

# Agile in large organizations

Thomas Luvö @tomluvoe [tom@samoh.se](mailto:tom@samoh.se)

# Thomas Luvö

- VP Synective Labs
- 10 years experience from the telecommunication industry, software development
- 2 years experience of agile transformation in a large organization

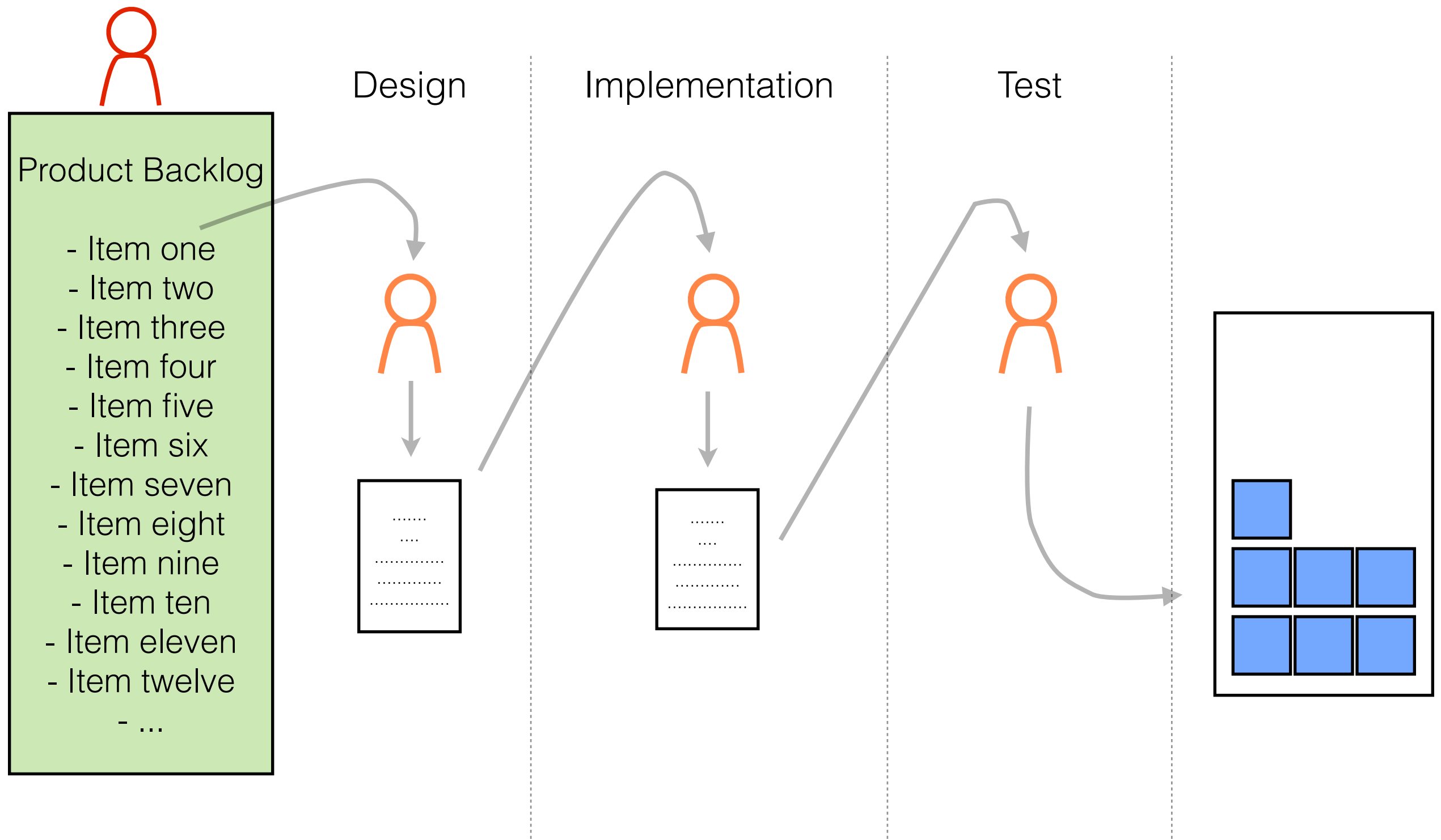
# Today's lecture

- Scaling Agile
- Agile transformation

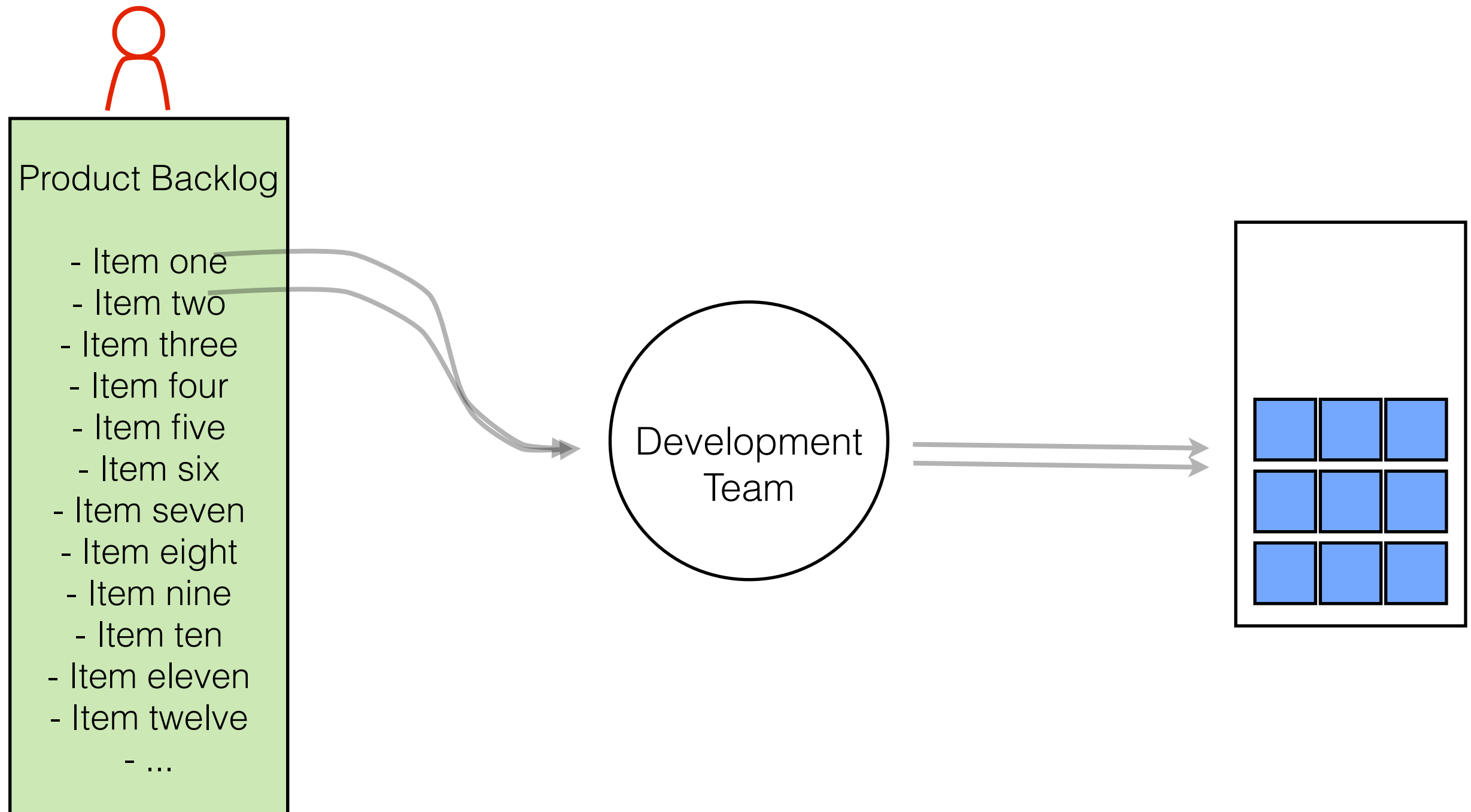
# Scaling Agile

1. Complex products
2. Large organizations

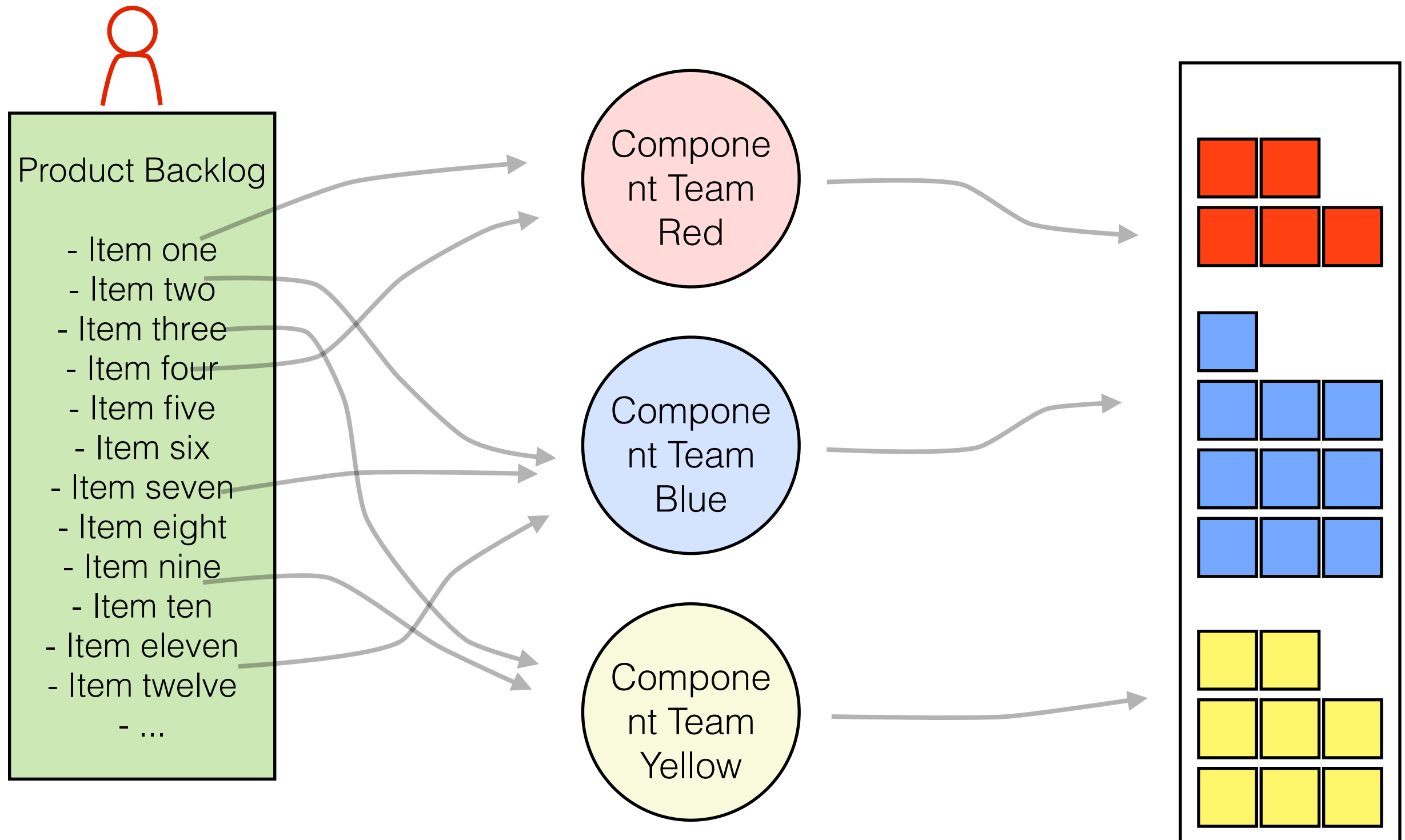
# “Waterfall”



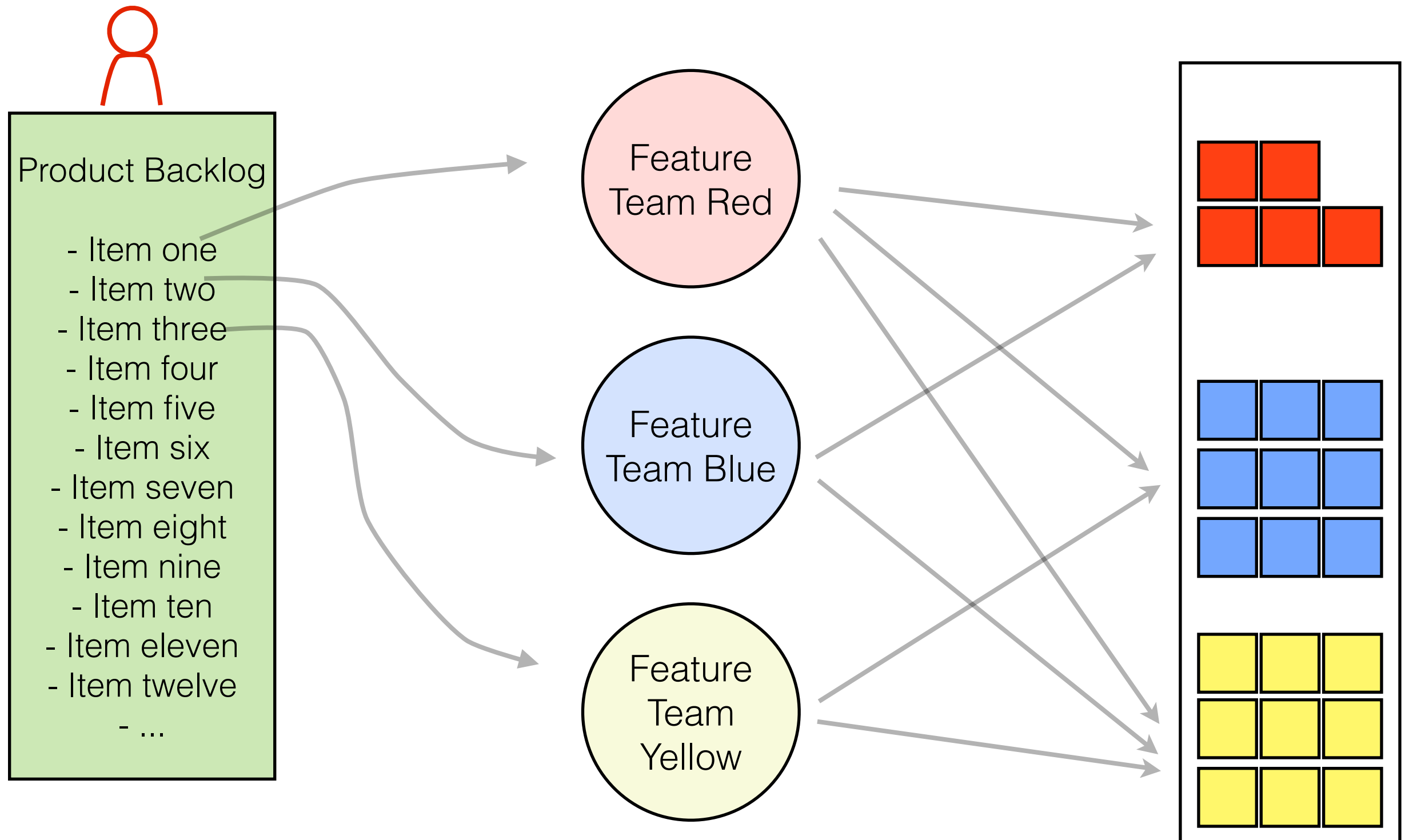
# Single XFT



# Component teams



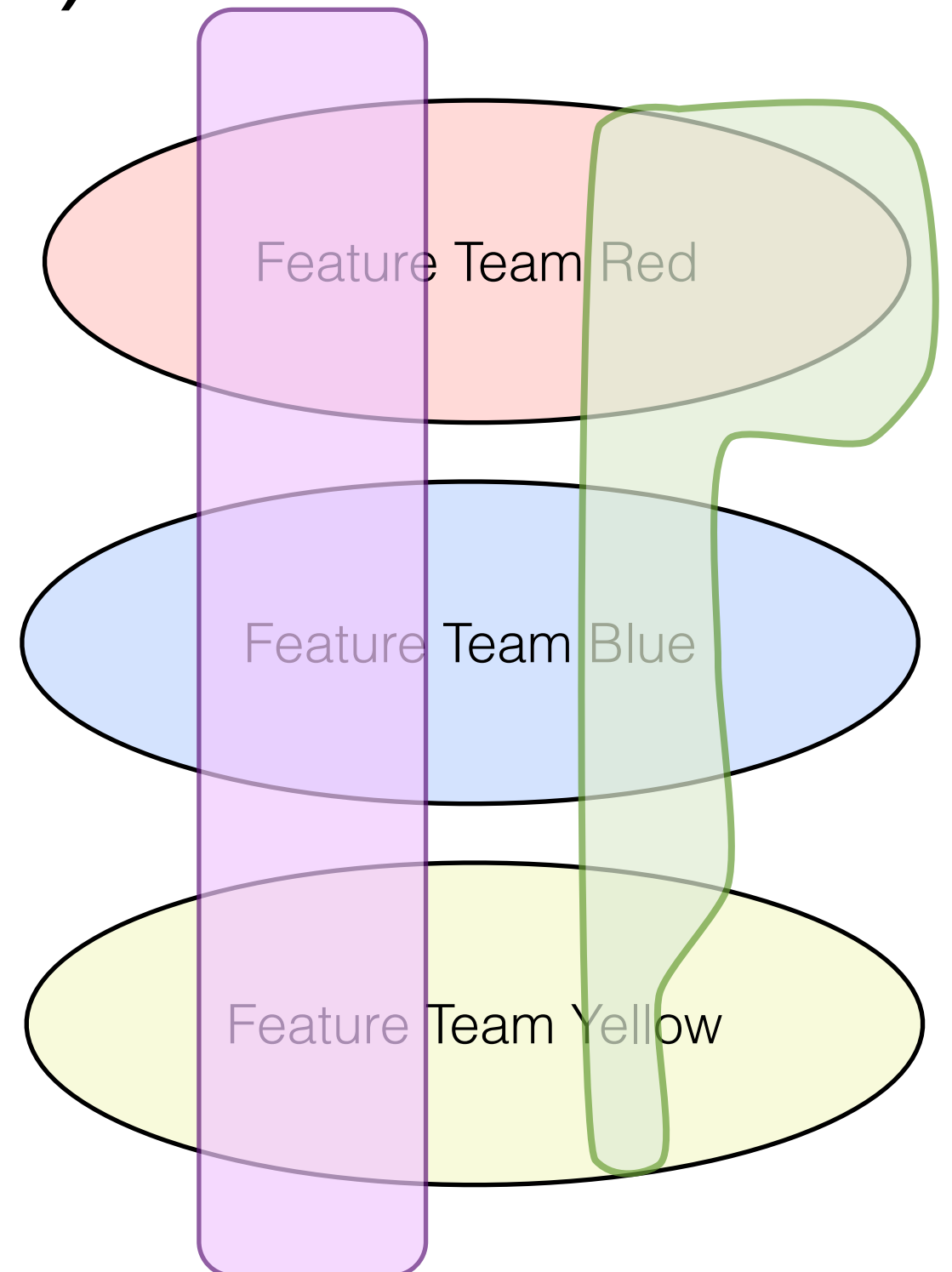
# Feature teams





# Community of practice (CoP)

- How do you share knowledge, experiences, solutions over each function (e.g. testing, software development)?
- How do you share information and knowledge for each product component?



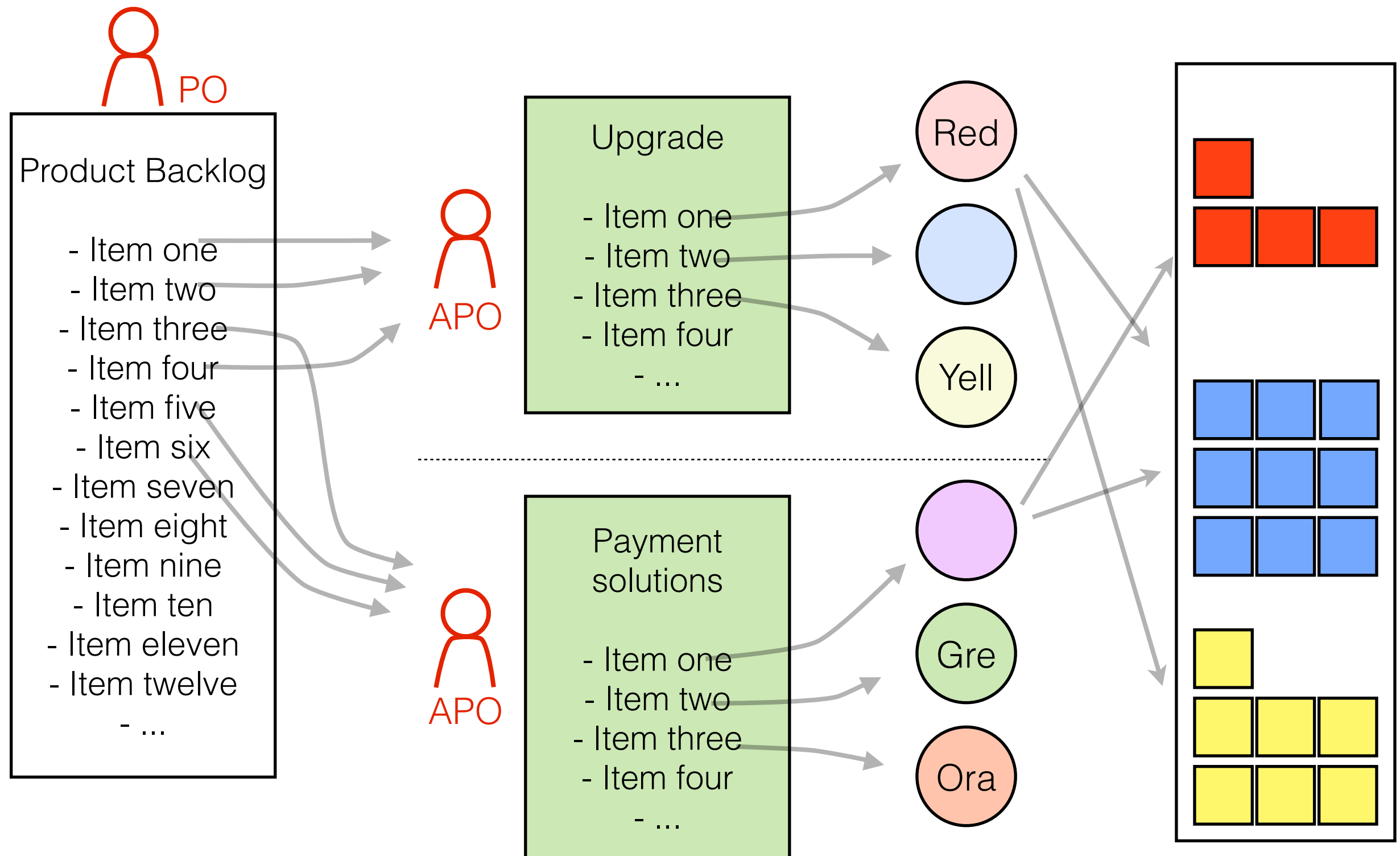
# For even larger products

- When we have a large product and many feature teams, we will reach a point where
  - It will be difficult for a single feature team to be able to work in the whole product
  - It will be difficult for a single product owner to work with so many teams in parallel
- Necessary to somehow divide the product backlog

# Requirement areas

- Customer centric areas of the product backlog
- Example
  - Upgrade
  - Performance
  - Reliability
  - Ads
  - Payment solution
  - User support system
  - ...

# Requirement areas



# Agile transformation

- 6 steps to transform a waterfall organization to agile

# Decisions

- People!
- Inside-out, or outside-in
- Getting the actual buy-in
  - Kuebler-Ross model

# 1. Investigate the value-flow

- Find your value flow, identify the most critical paths and plan for how they will be implemented in agile

## 2. Focus on the continuous integration and make it work

- Without continuous integration, a set of teams can not deliver to development branch with quality every week
- Without continuous integration the delivery chain will be broken



# 3. Transform product development to agile/scrum

- Set up the product owners who define the initial backlogs
- Transform system/design/verification divisions to cross-functional scrum teams (teach all how to do scrum)
- Start feeding the scrum teams the features from the backlog

## 4. Change the projects to agile, meetings, reporting, steering

- Remove old, mandatory progress and status reports
- Status reports are replaced by backlog status and Scrum boards
- Burndown chart will predict project's vital dates
- Steer with backlog priority

# 5. Common mindset everywhere!

- Champions

# 6. Transform roles

- Software designers - same same.
- Project managers - become product owners, or similar.
- Managers - ?
  - No more telling.
  - Vision, planning, Gemba walk, management daily meeting/scrum