Berner Fachhochschule
Haute école spécialisée bernoise
Bern University of Applied Sciences

Software Engineering and Design

Introduction to Scrum

Prof. Dr. Olivier Biberstein
Prof. Dr. Eric Dubuis

Bern University of Applied Sciences, Computer Science Department, Version September 2015.

Scrum

Scrum

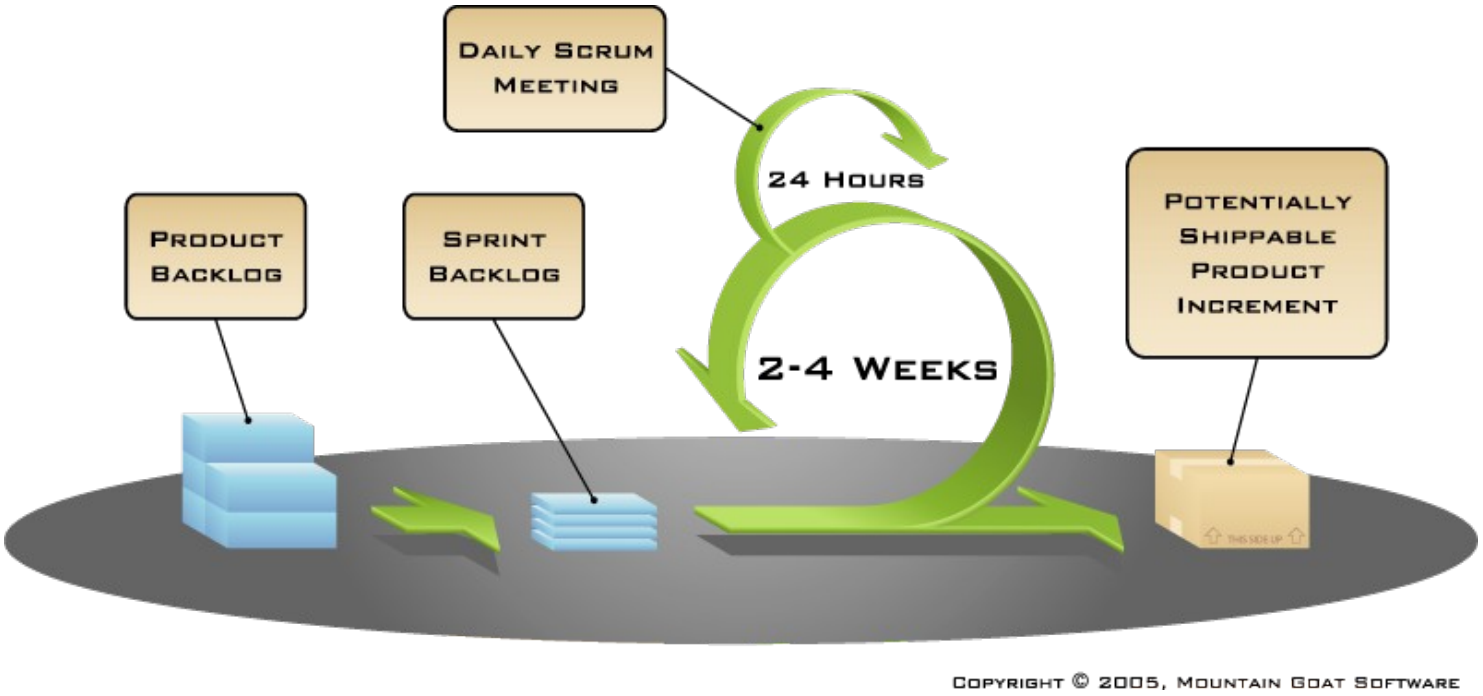
- Scrum is an agile process that allows us to focus on delivering the highest business value in the shortest time.
- It allows us to rapidly and repeatedly inspect actual working software (every two weeks to one month).
- The business sets the priorities. Teams self-organize to determine the best way to deliver the highest priority features.
- Every two 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 Origins

- Jeff Sutherland
  - Easel Corp 1993
- Ken Schwaber
  - Presentation at OOPSLA 1995 (Schwaber, Sutherland)
  - Author of several books on Scrum
- Mike Beedle
  - Scrum: A Pattern Language for Hyperproductive Software Development POoPD4 1999
- Ken Schwaber and Mike Cohn
  - Co-founder of Scrum Alliance in 2002, initially within the Agile Alliance http://www.scrumalliance.org/

Basic Characteristics

- Self-organizing teams
- Product progresses in a series of month-long “Sprints”
- Requirements are captured as a list of stories in “Product Backlog”
- “Sprint Backlog” consists in a list of tasks for a sprint



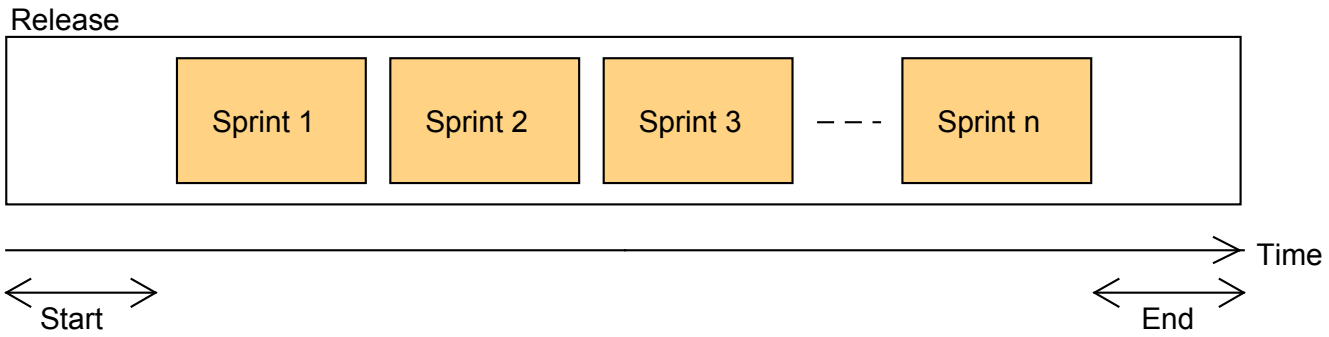
Sprints

- Scrum projects make progress in a series of “sprints”
- Analogous to iterations in other agile processes
- Product is designed, coded, and tested during the sprint
- Typical duration is 2–4 weeks or a calendar month at most
- A constant duration leads to a better rhythm
- No changes are allowed during a sprint (team, length, work)
- Length, amount of work, budget is the same for all the sprints

- Input: sprint Backlog, list of tasks (estimated 1-16 hours)
- Output: partial product potentially shippable

Release

- Larger projects require more than one single release



- Before the first sprint (start)
  - Form the team
  - Define the vision
  - First Backlog
  - Architecture
  - Fist planning of the release
- After the last sprint (end)
  - Delivery of the product
  - Deployment (cold, warm)
  - Packaging

Scrum Overview: Roles, Artifacts, and Meetings

Roles

- Product Owner
- Scrum Master
- Developers
- Customers, users, and managers

Artifacts

- Product Backlog
- Release Backlog
- Sprint Backlog
- Burndown chart

Meetings

- Product/release planning
- Sprint planning
- Daily Scrum
- Sprint review
- Sprint retrospective

Roles: Product Owner

- Single person that represents the customer
- Responsible for maximizing the return on investment (ROI) of the development effort
- Responsible for product vision
- Consider stakeholder interests
- Constantly re-prioritizes the Product Backlog according to the market value
- Final arbiter of requirements questions
- Accepts or rejects each product increment
- Decides whether to ship
- Decides whether to continue development

Roles: Scrum Master

- Supports the developer team
  - Coaches the team to produce high quality results
  - Ensures that the team adheres to Scrum values, practices and rules
  - Helps to resolve impediments
  - But does not manage the team: the team is self-managing
- Arranges (and participates in) all Scrum meetings
- Keeps tracks of the Backlog and measures progress
- Enforces timeboxes
- Shield the team from external interferences
- Communicates with customers and management outside the team

Roles: Development Team

- Turns Product/Release Backlog items into potentially shippable increments
- Negotiate commitments with the Product Owner, one sprint at at time
- Cross-functional (testing skills, domain experts, etc)
- Self-organizing, self-managing
- 5-9 members that participate in all Scrum meetings
- Most successful with long-term, full-time membership
- Has autonomy regarding how to reach commitments
- Show work results to Product Owner and other stakeholders

Roles: Other Stakeholders

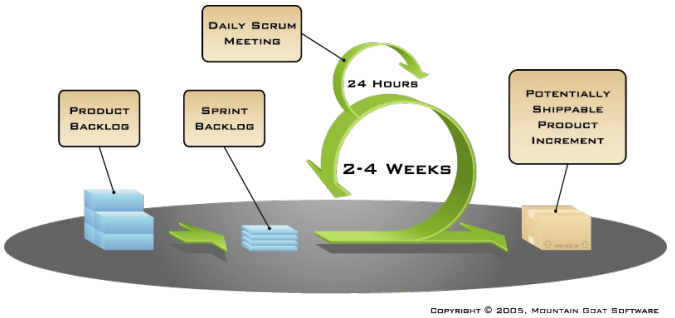
Customers and Users

- Participate in (product, release, and/or sprint) review meetings
- Represent business and user interests
- Provide feedback to Scrum team

Managers

- Participate in (product, release, and/or sprint) review meetings
- Provide healthy working environment and take resource decisions
- Represent business interests

Scrum Artifacts: Product Backlog



- A list of user stories (items) for implementing the product vision
  - User stories captures all major user and system requirements
- Prioritized by the Product Owner according to
  - business value, dependencies, deadlines, effort estimation, risks, …
- A "dynamic" list (may change after each sprint)

- *Product Backlog is not*
  - *Requirements specification with a lot of detailed requirements*
  - *Wish list with all possibly requirements*
  - *One-time contract between Product Backlog and development team*
  - *Description of concrete implementation activities*

Scrum Artifacts: Product Backlog Example

ID	Story Name	Item/Story Description	Priority	Estim. Effort [h]	Updat. Effort [h]	Actual Effort [h]	Status
1	Make Reservation	Allow a guest to make a reservation	High	10	8	9	Done
2	Cancel Reservation	As a guest, I want to cancel a reservation	High	7	-	0	Waiting
3	Change Dates Reservation	As a guest, I want to change the starting and ending date of a reservation	Medium	7	-	0	Waiting
4	Run RevPar	As a hotel employee, I can run RevPar report (revenue-per-available-room)	Low	20	14	12	In progress
5		Improve exception handling	Medium	8	-	0	Waiting
		...					Canceled

Scrum Artifacts: User Stories

- A concise, written description of a functionality valuable to either a user (or owner) of a software system
- **Template: As a [some role], I want [something], so that [some value]**
- Describe *who* wants, *what* and *what for* in one sentence
- Examples:
  - "As a registered user, I want to log in, so I can access subscriber content"
  - "As a sales person, I want to see statistics of my performance in graphical charts, so that I monitor my performance"
- User story *does not* define any details of the *implementation*!
- Try to avoid user interface concepts
- Generally written by the Product Owner

Scrum Artifacts: User Stories

- Every user story needs a definition of Done (acceptance criteria)
  - Success – valid user logged in and referrer to home page
  - Failure – display message
    - a) wrong authentication information
    - b) ...

Scrum Artifacts: Release Backlog

- For larger projects, the product may be developed in several releases
  - Like in XP, a release is a productive system delivered to the customer
- A number of items from the Product Backlog is selected for each release in a dedicated release planning meeting and thus transferred to the release Backlog
- Like with the Product Backlog, the Product Owner owns the release Backlog

Scrum Artifacts: Sprint Backlog

- Breaks down items/stories from Product Backlog into tasks
- Tasks are estimated in working hours (1, 2, 4, 8, or 16)
- Tasks longer than 16 hours are broken down
- Estimated work remaining is updated daily
- Needs to be continuously updated
  - Report on completed tasks incl. actual effort spent (daily sprint)
  - New task discovered
  - Correction of effort estimation or available capacity
- Team members sign up for (not yet assigned) tasks
- Only team members can change its Sprint Backlog during a sprint

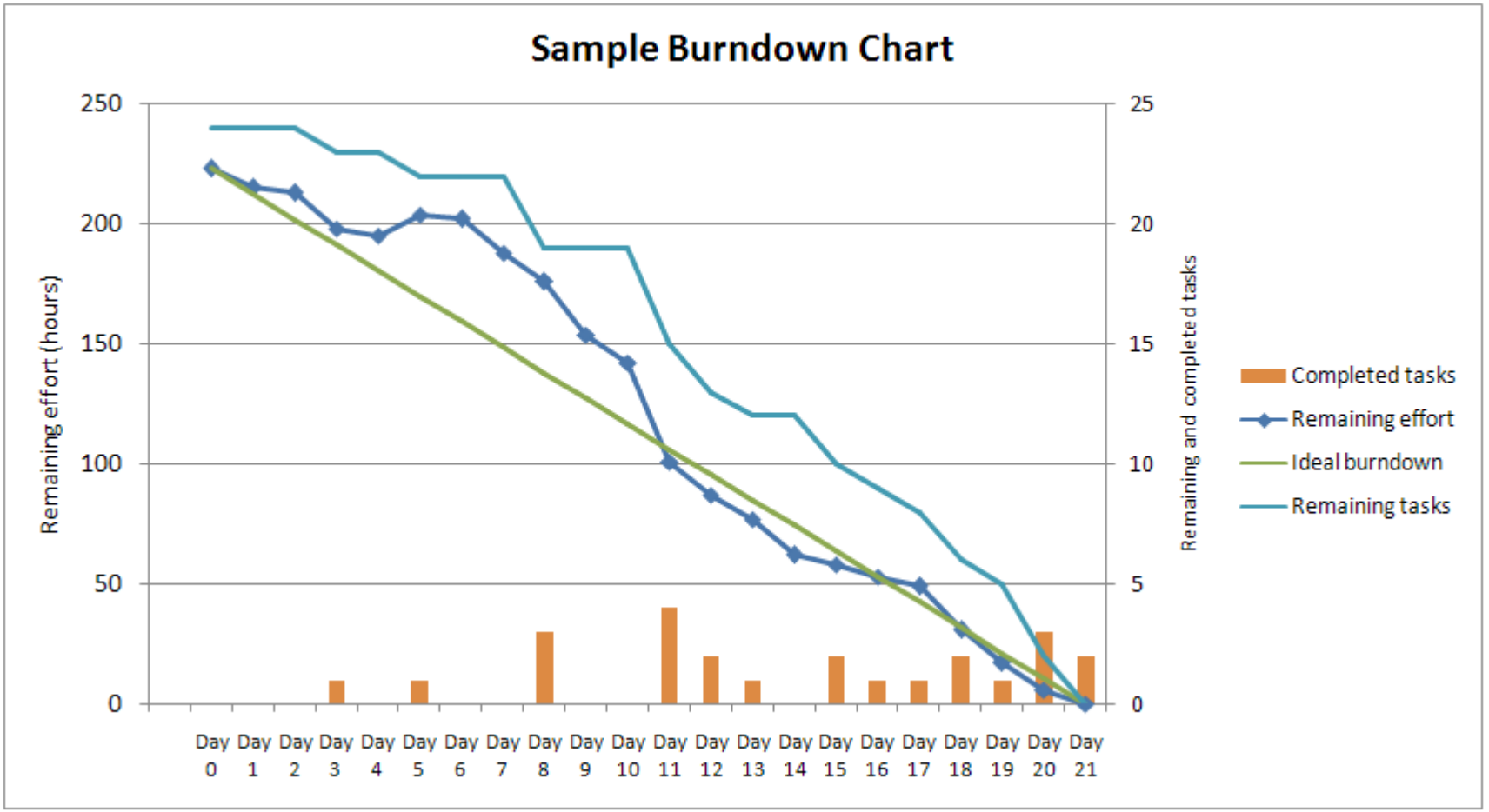
Scrum Artifacts: Sprint Backlog Example

ID	Story Name	Story/Task Description	Priority	Estim. Effort [h]	Updat. Effort [h]	Actual Effort [h]	Status
1	Make Reservation	Allow a guest to make a reservation	High	10	8	9	In progress
1.1		Task 1	High	2	-	-	In progress
1.2		Task 2	Medium	2	-	2	Done
1.3		Task 3	Medium	5	-	-	In progress
1.4		Task 4	Low	1	-	-	Waiting
2	Another Story	Its description	Medium	29	8	9	
2.1							
2.2							
3							

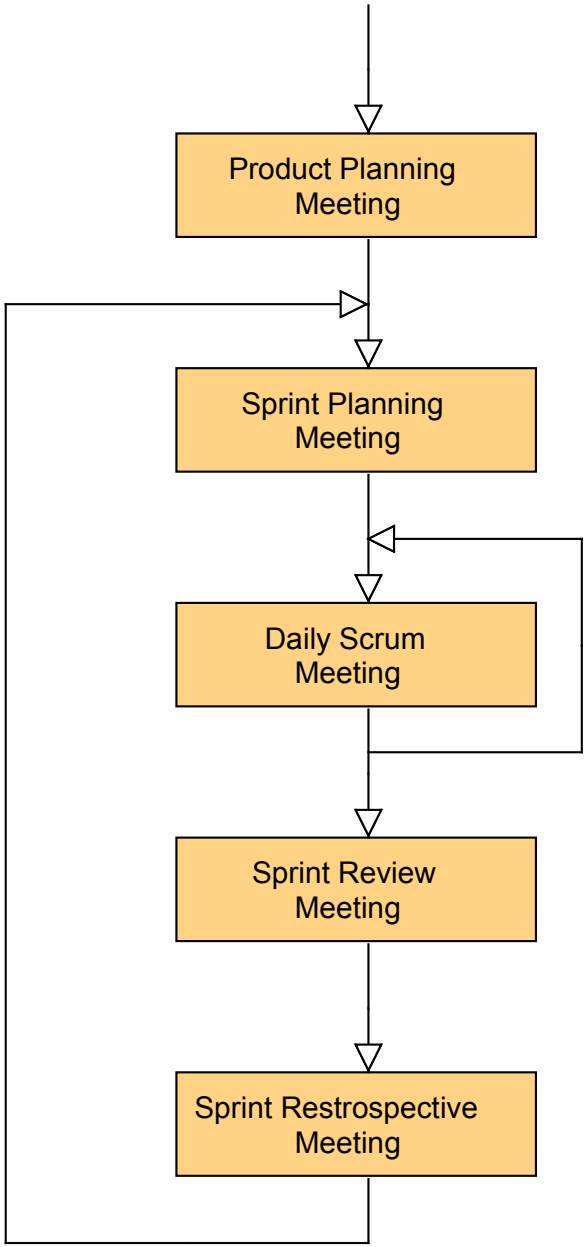
Scrum Artifacts: Burndown Chart

- Visualizes the remaining effort and task on a daily basic
- Updated on a regular basis
- Used to identify potential resource problems
  - Slope progressively computed estimate when the work will be completed
  - Allow resource or Backlog adjustment
- Used in all meetings with various granularities
  - Product
  - Release
  - Sprint

Scrum Artifacts: Sample Burndown Chart



Scrum Meetings



Scrum Meetings: Product Planning

- Attendance: all
- Product Owner presents Product Backlog with all relevant user stories with their priority
- Discussions and clarifications if needed
- Results:
  - Prioritized Product Backlog
    - Specifies what to build
    - Final decision by the Product Owner
  - Vision, high level architecture, most important non-functional requirements

Release planning (if product is to be delivered in releases):

- Select and prioritize items of Product Backlog for the next Release Backlog

Scrum Meetings: Sprint Planning

- Attendance:
  - Developers, Scrum Master, Product Owner
- Create Sprint Backlog for the upcoming sprint
  - Select stories from Product Backlog with the help of the Product Owner
  - Break down stories into tasks
  - Clarify details with Product Owner
  - Estimate effort for each task (1-16 hours)
  - Assign tasks to developers
- Risk assessment
- Result: delivery of the Sprint Backlog

Scrum Meetings: Daily Scrum

- Attendance:
  - Developers, Scrum Master, Product Owner
- “Stand Up” meeting (max. 15 min.)
- Meeting is held a the same place and time every day
- Short daily meeting where each team member answers 3 questions
  - What has been accomplished since last meeting?
  - What will be done before the next meeting?
  - What obstacles are in the way?
- Product Owner and other stakeholders answer questions only
- Results:
  - Updated Sprint Backlog
  - All team members are aware of the current state

Scrum Meetings: Sprint Review

- Attendance: all
- Team presents what it accomplished during the Sprint (or Release/Product)
- Typically takes the form of a demo of new features
- Product Owner accepts or reject the results
  - Reject implies adjustments of Backlog (additional sprints)

Scrum Meeting: Sprint Retrospective

- Take a look at what is and is not working
- Typically 15-30 min.
- How can we be more efficient, avoid to repeat same errors

Scrum Summary

https://www.youtube.com/watch?v=XU0llRltyFM