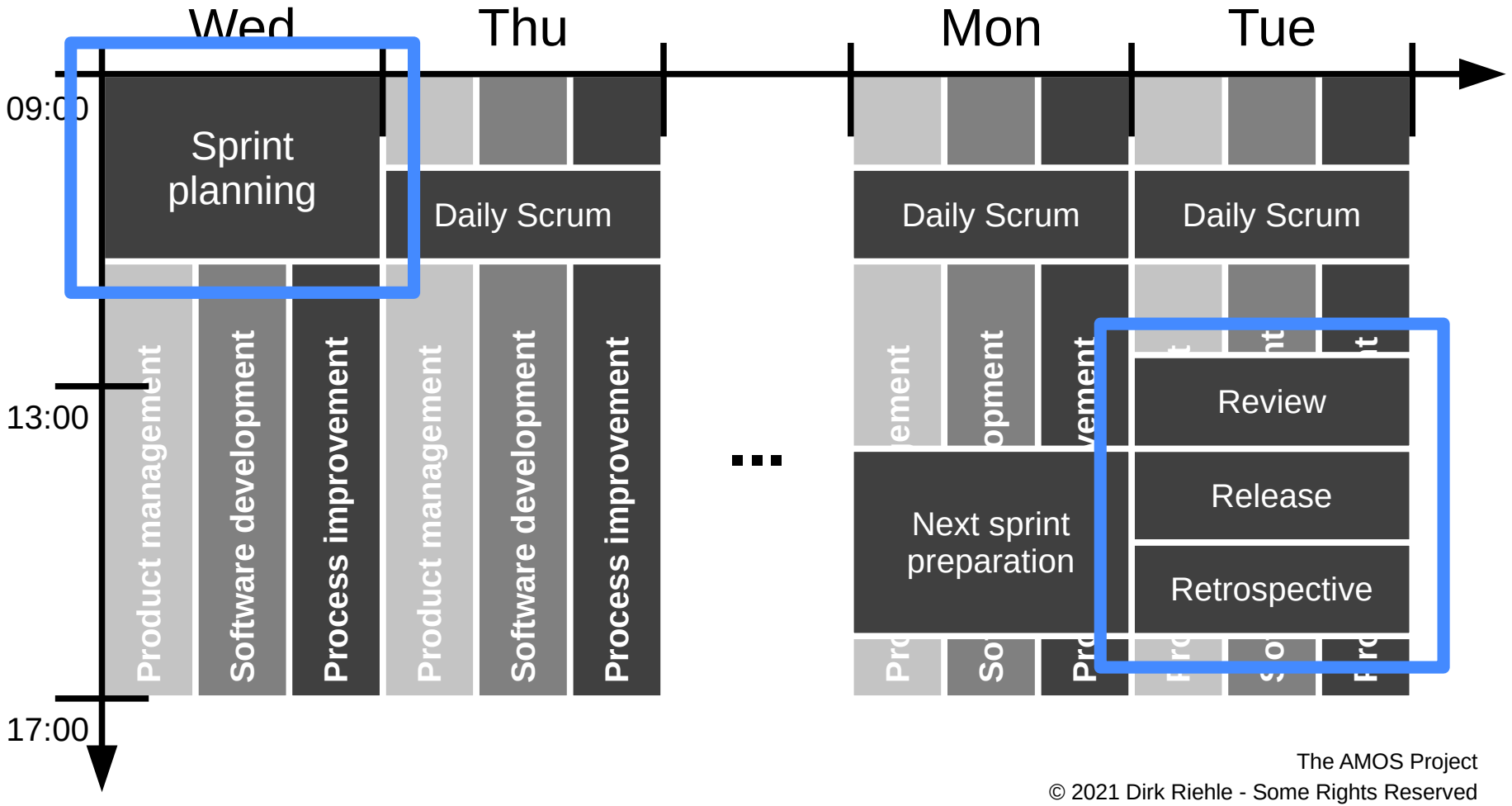
The Team Meeting

Prof. Dr. Dirk Riehle

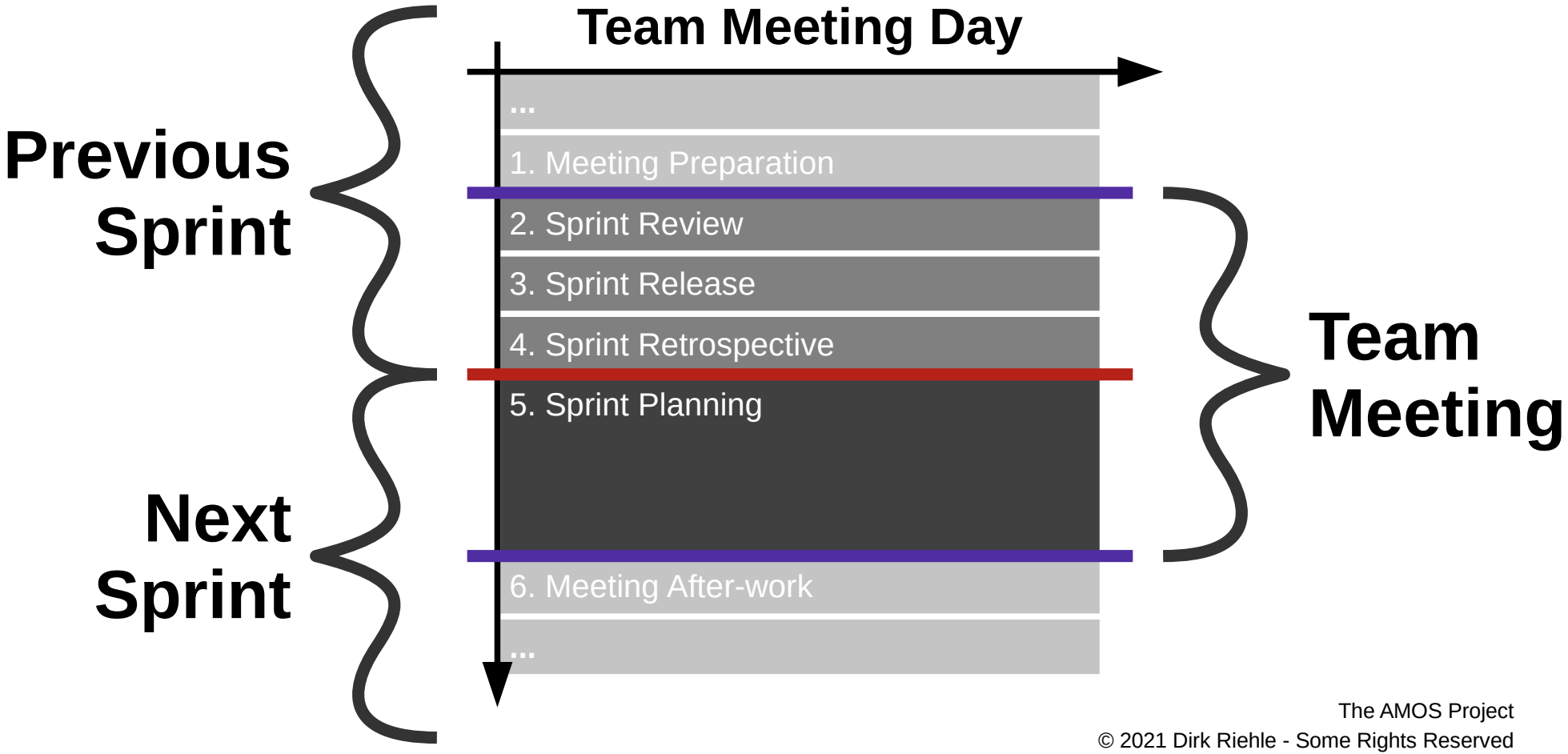
Friedrich-Alexander University Erlangen-Nürnberg

AMOS B02

The Scrum Sprint (Recap)



The AMOS Team Meeting



1. Meeting Preparation

- This sprint's **release manager**
  1. Ensure that a working demo system will be available
  2. Tag release candidate with **sprint-xx-release-candidate**
     - where xx is your sprint number (see project schedule)

- The next sprint's **product owner**
  - Ensure that product backlog is ready for sprint planning
    - Include new feature requests
    - May include bugs as issues
    - May include refactorings
  - This should be done in a next sprint preparation meeting

2a. Sprint Review

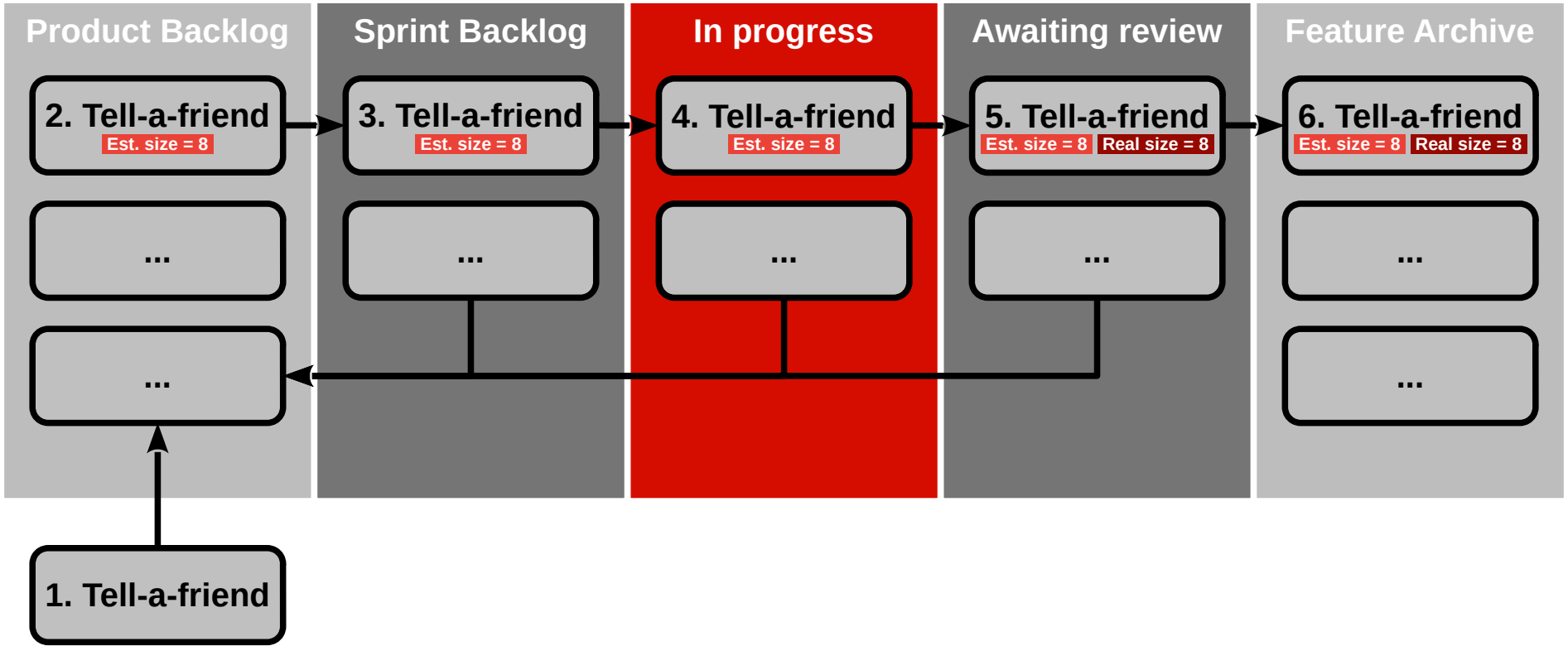
- This sprint's **release manager**
  1. Check-out fresh code base using release candidate tag
  2. Compile, build, and run tests for release candidate
  3. Deploy release candidate to test environment
- The release manager does not run the review

2b. Sprint Review

- This sprint's **product owner**
  - Walk through sprint backlog feature by feature
    1. Ask developer to demo current feature under review
    2. Check fulfillment of acceptance criteria
    3. Check fulfillment of definition of done, if required
    4. Check other criteria incl. logging output for problems
    5. If successfully implemented, move feature to feature archive
    6. If not successfully implemented, move feature back to product backlog

- **Software developer**
  1. Demo feature as requested by product owner
  2. Answer questions about feature design and implementation
  3. Provide real size as determined after implementation

Project Board and Feature Life-Cycle



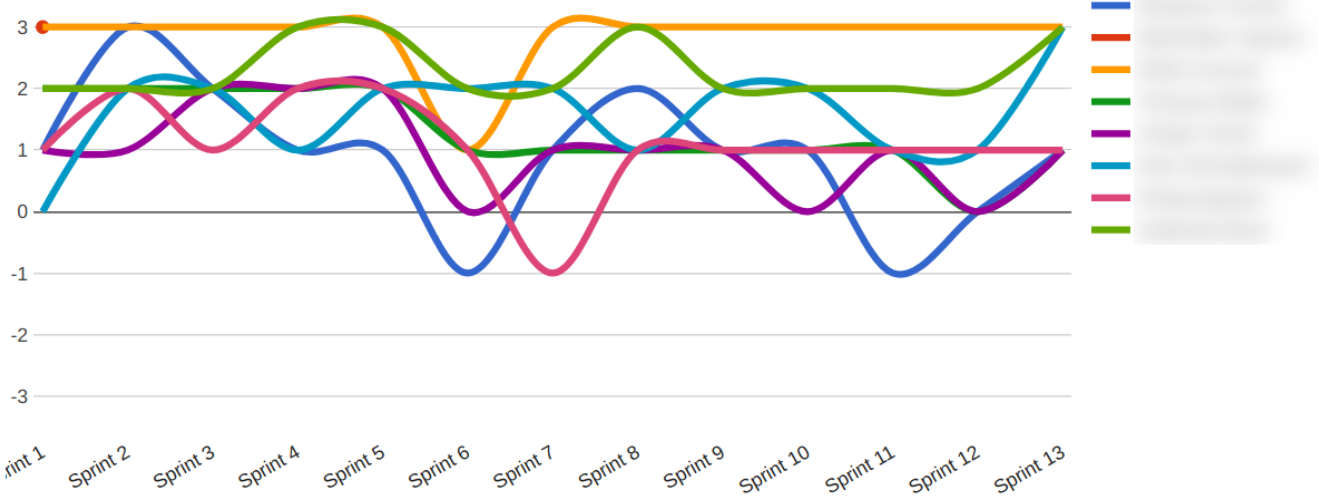
3. Sprint Release

- This sprint's **product owner**
  - Decide whether release candidate should be released
    - Only in case of significant regression should you not release
    - Later in the course you will use a definition of done
    - Specifics depend on type of release
  - Consult with software developers if necessary
- This sprint's **release manager**
  - If the release candidate is to be released
    1. Deploy sprint release to operations environment
    2. Tag release with **sprint-xx-release** where xx is your sprint number
  - If you maintain a change log (optional)
    - Update change log with release information

4. Sprint Retrospective

1. This sprint's **Scrum master**
   - Review this sprint's impediments
     - Report on progress
     - Review remaining problems
2. Next sprint's **Scrum master**
   - Make roll call as to new impediments
     - Take note of people's impediments
     - Allow everyone to make a suggestion
   - Note impediments in impediments backlog
3. Everyone
   - Answer to happiness index

Happiness Index



Current Sprint

13

Participants

Standup Emails

Done

Plans

Challenges

5. Sprint Planning

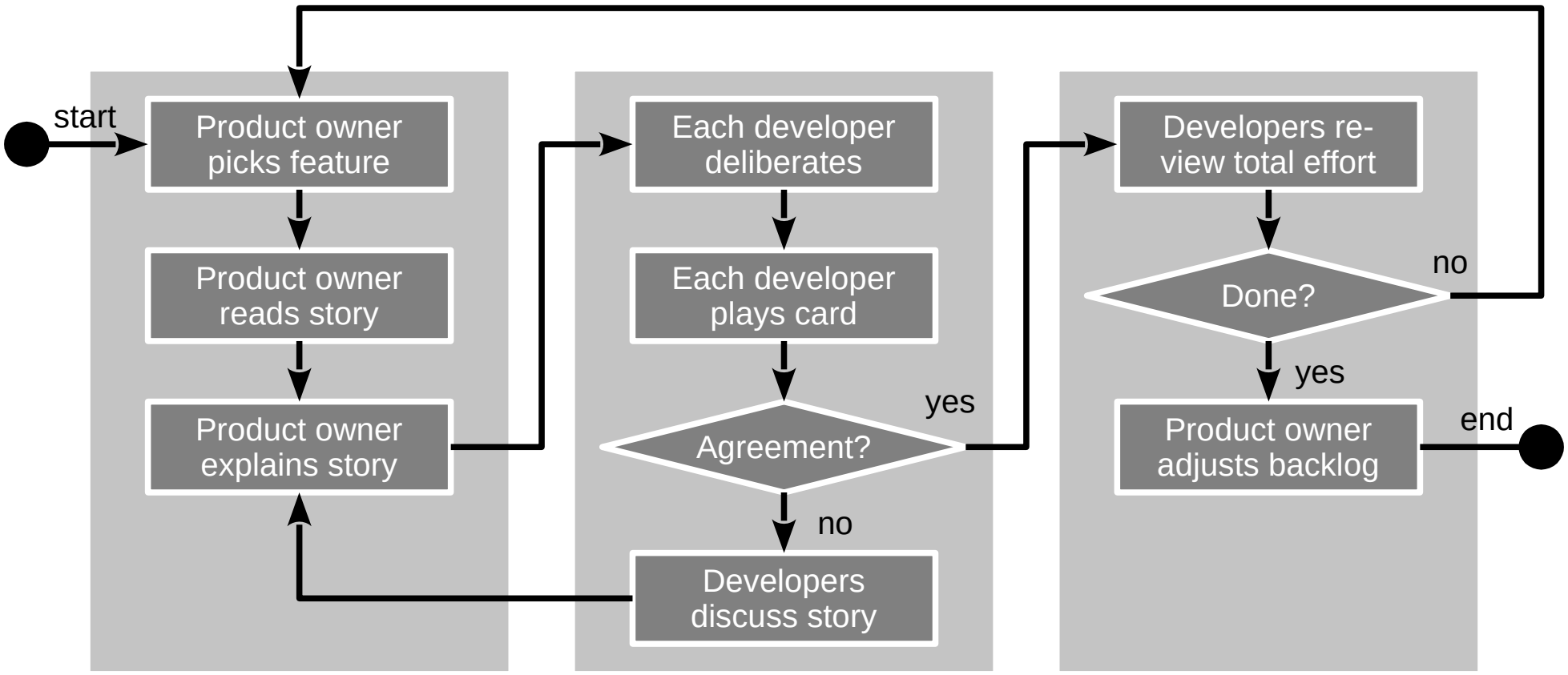
- This (now possibly new) sprint's **product owner**
  1. Reprioritize backlog items, if necessary, on-the-fly
  2. Work through top-prioritized backlog items one-by-one until finished
     - For each product backlog item, explain it, ask developers to estimate and commit
     - You are finished, if the team does not want to take on more backlog items
- Software developer
  - Estimate size of each backlog item using planning poker
  - After planning, commit to backlog items put into sprint backlog
- Everyone
  - Make sure roles for sprint are clear to everyone

Story Points

- Story point
  - An arbitrary numeric measure of size of a given feature
- Properties
  - Is a measure of size, not of effort or duration
  - Measured in non-linear increments, forcing choice
  - Is socially agreed upon, depends on team estimation history
  - Is independent of a particular person (and their skills)
  - Is mapped to time using the team's velocity (development speed)

Points	Meaning
0	No size
1	Trivial size
2	Small size
3	Medium size
5	Large size
8	Very large size
13	Too large (size)

Flow Diagram for Sprint Planning [1]



[1] A simple planning poker cards replacement can be found in the planning documents

6. Meeting After-work

- This sprint's **product owner**
  - Update planning documents to consistent state
    - Clean up product and sprint backlog
    - Ensure feature archive is current
    - Update release plan with actual effort for sprint
- **Software developers**
  - Plan programming tasks (1 feature = 1+ tasks)
    - Agree on which developer(s) work(s) on which tasks
    - If pair programming, ensure you document the pair
    - May happen right after meeting or during sprint
- This sprint's **Scrum master**
  - Work on resolving impediments during sprint
  - Document resolutions in impediments backlog

Thank you! Questions?

dirk.riehle@fau.de – https://oss.cs.fau.de

dirk@riehle.org – https://dirkriehle.com – @dirkriehle

Legal Notices

- License
  - Licensed under the CC BY 4.0 International license
- Copyright
  - © 2010-2021 Dirk Riehle, some rights reserved