Scrum Roles: Developer

- Responsible for delivering incremental releases of the product (by completing *User Stories*)

- Team's job is to deliver the stories they committed to for the sprint
  - it's not about your job it's about the job

Scrum Roles: Developer

- Provides the estimates for the *User Stories* in the *Product Backlog*

- Team members:
  - About 3..9 software engineers
  - are self-organizing (they direct their own work)
  - have total authority over how the work gets done, e.g.,:
    - tools, techniques used
    - task assignments (i.e., which team member will work on which task)
    - NB: PO can specify what do to, not how to do it

Scrum Roles: Scrum Master

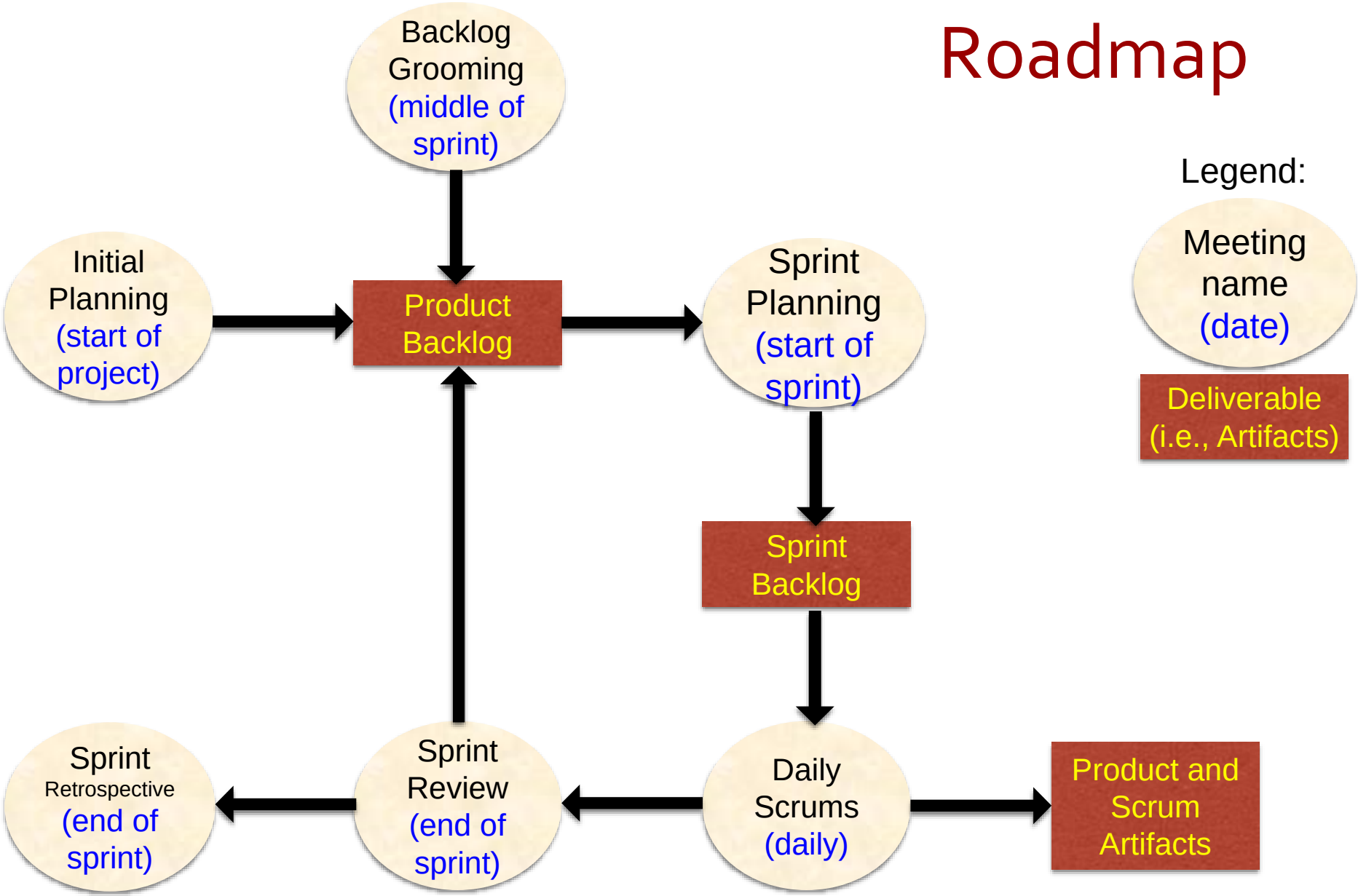
- Facilitator, not an R&D Project Manager or Supervisor, or Boss
  - Not responsible for the hiring and the firing
  - Not responsible for the performance ranking and spanking
  - Not a people manager

Scrum Roles: Scrum Master

- Team's Scrum expert: ensures Scrum methodology/process is followed, e.g.,:
  - Ensures requirements do not change during a sprint
  - Facilitates meetings and ensures agenda is being followed

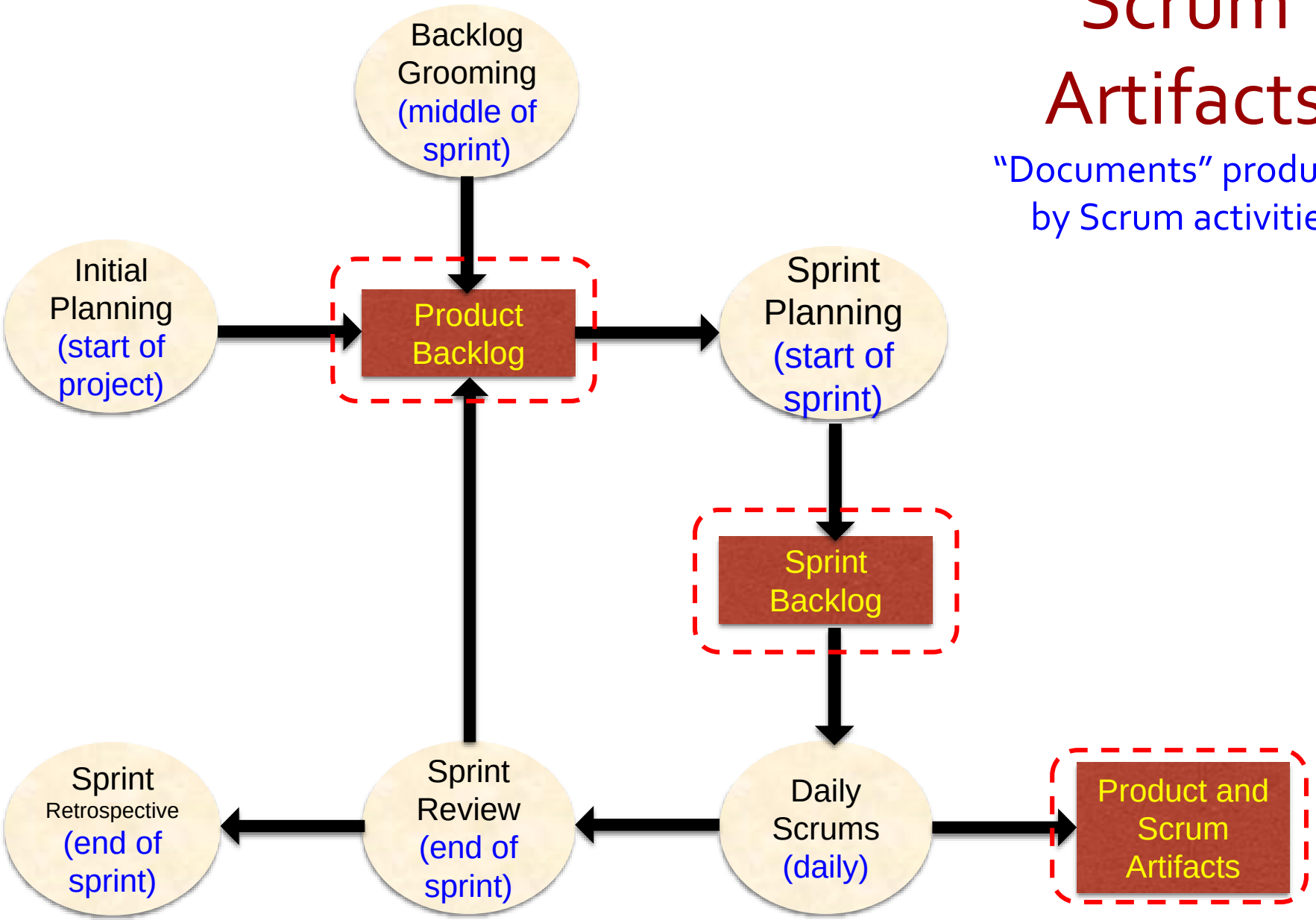
- Removes impediments identified during the *Daily Scrum* meeting

Scrum Process Roadmap

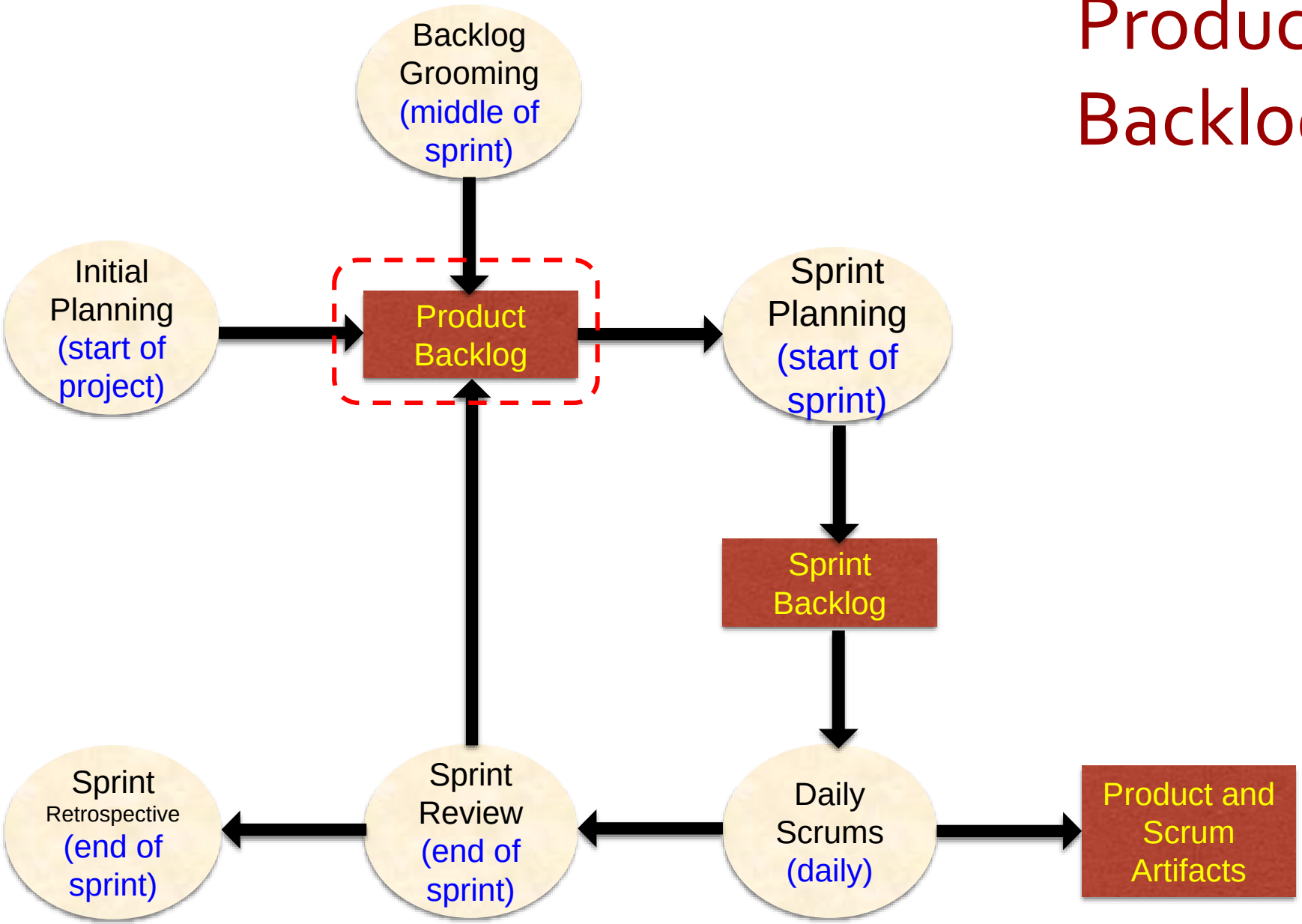


Scrum Artifacts

"Documents" produced by Scrum activities



Product Backlog



Scrum Artifacts: Product Backlog

- List of product requirements and deliverables
  - User Stories (functional and non-functional requirements)
  - In CS471: Unresolved Defect Reports
- Sorted by priority
- Estimated (by Development Team)
- Acceptance Criteria for each User Story

User Stories

- A short description written in the end user's business language (not engineering language*) of what a user needs the product to do

*Why? (classroom only discussion & quiz question)

User Stories

- Describe who, what and why using the
  Role-Goal-Benefit template
  (in customer's business language)

User Stories

- Describe who, what and why using the Role-Goal-Benefit template (in customer's business language)

- Example:

As a new user, I need to create an account so that the application can authenticate existing users when they login

User Stories

- Describe who, what and why using the Role-Goal-Benefit template (in customer's business language)

- Example:

As a new user, I need to create an account so that the application can authenticate existing users when they login

- What do we need to associate with each user story?

User Stories

- Describe who, what and why using the Role-Goal-Benefit template (in customer's business language)

- Example:

As a new user, I need to create an account so that the application can authenticate existing users when they login

- Acceptance Criteria 1 for user story
- Acceptance Criteria 2 for user story
- ...

Acceptance Criteria (AC) for User Stories (US)

- Use the Given-When-Then format. Example:

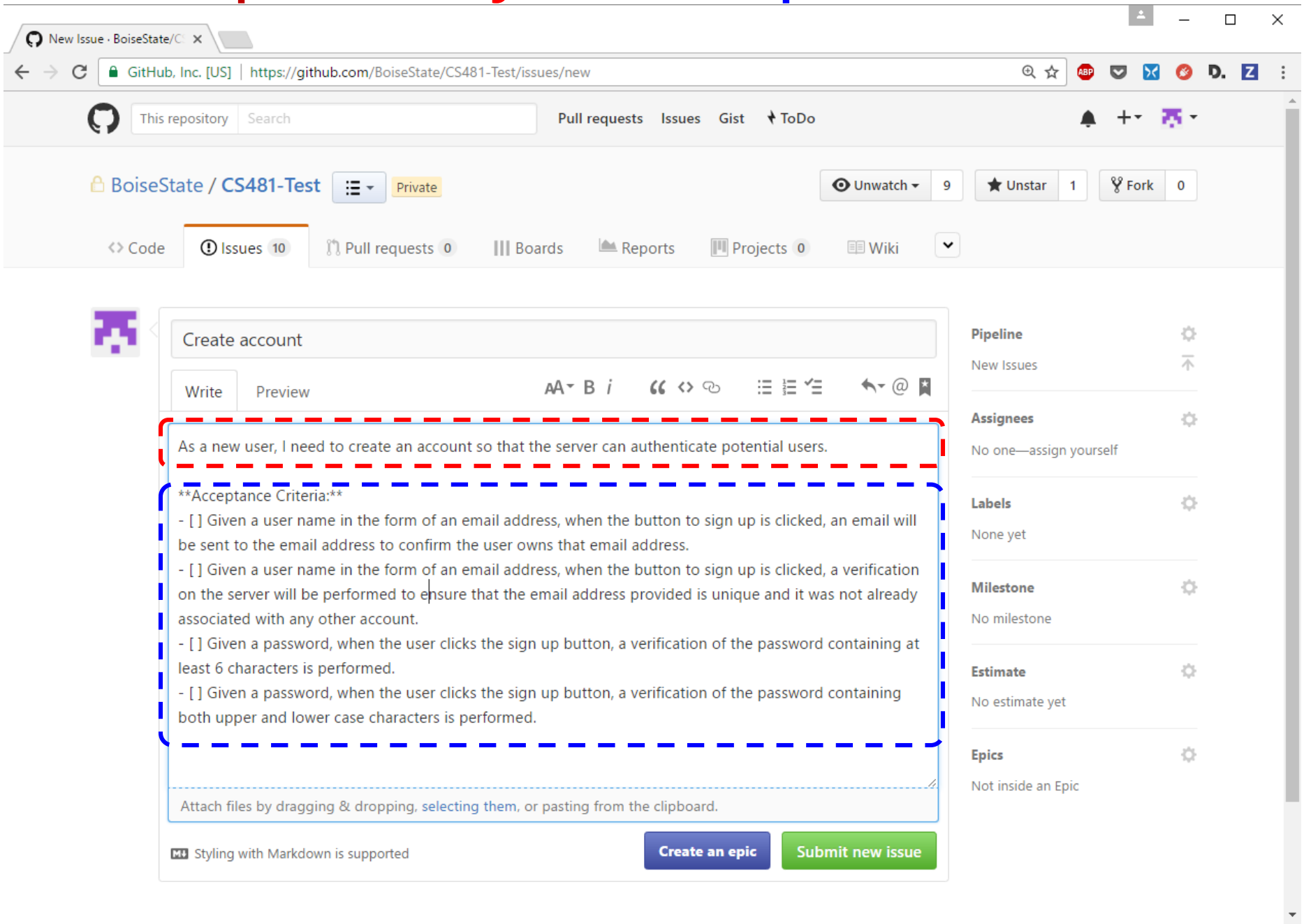
Acceptance Criteria (AC) for User Stories (US)

- Use the Given-When-Then format. Example:
  - Given a user name in the form of an email address, when the button to sign up is clicked, then a verification on the server will be performed to ensure that the email address provided is unique and it was not already associated with any other account.

Acceptance Criteria (AC) for User Stories (US)

- Use the Given-When-Then format. Example:
  - Given a user name in the form of an email address, when the button to sign up is clicked, then a verification on the server will be performed to ensure that the email address provided is unique and it was not already associated with any other account.
- The given part represents the pre-condition of the test
- The when part represents the behavior or the trigger
- The then part describes the expected outcome ("post-conditions" of the test)
- Status must distinguish between passing, failing and untested criteria
- Guideline: You will average 5..10 AC for each US

Example Story + Acceptance Criteria



Defect Report (Review Slide)

- Describes something the product has not correctly implemented

- When writing a bug report, what information should you provide?

- https://developer.mozilla.org/en-US/docs/Mozilla/QA/Bug_writing_guidelines

Defect Report Template

Short descriptive title:

Description

Steps to Reproduce:

1. TBD
2.
3.

Actual Results:

TBD

Expected Results:

TBD

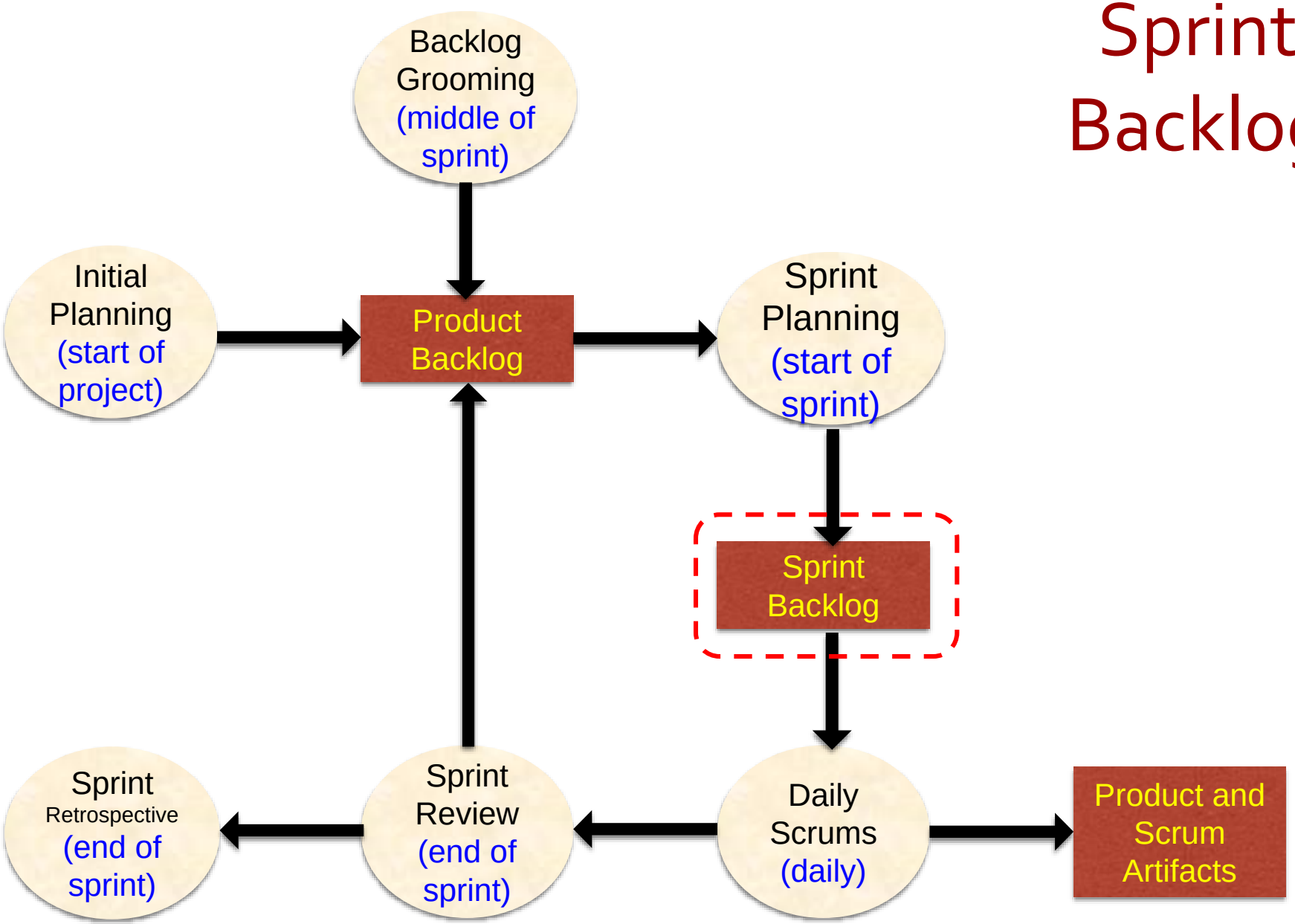
Other notes:

TBD

(In-class exercise only)
Contrast User Stories with Defect Reports

- What's common to both?
- What's the difference?
- Should we track Defect Reports differently than User Stories?

Sprint Backlog



Scrum Artifacts: Sprint Backlog

- List of "work planned" for next sprint
- Populated from highest priority items (usually stories) in Product Backlog

Scrum Artifacts: Sprint Backlog

- Contains both stories and tasks:
  - Stories (written in domain/customer language) are broken down into tasks
  - tasks are written in engineering language (they do not have a template)
  - tasks range between 0.5 hour - 2 days of work
  - tasks are created by the development team

Scrum Artifacts: Sprint Backlog

- Co
  - St
    bu
  - ta
    ha
  - ta
  - ta

- Who assigns developers to tasks?

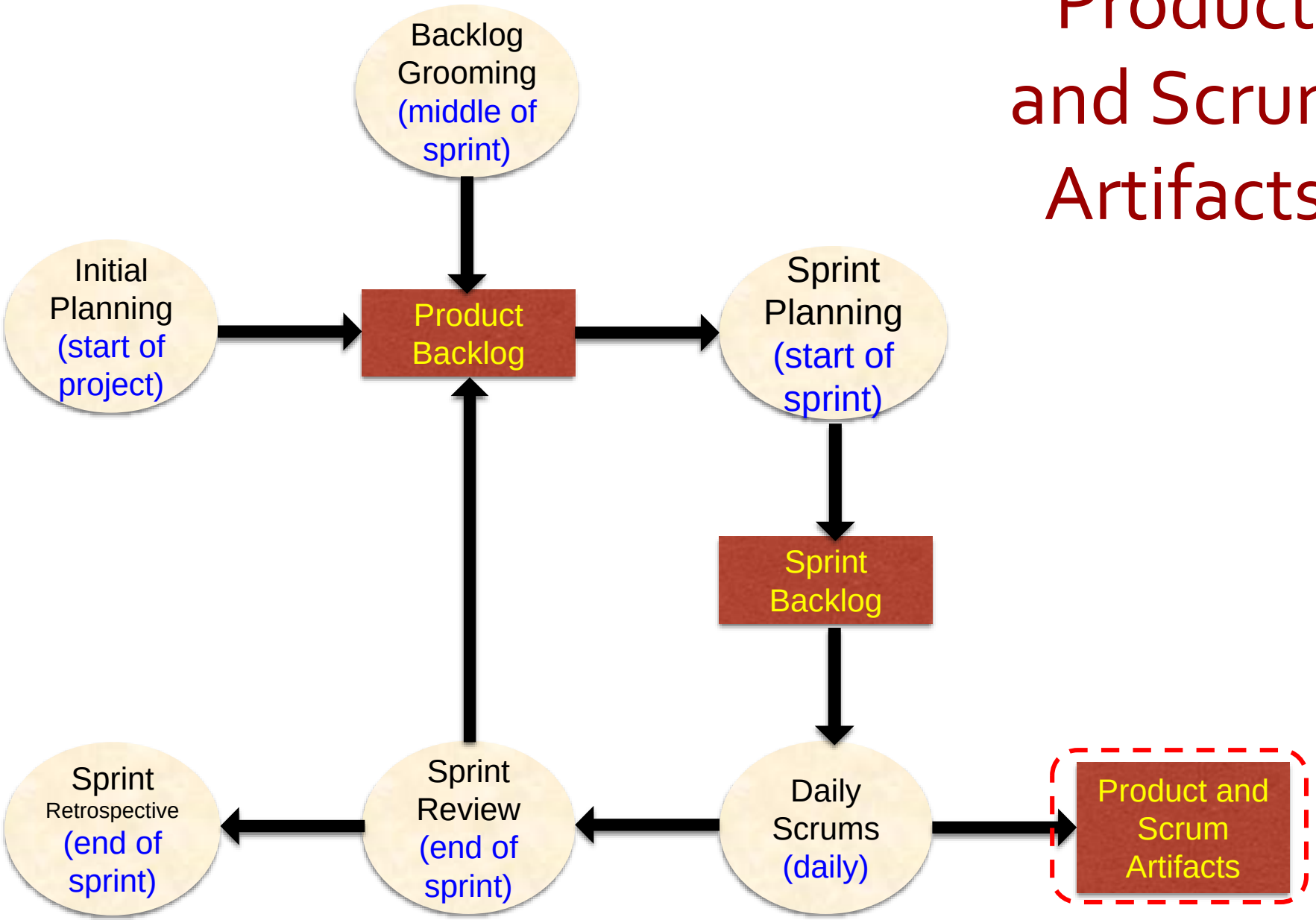
Scrum Artifacts: Sprint Backlog

- Contains both stories and tasks:
  - Stories (written in domain/customer language) are broken down into tasks
  - tasks are written in engineering language (they do not have a template)
  - tasks range between 0.5 hour - 2 days of work
  - tasks are created by the development team
  - developers sign-up for tasks ⇒ they estimate the duration of the task. Why?

Scrum Artifacts: Sprint Backlog

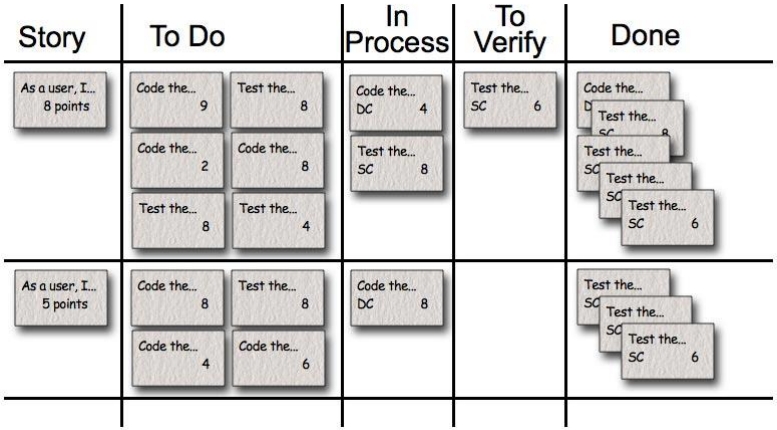
- Contains both stories and tasks:
  - Stories (written in domain/customer language) are broken down into tasks
  - tasks are written in engineering language (they do not have a template)
  - tasks range between 0.5 hour - 2 days of work
  - tasks are created by the development team
  - developers sign-up for tasks ⇒ they estimate the duration of the task. Why?
    - to accurately update the Burndown chart
    - same task can take different amount of time when it is performed by team members with different skills

Product and Scrum Artifacts

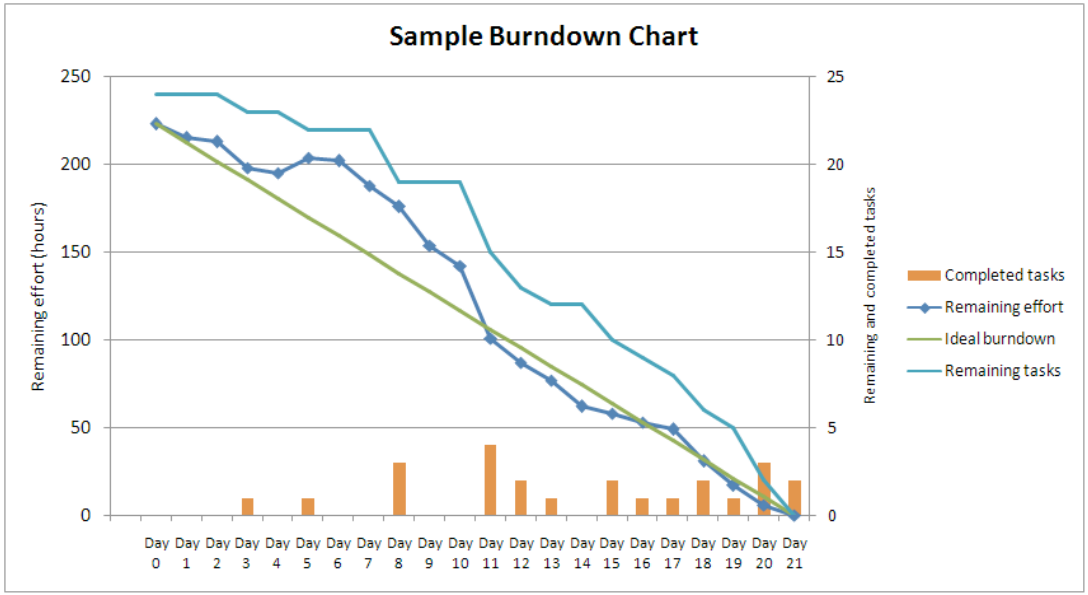


Scrum Artifacts

- Scrum artifacts coordinate and guide the development process

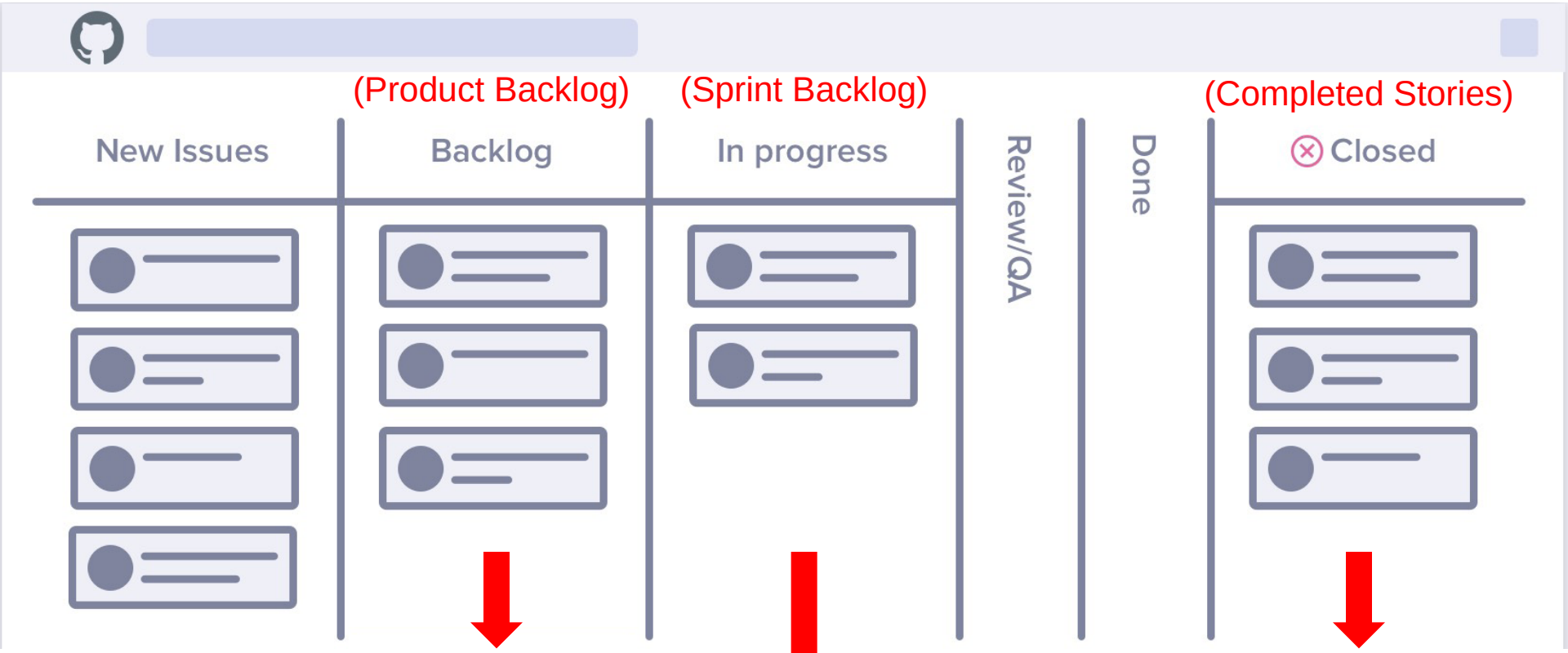


Task Board



Product Backlog
Sprint Backlog
Burndown Charts

Scrum Artifact – Sample Story Board



What features are planned?

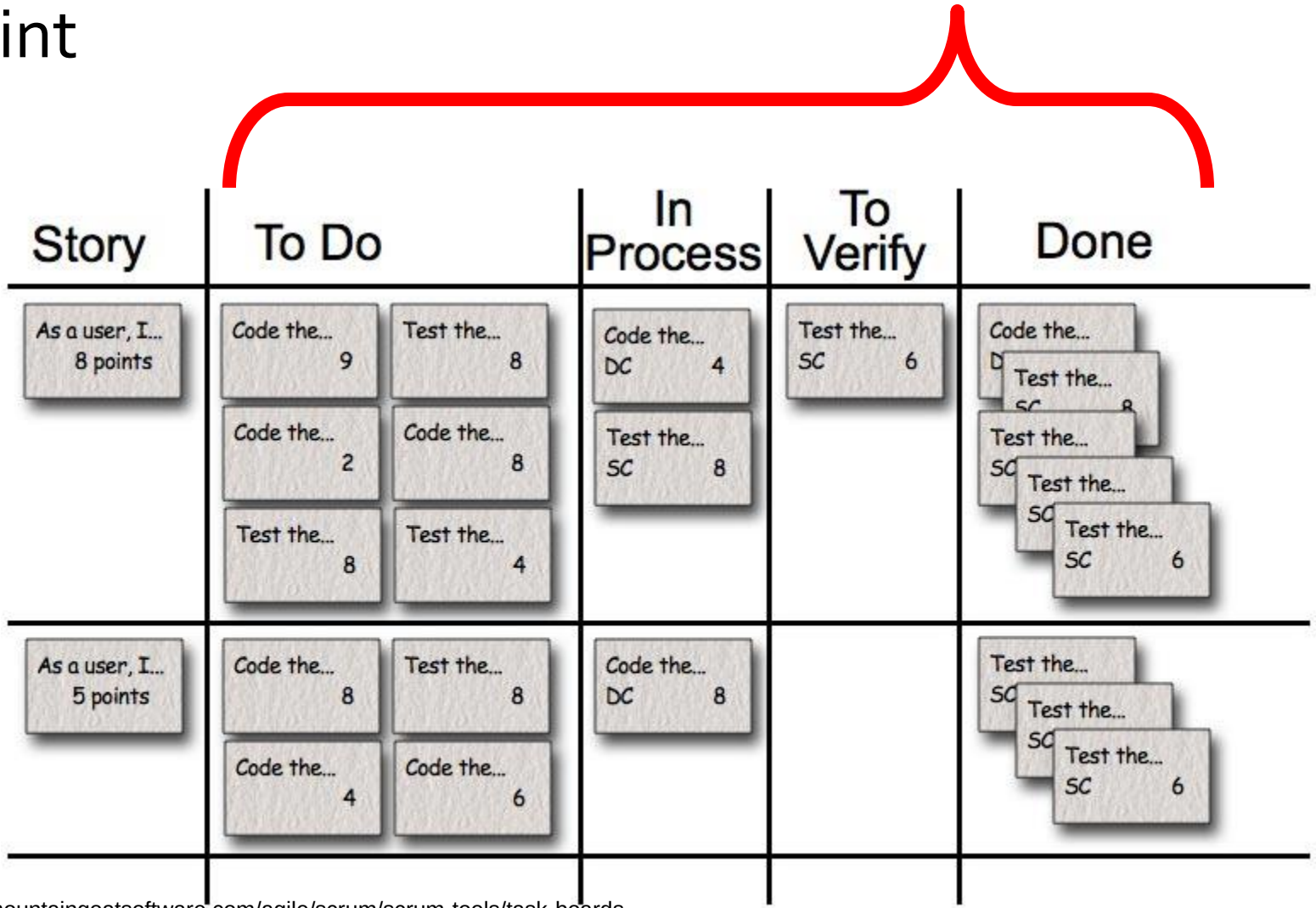
What features are we working on now?

- How will we implement those features?
- Who is or will do the work?

What features are completed?

Scrum Artifacts: Sample Task Board

- Communicates the status (e.g., To Do, In process / progress, to verify, done) of each task in current sprint

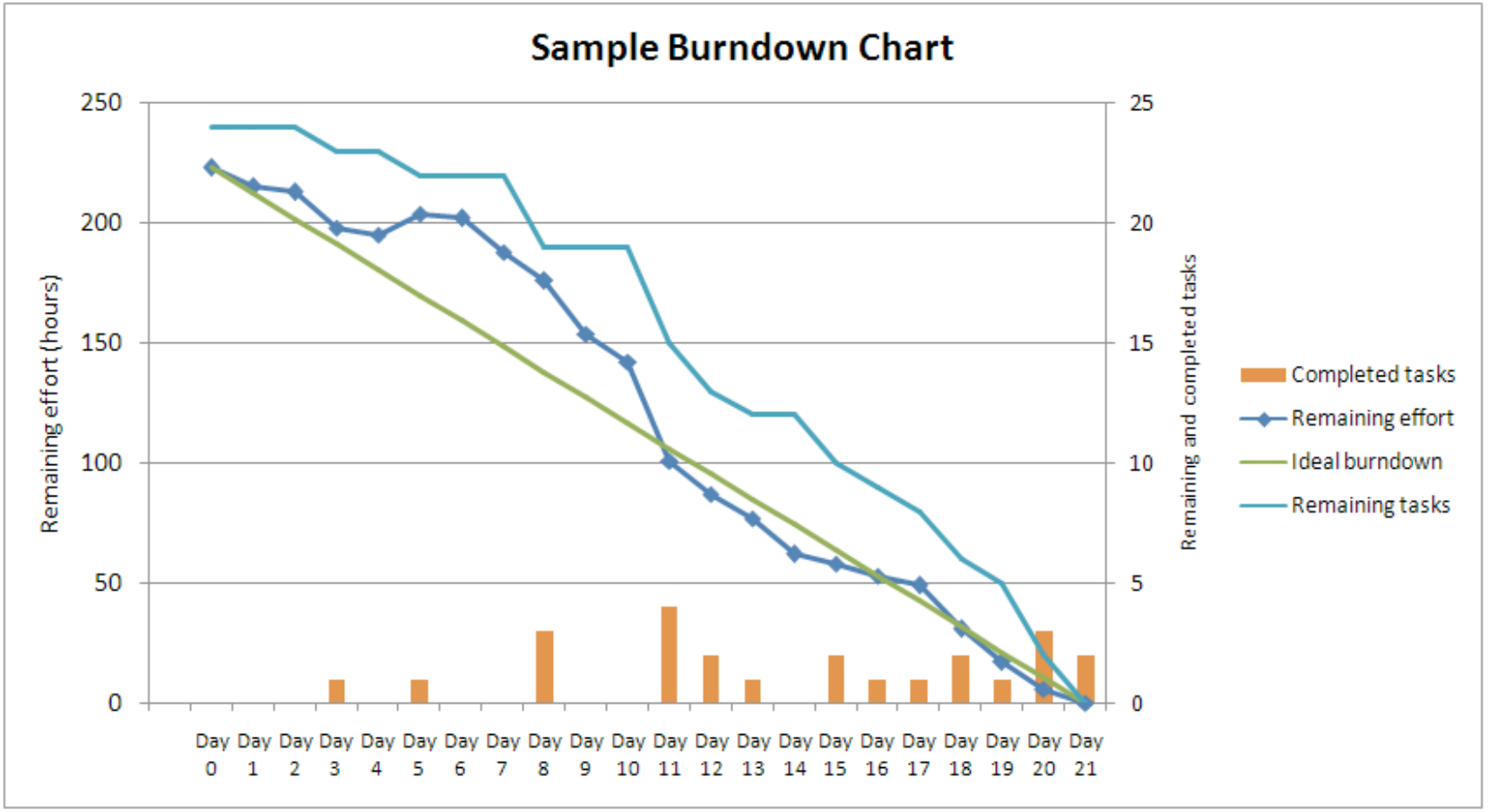


https://www.mountaingoatsoftware.com/agile/scrum/scrum-tools/task-boards

Scrum Artifacts:
(Sprint/Product) Burndown Charts

Scrum Artifacts: Burndown Charts

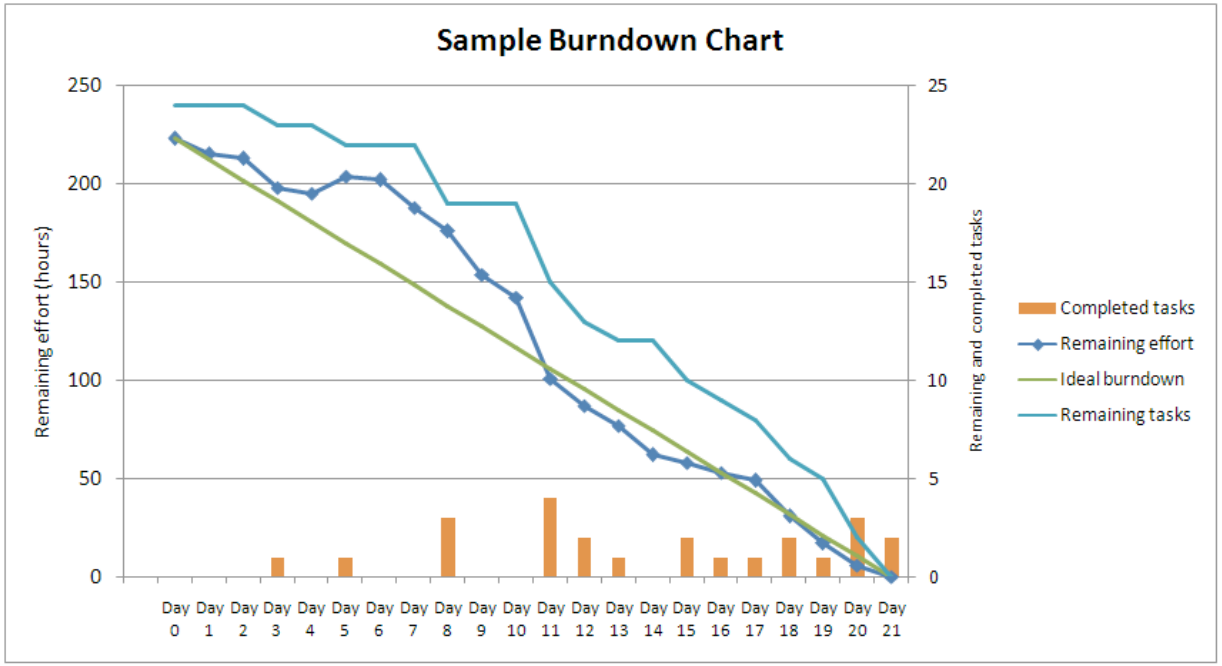
- Chart communicates the amount of work remaining
- Sprint Burndown Chart



(C) Pablo Straub. http://en.wikipedia.org/wiki/File:SampleBurndownChart.png. 2011

Scrum Artifacts: Burndown Charts

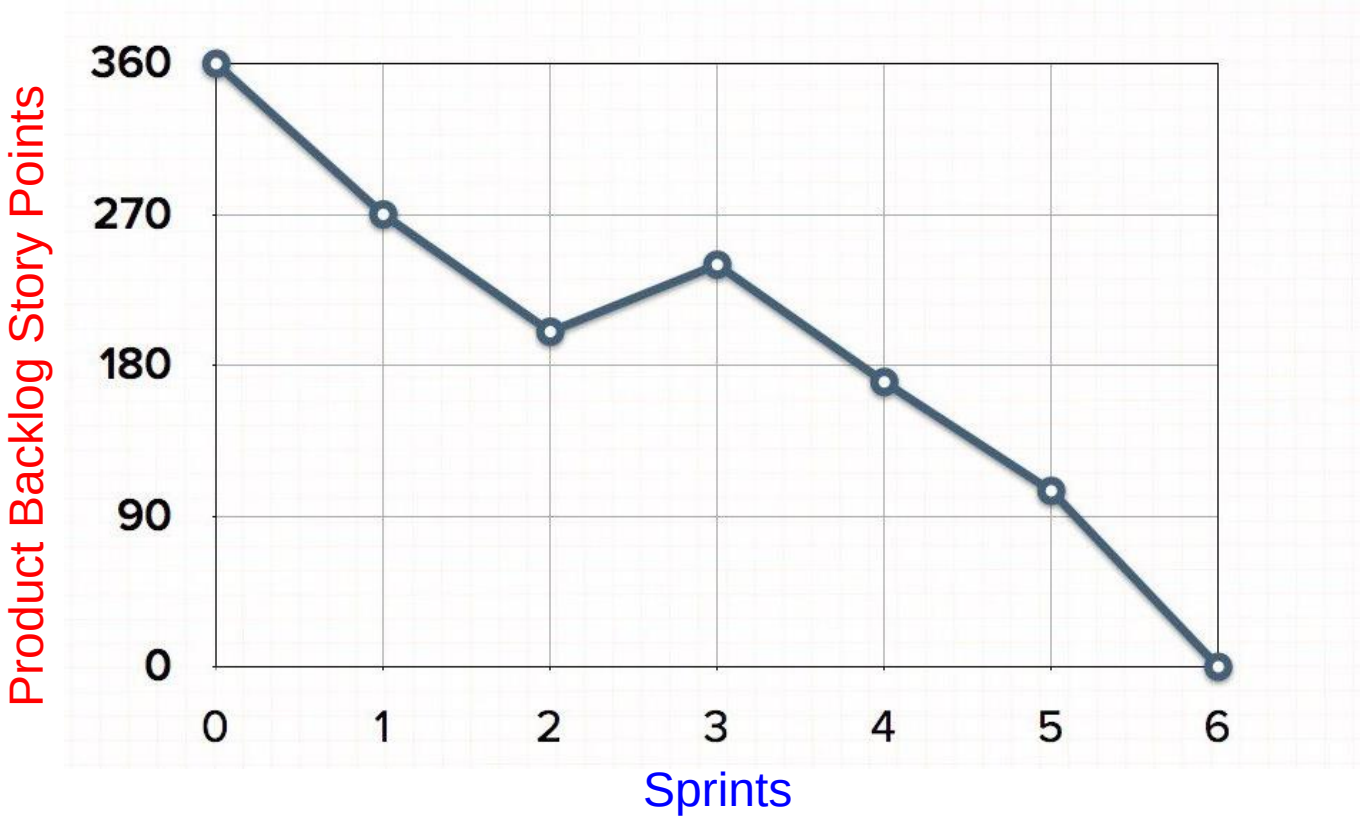
- Chart communicates the amount of work remaining
- Sprint Burndown Chart
- How would a Product Burndown Chart look like?
  - aka Release burndown chart



(C) Pablo Straub. http://en.wikipedia.org/wiki/File:SampleBurndownChart.png. 2011

Scrum Artifacts: Product Burndown Charts

- How would a Product Burndown Chart look like?
  - "days" on the x-axis are replaced by sprints
  - "effort" on y axis is replaced with remaining work until a major release



https://www.mountaingoatsoftware.com/agile/scrum/scrum-tools/release-burndown