cphbusiness

COPENHAGEN BUSINESS ACADEMY

Systems Development – Introduction to Scrum

Agenda for Scrum week

Scrum

- Process tool that describes how teams can work together to develop a product

Scrum Day 1

- Overview of Scrum
  - Agile Development using Scrum
  - Why/What/How?
  - Team contract for Fog project

Scrum Day 2

- Elephant Carpaccio exercise
  - Poker game
  - User stories
  - Estimation
  - Done criteria

Agenda for Scrum week

Scrum Day 3

- Sprint planning
  - Tasks
  - Daily scrum meeting

Scrum Day 4

- Scrum tools & work on PBL for Fog project

Scrum Day 5

- Group review & discussion of PBL version 1

Learning Objectives for Scrum

- Knowledge of Scrum basic process model
  - How to document and estimate customer requirements
  - How to turn these requirements into an operational format the developers can use to control their daily work
  - How to monitor and manage the development effort
  - How to calculate team velocity, meaning how much work a team can handle in time-boxed period
  - How to work in an iterative manner where software is build piece by piece

Main literature

Henrik Kniberg *Scrum and XP from the Trenches:*

http://www.infoq.com/minibooks/scrum-xp-from-the-trenches

- Pages : *pp. 1-13 day 1*
- *pp. 14-50 day 2*
- *pp. 51-68, 75-92 day 3*

Traditional waterfall example
To build a house!

Phase 1 – idea/analyse

Phase 2 – design

Phase 3a – foundation

Phase 3b – wall

Phase3c – room

Phase 3 – Construction

cphbusiness

Traditional Waterfall vs. Iterative Approach

- General comparison of two paradigms (i.e. methodology classifications)

Comparing Waterfall To Iterative

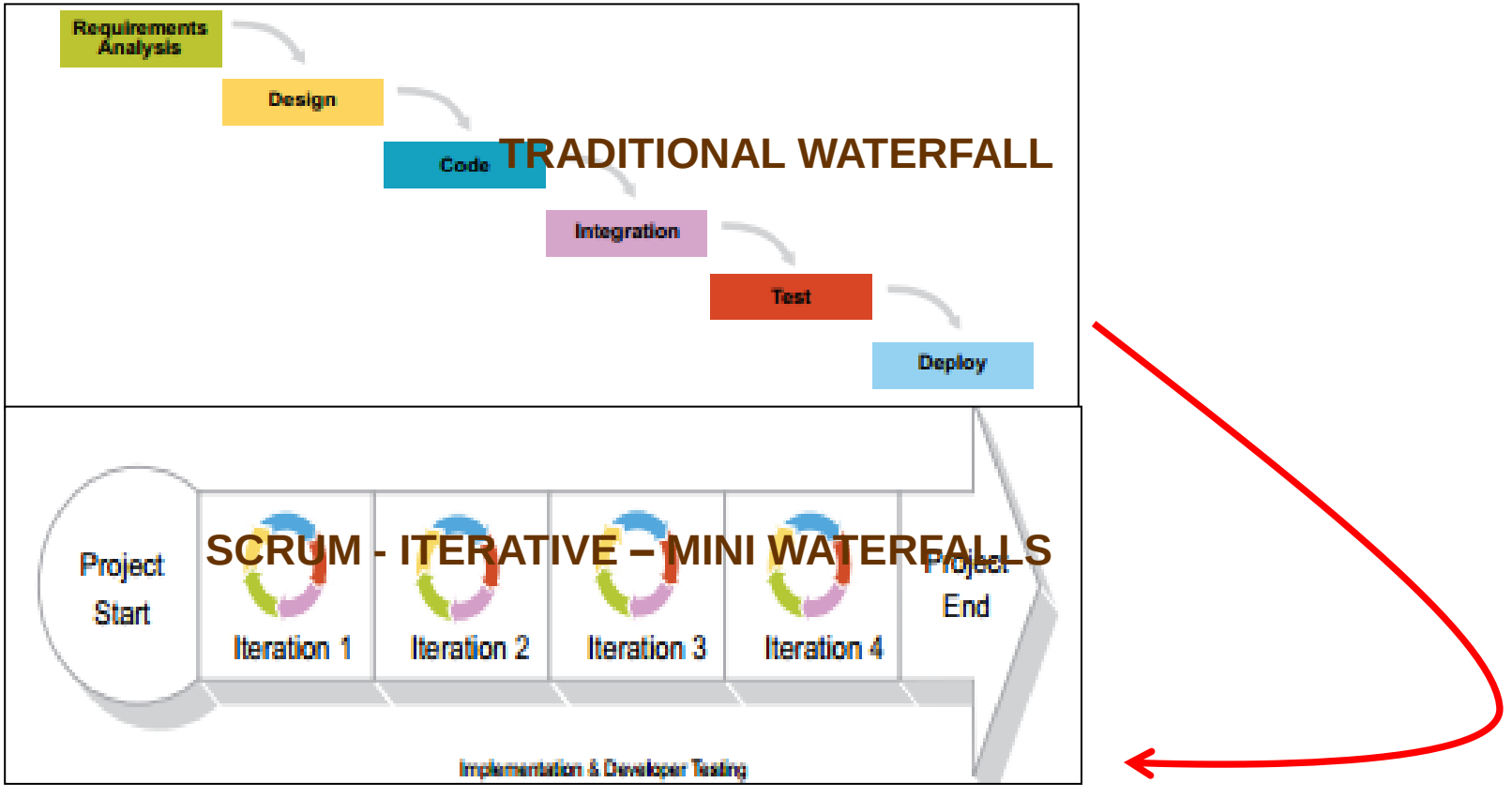
Waterfall	Iterative
Risk averse	Actively attacks risk
Subjective measurement of progress	Objective measurement of progress
Delays integration and testing	Continuous integration and testing
Nothing runs until the end	Something "runnable" produced every iteration
Difficulties at the end of the project	Difficulties at the start of the project

Source: http://www.ibm.com/developerworks/rational/library/4029.html

Traditional "waterfall" development depends on a perfect understanding of the product requirements from the beginning and minimal errors made in each phase.

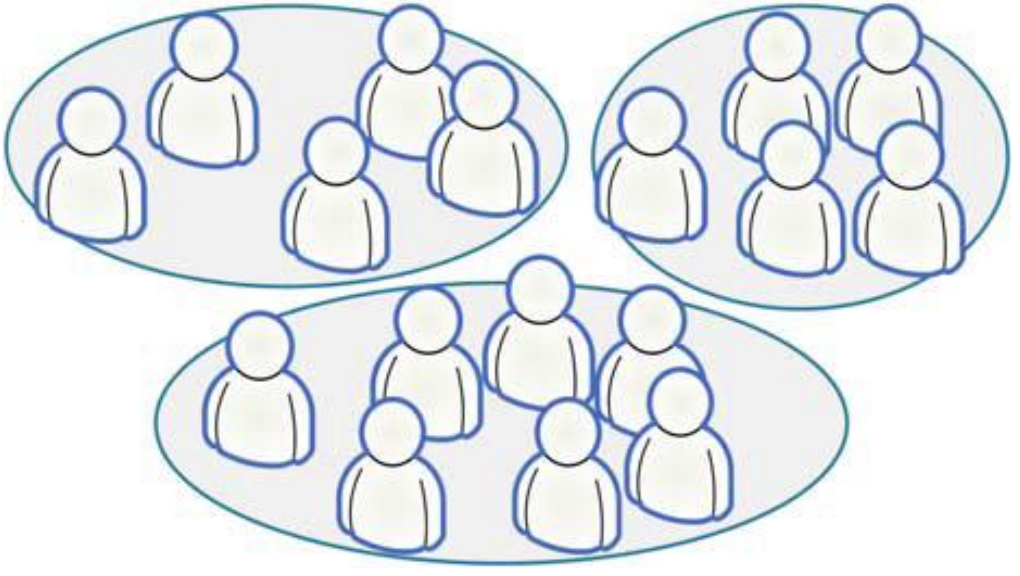
Scrum in a Nutshell 1

- The Scrum is iterative process
  - Many small water falls, called **sprints**



Scrum in a Nutshell 2

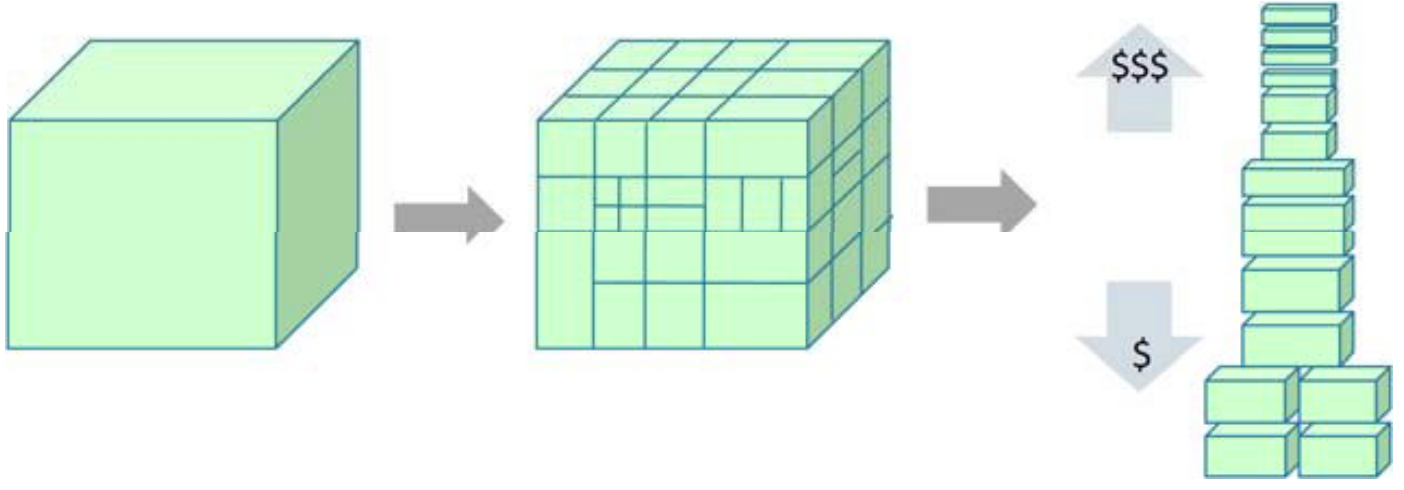
- Split your organization into small, cross-functional, self organizing teams.



Source: Kniberg ” KANBAN AND SCRUM – MAKING THE MOST OF BOTH”

Scrum in a Nutshell 3

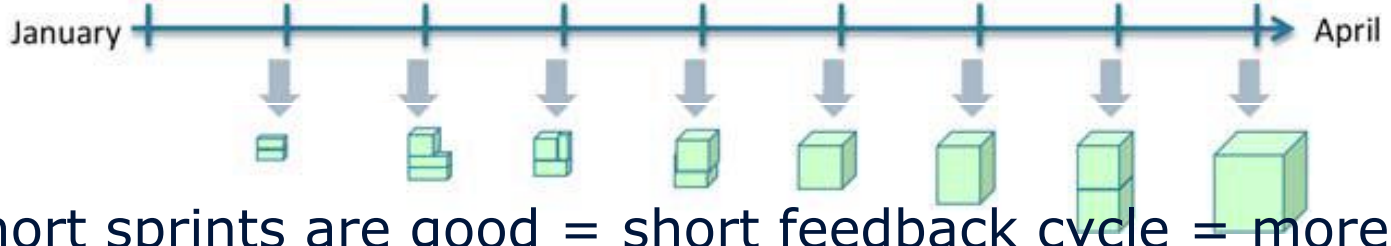
- Split your work into a list of small, concrete deliverables.
  - Sort the list by priority
  - Estimate the effort of each item



Source: Kniberg " KANBAN AND SCRUM – MAKING THE MOST OF BOTH"

Scrum in a Nutshell 4

- Split time into short fixed-length iterations (usually 1 – 4 weeks), with potentially shippable code demonstrated after each iteration.



- Short sprints are good = short feedback cycle = more frequent deliveries = more frequent customer feedback = less time spent in the wrong direction = learn and improve faster.
- Long sprints are good too. The team gets more time to build up momentum, they get more room to recover from problems and still make the sprint goal; less overhead for sprint planning meetings, demos etc. (Kniberg p.19-20)

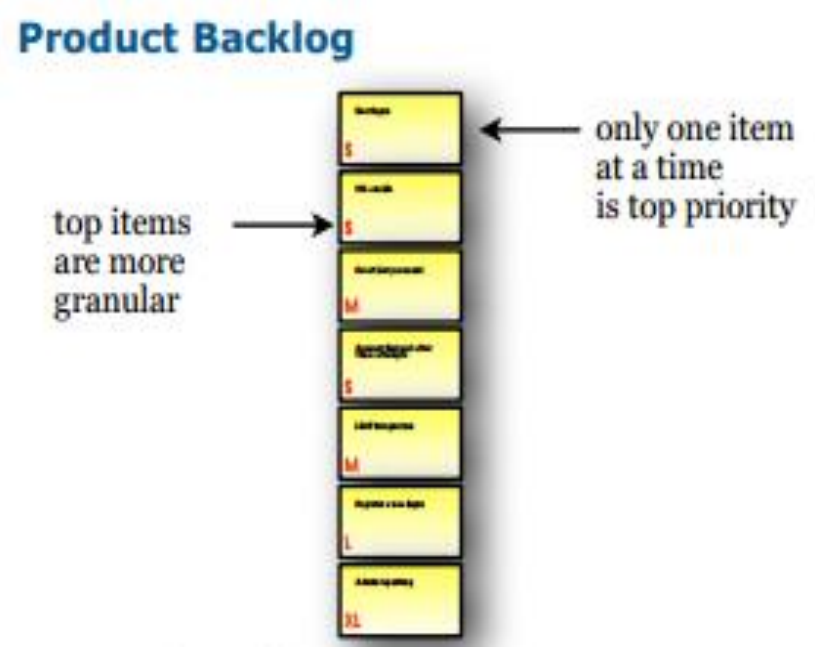
Scrum in a Nutshell 5

- After each iteration …
  - Optimize the release plan and update priorities in collaboration with the customer, based on insights gained by inspecting the release
  - Optimize the process by having a retrospective after each iteration.

Source: Kniberg ” KANBAN AND SCRUM – MAKING THE MOST OF BOTH”

The Product Backlog

- A prioritized list of everything that might be needed in the product
  - requirements, features etc.
  - Things that the customer wants, described using the customer's terminology.



Product Backlog Item

- Often called (user) story, or just PBI.
- Example:

User Story

- … is short, simple description of a feature told from the perspective of the person who desires the new capability (typically user or customer)
- User stories can be written informally:

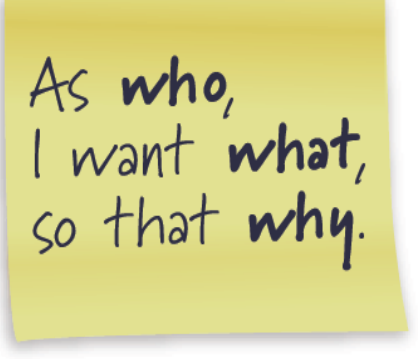
  Registered users can reset their password

- Or use a bit more formal template

  ***As** a registered user,*

  ***I want** to reset my password,*

  ***so that** I can get back into the site if I forget my password*

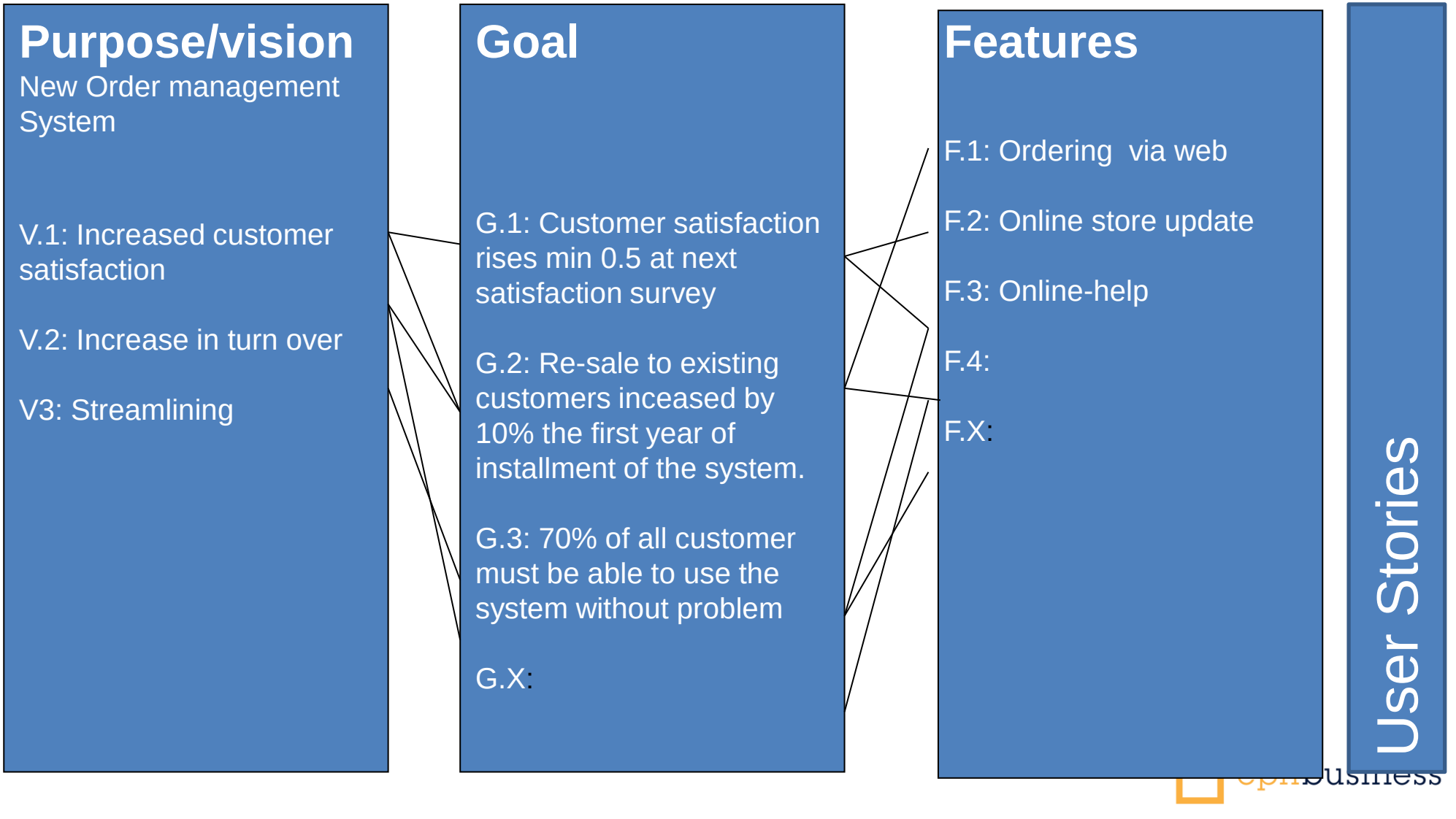


Product Backlog Example

- Story example where feature description is specified in "How to demo" field, i.e. test steps description (acceptance criteria) (Kniberg p. 10)

PRODUCT BACKLOG (example)					
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.

The traceability model & scrum!



Who is Responsible for Product Backlog?

- The Product Owner!
  - Represents the stakeholders, i.e. the voice of the customer.
- Is responsible for maximizing product value
- Is responsible for managing the PBL:
  - Creating Product Backlog items (user stories)
  - Prioritizes the items in the Product Backlog
  - Ensuring the Development Team understands items to the level needed.

Scrum Roles

- **Product Owner**: responsible for the business value of the project
- **Development Team**: self-organizes to get the work done
- **Scrum Master**: ensures that the team is functional and productive

Product Owner

- Represents the stakeholders (= customer voice)
- Is responsible for maximizing product value
- Is responsible for managing the PBL:
  - Create Product Backlog items (user stories)
  - Prioritize Product Backlog items
  - Ensure the teams understands items

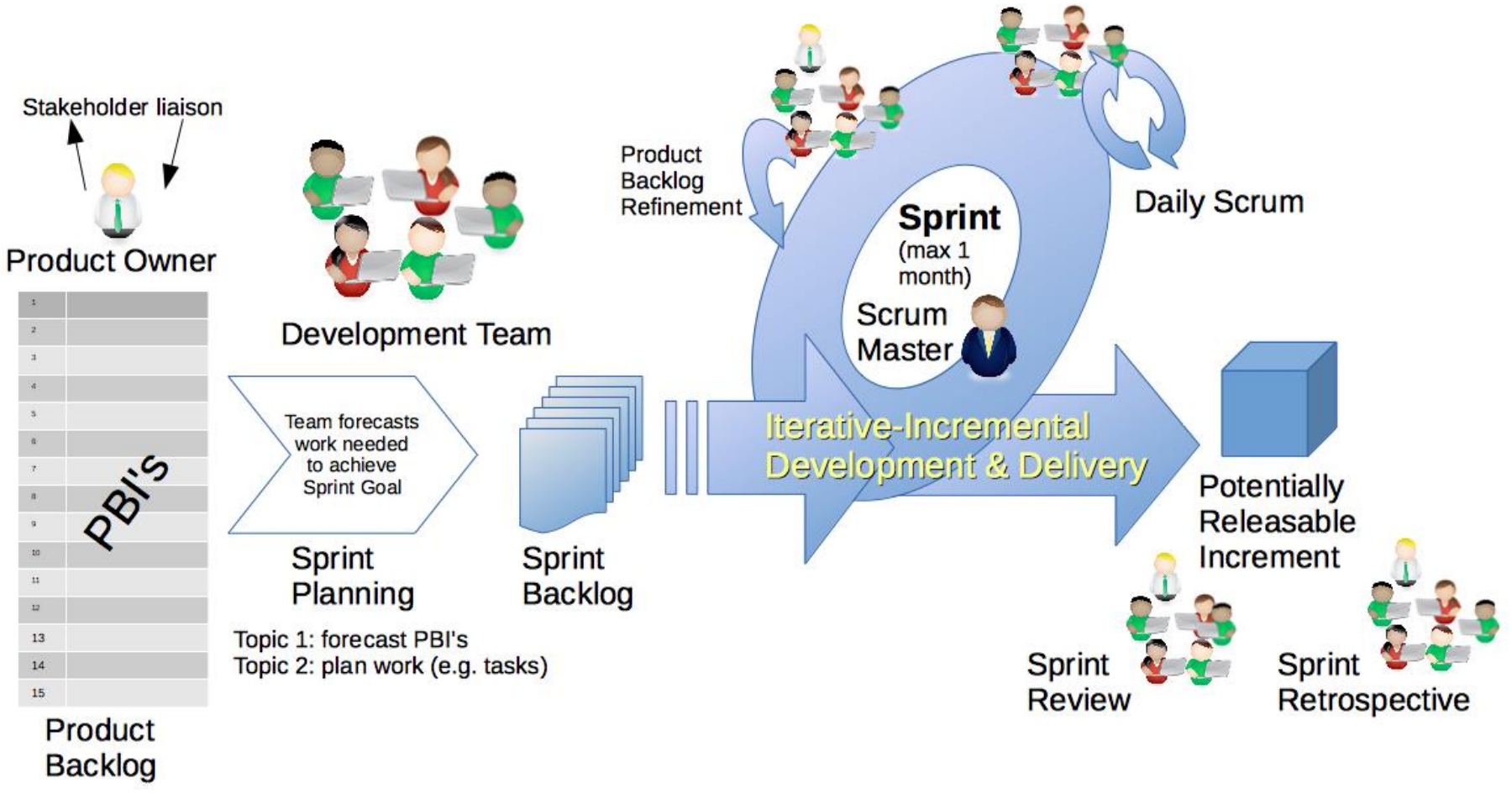
Scrum Master

- The Scrum Master is the process owner ??
  - responsible for ensuring Scrum is understood and enacted
  - Helps the team perform at their highest level
  - Protector of the team (coach)

Scrum Master

- Servant Leader
- Monitoring & Tracking
- Reporting & Communication
- Process Check Master
- Quality Master
- Resolve Impediments
- Resolve Conflicts
- Shield the team
- Performance Feedback

Summary


Stakeholder liaison
Product Owner
Development Team
Product Backlog Refinement
Sprint (max 1 month)
Scrum Master
Daily Scrum
PBI's
Product Backlog
Team forecasts work needed to achieve Sprint Goal
Sprint Planning
Topic 1: forecast PBI's
Topic 2: plan work (e.g. tasks)
Sprint Backlog
Iterative-Incremental Development & Delivery
Potentially Releasable Increment
Sprint Review
Sprint Retrospective

What can you say about Fog?

- Put Vision, Goal and features into the (traceability-model)
  - **Features provide input for the user stories**
- Draw activity diagram of significant processes
- Make a team contract
  - Consider scrum master role
- Use ”Team contract” template on github for inspiration
- On Friday, all groups show version 1 of their PBL

Scrum in a nutshell

Let’s watch Introduction to Scrum video (17 minutes ++):

http://scrumtrainingseries.com/

It is mandatory that the product increment be released to production at the end of each Sprint?

- True
- False

The purpose of a Sprint is to produce a done increment of working product

- True
- False

How would you decide your length of Sprinit

On what basis should you decide on the next Spirint?

Other questions

- When should you begin the next spirint?
- How would you describe scrum for your team?
- What is the main reason for the Scrum Master to be at the Daily Scrum?
- What is the time-box for a Sprint/Sprint planning/Daily scrum(15 minutes?)
- What should you do if you see a Sprint is not really working?
- When is a Sprint over?
- What is the team definition of “Done”
- What is a Sprint Review?