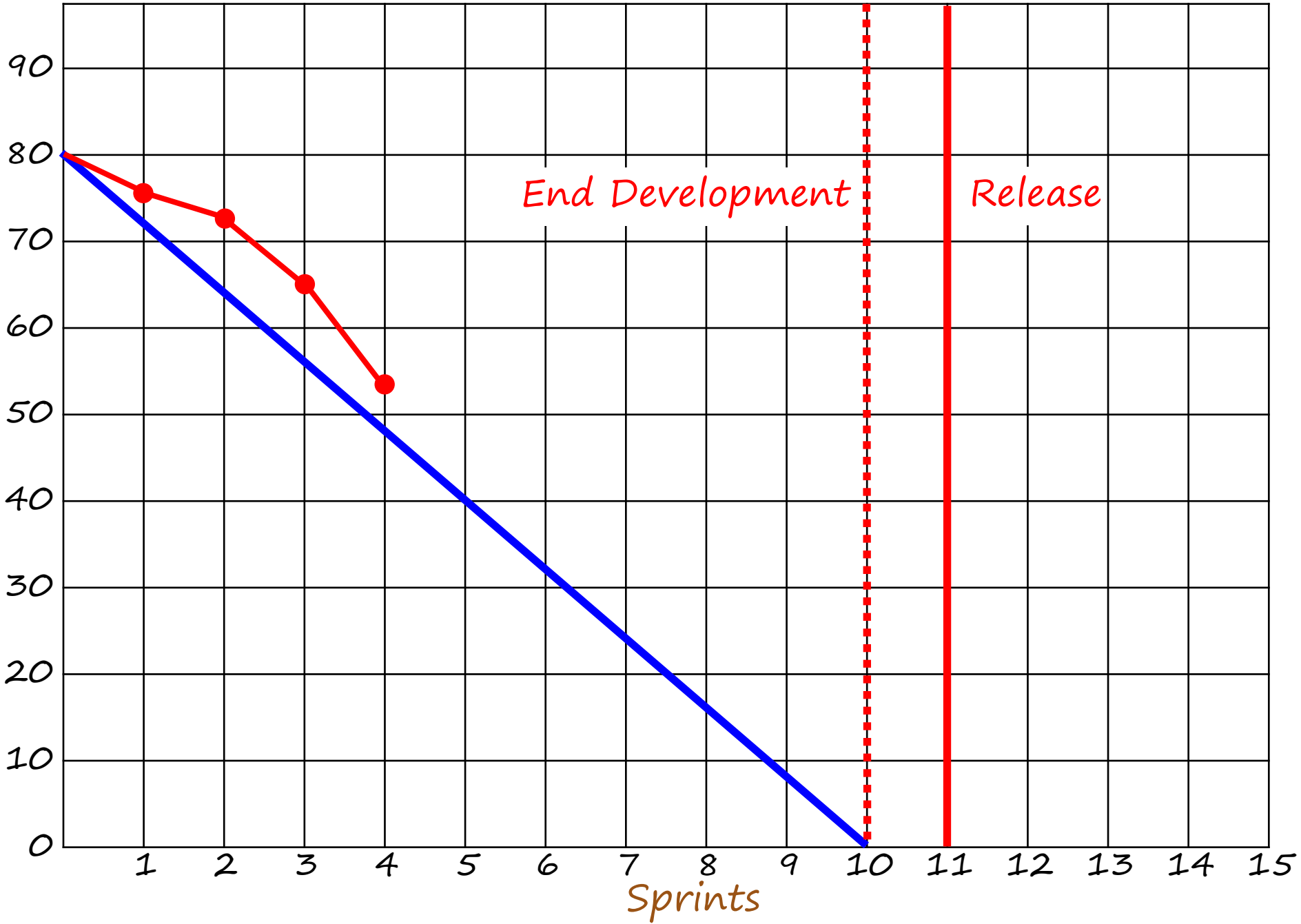
This is 620 points of Product Backlog

Release Planning

- A very high-level plan for multiple Sprints (e.g. three to twelve iteration) is created during the Release planning.
- It is a guideline that reflects expectations about which features will be implemented and when they are completed. It also serves as a base to monitor progress within the project.
- Releases can be intermediate deliveries done during the project or the final delivery at the end.
- To create a Release Plan the following things have to be available:
  - A prioritized and estimated Scrum Product Backlog
  - The (estimated) velocity of the Scrum Team
  - Conditions of satisfaction (goals for the schedule, scope, resources)

- Depending on the type of project (feature- or date-driven) the release plan can be created in different ways:
  - If the project is feature-driven, the sum of all features within in a release can be divided by the expected velocity. This will then result in the number of sprints needed to complete the requested functionality. Selected items from product backlog for prioritize releases
  - If the project is date-driven we can simply multiply the velocity by the number of Sprints and we'll get the total work that can be completed within the given timeline.
- Like the Scrum Product Backlog the Release plan is not a static plan. It will change during the whole project when new knowledge is available and e.g. entries in the Scrum Product Backlog are changed and re-estimated. Therefore the Release Plan should be revisited and updated in regular intervals, e.g. after each Sprint.

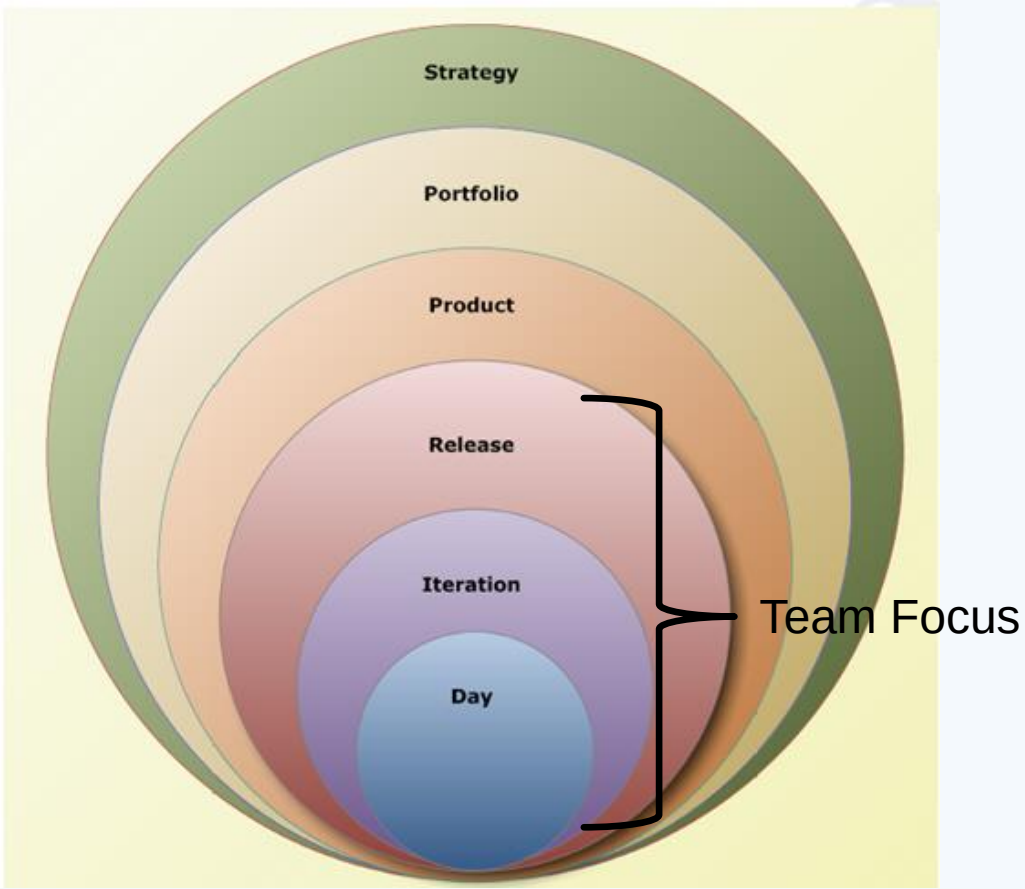
Release Burndown Chart

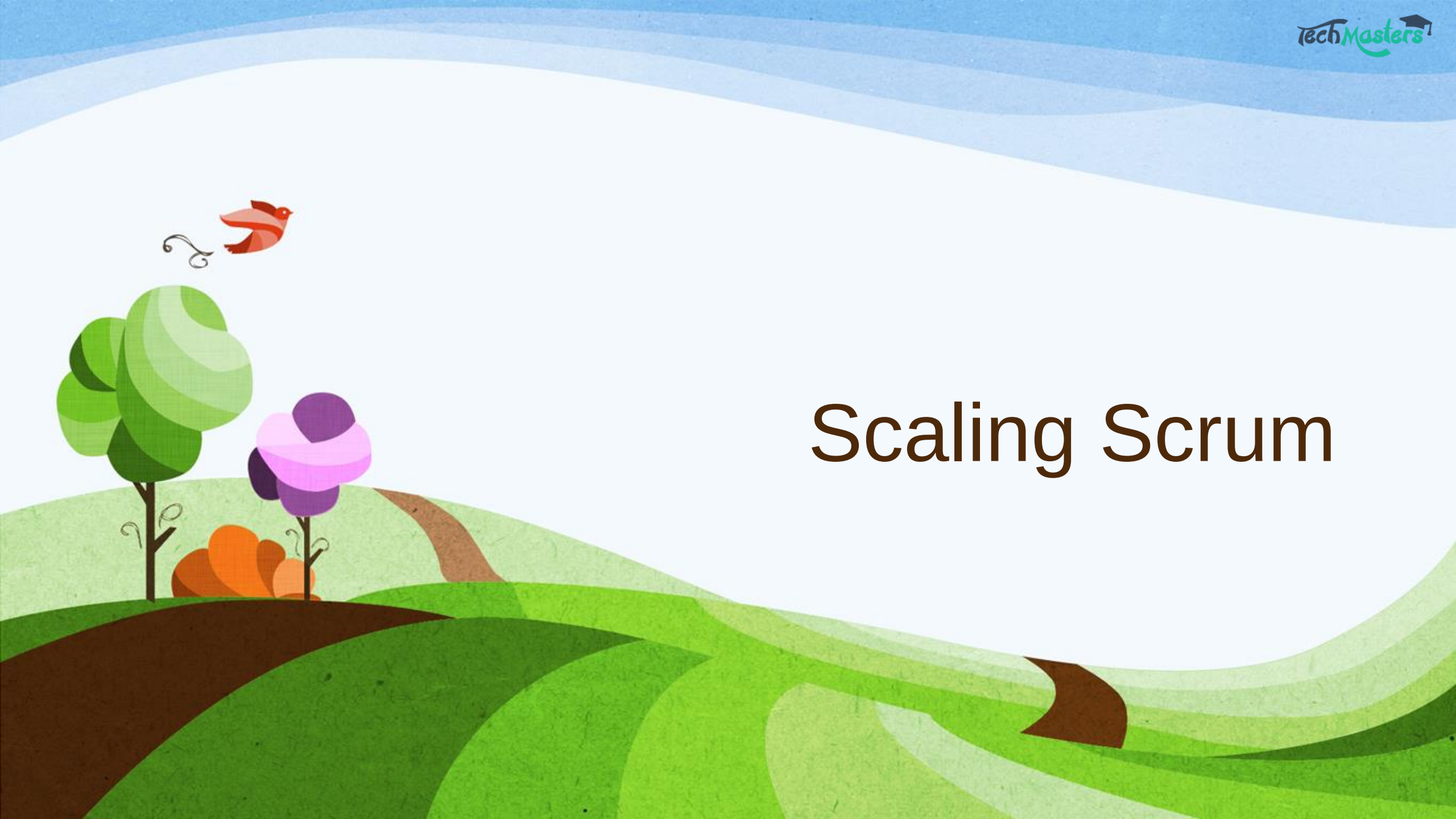


MoSCoW

- **MoSCoW** is a technique used in management, business analysis, and software development to reach a common understanding with stakeholders on the importance they place on the delivery of each requirement - also known as *MoSCoW prioritization* or *MoSCoW analysis*.
- M: Must - Important ones with highest priority
- S: Should - should be included if it is possible
- C: Could - nice to have but not necessary
- W: won’t - not be implemented in a given release

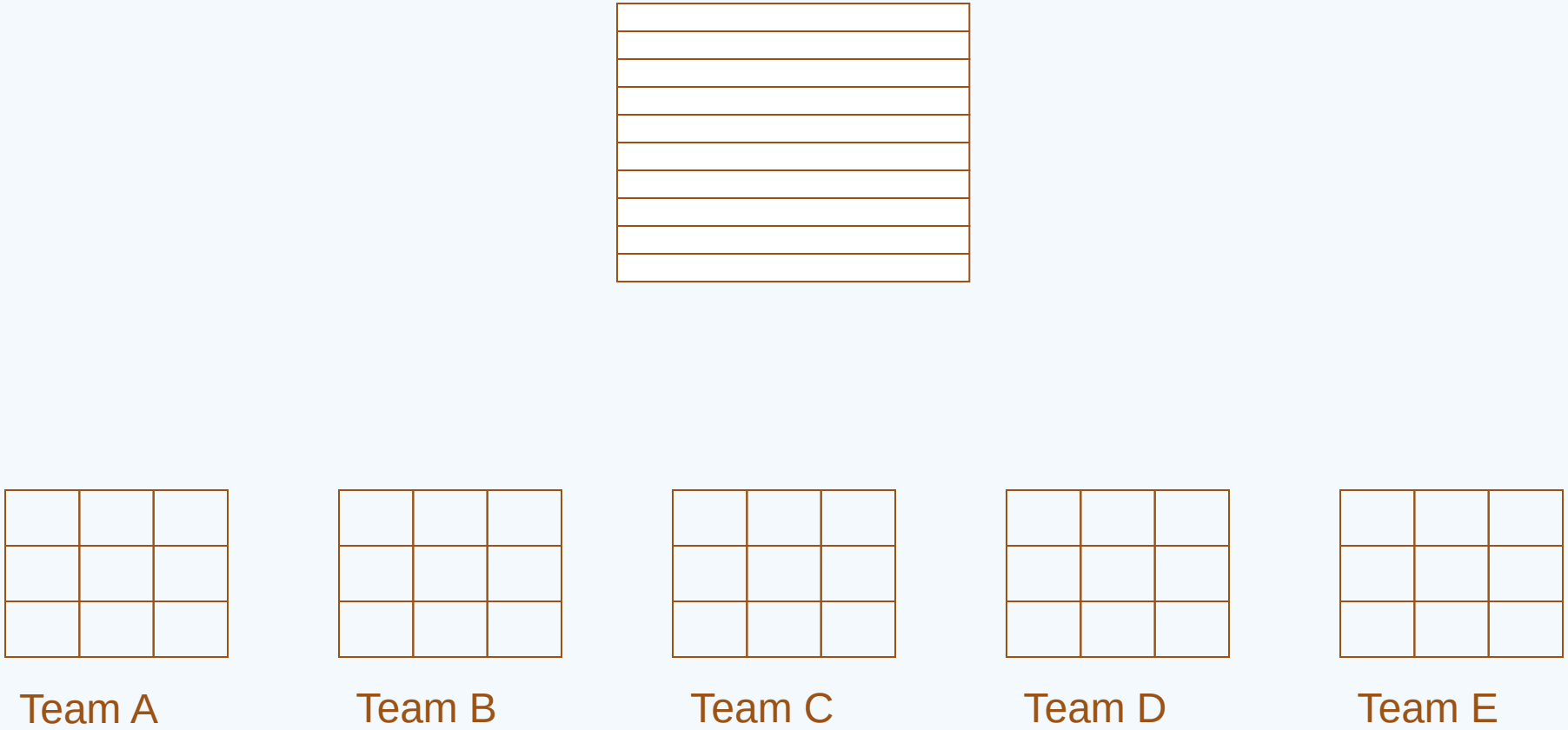
Planning Onion



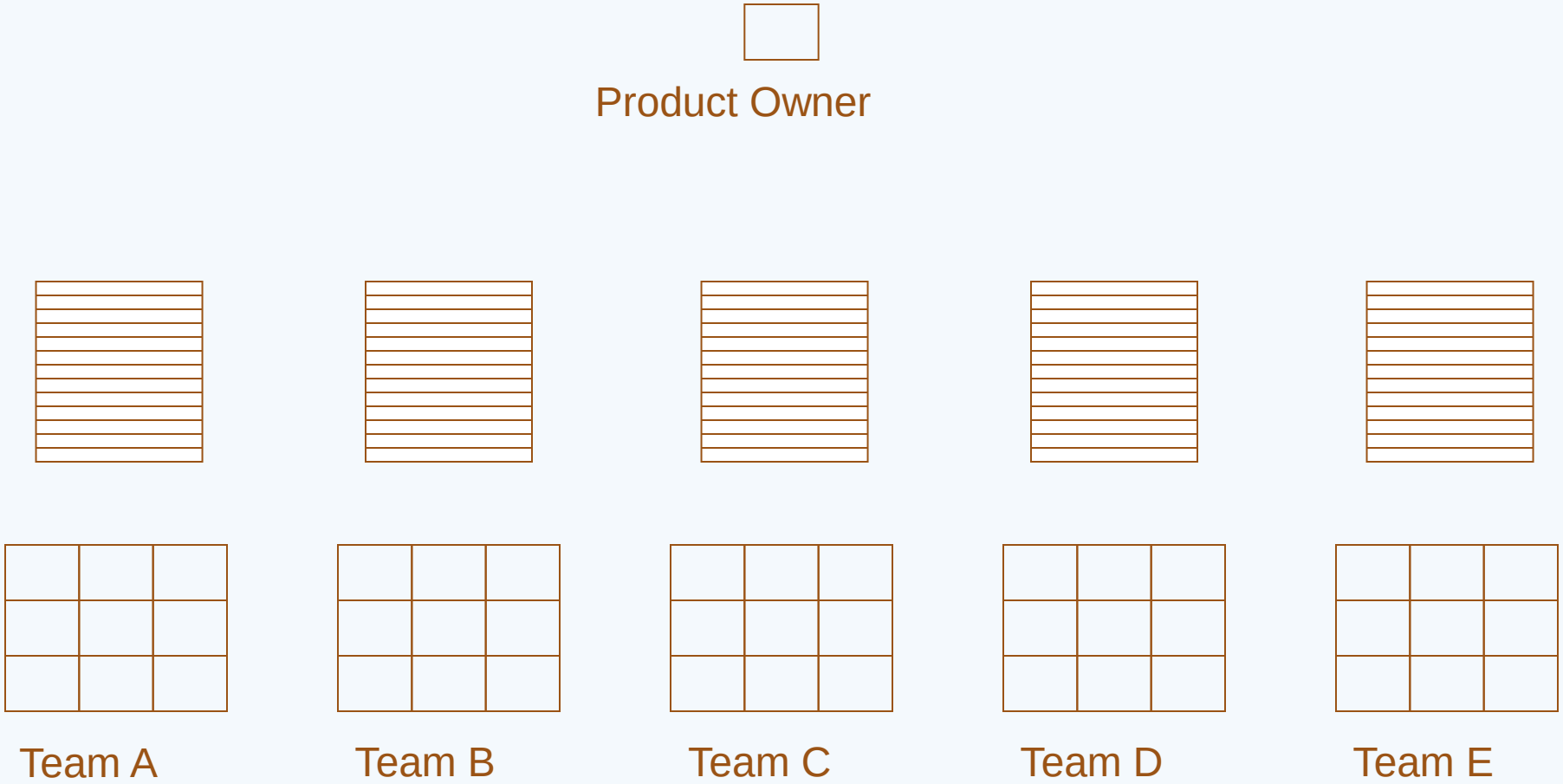


Scaling Scrum

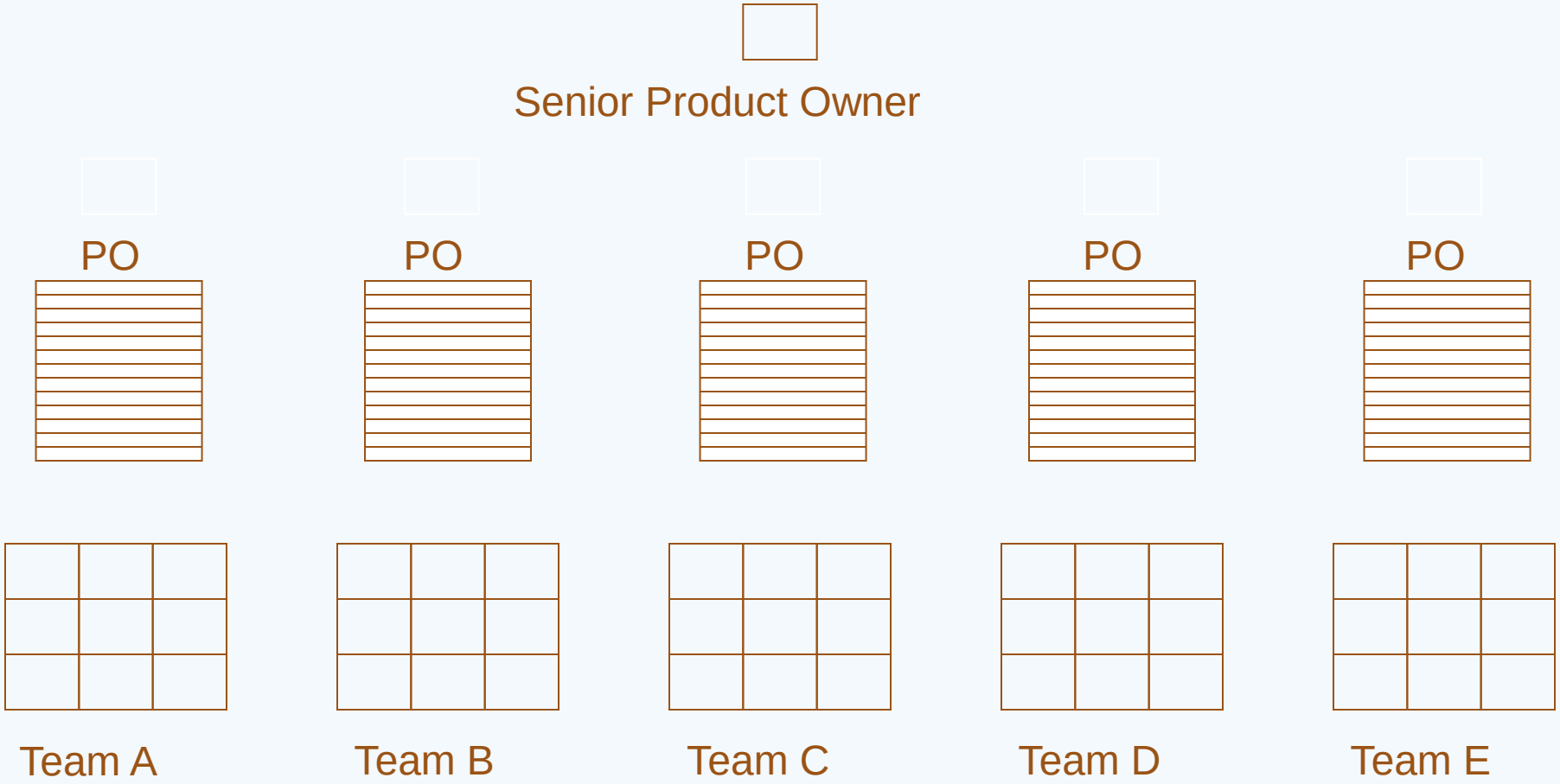
Scaling Scrum



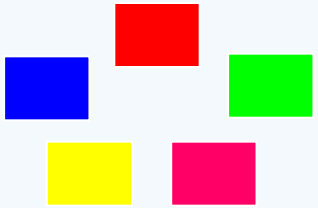
Scaling Scrum



Scaling Scrum



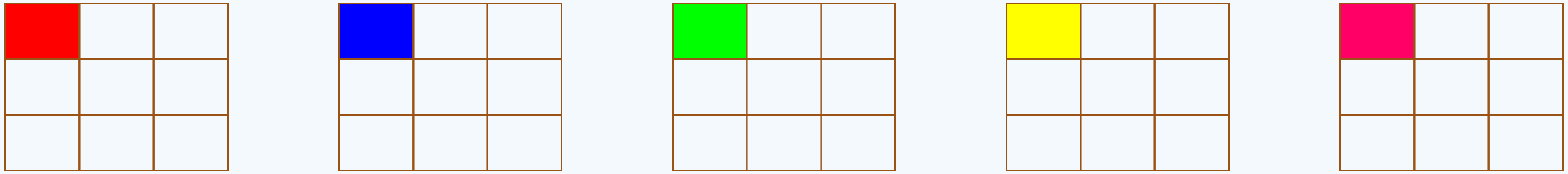
During the Sprint



Scrum of Scrums

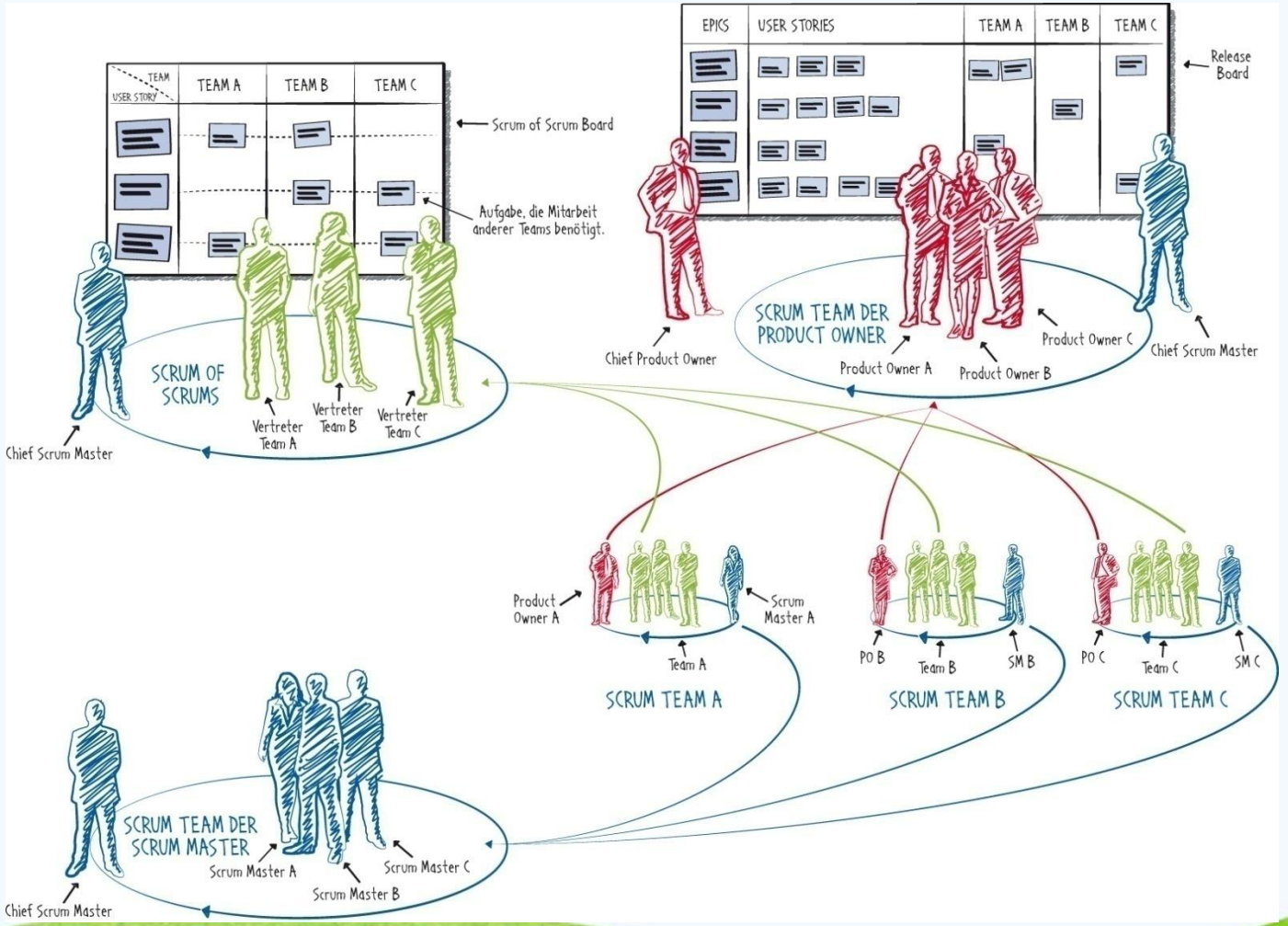
Daily / 2-3 times per week

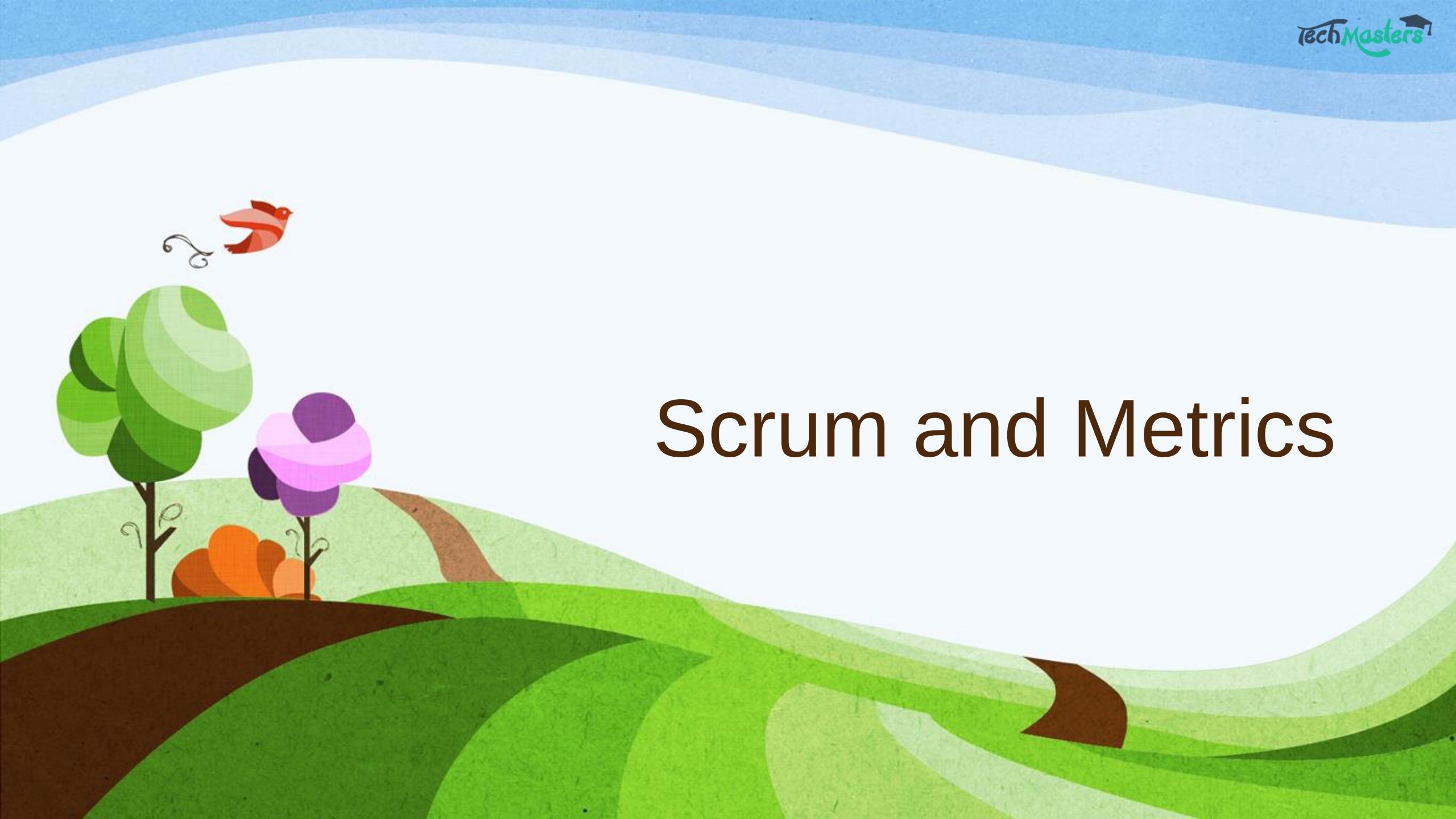
Coordination, Dependencies Mgt, Block Surfacing



Team A Team B Team C Team D Team E

Visual Illustration



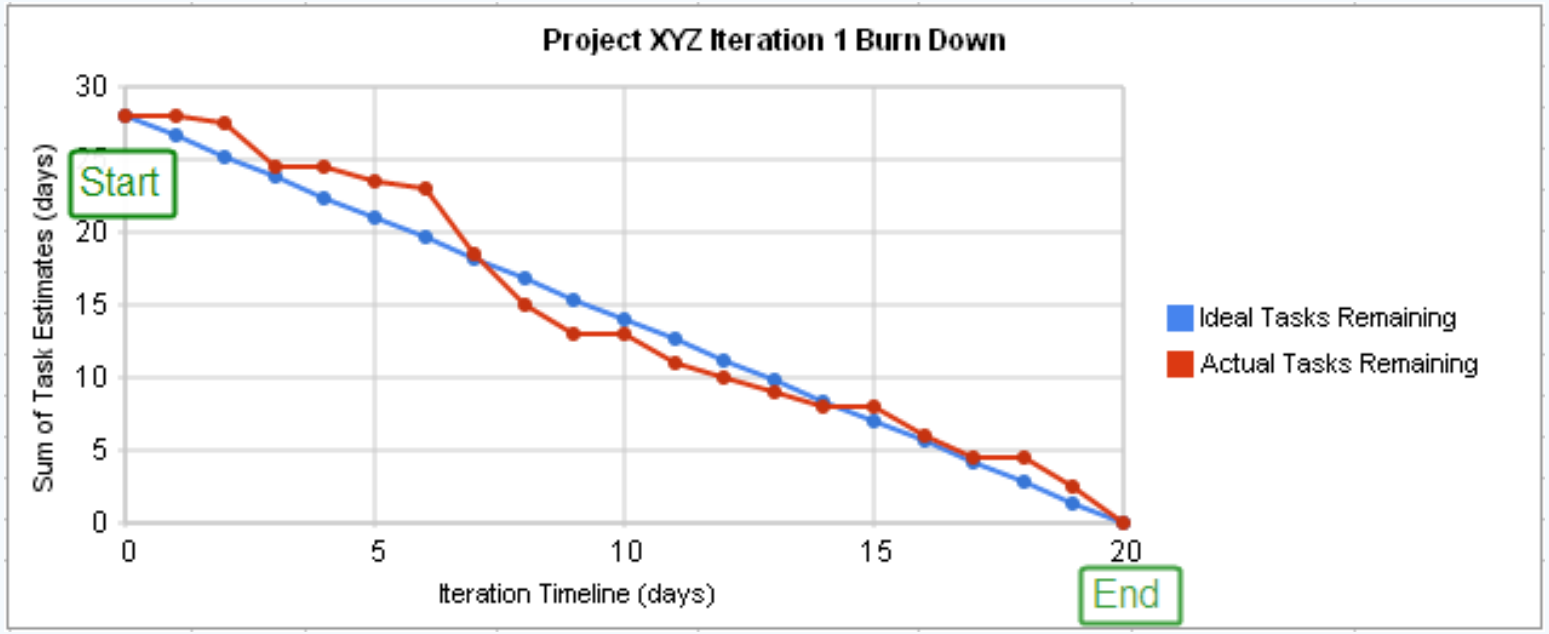


Scrum and Metrics

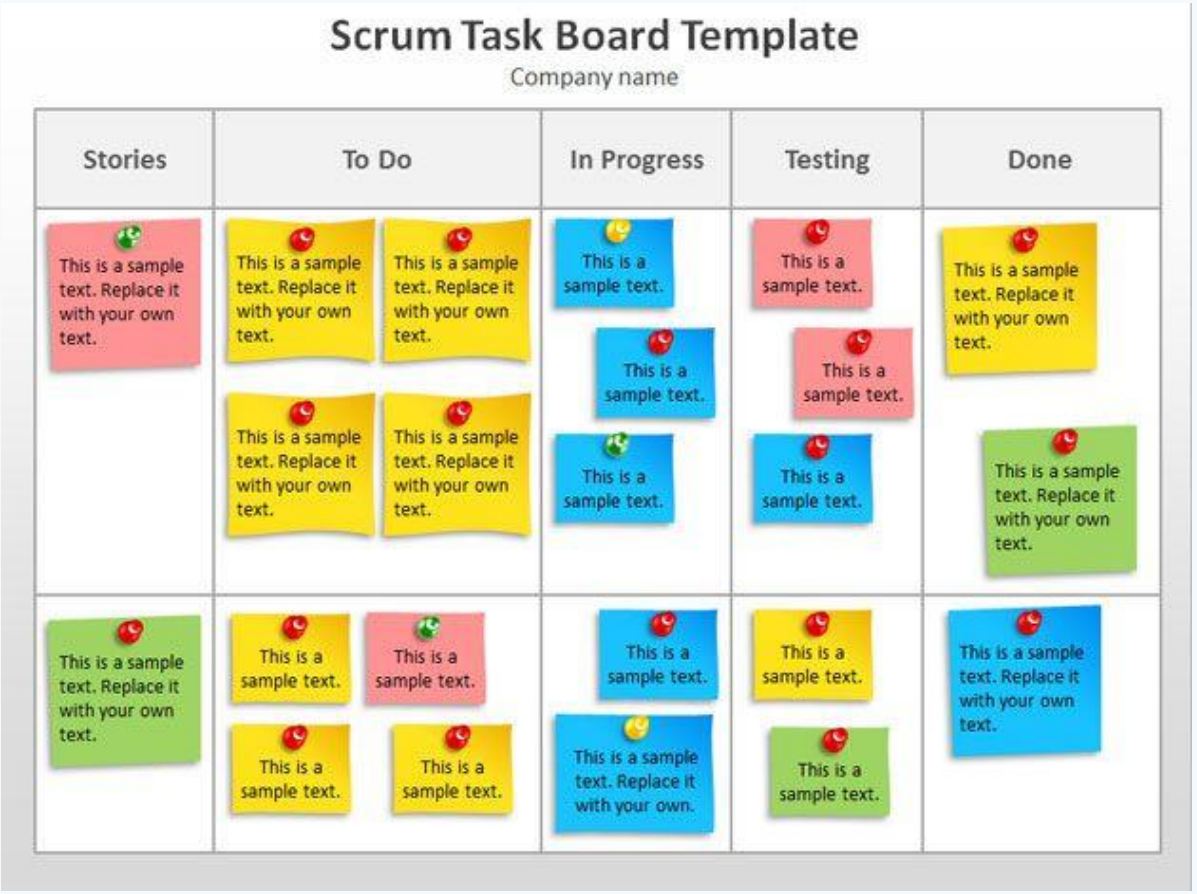
Scrum and Metrics

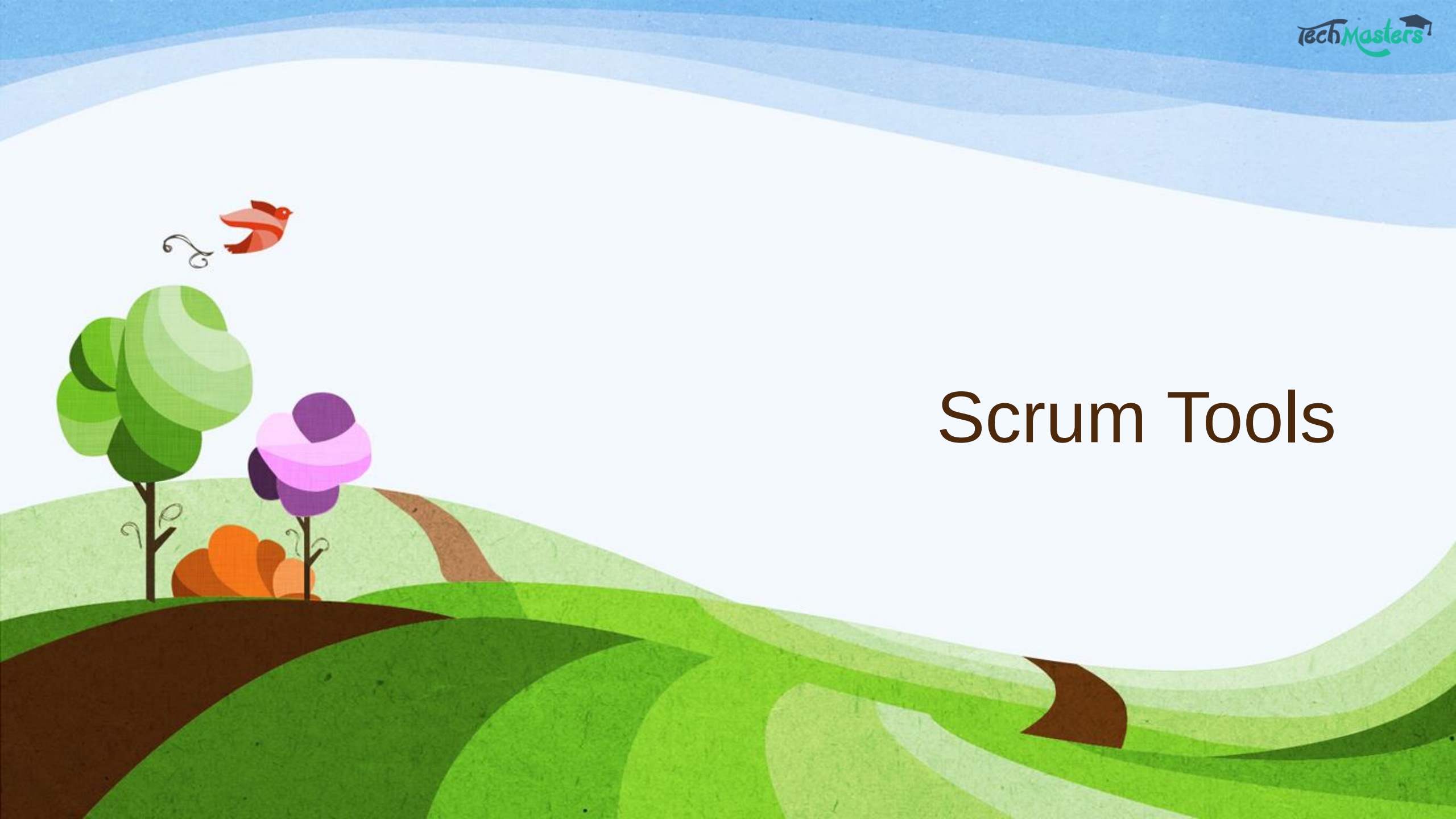
- Scrum doesn’t prevent you from tracking or measuring any aspect of the development process you choose
- However, you must be mindful of unintended consequences of measurement
  - Example: Individual burndown charts
- Recording or reporting measurements will probably require effort
  - If this effort comes from team-members, it should be made visible in their task time estimates or as Backlog Items

Burndown chart



Task Board


Scrum Task Board Template
Company name
Stories
To Do
In Progress
Testing
Done
This is a sample text. Replace it with your own text.
This is a sample text.
This is a sample text. Replace it with your own text.
This is a sample text. Replace it with your own text.



Scrum Tools

- Open-source or Free Tool – X-planner, www.icescrum.org, trello
- Enterprise Tools – Rally, VersionOne, ScrumWorks, Mingle, Jira
- Lightweight Tools – ScrumNinja, BananaScrum
- Keep it simple and do not rely fully on tools , at least in first few months of using scrum.

Using Scrum for Multi-Location Development / Remote teams

Distributed Scrum Practices

- Structure A: Product Owner is in US, team is in various part of the world
- One Common Approach
  - ScrumMaster located with team with most members (preferred)
  - 2-week Sprints
  - Team holds daily standup meetings in own location
    - ScrumMaster emails blocks list to Product Owner for assistance clearing in US

Distributed Scrum Practices

- Sprint Planning
  - Team has 1 hour real-time meeting with Product Owner to discuss goals of Sprint, broad review of Product Backlog Items (example: Weds night BDT)
  - Team spends 1-3 hours doing preliminary analysis and breakdown of Product Backlog Items (example: Thurs afternoon BDT)
  - Team spends 1-3 hours real-time with Product Owner completing analysis and breakdown of Product Backlog Items, and makes commitment (example: Thurs night BDT)
  - Sprint begins (example: Fri morning BDT)
- Sprint Review and Retrospective
  - Videoconference + Webex (example Weds morning BDT)

Distributed Scrum Practices

- Product Owner in Foreign Location travels to Bangladesh for project kickoff
- All real-time meetings with Product Owner are visual
  - Videoconference + Webex
  - Use a webcam if you have to
- Mailing list for Product Owner and all team members, with most project-related emails cc:ed to this list
  - Verbose subject lines to emails
- At least 1 weekly 1-hour real-time check-in between Product Owner and team
- In-person planning and review regroup in BD or US between Product Owner and team at least once per quarter

Distributed Team

- Stucture B: Team split between two locations
  - For example, 3 developers, 1 tester, and 1 analyst in US, 5 developers and 1 tester in Dhaka
- 2 ScrumMasters, one in each location
  - Lighter load than a full ScrumMaster – may be a team-member
- 2 separate daily standup meetings
  - Notes from each team's standup emailed to the other and read at the beginning of the meeting
  - (this is in addition to all the previous recommendations)

Adopting Scrum in Bangladesh

Step 1

- Management (CEO, CTO, Directors, Investors etc) must have clear idea about scrum
- Learn from Agile/Scrum professionals rather than own intuition

Step 2

- Start with a small team as pilot project
- Train the team about scrum paradigm

Step 3

- Give team time to understand scrum and try to motivate them to be self managed. As a nation, we are not used to with self management properly.
- Do not force them to start immediately

Step 4

- Run a test or small project for first time
- Get help from experts during the first few iterations

Step 5

- Measure teams output with expected results
- Try to find out where team is struggling and fix those

Step 6

- Keep patience during the whole transformation
- It can take few weeks to few months for a team to adopt

Step 7

- Once the team outputs the expected result, try to repeat it for other teams
- It can work as a base for others

Step 8

- Always plan, execute and adopt. Do not assume that team understood scrum properly and no need to invest time on study
- Empower team members, give them opportunity to understand themselves

Step 9

- Keep a budget for whole transformation
- You might not get an immediate result during the whole transformation process

About Me

M. Mizanur Rahman

Certified Scrum Master & Certified Scrum Professional

Founder & CTO, Informatix Technologies

Founder, TechMasters

Email: mizan@informatixbd.com

Blog: http://booleandreams.wordpress.com

Join our Agile Community @ FaceBook

- Agile Bangladesh:
- https://www.facebook.com/groups/agilebangladesh/