



Agile Scrum Introduction

Mike Li

Email: Mike.Li@autodesk.com lidingshan@gmail.com
MSN: lidingshan@msn.com
Blog: <http://www.cnblogs.com/relaxtintin/>

About me ...

- 1998 ~ 2005, @Baosight, PM
- 2005 ~ 2008, @Infosys China, SPM
- Now, @ACRD AEC, Project Manager

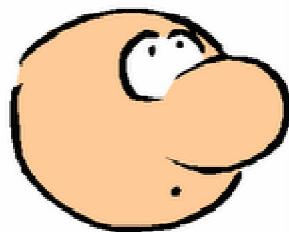


Training Target

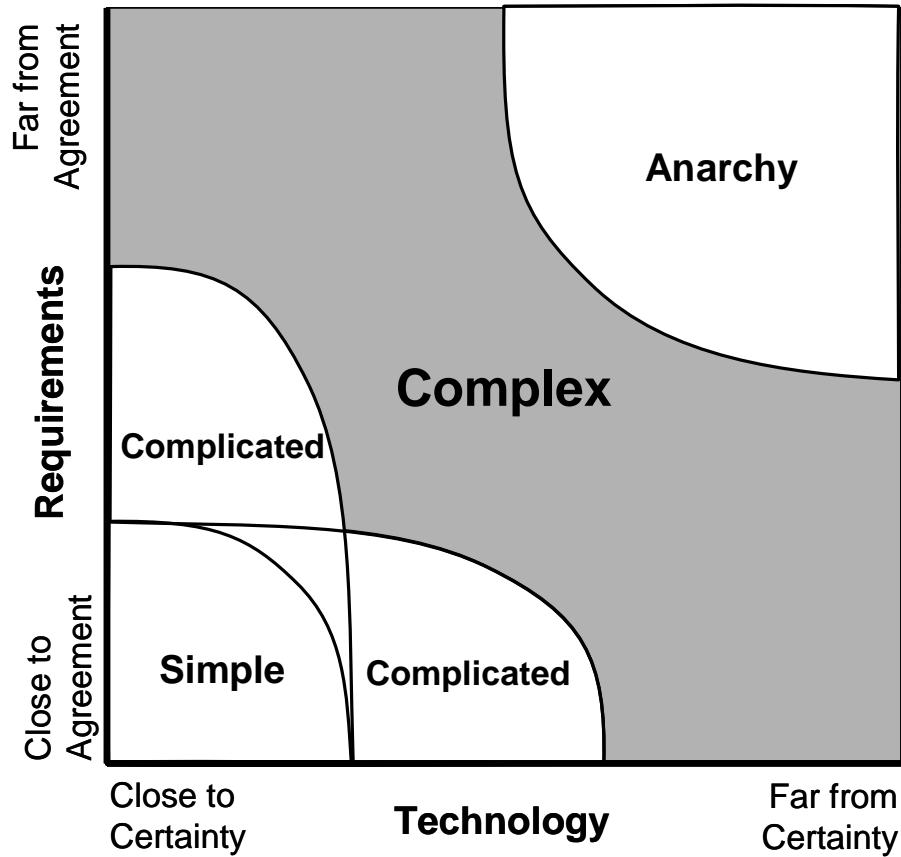
- Agile core value
- Scrum framework
- How to play
- How to start

What agile is

Software implementation is Simple



Complexity



Defined vs. Empirical

Defined Process

- Command & control
- Predict everything
- Force the execution of the plan
- Change management

Empirical Process

- Learning during execution
- Plan will change
- Welcome change
- Inspect & adapt

A process of
Knowledge Discovering

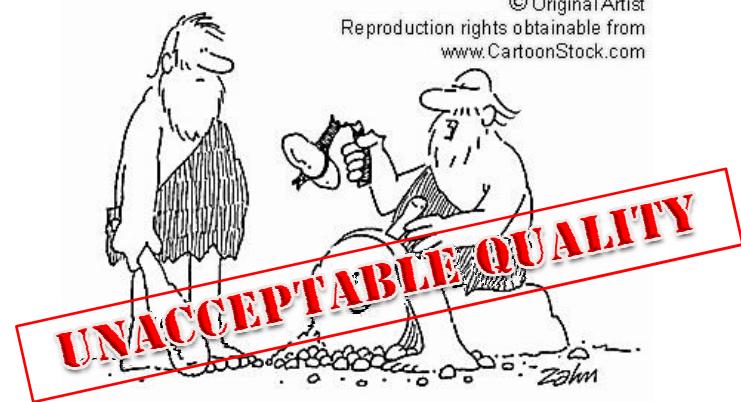
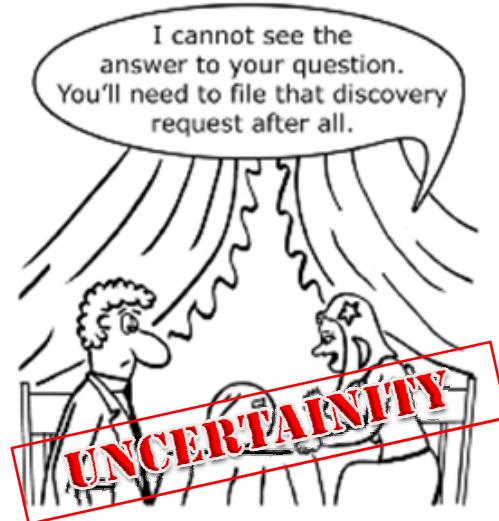
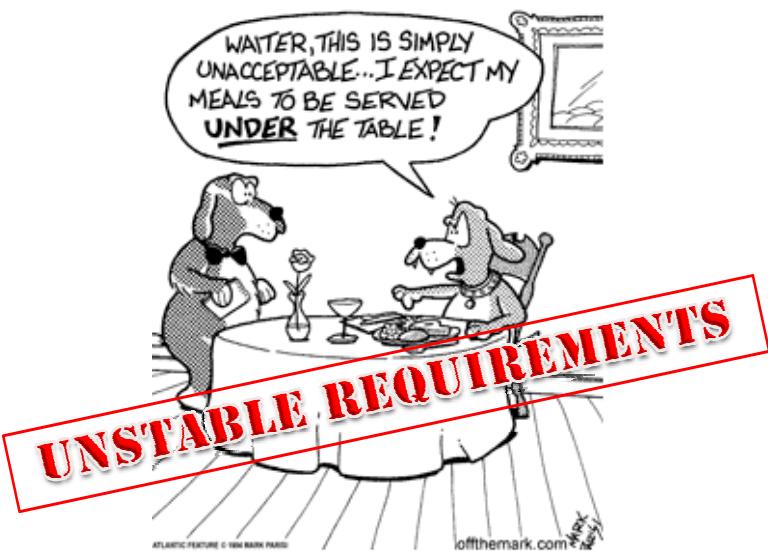
The Courage of Geese



"It is typical to adopt the defined (theoretical) modeling approach when the underlying mechanisms by which a process operates are reasonably well understood. When the process is too complicated for the defined approach, the empirical approach is the appropriate choice."

▪ *Process Dynamics, Modeling and Control*, Ogunnaike and Ray, Oxford University Press, 1992

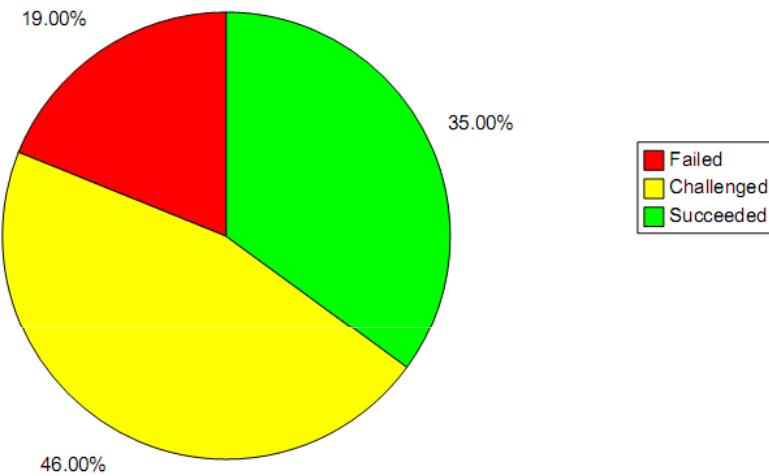
Challenges



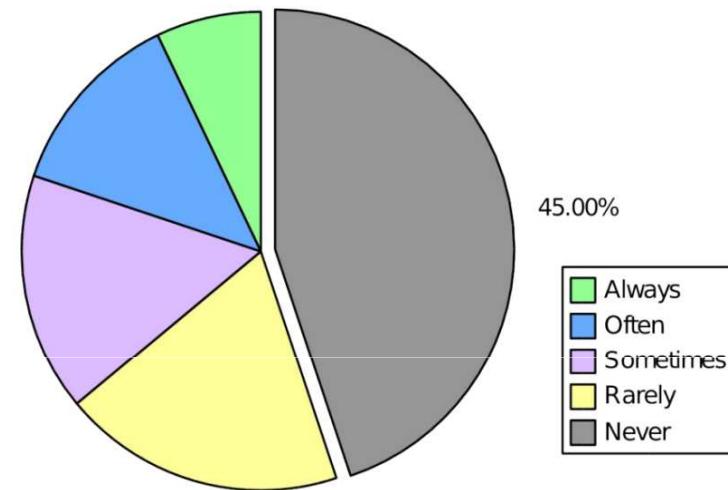
"They don't make them the way they used to."



IT Project Resolution (2006)



Features Used



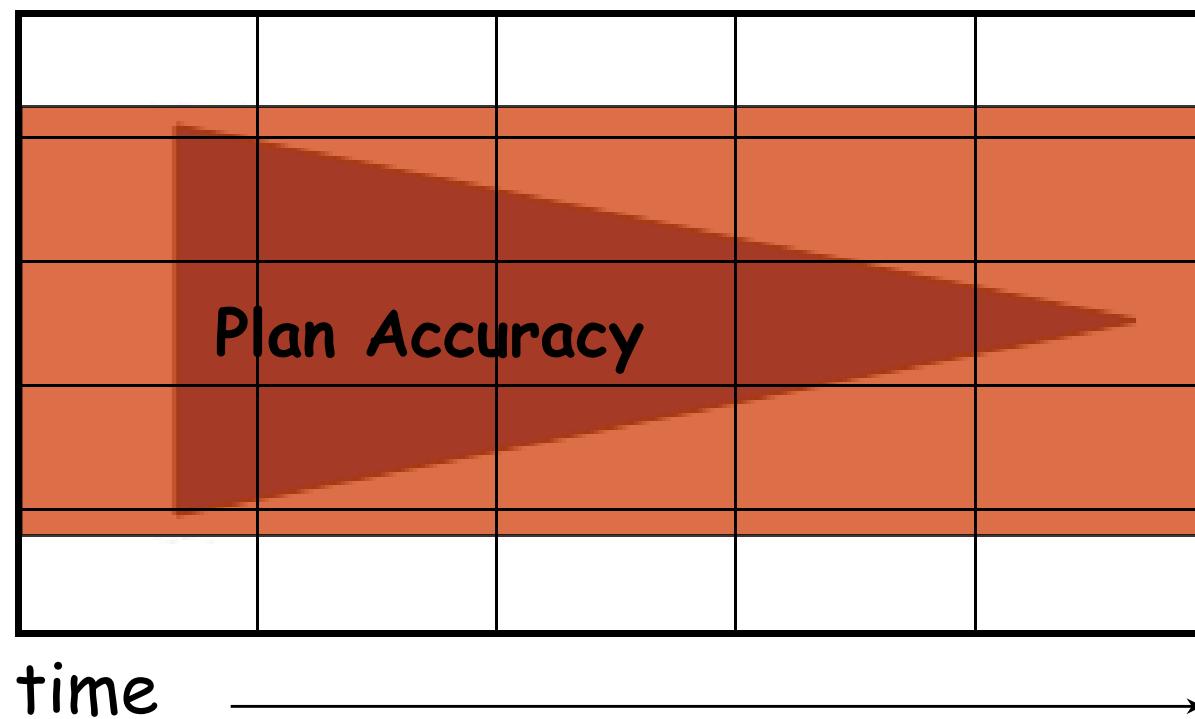
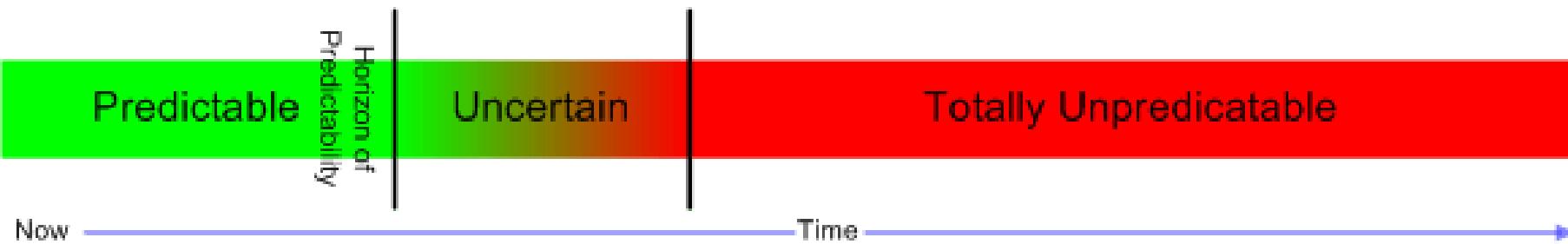
Based on the report of US DoD (United States Department of Defense) , **75%** software projects that developed with Waterfall model in early age were failed , some of those software products were never used, there were only **2%** software products could be used normally and not need to change a lot

Why so difficult

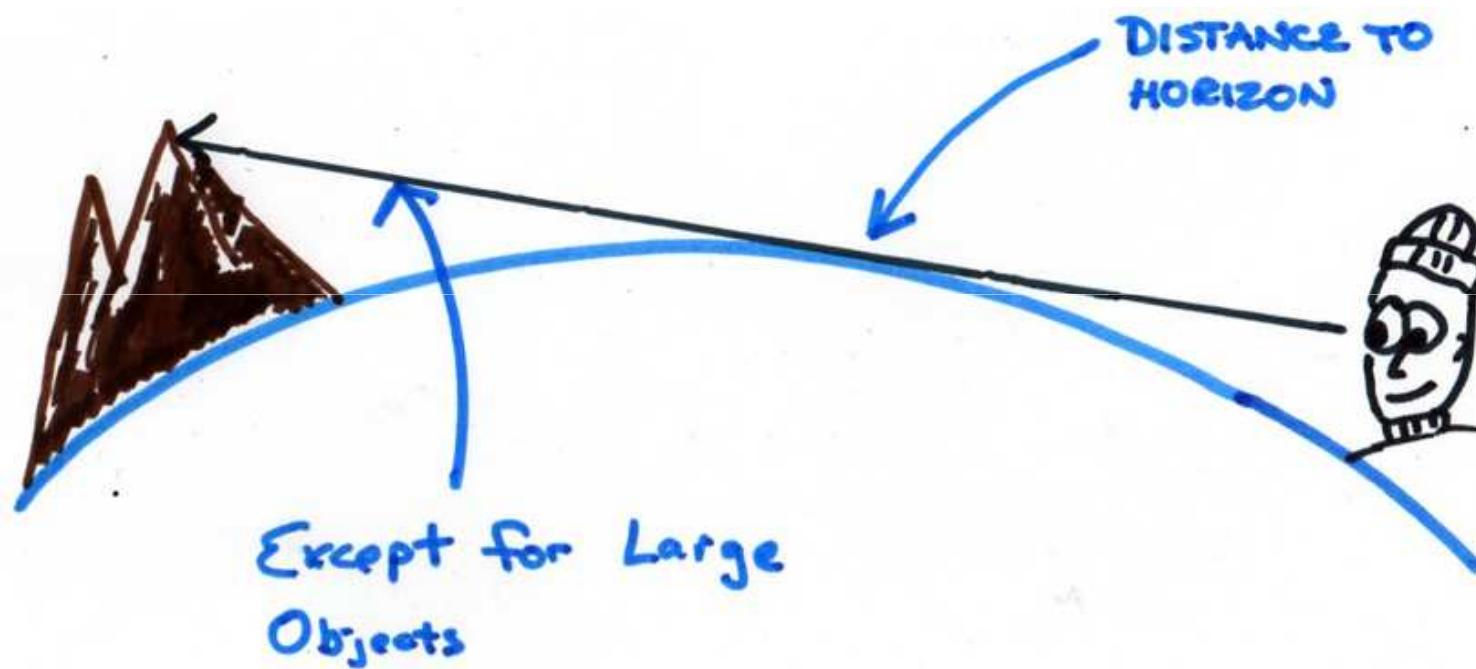


Things are keeping on changing...
...and, the more we predict for the future,
the more uncertainty we have to face...

Why so difficult



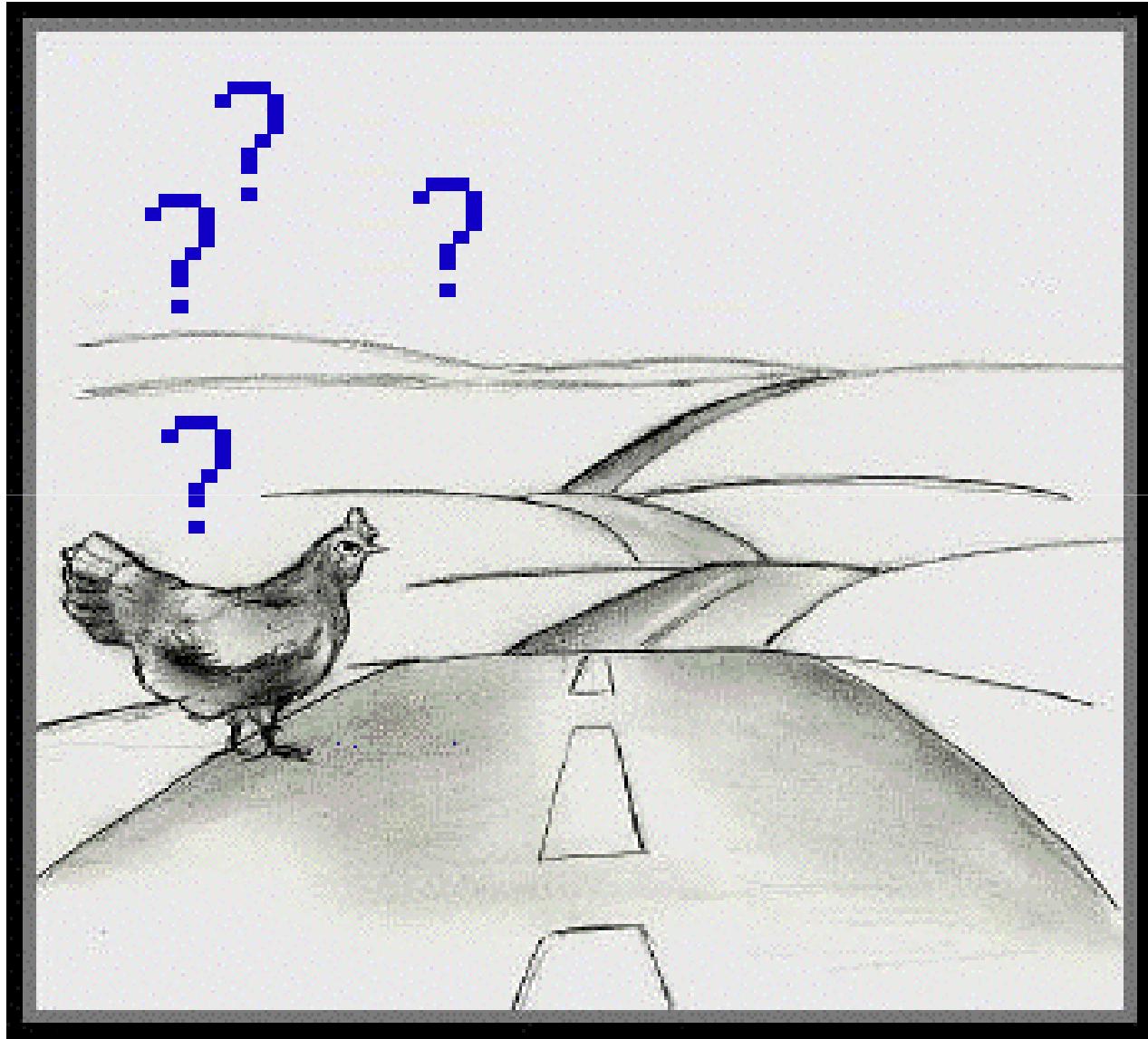
Why so difficult



Why so difficult



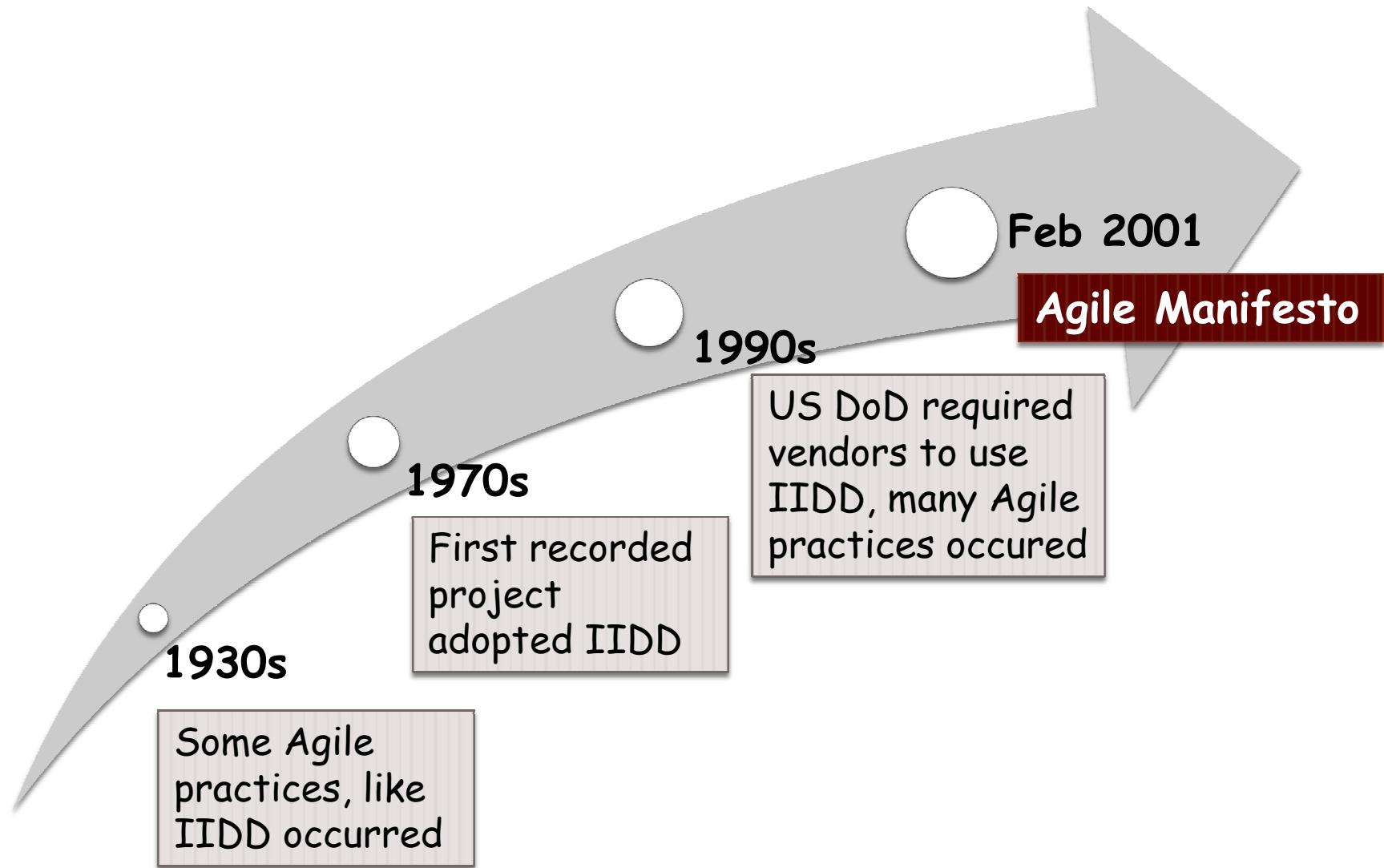
...because we do not know in advance how hard something might be or how long it will take to do with any degree of absolute certainty.



What Agile Is

**It is NOT a methodology,
process, or framework**

History



The Agile Manifesto

We are uncovering better ways of developing software by doing it and helping others do it.
Through this work we have come to value:

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

That is, while there is value in the items on the right, we **value** the items on the left more



Agile Principles

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

amount agility build come contract customer effective requirements team value working software processes work
continuous collaboration accordingly adjusts best architectures become better design documentation couple developing enhances comprehensive conveying customer's early follow pace left must others reflects projects throughout timescale simpler--the sustainable technical teams trust ways valuable
agile documentation couple developing enhances comprehensive conveying customer's early follow pace left must others reflects projects throughout timescale simpler--the sustainable technical teams trust ways valuable
done--is environment emerge harness get information job late maintain essential motivated frequently preference right satisfy sponsors support tools together weeks within
build art behavior collaboration accordingly agility
couple even give deliver good negotiation self-organizing welcome
individuals need primary people reflects projects throughout timescale simpler--the sustainable technical teams trust ways valuable
team value requirements self-organizing welcome
working welcome
software processes work

Agile Practices

- Scrum
- XP
- Crystal
- Lean
- FDD
- Agile Modeling
- etc.

Scrum framework

Origins

- Jeff Sutherland
 - Initial scrums: Easel Corp 1993
 - IDX - 500+ people doing Scrum
- Ken Schwaber
 - Scrum presented at OOPSLA '96 with Sutherland
 - Three books on Scrum
- Mike Beedle
 - Scrum patterns in PLOPD4
- Ken Schwaber and Mike Cohn
 - Co-founded Scrum Alliance in 2002

What is Scrum - Elevator Speech



Scrum is an agile project management framework that's an alternative to some traditional project management and product development techniques. It is simple, iterative, and incremental.

- Peter Borsella

Scrum in 100 Words



Mike Cohn

- Scrum is an agile process that allows us to focus on delivering the highest business value in the shortest time.
- It allows us to rapidly and repeatedly inspect actual working software (every one weeks to one month).
- The business sets the priorities. Teams self-organize to determine the best way to deliver the highest priority features.
- Every one weeks to a month anyone can see real working software and decide to release it as is or continue to enhance it for another sprint.

Scrum is Focusing on...



More and More Organizations are Adopting Scrum

Google

IBM

YAHOO!

ORACLE®

NOKIA

Microsoft®

SAP®

accenture

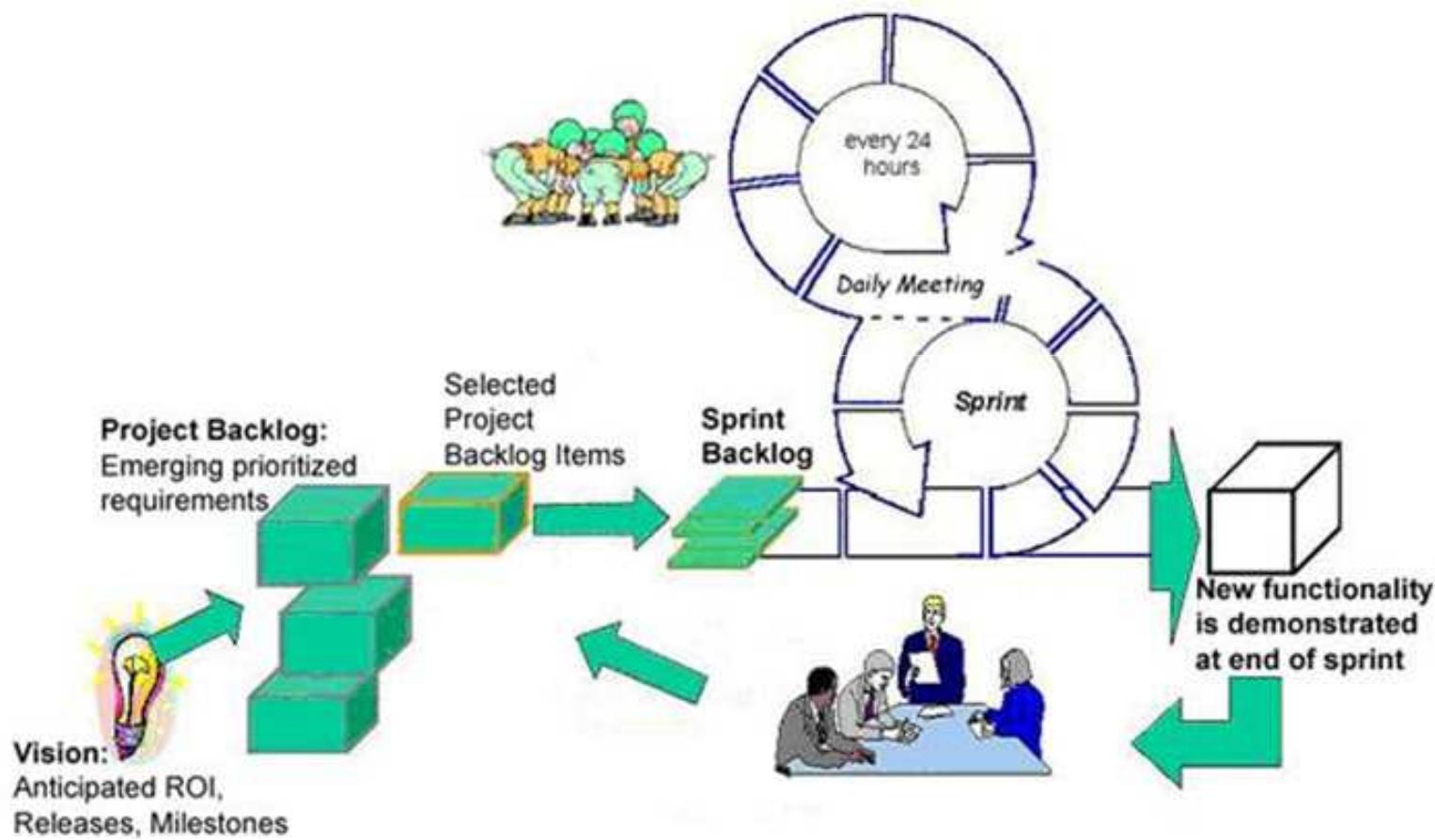
ERICSSON



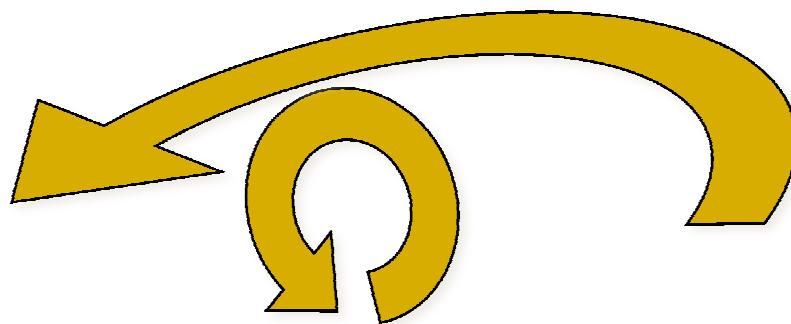
After Scrum Adoption...

- 36% improvements on average productivity
- 52% improvements on people moral
- 62% improvements on people responsibility
- 81% improvements on collaboration
- 44% improvements on quality

Big Picture of Scrum



Three Characters



Iterative Development

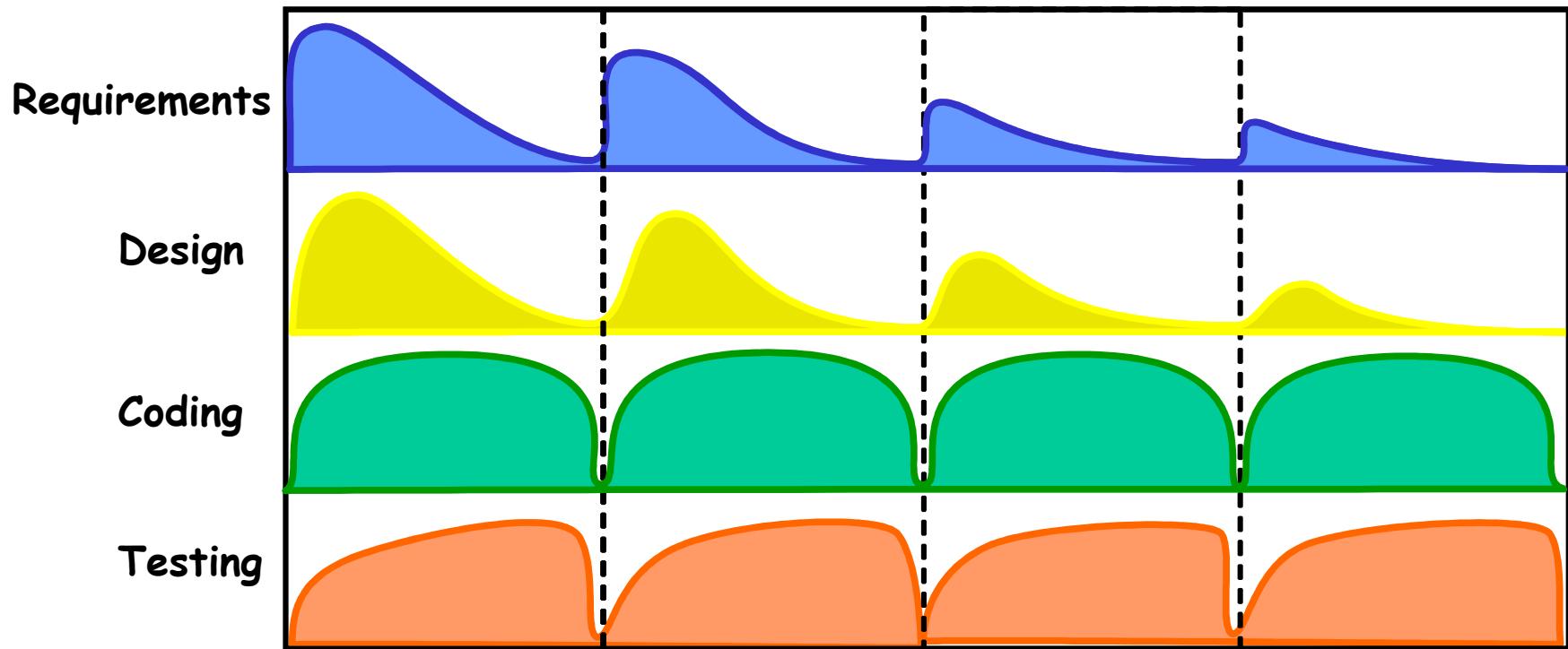


Thanks to Peter Borsella



Self Management

Iterative Development



Iterative Development - Cont.



aka, Sushi Delivery...
pace, rhythm, reliability.

Potential Shippable Product

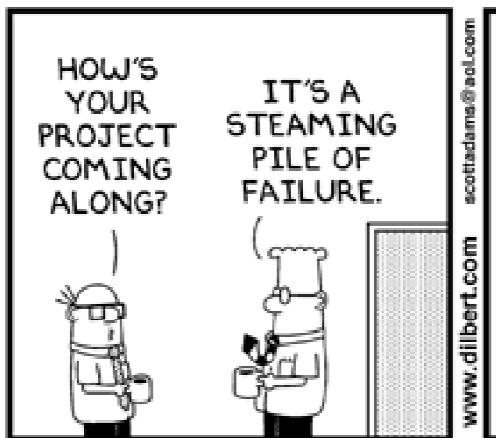
Self Management

- Pull instead of Push
- Transparent
- Empower people
- Cross-functional team
- Management through coach, feedback



Visibility

- Everything is open to everyone
- Expose problem early is good
- Tell the bad news at the first time is welcome



© Scott Adams, Inc./Dist. by UFS, Inc.

Who are in the game

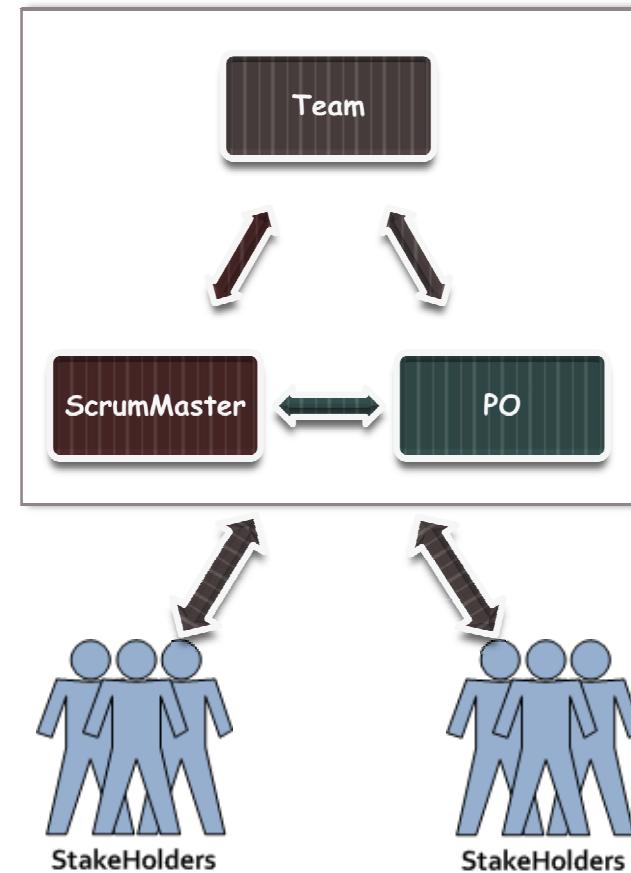
Pig & Chicken



- Pig - who will make commitment
- Chicken - who has interests

Roles in Scrum

- Product Owner
- Team
- ScrumMaster



Product Owner

- Customer or customer representative
- Decision maker for scope and schedule
- Communicate with users about the requirements
- Answer questions to team about requirements
- Help to set up high performance team

What

When

Acceptance
Criteria

Team

- Often include 5 ~ 9 members
- Cross-functional
- Setup sprint backlog with PO together
- Estimating
- Implementation
- Update sprint status
- Help PO to maintain product backlog
- Team management and improvement

How to Make

How to Improve

ScrumMaster

- Who has good understand about Agile & Scrum
- Guide PO and Team to practice Scrum
- Protect team
- Remove obstacles
- Help to setup and organize related meetings

If follow Agile principle

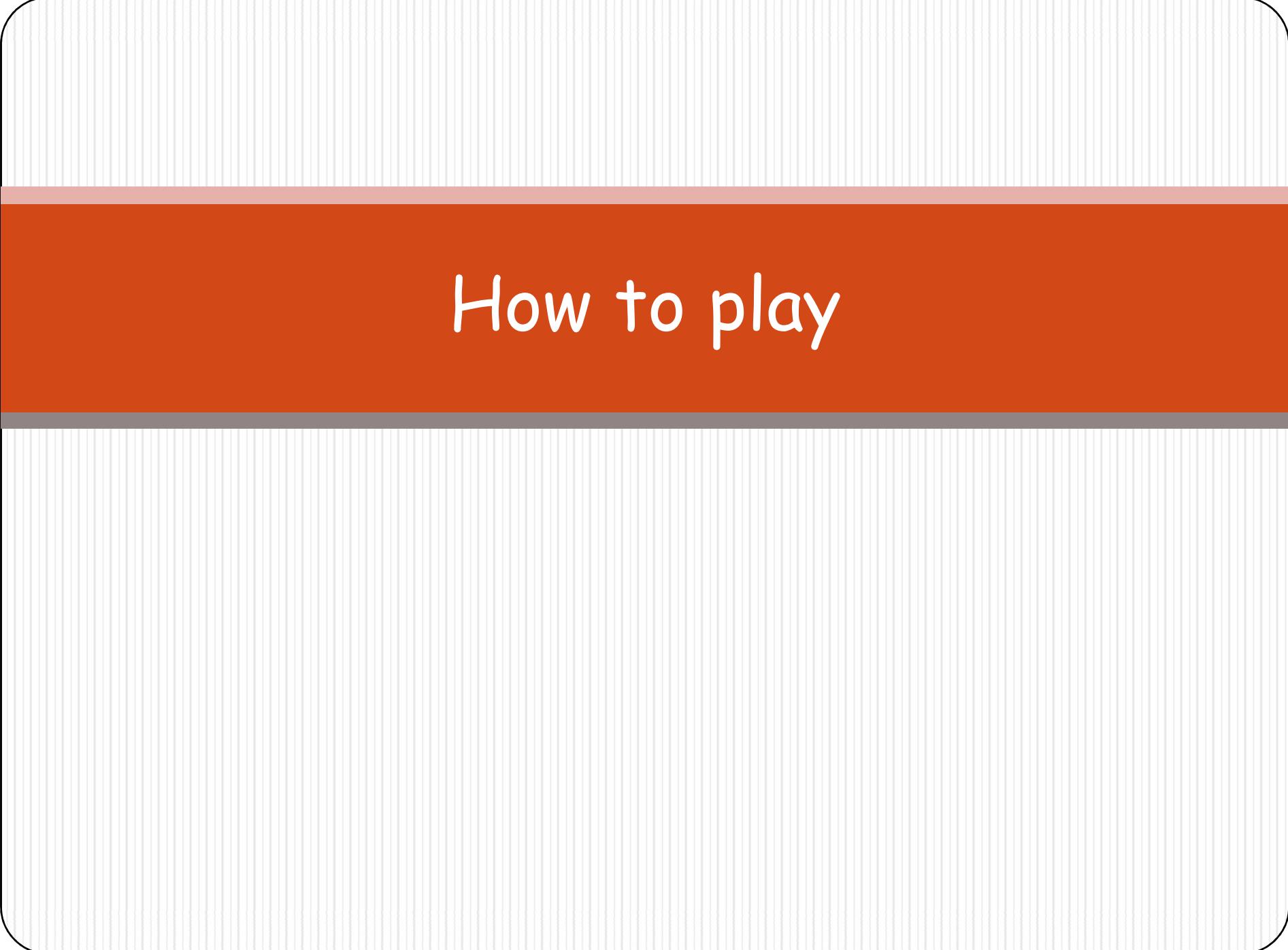
Any obstacles

Any helps needed

Questions

- Who is Pig? Who is chicken?
- Product owner
- ScrumMaster
- Developer
- Tester
- Department manager
- CTO
- Customer IT manager





How to play

Time Box



- All of the activities are in TimeBox
 - Help on team focus
 - Help to set up healthy pace of the team

User story

What it is

- Ron Jeffries Three Cs
 - Card
 - Stories are traditionally written on note cards
 - Cards may be annotated with estimates, notes and etc.
 - Conversation
 - Detail behind the story come out during the conversations with product owner
 - Confirmation
 - Acceptance tests confirm the story was coded correctly

As who, I want
what so that why.

Format

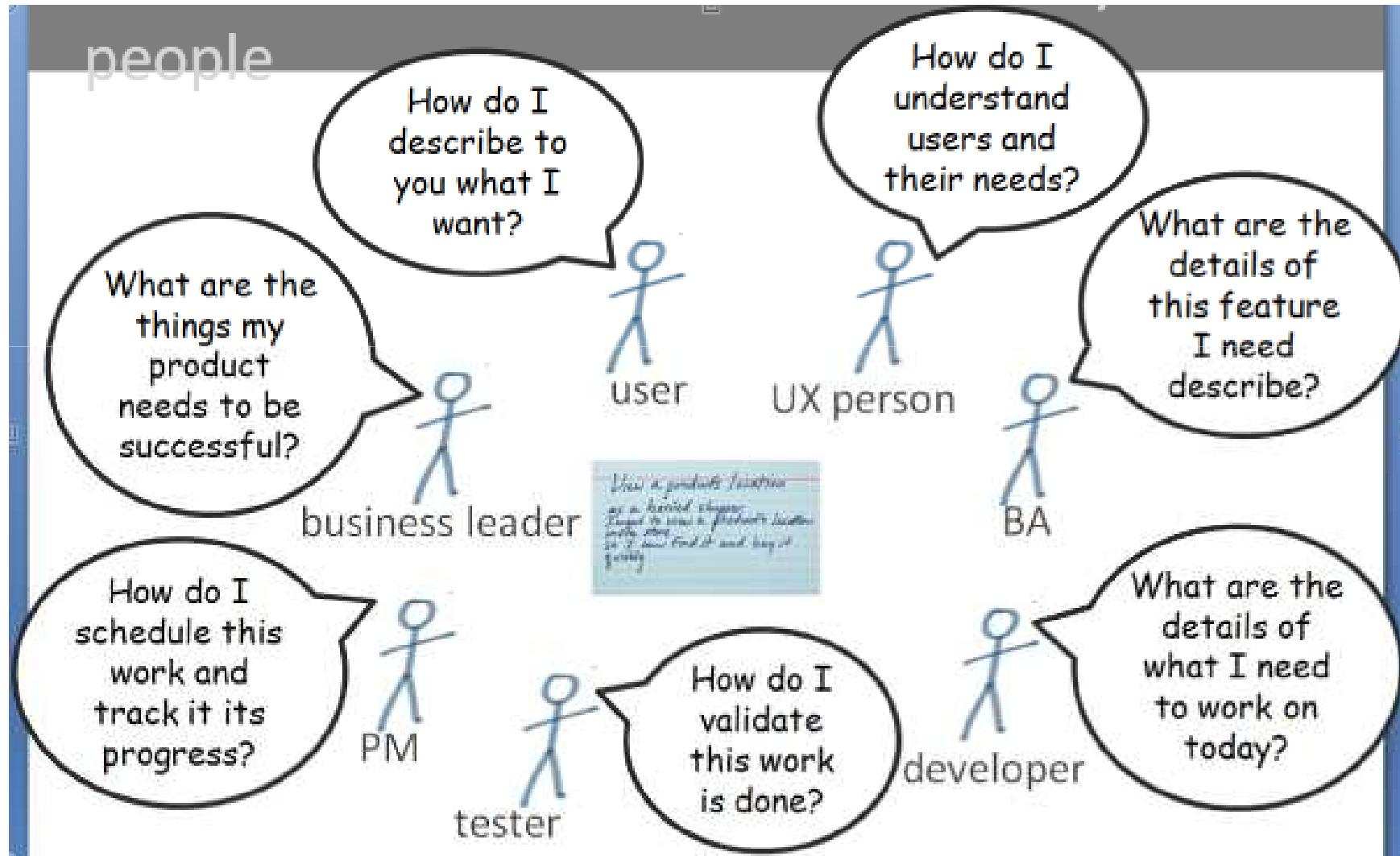
As { Who }, I want to { Do What }, so I can { Get Value }

As a purchaser , I want to be able to see other people's comments, so I can be suggested to buy or not.

Purpose

- User's need
- Product description
- Planning item
- Token for a conversation
- Mechanism for deferring conversation

Communication Boundary



INVEST - Good User Story

I - Independent

N - Negotiable

V - Valuable

E - Estimable

S - Small

T - Testable

Where is the detail

- As a purchaser, I can cancel my order
 - Can order be cancelled if it was ship?
 - Refund all or partial?
 - Refund to credit card or pay pal?
 - If it was ship, are there any shipment fee be charged?
 - Is it necessary to confirm with the member?
 - How to confirm?
 - Are there different policy for different members, like VIP?

Detail Information

- As acceptance criteria
 - Refund all for VIP members
 - Charge 5% cancellation fee for normal members
 - Normal members cannot cancel order if it was ship
 - Charge shipment fee if it was ship already
 - Send out confirmation mail to store owner
 - Send out confirmation mail to member

Detail Information - cont.

- Divide to smaller stories
 - As VIP, I can cancel order even it has been shipped
 - As normal member, I can cancel order before shipment
 - As store owner, I can receive confirmation mail for order cancellation

Product backlog

Product Backlog

- A list of user stories
- Sorted by priority
- Maintained by PO
- JIT scope
- Not specification

Issue Type	Key	Summary	Status	Priority	Story Points
User Story	SAD-1089	As a tester, I want to be able to set default configuration information of OS, build #, browser version and environment parameters for TC.	Closed	1. Blocker	3
User Story	SAD-1496	As an User, I can see the result of automation test cases from "Excute Test"	Closed	2. Critical	8
User Story	SAD-1113	As a test designer, I need input automation scripts code information for every automated test case in specific test plan	Closed	2. Critical	8
User Story	SAD-885	Release engineer can run TQ performance test from teamcity.	Open	2. Critical	
User Story	SAD-722	As a viewer of load test results, I can easily find and view TQ Load Test results and chart trends over time	Closed	2. Critical	5
User Story	SAD-721	As a release engineer I can run all existing TQ Load Tests for 12 hours and see no errors	Closed	2. Critical	
User Story	SAD-646	As a engineer, I can read the wiki about Configurator and attend a brown bag demo on the tool and get a fuller understanding of the tool and how it affects me	Closed	2. Critical	8
User Story	SAD-645	As a release engineer, I can write TQ load tests focused on search functionality and have them easily and repeatedly executed and they will pass	Closed	2. Critical	20
User Story	SAD-644	As a release engineer, I can write TQ load tests focused on basic file functionality and have them easily and repeatedly executed and they will pass	Closed	2. Critical	40
User Story	SAD-597	As a Release Engineer I can successfully convert three WebAii into WebTest/load tests using WebAii Canvas updated beta release	Closed	2. Critical	20
User Story	SAD-514	As a release engineer, I can write 2 load tests exercising Vanilla Tequila so that I can become more familiar with the VS 2008 LT framework	Closed	2. Critical	13

Product Backlog - cont.

- Three properties for good product backlog
 - User story
 - Estimation
 - Priority

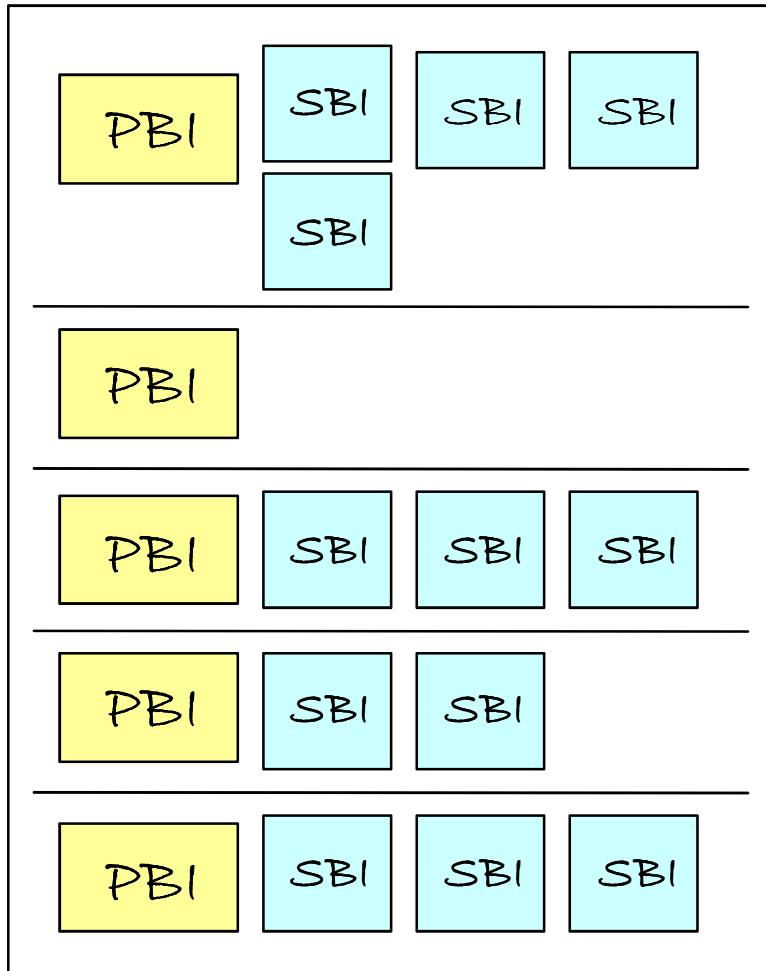
Sprint

Sprint

- The name for one iteration in Scrum
- The basic pace of Scrum
- Normally the length is 1 ~ 4 weeks
- The length is fixed in order to have stable team rhythm
- Complete everything



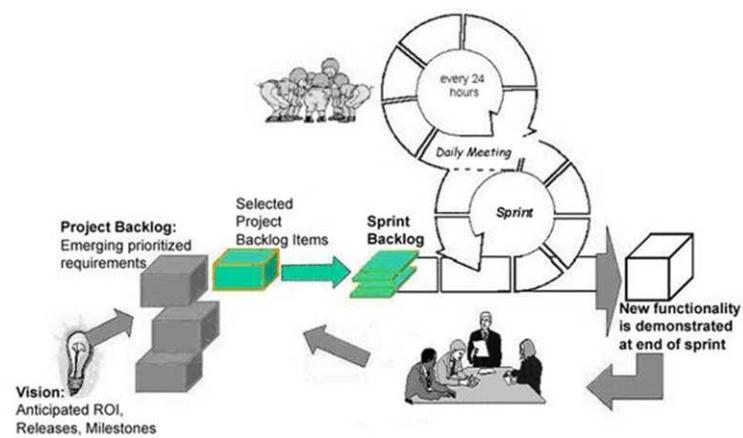
Sprint Backlog



- Created during Sprint planning
- Task list to complete current Sprint
- Owned by team
- Each team member can update it
- Tasks could emerge / disappeared

Start Point - Sprint Planning

- Held at the beginning of one Sprint
- Required a prioritized product backlog
- Involved PO & all of the team members
- Is going to create Sprint Backlog
- No more than 2 hours for one week



Sprint Planning Part I

- Select user stories
- Make estimation
- Set up sprint goal



- Choose high priority user story
- Brief detail of each to team
- Answer question

- Understand detail of each story
- Know team's capacity
- Make estimation
- Decide IN or OUT

Sprint Planning Part II

- Breaking down task
- Finalize Sprint backlog



- Answer question

- Breaking down task
- Estimate task effort
- Check team's capacity

Daily Standup

- Communication opportunity to the whole team
- Everyday, every team member
- Same time, same place
- No more than 15 mins
- All stand (Why?)



Three Questions

- What did I get **DONE** yesterday?
- What will I get **DONE** today?
- Are there any **IMPEDIMENTS** slowing my progress?

It is NOT...



- A status report meeting to manager
- A solution discussion meeting for any issues

It is ...

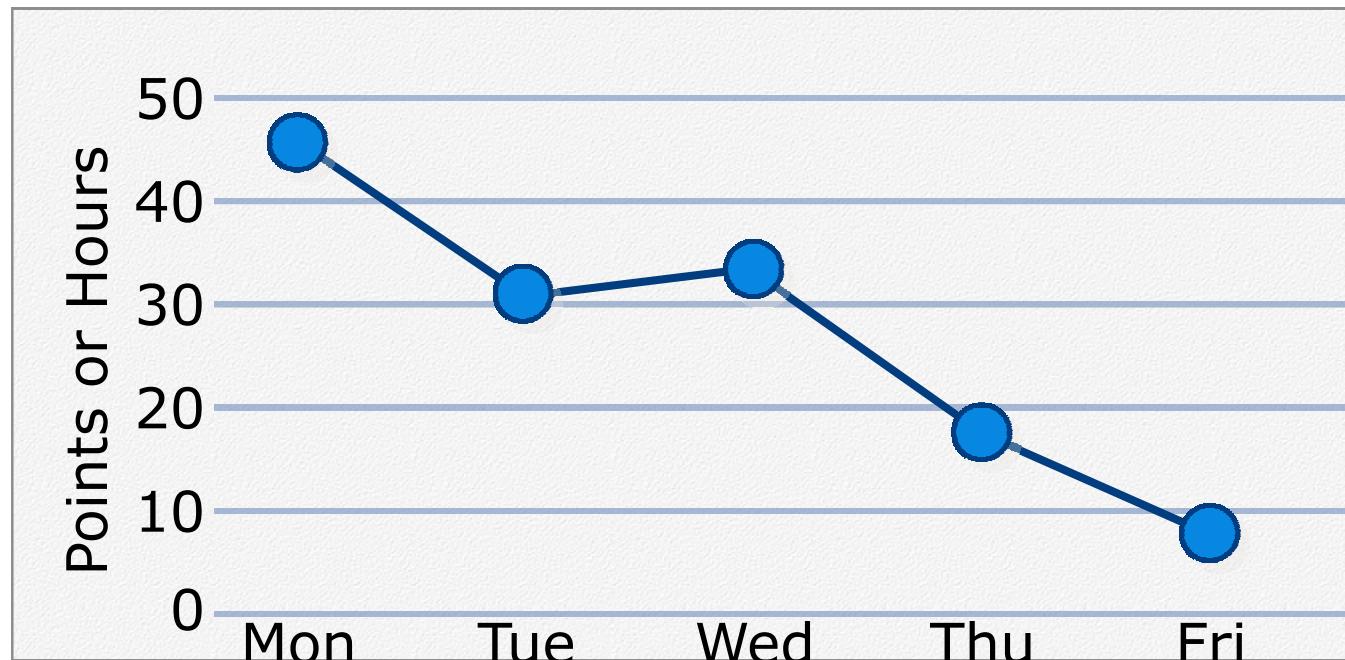


- Opportunity to update your status to your team
- Plan with your team member together for new day
- Raise any obstacles that stop you

Burndown Chart

- The most important status tracking tool
- Updated daily by the team
- Focus on left works
- Have different approach
 - Estimated remaining efforts
 - Tracking DONE

Tasks	Mon	Tues	Wed	Thur	Fri
UI Coding	8	4	8		
Biz Layer Coding	16	12	10	7	
Biz Layer Test	8	16	16	11	8
Make Online Help	12				

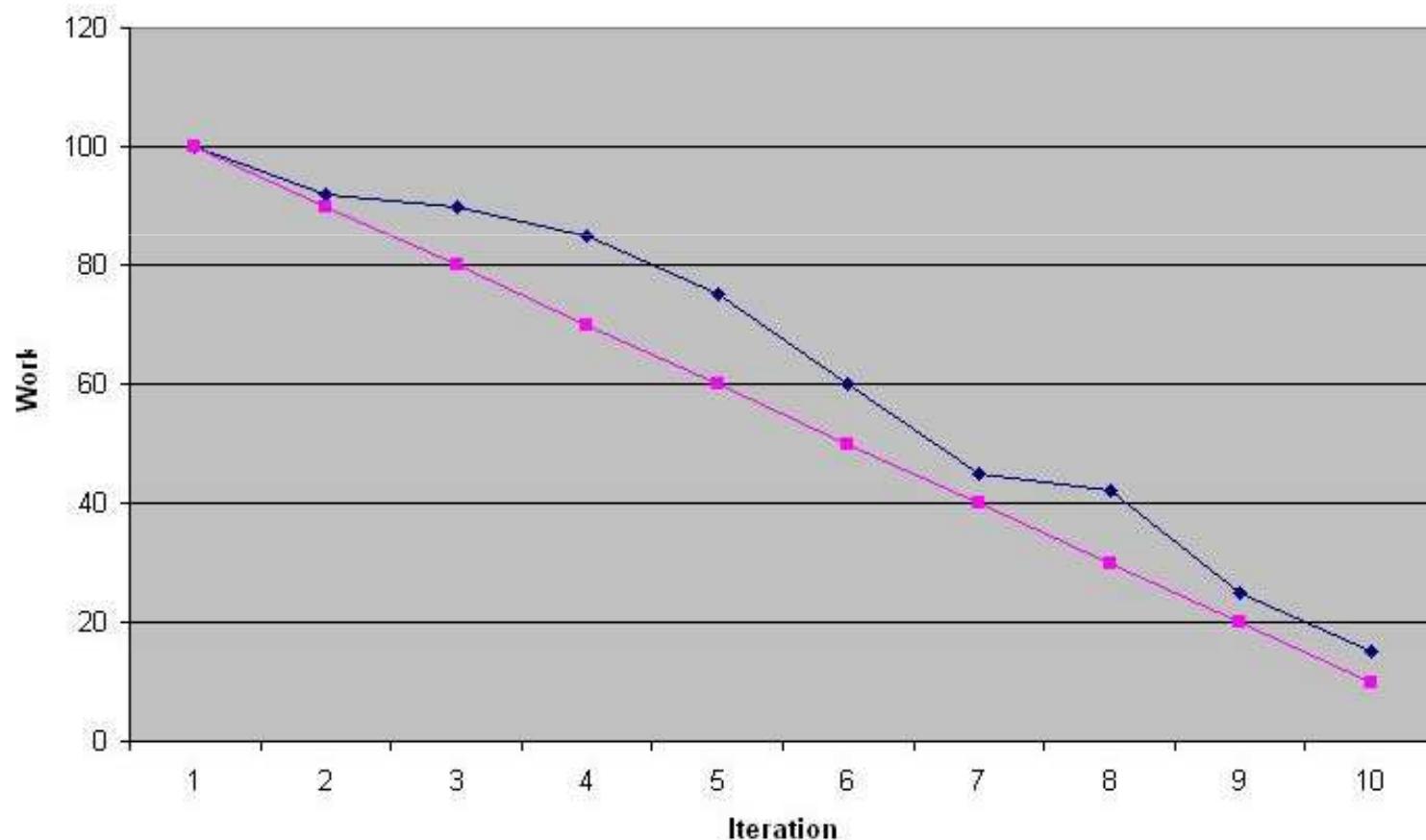


Thanks to Mountain Goat
Software, LLC



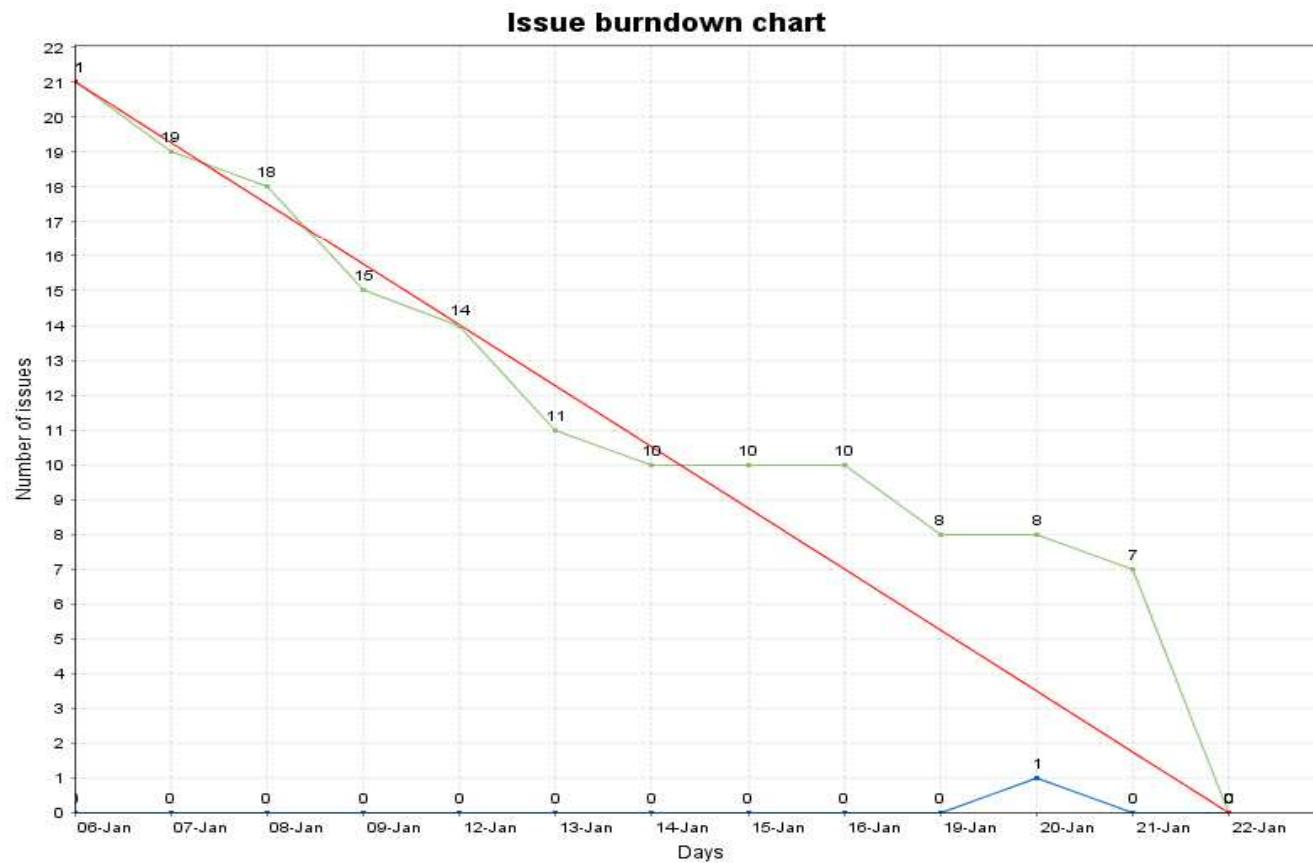
Hour Burndown

- How many remaining efforts?



Issue Burndow

- How many remaining tasks?



Are There High Level Burndown?

- Release burndown
 - Remaining size of total user stories
 - Help to predict the possible release time
 - Maintained by PO

Sprint Review

- Held at the end of the Sprint
- Whole team participates
- Invite the whole world
- Team demo potentially shippable features - what was DONE
- PO decides to accept the result or not
- Simple tips
 - Prepare it no longer than 2 hours
 - No slides



Sprint Review

- An opportunity to get feedbacks from PO and stakeholders
- PO may adjust product backlog according to sprint result
- Help team to establish trust with PO and stakeholders

Sprint Retrospective

- Held after Sprint review
- Typically 15 ~ 30 minutes
- Whole team participates
 - ScrumMaster
 - PO
 - Team
 - Stakeholders



Sprint Retrospective

- The opportunity to see what is and is not working
- Make action plan for improvement

Sprint Retrospective

- Start / Stop / Continue
 - The whole team gathers and discusses what they'd like to:

Start Doing

Stop Doing

Continue Doing

This is just one
of many ways to
do the
retrospective

Sprint Retrospective



- Some Tips:
 - Everyone should talk
 - Regardless of what we discover, we understand and truly believe that everyone did the best job they could
 - Close-door discussion
 - Nothing related to personal performance

Roles

- Product owner
- ScrumMaster
- Team

Ceremonies

- Sprint planning
- Sprint review
- Sprint retrospective
- Daily scrum meeting
- Sprint

Artifacts

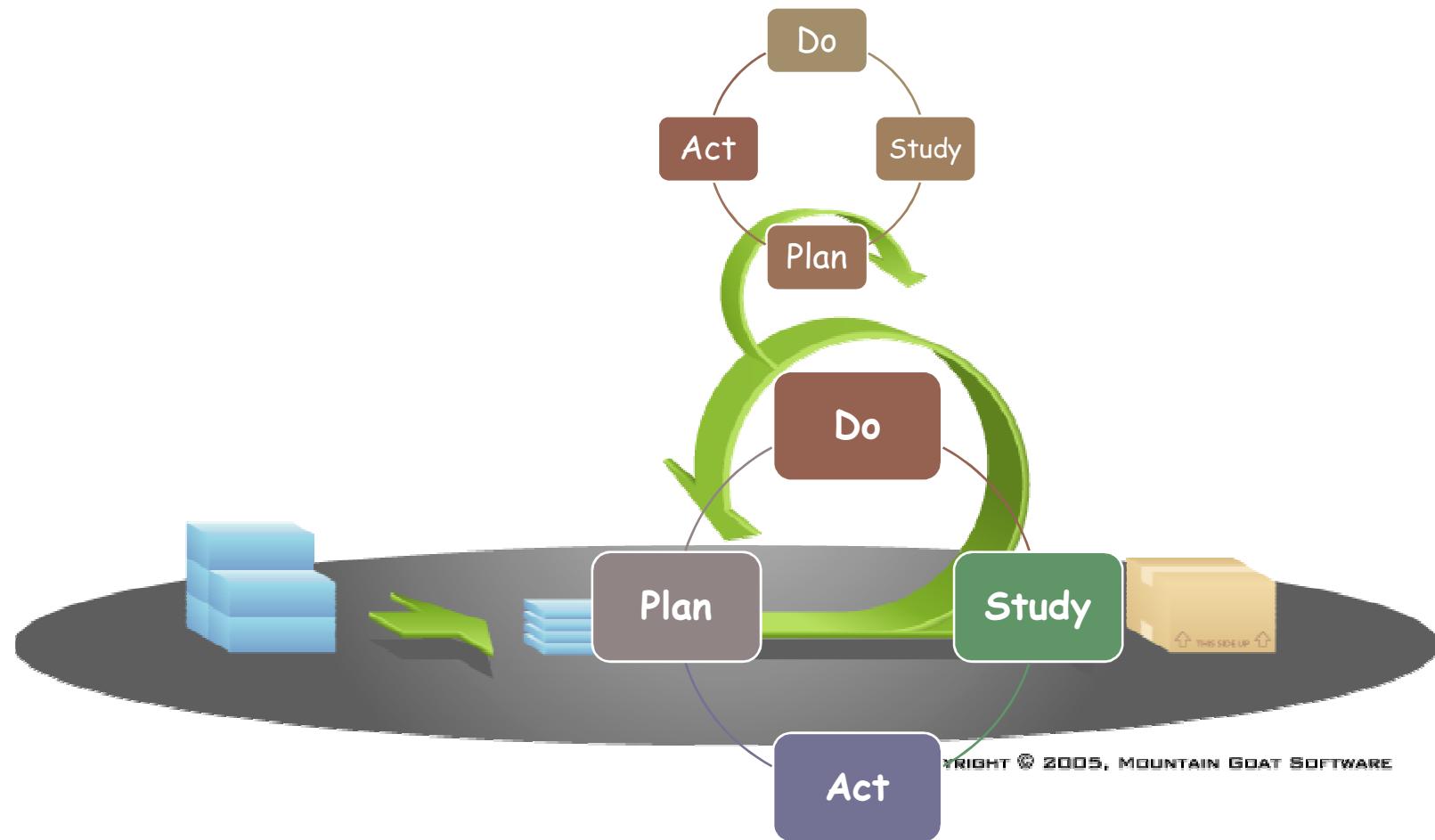
- Product backlog
- Sprint backlog
- Burndown charts



Thanks to Mountain Goat
Software, LLC



Deming Cycle



Agile requirements management

User Story Type

- Epic

As a purchaser, I can buy a book online

- Theme

- Support banking card pay
- Support pay pal
- Support pay after receiving the book

- User Story

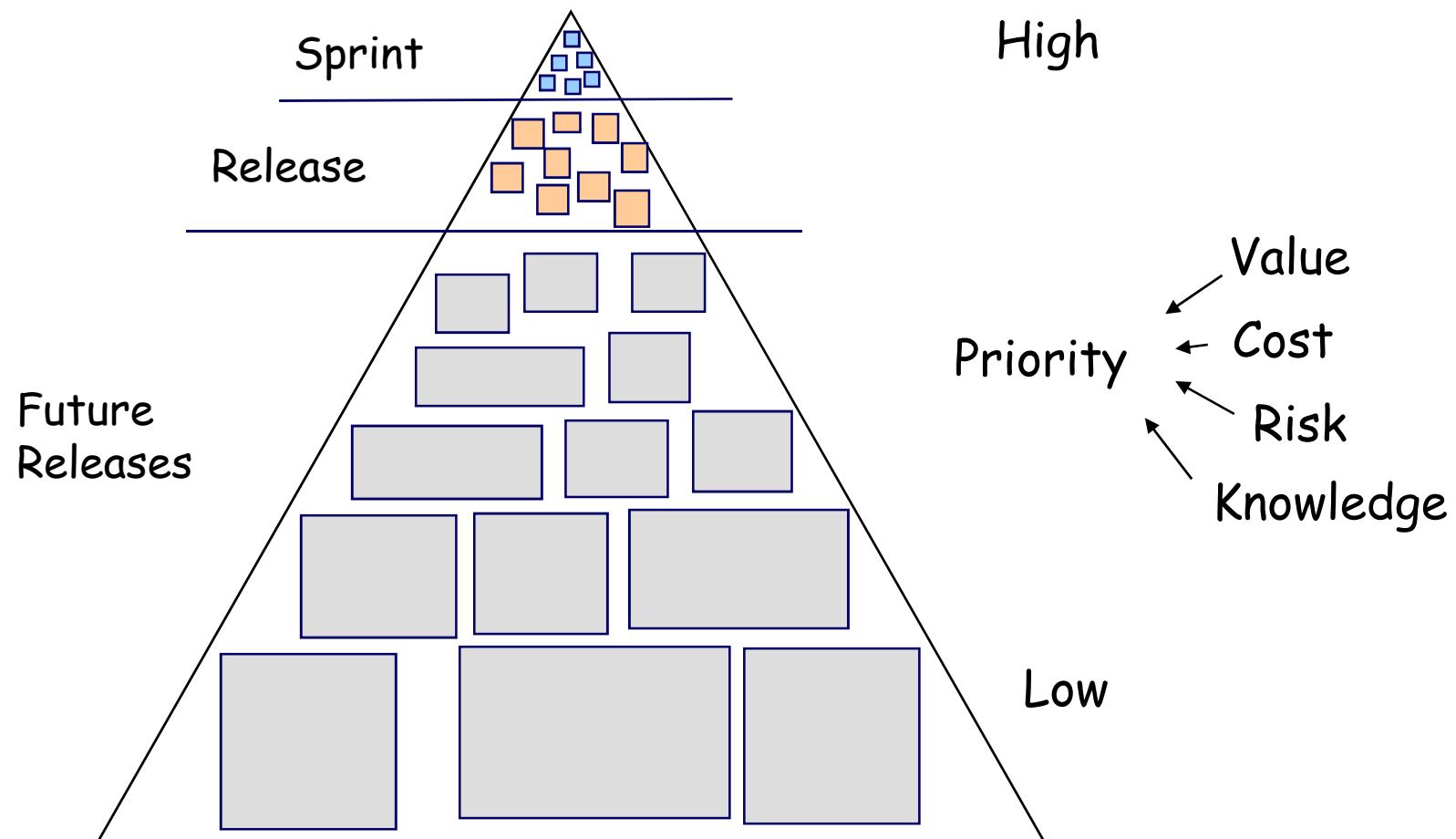
As a purchaser, I can pay with my VISA card online to buy a book

Just In Time



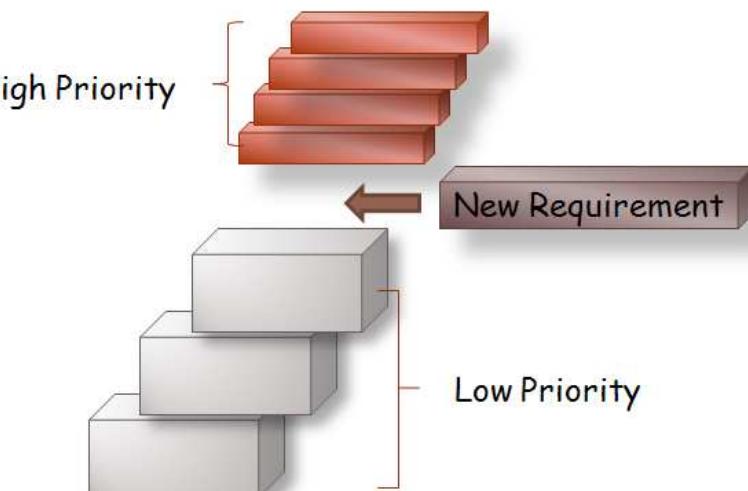
User stories (mostly epics) are then iteratively refined into more granular user stories with more details but only when they are needed.

Focus on Most Important Req.



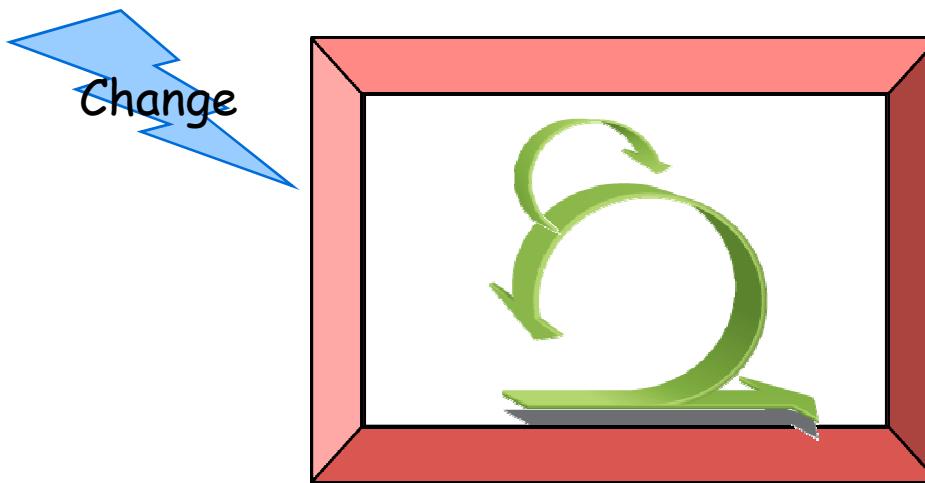
Change Requirement

- PO manage the requirements change through product backlog maintenance
 - Add new user stories
 - Change priority according to the situation
 - Decide to stop or not according to earned values

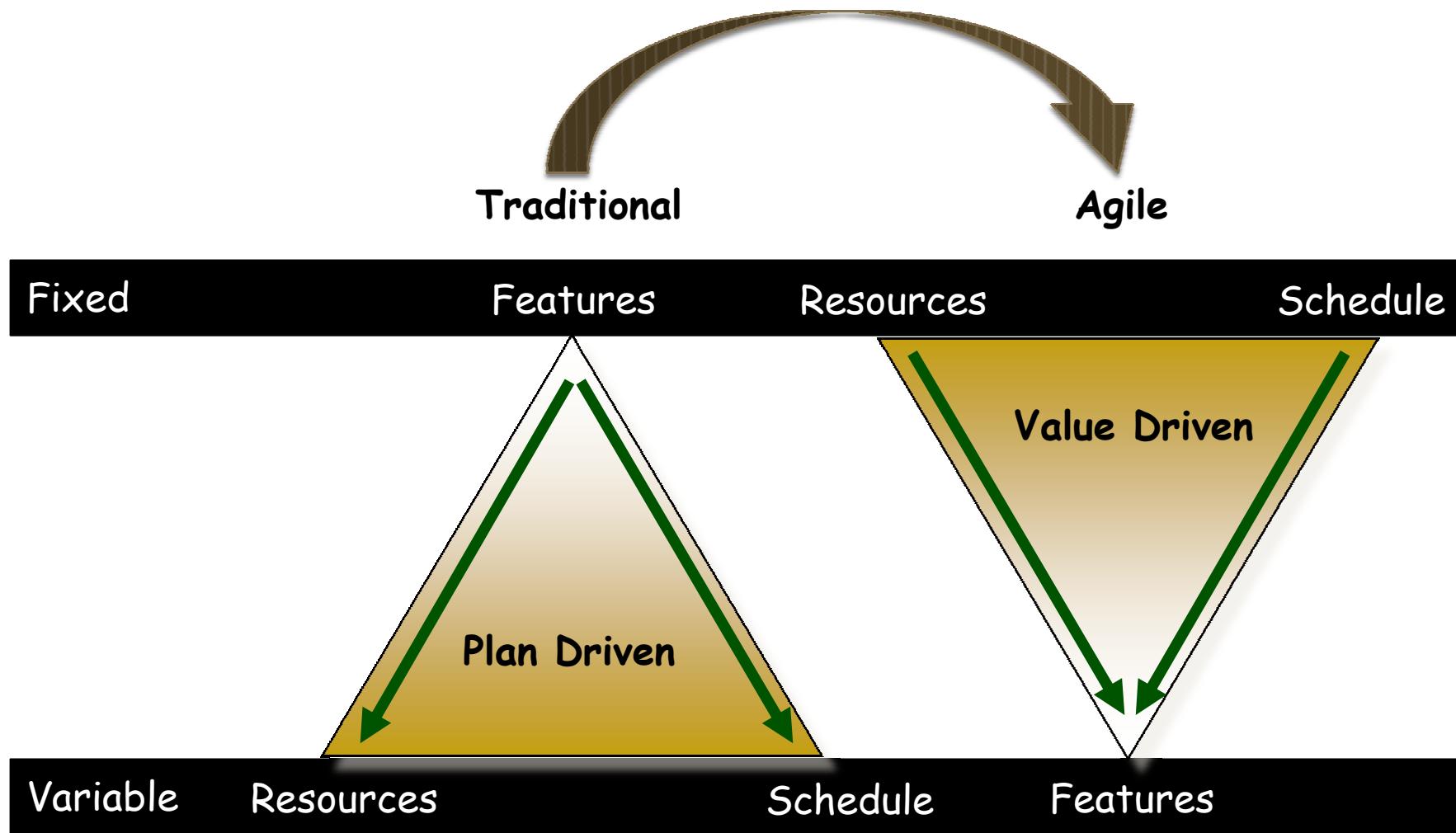


Can Scope be Changed in Sprint?

- The scope should be stable in one Sprint
 - Help the team to set up stable rhythm
 - Help the team to get to know the capability
 - Help to have better plan



Project Management Iron Triangle



Case Study

- Case 1
 - There are still more time left after complete all commitments
- Case 2
 - PO wants to add a very important feature to the sprint
- Case 3
 - During the planning, the team cannot commit even one story

Divide User Stories

- Divide on Data Boundary
- Divide on Operation Boundary
- Divide on Cross-cutting Functions
- Divide on Functional and None-Functional
- Divide on Priority

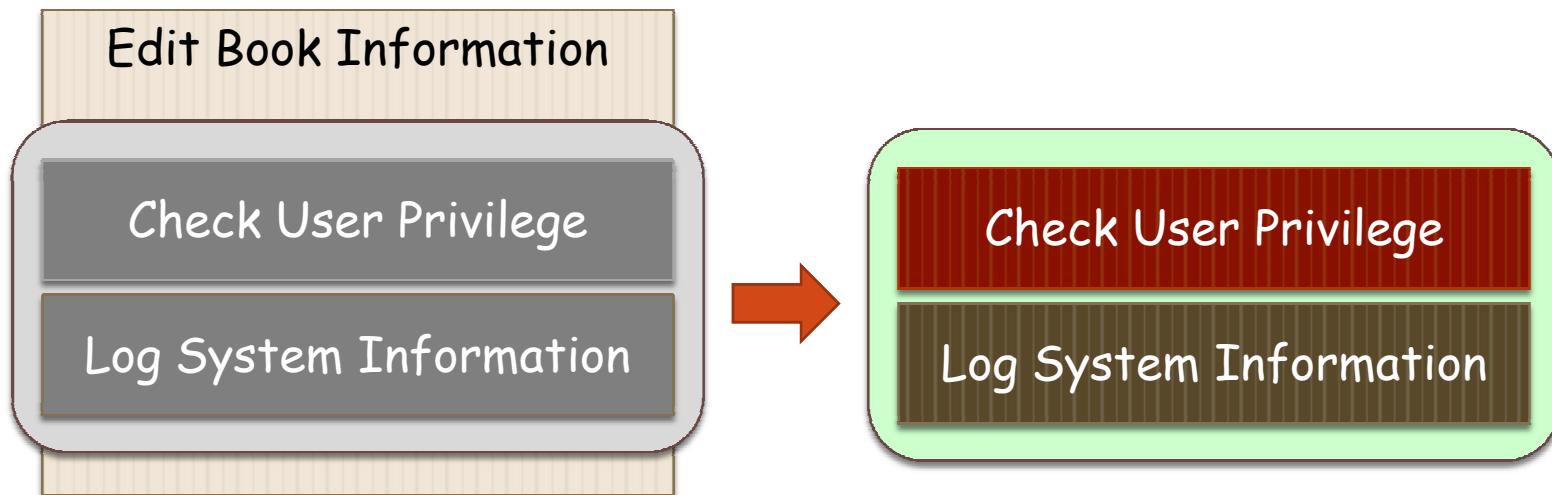
Data Boundary

- Inquiry book
 - By Title
 - By Author Name
 - By Key Words
 - By Category

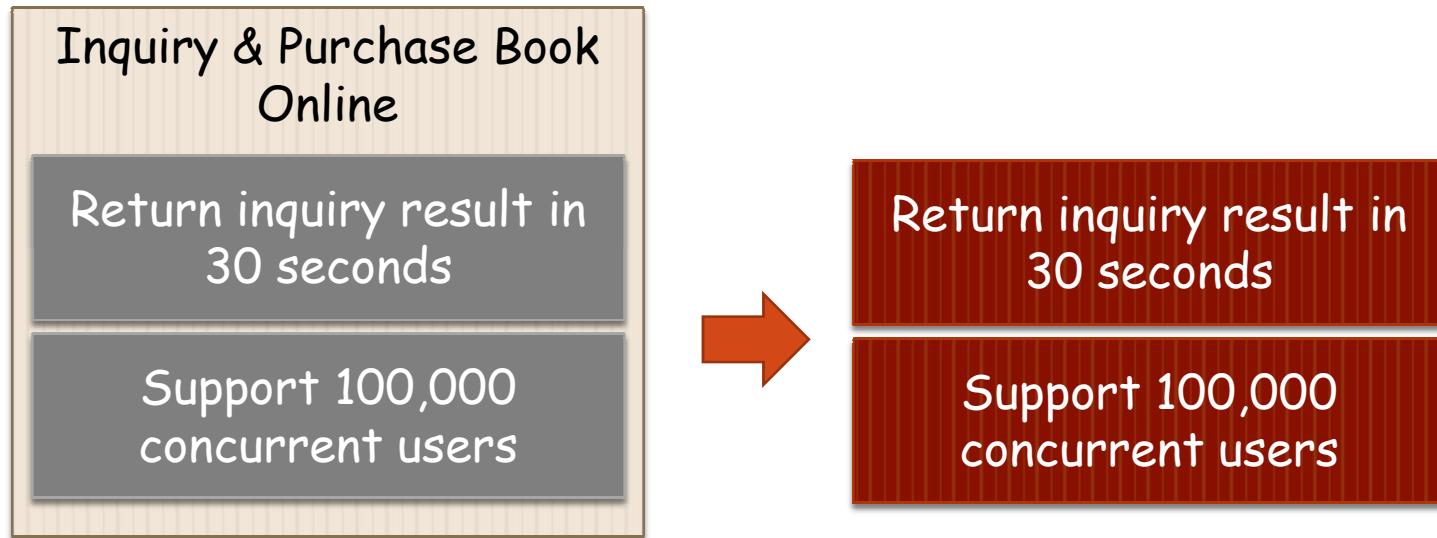
Operation Boundary

- As a seller, I can add new book
- As a seller, I can edit a book information
- As a seller, I can remove a book from the list

Cross-cutting Functions



Functional & Non-functional

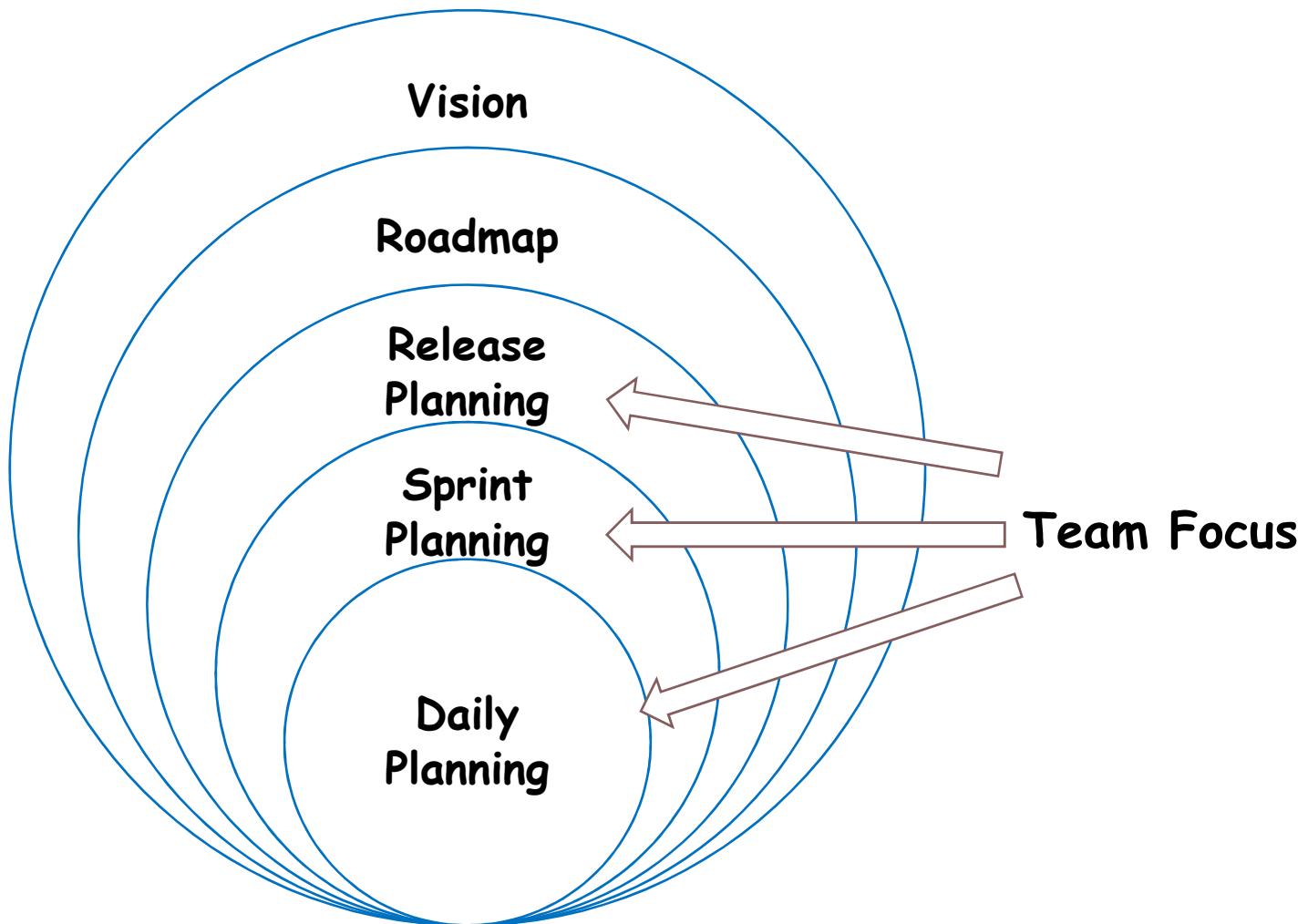


Priority

- As a user, I want be able to logon the system with my id and password
 - If the id and password are both matched, user can logon system;
 - If the user tried wrong id or password for 3 times, the account should be locked;
 - If the logon request had been rejected, a warning message should be sent to the user

Agile Planning & Estimating

Overview



Vision



- Long term target
- Direction
- Won't change frequently

Roadmap

- Very high level plan
- Big paces to archive the vision

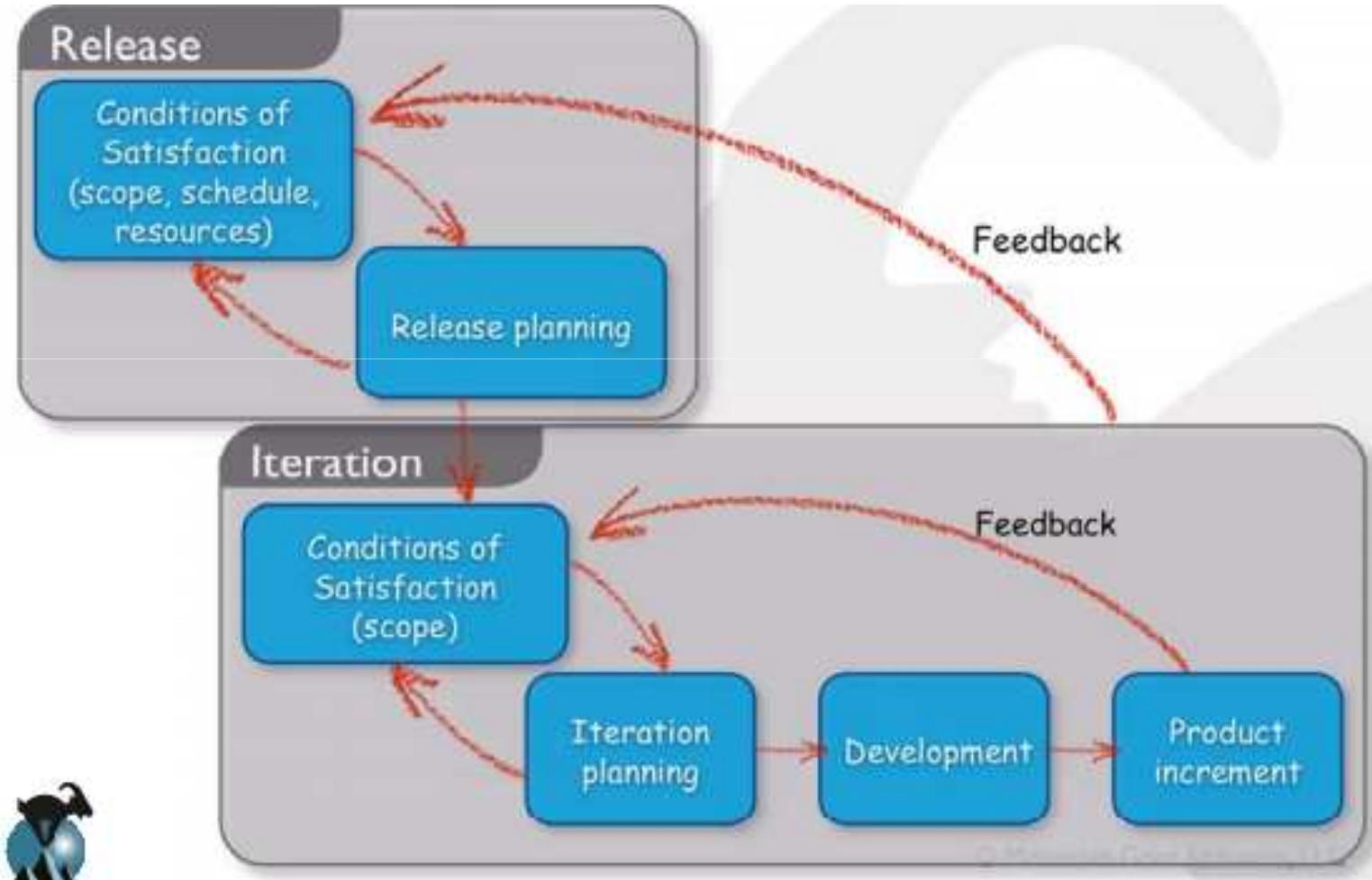


Release Plan

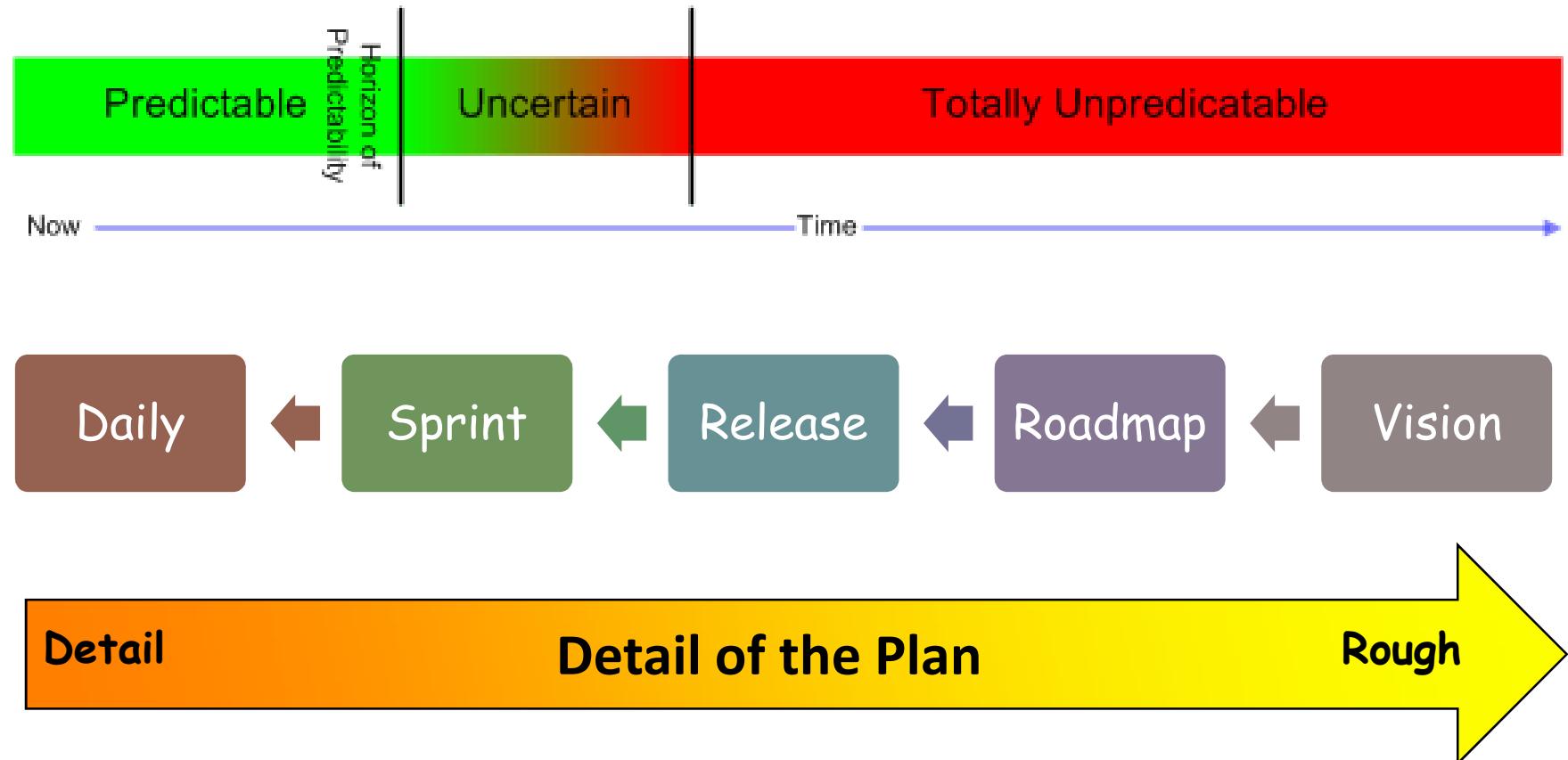
- Focus on 3 ~ 9 month



Agile Approach to Planning



Just in Time





How long will you take to move all of them?

How to Estimate it?

- We need to estimate the size first
 - How many wheelbarrow loads it represents?
- Get my velocity
 - After one hour, how many wheelbarrow loads I've moved
- Extrapolate the total duration

Size

- I think that is 100 wheelbarrow loads

Velocity

- After one hour, I've moved 20 wheelbarrow loads

Duration

- So I will move all of them in totally about 5 hours

Agile Estimation is Relative

- Allow different delivery conditions
 - Who will complete the work?
- Allow self-adjusted velocity
 - Velocity will be faster when get more familiar with the program language, but the size should be the same
- More easier to get agreement among the team

Story Point

- Probably the most commonly used estimating unit among agile teams today
 - Name is derived from agile teams commonly expressing requirements as "user stories"
- Based on a combination of the size and complexity of the work
- Unitless but numerically relevant estimates
 - A 10-point user story is expected to take twice as long as a 5-point user story

Why Story Point

- Forces the use of relative estimating
 - Studies have shown we're better at this
- Focuses us on estimating the size, not the duration
 - We derive duration empirically by seeing how much we complete per iteration
- Puts estimates in units that we can add together
 - Time based estimates are not additive

Planning Poker

- An iterative approach to estimating, loosely based on wideband Delphi
 - PO briefs one user story
 - Team discusses about the work to implement the story
 - Each team member select one value card to give his/her estimation
 - Discuss differences (especially outliers)
 - Re-estimate until estimates converge



Example



Estimator	Round 1	Round 2
Susan	3	5
Vadim	8	5
Ann	2	5
Chris	5	8



Advantage of Planning Poker

- Estimated by people who will do the work
- Estimators are required to justify estimates 2~3 times
- Focus on more important thing
- Combine individual estimates through group discussion and lead to better estimates
- Emphasizes relative rather than absolute estimating
- Estimates are constrained to a set of values so we don't waste time in meaningless argument
- Everyone's opinion is heard
- It's quick & fun



Velocity



- Total story points that team can complete in one Sprint

It is Always a Range

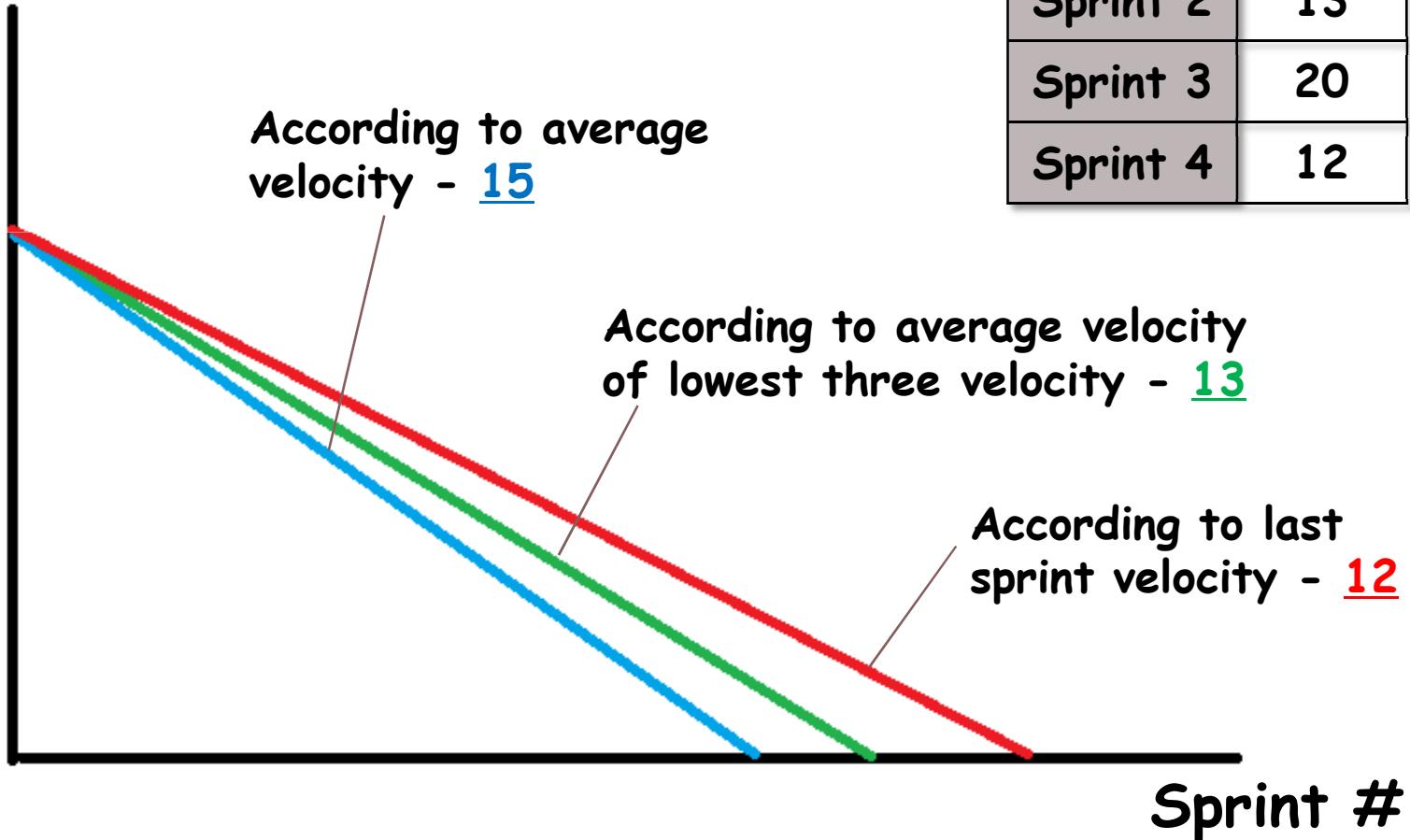
Total Left Story
Point

According to average
velocity - 15

Sprint 1	14
Sprint 2	13
Sprint 3	20
Sprint 4	12

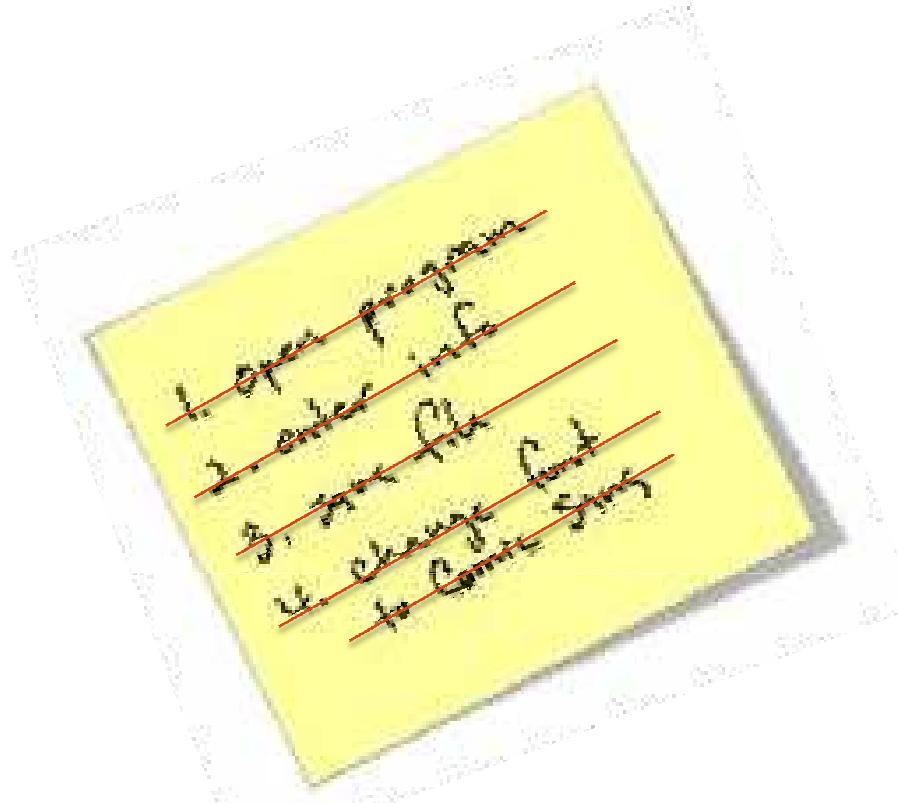
According to average velocity
of lowest three velocity - 13

According to last
sprint velocity - 12



Productive scrum

DONE



- Complete all of the things
- Deliver potential shippable product on the end of each Sprint

Ready

- Is Product Backlog ready?
- Is the team ready?
- ...



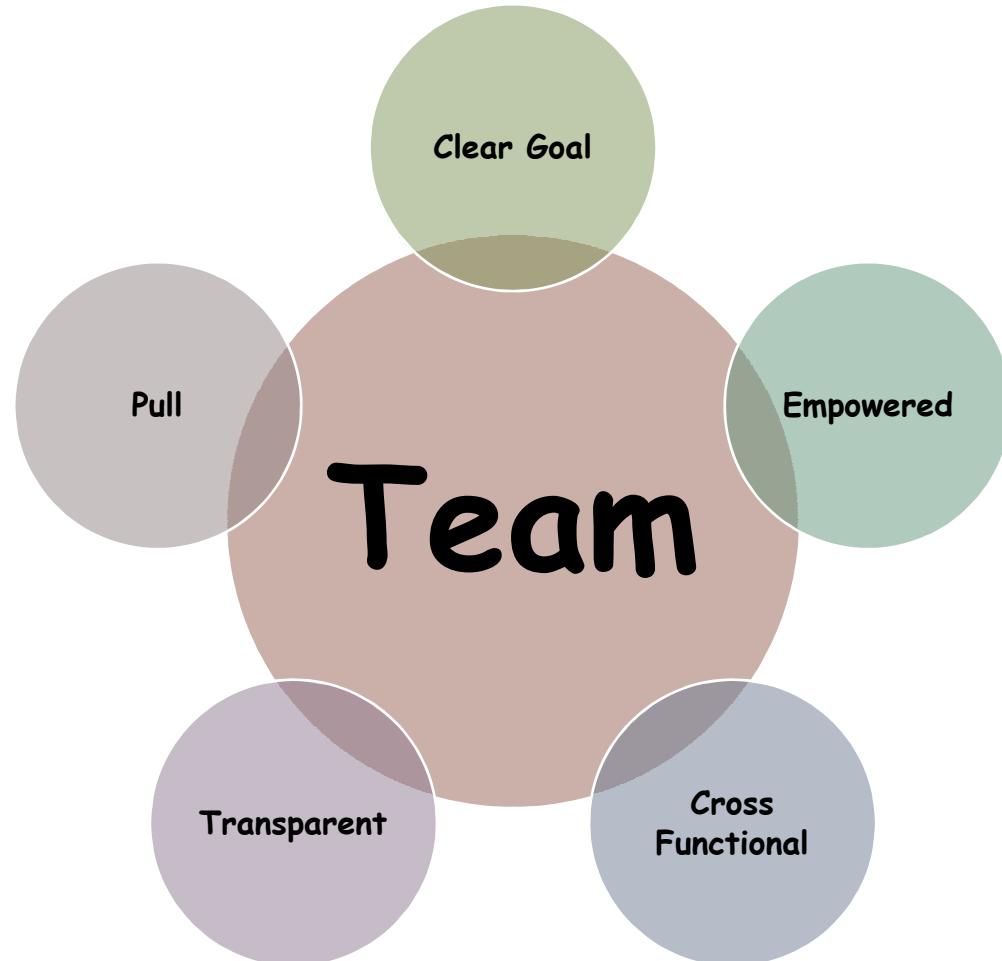
Self-organized Team



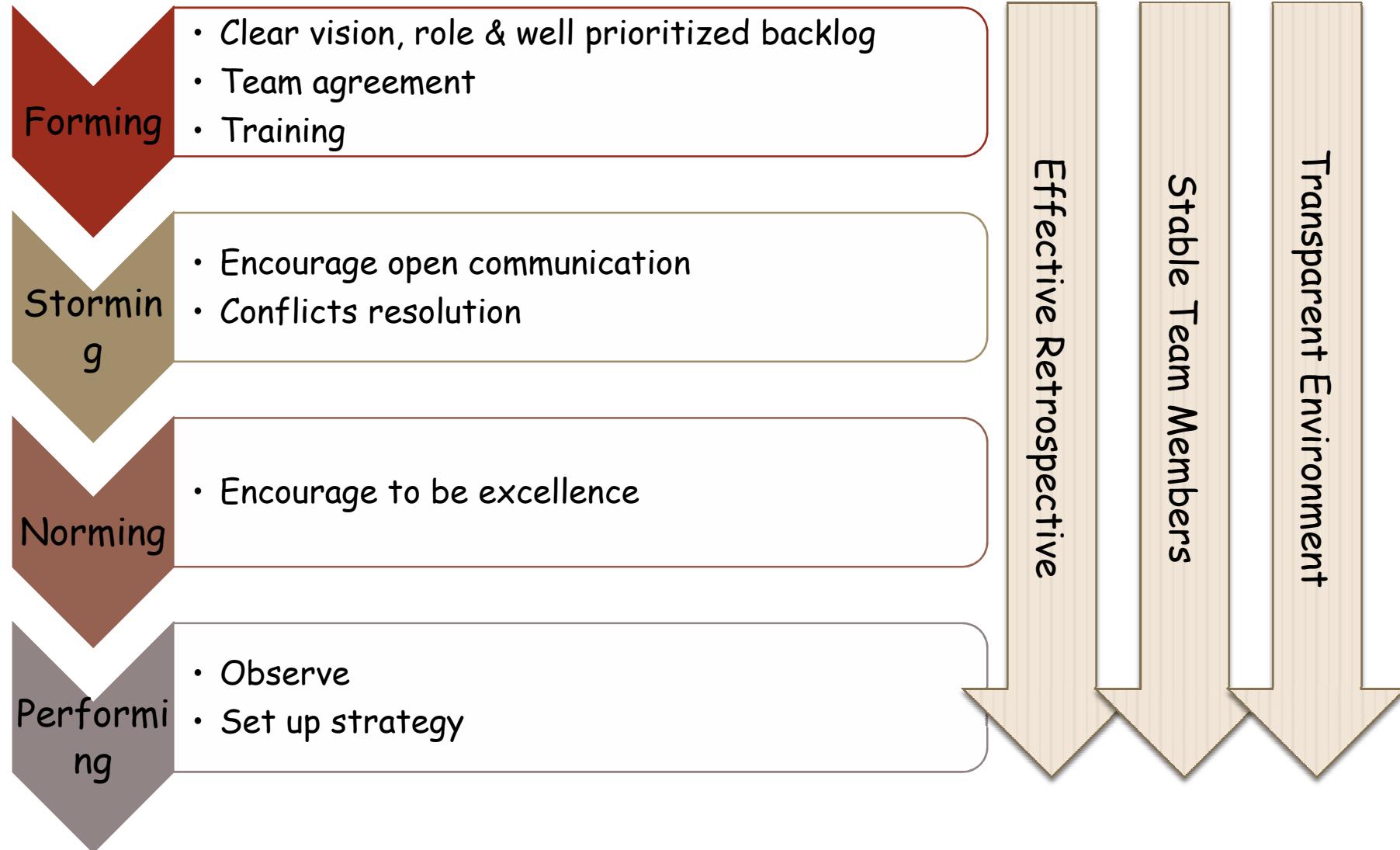
What is Team

- A team is a small number of people with complementary skills who are committed to a common purpose, performance goals, and an approach for which members hold themselves mutually accountable.

How to Build



How to Build



Discussion

- The team agreements you wanted

Informative Workspace

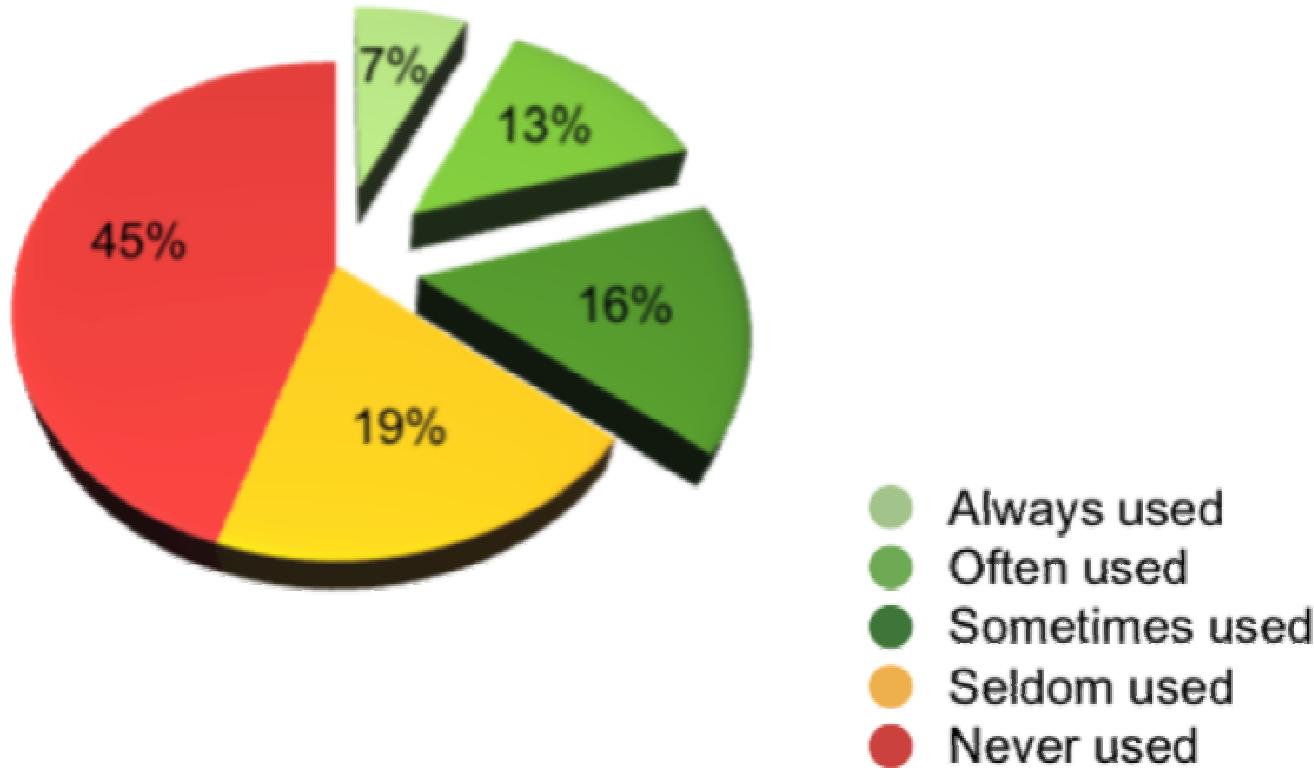
- Communication
- Transparency
- Fast Feedback
- Healthy Pressure





Quality

Where do we start from



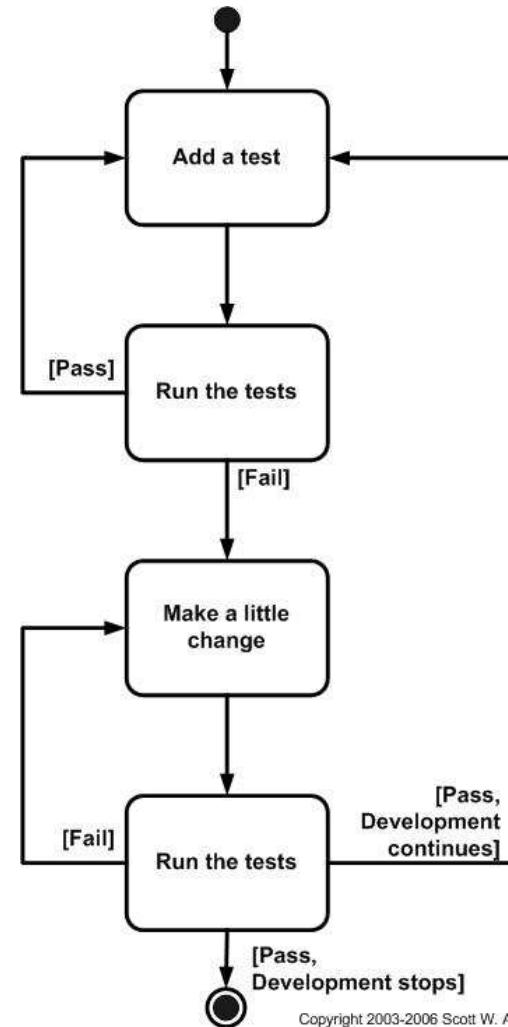
- Start from high quality requirements

Testing

- Agile process place very strong emphasis on quality and verification through testing
- In Agile, testing is a continues activity, not an end-of-project "phase"
- An activity is never complete unless there is a test for it

Test Driven Development

- “Only ever write code to fix a failing test.”
 - Very high test coverage
 - 100% pass rates
 - High quality
 - The ability to move fast
 - Better design
 - Confidence to refactor
 - Low accumulation of technical debt



Copyright 2003-2006 Scott W. Ambler

Scrum scaling

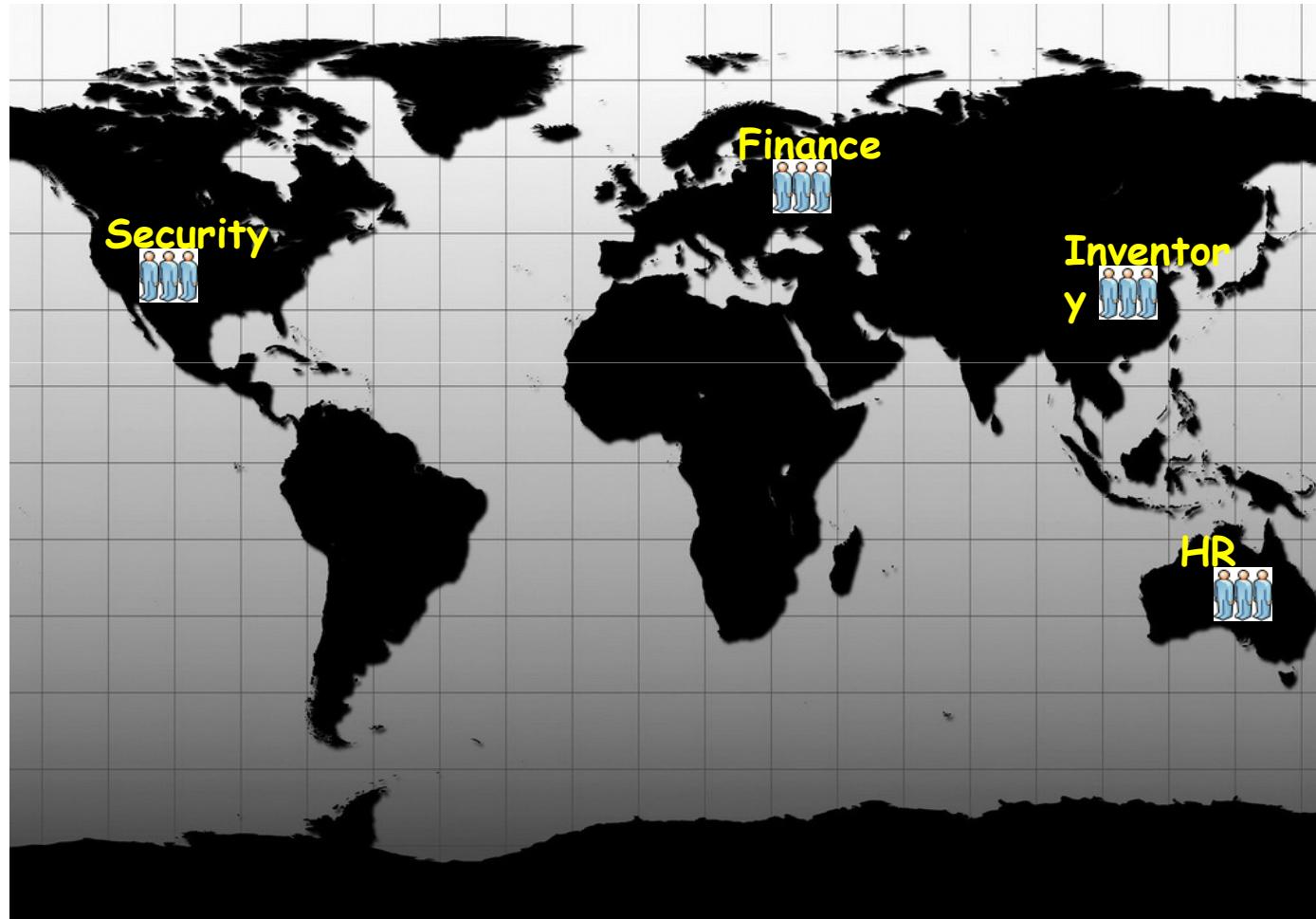
Complicated Project



Divide Teams

- Divide independent teams as possible as you can
 - By feature
 - By components
- Arrange each team's members work on same location

Divide Teams



Establish Collaboration Platform

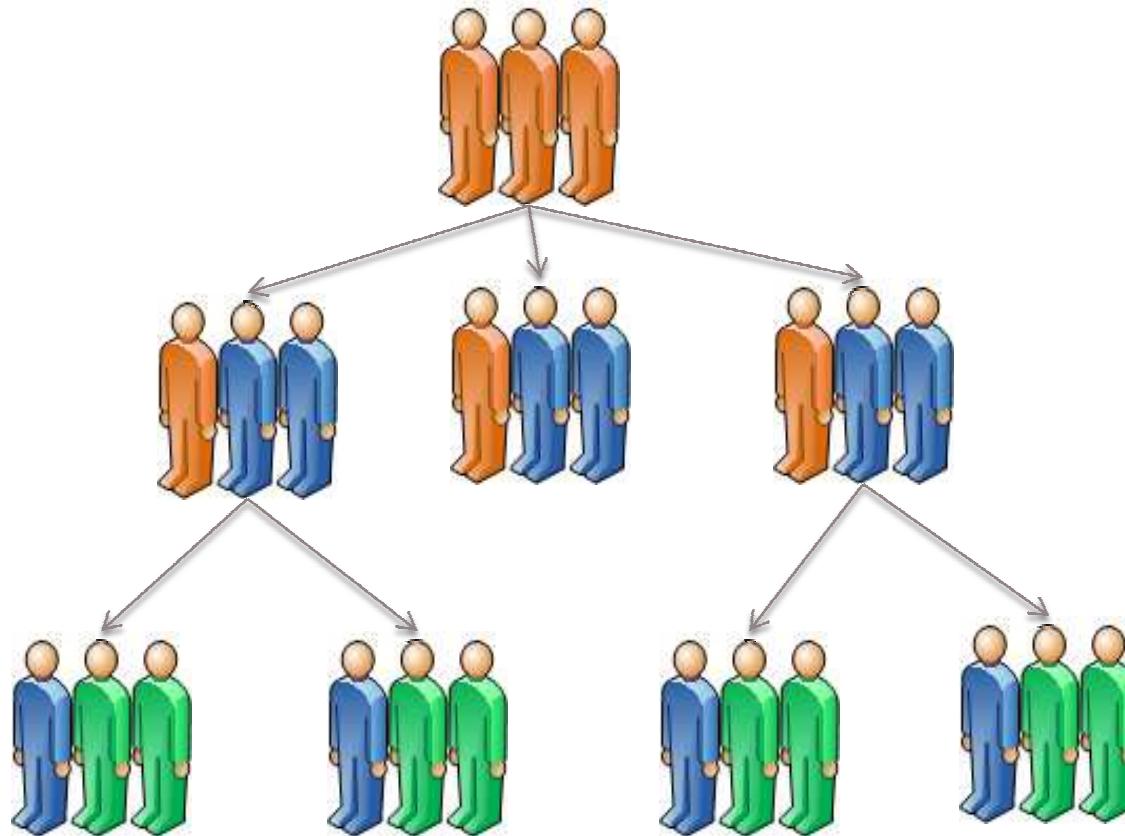
- Product owner group
- Continue Integration environment
- Efficient communication platform
 - Messenger
 - VOIP
 - Video Tel system
 - Information sharing platform
 - Wiki
 - Project site
 - ...



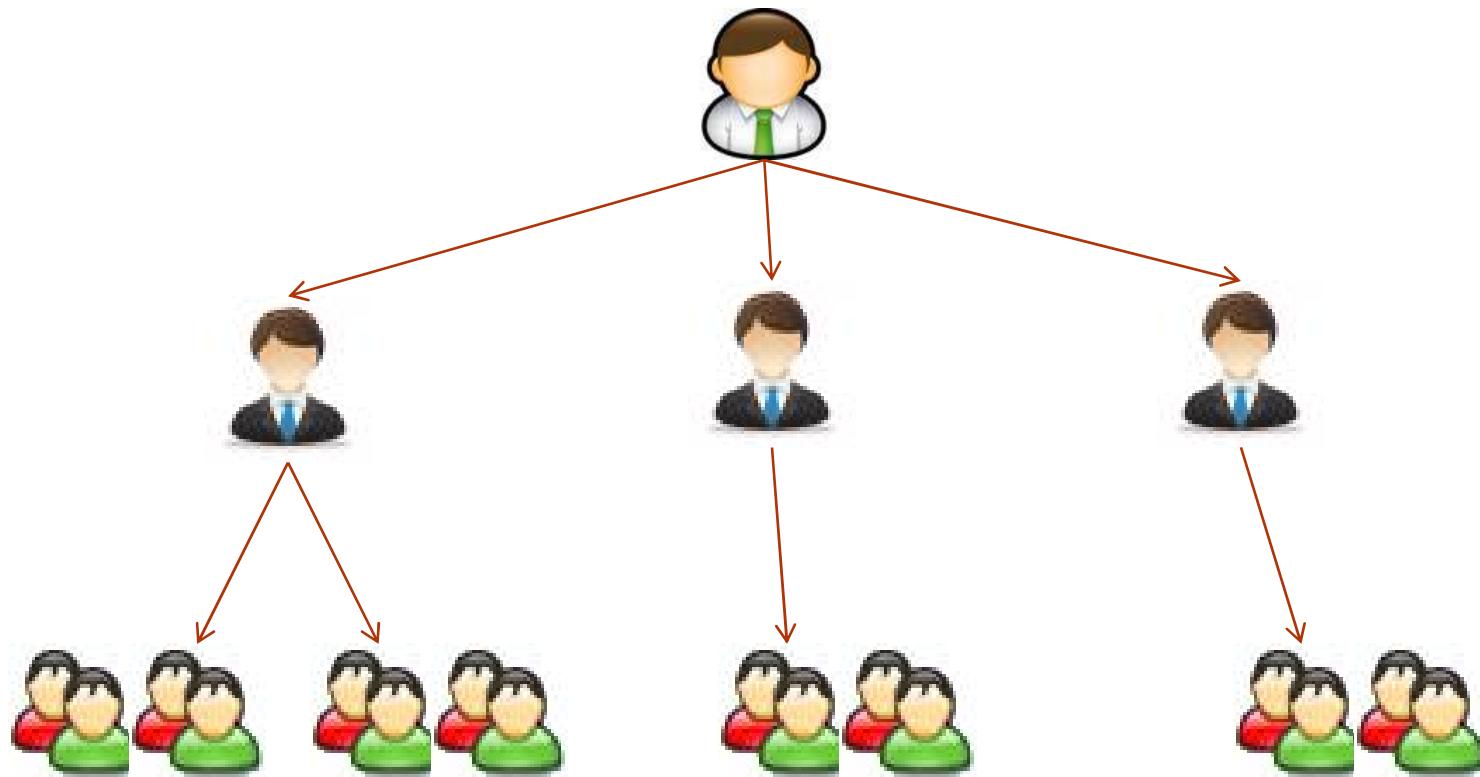
Budslife@Flickr

One to more...

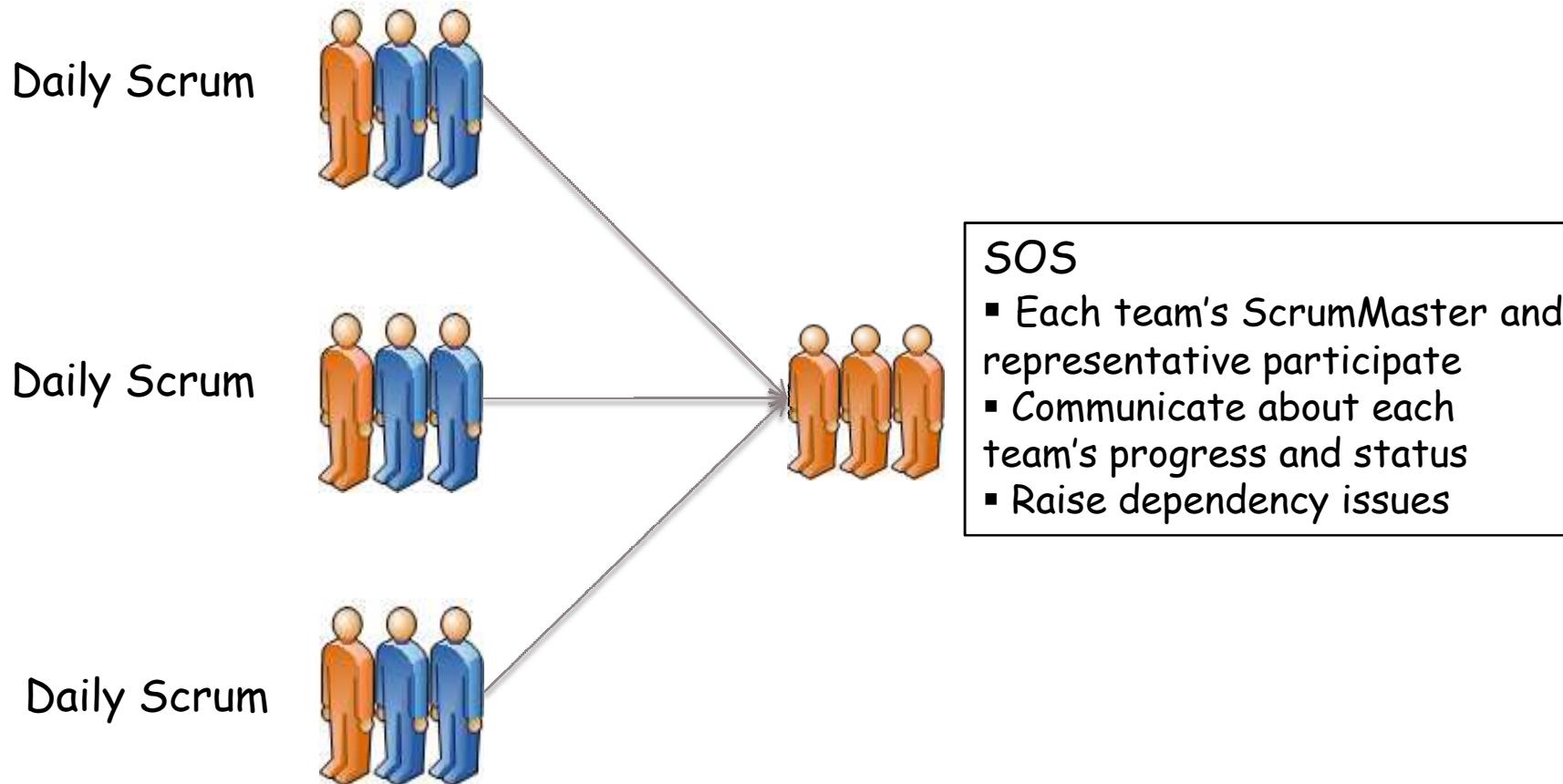
- Necessary design
- Important tech. std.
- Implement some features



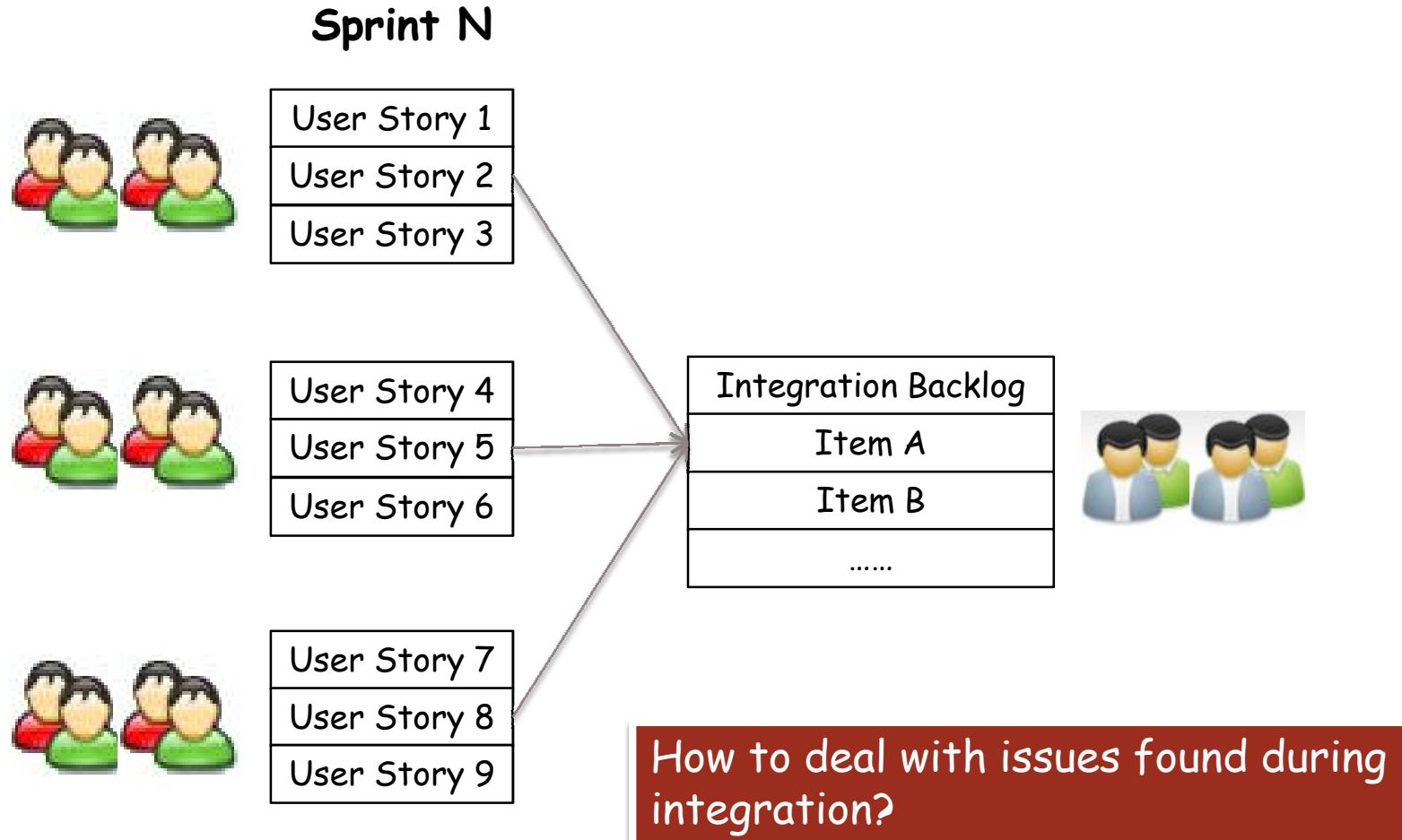
Product Owner Group



Scrum of Scrums



Independent Integration Team

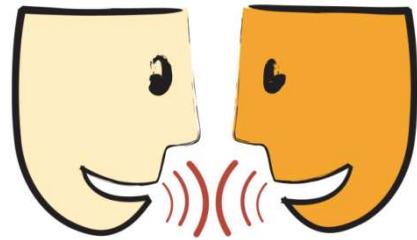


Product Demo

- Sprint review only demo the part the completed features of one team. For complicated project, frequently review for the whole product integrated features is needed.
 - All PO and ScrumMaster should participate
 - All of the teams or teams' representatives
 - All of the key stakeholders
 - Open to everyone

How to start

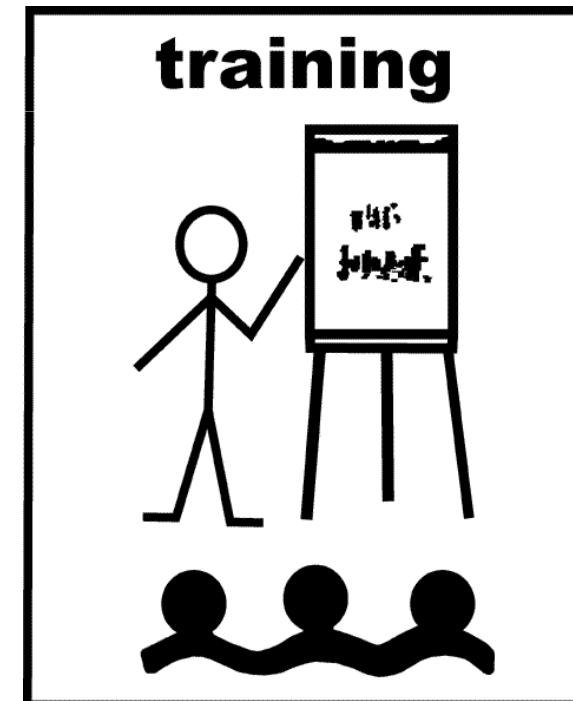
Communication



- Customer background & expectation
- Senior managers expectation
- Team situation
- Current process
- Existing issues, risks
- Expectation for Agile Scrum
- ...

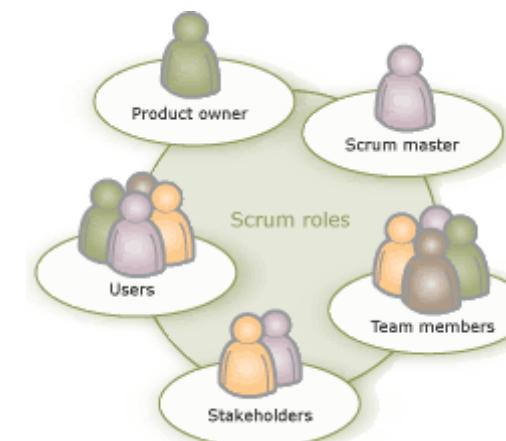
Training

- Training for managers
 - Agile Core value
 - Scrum basic concept
 - Get support
- Training for the team
 - Agile core value
 - Scrum framework
 - Agile Scrum practices
- Training for the customers
 - Agile core value
 - Scrum basic concept
 - Responsibilities of PO
 - Get Customer's support



Setup Team

- Find proper ScrumMaster
 - Good understanding about Agile & Scrum
 - Have experiences of project management
 - Very good communication skills
 - Have good coaching skills
- Find proper PO
 - Good understanding about Agile & Scrum
 - Very good at Agile requirement management
 - Very good communication skills
- Set up team
 - Size is about 5 ~ 9
 - Cross-functional

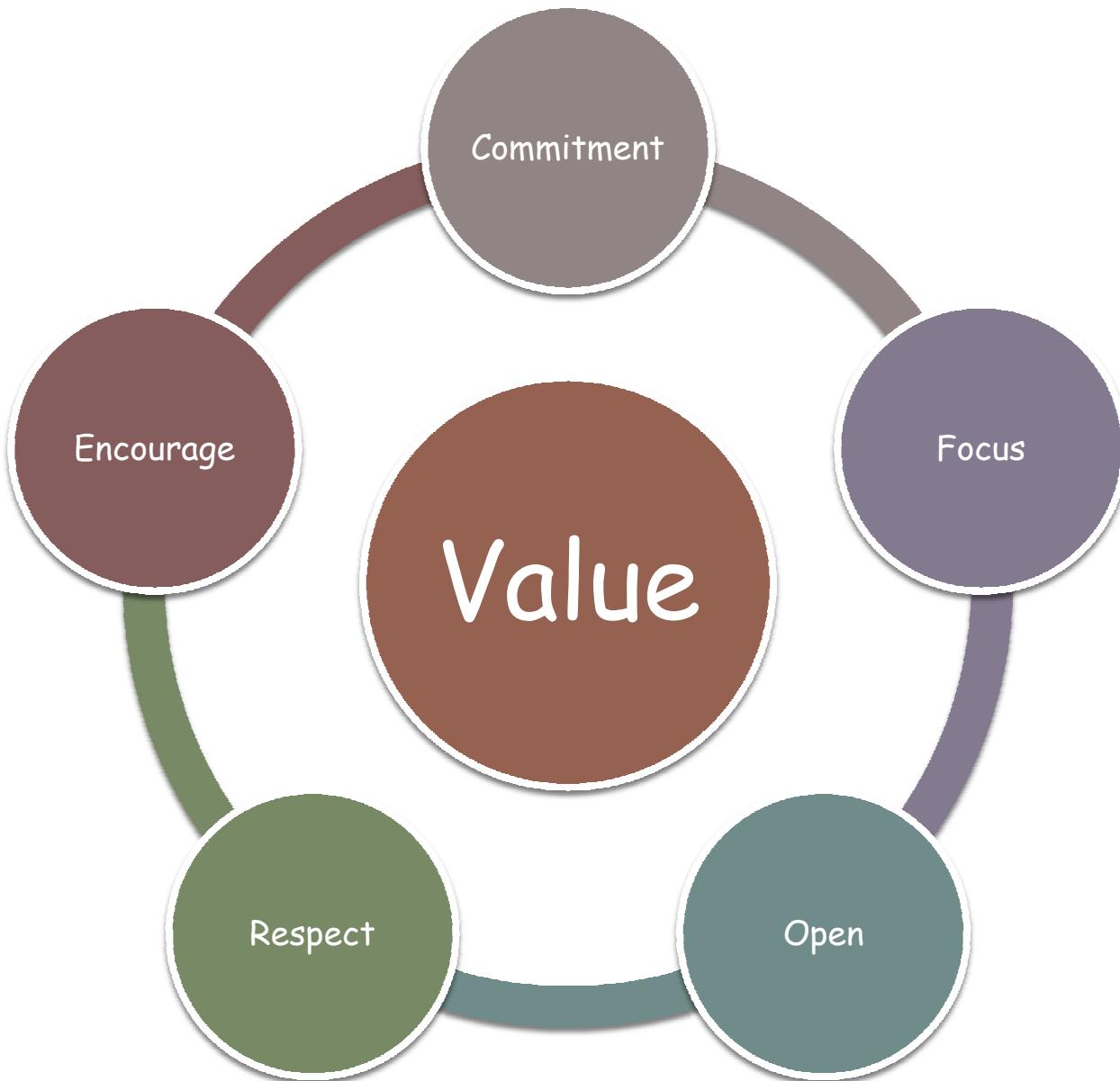


Establish the Base

- Hold vision meeting
 - Get agreement of the target
 - Identity important risks
 - Get agreement for sprint length, daily Scrum location and time
- Set up team agreement, like
 - Respect and trust to each other
 - To be excellent
- Set up well prioritized product backlog
- Get agreement for DONE definition
- Release planning

Vision
Mission
& Values

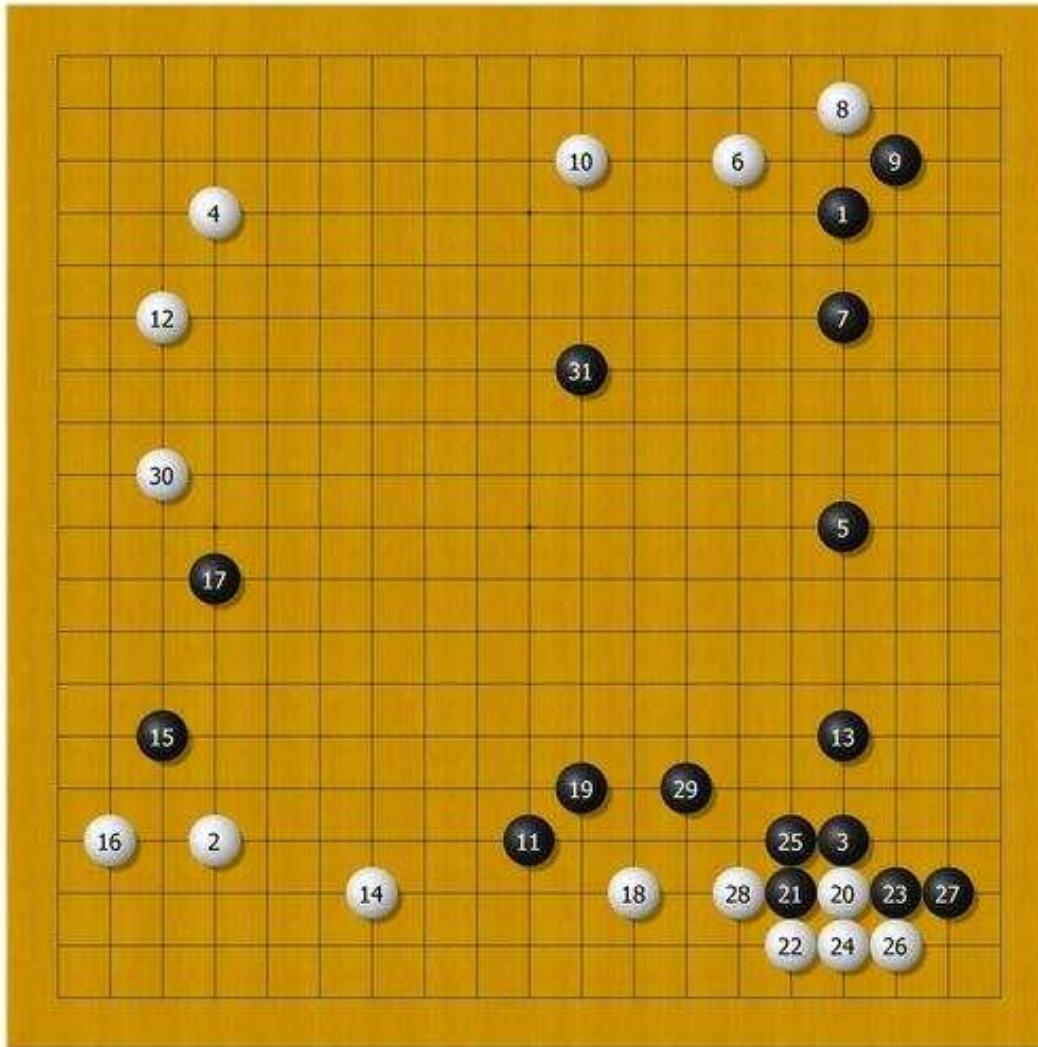
Conclusion



Is Agile Scrum Silver Bullet

Implement Scrum and all the reasons that an organization has trouble delivering quality software on schedule are thrown up in your face, day after day, month after month - made obvious and critical by Scrum

- Ken Schwaber



Agile like Weiqi, simple but not easy
- UPerform, Bill Li

**What will you do different in the
future?**

Reference

1. **Scrum Alliance** (<http://www.scrumalliance.com>)
2. **Agile Alliance** (<http://www.agilealliance.com>)
3. **ScrumChina Yahoo Group**
(<http://tech.groups.yahoo.com/group/ScrumChina/>)
4. **Agile Estimating and Planning** - Mike Cohn / ISBN: [978-7-302-15480-8](#)
5. **Agile Project Management with Scrum** - Ken Schwaber /
ISBN: [978-0-7356-1993-7](#)
6. **Agile Retrospectives** - Esther Derby & Diana Larsen / ISBN: [0-9776166-4-9](#)

