Vadå Agile?

SAST Q2

2007-04-24

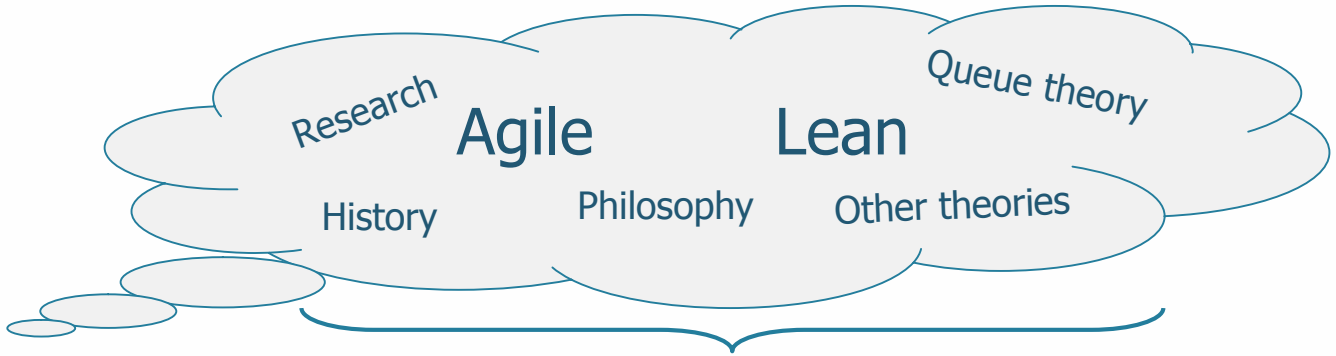
Henrik Kniberg

Who am I?

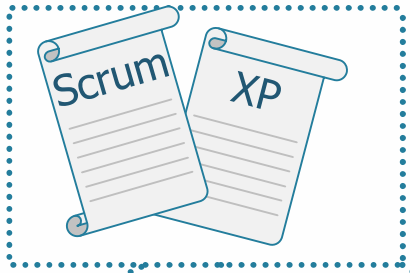
- **Henrik Kniberg**
  - 070 4925284
  - henrik.kniberg@crisp.se
- **Consultant & partner at Crisp AB (www.crisp.se)**
- **Java & Agile software development**

- **Goyada 1998 – 2003**
  - Co-founder & CTO
  - Experimented with XP
  - 30 developers
- **Ace Interactive 2004-2005**
  - System design team lead
  - Experimented with TDD
  - 20 developers
- **Tain 2006 – 2007**
  - Chief of development
  - Experimented with Scrum & XP
  - 40 developers
  - Wrote ”Scrum and XP from the Trenches”

Agenda

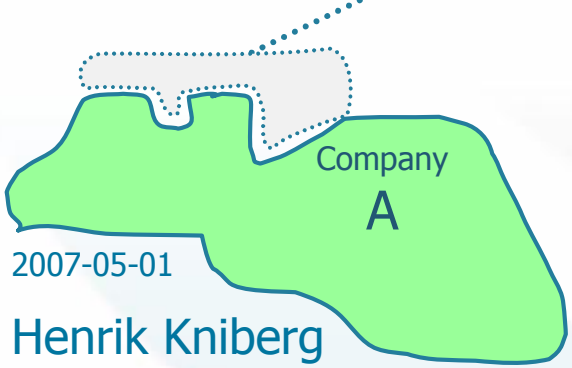


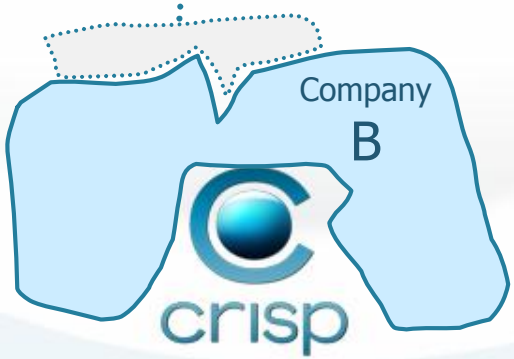
1 Principals

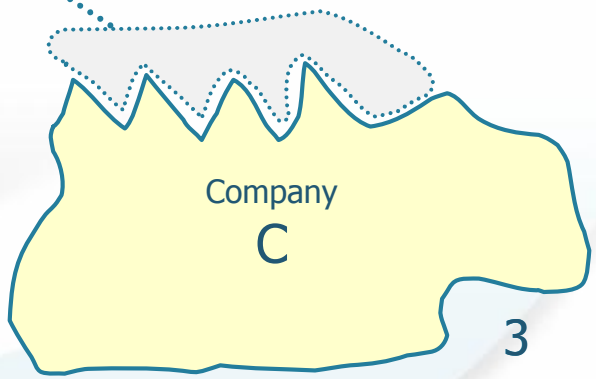


2 Practices

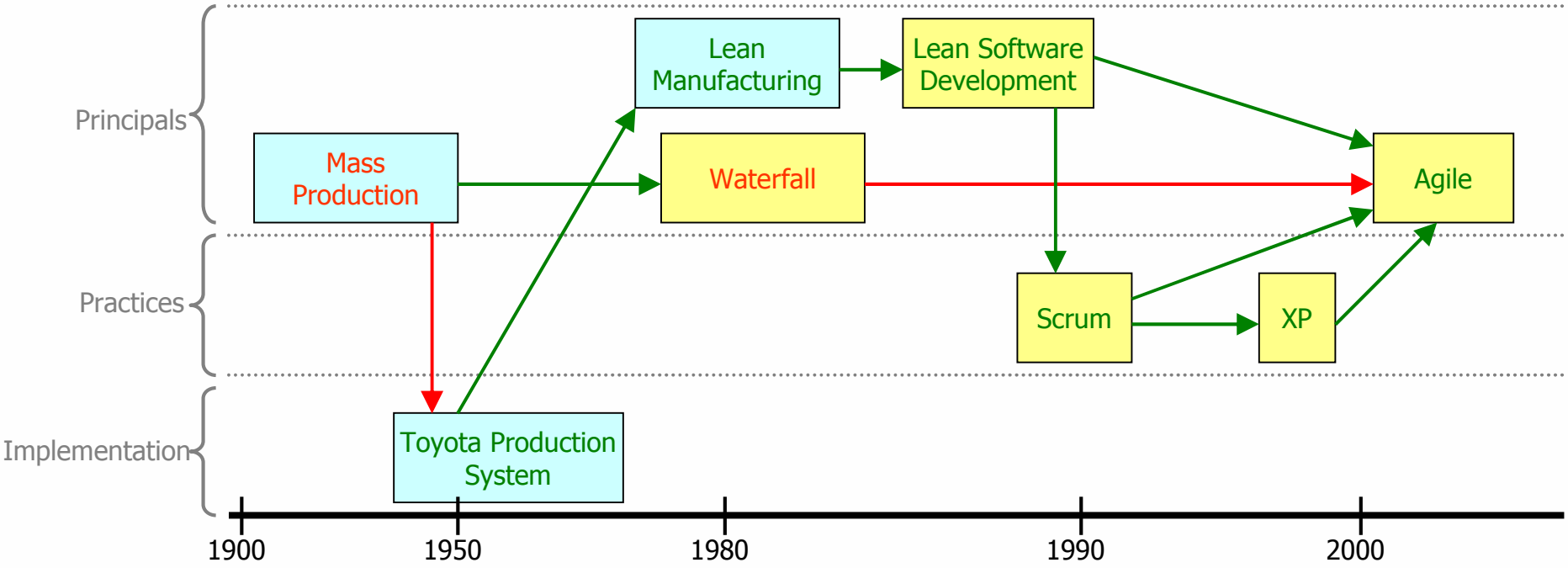
(2) Implementation







History


Principals
Practices
Implementation
Lean Manufacturing
Lean Software Development
Mass Production
Waterfall
Agile
Scrum
XP
Toyota Production System
1900
1950
1980
1990
2000

A short history of XP

From: Kent Beck
To: Jeff Sutherland <jsutherland>
Reply: 70761.1216@compuserve.com
Date: Mon, 15 May 1995 18:01:15 -0400 (EDT)
Subj: HBR paper

Is there a good place to get reprints of the SCRUM paper from HBR? I've written patterns for something very similar and I want to make sure I steal as many ideas as possible.
Kent

Agile Manifesto

www.agilemanifesto.org

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**

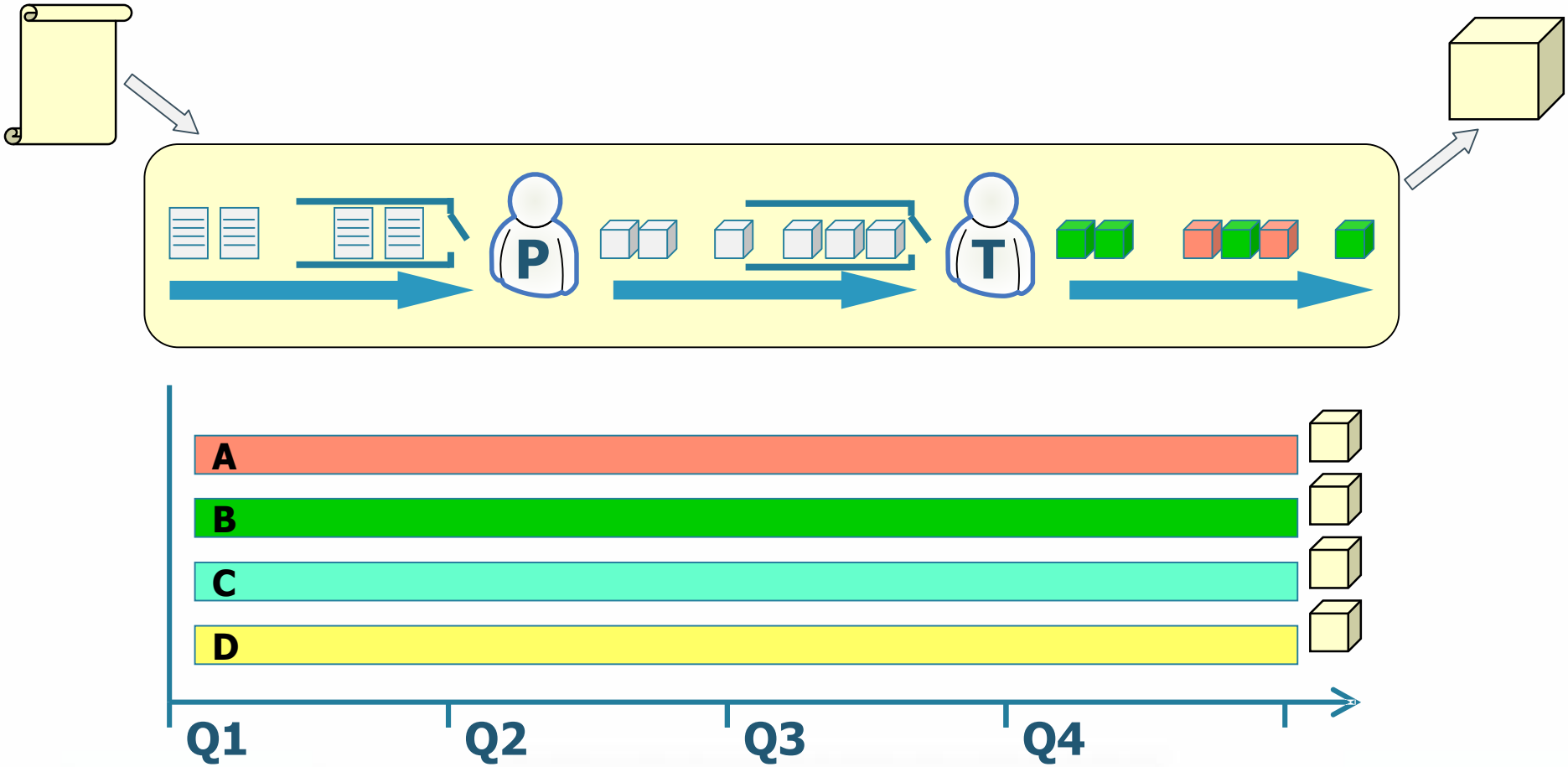
Working software over **comprehensive documentation**

Customer collaboration over **contract negotiation**

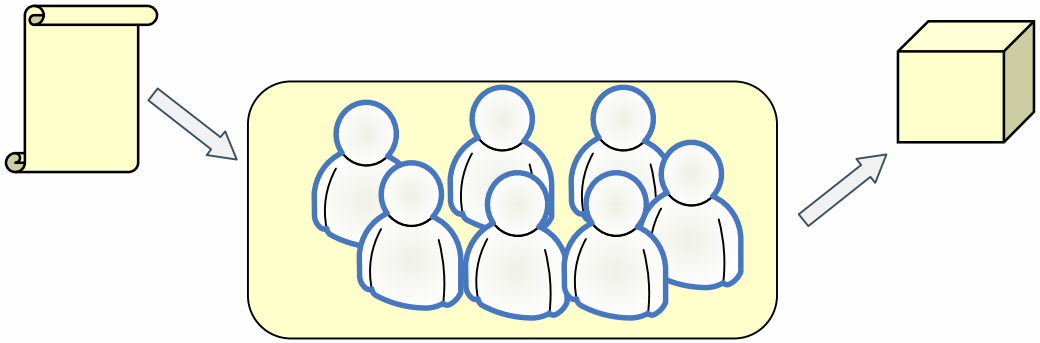
Responding to change over **following a plan**

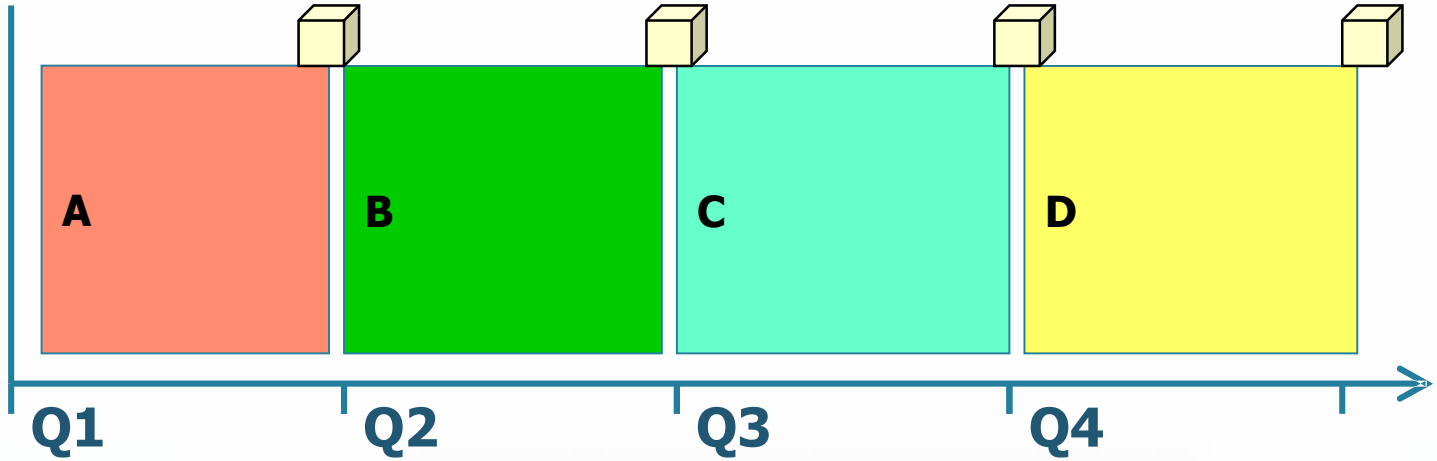
That is, while there is value in the items on
the right, we value the items on the left more.

Traditional company: Resource optimization



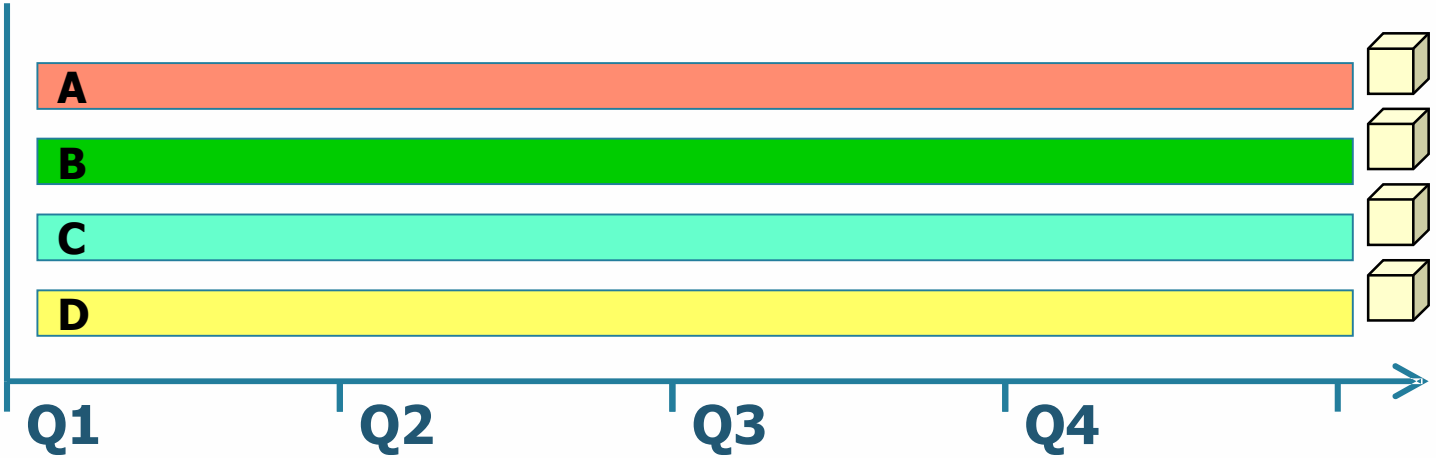
Agile company: Time-to-market optimization



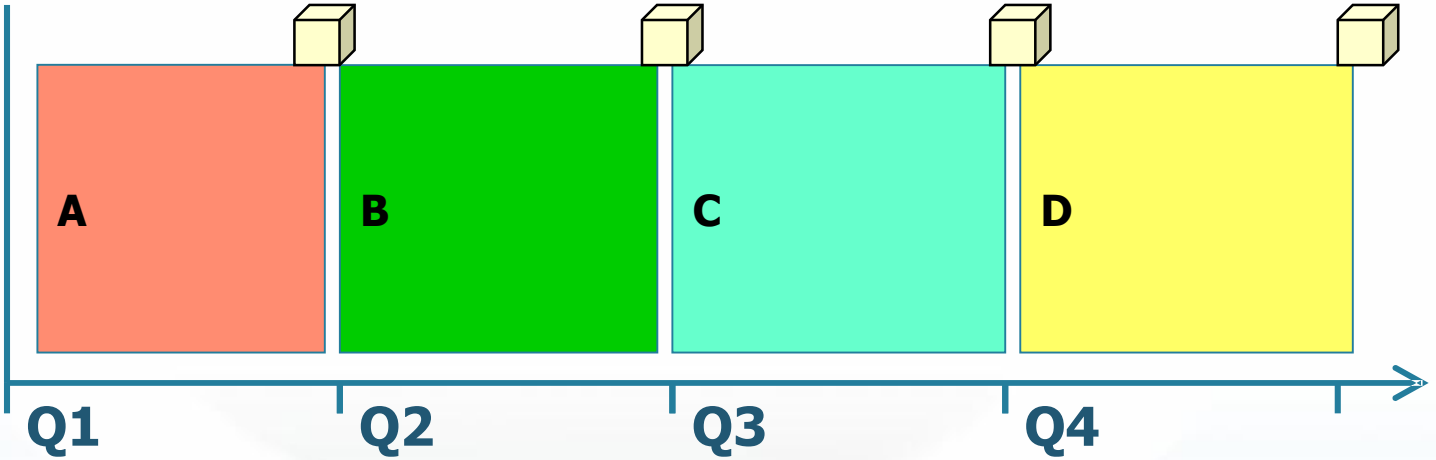


Comparison

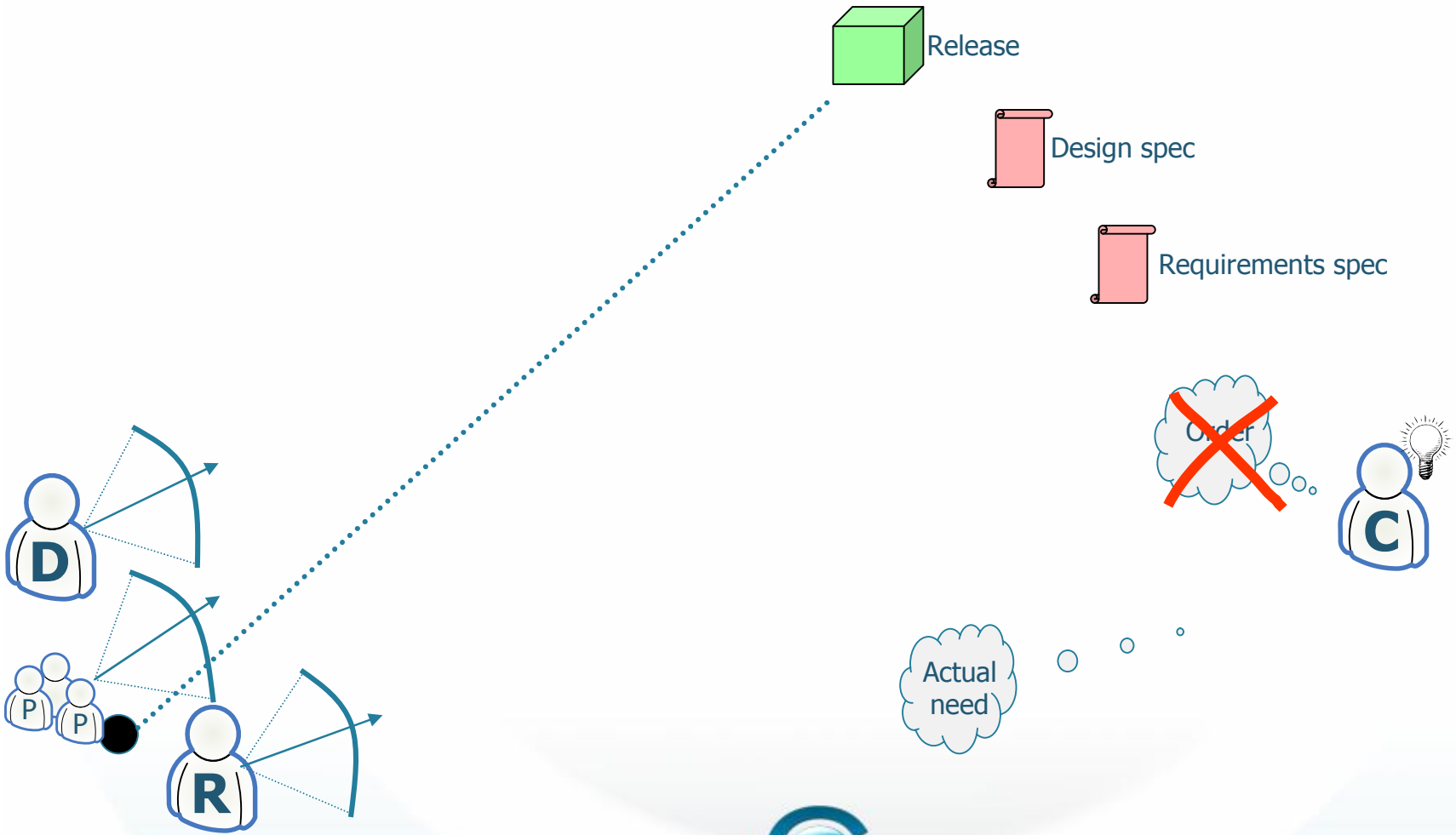
Resource optimization



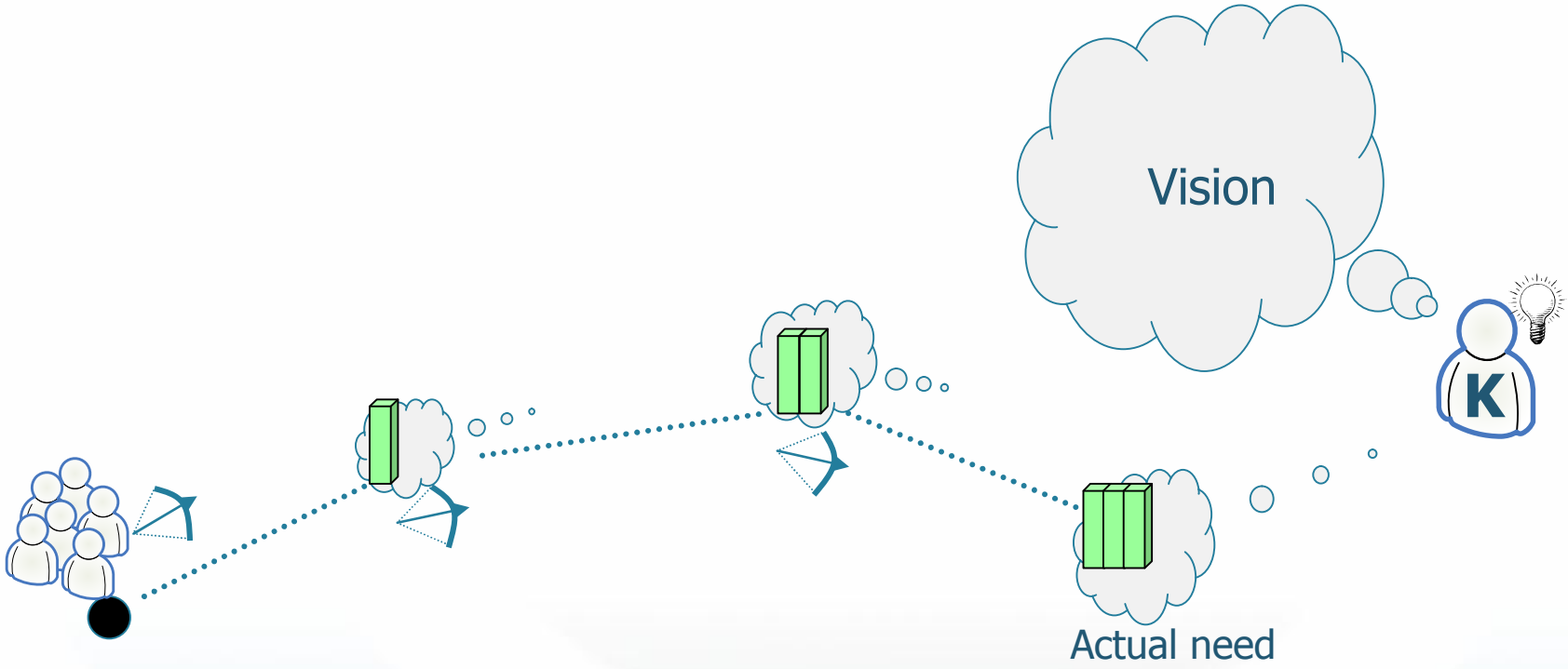
Time-to-market optimization



Predictive approach



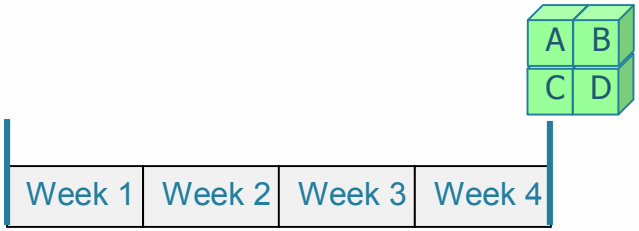
Adaptive approach



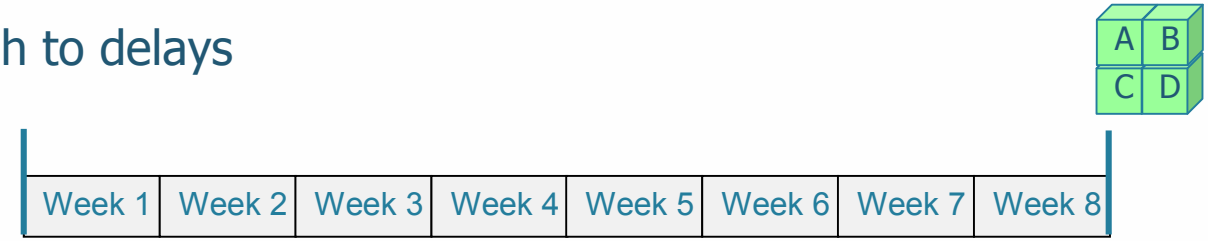
Handling delays

Placing the right decisions into the right hands

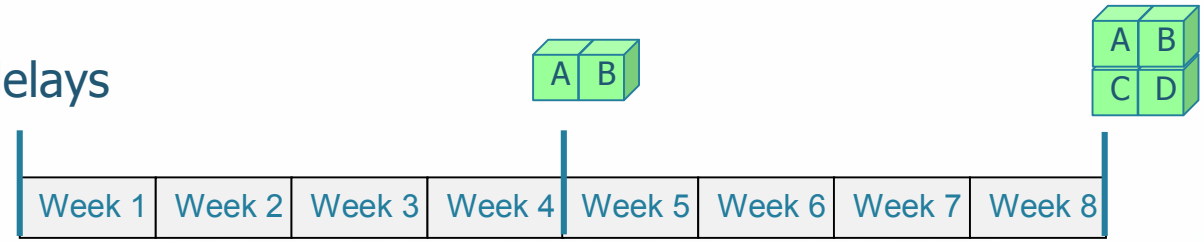
The Plan



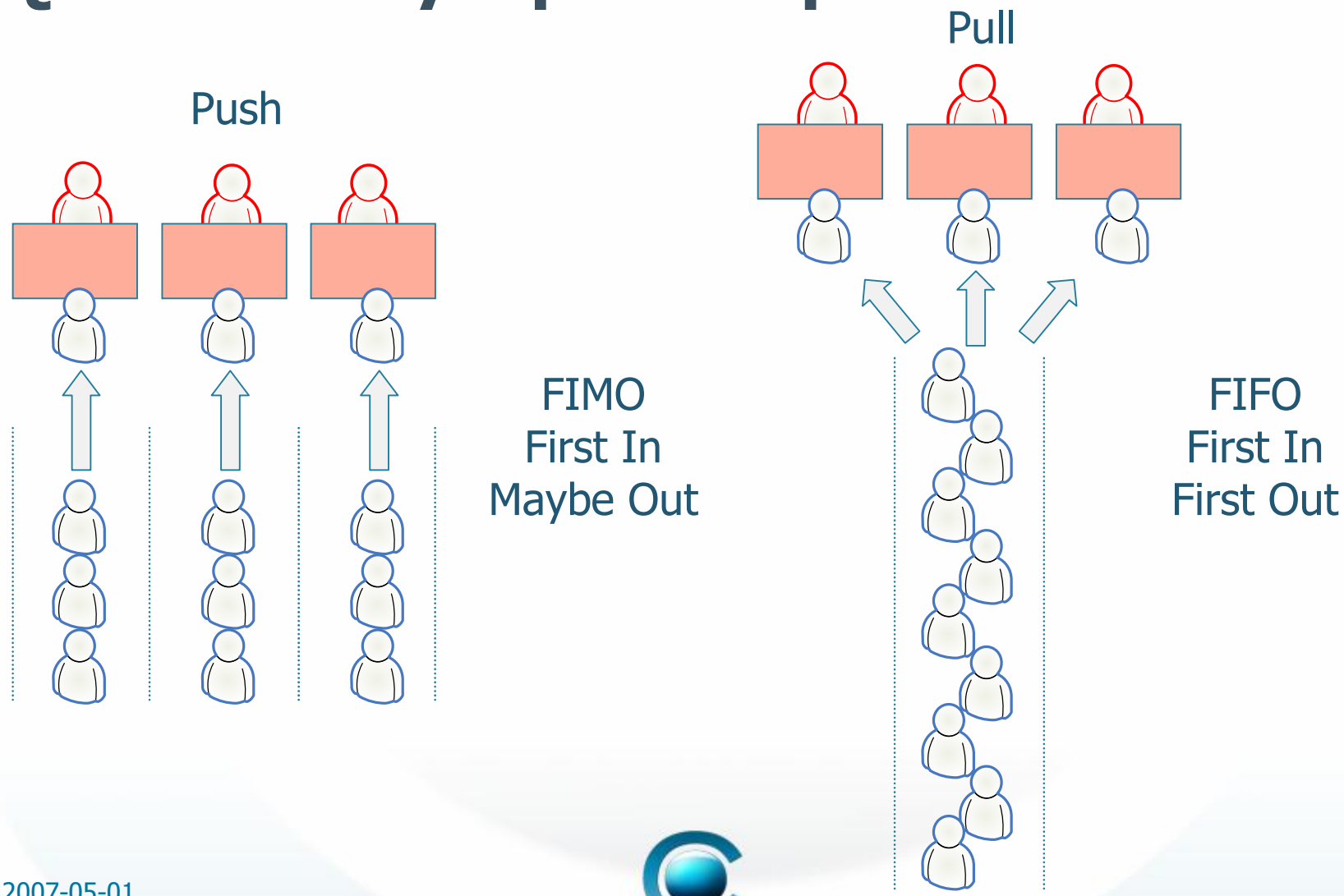
Traditional approach to delays



Agile approach to delays

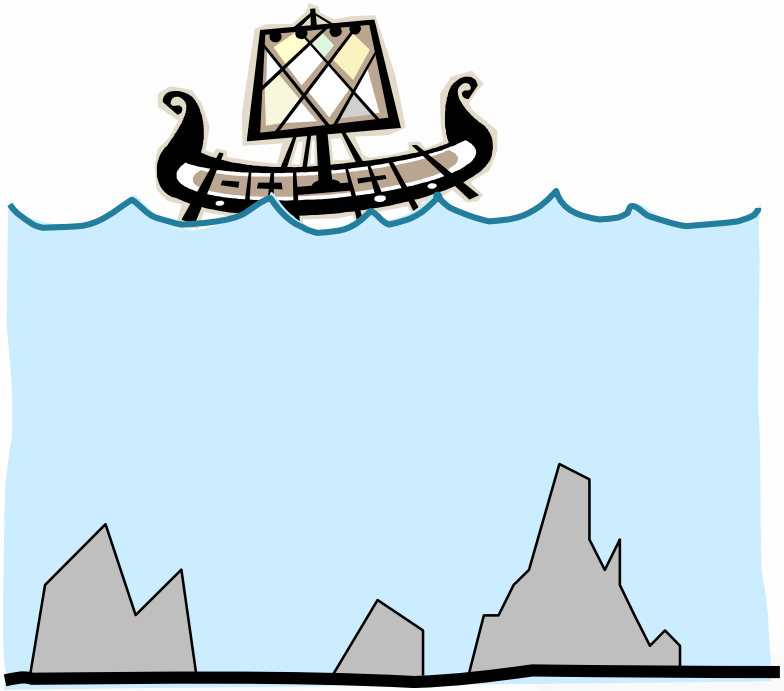


Queue theory – push vs pull

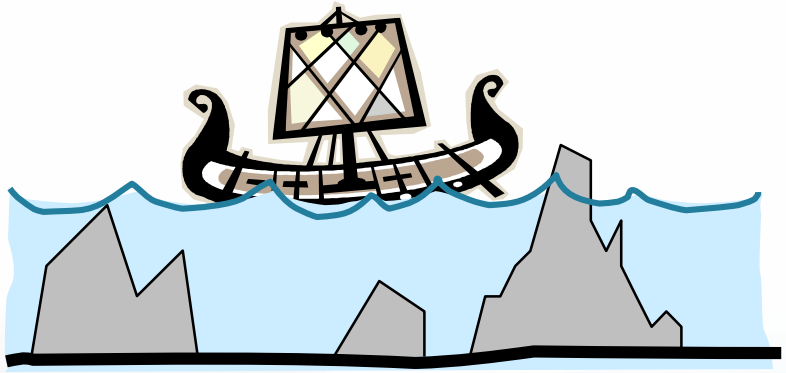


Continuous improvement

Traditional strategy:
Avoid problems



Agile strategy:
Find and remove problems



Lean principals

Principals:

- Eliminate waste
- Focus on learning
- Build quality in
- Defer commitment
- Deliver fast
- Respect people
- Optimize the whole

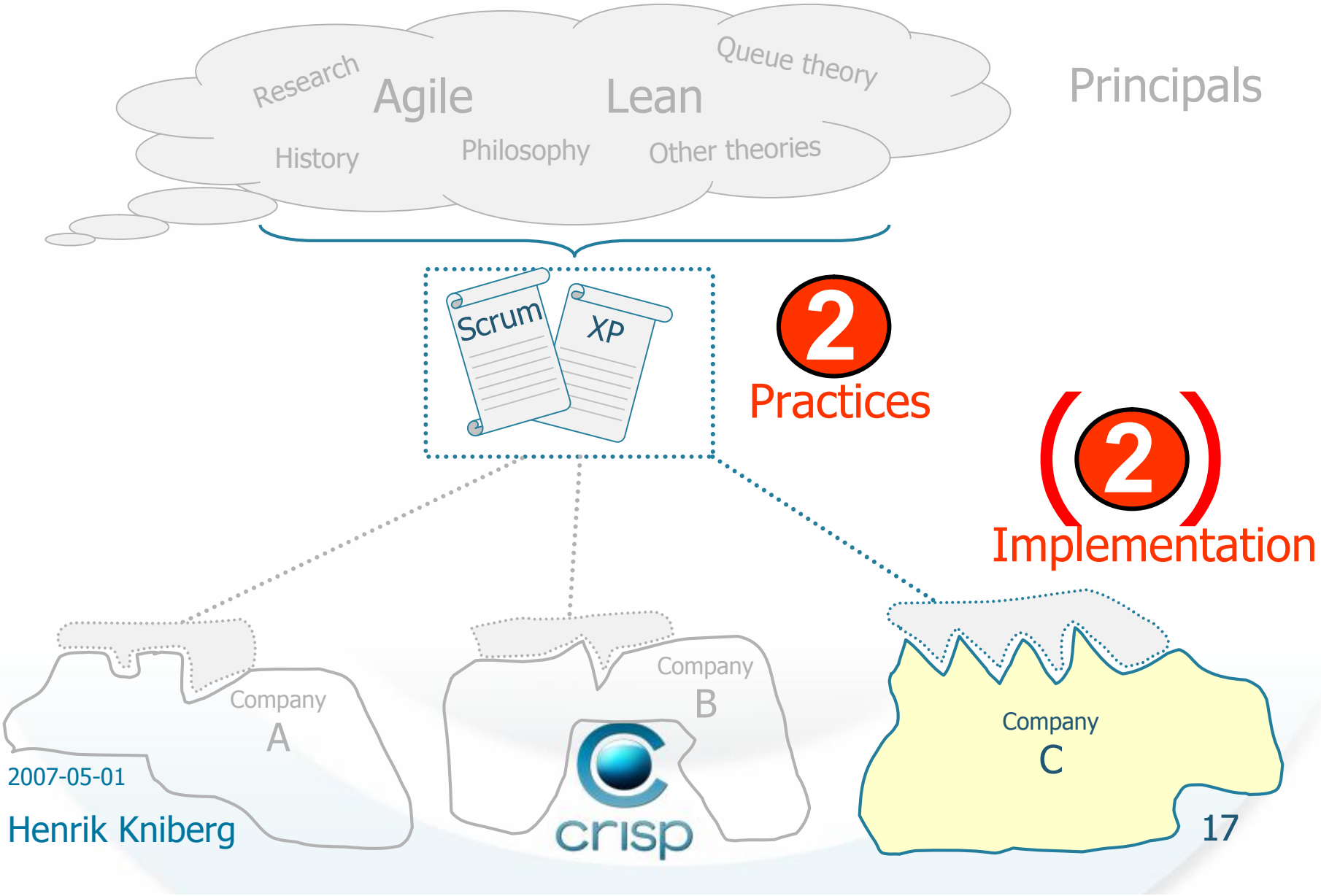
Result:

- High productivity
- Rapid response
- Superior quality
- Lasting value

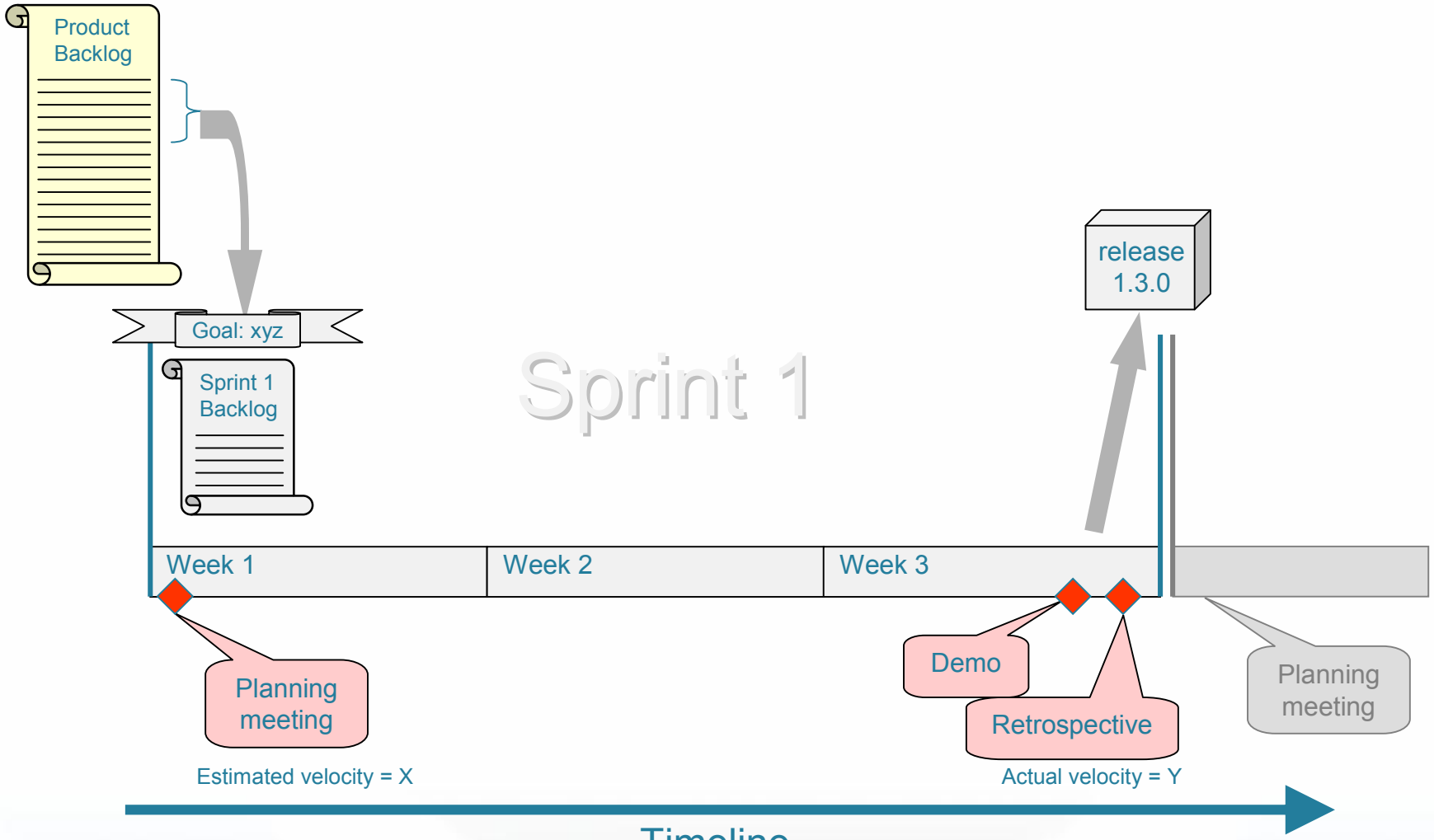
Scrum principles

- Transparancy
- Short feedback loop
- Strict prioritization
- Continuous improvement
- Self-organizing team
- Timeboxing
- Face-to-face communication
- Simple tools
- Frequent & regular delivery of whole system
- Plans are needed, but they are always wrong
- Yesterday's weather

Part 2 - Practices



Scrum overview



Scrum practices

2 artifacts

- Product backlog
- Sprint backlog

3 roles

- Product owner
- Scrum master
- Team

4 activities

- Sprint planning meeting
- Daily scrum
- Sprint demo
- Sprint retrospective

Product owner

Sets priorities
Represents all stakeholders
Not the boss!

Rights

- **Continuously update priorities & requirements**
  - Changes affect *next* sprint
- **Receive time estimates**
- **See sprint progress**
- **New release after each sprint**

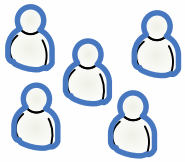
Responsibilities

- **Dedicated**
- **Maintain product backlog**
- **Continuously update priorities & requirements**
- **Partipicate**
  - Spring planning
  - Sprint demo
  - Sprint retrospective
  - (Daily scrum)
- **Be available to team**
- **Don’t disturb ongoing sprint**

Scrum master

Removes impediments
Enforces Scrum practices
Not the boss!

- **Part of the team**
- **Examples of scrum master tasks:**
  - Lead sprint planning meetings, sprint demos, sprint retrospectives
  - Announce sprint demos
  - Ensure that sprint backlog is maintained
  - Help product owner maintain product backlog & make business tradeoffs
  - Ensure continuous improvement
  - Ensure sprint plan is updated
    - add/remove stories when necessary
- **What the scrum master is not:**
  - Scrum master is not team leader or project leader.
  - Everyone on the team shares equal responsibility for the sprint goal. Scrum master has no more responsibility than anyone else.
  - Scrum master's role is not to organize the team. Team is primarily self-organizing.



Team

- **5 – 9 full-time individuals**
- **Self-organizing**
- **Shared responsibility for reaching sprint goal**
- **Works as team, rather than set of individuals**
- **Cross-functional**
  - Contains all competence needed to reach goal
- **Sits together**

Team member

Rights

- Work in peace during sprint
- Direct access to product owner to clarify stories
- Work however he likes to achieve the goal
- Not create unnecessary artifacts
- Create own time estimates
- Do one thing at a time
- Choose how much to commit to

Responsibilities

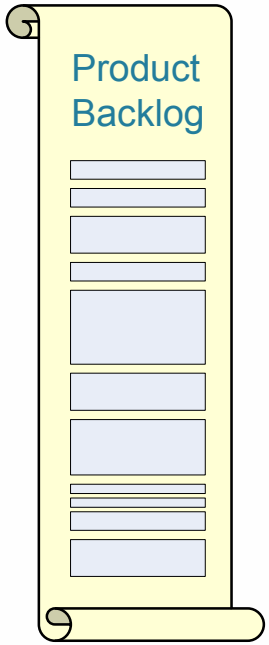
- Attend daily scrum, sprint planning, sprint demo, sprint retrospective
- Update time estimates continuously
- Follow the product owner's priorities
- Report impediments
- Continuously reduce or increase scope of sprint as necessary, to ensure realistic expectations.

Product backlog

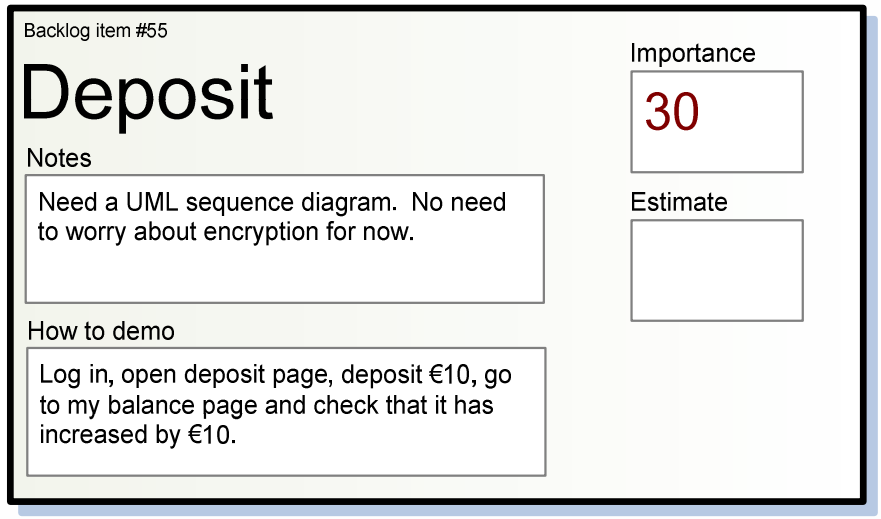
Contains "User Stories"

Each story has:

- **Priority**
- **Business value**
- **Deliverable**
- **Estimate (set by team)**



Product backlog - sample



ID	Name	Imp	Est	How to demo	Notes
1	Deposit	30	5	Log in, open deposit page, deposit €10, go to my balance page and check that it has increased by €10.	Need a UML sequence diagram. No need to worry about encryption for now.
2	See your own transaction history	10	8	Log in, click on “transactions”. Do a deposit. Go back to transactions, check that the new deposit shows up.	Use paging to avoid large DB queries. Design similar to view users page.

Sprint

- **= iteration**
- **Has a well-defined goal (if possible)**
- **Fixed length (typically 2 – 4 weeks)**
- **Variable scope**
  - There is a plan, but it may be updated during sprint.
  - Only team may update the plan.
- **Results in potentially shippable product increment**
  - = ”ready-to-use” code
- **Important to define what ”done” means for each story**
  - Optimally, ”done” should include ”everything”. Requirements, design, development, test, documentation, installation, etc.
- **Undisturbed**
  - Changes and new requests are deferred to next sprint

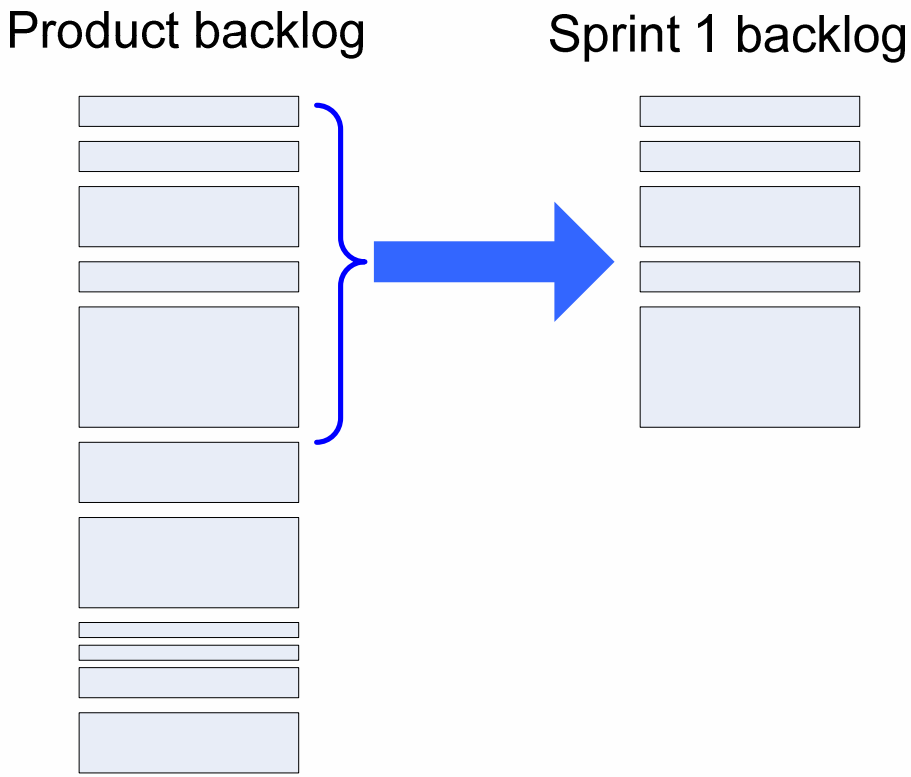
Sprint planning meeting

Purpose:
- Decide what to deliver next sprint

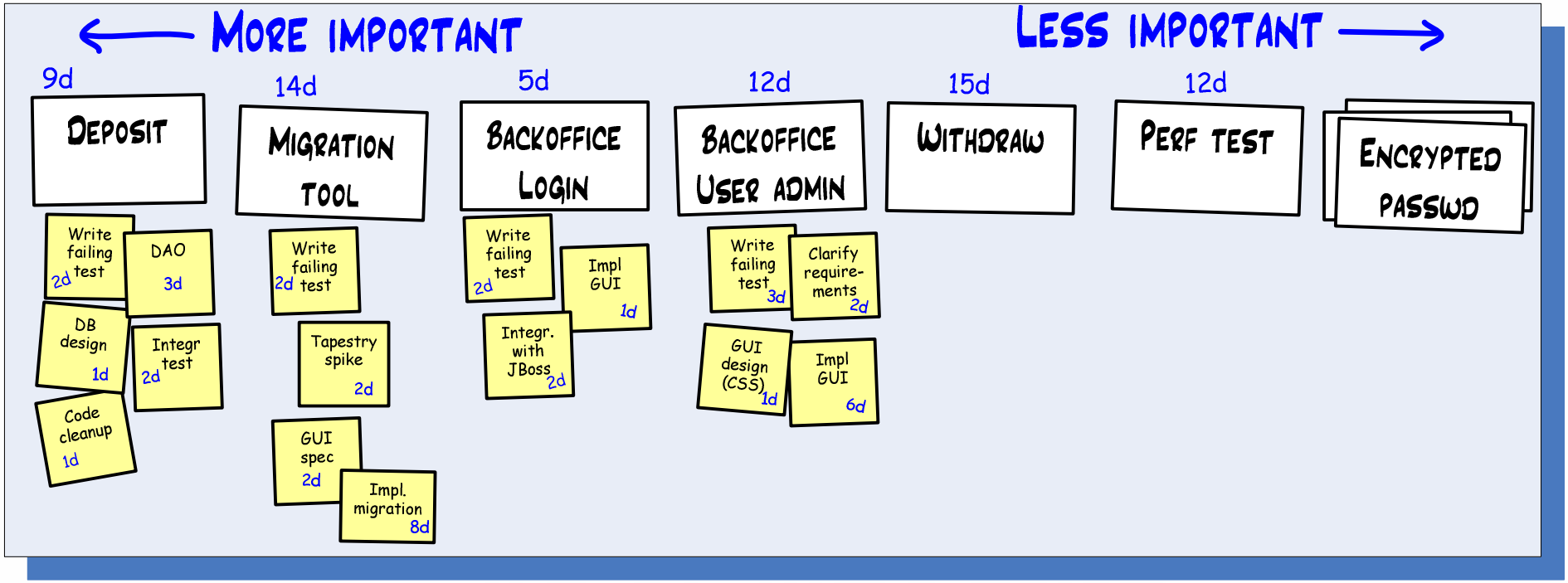
Output of meeting:
- Sprint goal
- Sprint backlog

Activities during meeting:
- Clarify stories
- Define definition of ”done”
- Define sprint demo
- Break apart or combine stories
- Time estimate
- Update priorities
- Calculate resource availabilty
- Decide which stories to include in sprint

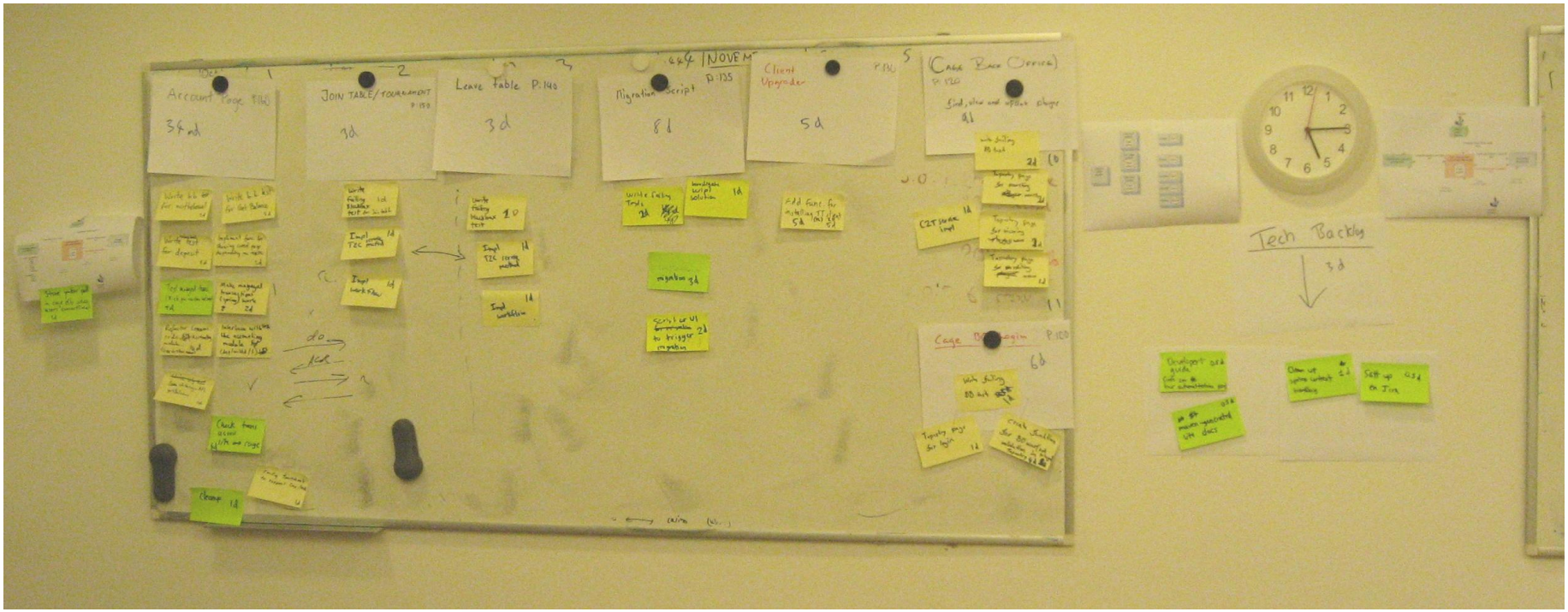
Key responsibilities:
- Team: time estimate
- Product owner: prioritize



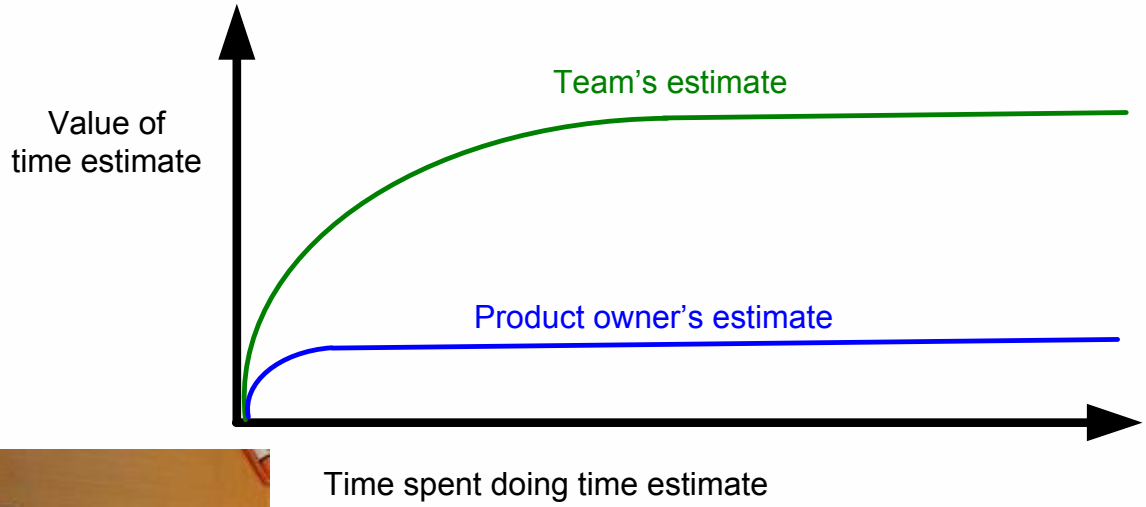
Sprint planning meeting - example

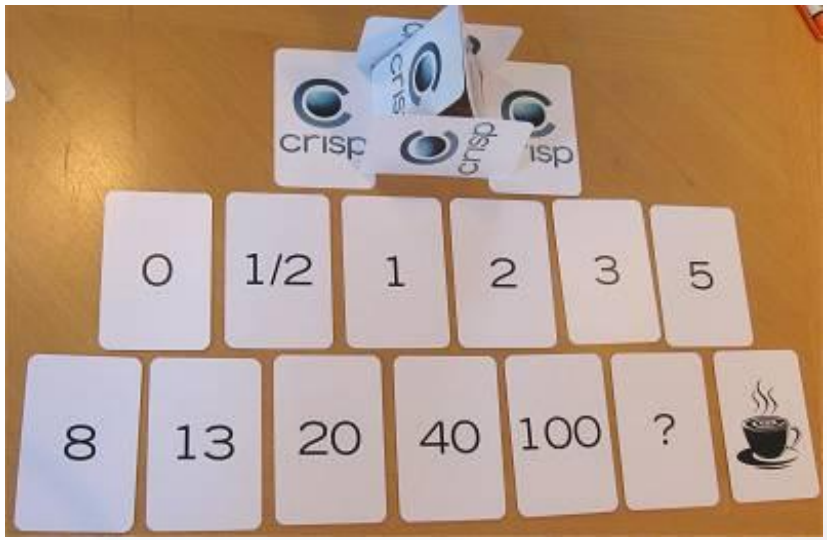


Spring planning meeting - example



Team time estimates & planning poker





The sprint commitment

Team's commitment to the product owner:

- **"We sincerely believe that we can reach the stated sprint goal."**
- **"We promise to do everything in our power to reach the sprint goal, and to let you know immediately if we no longer believe we can reach it."**
- **"We sincerely believe that we can complete all stories included in the sprint backlog".**
- **"We promise to release at the end of the sprint, we will *not* be delayed even a single day."**
- **"We promise that, if we fall behind schedule, we will remove the lower priority stories first."**
- **"We promise that, if we get ahead of schedule, we will add stories to the sprint from the product backlog, in priority order."**
- **"We promise to display our progress and status on a daily basis."**
- **"We promise that every story that we do deliver is *complete.*"**

Common misconceptions

- **"We promise to achieve this goal"**
- **"We promise to deliver all stories included in the sprint backlog"**

Sprint info page - example

Jackass team, sprint 15

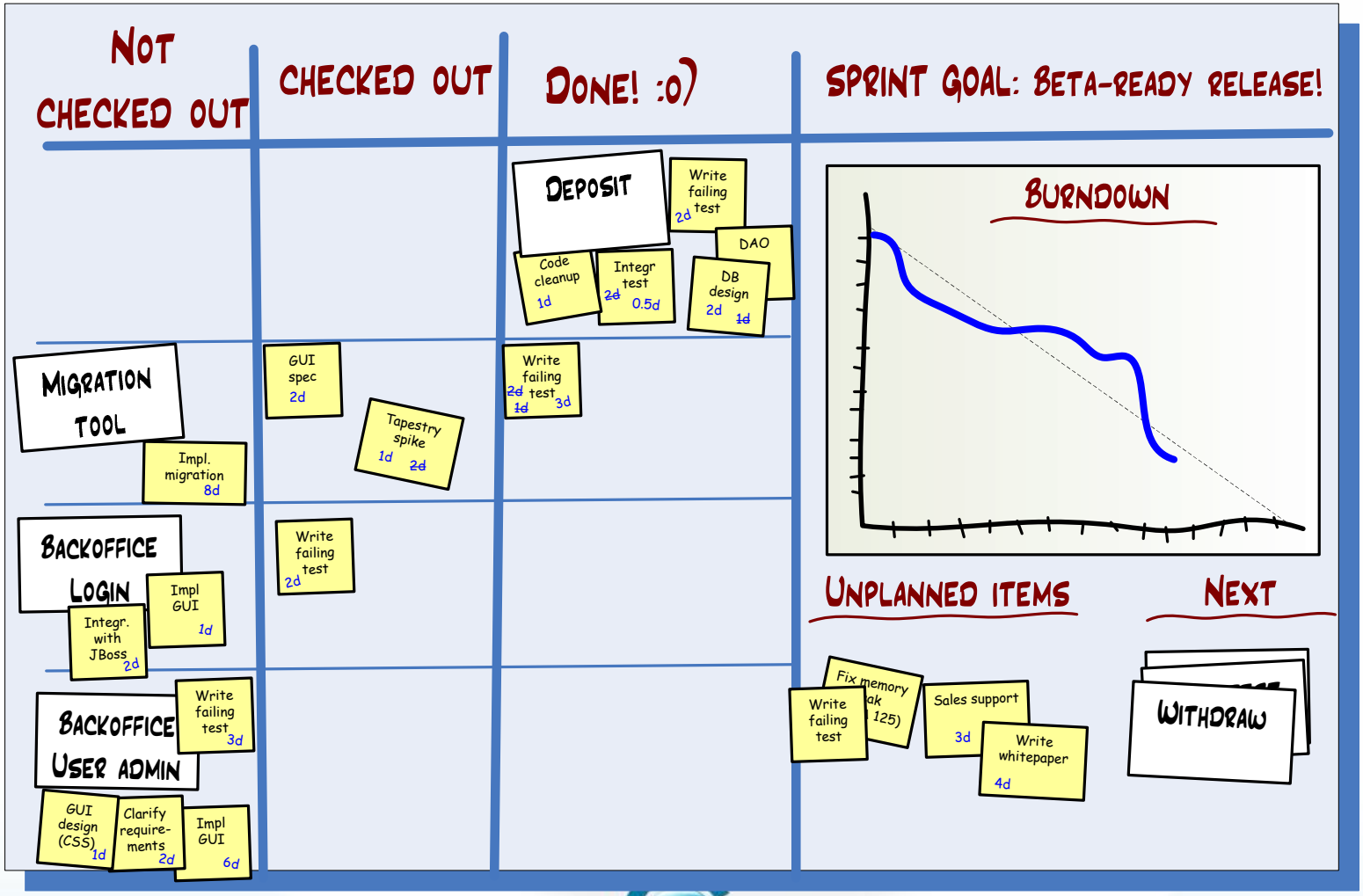
Sprint goal
- Beta-ready release!

Sprint backlog
- Deposit
- Migration tool
- Backoffice login
- Backoffice user admin

Schedule
- Sprint period: 2006-11-06 to 2006-11-24
- Sprint demo: 2006-11-24, 13:00, in the cafeteria
- Daily scrum: 9:30 – 9:45, in conference room Jimbo

Team
- Jim
- Erica (scrum master)
- Tom (75%)
- Niklas
- Eva
- John

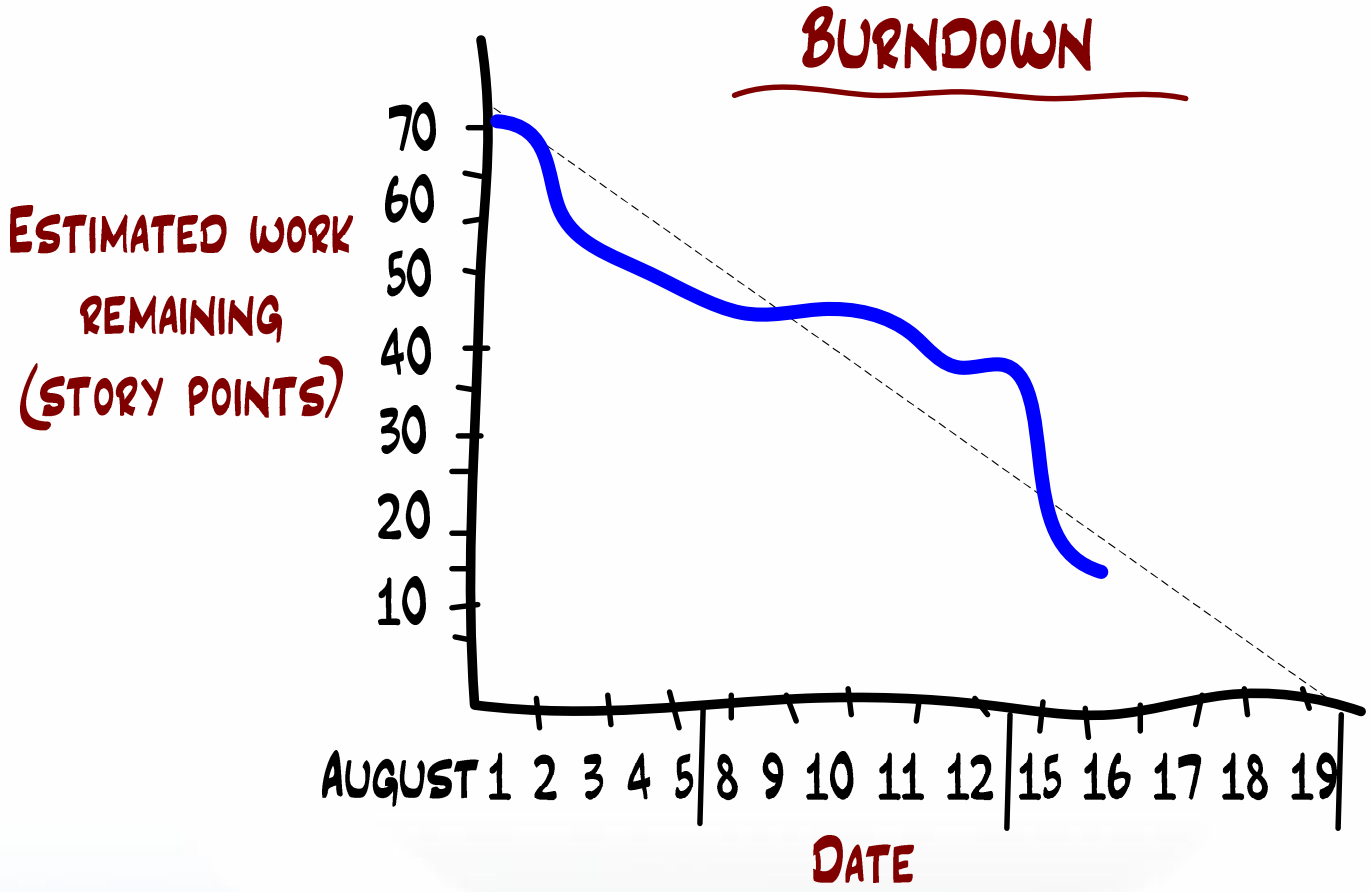
Sprint backlog - example



Daily Scrum meeting

15 minutes
- **What did I accomplish yesterday?**
- **What will I accomplish today?**
- **What's stopping me?**

Sprint burndown chart



Sprint demo (a.k.a Sprint review)

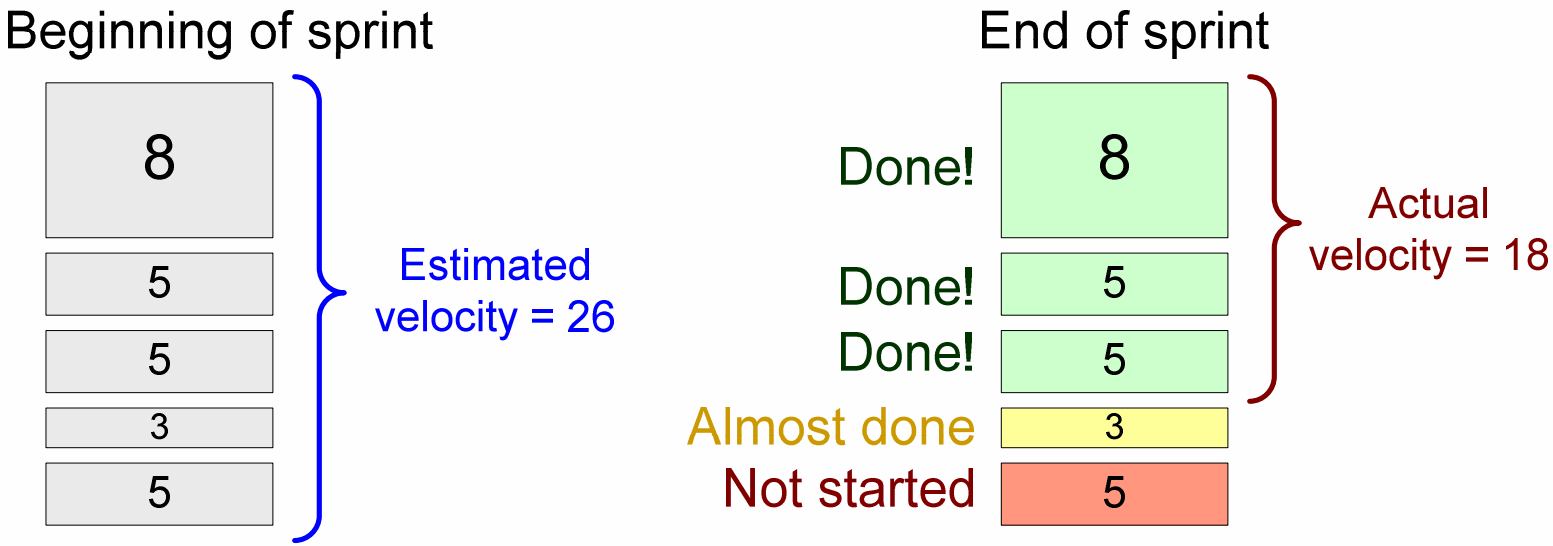
What have we accomplished?

- **Team demonstrates working code to stakeholders**
- **Only 100% completed stories are demonstrated**
  - Partially completed stories ignored
- **Direct feedback from stakeholders**
- **Feedback incorporated into product backlog**

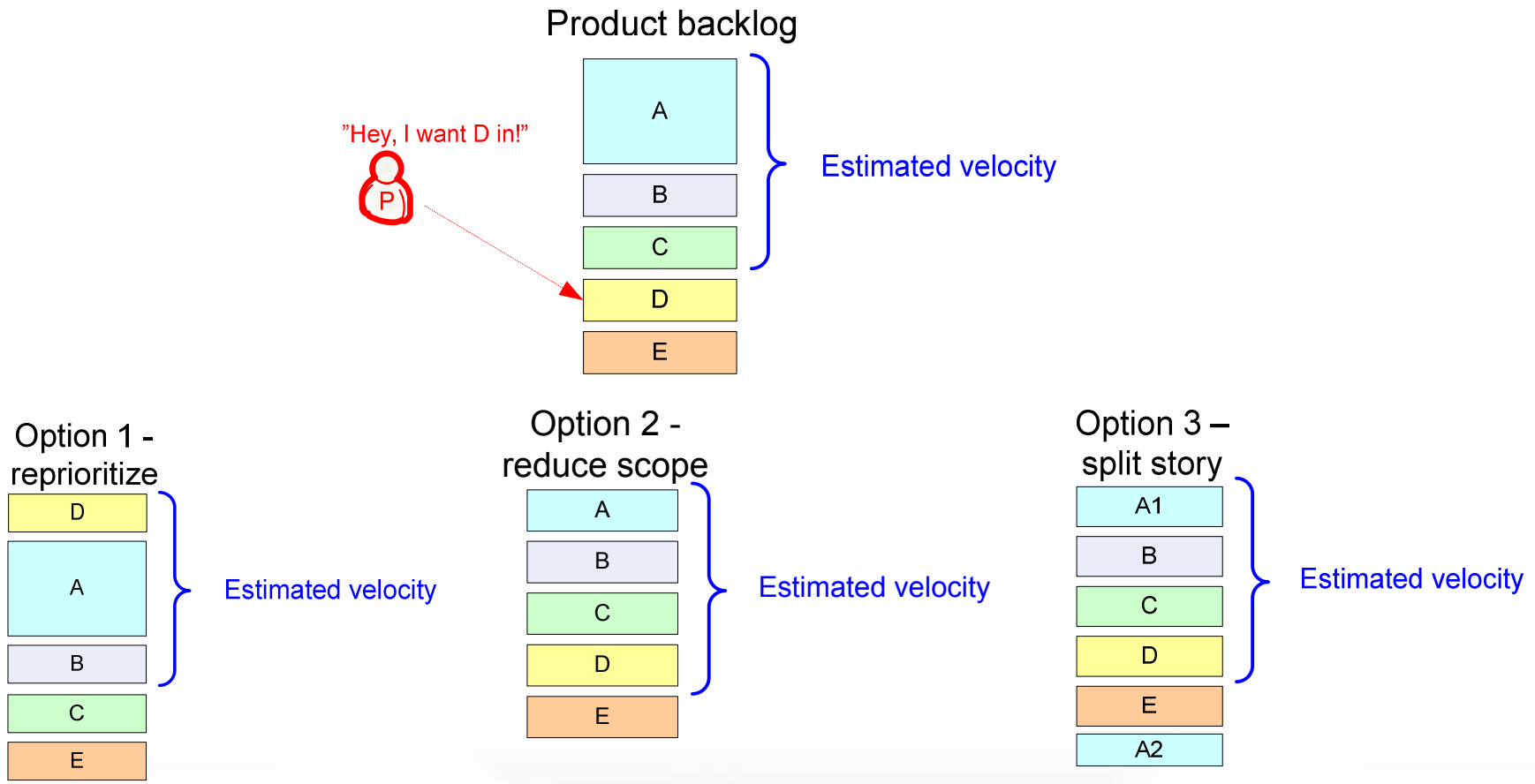
Sprint retrospective

What do we do differently next sprint?

Velocity tracking



Planning using velocity



Release planning

Imp	Name	Estimate
130	banana	12
120	apple	9
115	orange	20
110	guava	8
100	pear	20
95	raisin	12
80	peanut	10
70	donut	8
60	onion	10
40	grapefruit	14
35	papaya	4
10	blueberry	
10	peach	

Estimated velocity = 45

Imp	Name	Estimate
		Sprint 1
130	banana	12
120	apple	9
115	orange	20
		Sprint 2
110	guava	8
100	pear	20
95	raisin	12
		Sprint 3
80	peanut	10
70	donut	8
60	onion	10
40	grapefruit	14
		Sprint 4
35	papaya	4
10	blueberry	
10	peach	

Calculating velocity

(when there is no reliable "yesterday's weather")

	Available days
Tom	15
Lisa	13
Sam	15
Dave	7
	50 available man-days

Last sprint's focus factor:

$$\text{(focus factor)} = \frac{\text{(actual velocity)}}{\text{(available man-days)}}$$

This sprint's estimated velocity:

(available man-days) x (focus factor) = (estimated velocity)

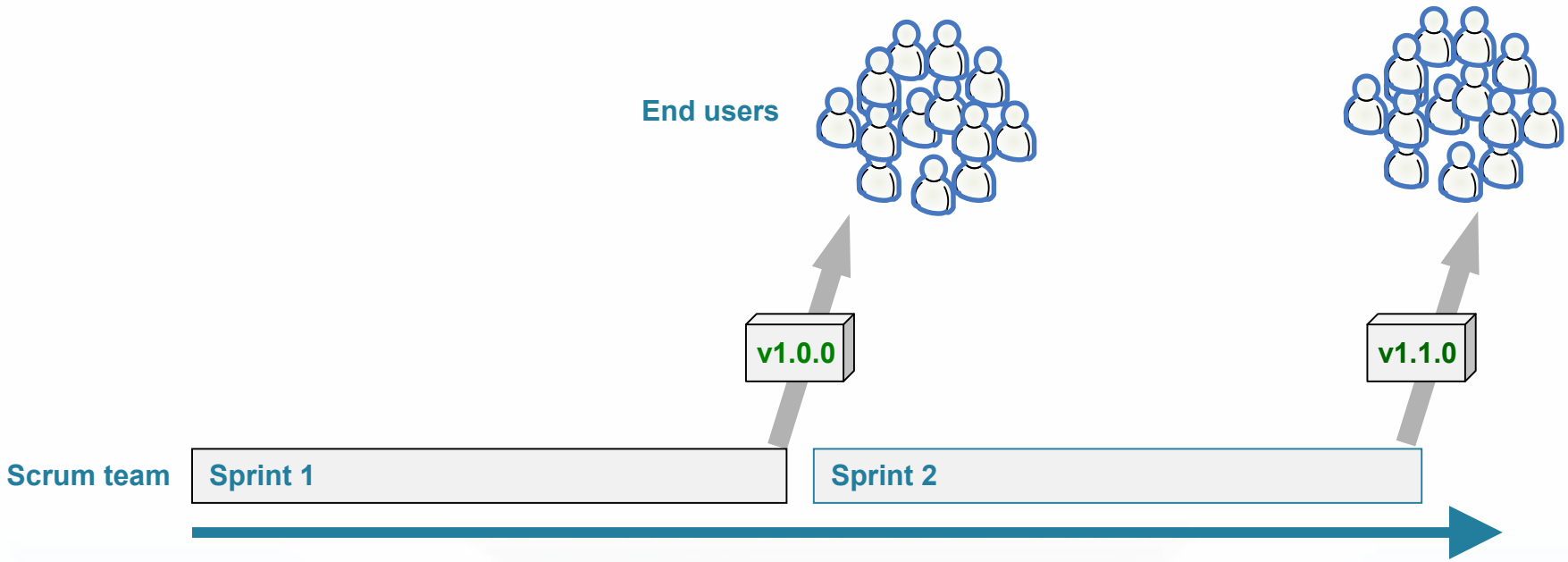
Last sprint's focus factor:

$$40\% = \frac{18 \text{ story points}}{45 \text{ man-days}}$$

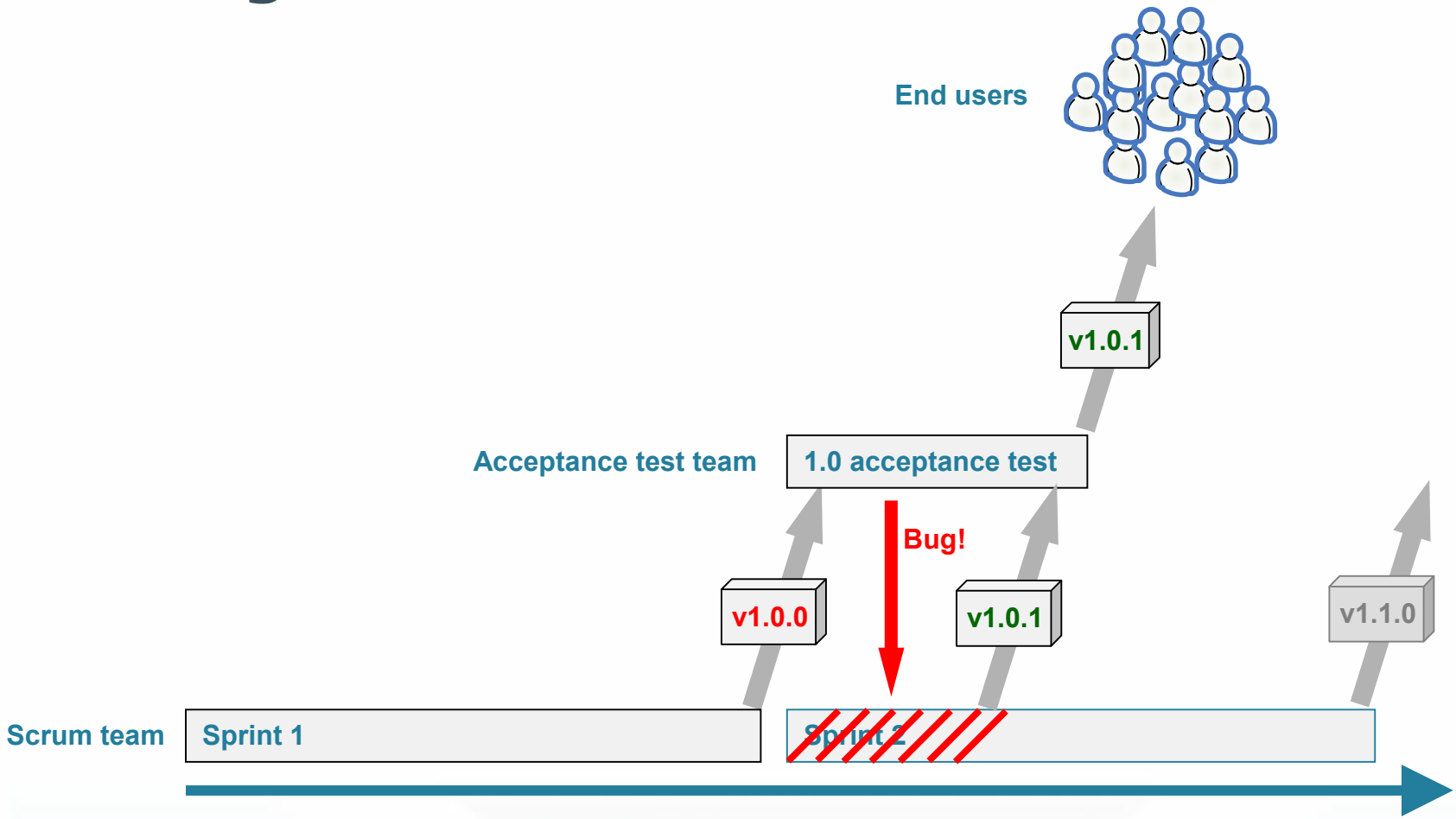
This sprint's estimated velocity:

50 man-days x 40% = 20 story points

Testing – ideal case

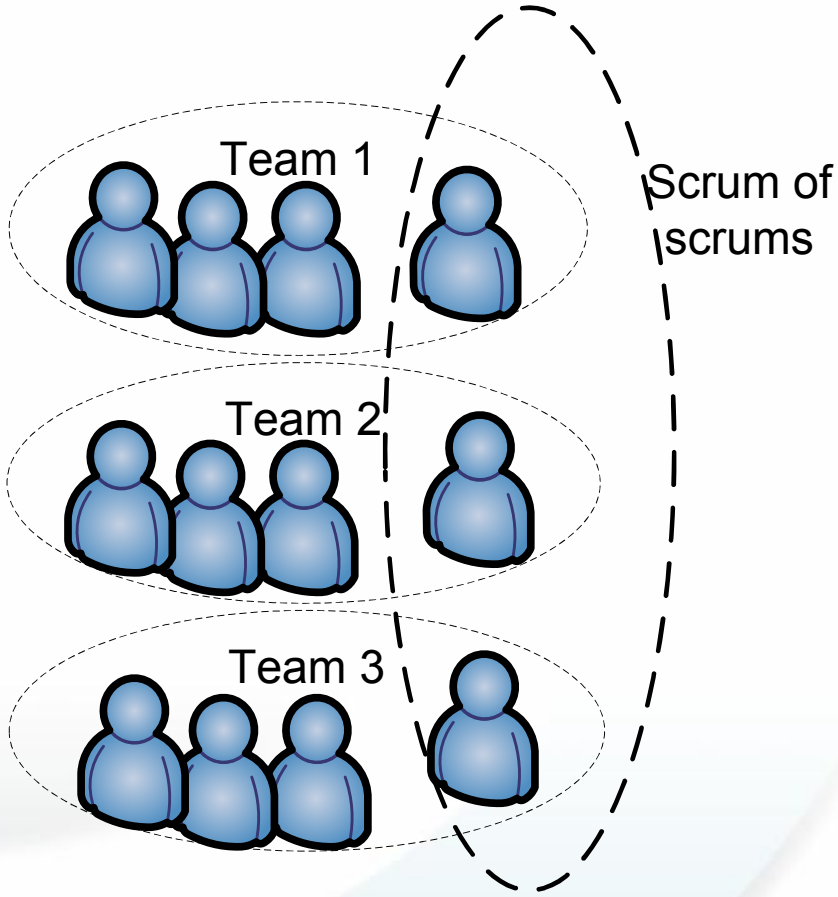


Testing – common alternative

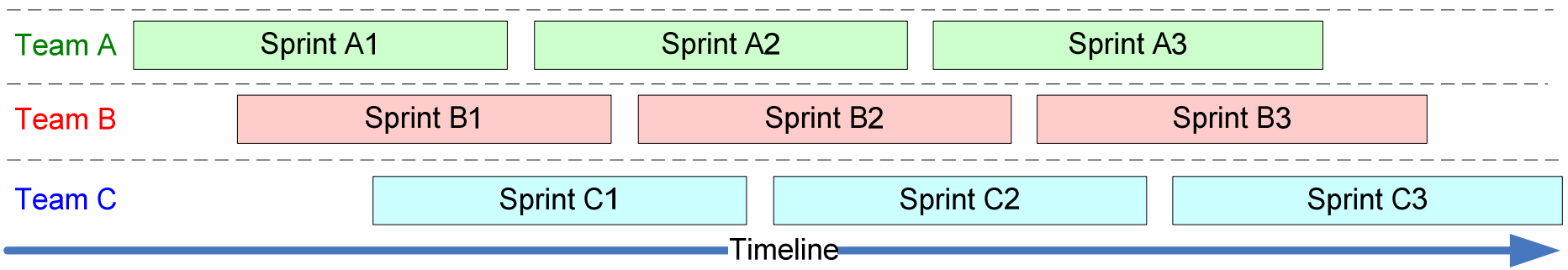


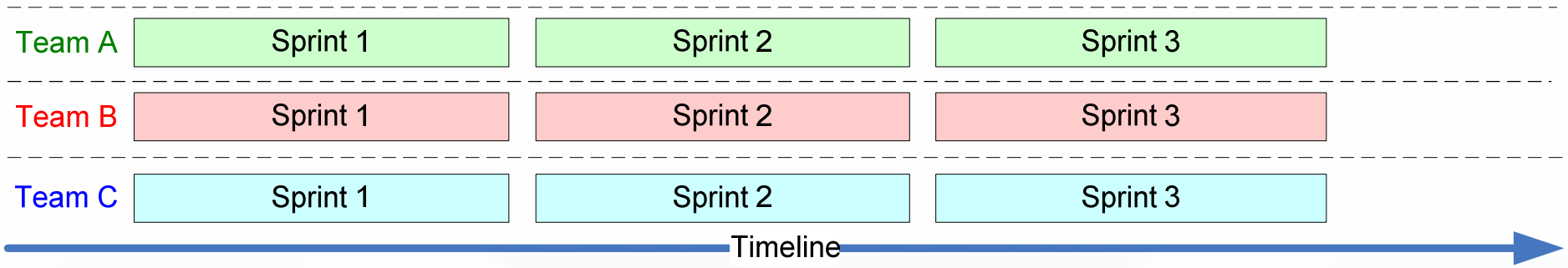
Scaling

- Start with single ”seed team” that creates baseline architecture, prototypes, and development environment.
- Every ”seed team” member spans a new team.
- Scrum of Scrums = virtual scrum team responsible for integration & coordination



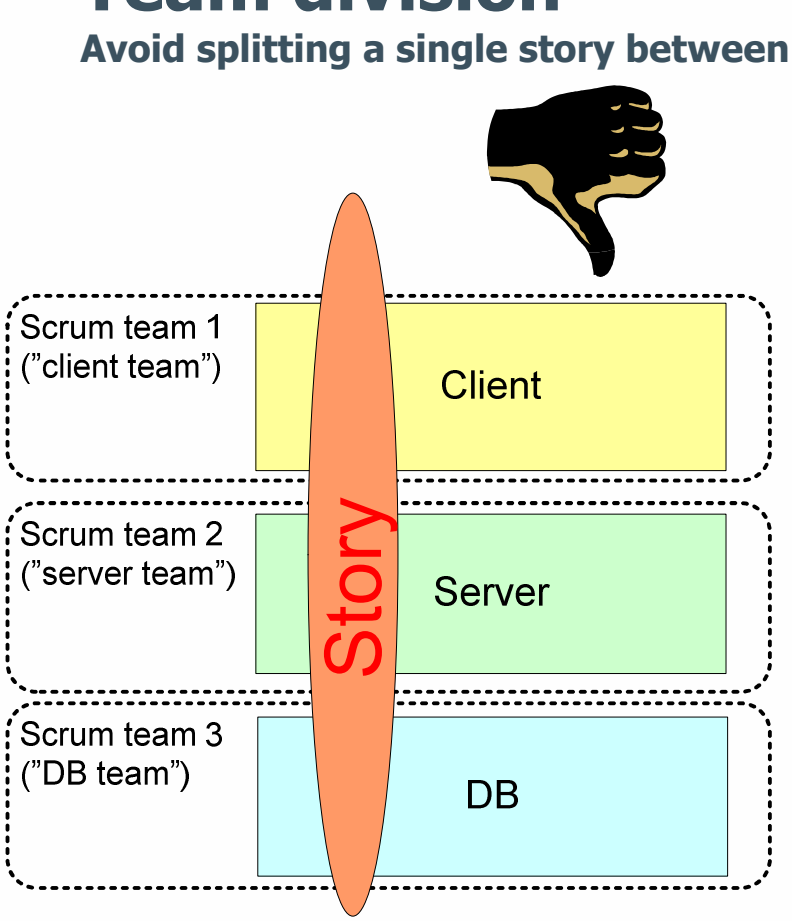
Simultaneous sprints

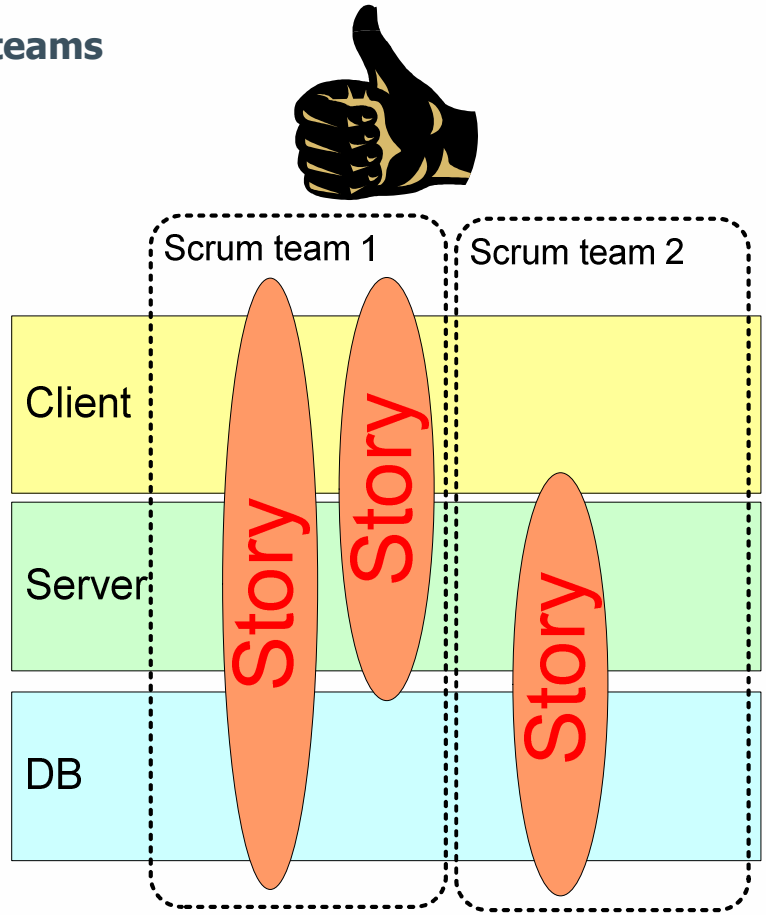




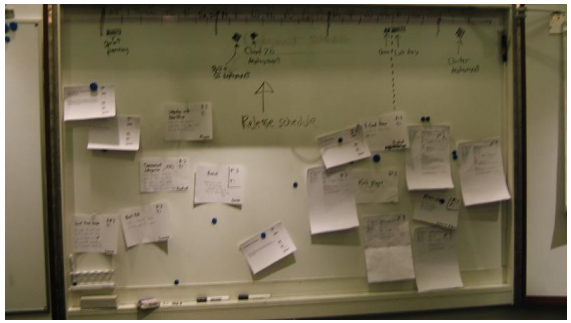
Team division

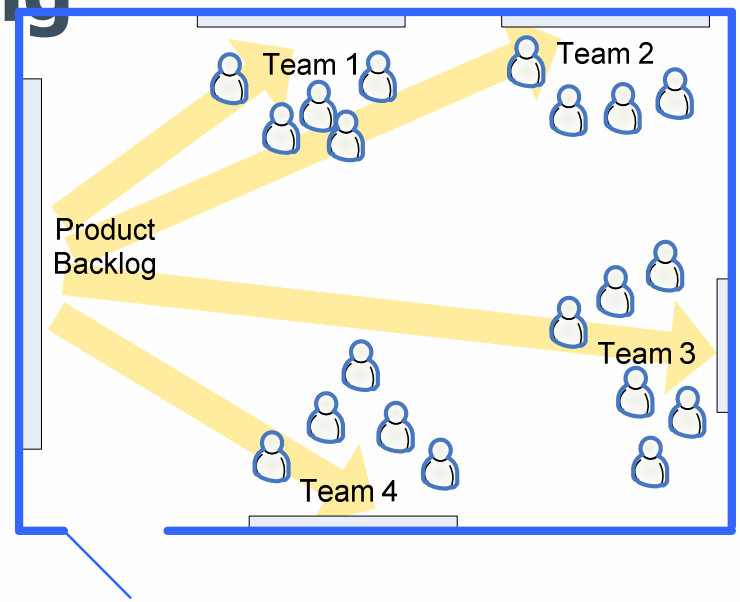
Avoid splitting a single story between multiple teams



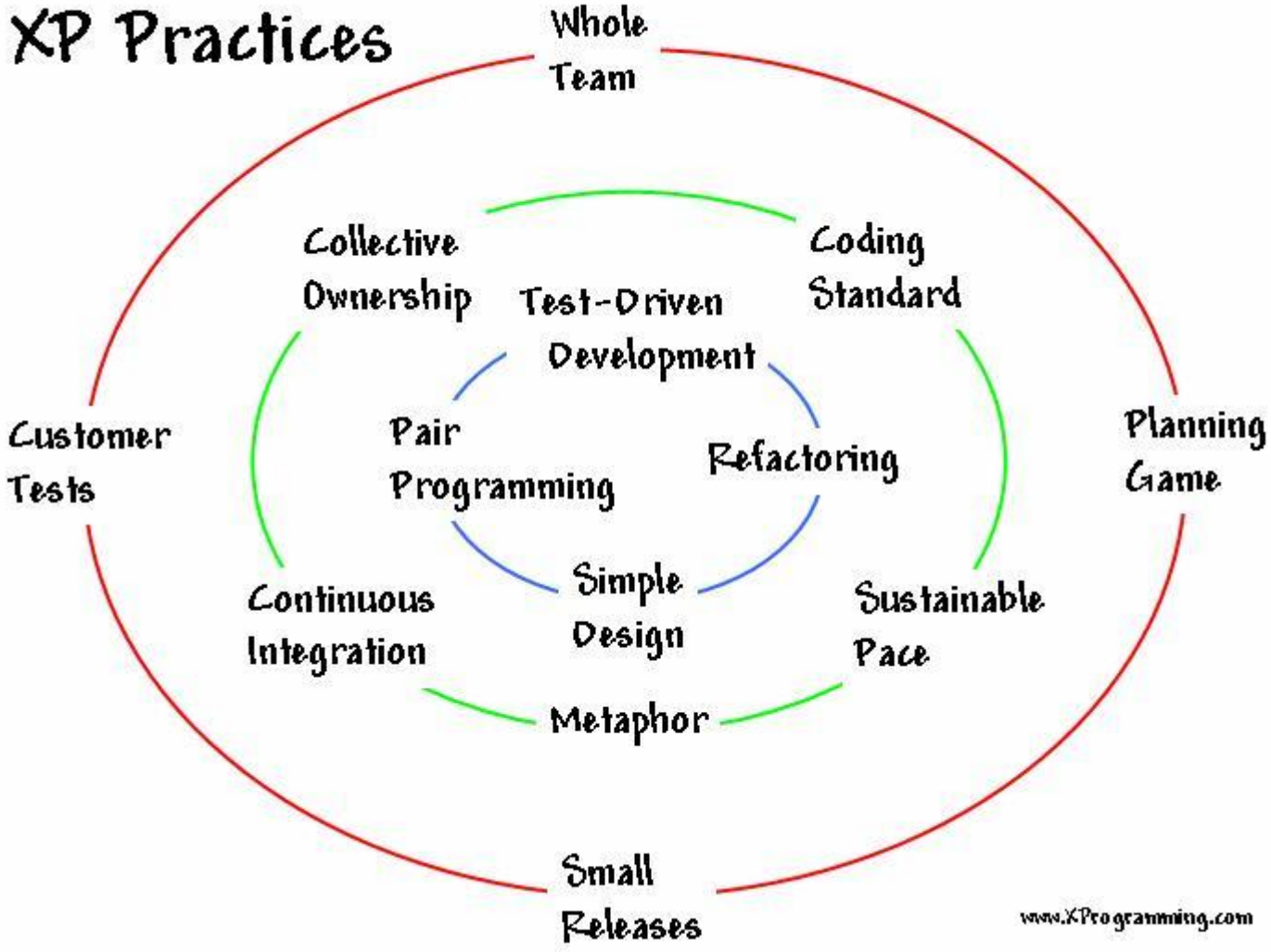


Multi-team sprint planning





XP fills the gaps in Scrum



Summary

Advantages of Scrum & XP

- **High quality**
- **High transparency**
- **Motivated teams**
- **Right decisions in the right hands**
- **80/20 rule**
- **Continuous improvement**
- **Low administrative overhead**
- **Managers focus on coaching and creating a good work environment**
- **... rather than telling people what to do**
- **Can be sneak-implemented to a certain extent :o)**

Disadvantages

- **Requires highly motivated & disciplined teams**
- **Hurts in the beginning**
- **Requires a corporate culture that sees problems as opportunities to improve**
- **Uncomfortable for people that like the illusion of detailed long-term plans**
  - Typically ”old school” managers and customers
- **Proper implementation requires change in all layers of a company, including suppliers and customers**

References

- **http://www.google.com :o)**
- **http://agilemanifesto.org/**
- **http://www.mountaingoatsoftware.com/scrum/**
  - Good overview of Scrum, great articles
- **Agile software development with scrum**
  - The original scrum book
- **Agile project management with scrum**
  - Second scrum book. Full of interesting war stories.
- **Managing the design factory**
  - Theoretical foundations of lean & agile software development. Great book!
- **Scrum and XP from the Trenches**
  - My book. Hands-on description of Scrum & XP in ”real life”.

JEFF SUTHERLAND
KUNG AV SCRUM

SEMINARIUM 30 MAJ:

”Why Scrum works - and when it doesn’t”

Jeff går igenom Scrum och tar upp frågor som så många ställer:

- Fungerar Scrum verkligen i praktiken? Skalar Scrum?
- Vilka typiska misstag gör företag som ska börja med Scrum?
- Vilka typiska missuppfattningar finns om Scrum?
- När passar inte Scrum?
- Vilken typ företag behöver Scrum mest?
- Hur får jag in Scrum i mitt företag?

Tid: 30 maj 9.00–13.00
Lunch och mingel ingår!
Plats: Stockholm City Conference Centre
Norra Latin, Drottninggatan 71B, Stockholm.
Bokning: www.crisp.se/scrum

INTRODUKTION TILL SCRUM:

Beställ Scrumintroduktion till ditt företag. Vi erbjuder två paket:

- Snabbintroduktion på 3 timmar. Sammanfattning av Scrum teori och praktik samt en kort övning/rollspel.
- Heldag. Samma som ovan men med mer tid att gå in på praktiska exempel och ditt företags unika situation.

Kontakta scrum@crisp.se eller ring +46 8 556 950 15 för mer information.

INTENSIVKURS 28–29 AUGUSTI:

Certified Scrum Master kurs

2 dagars intensivkurs fullspäckad med övningar, rollspel, diskussion och teori. På köpet blir du certifierad Scrum Master och medlem i Scrum Alliance.
Lärare är Jeff Sutherland och Henrik Kniberg.

Tid: 28–29 augusti, 9:00–17:00
Plats: Stockholm City Conference Centre
Norra Latin, Drottninggatan 71B, Stockholm.
Bokning: www.crisp.se/scrum

Jeff Sutherland grundade Scrum 1995 tillsammans med Ken Schwaber. Scrum är den mest välanvända agila metoden och används idag av hundratals företag världen runt, inklusive CMMI-5-företag. Jeff har hjälpt företag som Google, Microsoft, Oracle, Yahoo och Adobe att införa Scrum i både stor skala och i distribuerade team.

Henrik Kniberg från Crisp och specialist på Java och agila metoder. Henrik är författare till den populära boken «Scrum and XP from the Trenches», som i detalj beskriver erfarenheter från ett långtidsuppdrag som utvecklingschef på Tain AB där han implementerade Scrum och XP genom hela organisationen.

crisp

Wallingatan 18 , 111 24 Stockholm.
08-556 950 15. info@crisp.se