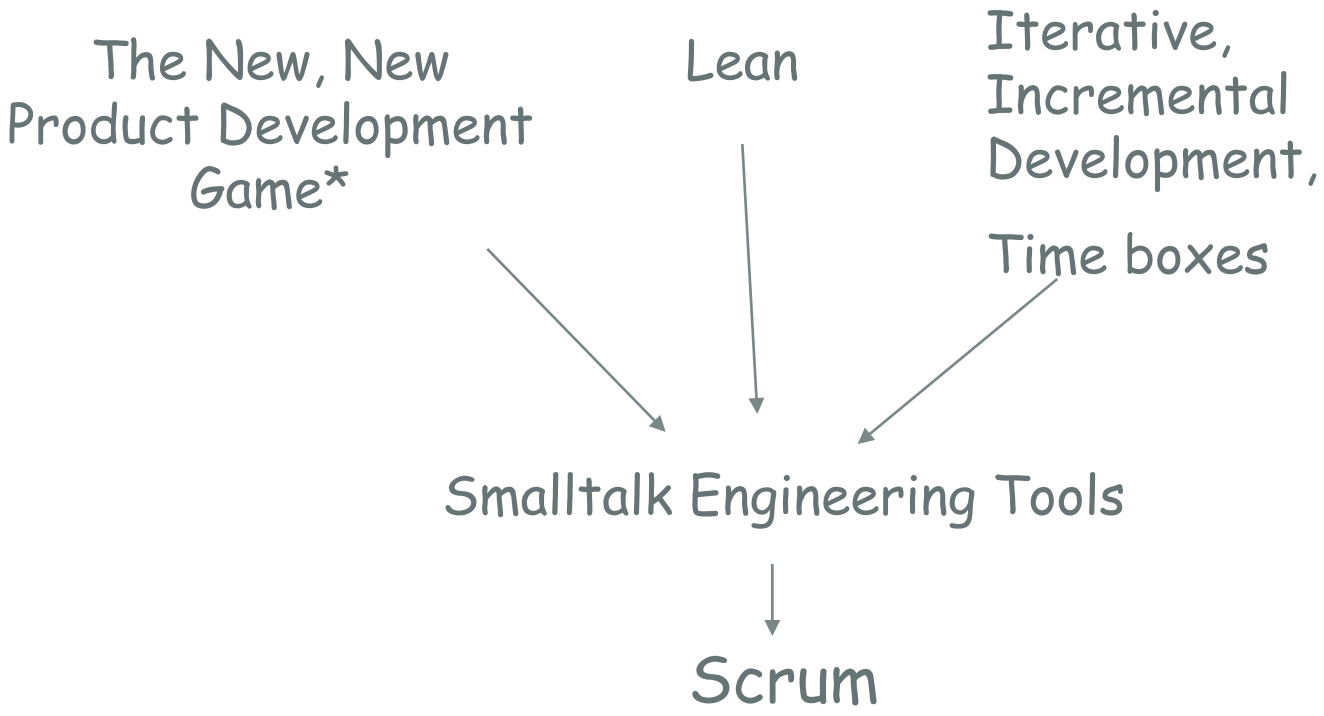
Scrum Introduction

Petri Heiramo

Agile Coach, CST

Origins of Scrum



* Harvard Business Review, Jan. 1986, Takeuchi and Nonaka

Scrum started in the Harvard BR.

"The… 'relay race' approach to product development…may conflict with the goals of maximum speed and flexibility. Instead a holistic or 'rugby' approach—where a team tries to go the distance as a unit, passing the ball back and forth—may better serve today's competitive requirements."

Hirotaka Takeuchi and Ikujiro Nonaka, "The New New Product Development Game", *Harvard Business Review*, January 1986.

Scrum first mentioned

Scrum is a Framework

- Project management framework
  - Does not specify technical practices
  - Provides only a minimal set of project ”control” tools
- Key practices
  - Time-boxed sprints
  - Self-managing teams
  - Stand-up daily meetings
  - Iterative development
  - End-of-iteration demo
  - Continuous learning

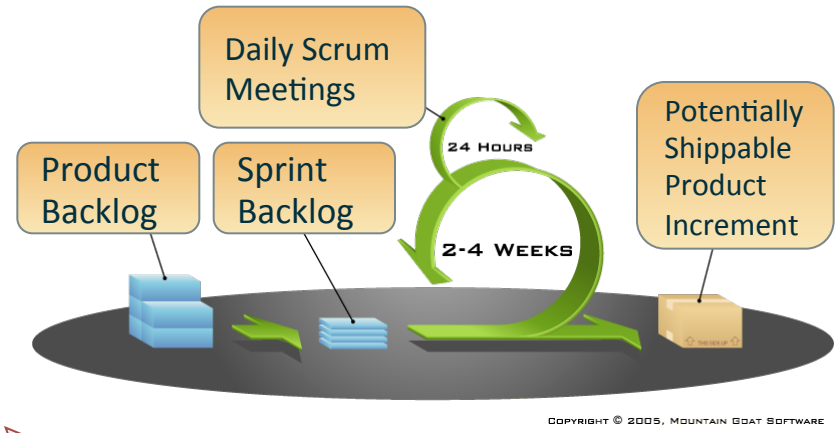
Scrum Process

"Pre-Game"

Initial planning

Initial concept

"Development Sprints"



"Post-Game"

"Clearing the table" …

…

Start sprints when project can

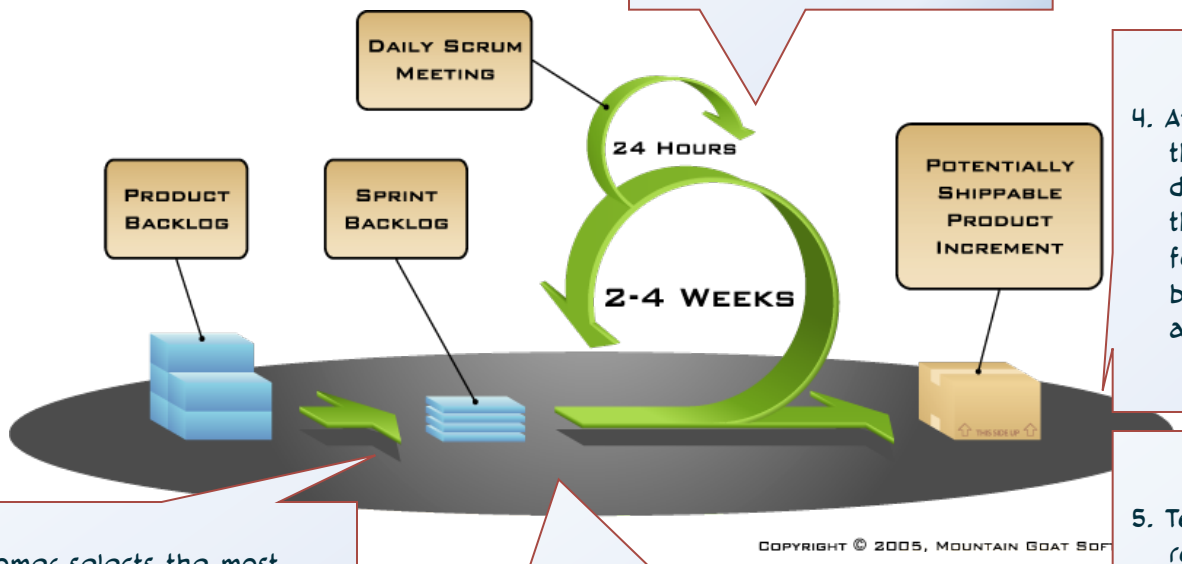
- define initial product backlog, and
- prioritize enough work for the first sprint

Development ends

at planned time, or

When there are no more features valuable enough to be developed

Sprint Structure



3. Team tracks and manages the progress of the sprint through daily stand-up meetings.

4. At the end of the sprint the team presents the developed functionality to the customer for feedback. The demo is based on the agreed acceptance tests.

1. Customer selects the most valuable features to be implemented in the sprint based on feedback from the team. An acceptance test for each feature is defined.

2. Team breaks the features into tasks and plans the implementation of the functionality in the sprint.

5. Team conducts a retrospective to identify improvement opportunities for the team and the development process.

No changes during a sprint

Why is the development peace so important for productivity?

- Plan sprint durations around how long you can commit to keeping change out of the sprint
- PO can terminate an iteration if its goal is invalidated

Key Roles and Responsibilities

Product Owner

Goals
- Get the correct product
- Develop features with most value

Responsibilities
- Create a product vision
- Release planning
- Facilitate real user feedback
- Determine business value
- Define acceptance criteria

ScrumMaster

Goals
- Customer gets value for money
- Protect team from outside interference
- Product owner and team fulfill their responsibilities

Responsibilities
- Nurture Team's self-management
- Support team to be productive
- Support product owner in determining business value
- Eliminate discovered Impediments
- Facilitate the Scrum process
- Ensure communication
- Manage project environment

Development Team

Goals
- Develop most value to customer
- Maintain ability to deliver change

Responsibilities
- Estimate effort and dependencies
- Plan iteration execution
- Develop regular deliveries of ready functionality

Who is a Product Owner

- Mishkin Berteig has suggested the following definition:

  *”It is the Product Owner who is **ultimately responsible for defining business value, priority and details** for all the work done by the development team. Product Owner’s authority stems from a deep understanding of the project’s goals and a respected position among stakeholders.”*

- Typically a project or product manager in a customer organization

- The link between an Agile development team and stakeholders
  - PO has to be in direct contact to both directions

Desirable Characteristic for PO

- Visionary and Doer
  - Can envision the product, and see it to completion
- Leader and Team Player
  - Primus inter pares – first among peers
- Communicator and Negotiator
  - Communicates with and aligns different parties
- Empowered and Committed
  - Enough authority and support, committed to development effort at all levels
- Available and Qualified
  - Enough time (not overworked), right skills

Central Success Criteria

- One PO per project
  - When there are multiple PO's, conflicts regarding requirements and priorities are easily developed
- Ability to represent all stakeholders and user groups
  - All important needs are heard
  - Can meaningfully prioritize different needs
- Authority to make business decisions regarding project's features and priorities
- An active participation in the project
  - Requirements development and prioritization throughout the whole project
- Ability to increase Agility in the project's environment
- Focus on business value and features, not on micromanaging the team

Common Mistakes with PO's

- Underpowered PO
- Overworked PO
- Partial / split PO responsibility
- Distant PO
- Proxy PO
- PO Committee

ScrumMaster Role

”As a ScrumMaster, you are responsible for:

- Removing the barriers between development and the customer so the customer directly drives development
- Teaching the customer how to maximize ROI and meet their objectives through Scrum
- Improving the lives of the development team by facilitating creativity and empowerment
- Improving the productivity of the development team in any way possible
- Improving the engineering practices and tools so each increment of functionality is potentially shippable”

Ken Schwaber

Further Aspects on the Role

- NOT the traditional project manager
  - There are similarities, though
- Scrum Master must act **without resorting to traditional command-and-control style**
- It helps SM to an understanding of the “big picture” of the project
- Being a good SM is just as time consuming as being a good project manager (if not more)

How Many Teams for One SM?

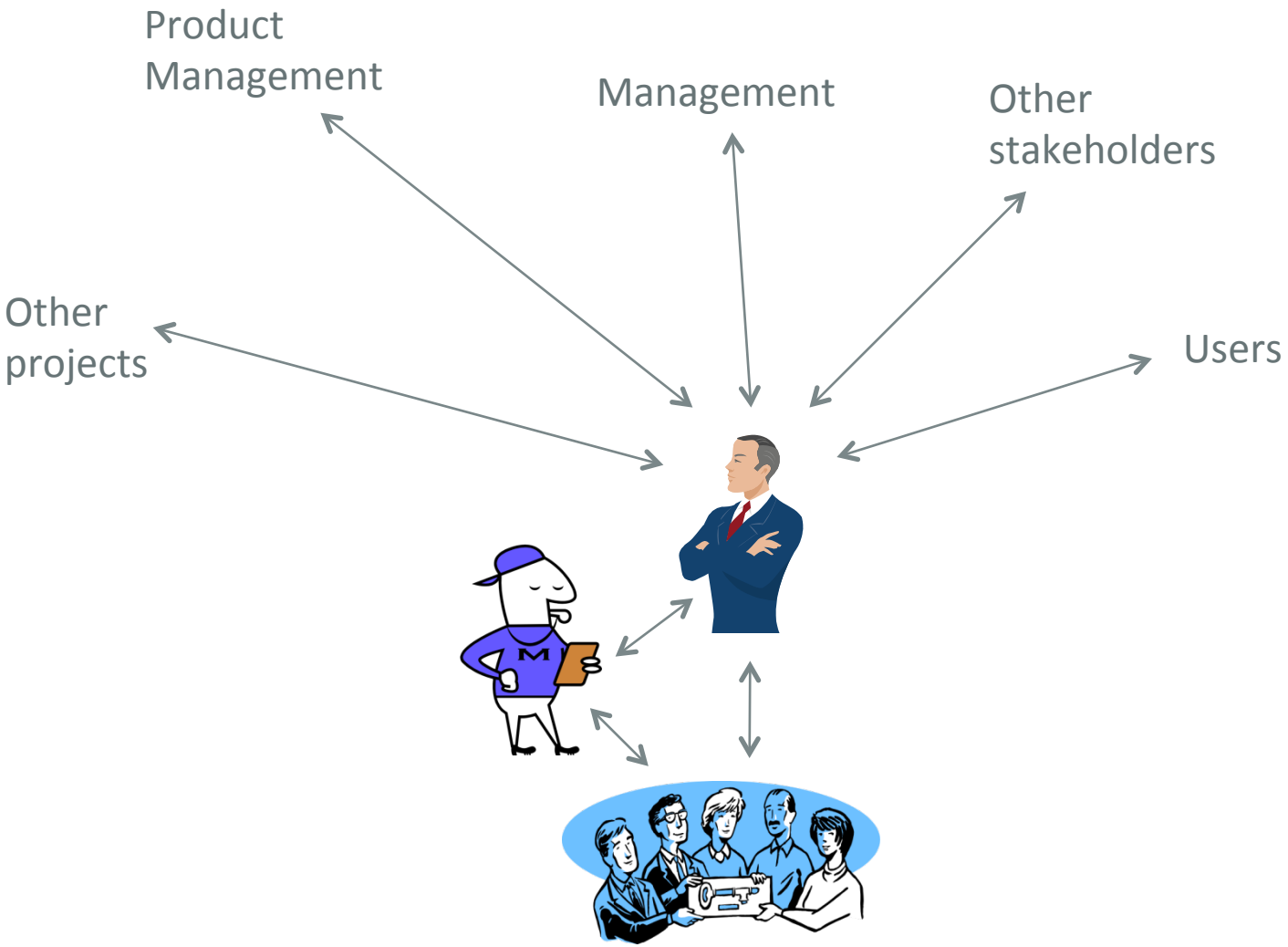
An adequate ScrumMaster can handle two or three teams at a time. If you're content to limit your role to organizing meetings, enforcing timeboxes, and responding to the impediments people explicitly report, you can get by with part-time attention to this role. The team will probably still exceed the baseline, pre-Scrum expectation at your organization and chances are nothing catastrophic will happen.

But if you can envision a *hyperproductive* team -- a team that has a great time accomplishing things no one else can -- consider being a *great* ScrumMaster.

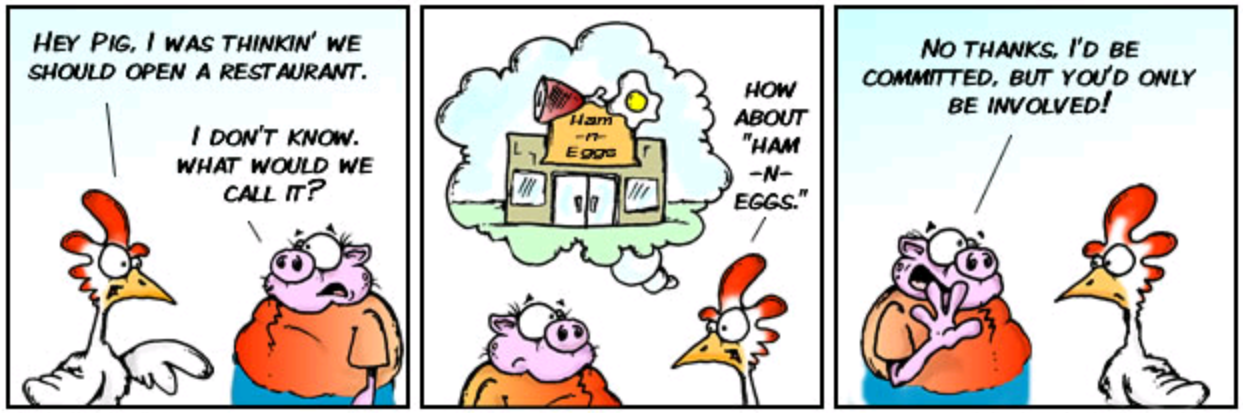
A great ScrumMaster can handle *one* team at a time.

Michael James, http://danube.com/system/files/A_ScrumMaster's_Checklist_blog.pdf

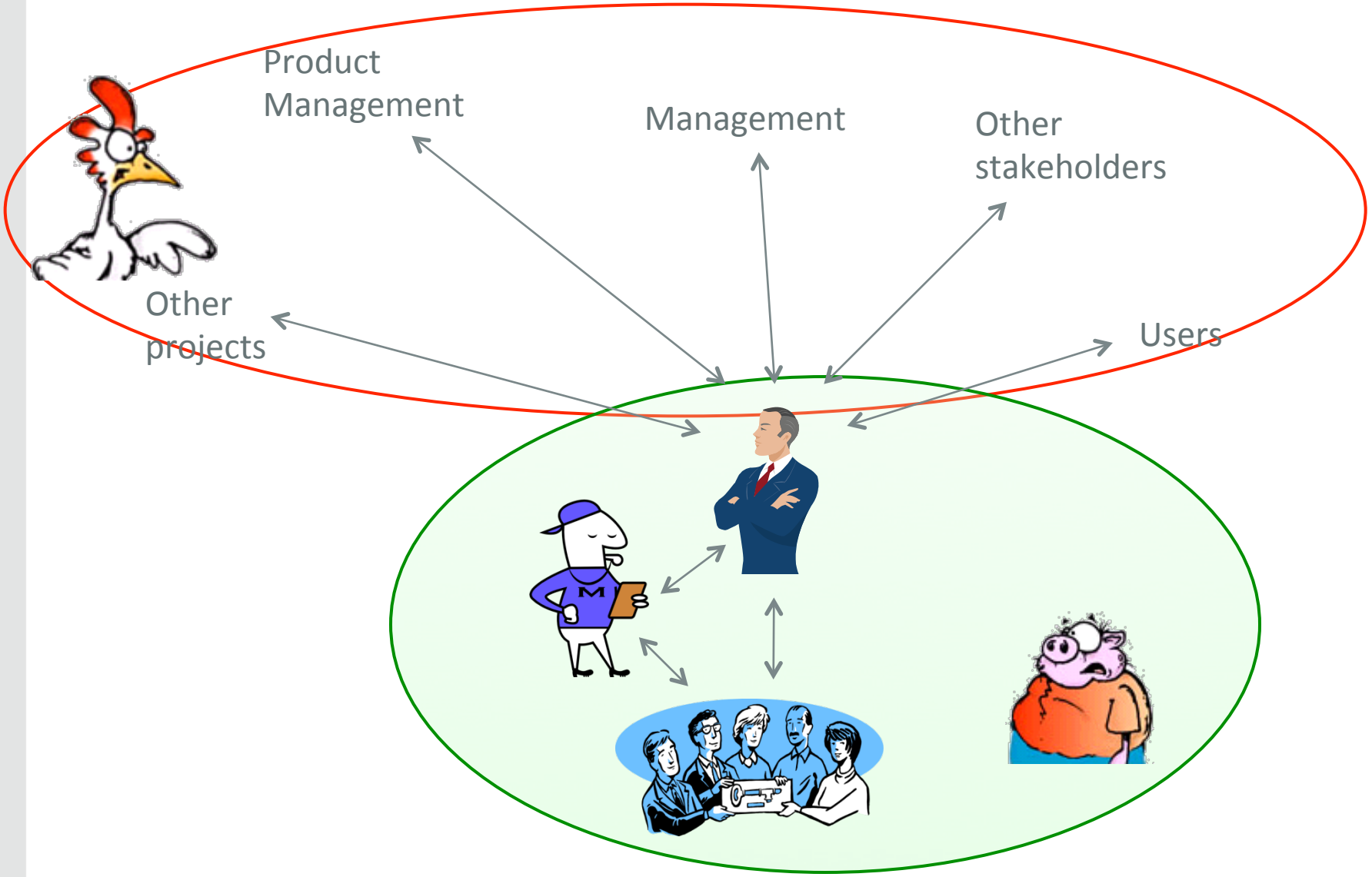
Stakeholders



Pigs and Chickens



Stakeholders



Scrum Meetings

- Sprint Planning
  - Sprint goal (team with product owner)
  - Design and task planning (team)
- Daily Scrums
- Sprint Review
- Sprint Retrospective

Planning Activities

- Release planning
  - Analyze and evaluate product backlog
    - Is it up to date? Are there new stories to be evaluated?
    - Is it prioritized?
  - Plan longer term goals
    - When do we need to be ready? What does ”ready” mean at that time?
- Sprint Planning
  - Decide a sprint goal and agree sprint scope
  - Divide the work needed to tasks (work planning)
  - Estimate the size of the tasks in ideal hours
    - Confirm team commitment by comparing to estimated capacity

Daily Scrums

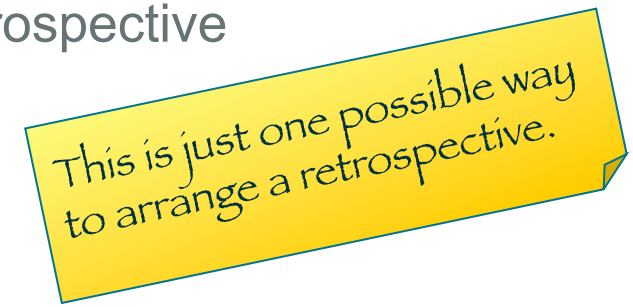
- Purpose: The team coordinates its activities and collaboration among themselves
  - NOT: Team reporting to ScrumMaster
  - NOT: Team doing design
  - This is a work planning meeting, no other activities
- Project's internal meeting
  - Others can participate, but not talk
- Design issues and other topics shall have separate meetings arranged after the daily scrum

Sprint Review

- Team presents what it accomplished during the sprint
- Typically takes the form of a demo of new features or underlying architecture
- Informal
  - 2-hour prep time rule maximum
  - No slides
- Whole team participates
  - PO, Team, SM
- Invite the world
  - Show your progress to your stakeholders
  - Get stakeholder feedback

Sprint Retrospective

- Inspect & Adapt
  - Key rule to successful Agile development
- Arranged at the end of every sprint
  - PO, Team, SM, possibly also stakeholders
- Discuss together
  - What to start doing
  - What to stop doing
  - What to continue doing
- Select a few (maybe one or two) concrete actions for the next sprint
  - Review at the next retrospective



Scrum Artefacts

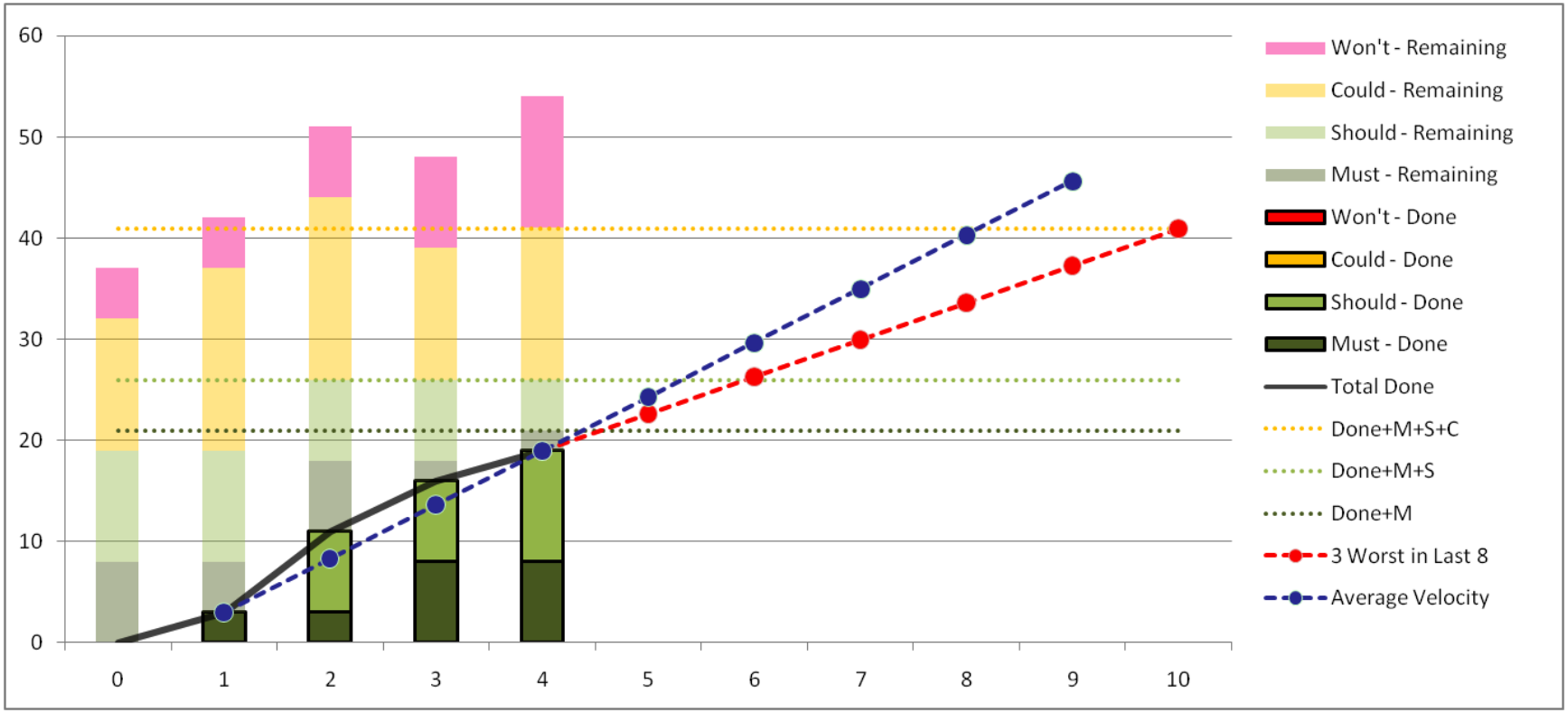
- Product Backlog
  - ”Feature list” / requirements
- Sprint Backlog
  - ”Task list”, implementation plan
- Burndown charts
  - Release burndown
  - Sprint burndown

Product Backlog

- Contains all planned and potential functionality
- Prioritized by Product Owner on business value
- Each sprint, a set of functionality is chosen for implementation
  - Decided in collaboration between Product Owner and Team

Story ID	Story name	Status	Size	Sprint	Comments
1	User can see main view and browse the file list	Done	8	1	Use hard-coded directory, no subfolders
2	User can set initial directory	Assigned	5	2	Simple popup from Options menu, application starts in the directory
3	User can browse into subdirectories	Assigned	5	2	Recursive
4	User can open text files	Planned	3	3	Use Notepad to display
5	User can open any recognized files	Planned	8	3	Retrieve association information from system, invoke correct application
6	User is displayed an error message when trying to open an unrecognized file	Planned	2		Simple popup, no chance to associate to a program
7	User can use an internal editor to edit text files	Removed	20		Simple editing

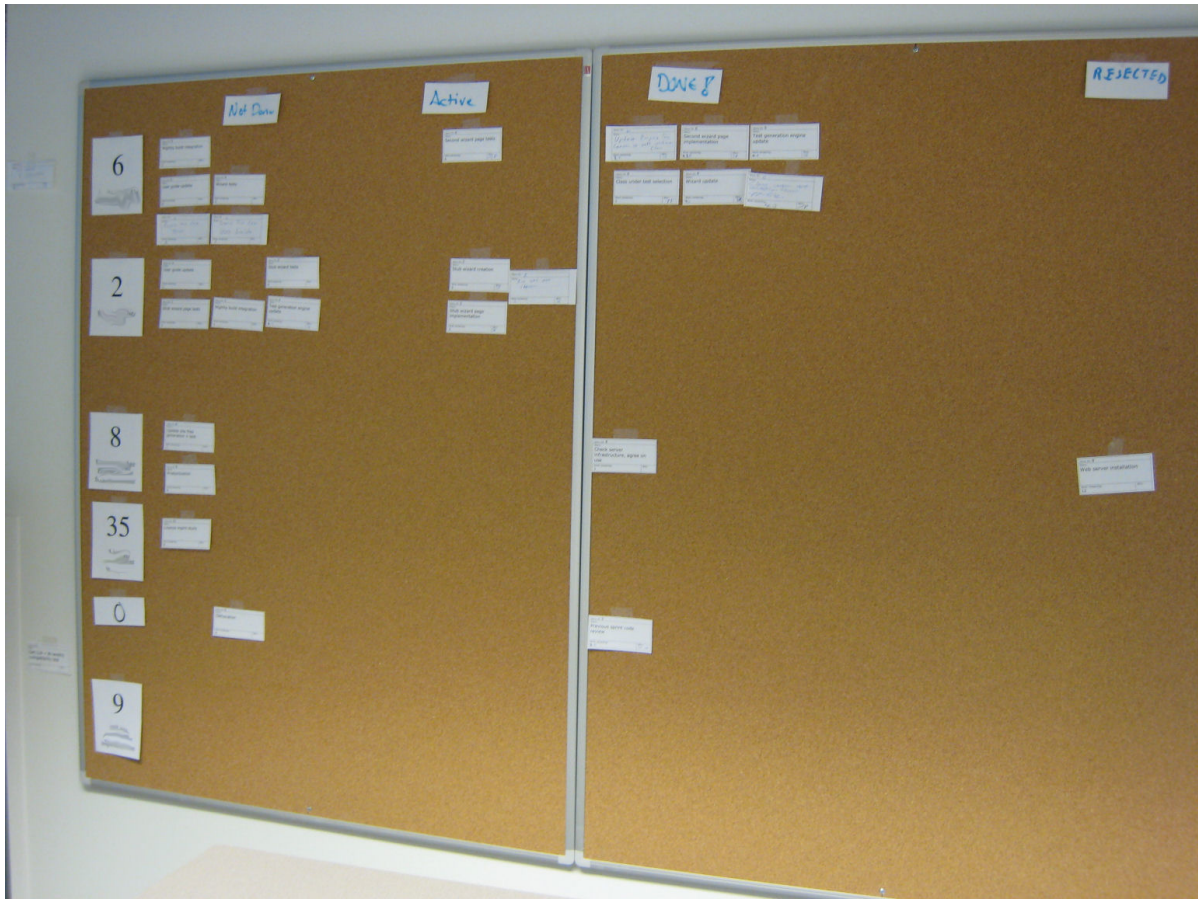
Burn-Up



Sprint Backlog

- Created by team for each sprint
  - Consists of tasks needed to implement planned functionality
- Used by the team to track their progress through the iteration
- Focus of sprint backlog planning is to plan the sprint execution and design the features
  - Estimates give a means to evaluate remaining work
  - The accuracy of task estimation is not so relevant, as long as the team can meet its commitment
  - Tasks can be changed freely during the sprint

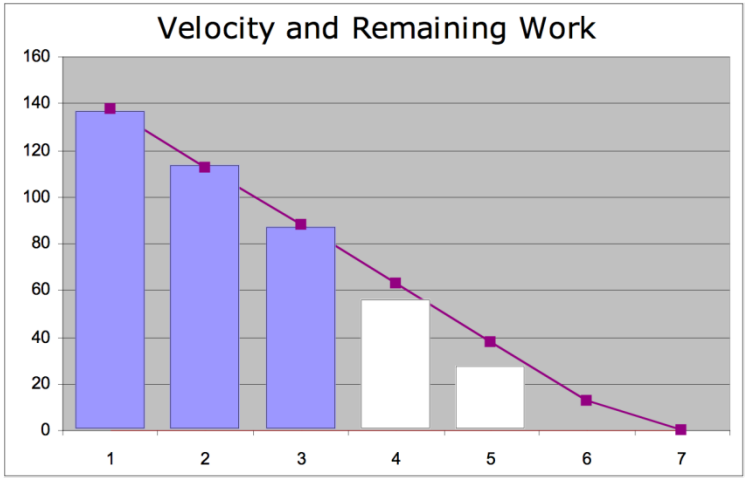
Sprint Backlog on a Wall



Tracking Progress

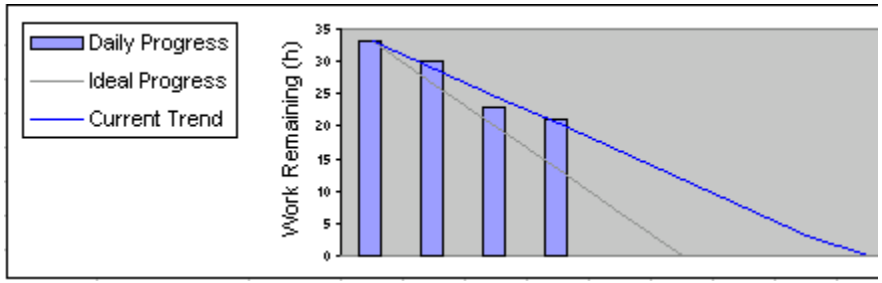
At sprint level:

- Shows remaining work in the project
- Gives an estimate of development speed and estimated end



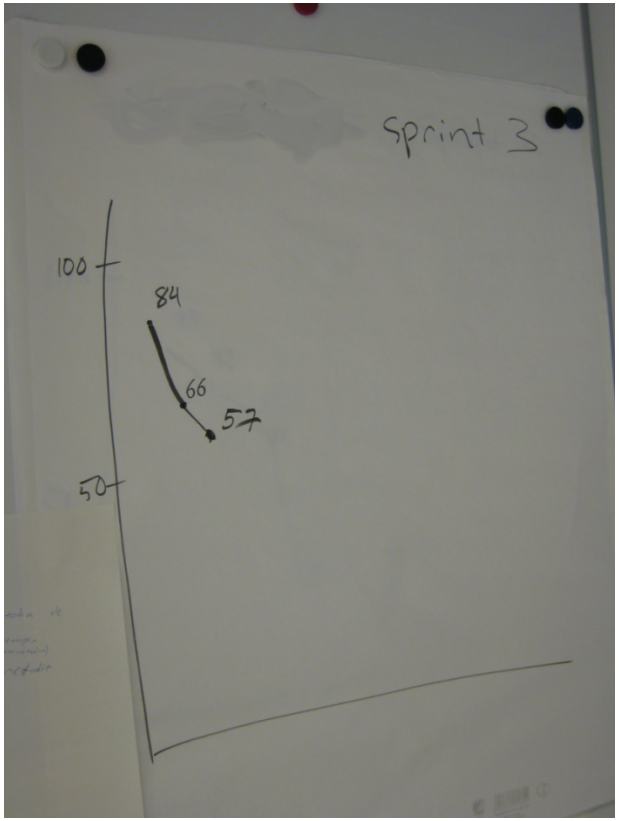
At daily level:

- Team tracks remaining work in the sprint
- Tells the team how they are progressing and can they expect to finish the sprint on time



Burndown in Practice

- "Big visible charts"
  - Often the best means of radiating information
  - Simple is beautiful
- If teams use a tool for task tracking, it will provide the burndown



What does a PO do during the Sprint?

- Communicate with stakeholders and users
  - Collect feedback from completed features
  - Collect feedback on priorities and the usefulness of planned features
- Ensure that the team gets clarifying details about agreed features
- Communicate with the team about issues arising during the Sprint
- Clarifies the contents and priorities of the Product Backlog
- Discusses with the ScrumMaster and stakeholders regarding project's plans and details them as necessary
- Removes problems restricting the productivity of the team

A PO Reading List

- *Agile Product Management with Scrum* - Roman Pichler
- *Agile Estimating and Planning* - Mike Cohn
- *Lean Software Development: An Agile Toolkit* - Mary and Tom Poppendieck
- *Implementing Lean Software Development: From Concept to Cash* - Mary and Tom Poppendieck
- *User Stories Applied for Agile Software Development* - Mike Cohn