Knowledge United
Leverage the Power of Learning

cc pace
Committed Partner. Creating Results.

Scrum Team Training

Why CC Pace

CC Pace is one of the nation's most trusted Agile technology companies, having pioneered and perfected Agile software development, coaching, and training since the movement's earliest beginnings

Our trainers are practitioners with a penchant for training – delivering uncommon knowledge with the flexibility and interactivity that promotes real learning

Class Approach

We use a combination of:

- Lecture
- Simulation, Exercises
- Discussion, Dialogue

In an environment that:

- Is Relaxed
- Uses and Encourages Open Questions
- High-level overview first, supported by details as the day progresses

Knowledge United
Leverage the Power of Learning

Expectations

- **Exercises**
  - At times instructions may be vague and parameters may change.
- **Class Material**
  - When questions arise, there may be no black and white answer (after all this is Agile) . During these times, the instructor will share industry experience.
  - This training does not cover training for use with the Rally toolset.
  - SDLC Governance is not covered as part of this course.
  - We will use a 'parking lot' for Intuit specific processes.
- **Attendees**
  - Please be patient as this will help lay the foundation for the methods and principals you will continue to learn and employ.
  - You are not expected to be proficient in Scrum by the end of this class (that is what coaches are for!).
  - This class may be a mix of Agile experience. Why is this?
    - Encourage adoption of all Agile practices, and helps to get everyone on the same page and establish a baseline.
    - Help others in the class with less experience.

Knowledge United
Leverage the Power of Learning

Training Agenda

- **Opening Activities**
- **Why Agile**
  - The Industry Trend
  - The Problem Space
- **Agile Overview**
  - The Meaning of Agility
  - Values & Principles
  - A Different Way
  - Methods
- **Scrum Overview**
  - Source
  - Theory
  - Roles, Ceremonies and Artifacts
- **Scrum Building Blocks**
  - Story Time
  - The Backlog
  - Estimating Stories
  - Release Planning
  - Sprint Lifecycle
  - The Board
  - Daily Scrum
- **Team Charts**
- **Blended**
- **Distributed**
- **Debrief**
- **Closing Activities**

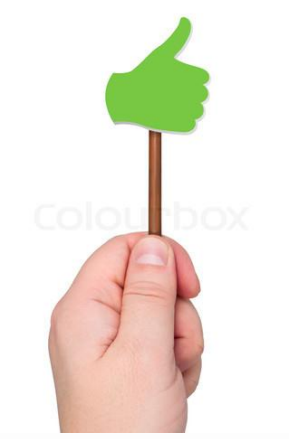
Please feel free to use the speedup or slow down symbols under the raised hand icon to let me know if I need to slow down the pace or pick the pace up. I will also check in periodically.

Knowledge United
Leverage the Power of Learning

General Instructions:

Please raise your hand if you wish to participate by going to the toolbar at the top of the screen and selecting the raised hand icon. Please lower your hand by going to the toolbar and selecting lower hand when done.

To agree or disagree please go to the raised hand icon on the toolbar and select agree or disagree



Let's Establish Team Norms

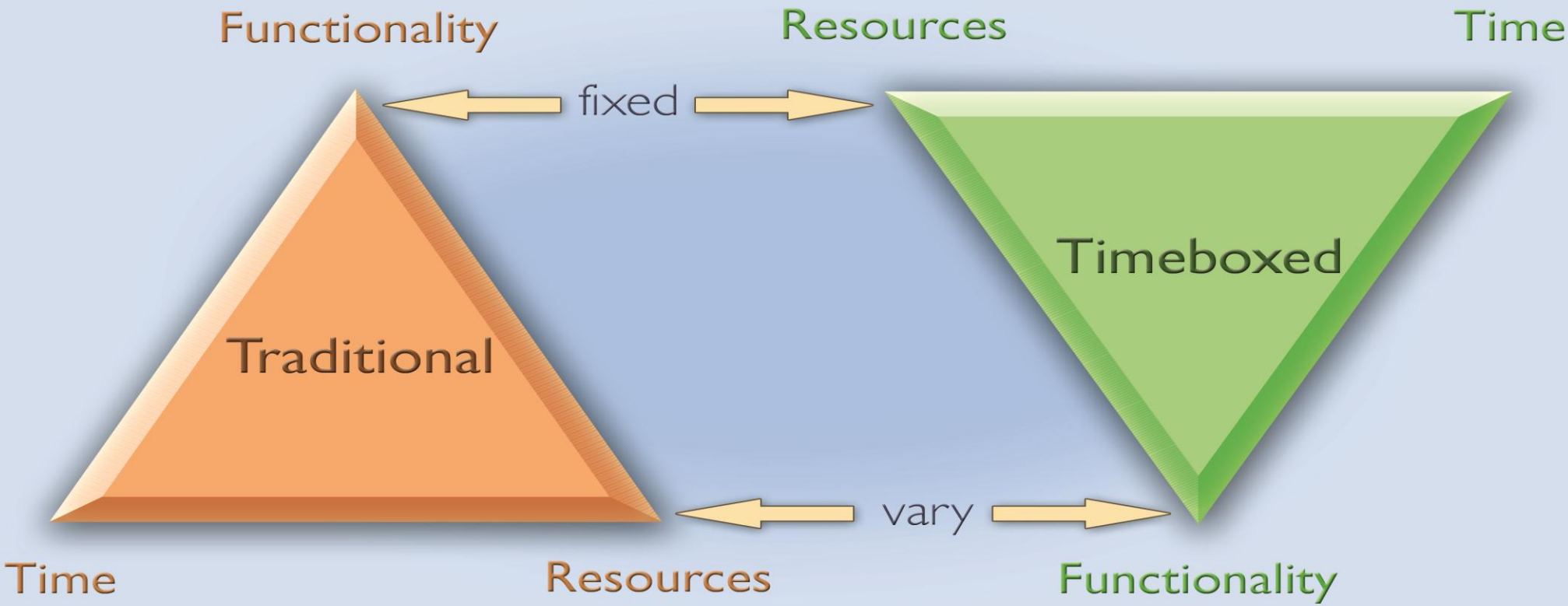
- Example – multiple breaks

Knowledge United
Leverage the Power of Learning

Why Agile?

The "Problem Space"

A New Way of Thinking



The Problem

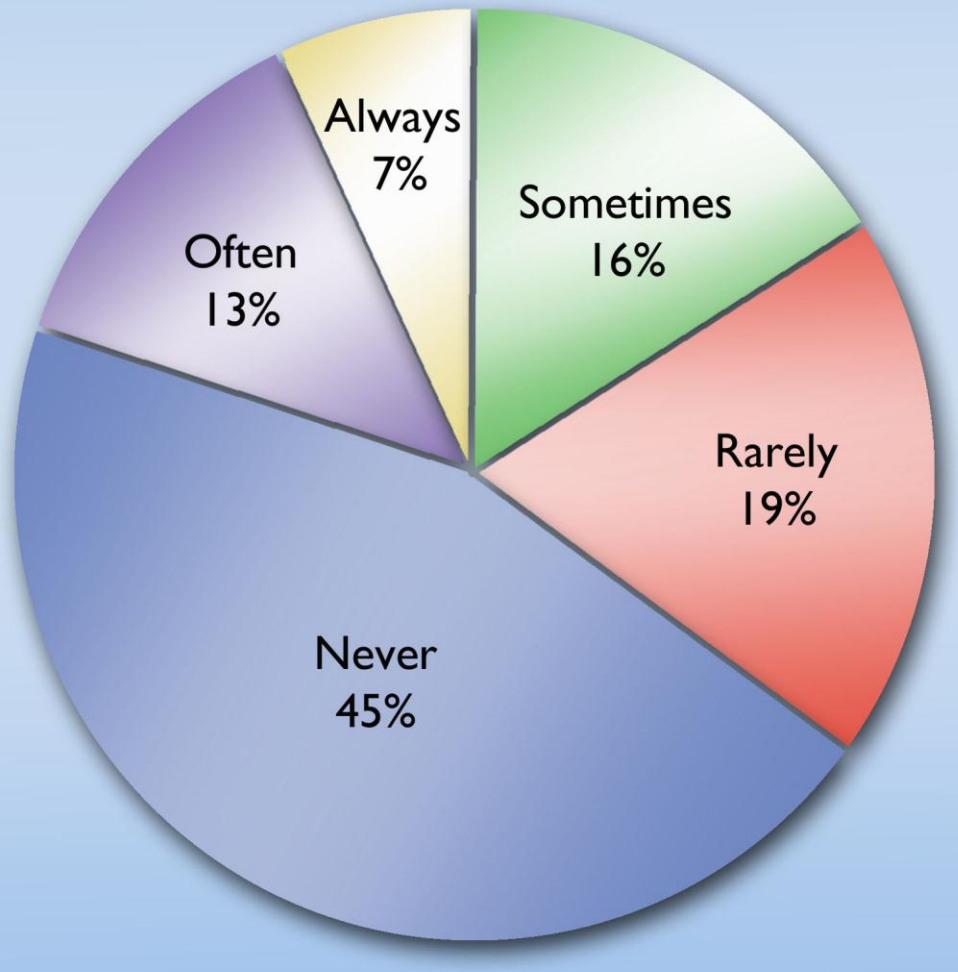
Traditional methods like Waterfall have been highly unsuccessful when it comes to producing complex software.

Source: Chaos Report from Standish Group (2008).

Source: Chaos Report from Standish Group (2012).

The Problem

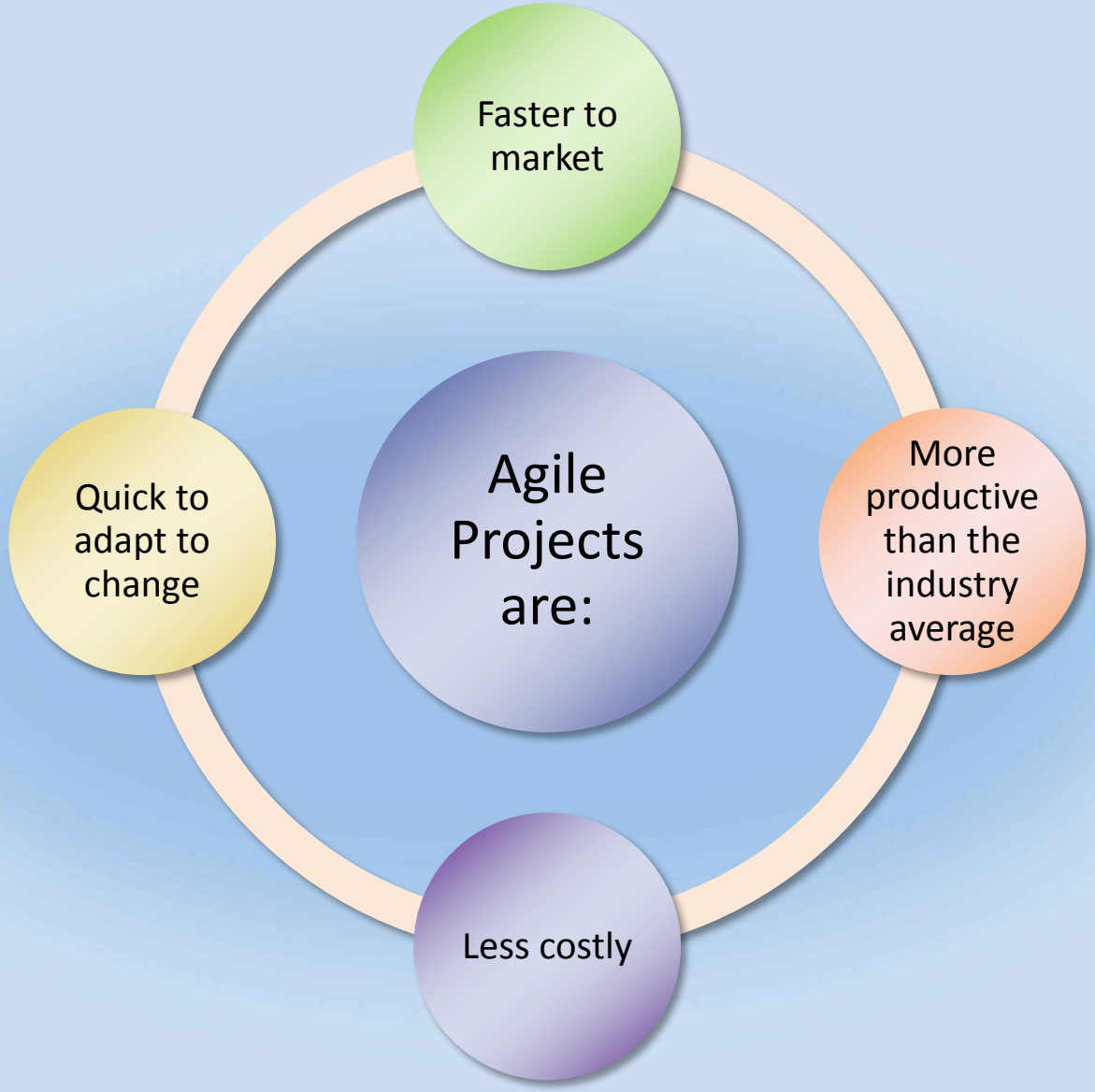
…Of those that do succeed, 80% of the features built are not used frequently.



Source: Standish Group Study Reported at XP2002 by Jim Johnson, Chairman

Why Agile?

Why is the Industry Moving to Agile?



…while significantly improving customer satisfaction and having no adverse effect on quality

What's Your Target?

Waterfall is designed to hit fixed targets.

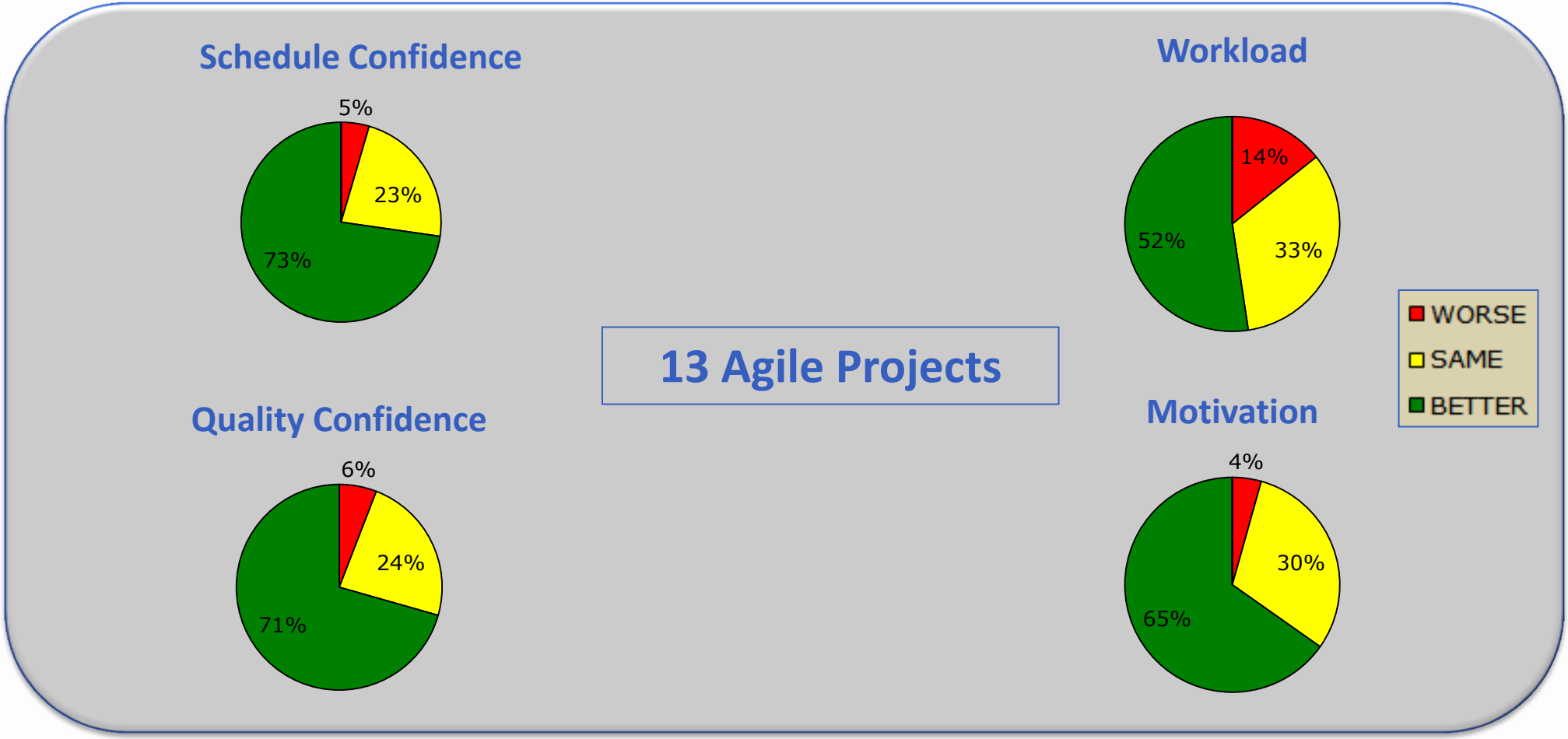
Agile is designed to hit moving targets

The process you design to hit a moving target is different from the process you design to hit a fixed target.

Knowledge United
Leverage the Power of Learning

Why did the Small Business Group (SBG) Move to Agile?

"Compared to your last project…"



Knowledge United
Leverage the Power of Learning

Intuit Confidential

Self-Organize

- I will move each of you into a room, where you will be able to talk with each other
- Each team will have **15 minutes** for this exercise
- **Introduce** yourselves and provide a little background
- Select a **team name** (need to be able to say it in polite company)
  - Examples – movie titles, Greek alphabet, Scrum of the Earth, Pond Scrum, Scrum delicious
- Develop 1- 2 **questions** about Agile/Scrum that your group would like to have answered before the end of the course.
- Select someone to **report back** to us from your team
- I will move each of you back to the main training room and we will **review**

Knowledge United
Leverage the Power of Learning

Agile Overview

What is Agility?

Ag•ile: *adj.*

1. Quick and well-coordinated in movement
2. Marked by an ability to think quickly; mentally accurate or aware

- Quick, well-coordinated actions
- Think quickly
- Make appropriate decisions
- Take suitable action
- Be quick, resourceful, adaptable
- React and adapt to changing situations

Knowledge United
Leverage the Power of Learning

"Agile" is an approach for undertaking projects which is designed to promote "agility"

This conceptual framework is based on a common set of values and principles expressed in the "Agile Manifesto"

There are 4 Agile Values and 12 Agile Principles

Knowledge United
Leverage the Power of Learning

What is “Agile”? – The 4 Values

“We are uncovering better ways of developing software by doing it and helping others do it. .. while there is value in the items on the right, we value the items on the left more”

Individuals and Interactions	Over	Processes and Tools
Working Software	Over	Comprehensive Documentation
Customer Collaboration	Over	Contract Negotiation
Responding to Change	Over	Following a Plan

Agile Manifesto 2001, www.agilealliance.org

The 12 Principles of Agile

1. Early, continuous delivery
2. Welcome changing requirements
3. Deliver working software frequently
4. Business people and developers working together daily
5. Build projects around motivated individuals
6. Face-to-face conversation is the best form of communication

The 12 Principles of Agile

7. Working software is the principle measure of success

8. Agile processes promote a sustainable pace

9. Continuous attention to technical excellence & good design

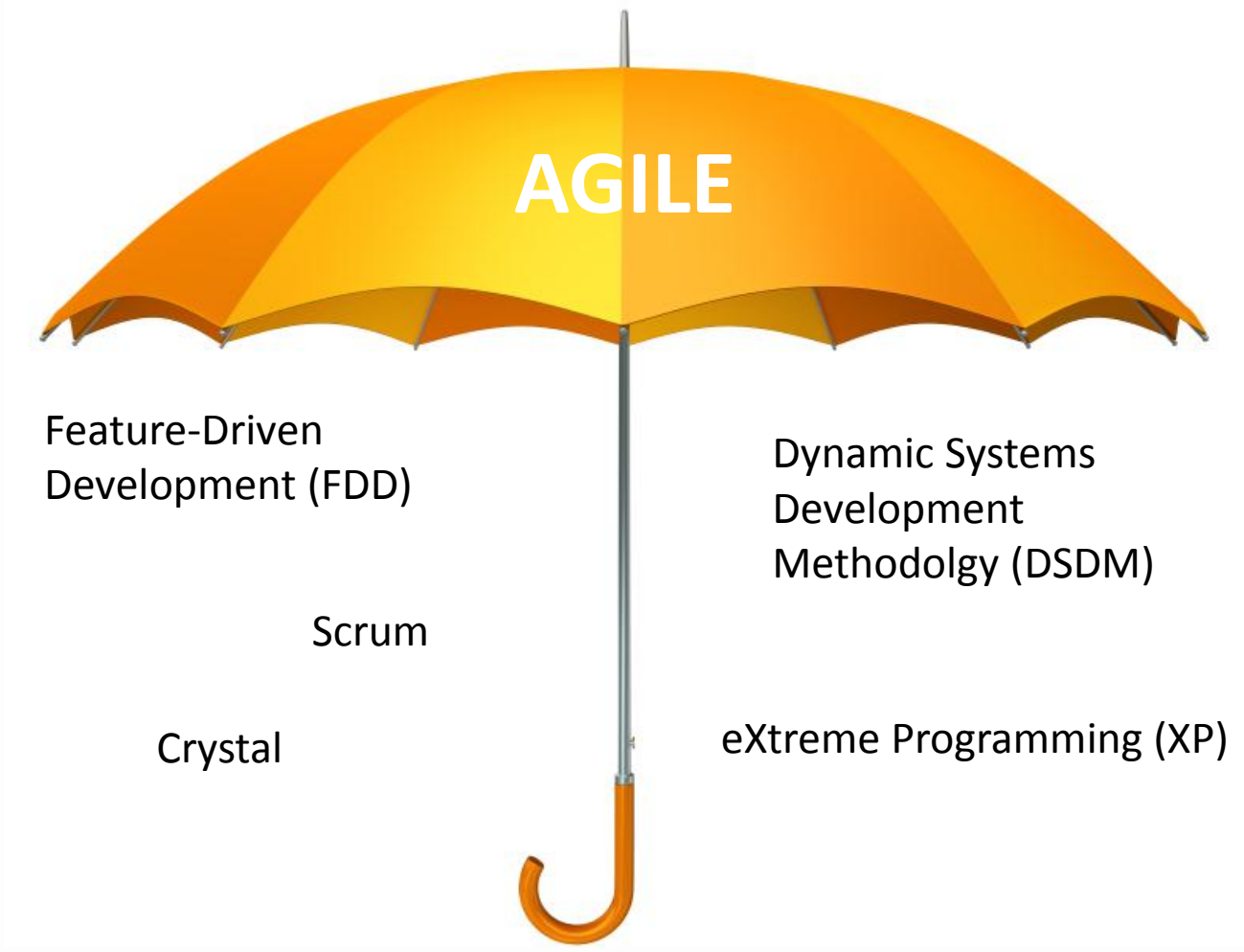
10. Simplicity

11. Self-organizing teams

12. The team reflects regularly on how to become more effective, then tunes, adjusts and adopts its behavior accordingly

The Agile Umbrella

"Agile" describes a series of related methodologies.



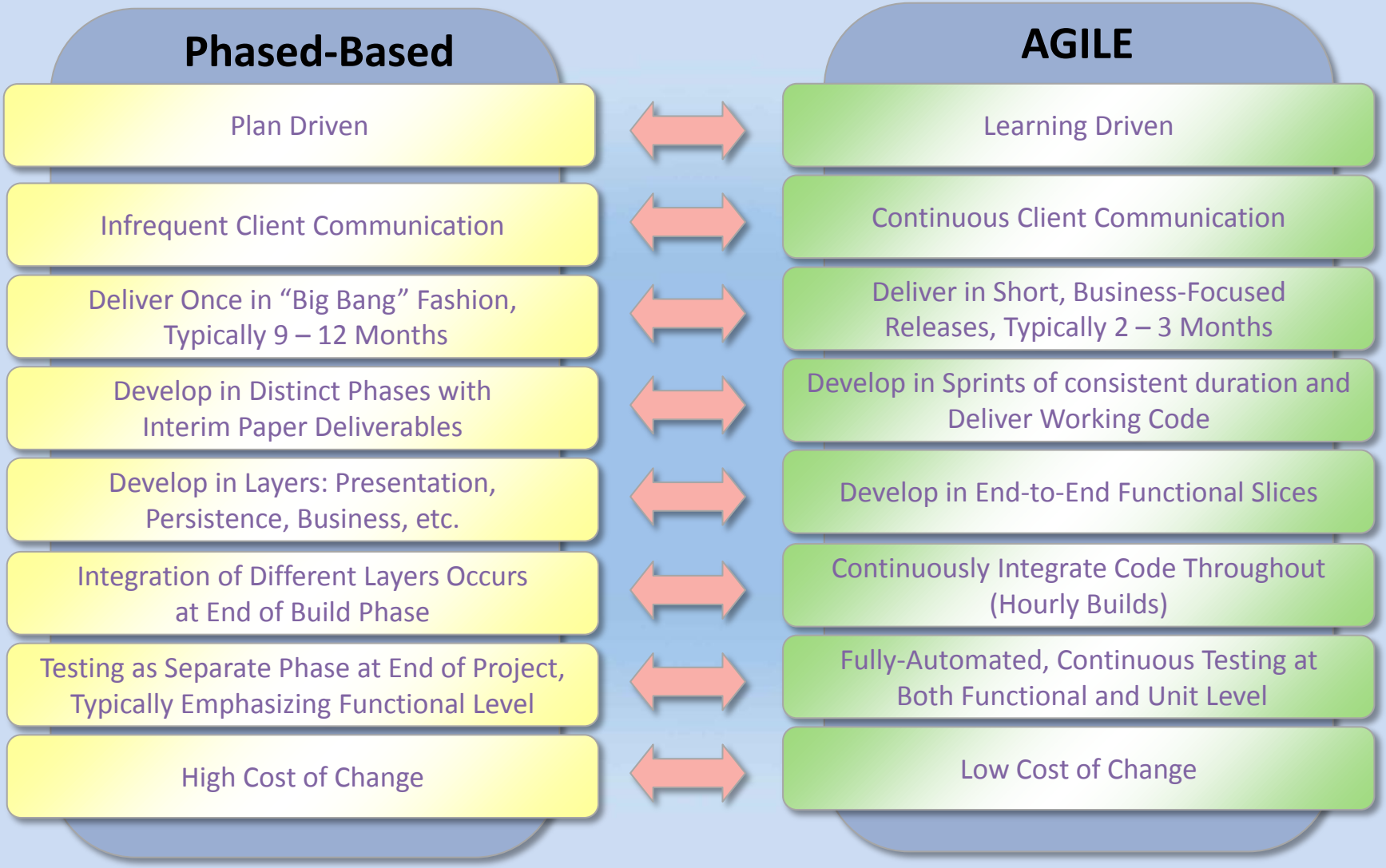
Knowledge United
Leverage the Power of Learning

A Way of Thinking

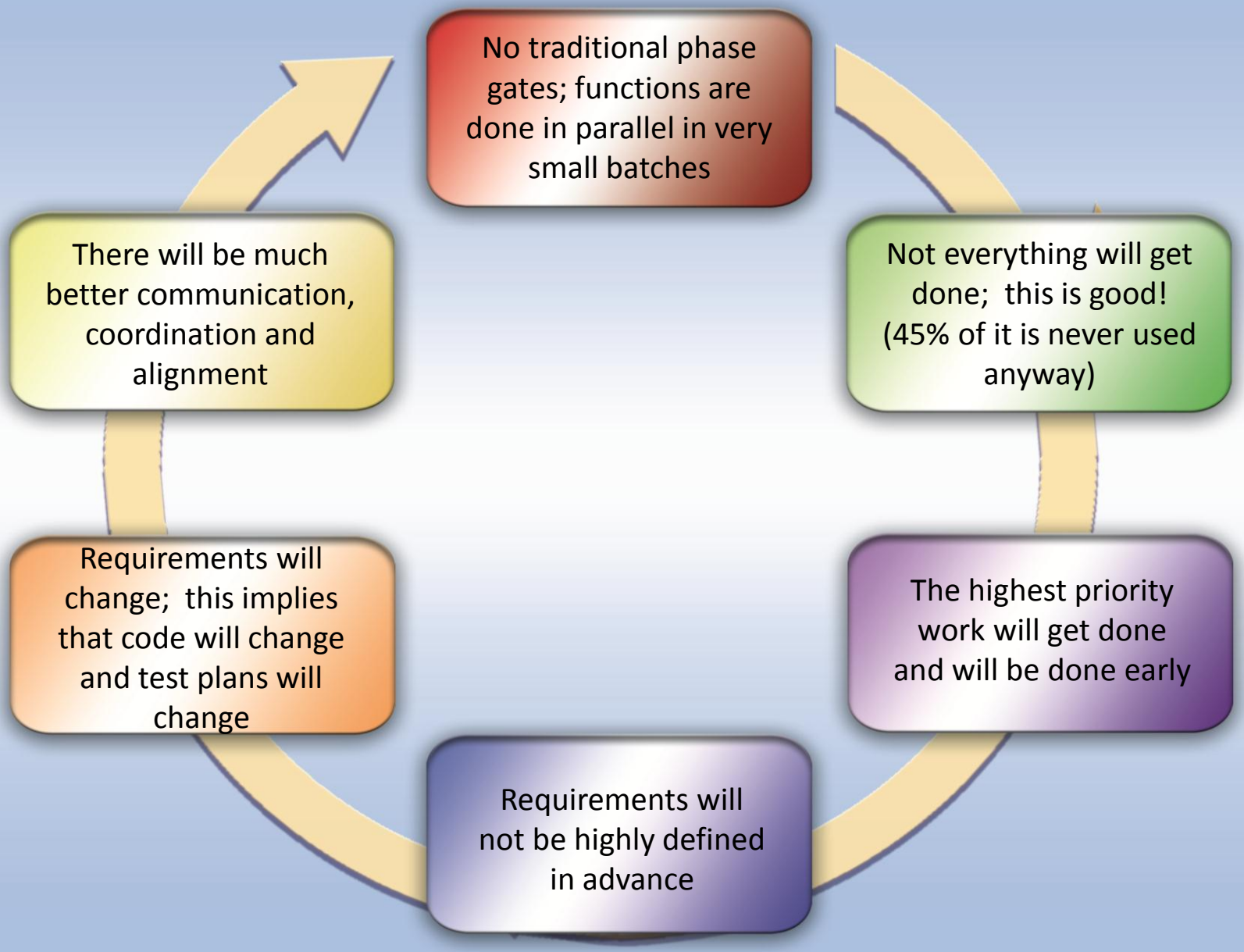
Embrace Change… in Attitude… in the Way You Work

- Uncertainty is OK
- Emergence is GOOD
- Simplify & be Mindful of Waste
- Excellence
- Think Small Slices
- Minimize Work In Progress (WIP)
- Those who do the work do the Steering & Planning
- Don’t overburden resources
- Trust & Support each other
- I am RESPONSIBLE
- Ask yourself “What is the best thing I can do today to move the project forward ?”

Agile - A Different Way of Doing



Implications of Agile



Why Isn't Moving to Agile Easy?

"There are 'just' two problems left to solve in business – people and change."

-- Bob Eichenger, Co-Founder, Lominger International

1. It requires thinking and acting in new ways
2. Roles/responsibilities will change
3. Interactions will change
4. Processes will change
5. People resist change

Knowledge United
Leverage the Power of Learning

A Journey

TDD

Automated Testing

Continuous Integration

Adaptive Planning

Iterative & Incremental Delivery

Co-located Teams

Daily Meetings

Regular Reflection, Inspection, & Adaption

Responsive To Change

Un-responsive To Change

Everyone's journey will be different, because starting points are different, environments are different, and obstacles are different.

Knowledge United
Leverage the Power of Learning

Agile Forms

Forms of Agile

1. Agile by the Book
2. Blended Agile
3. _____Agile

Knowledge United
Leverage the Power of Learning

What is Blended Agile?

- **Blended Agile** is a term we use to describe a combination of Agile practices/techniques integrated with more traditional methods
- This is generally used in a scenario where an organization cannot flip a switch overnight and move to a full set of Agile practices
- A company will slowly transition to Agile, gradually adding Agile practices over time while still holding on to some traditional practices
- Blended Agile is not Agile in it's purest form. Therefore, you will not obtain all of the benefits of Agile

Knowledge United
Leverage the Power of Learning

Why Use a Blended Approach? Benefits?

- Companies must collaborate with internal areas and outside organizations utilizing traditional development processes
- Companies have standing processes that can not be changed with a flip of a switch
- Company business must continue
- It takes time to move to a full Agile capability
- Companies have real obstacles that do not allow them to move as quickly
- Reduces the perception of risk
- Gives the organization time to learn and to adjust which practices are right for their organization

Knowledge United
Leverage the Power of Learning

Scrum Overview

What is Scrum?

The term “scrum” originally comes from a strategy in rugby where it involves getting an out-of-play ball back into the game with teamwork.

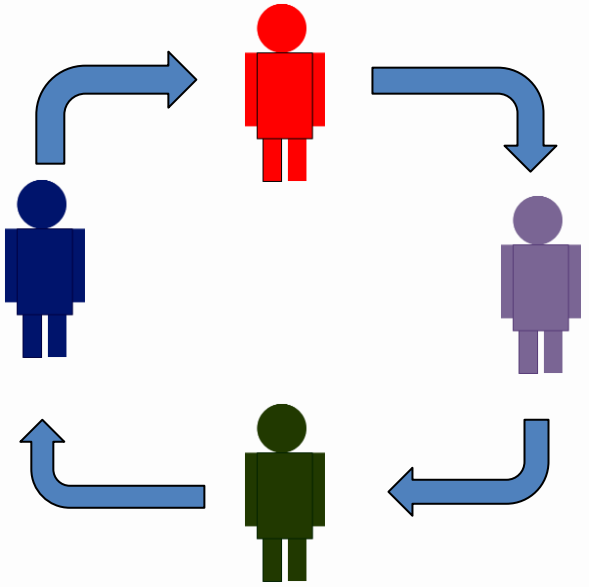
Knowledge United
Leverage the Power of Learning

What Does the Team Look Like?

From Silos, Lack of Communication, Multiple Handoffs

TO

Cross Functional Roles, High Collaboration, Results Oriented



SHARED VISION

- Everyone headed in the same direction

INTERDEPENDENT RHYTHM

- Each member dependent on the one next to him
- Discrete members work together in an efficient formation resulting in fluid motion

COMMUNICATION

- Shared language of encouragement helps maintain speed and direction

COLLABORATION & LEADERSHIP

- Rotation and distribution of position and responsibility to leverage skill / experience and avoid fatigue

Adaptive**.....***Efficient*****.....*****Enduring*****.....***Fast*****.....*****Supportive*****.....***Shared-Leadership***

Knowledge United
Leverage the Power of Learning

Scrum Building Blocks

- *Roles, Ceremonies and Artifacts*

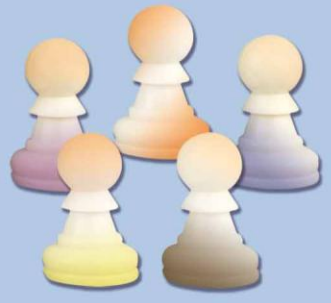
Key Scrum Roles and Responsibilities

PRODUCT OWNER

- Balances workload to maximize their time with the scrum team
- The 'single voice' on product features and release content
- Maintains the product vision, release goals, and maximizes ROI
- Decomposes features into a backlog of stories
- Continually grooms and prioritizes backlog of stories
- Accepts or rejects work results

SCRUM MASTER

- Ensures that the team is fully functional and productive
- Enables close cooperation across all roles and functions and removes barriers
- Shields the team from external interferences
- Ensures that the process is followed.



DEVELOPEMENT TEAM

- Cross-functional, includes all skills needed to ensure "Done"
- Seven plus/minus two members
- Tasks Work, Estimates Work, Volunteers to perform work
- Has the right to do everything within the boundaries of the project guidelines to reach the iteration goal
- Organizes itself and its work
- Demos work results to the end-user and stakeholders

All three roles are mutually accountable for success of project

Scrum Master vs. Product Owner

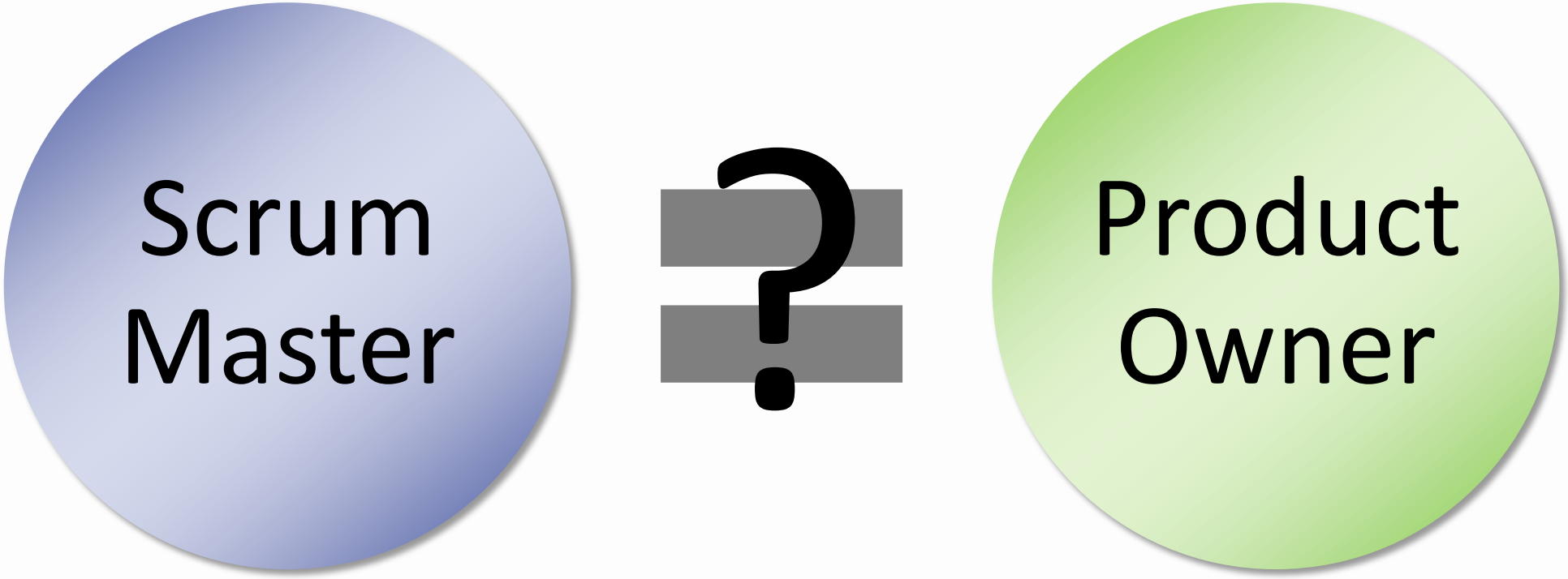
Scrum Master

- Voice of Team
- Communicates with the team
- Emphasis on the Agile Process
- Helps to communicate the level of difficulty

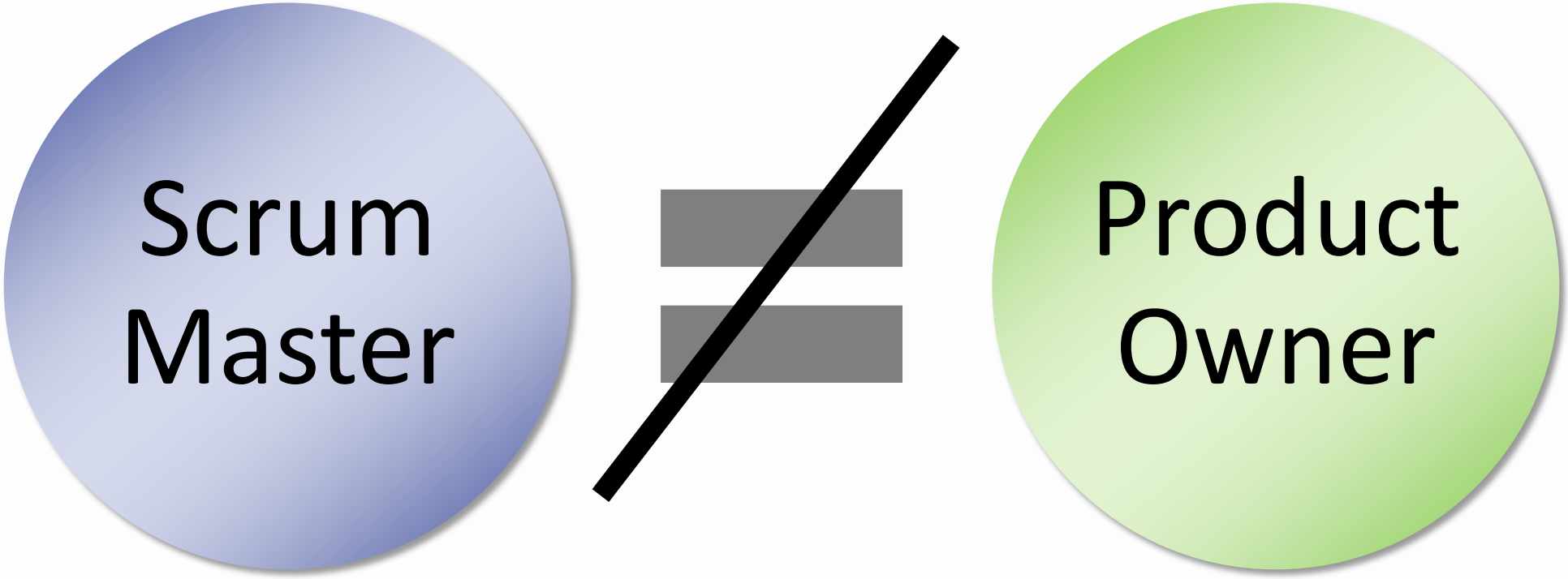
Product Owner

- Voice of Customer
- Communicates with the customer
- Emphasis on the "Features"
- Communicates the Priority

Can the Scrum Master and Product Owner be the same?



The roles conflict



Knowledge United
Leverage the Power of Learning

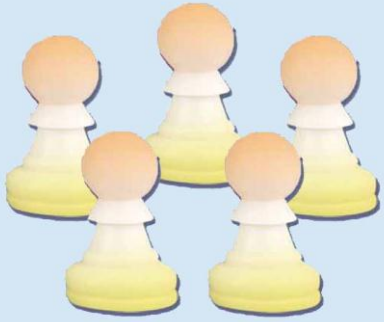
The Quandary?

The scrum master is not a project leader, thus is not held accountable for outcomes.

The team as a whole is responsible for outcomes.

Knowledge United
Leverage the Power of Learning

Extended and Release Team Roles and Responsibilities



EXTENDED TEAM AND RELEASE TEAM

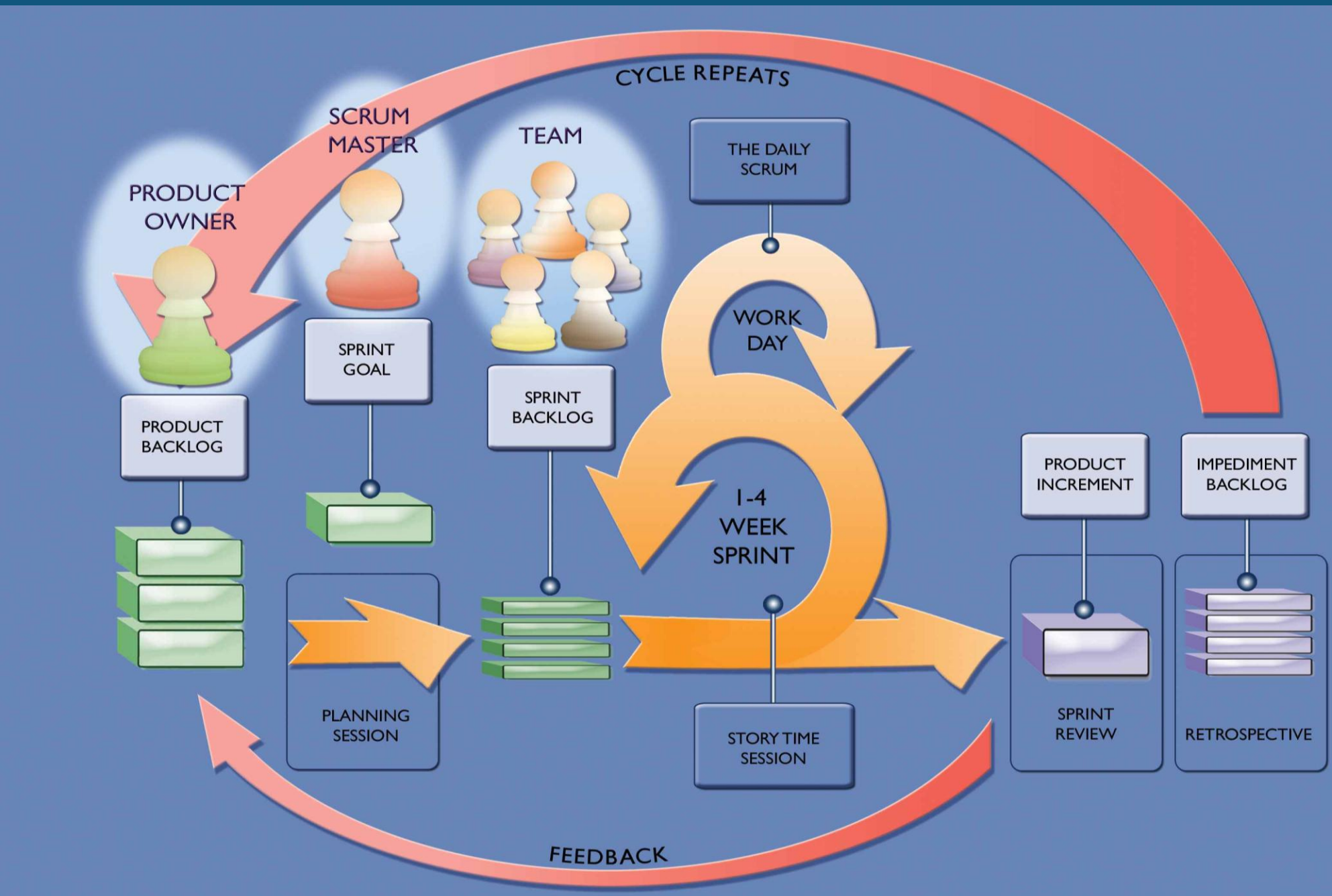
- Provides input into the backlog
- Attends the daily scrums when the have tasks
- Attends and provides input for the design and planning sessions
- Attends and provides input for Sprint Reviews
- Attends and provides input for the Retrospectives

COACHES

- Works with and supports the team in achieving their project goal
- Co plans and facilitates key meetings
- Consults with team members and analyzes data to recommend and implement improvements
- Collects best practices to use
- Helps teams roll out specific standards

These Roles – Extended/Release and the Coach support the team in achieving success of the project

Roles


CYCLE REPEATS
SCRUM MASTER
TEAM
PRODUCT OWNER
THE DAILY SCRUM
WORK DAY
SPRINT GOAL
SPRINT BACKLOG
PRODUCT BACKLOG
1-4 WEEK SPRINT
PRODUCT INCREMENT
IMPEDIMENT BACKLOG
PLANNING SESSION
STORY TIME SESSION
SPRINT REVIEW
RETROSPECTIVE
FEEDBACK

External Roles

- Users, Customers and Management

- An external role has minimal direct involvement with the team
- External roles Interact with:
  - Product Owner about Product Scope
  - Scrum Master about Process

Knowledge United
Leverage the Power of Learning

Discussion

On your projects today –

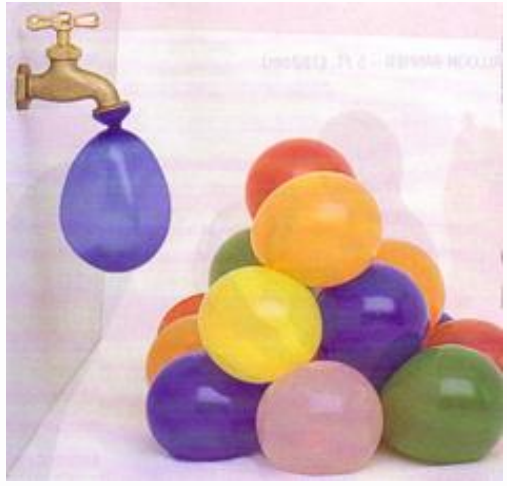
- What types of roles do you have?
- Is the customer working with you daily?
- Does your team always have everyone they need to complete your promise?

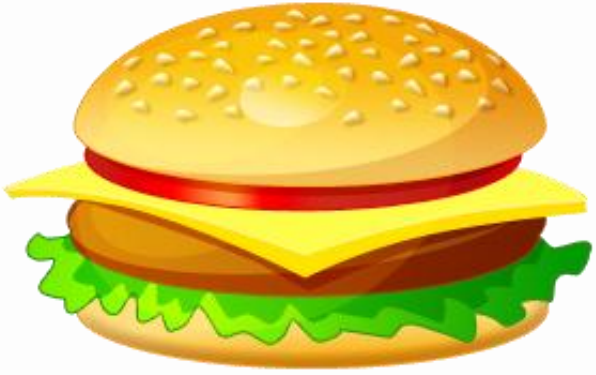
Take 5 minutes and answer the questions, then we will discuss

Please raise your hand when you are done

Knowledge United
Leverage the Power of Learning

"Create the Story" Exercise





Knowledge United
Leverage the Power of Learning

Ceremonies - Collaborative Planning

Product Plan

- Vision
- Business goals
- Roadmap
- Product Backlog
- Release Themes
- Business Value

Knowledge United
Leverage the Power of Learning

Ceremonies/Events - Collaborative Planning

Product Plan

- Vision
- Business goals
- Roadmap
- Product Backlog
- Release Themes
- Business Value

Release Plan

- Release goals
- Agreed Approach
- Initial Story Narration
- Relative Sizing
  in Story Points
- Definition of Done
- Team Norms
- Team Velocity
- Resource Plan

Knowledge United
Leverage the Power of Learning

Ceremonies/Events - Collaborative Planning

Product Plan

- Vision
- Business goals
- Roadmap
- Product Backlog
- Release Themes
- Business Value

Release Plan

- Release goals
- Agreed Approach
- Initial Story Narration
- Relative Sizing in Story Points
- Definition of Done
- Team Norms
- Team Velocity
- Resource Plan

Sprint Plan

- Team goals
- Sprint Backlog
- Story Narration for tasking
- Tasks
- Task Estimates
- Volunteer
- Sprint Resource Plan
- Team Norms

Knowledge United
Leverage the Power of Learning

Ceremonies/Events - Collaborative Planning

Product Plan

- Vision
- Business goals
- Roadmap
- Product Backlog
- Release Themes
- Business Value

Release Plan

- Release goals
- Initial Story Narration
- Relative Sizing in Story Points
- Definition of Done
- Team Norms
- Team Velocity
- Resource Plan

Sprint Plan

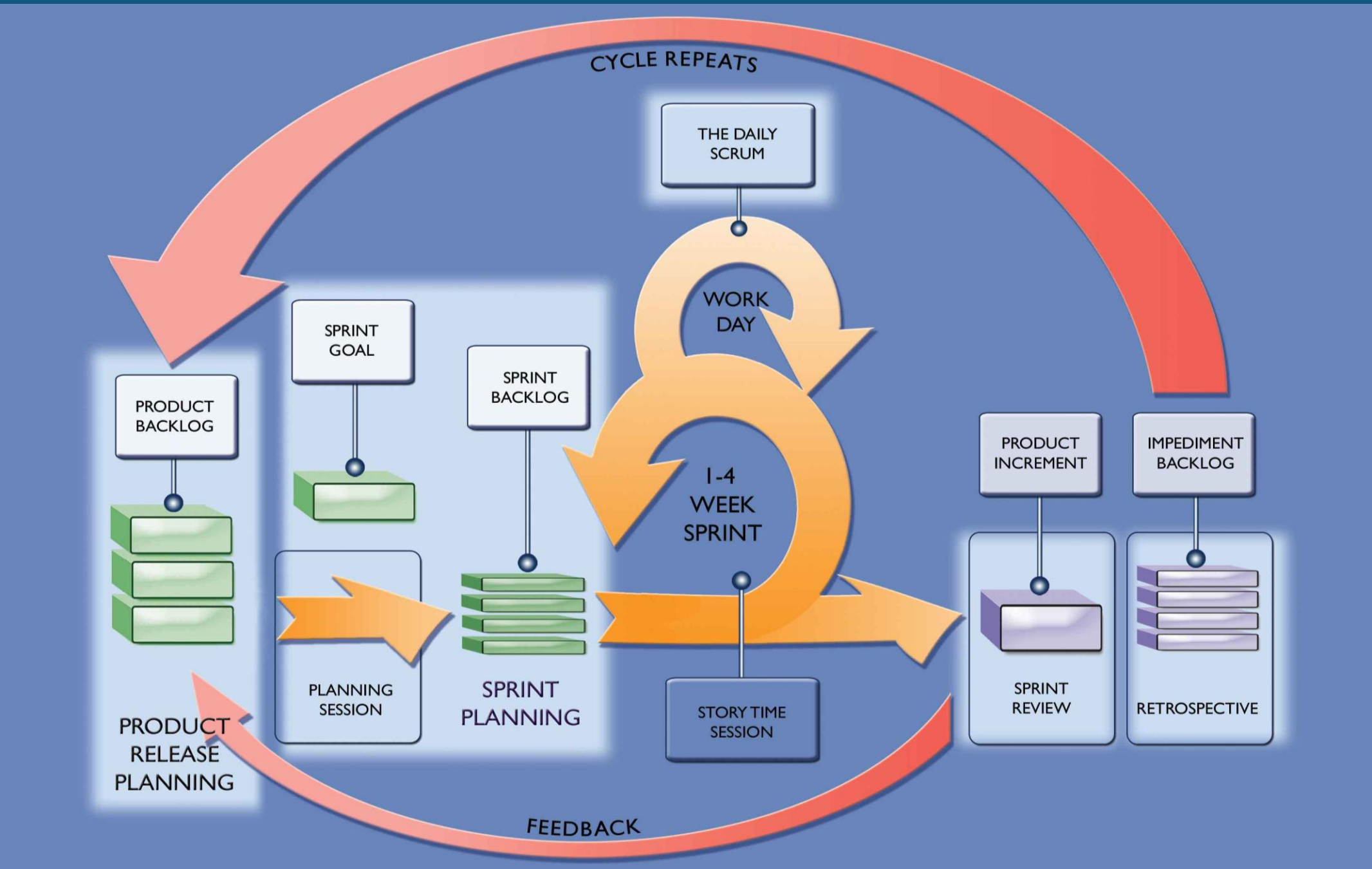
- Team goals
- Sprint Backlog
- Story Narration for tasking
- Tasks & Estimates
- Team Norms
- Volunteer
- Sprint Resource Plan

Daily Plan

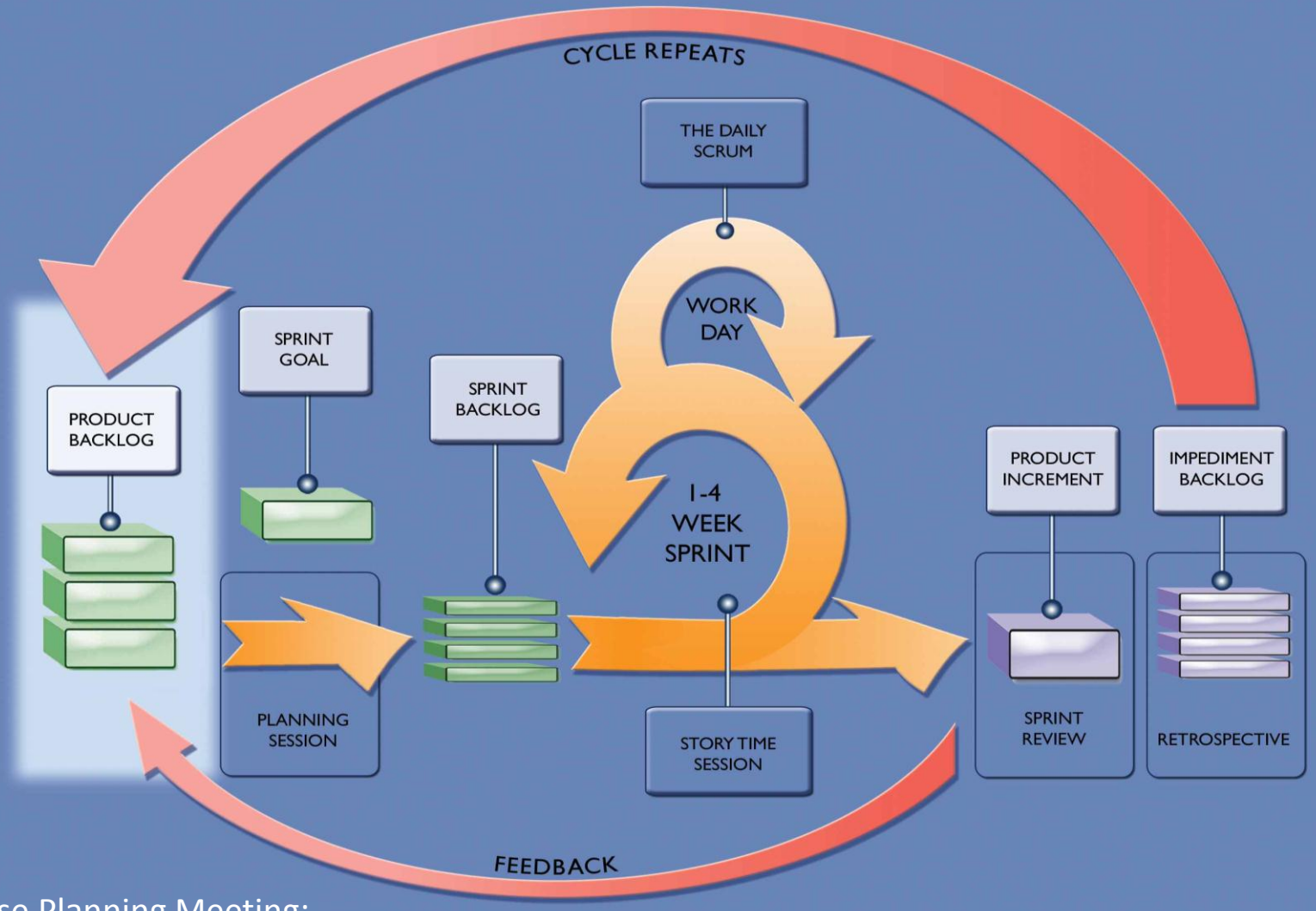
- Daily Scrum done standing, 15 min max
- What's Done
- What's Impeding Progress
- What's Next

Knowledge United
Leverage the Power of Learning

Scrum Ceremonies/Events



Ceremonies: Product Release Planning



Product Release Planning Meeting:

- A time-boxed meeting that initiates a Project or Release
- The Product Owner, Scrum Master, and Team meet to determine what will be worked on during the Project/Release
- The PO proposes their prioritized stories, and the Team reflects on how to solve each story
- Two actual meetings occur
  - Product Scope Confirmation Meeting – Team hears the scope and advises the Product Owner on concerns and technical approach
  - Backlog Sizing – the Team relatively sizes all stories and generally uncovers missing stories from the technical perspective

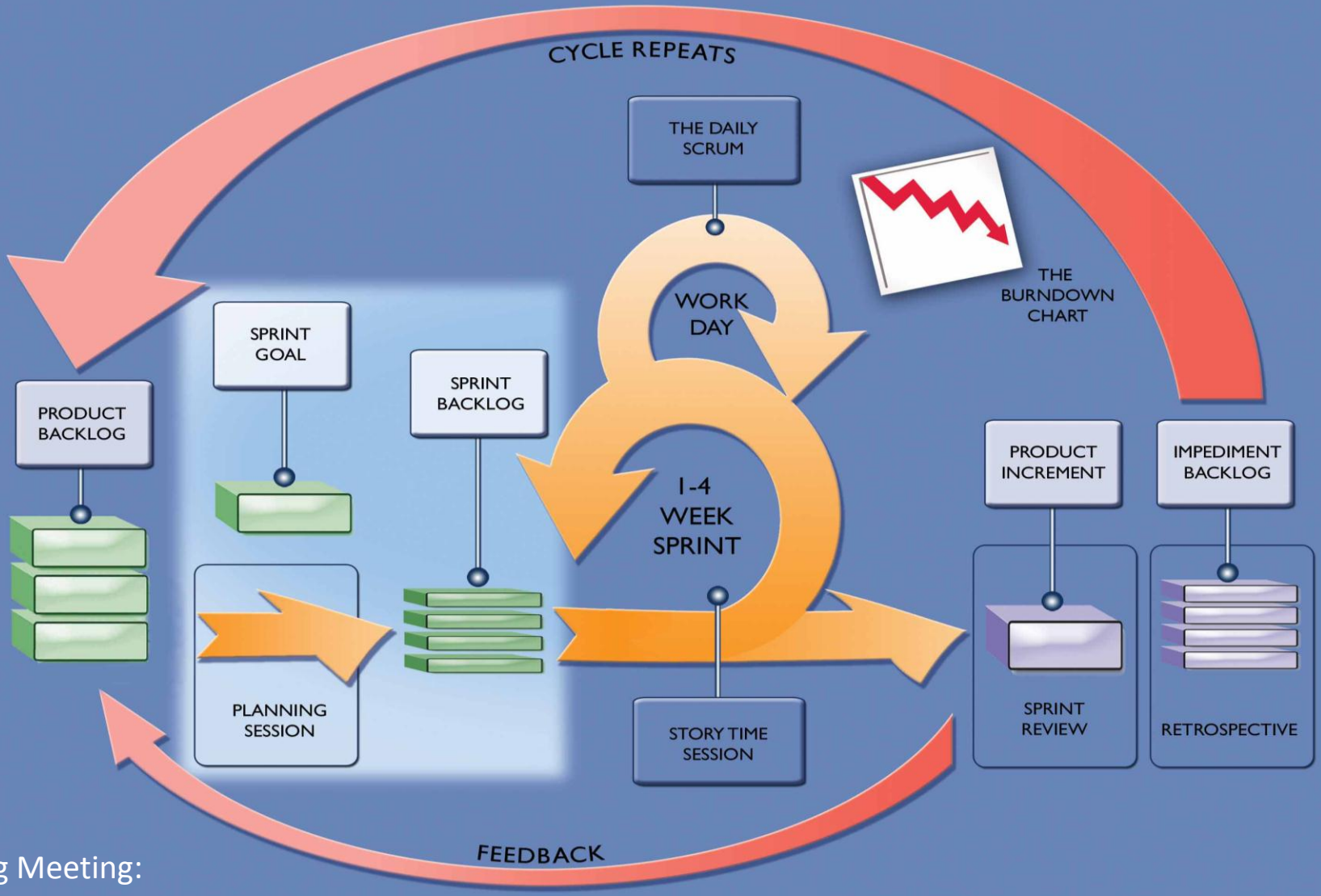
Release Planning

Also Includes:

- An initial team velocity
  - Based on previous team performance when available
- A target date for the release
  - A function of total points and velocity over time
- A Team Norm that agrees on the definition of DONE, for example:
  - Product Owner has seen a demo of the code and agrees the story is done
  - Designer confirms approach meets shop standard and guidelines
  - Team confirms all reasonable testing is complete
  - Documentation meets compliance
  - Artifacts have undergone all reviews including peer review

Knowledge United
Leverage the Power of Learning

Ceremonies: Sprint Planning



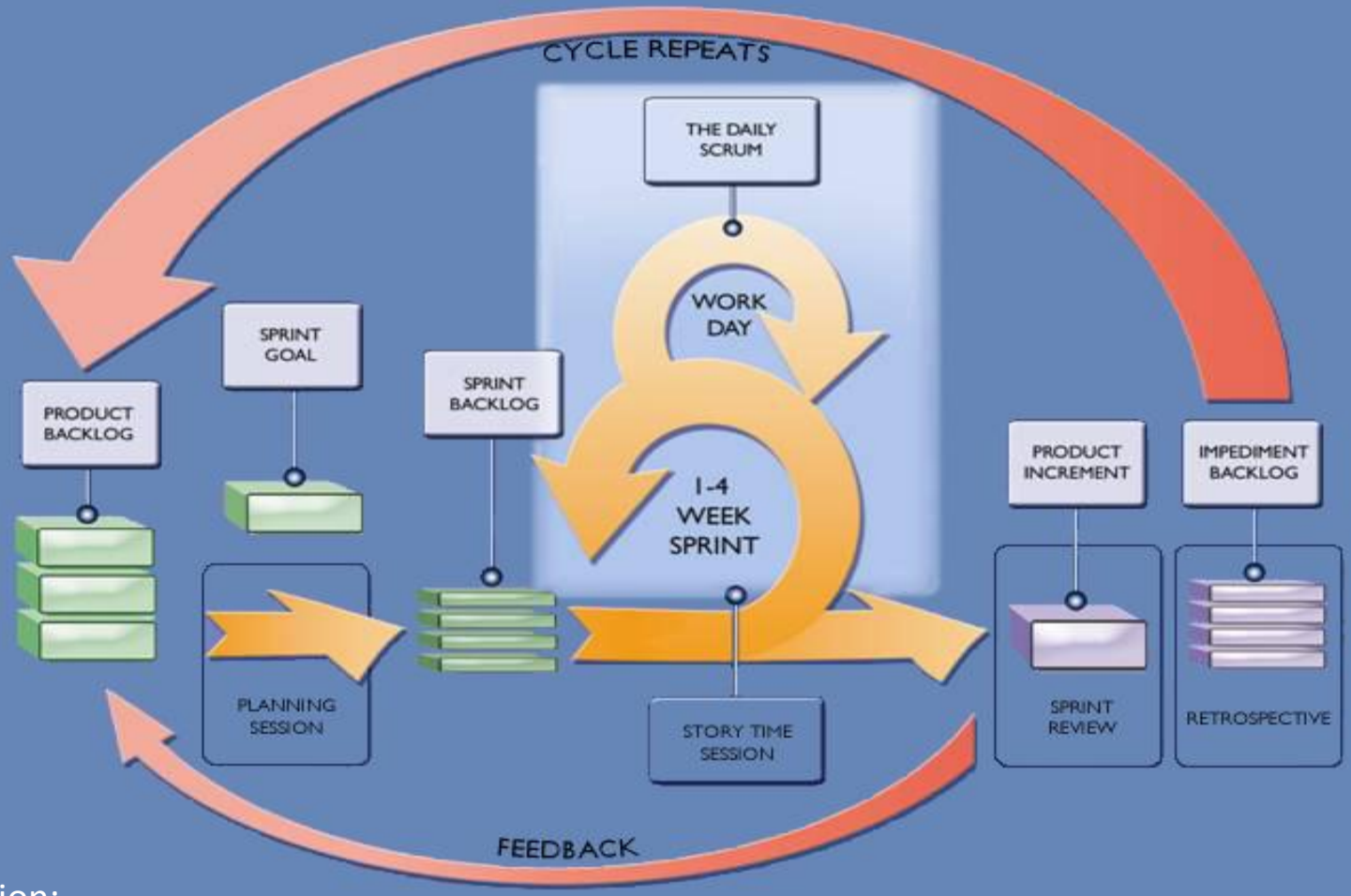
Sprint Planning Meeting:

- Select the highest priority items from the Product Backlog
- The Product Owner and Team agree on a Sprint Goal
- The team, which includes the Scrum Master and Product Owner, collaborate and discuss the proposed Sprint Backlog
- The Team breaks down Backlog items into Tasks
- Additional collaboration and negotiation occurs if necessary
- The Team members sign up for tasks/cards
- Final Agreement is reached, then the Sprint begins

Also Includes:

- Developing an action plan / task list necessary to complete the user stories selected
- Verifying Team Velocity
- Inclusion of lessons learned
- Coming to agreement on what DONE means for the Sprint
  - Identifying an acceptance tests for the sprint (i.e., a Sprint Goal)
  - Confirming “done” (acceptance tests) for each story in the sprint

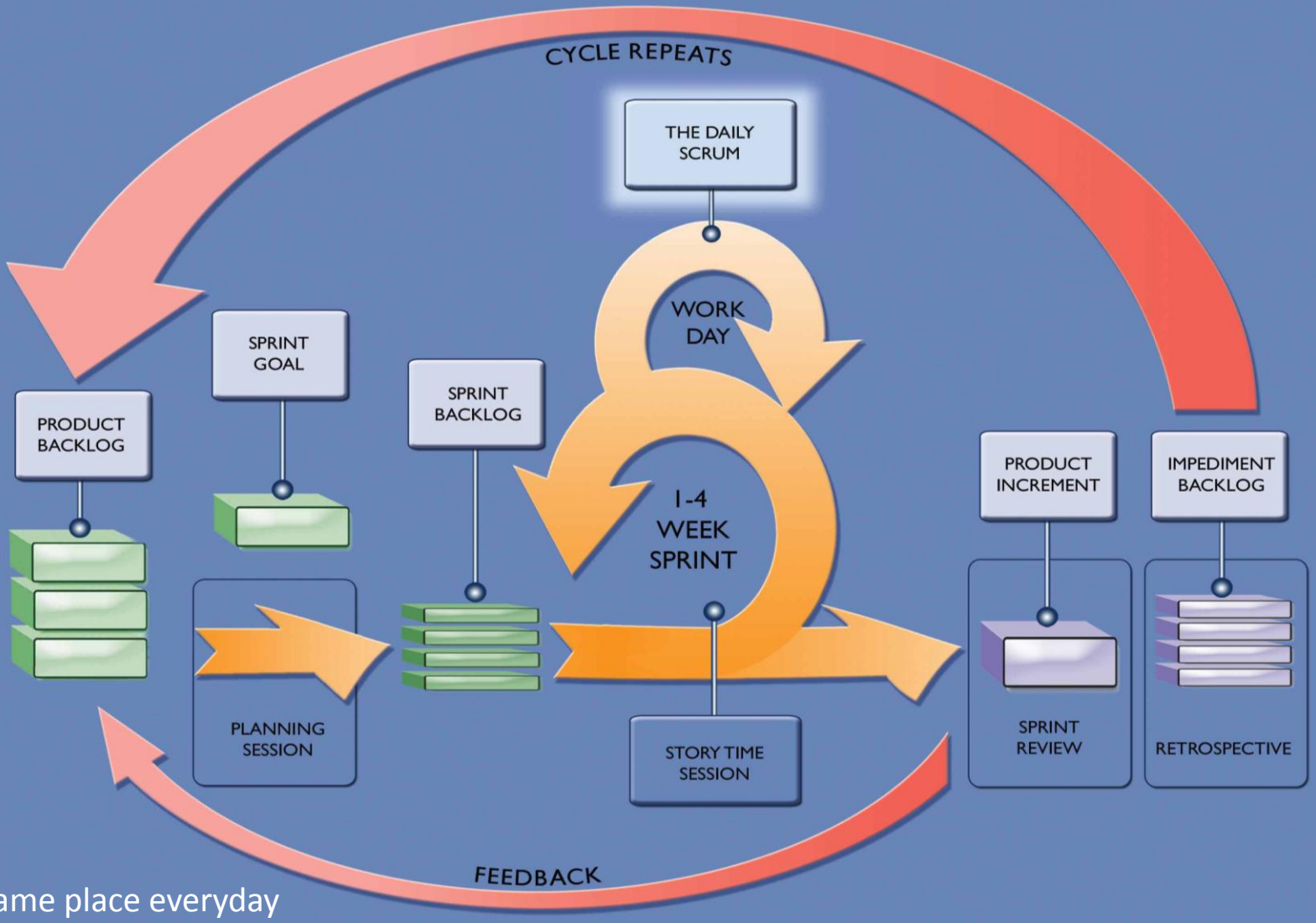
Ceremonies – Execution, Inspection and Adaption



Sprint Execution:

- The team is left alone during a Sprint to focus on the commitments they made
- The team can seek outside help, but no one outside the team should disrupt their rhythm by interrupting or providing different direction
- Sprints can be abnormally terminated if:
  - the Team is unable to achieve the Sprint Goal (e.g., the technology is unworkable, or the team is interfered with)
  - the Sprint Goal is no longer valid (e.g., business priorities change)
- If change continues to come into Sprints, then the Sprint length may be too long
- Keep in mind….changes only have to wait until the next Sprint!!

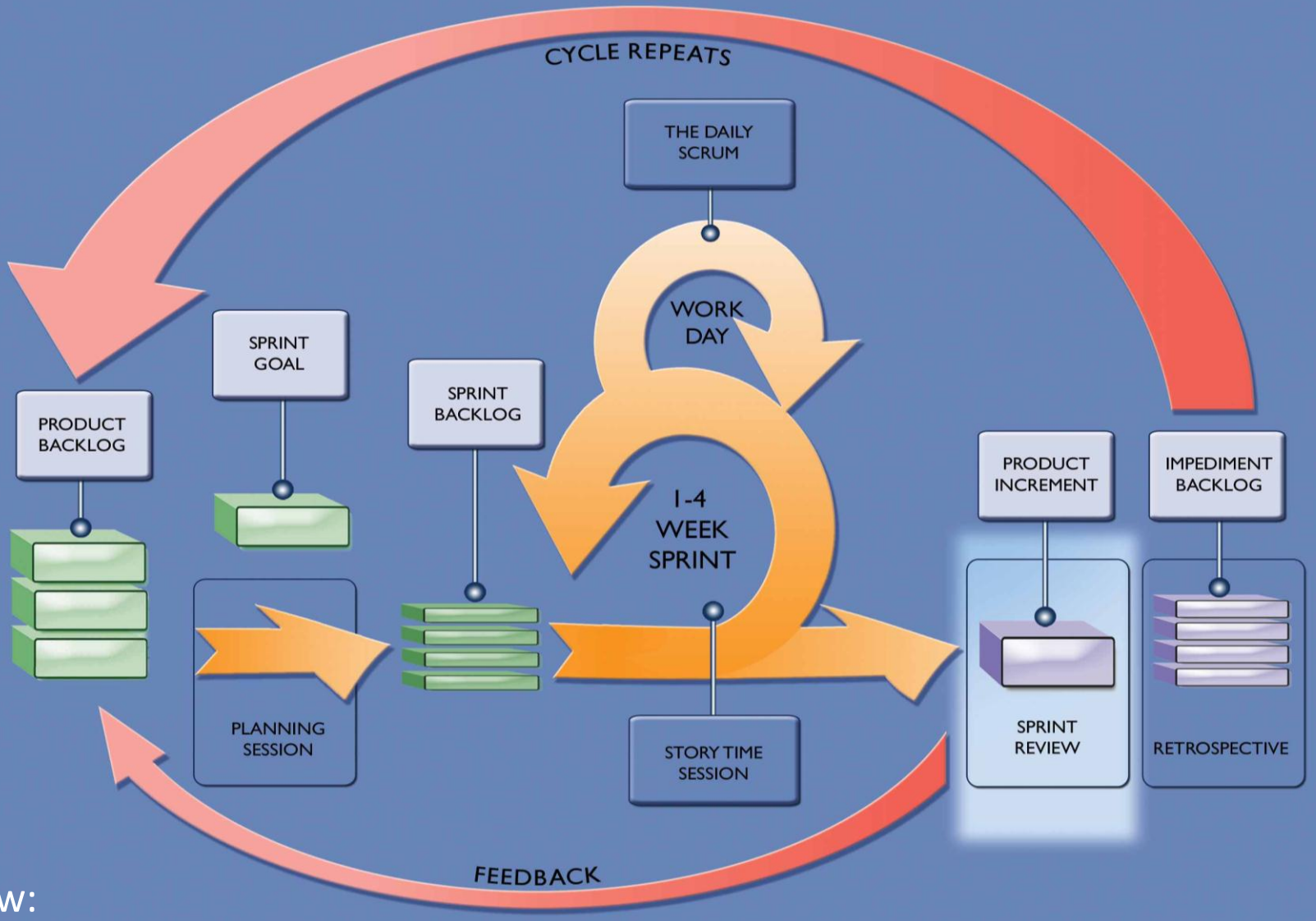
Ceremonies: The Daily Scrum



Daily Scrum

- Same time, same place everyday
- Standing, 15 minutes max
- Each team member explains:
  - What he/she accomplished since the last scrum
  - What he/she is going to do until the next scrum
  - What impediments are in the way

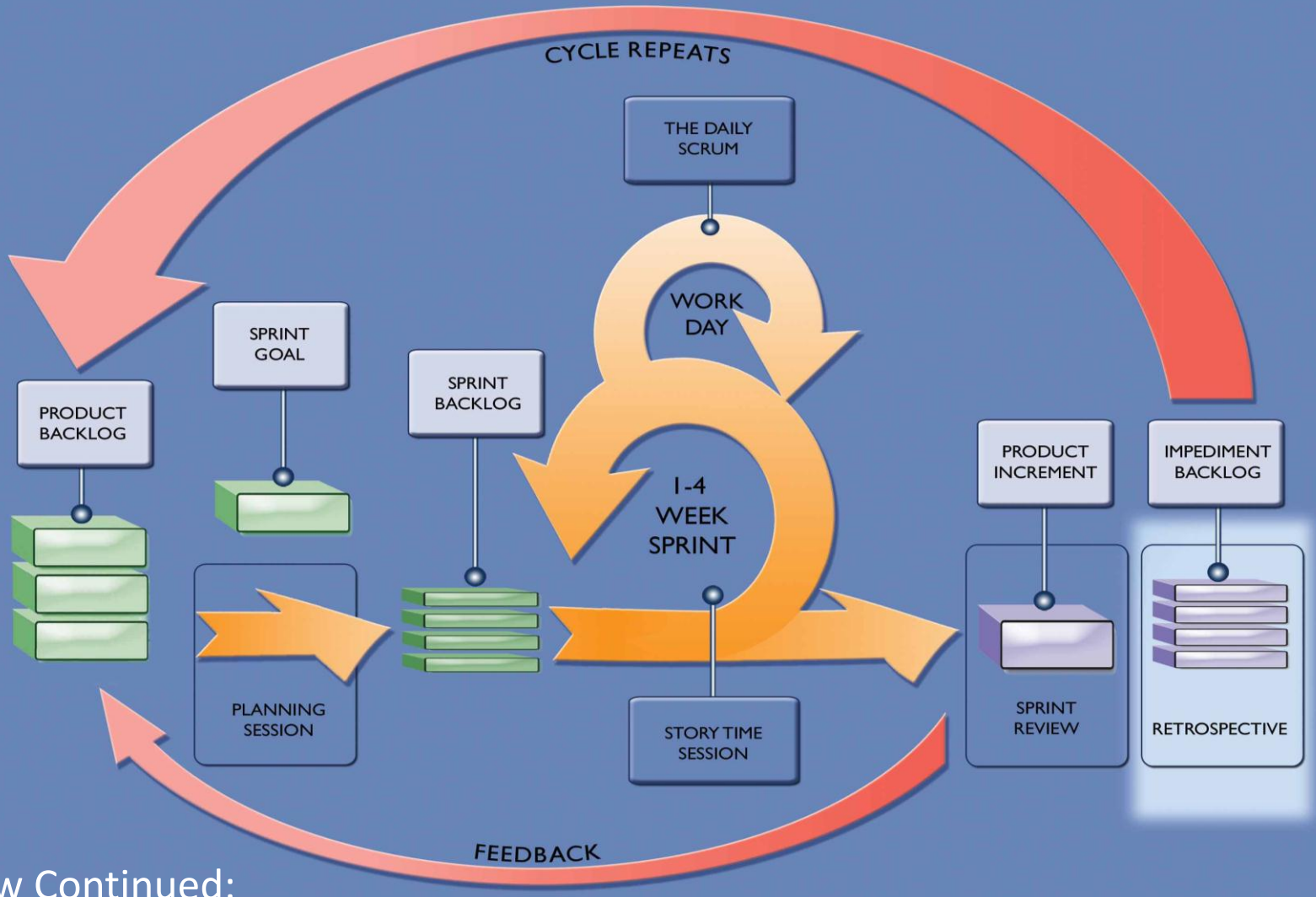
Ceremonies – Execution, Inspection and Adaption



Sprint Review:

- Scrum Team, ScrumMaster, Product Owner and stakeholders collaborate on what has been completed
  - Product owner identifies what has been done
  - Team discusses problems and successes of the Sprint
  - Team demonstrates work
  - Final Acceptance of the Sprint

Ceremonies – Execution, Inspection and Adaption



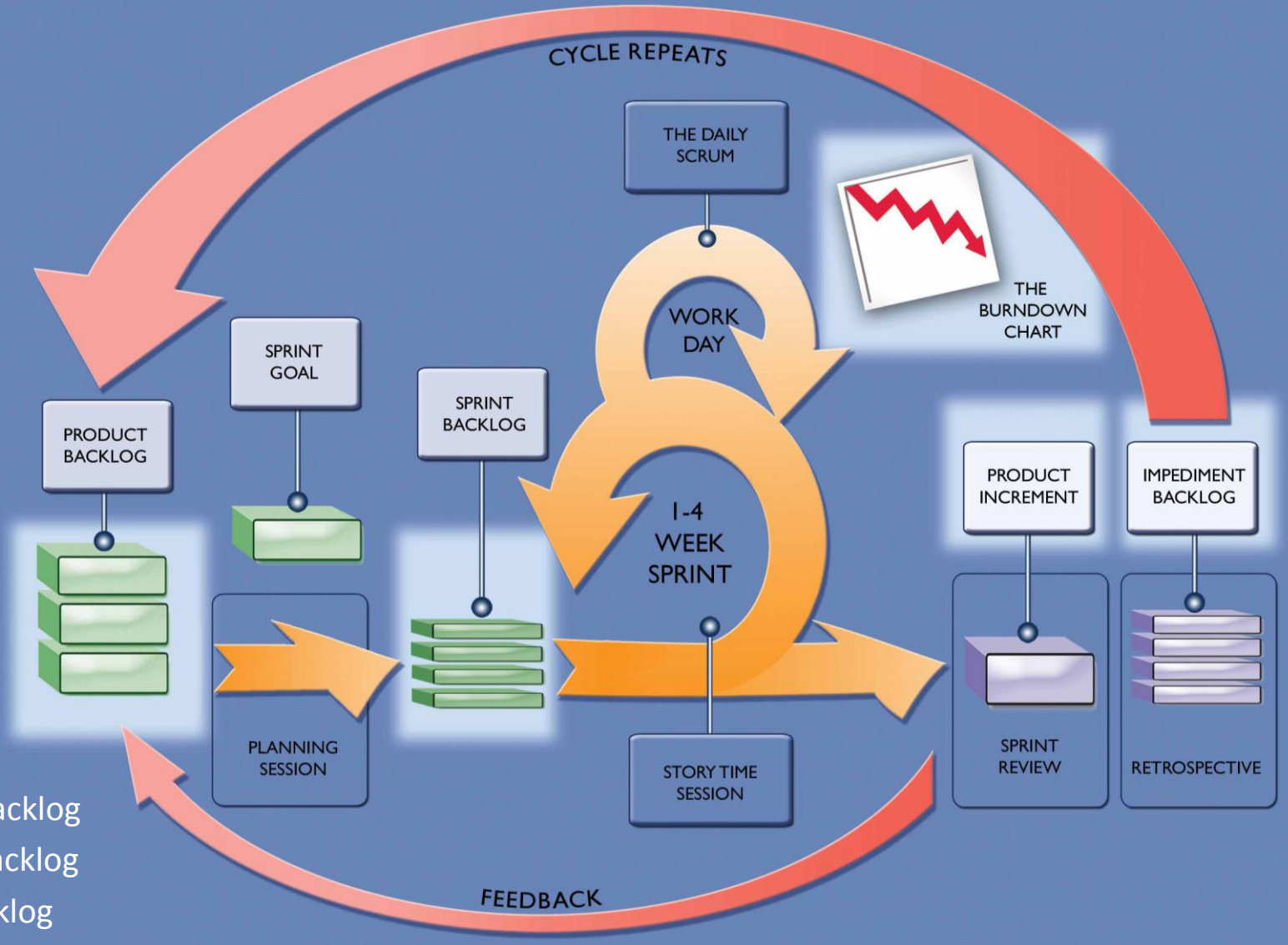
Sprint Review Continued:

- The Team calculates the actual story points earned and the actual Ideal Developer Days
- Retrospective
  - Team Reflects on the past Sprint and records:
    - Things that went well and they would like to see continue
    - Things that could be improved

On your projects today:

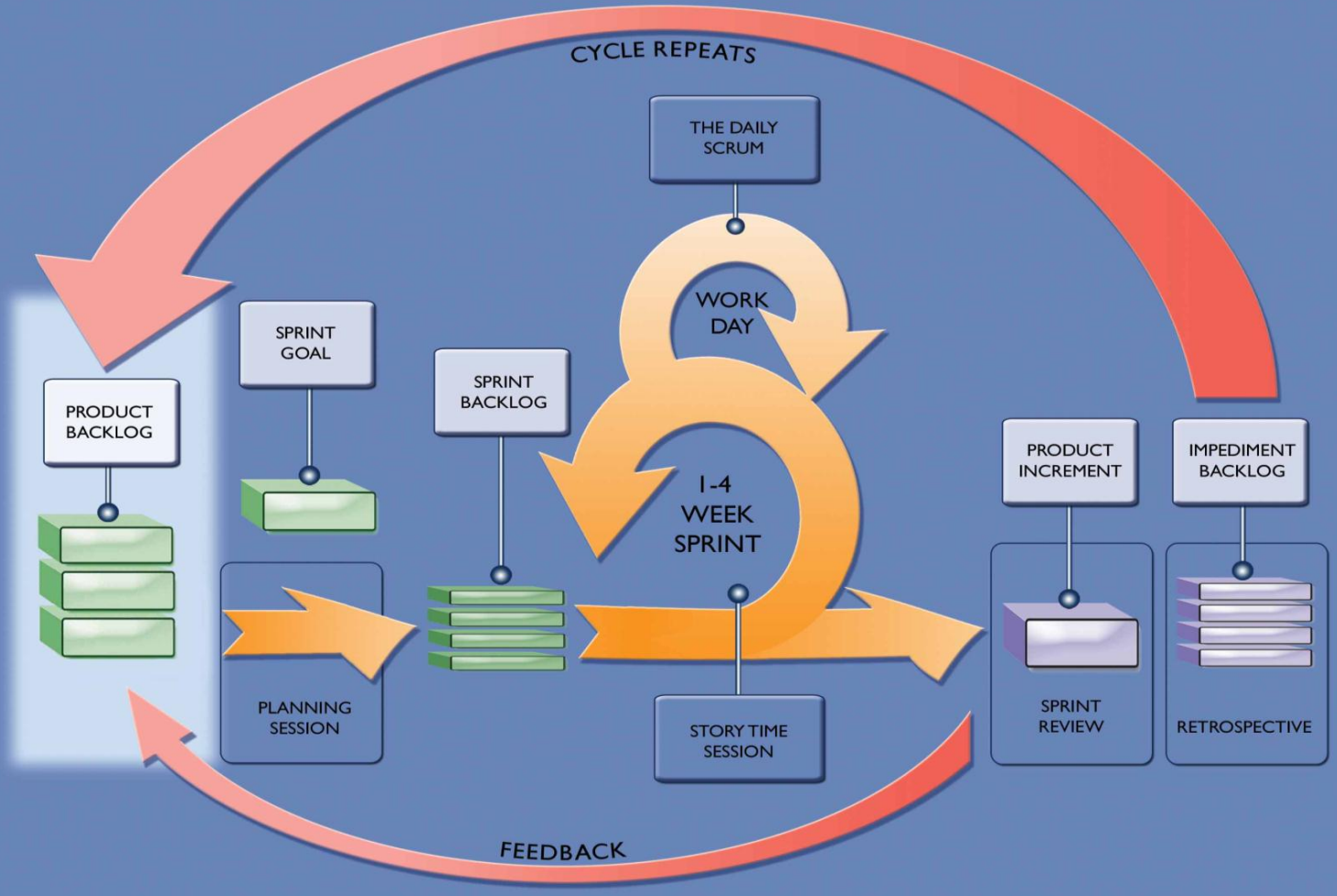
- What types of ceremonies do you have?
- Is there any room for improvement, simplification or thinning out?
- Are the right people always there?
- Take 10 minutes and answer the questions, then we will discuss.
- Report back a summary of the responses, for example
    - 2 people inspect but do adapt on their projects
    - 1 person does both
- Please raise your hand when you are done.

Scrum Artifacts



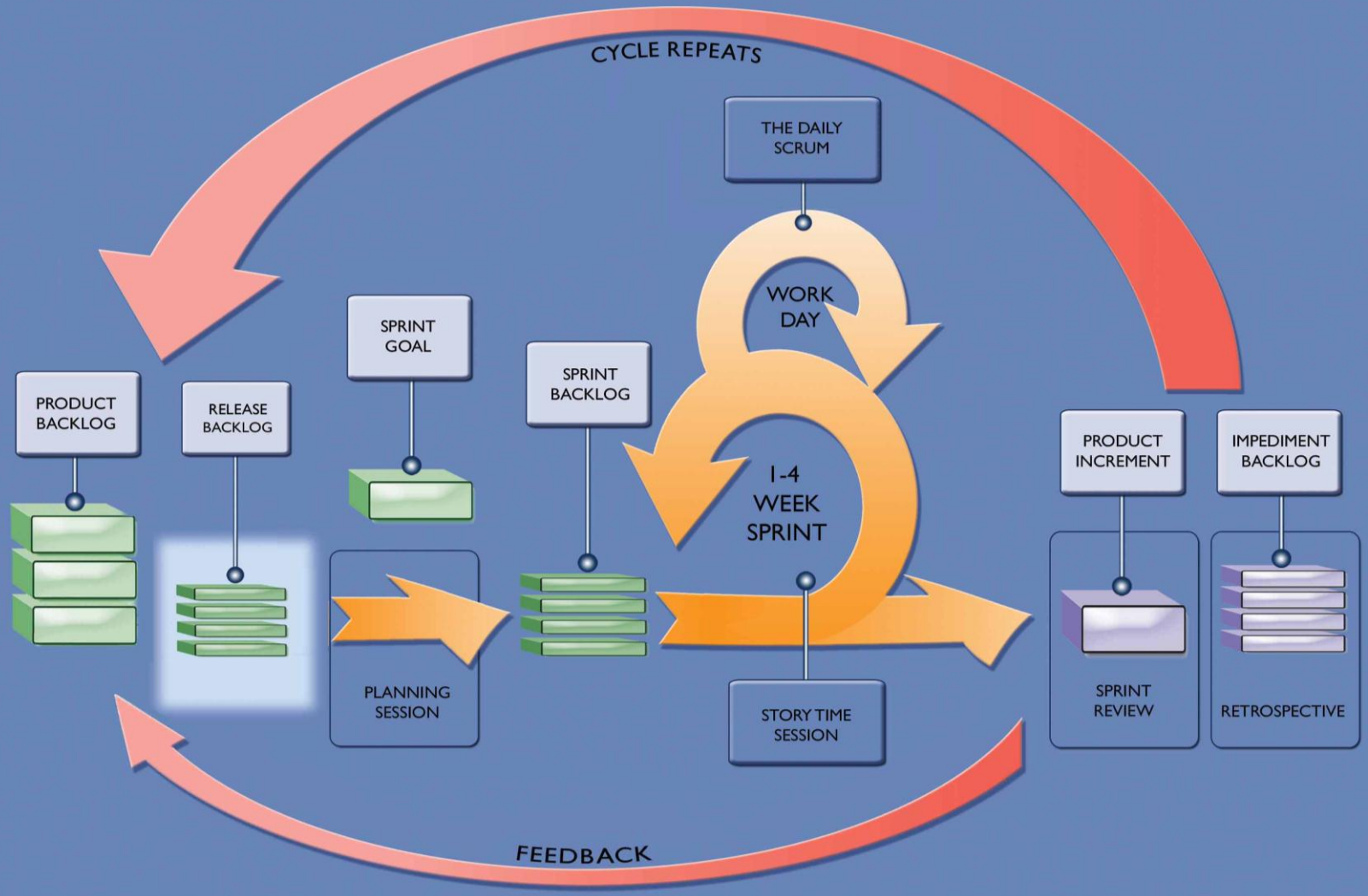
- Product Backlog
- Release Backlog
- Sprint Backlog
- The Burndown Chart
- Story Cards
- Task Cards
- Product

Artifacts: The Product Backlog



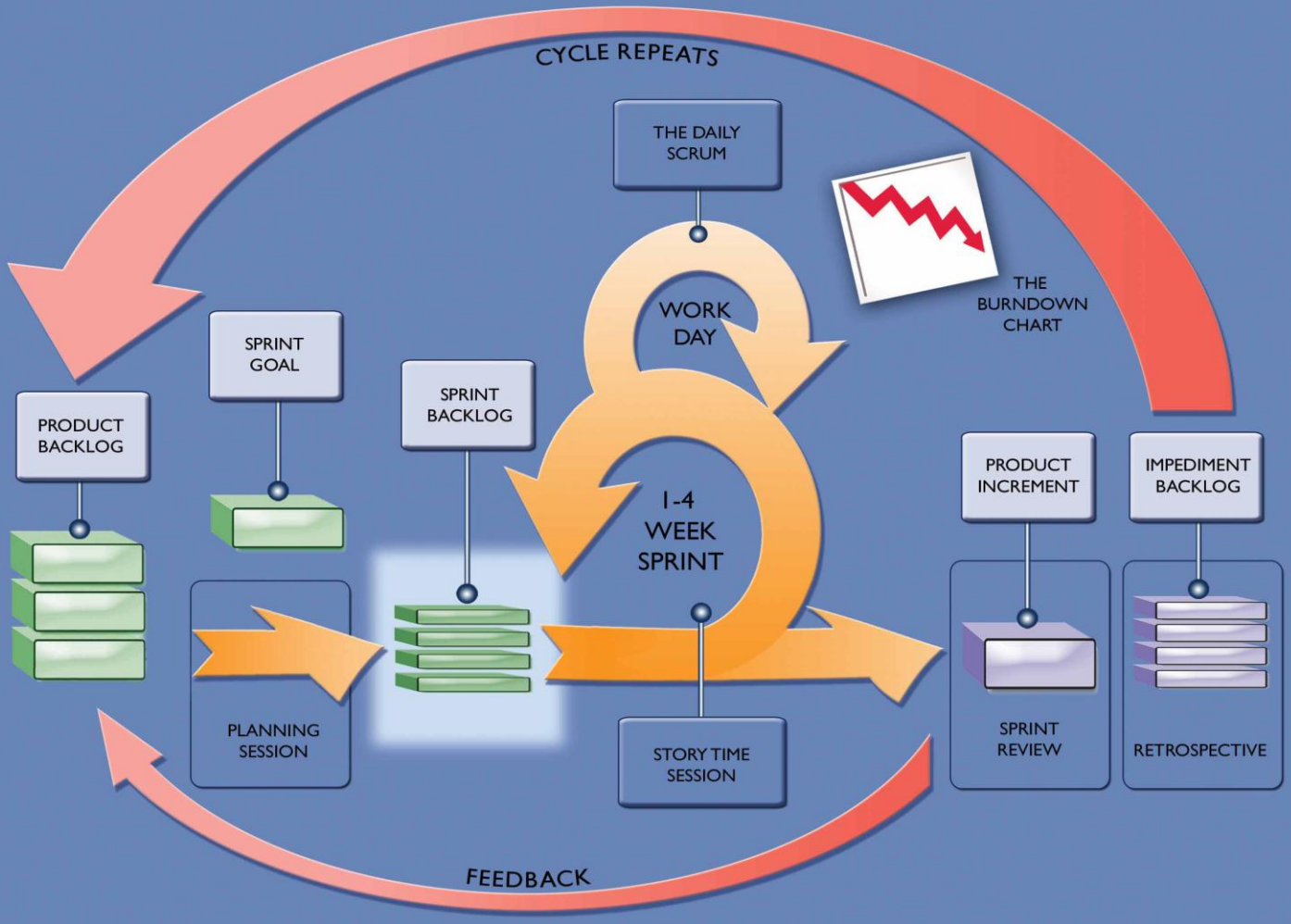
- A prioritized list of functional and non-functional requirements (all desired work) for the project. Also referred to as Story Cards or Stories.
- The Product Owner is responsible for maintaining the Product Backlog contents and ensuring that Business Value is applied to match the needs of the business. The Product Backlog at this level includes the Stories and the Business Value of the Stories.
- The Comprehensive Product Backlog includes Story Cards which have the following attached: Business Value, Prioritization, Estimations and acceptance criteria ("done").
- The Product Backlog is continually being updated with additions, deletions and modifications to stories.

Artifacts: Release Backlog



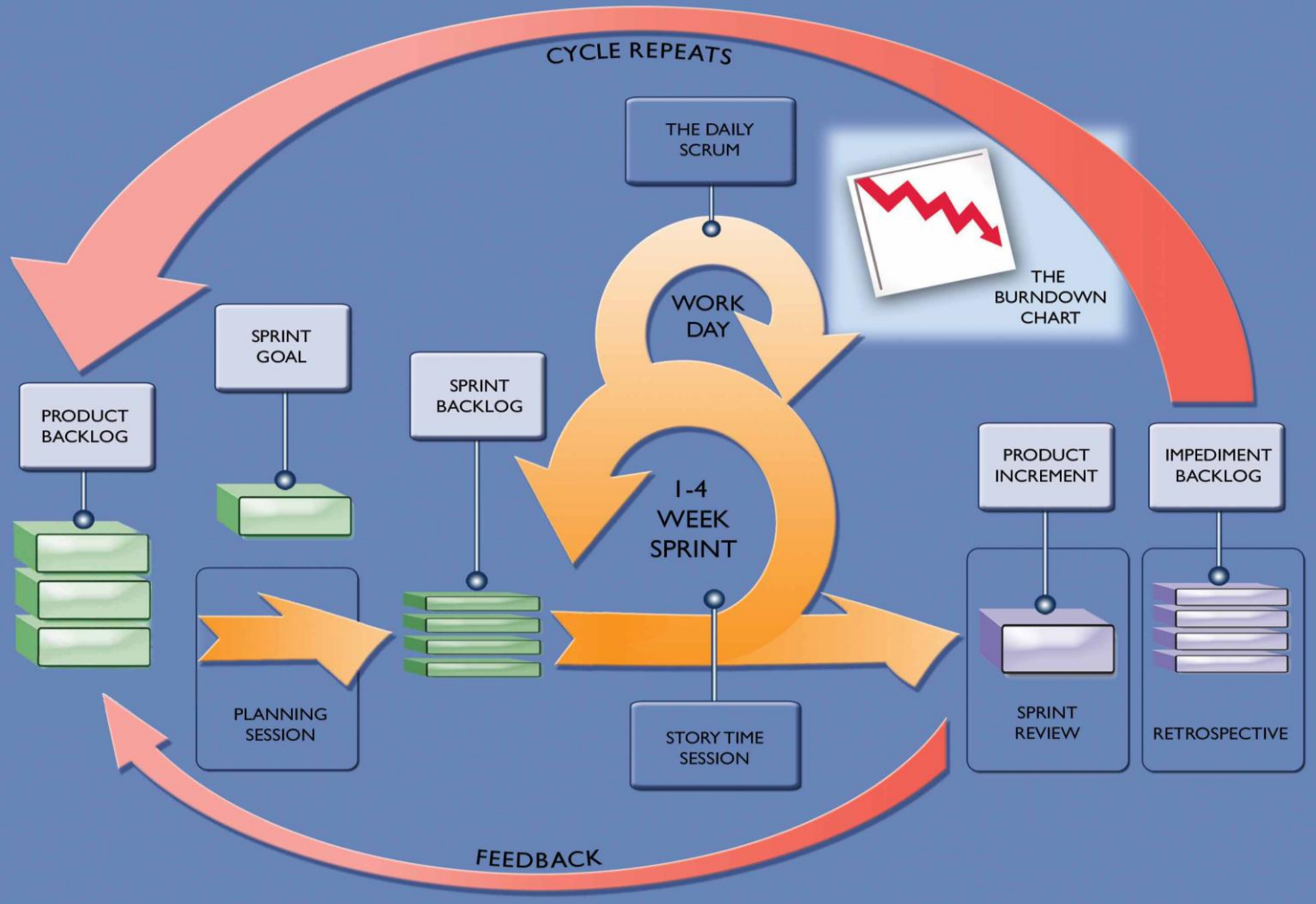
- The list of stories associated to the Release.
- At this level the Release Backlog will include high level estimation known as story points
- The Release Backlog is an output from the release planning meeting
- The Release Backlog includes all stories anticipated to be delivered at the end of the

Artifacts: The Sprint Backlog



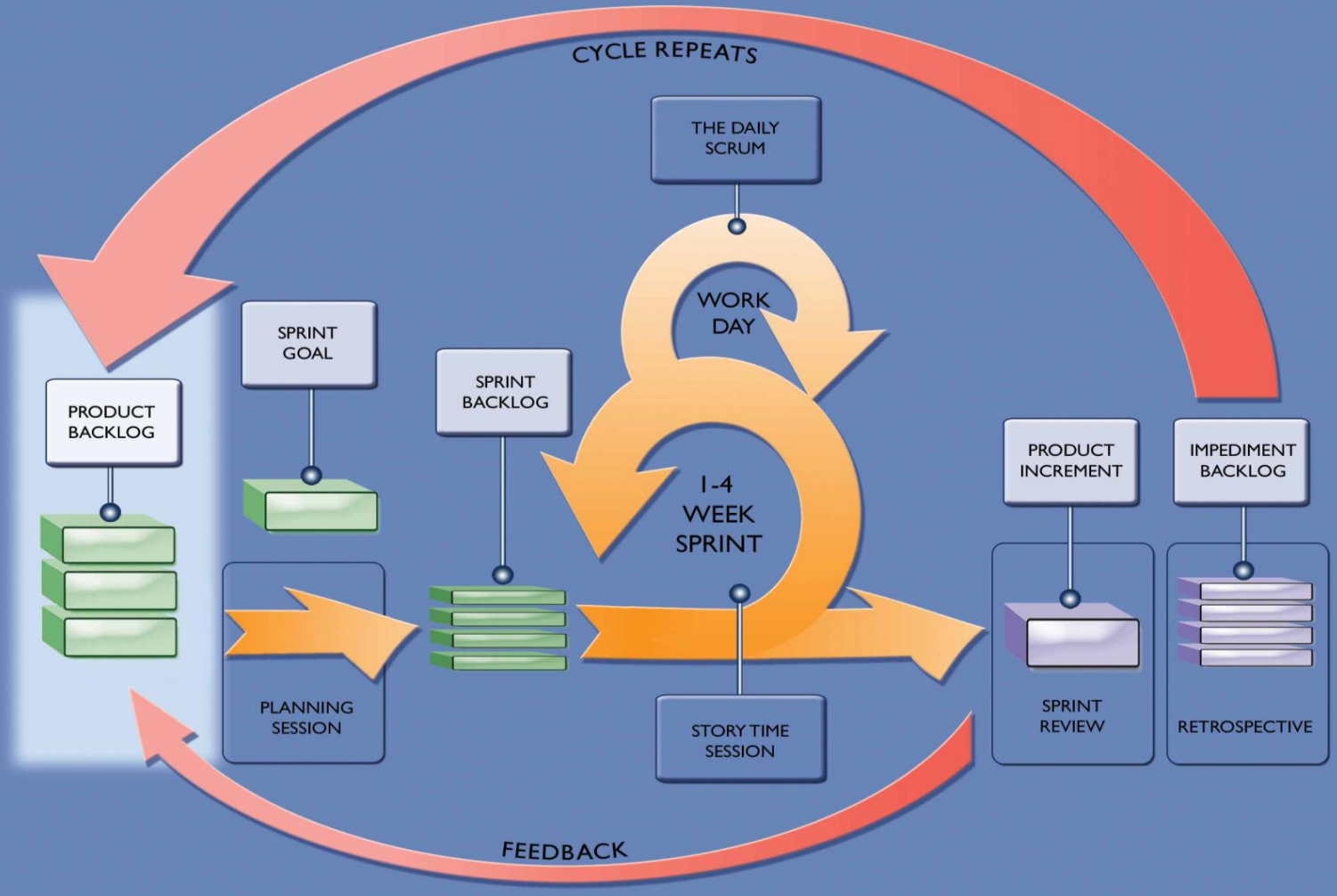
- The list of stories and associated tasks and estimates that define a Team's work for a Sprint
- The Sprint Backlog is an output from the Sprint Planning meeting
- The Sprint Backlog belongs to the Team
- It's the tactical plan for turning the committed backlog stories into system functionality
- The Sprint Backlog is used:
  - to create a physical "team board" which is an information radiator in team rooms
  - In electronic format to track daily burndown information

Artifacts: The BurnDown Chart



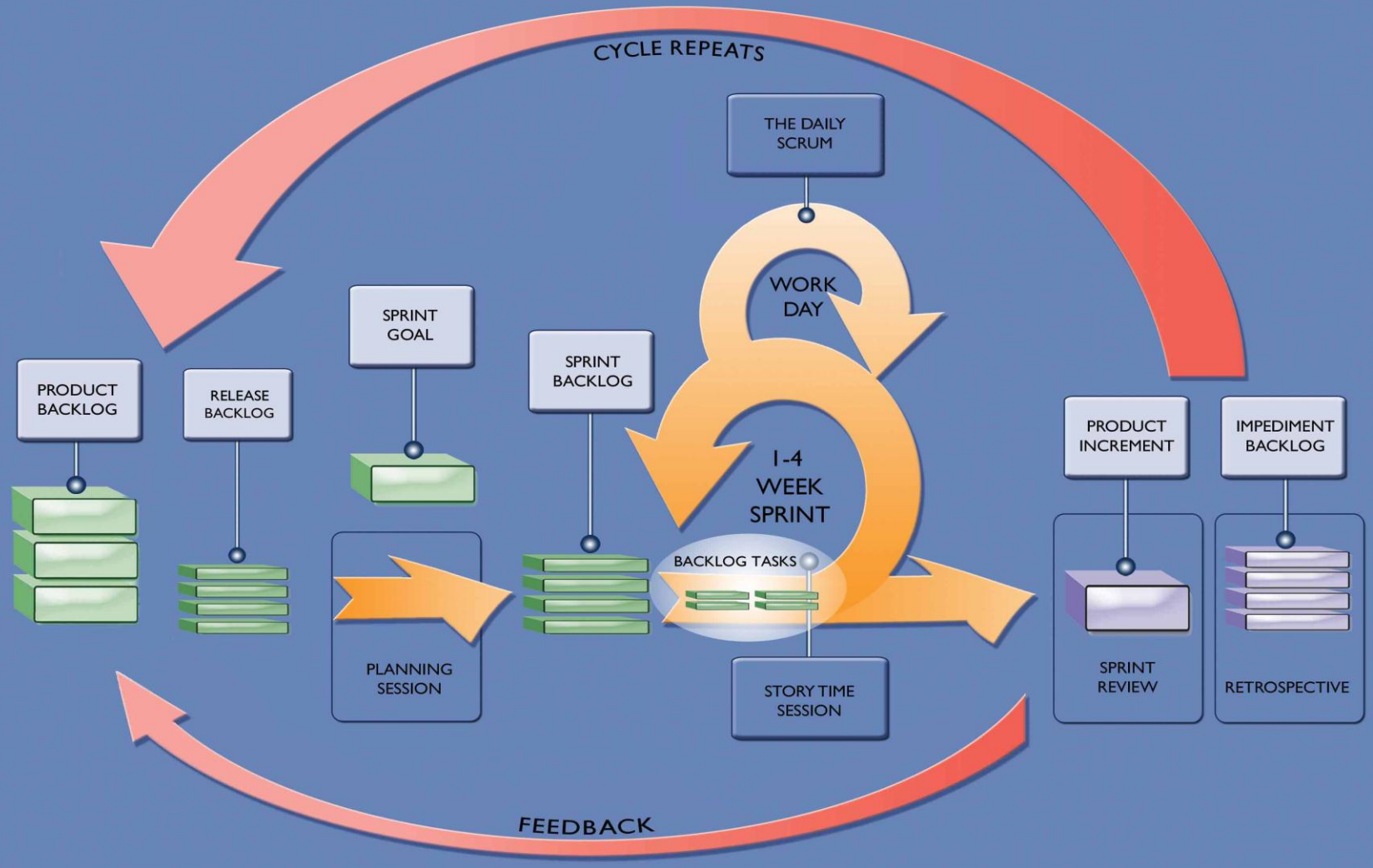
- A graph of work remaining plotted across time
- Plotted daily, the trend line gives an indication of progress against Sprint tasks and whether they can be completed by the end of the Sprint
- The BurnDown chart is used by the team to maintain a sustainable pace
- One quick look and anyone can tell if the team is ahead or behind in their promise to deliver

Artifacts: Story Card



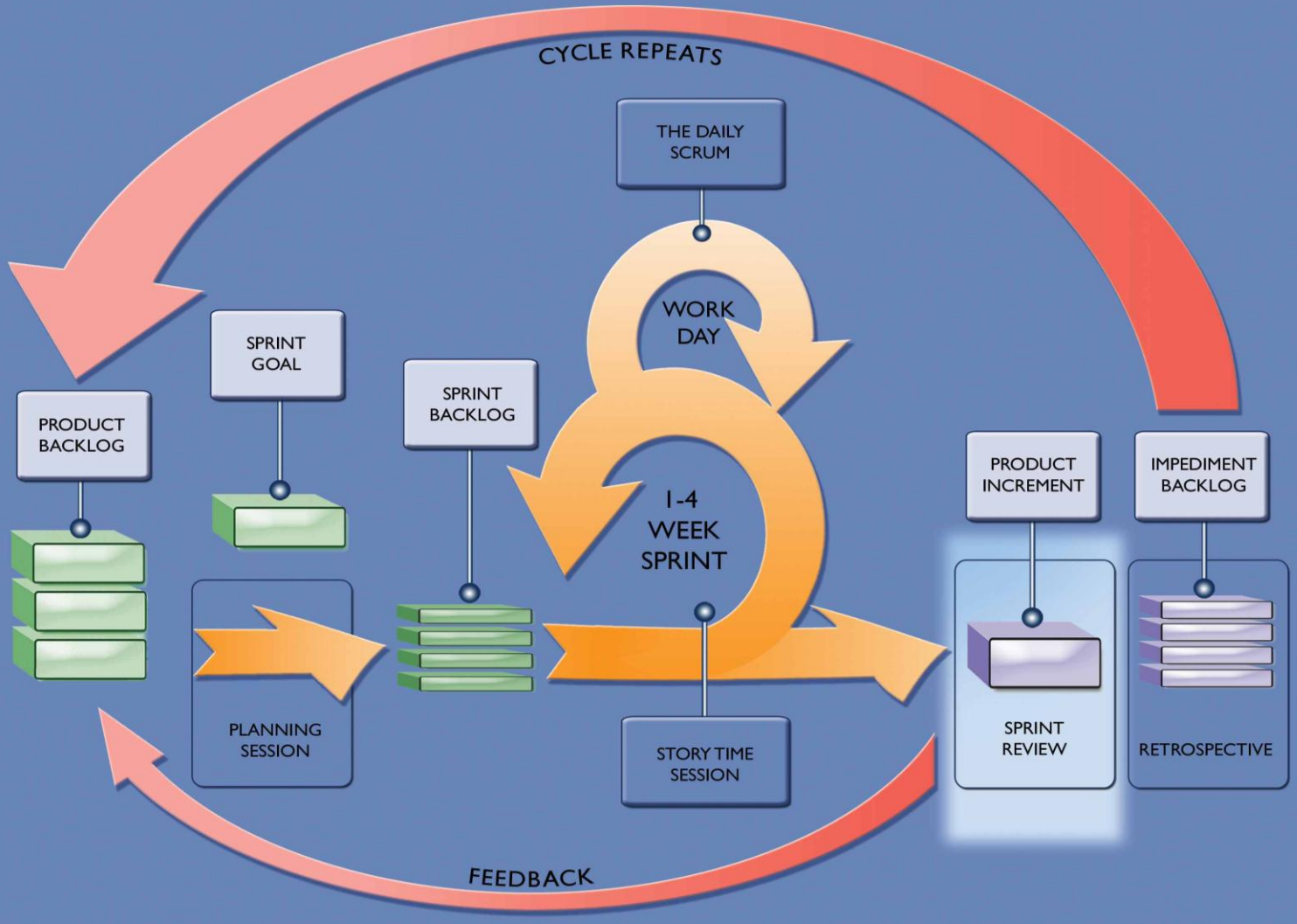
- Each customer request is placed on a story card
- Story cards are often an index card and/or a post-it
- Product Owners use the card as a reminder to have a conversation with the team
- The team will task and estimate the work
- Story Cards are often the primary form of requirements documentation

Artifacts: Task Card



- Each Story will require one or more tasks to create the solution
- Each task is recorded on a Task Card
- All task cards are estimated
- Team members volunteer to perform a task by placing their initials on the card

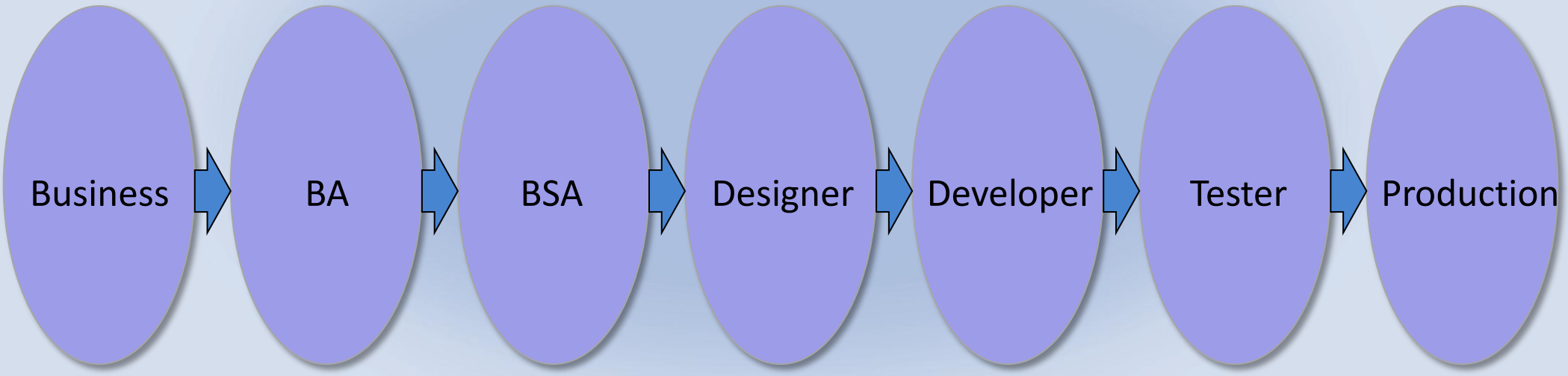
Artifacts: Product



- The outcome from the Scrum process is the creation of customer valued products and/or services
- Generally several sprints are necessary to get to an assembled packet of product units that would be considered a potentially shippable product
- Each Release generally has a theme or goal that the assembled products will satisfy

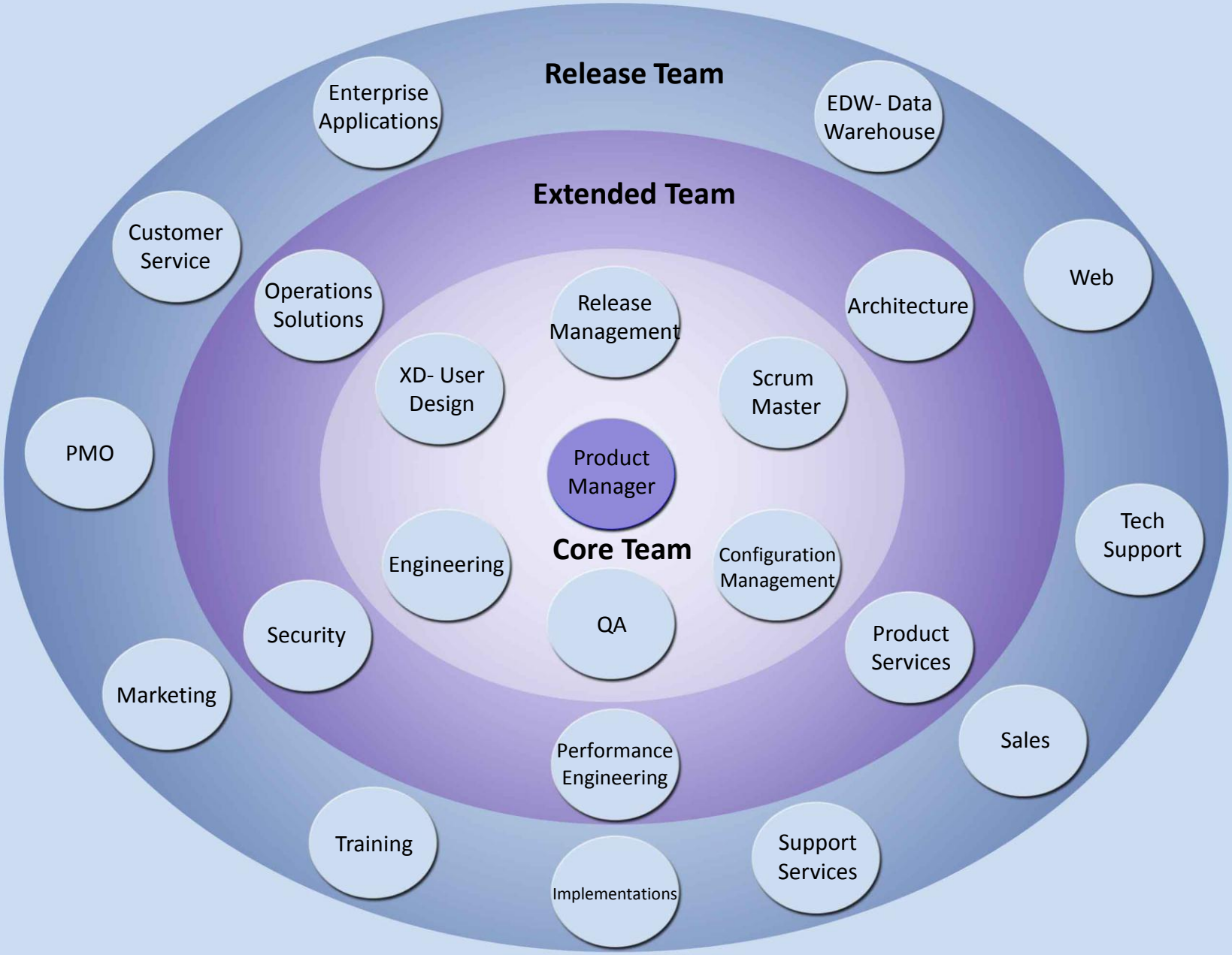
Process Work Flow

Waterfall Team Structure: Work flows serially through functions.



Integrated Teams – Human Work Cell

Agile Team Structure: Work is distributed in parallel across functions



Discussion

On your projects today –

- How are you organized?
- What is your teams physical distribution?
- Take 10 minutes and answer the questions, then we will discuss.
- To report back, summarize as in previous exercise.
- Please raise your hand when you are done.

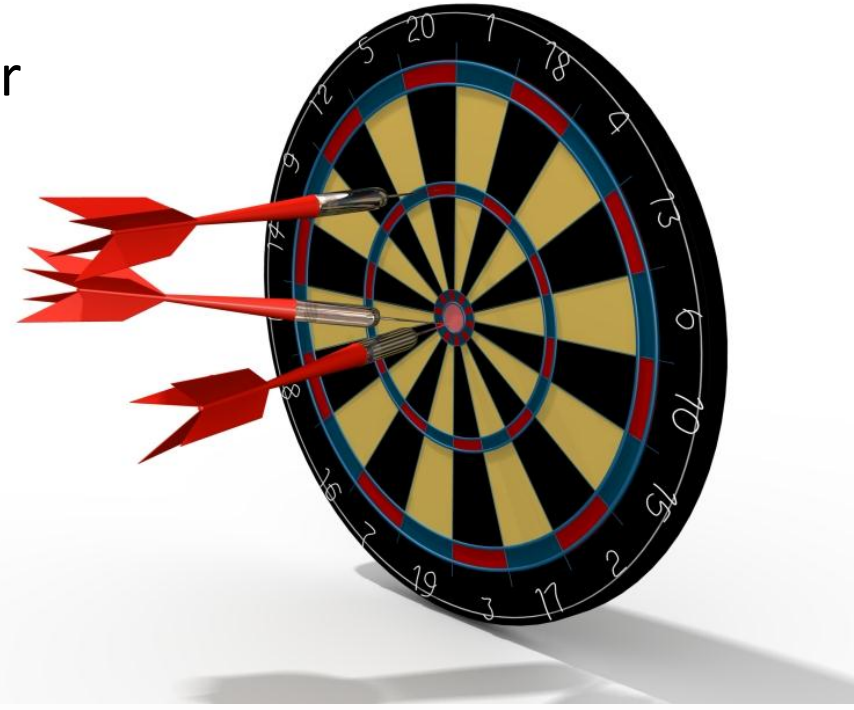
Knowledge United
Leverage the Power of Learning

Scrum Building Blocks

Story Time

Assumption Exercise

Six friends visited their local club to play at a darts competition. The competition entry fee was $25 per person. Prize money was $500 for the winner, $200 for the runner-up and $100 for third-place. There were no other prizes.



None of the friends won a single game. There were no disqualifications, and yet the friends came away collectively $300 in profit from having played. How?

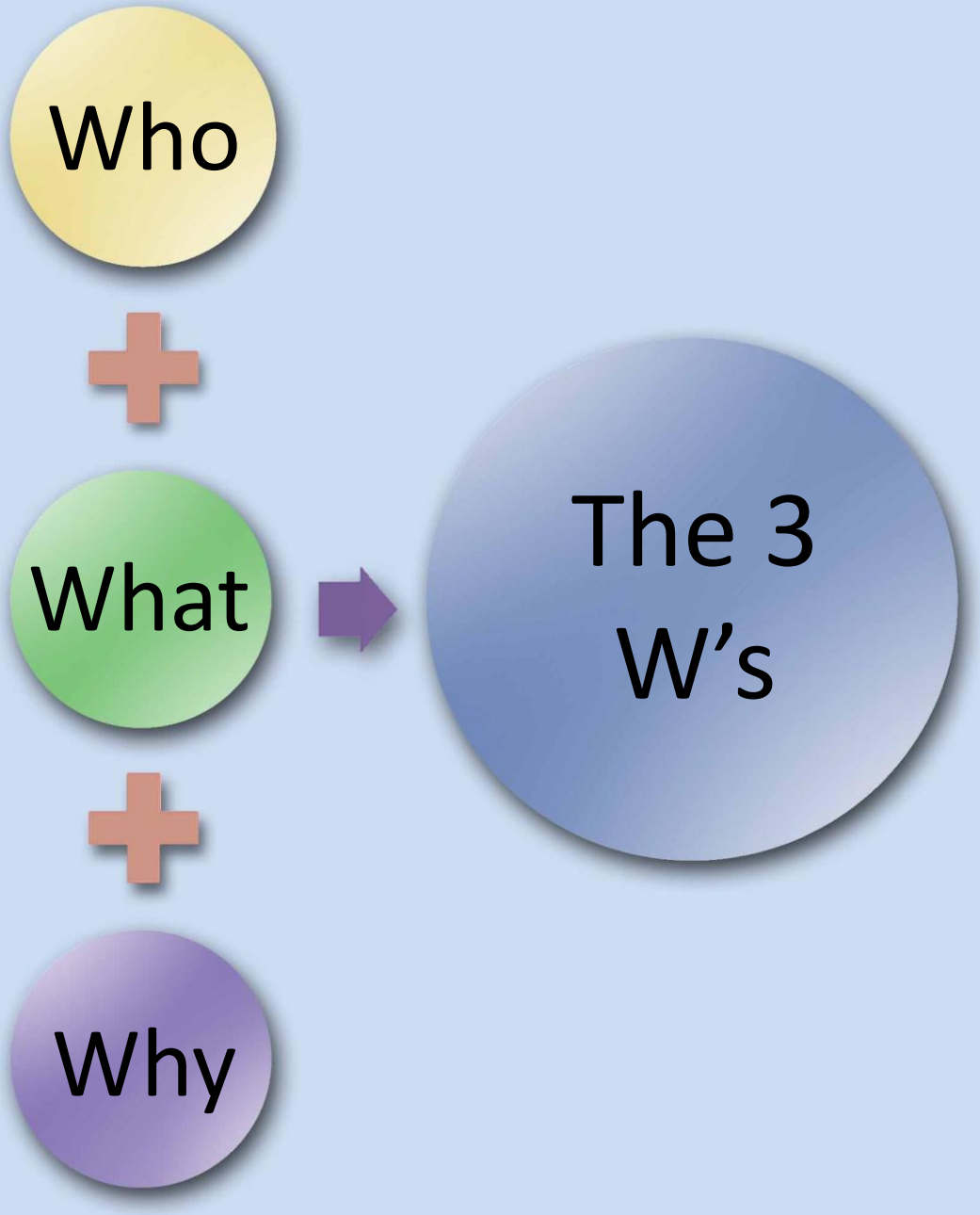
Knowledge United
Leverage the Power of Learning

What is a User Story?

- A placeholder for more detailed conversations, not a requirements write up
- A short communication of the feature that is of value to the end user/customer
- A representation of progress of the overall initiative to the end user/customer
- A short written description of the feature used for planning and serves as a reminder
- A progressive elaboration of valued features
- An artifact

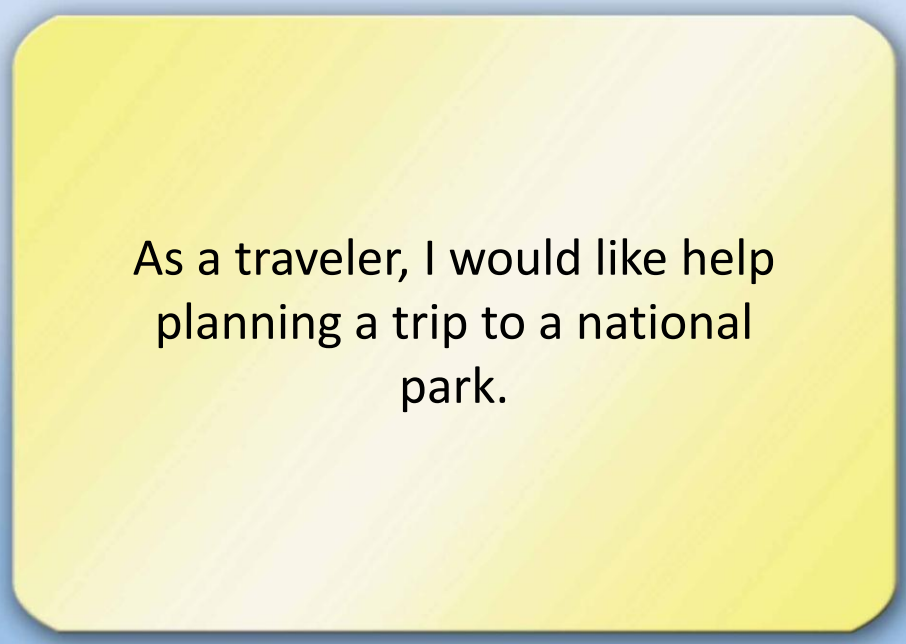
Knowledge United
Leverage the Power of Learning

The 3 W's



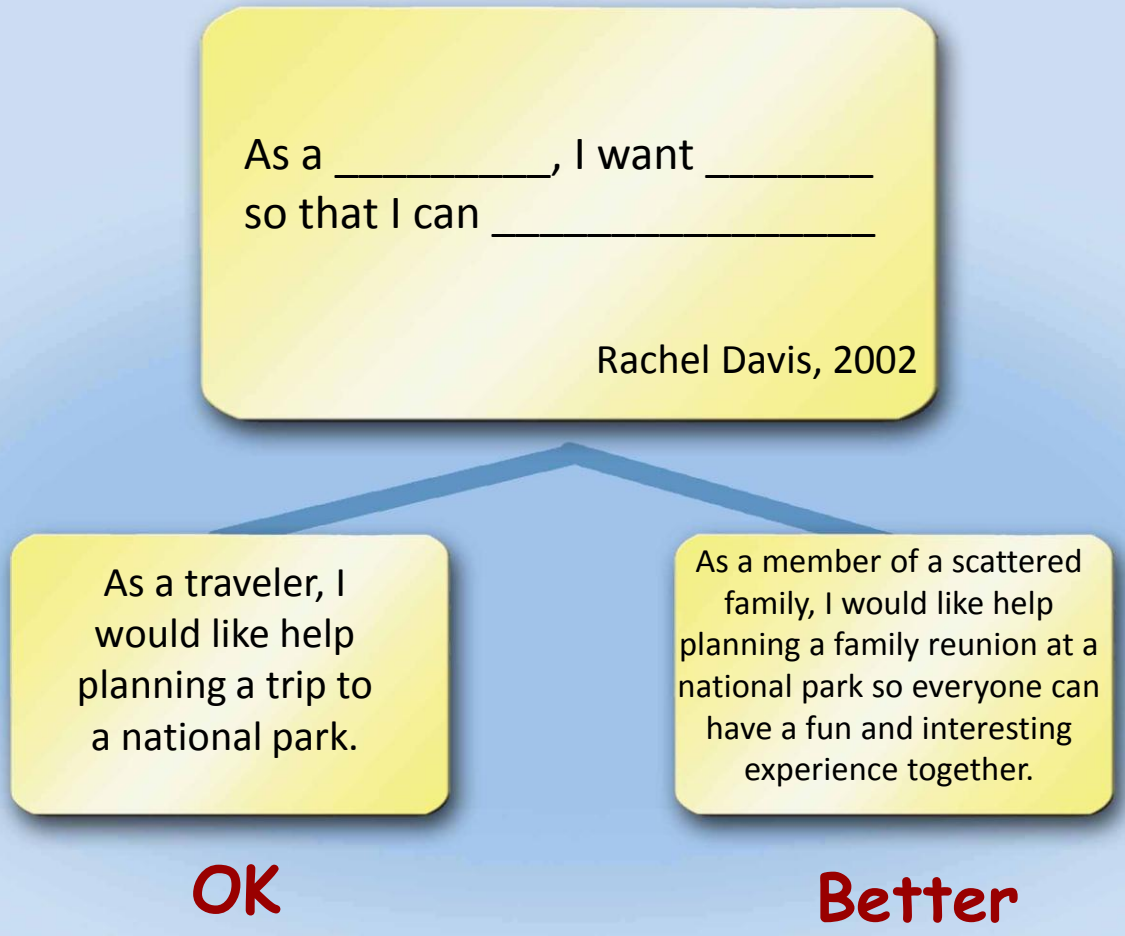
Scrum Requirements as User Stories

What does a Story look like?



Scrum does not mandate how product backlog items are described, *Cohn 2004*

Scrum Requirements as User Stories



A story should have a user, something they value and how they will benefit.

Scrum Requirements as User Stories

Some stories have key conditions, constraints and non-functional requirements.

As a member of a scattered family, I would like help planning a family reunion at a national park so everyone can have a fun and interesting experience together.

OK

As a member of a scattered family with limited budget and time, I would like help planning a family reunion at a national park so everyone can have a fun and interesting experience together.

Better

Epic or Story?

As a traveler, I would like help planning a trip to a national park.

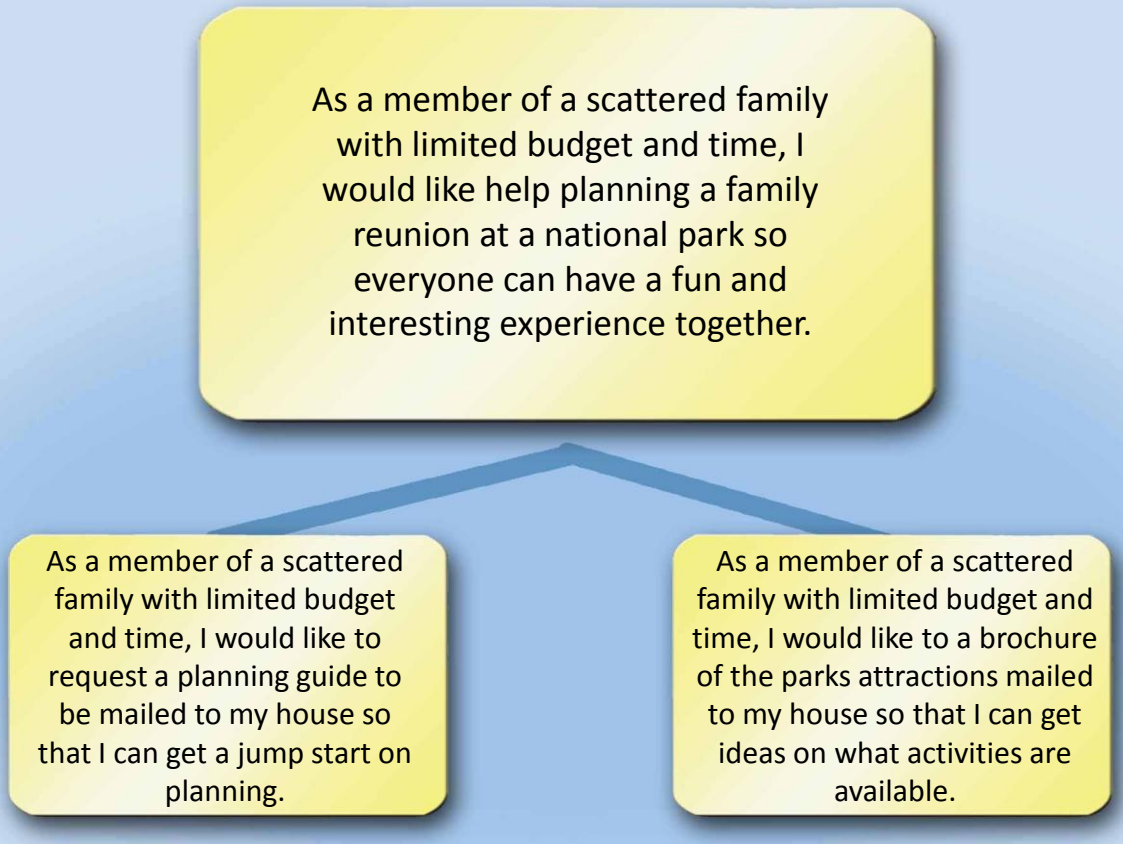
As a member of a scattered family with limited budget and time, I would like help planning a family reunion at a national park so everyone can have a fun and interesting experience together.

As a planner for high school educational trips, I would like to help planning a school trip at a national park so everyone can have a fun learning experience.

As a planner for high school educational trips, I would like to help planning an overnight school trip at a national park so everyone can have a fun learning experience.

As a planner for high school educational trips, I would like to help planning a school day trip at a national park so everyone can have a fun learning experience.

User Stories – The Details as Sub-Stories



Details can be in the form of smaller sub-stories

User Stories – The Details as Acceptance Criteria/Tests

As a member of a scattered family with limited budget and time, I would like to request a planning guide to be mailed to my house so that I can get a jump start on planning.

Acceptance Test

- Shows a list of restaurants
- Demonstrates the food and average expense
- Verifies safety concerns

Details can be in the form of tests

(conditions of satisfaction or acceptance by the product owner)

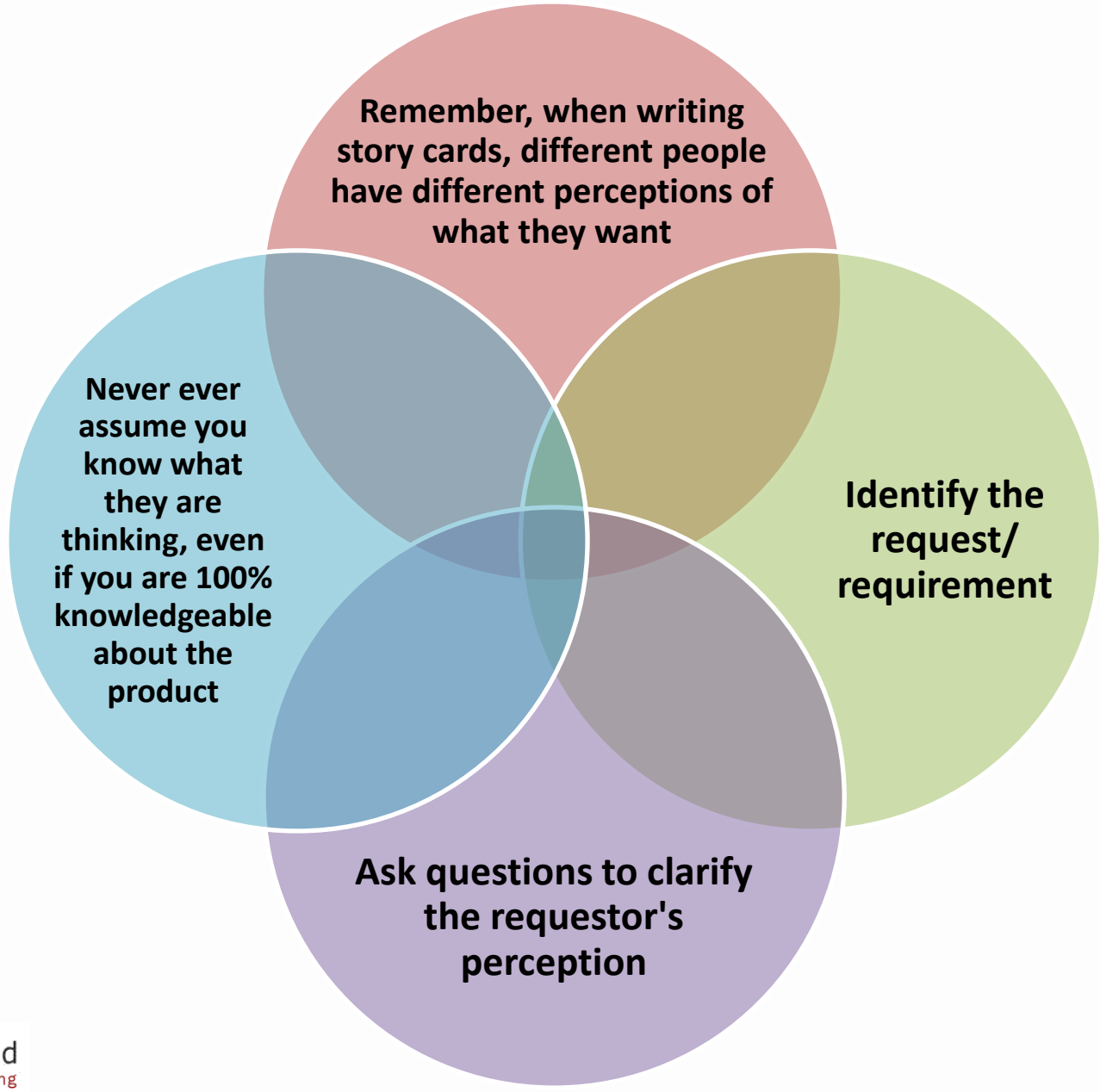
Exercise - Same Item different description

Instructions:

- Each team will have 15 minutes for this exercise
- First, each individual should write down the days of the week
- Now associate a color with each day
- Write down the color next to each day and a short explanation for why you selected that color
  - While you are working on this individually, I will move each of you into a room.
  - Once everyone has completed their list, discuss the differences and why
    - Select one color for each day to represent your team
    - Select someone to report back to us from your team
- I will move each of you back to the main training room and we will review

Knowledge United
Leverage the Power of Learning

Perception



Knowledge United
Leverage the Power of Learning

Intuit Sample Storycards:

Example 1
As an FI Admin, I would like to search for a user in purchase rewards, get suggestions, and select a suggested user.

Example 2
As an FI Admin, I would like to view the Purchase Rewards statuses for all the accounts of the single user I searched

Example 3
Technical Story: As an engineer I'd like a common way to load the request context from rest request headers, and to set request headers from the local request context

Example 4
As an FI admin, I want a monthly report to identify records I need to manage in the ACH exception process (ACH Report)

Done - Intuit Example #1

Story Done Checklist

- ❑ All story tasks complete
- ❑ Code coverage goals achieved
- ❑ All testing is complete
  - ✓ Story
  - ✓ Functional
  - ✓ Unit
- ❑ No open (not fixed or deferred) issues
- ❑ Story seen & accepted by customer
- ❑ New feature enhancements recorded as new Stories in the backlog

Iteration Done

- ❑ 10 Commandments followed for all check-ins
- ❑ 100% review for docs, design, code, & automation
  - ✓ At least one rep from Dev & QA required
- ❑ 100% Unit Tests pass
- ❑ 100% User Story Acceptance Tests pass
- ❑ Integration/Functional Tests pass – 98%
  - ✓ All Test Stoppers are fixed
  - ✓ All CRs involving QB crashes are fixed
  - ✓ No hi impact/sev CRs deferred w/o cause
  - ✓ All exceptions understood, documented, and with target Iteration to resolve noted
- ❑ Latest debug/non-debug builds on branches pass all pre-defined BATs/FSTs w/ no project-caused failures
  - ✓ BATS
  - ✓ Core FST
  - ✓ MU FST
  - ✓ Payroll FST
  - ✓ QB Performance FST
  - ✓ Reports FST
  - ✓ Project Specific FST (when created)
- ❑ acceptance tests for Performance Tests are met
- ❑ Automation Complete for features of iteration
  - ✓ Unit Tests automated when applicable by Dev
  - ✓ Risk-Based automation approach utilized by QA

Release Done

- ❑ Code Complete
- ❑ No open defects (Sev 1, Sev 2 and Sev 3); All defects fixed or deferred with PM approval)
- ❑ 100% Functional Test Execution (full feature regressions)
- ❑ 100% End 2 End System Functional Regression.
- ❑ Consumer BATs executed with 100% pass rate (if available)
- ❑ Release Notes and BOM Provided
- ❑ Code Coverage is 80% as possible (may not be able to hit this one)
- ❑ Exhaustive Performance Testing Completed with scalability plan – No performance degradation, meets targeted performance/scalability criteria
- ❑ Security Testing Completed (including Security Audit) – no open security related defects
- ❑ RAS Testing Completed – must meet targeted reliability criteria.

Knowledge United
Leverage the Power of Learning

Do's and Don'ts

DO

- Consider all customer needs / viewpoints
- Remember this is an iterative approach
- Verify assumptions to ascertain if they are facts
- Map the acceptance tests to the stories
- Be clear and concise
- Slice the stories
- Collaborate
- Remember the components of a story
- Ask *why*?

DON'T

- Write all stories from one user's perspective
- Consider the story writing as done (remember iterative)
- Assume all users have the same end result in mind
- Look for perfection
- Never assume
- Use subjective words
- Use and/or
- Forget the definition of done
- Change vocabulary

Tips:

- Ask **open ended** questions
- 3-5 acceptance criteria is good
- If you have 6 or more acceptance criteria, **consider slicing**
- *Clear acceptance tests*
  - Map the acceptance tests to the story
  - Do they both still make sense?
- **Slice** the stories
- Provide context
- Provide the outcome
- Remember the 3 W's

Knowledge United
Leverage the Power of Learning

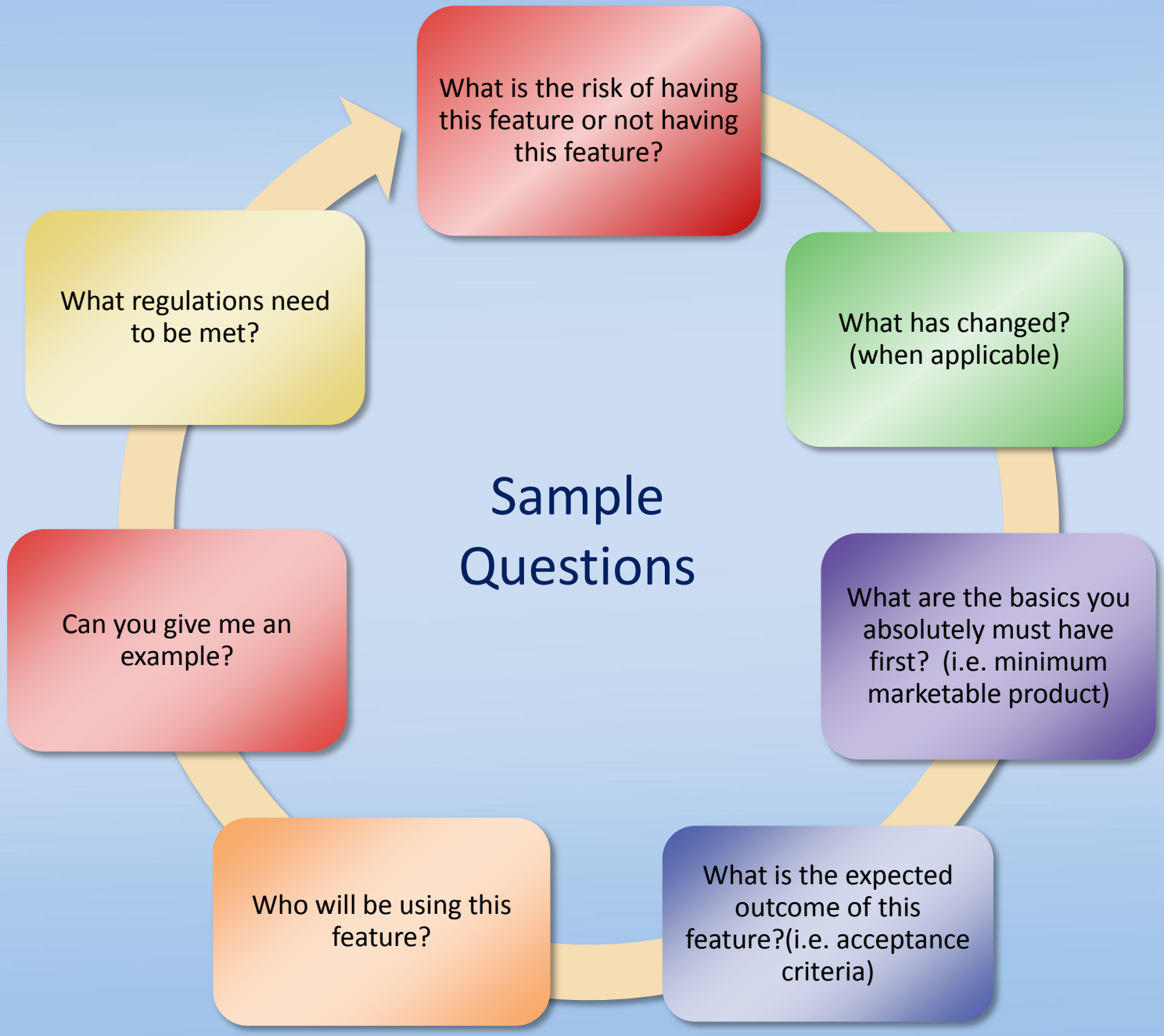
Asking the Right Questions

- Approach your inquiry with genuine curiosity. If you come from a place of curiosity and use appropriate tone and language vs. knowledge and/or arrogance, your customers are more likely to open up instead of feeling defensive.

- First and foremost: Why? Why? Why?
  - Context will open up areas the end user/customer assumes you know about
  - Ask open ended questions that can not be answered with a yes or no

Knowledge United
Leverage the Power of Learning

Asking the Right Questions



Story exercise

A customer has decided to take your team to the movies to reward you for great work. The Product Owner shares the customer vision for this project. Each of you will enjoy the overall movie-going experience.

What parts make up a full story?

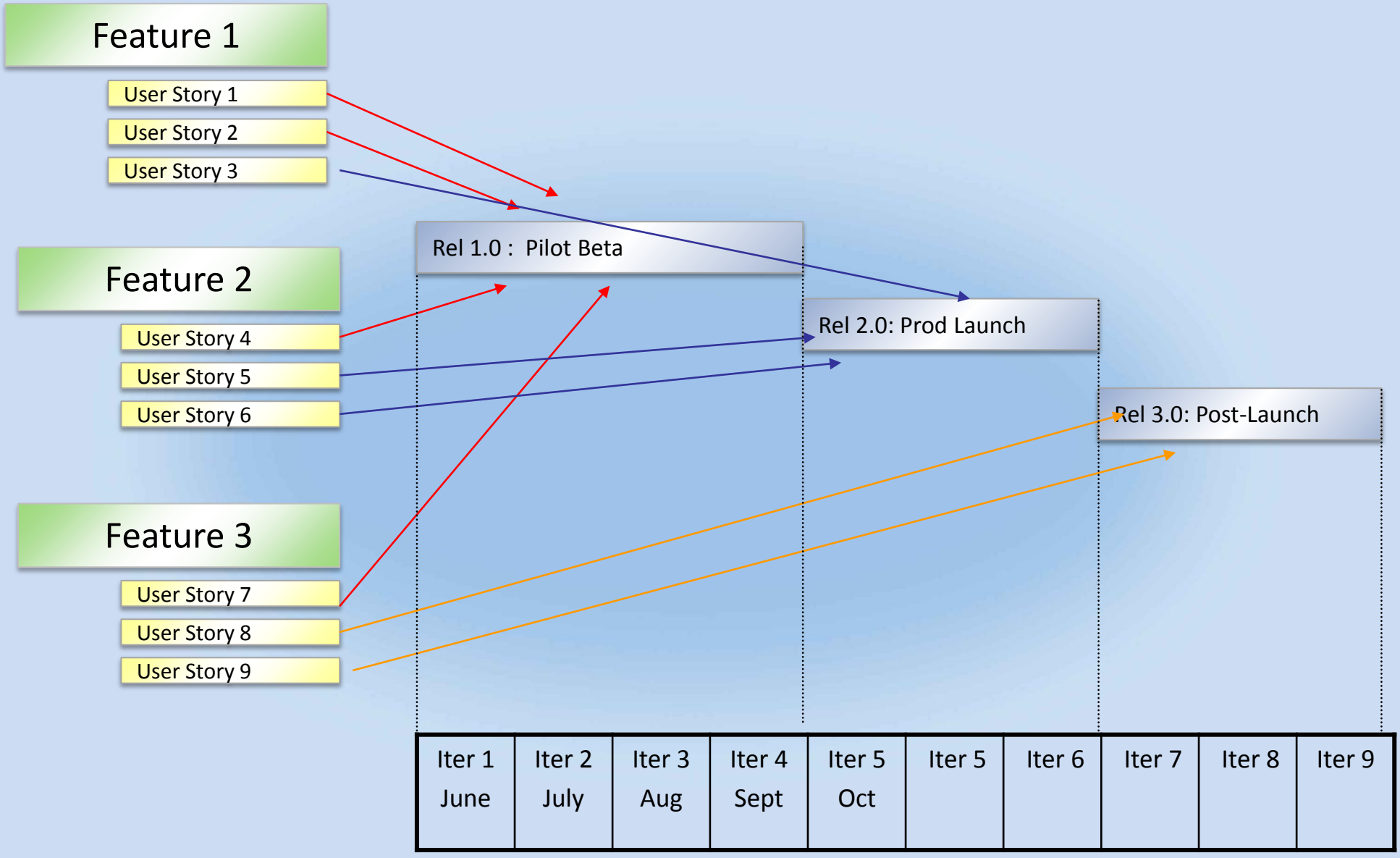
Write the story cards for this project. At least 5 cards. Be prepared for one member to read these to the class.

Remember what you have learned:

Hint – what makes a story complete?

Knowledge United
Leverage the Power of Learning

Agile Planning – Mapping to Timeline

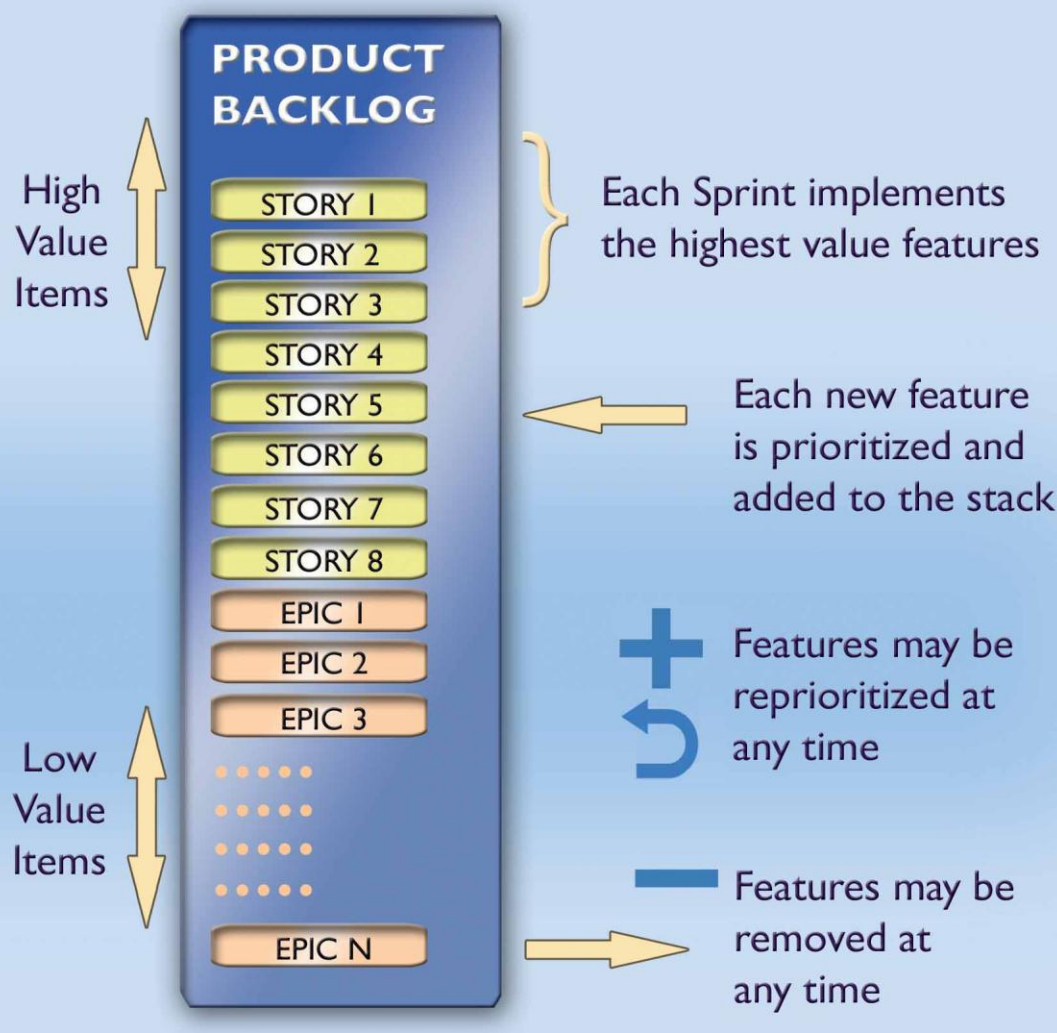


Scrum

The Backlog

The Product Backlog

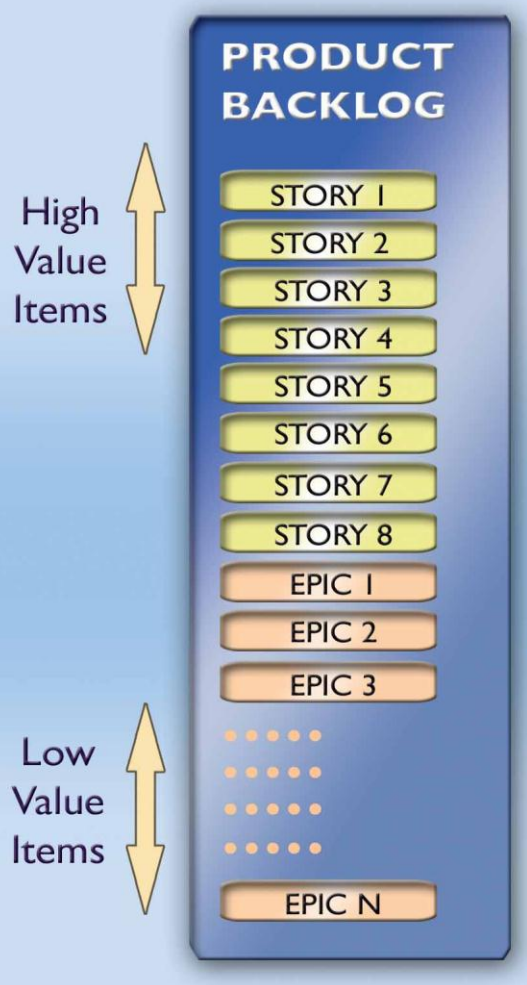
- Product Owners own a backlog of product items and maintain this backlog in an ongoing manner
- The Product Backlog consists of System features
- Product Owners will continually groom and prioritize the list
- New features can be added at any time
- Existing items can be modified and withdrawn as needed
- Features are estimated in both Business Value and Story Points



"The lower the priority, the less detail." *Schwaber and Beedle 2002, 33*

The Product Backlog Prioritization

- Each Product Backlog Item has some business value
- Business Value is used to prioritize backlog items
- Assessing business value can be complicated in that it could consist of aggregating one's belief on the following:
  - Revenue it could drive
  - Demand from financial institutions
  - Demand from prospects
  - Importance to end users
  - Frequency of use
  - Risk of not having the feature
  - Building of core capabilities
  - Alignment to strategy



This can be a complex, subjective and time consuming activity!

Simplified Backlog

User Story	Business Value	Story Points	Effort (IDD)
US #1	8	4	
Task #1:			1
Task #2:			3
Task #3:			2
US #2	4	8	
Task #1:			4
Task #2:			6
Task #3:			2

Knowledge United
Leverage the Power of Learning

Example Backlog - Rally

Backlog NEW

Page Tools

+ Add New

Show: User Stories Defects Defect Suites Columns

	Rank	ID	Name	Plan Esti...	Priority	Owner
				137.5 P	All	All
	1	US1236	As a PM, I need the Rewards Summary page to time-out with the IB experience.	2.00	--	Jeremy
	1.4	US1756	(PuR) As a PM, I want A/B testing capabilities. Design 3	5.00	--	Jeremy
	1.5	US1755	(PuR) As a PM, I want A/B testing capabilities. Design 5	5.00	--	Jeremy
	2	US468	As an end user, I want a print view that lists all currently pending bill payments with confirmation numbers and transfers,	3.00	--	Matt
	3.1	US483	Make opening and closing sections in the Account area simultaneous		--	Matt
	4	US1216	As a PM, I need to update the user experience to be relevant for users whose FIs choose points based rewards		--	Jeremy
	4	US1733	As IFS, when SDP loads I want to display the shell of the PR first while the content loads		--	Jeremy
	5	US1735	As a PM, if the widget does not get an expected response from OPS with in expected time frame, I would like to display alternative content.	2.00	--	Brian
	8	US613	Epic: Outside accounts widget (Account Aggregation)	0.00	--	
	8.1	US611	Epic: Re-design transfer widget as collapsible tool at top of account bar	0.00	--	Jessica
	11	US1892	Upgrade to latest SDP to the latest WHP version.		--	Srini
	11	US3280	Technical Story: as OPS I want to match up messages in sdp.log and access.log		--	Sergey
	11.5	US2035	As a financial institution, I want to mask any login/user values on the page		--	Tom
	12.16	US772	As an end user, I want to be able to export to CSV and OFX formats	8.00	--	Brian
	14	US273	As an end user, I want to be able to change the date range for the View My Spending section	1.00	--	Brett
	15	US1241	As an end user, I want TOB pages to resize themselves depending on a user's screen width	5.00	--	Tom
	16	US688	As a PM, I want the ability to test different zero state messages	13.00	--	Matt
	20	US1734	As operations, I want to move static resources to Apache	1.00	--	Brian
	20.5	US2101	As an end user, I want to have a fourth sorting critieria (account number)	1.00	--	Sunil
	22	US1908	As an end user, I want to specify the default account that displays first in the account bar when I log in,	5.00	--	Justin
	36	US2002	As a PM, I would like to present a new pop-up window explaining the program to users on their first visit to the Purchase Rewards summary page.	2.00	--	Jeremy
	37	US2000	As an end user, I would like to more easily distinguish online only Purchase Rewards offers from others.	1.00	--	Jeremy
	38	US1996	As an end user, I would like to be able to use my debit card to redeem rewards from any of my PR eligible accounts.	13.00	--	Jeremy
	39	US2034	Global Out out. As an end user, i would like to out of all my accounts by one action.	3.00	--	Jeremy

Scrum

Estimating Stories

Estimating Using "Story Points"

- A method of estimation made popular by Mike Cohn in "Agile Estimation and Planning".
- User stories are estimated using a relative measure of effort and size known as **story points**.
- Duration, if requested, can be determined later during Sprint Planning and at the task level.
- Story Point Estimation is done using a scale such as :
  - Fibonacci series – 0,1,1, 2, 3, 5, 8, 13, 21, 34, 55
  - Lean Approach – 1, 2, 4, 8
- After a few sprints it will become clear how many story points a team can typically do in a sprint; this is called their **VELOCITY**, which is a measure of their **CAPACITY** for work during a sprint

Story Points are a coarse-grained, relative measure of effort and size, Pichler, 2010, 64.

Knowledge United
Leverage the Power of Learning

Story Points and Priority

Considerations used in estimating story points:

- Complexity
- Size
- Risk

Considerations used in estimating priority:

- Business Value
- Complexity
- Risk
- Dependency
- Cost

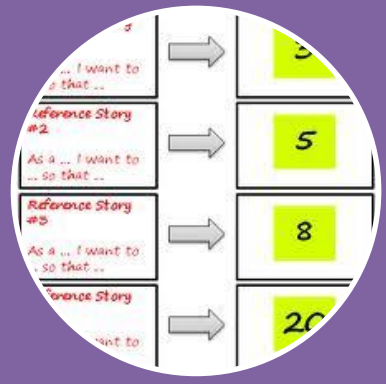
Business Value vs. Priority

When can priority be higher then business value?

- Non-functional requirements
  - i.e. logging
- Morale
  - i.e. drop down vs. free form

Knowledge United
Leverage the Power of Learning

Exercise



Using your movie stories, assign story points to at least 4 stories.

- Hint - Which stories involve the highest effort?

Knowledge United
Leverage the Power of Learning

Story exercise

A customer has decided to take your team to the movies to reward you for great work. The Product Owner shares the customer vision for this project. Each of you will enjoy the overall movie-going experience.

This release's goal is to get you and your team to the movies on time. What actions will take place from the time of decision to the time you actually sit in the theater and the movie begins?

Write the story cards for this release. What are the top 3? Be prepared for one member to read these to the class.

Remember what you have learned:

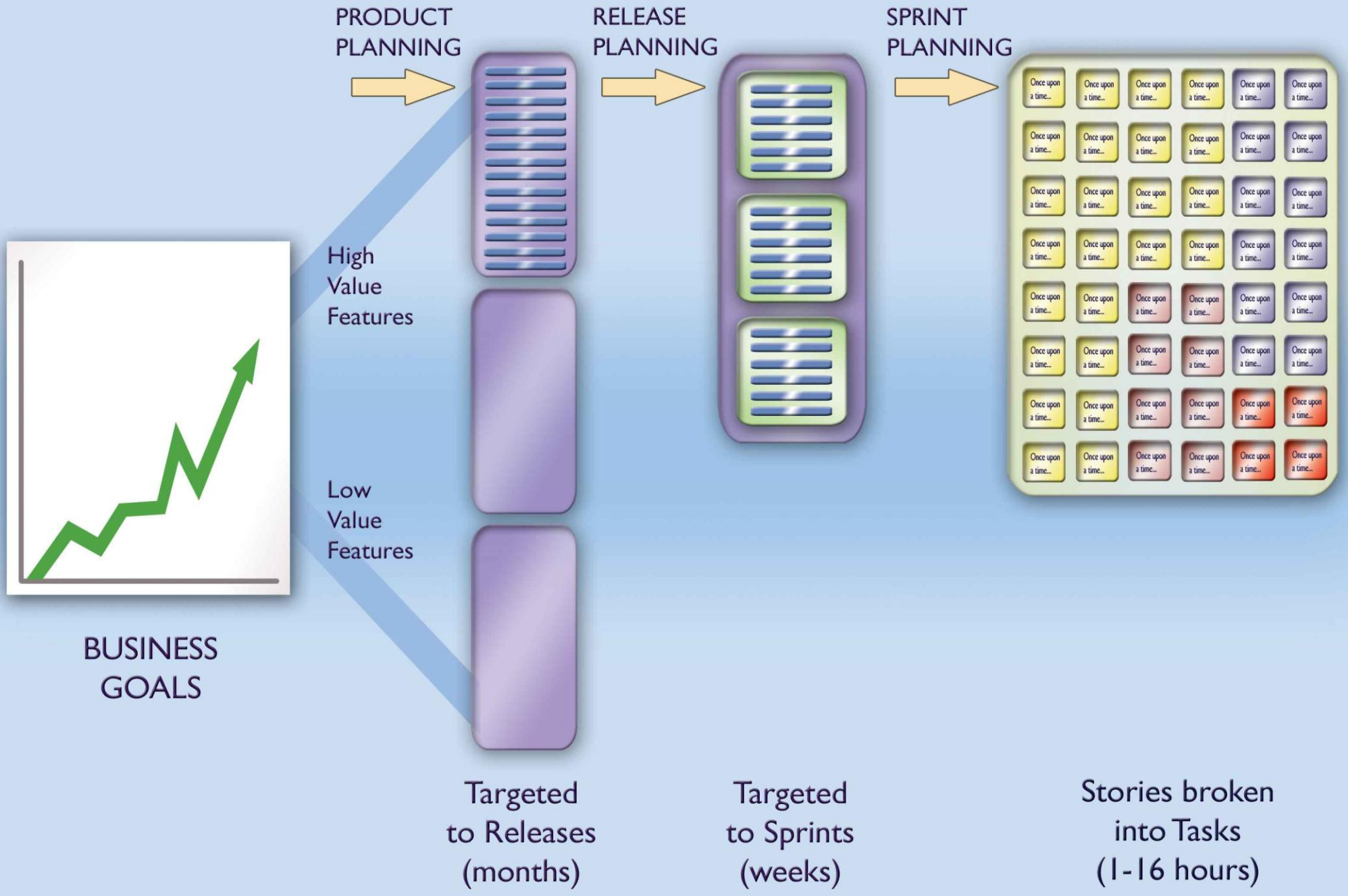
Hint – what makes a story complete?

Knowledge United
Leverage the Power of Learning

Scrum

Sprint Planning

Commit to Stories and Create Tactical Plan



Iteration 0 and Hardening Sprint

Sprint 0	Sprint 1	Sprint 2	Sprint 3	Sprint 4
				
Release Backlog Definition Of Done Prioritized Stack Relative Sizing Estimates Architectural Spike Learning Events Customer Interaction Plan Infrastructure Setup Sprint 1 Stories	• Implement Sprint 1 • Sp 2 Use Case Docs	• Implement Sprint 2 • Sp 3 Use Case Docs	• Implement Sprint 3 • Sp 4 Use Case Docs	• Hardening Sprint

Most Intuit teams prefer a 3-week sprint!

Task Level Example

Business Value (BV) is used by the Product Owner to determine which stories have the most value to the business. These stories are prioritized first. BV is assigned before Prioritization.

Story Points are being used by the Scrum Team members to determine the level of effort and achieve a high level understanding of effort for planning purposes.
Ideal hours vs. Man hours are utilized by the Scrum Team at the Task Level to determine effort level in more detail

Four (4) ideal hours a day to every 8 man hours is being used in the below example to calculate task length

Example:

Man hours	Ideal Hours	Story Card A includes 2 tasks	Total Ideal Hours for Story Card A	Total Man Hours
8	4	Task 1 = 5 hours Task 2 = 3 hours	8	16

Notes: *The four hours may change over time*
Ideal Days – actual effort time (uninterrupted)
Man days – days in an iteration
Elapsed Time – The actual amount of time it takes to complete a story.

Knowledge United
Leverage the Power of Learning

Rally: Agile Lifecycle Management Tool

- **What is Rally?**
  - Rally is the software tool Intuit uses to manage Agile projects:
    - Store and manage product backlogs
    - Store and manage user stories & associated tasks
    - Plan releases & iterations
    - Track release, team & iteration status
    - Provide standard & custom reports/charts
    - Much, much more
- **Reference:** Find more information on Rally, recorded training sessions and internal team support contacts here:
  - http://wikis.intuit.com/agile/index.php/Tools

Exercise

Plan the first sprint for this release

Task out your movie stories at least 3.

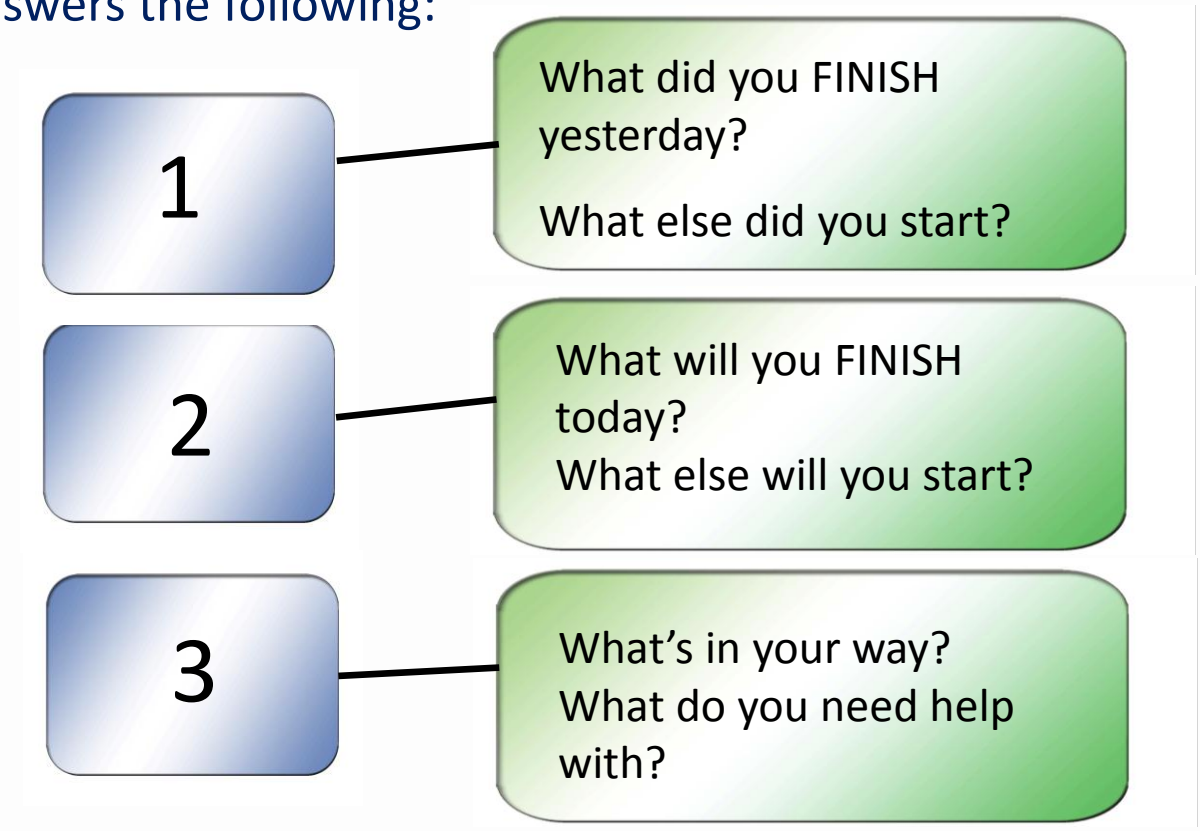
Hint : Tasking is done by the team, not the product owner.

Knowledge United
Leverage the Power of Learning

Daily Sprint Standup

Daily Scrum / Standup Meeting

Each participant answers the following:



Meetings are for the core team members; extended team and management are encouraged to attend for visibility, openness, and information:

- Held Daily at the same time
- Target length: 15-minutes
- Not for problem solving
- Help each other succeed – do whatever it takes to move the project forward

Knowledge United
Leverage the Power of Learning

Exercise – Daily stand up

Lets do a Stand-Up, *virtual style!*

Everyone stand up where you are

Keeping the class in mind, answer the three daily scrum questions. Use this morning's class to represent yesterday and this afternoon's class to represent today. Then mention any impediments preventing you from continuing this afternoon (i.e. I have to leave for a 15 minute meeting at 4pm or I have no impediments).

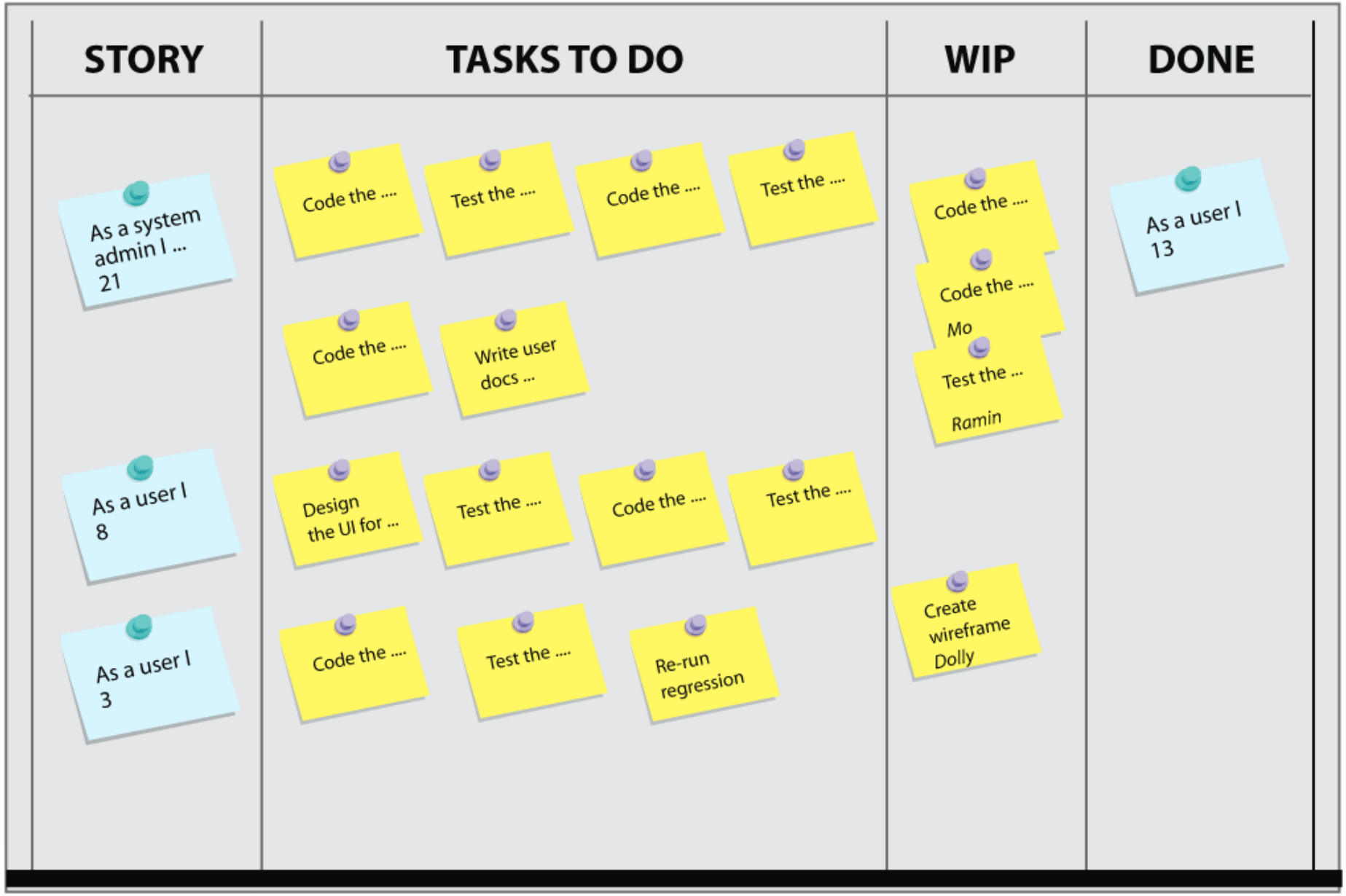
I will *call your name* when it's your turn

Knowledge United
Leverage the Power of Learning

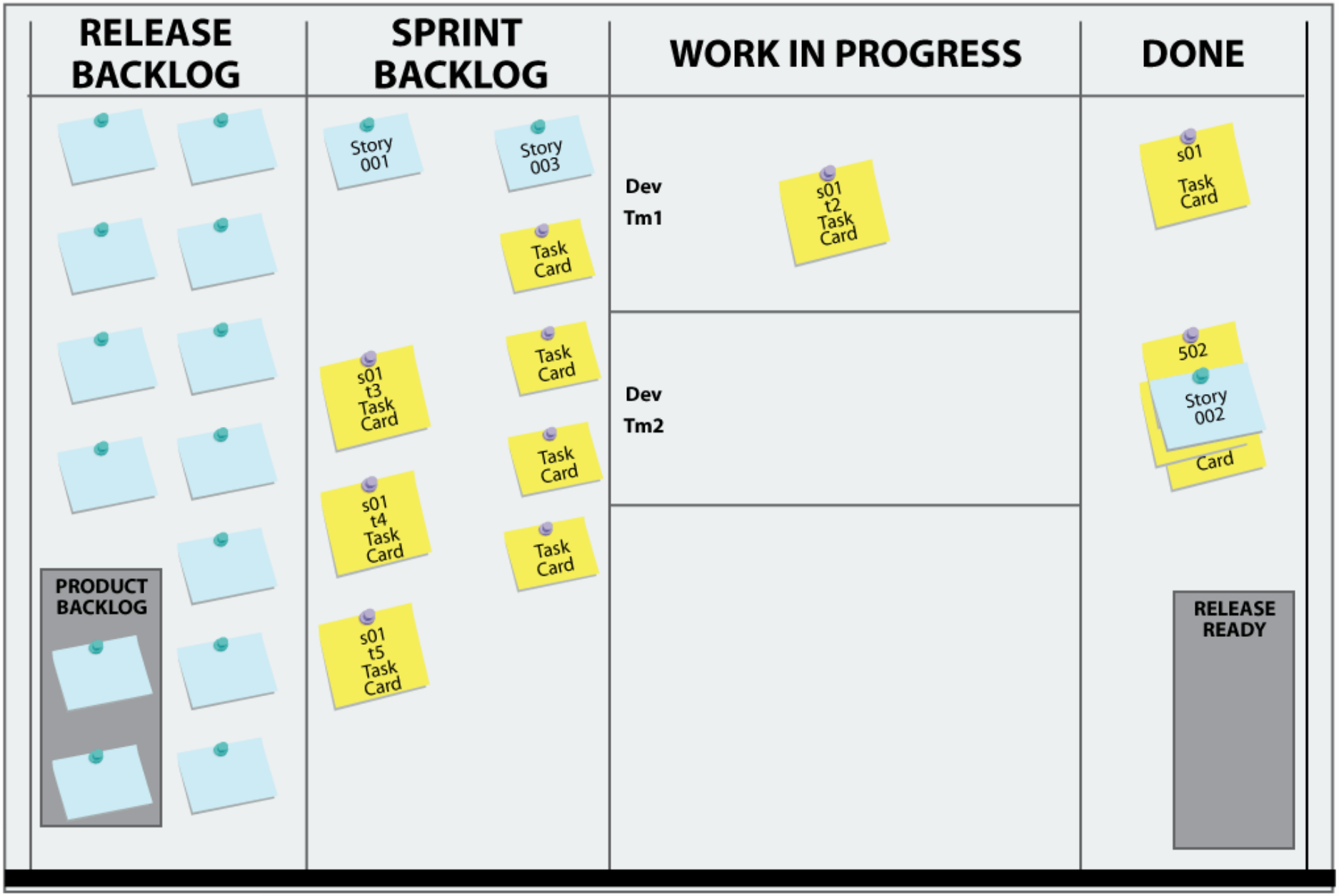
Scrum

The Board

Example 1 - Team Task Board



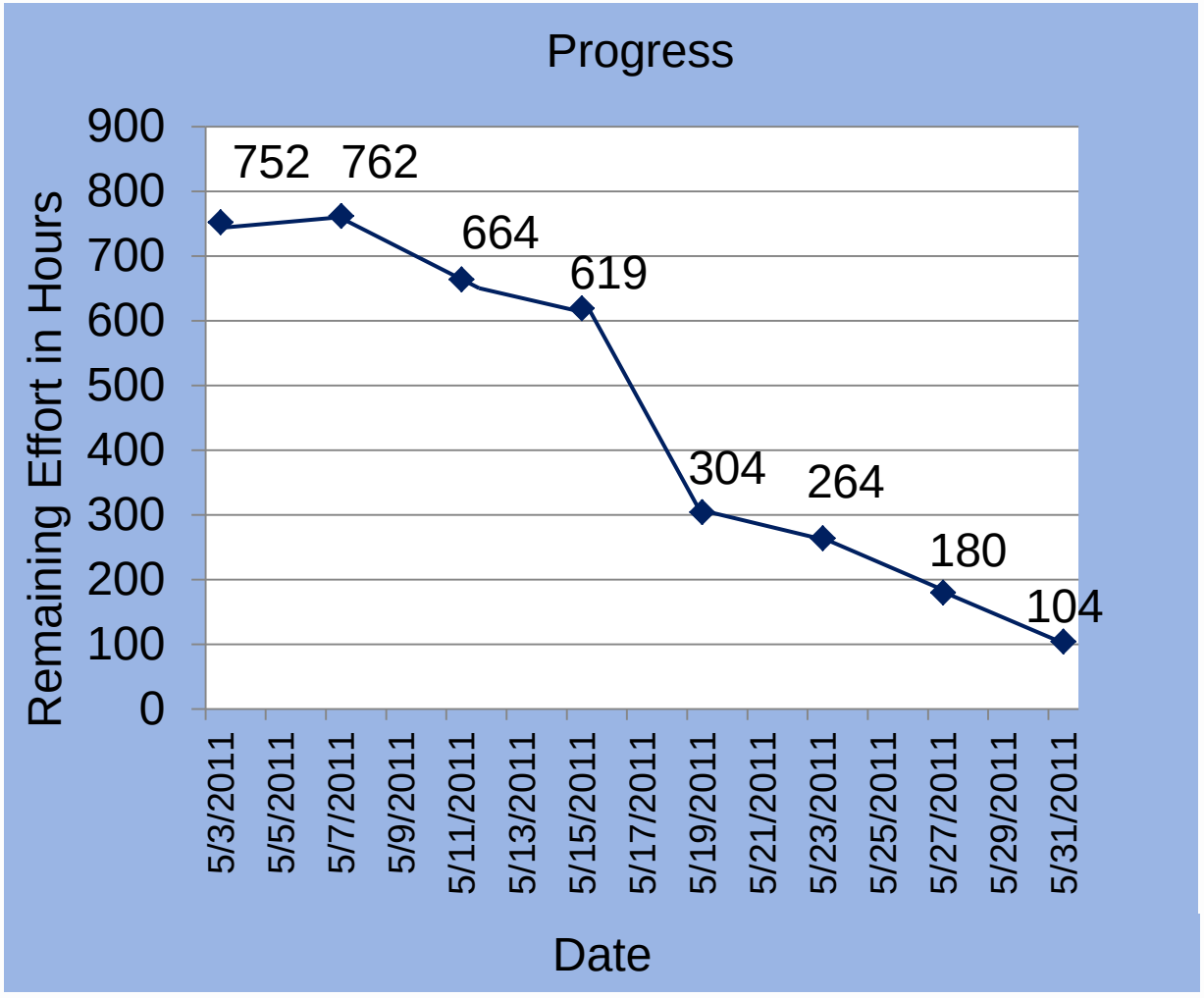
Example 2 – Team Task Board



Burn Charts

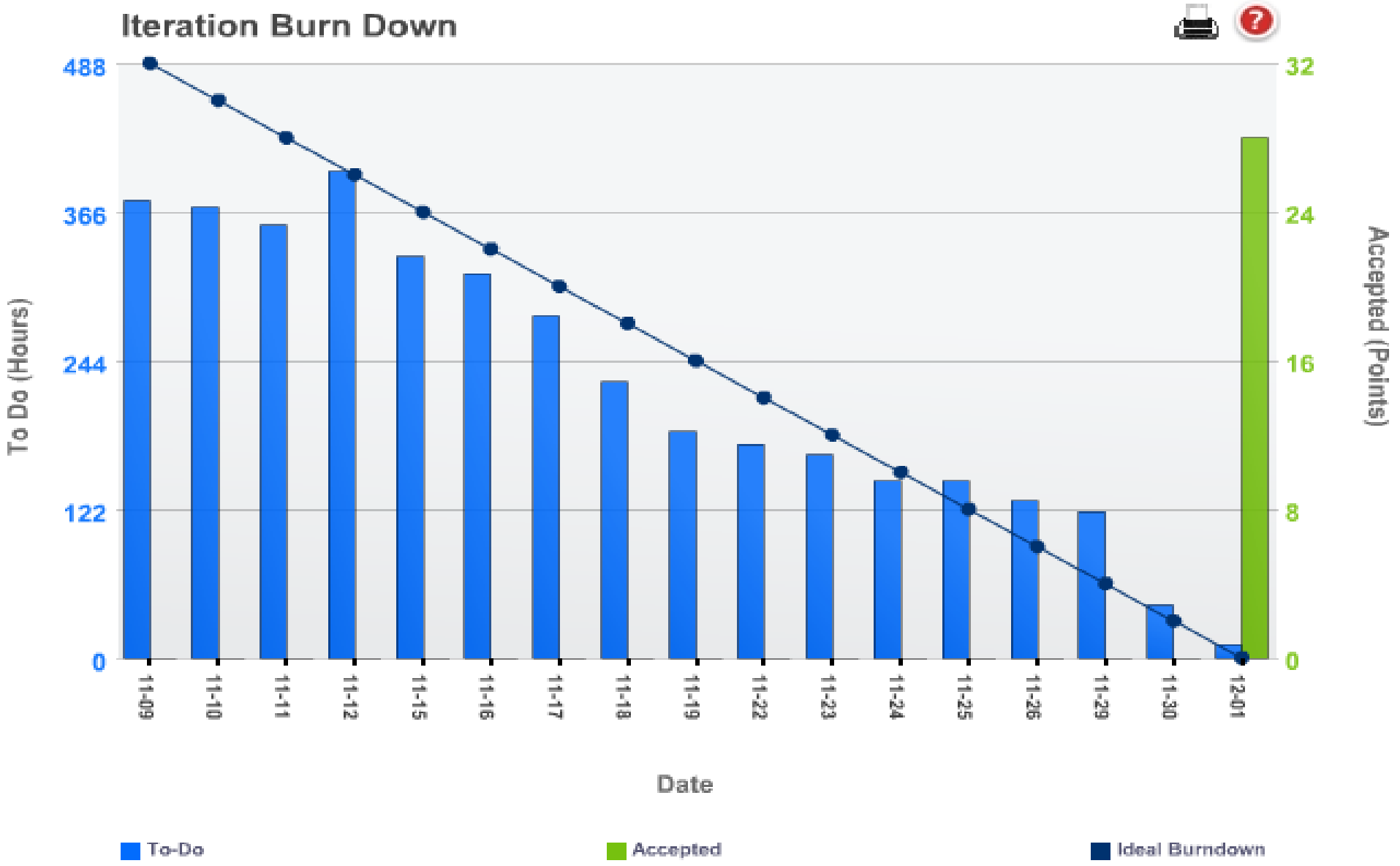
Example 1 - Sprint Burndown Chart

- On Day 1 of the sprint (after Sprint Planning is done), plot the total number of hours for all tasks
- Each day of the sprint, plot the hours REMAINING for the work left in the sprint
- This will give you a Burndown graph



Knowledge United
Leverage the Power of Learning

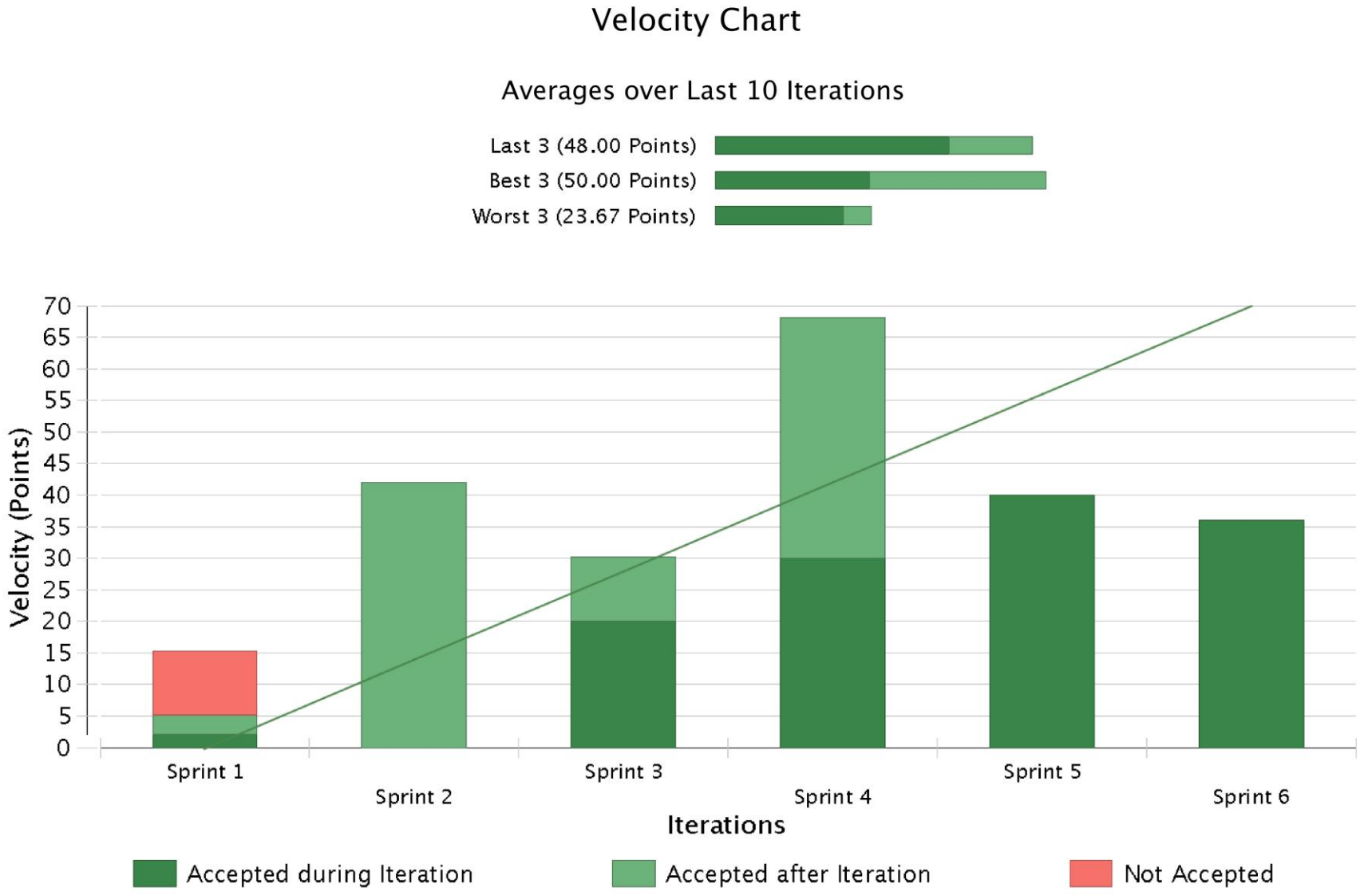
Example Sprint Burn down - Rally



Knowledge United
Leverage the Power of Learning

Intuit Confidential

Velocity Chart



Key Components for Blending

Thoughts to Remember in Blending

Keep it simple

Being Agile is the ability to move quickly and respond to change

- This is a core principal in blended as well

Build the internal and external capabilities

Academic vs. Practical

- Training is not enough, you actually need experienced coaching ongoing journey
- Always refer back to the principles for alignment
- Employ changes based on lessons learned, inspect and adapt

Knowledge United
Leverage the Power of Learning

Strategies to Avoid Conflicts

Strategies employed to resolve conflicts and reduce related risks

Plan Collaboratively and Use an Incremental Life Cycle

- Employ an incremental life-cycle model to help in aligning the Agile team's work within the overall project schedule. This is fundamental and crucial to bridging the old and new.
- Results in earlier communication concerning priorities and risks

Knowledge United
Leverage the Power of Learning

Keep Communication Lines Open

- Documentation
- Presentation
- Education
- Establish common vocabulary
- Project
  - verbal status
  - examples of the outcomes
  - visual boards
  - virtual boards
  - retrospectives

Background/Problem Description

- This Fortune 500 company was looking to become more effective by improving on the integration points of multiple systems and improving the functionality of a trading system.
- Time-to- market was imperative and budget was limited. Since there were so many tentacles and integration points from other departments and third party companies, this project also was restricted by several regulations.
- The company was ingrained with older methods of project management and software development.

Real-Life Blended Case

Method

- The client chose to employ a Blended Agile methodology that would improve delivery time, provide working software that satisfied the need of the customers, and conserve budget dollars.

Approach

- Held discovery sessions to determine readiness to use a Blended Agile methodology
- Formed a Blended Agile team
- Created the setup/iteration zero for the blended Agile team.
- Analyzed which Agile practices would be feasible to utilize within their blended methodology now and in the future, while continuing to introduce new practices.
- Determined which existing processes and best practices to utilize

Knowledge United
Leverage the Power of Learning

Real-Life Blended Case

- Program Development
  - Governance approach
- IS Senior Management Updates weekly
- IS Steering Committee Update
- Agile Competency Development
  - Training – overviews, lunch and learns, team meetings
  - Agile techniques training – stories, backlog, planning, vision box, etc.
  - Extensive training/coaching of the slated Agile coaches
  - Mentoring – Coaches, Team, Managers

Knowledge United
Leverage the Power of Learning

Real-Life Blended Case

- Co-location was sporadic and only occurred during the core hours
- Resources were part time, not dedicated
- Instituted Core Hours
  - 9 hours a week
- Obtained executive buy-in and support for the project and a blended Agile methodology
- Established acceptance criteria for project success
- Physical Environment
  - Room which was shared with other projects
- Team
  - 2 Developers/analysts, 2 extended team developers, 2 Product Owners/QA, Agile Coach apprentice, Agile Coach

Knowledge United
Leverage the Power of Learning

Real-Life Blended Case

Key Results

- The organization began to view blended Agile as just a different technique for project development
- Future team coaches were developed through the process
- Teams became true teams and morale increased
- Earlier releases into production were enabled giving the customer faster workable software
  - Although only 9 hours a week, when the first release was implemented the management/team calculated the actual time form start to finish to be about two weeks.
    - Delivering in two weeks workable software was phenomenal for this company.
    - Knowledge of Agile and forming blended practices occurred simultaneously
    - Determining which practices to use occurred simultaneously
- Cross-functional knowledge was shared and then utilized for future projects
- 2 Teams Started
- Slated two more teams

Knowledge United
Leverage the Power of Learning

Lessons learned

- Establishing a blended Agile project is crucial to the success of the project; the set-up differs per project
- No two blended Agile projects are exactly alike from a practice perspective.
- Determine up front the right team members
- Establish an on boarding process including team introductions and a project boot camp and implement
- Consider how to use team members time wisely
- Research should be done up front

Lessons Learned

- Each organization is different
- Any combination of best practices is good
- Consistently evaluate and adapt
- Learn your existing process first
- The amount of time dedicated is ok
- Co-location is helpful but not necessary
- Communicate the same message
- Make sure all organizational levels are on the same page
- Learn Agile principals first, then begin to blend
- Consider academic but be practical

Distributed Agile

A Few Terms:

- **Distributed** is a research and development project that is done across many business worksites or locations. It is a form of R&D where the project members may not see each other face to face, but they are all working collaboratively toward the outcome of the project (Wikipedia)
- **Remote** means located far away; distant in space., *Computer Science* Located at a distance from another computer that is accessible by cables or other communications links (free dictionary)
- **Off Shore** is the relocation by a company of a business process from one country to another country (Wikipedia)
- **On Shore** is outsourcing of work within the same country
- **Outsourcing** is when an organization subcontracts work to an outside vendor
- **Virtual Team** (also known as a geographically dispersed team or distributed team) is a group of individuals who work across time, space and organizational boundaries with links strengthened by webs of communication technology. (Wikipedia)

A Few Reasons:

- 24 hour development to reduce time to market
- Larger talent pool of experts
- Expertise in Global Markets
- Choice of additional expertise/view points
- Reduce Cost

- Modification of Agile Processes
  - Customer Collaboration
  - Communication
  - Coordination
  - Feedback cycle
  - Workload distribution
  - Collective ownership
  - Meeting formants
  - Role adaptation
- Cultural Differences

Lessons Learned from Distributed Agile Projects

Never, never assume anything; be clear and specific in your communication. Repeat back your understanding of what is being communicated.

Spend time up front in the planning; do not try to skimp on this, you will pay for it

Mirror all the functions of the team (i.e. meetings and even a party)

Use tools such as wikis to hold common information

Shorter iterations such as a 2 week timeframe are better to avoid issues and challenges

Have a lead in each location and a back up lead

Distribute work by stories, not by functionality or feature components. This will help build the team

Use multiple channels for communication all the time

Background/Problem Description

- Financial Services client looking to become more competitive by developing an in-house backend customer service/invoicing system. The existing system could not handle the types of customer service/invoice related issues that had arisen since the growth of the company and growth in various products and programs. Not all management was on board due to other priorities. Also CMMI was initiated by other divisions and since there were so many tentacles into integration points from other areas this project also was required to be CMMI compliant.
- To stay competitive the client needed flexibility in handling customers' challenges, concerns, comments and their invoice requests. Time to market was imperative and budget was limited. Management buy-in and support for the project was needed from all of the integration departments.
- The company was fairly new to Agile with 2 Agile projects under their belt including one distributed Agile project.

Distributed Case

Method

- Employ a methodology that would improve the delivery time, provide working software that satisfied the need of the customers and conserve budget dollars. Distributed Blended Agile was chosen.

Approach

- Discovery sessions held
- An Agile distributed team was formed
- The setup/iteration zero created for a distributed Agile team
- Blended Agile with their existing process
- Considered lessons learned form the previous distributed Agile project employed these lessons where feasible.
- Considered what worked on the last and their first distributed Agile project and analyzed which of these practices would be feasible to utilize again while introducing some new practices.
- Coached the team and management in distributed Agile practices

Knowledge United
Leverage the Power of Learning

Distributed Case

- Iterative development
  - Daily Stand Ups
  - Retrospective
  - Customer demos
  - Iterative planning
  - Product Planning
  - Release planning
  - Burn down charts
  - Velocity tracking
  - Coached the team in distributed communication.
  - Coached the team in distributed teamwork
  - Cross-pollination of team members
- Obtained executive buy-in and support for the project and distributed Agile

Knowledge United
Leverage the Power of Learning

Distributed Case

- Established acceptance criteria
- Improved on existing reporting
- Partnered with the business to improve the process by looking beyond the initial requirements
- Improved on the distributed QA process
- Improved on distributed pair programming.
- Utilized continuous integration and TTD practices
- Distributed Agile mapped back to CMMI
- Seeded the new distributed Agile team with a few members from the first distributed Agile project.
- Set up agile distributed support team
- **Key Results**
  - New system delivered which exceeded initial customer expectations
  - Increased customer satisfaction
  - A distributed Agile project with many integration points can be extremely successful.

Knowledge United
Leverage the Power of Learning

Distributed Case

- Future team coaches developed
- Teams became true teams
- Morale increased
- Additional projects were targeted for distributed Agile practices
- An interest developed to create their own distributed areas
- Earlier releases into production/faster workable software
- Cross-functional knowledge shared and then utilized for future projects
- Successful turnover and knowledge dump for future system support
- Employing previous lessons learned from a different project was initially difficult for the team
- Architecture reviews were implemented
- New way of supporting existing software was implemented.

Knowledge United
Leverage the Power of Learning

Lessons learned:

- Establish an on-boarding process including team introductions and a project boot camp and implementation
- Always be aware of the cultures so that local customs and celebrations do not blind-side you
- Stay on top of the news for the countries you are working with so that if there is an issue of any kind (i.e. political, natural disaster etc.) you are aware and can act accordingly
- Put a good technical infrastructure in place, I can not stress enough the importance of automation
- Do not assume the Agile team knows how to collaborate across the globe
- Involve off site folks in client meetings
- Have different environments for testing and development but shared by all teams, not environments per team

Key Observations

- The organization began to view distributed Agile as just a different technique for Agile vs. a challenge and a huge wall to get around. This opened a new option for not just development but for support.
- The second distributed Agile project looked to the first for advice and assistance.
- Distributed Agile coaches began to rise to the top.
- As the project continued CMMI requirements decreased since delivery of customer value continued.
- Establishing the set up of distributed is crucial to the success of the project; the set up differs per project.
- No two distributed Agile projects are exactly alike from a practice perspective.
- Cross-pollination helped foster trust and an understanding of each location.
- Continuous Integration was a clear advantage which helped to insure builds are quick so that issues can be immediately addressed.

Resources & References

Agile Resources

- Agile Software Development with Scrum by Ken Schwaber

  http://www.amazon.com/Agile-Software-Development-Scrum/dp/0130676349/ref=sr_1_1?ie=UTF8&s=books&qid=1280433284&sr=8-1

- Succeeding with Agile: Software Development Using Scrum by Mike Cohn

  http://www.amazon.com/gp/product/0321579364/ref=s9_intb_gw_t2?pf_rd_m=ATVPDKIKX0DER&pf_rd_s=center-2&pf_rd_r=0WCM8XQW2168NGV3JQTE&pf_rd_t=101&pf_rd_p=470938631&pf_rd_i=507846

- Agile Manifesto & Principles

- www.agilemanifesto.org
  www.agilemanifesto.org\principles.html

- Agile Alliance

  http://www.agilealliance.org/

Knowledge United
Leverage the Power of Learning

Questions

Knowledge United
Leverage the Power of Learning

Appendix

Scrum Glossary

- **Agile** the name coined for the wider set of ideas that Scrum falls within; the Agile values and principles are captured in the Agile Manifesto
- **Chicken** (arch.) term for anyone not on the team, the term offended some people so is now rarely used, cf. Pig
- **Daily Scrum** a fifteen-minute daily team meeting to share progress, report impediments and make commitments
- **Done** also referred to as "Done" or "Done Done", this term is used to describe a product increment that is considered releasable; it means that all design, coding, testing and documentation have been completed and the increment is fully integrated into the system
- **Emergence** the principle that the best designs, and the best ways of working come about over time through doing the work, rather than being defined in advance, cf. Empiricism, Self Organization
- **Empiricism** the principle of "inspect and adapt" which allows teams or individuals to try something out and learn from the experience by conscious reflection and change, cf. Emergence, Self Organization
- **Epic** a very large user story that is eventually broken down into smaller stories; epics are often used as placeholders for new ideas that have not been thought out fully. There's nothing wrong with having an epic, as long as it is not high priority
- **Estimation** the process of agreeing on a size measurement for the stories in a product backlog. Done by the team, usually using Planning Poker
- **Impediment** anything that prevents the team from meeting their potential (e.g. chairs are uncomfortable). If organizational, it is the Scrum Master's responsibility to eliminate it. If it is internal to the team, then they themselves should do away with it
- **Impediment Backlog** a visible list of impediments in a priority order according to how seriously they are blocking the team from productivity

Scrum Glossary

- **Pig** (arch.) term for a team member, the term offended some people so is now rarely used
- **Planning Poker** a game used to apply estimates to stories; it uses the Delphi method of arriving at consensus
- **Product Backlog** a prioritized list of stories that are waiting to be worked on
- **Product Backlog Item** any item that is one the backlog list, which will include user stories, epics and possibly technical stories to deal with technical debt, etc.
- **Product Owner** person whom holds the vision for the product and is responsible for maintaining, prioritizing and updating the product backlog
- **Release Burndown Chart** a visible chart to show progress towards a release
- **Retrospective** a session where the Team and Scrum Master reflect on the process and make commitments to improve
- **Scrum Master** a servant leader to the team, responsible for removing impediments and making sure the process runs smoothly so the team can be as productive as possible
- **Scrum Meetings** Story Time, Planning, Review, Retrospective, Daily Scrum
- **Scrum Roles** there are only three: product owner, Scrum Master, team
- **Spike** a short, time-boxed piece of research, usually technical, on a single story that is intended to provide just enough information that the team can estimate the size of the story
- **Sprint** a time boxed iteration
- **Sprint Burndown** a visible chart that indicates on a daily basis the amount of work remaining in the sprint
- **Sprint Goal** aka Sprint Theme, the key focus of the work for a single sprint
- **Sprint Planning** a meeting between the Team and the Product Owner to plan the sprint and arrive at an agreement on the commitment
- **Sprint Task** a single small item of work that helps one particular story reach completion

Scrum Glossary

- **Stakeholder** anyone external to the team with an interest in the product being developed
- **Story** a backlog item usually using the template form: as a [user] I want [function] so that [business value]
- **Story Point** a unit of measurement applied to the size of a story, cf. Fibonacci Sequence
- **Story Time** the regular work session where items on the backlog are discussed, refined and estimated and the backlog is trimmed and prioritized
- **Task** see Sprint Task
- **Task List** the tasks needed to complete the set of stories committed to a sprint
- **Taskboard** a wall chart with cards and sticky notes that represent all the work of a team in a given sprint; the task notes are moved across the board to show progress
- **Team** the development team, responsible committing to work, delivering and driving the product forward from a tactical perspective
- **Team Member** any member of the team, including developers, testers, designers, writers, graphic artists, database admins...
- **Timeboxing** setting a duration for every activity and having it last exactly that (i.e. neither meetings nor sprint are ever lengthened - ever)
- **Velocity** the rate at which a team completes work, usually measured in story points.
- **Vision Statement** a high-level description of a product which includes who it is for, why it is necessary and what differentiates it from similar products
- **XP Practices** the set of development practices, including pair-programming, test-first, or test-driven development (TDD) and continuous refactoring, which are drawn from the XP methodology; many Scrum teams find these practices greatly improve productivity and team morale