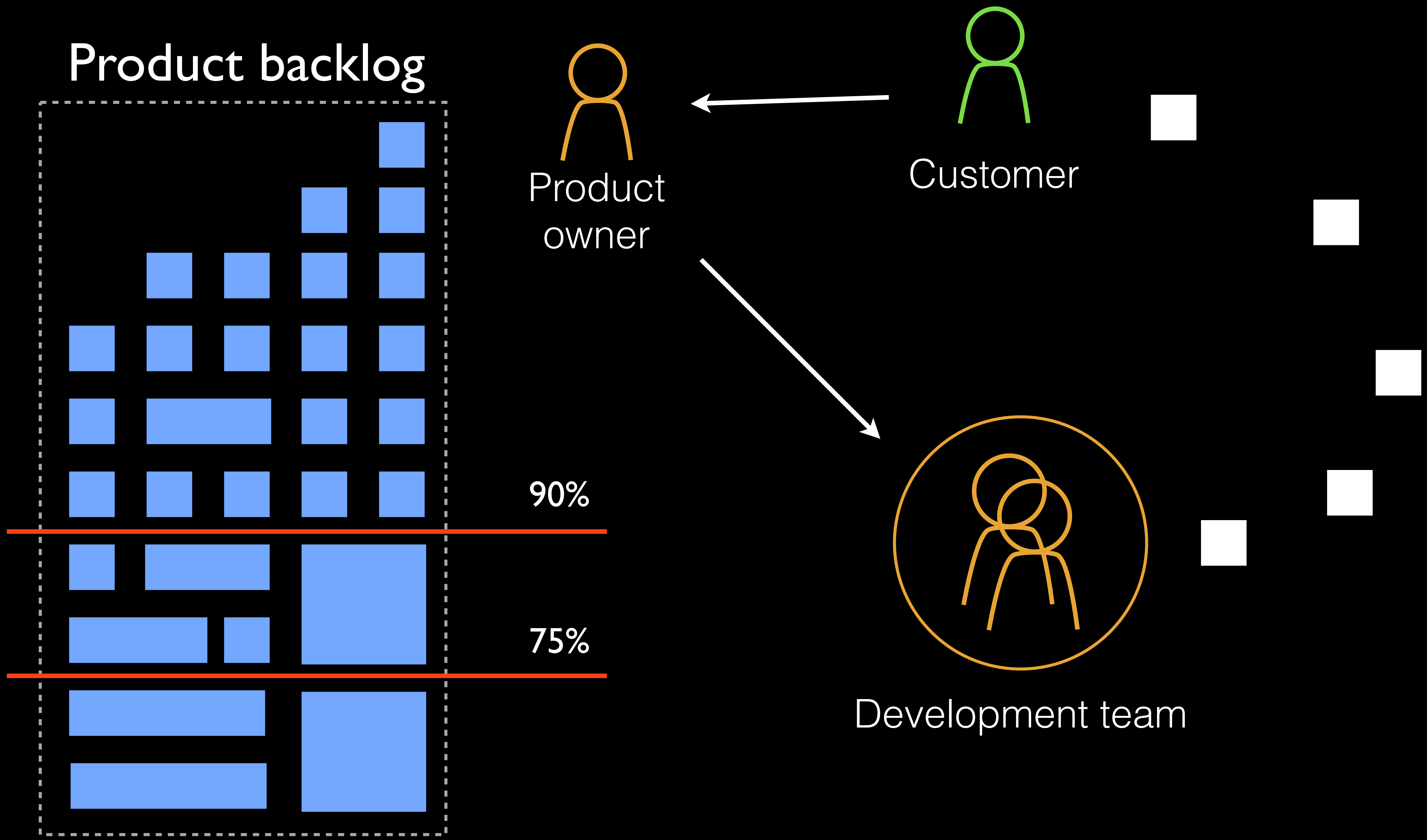
Software Engineering Project

Thomas Luvö tom@samoht.se @tomluvoe

Scrum (Agile)



Product backlog

Product owner

Customer

Development team

90%

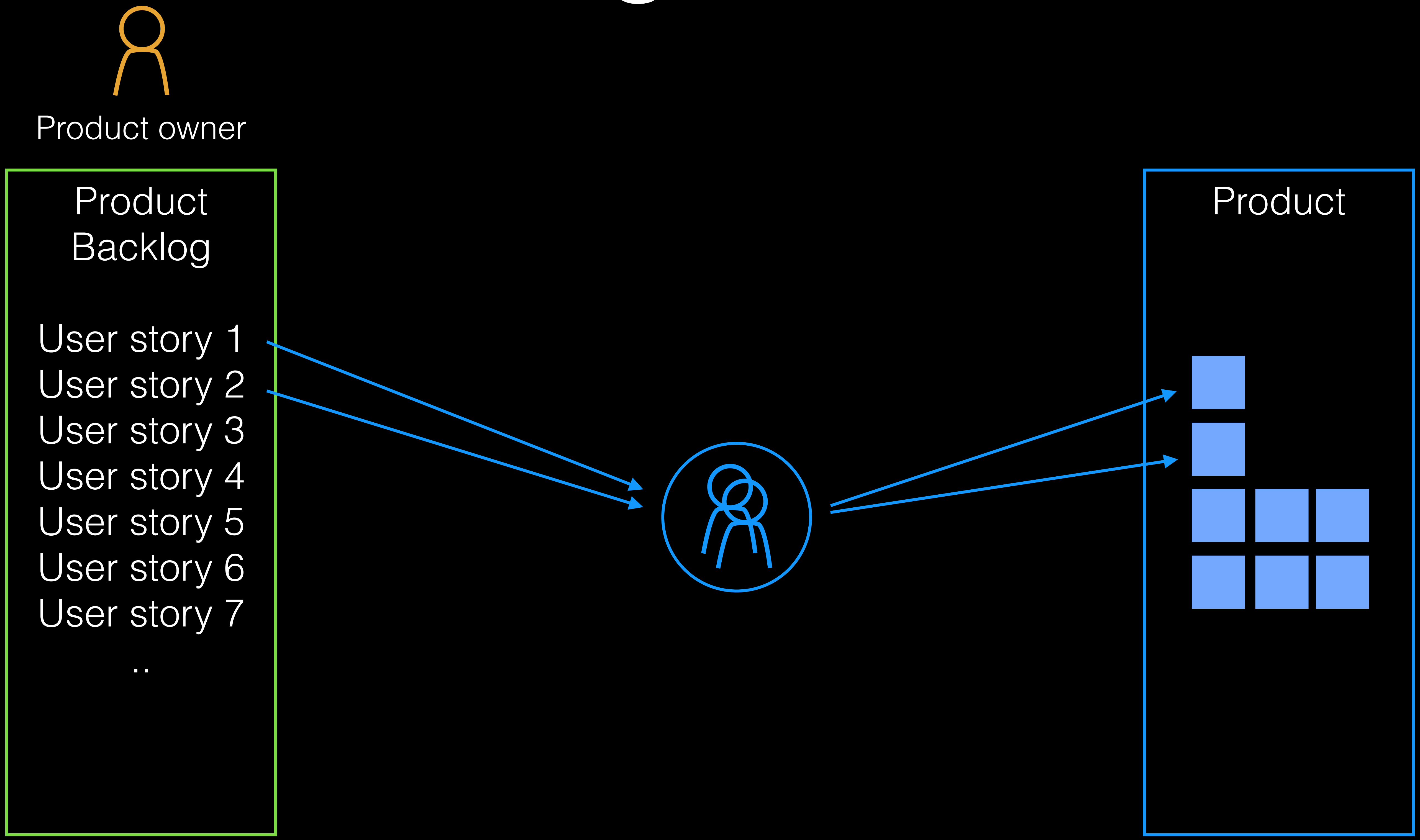
75%

Scaling Agile

1. Complex products

2. Large organizations

Single XFT



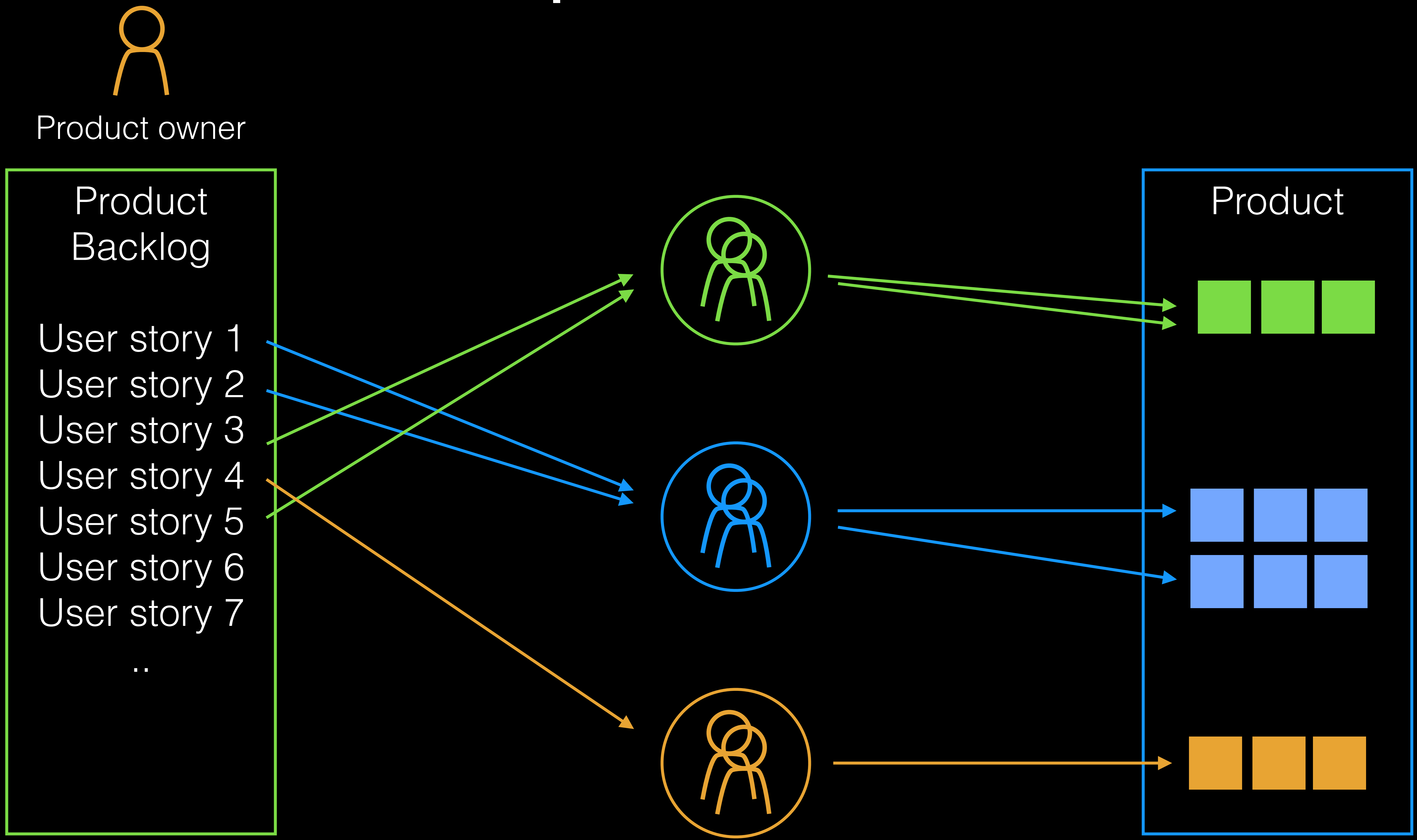
Product owner

Product
Backlog

User story 1
User story 2
User story 3
User story 4
User story 5
User story 6
User story 7
..

Product

Component XFT



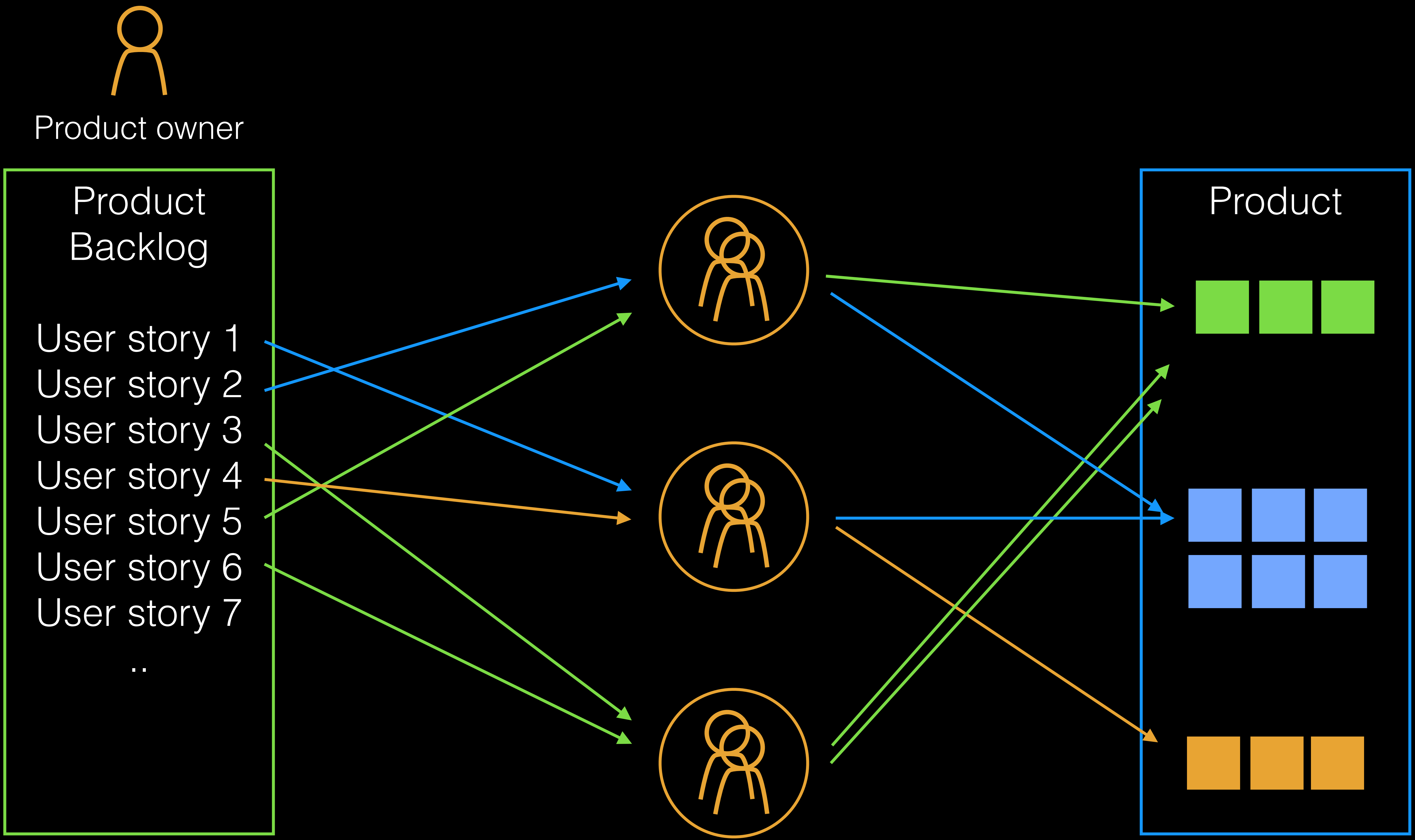
Product owner

Product Backlog

User story 1
User story 2
User story 3
User story 4
User story 5
User story 6
User story 7
..

Product

Feature XFT



Product owner

Product Backlog

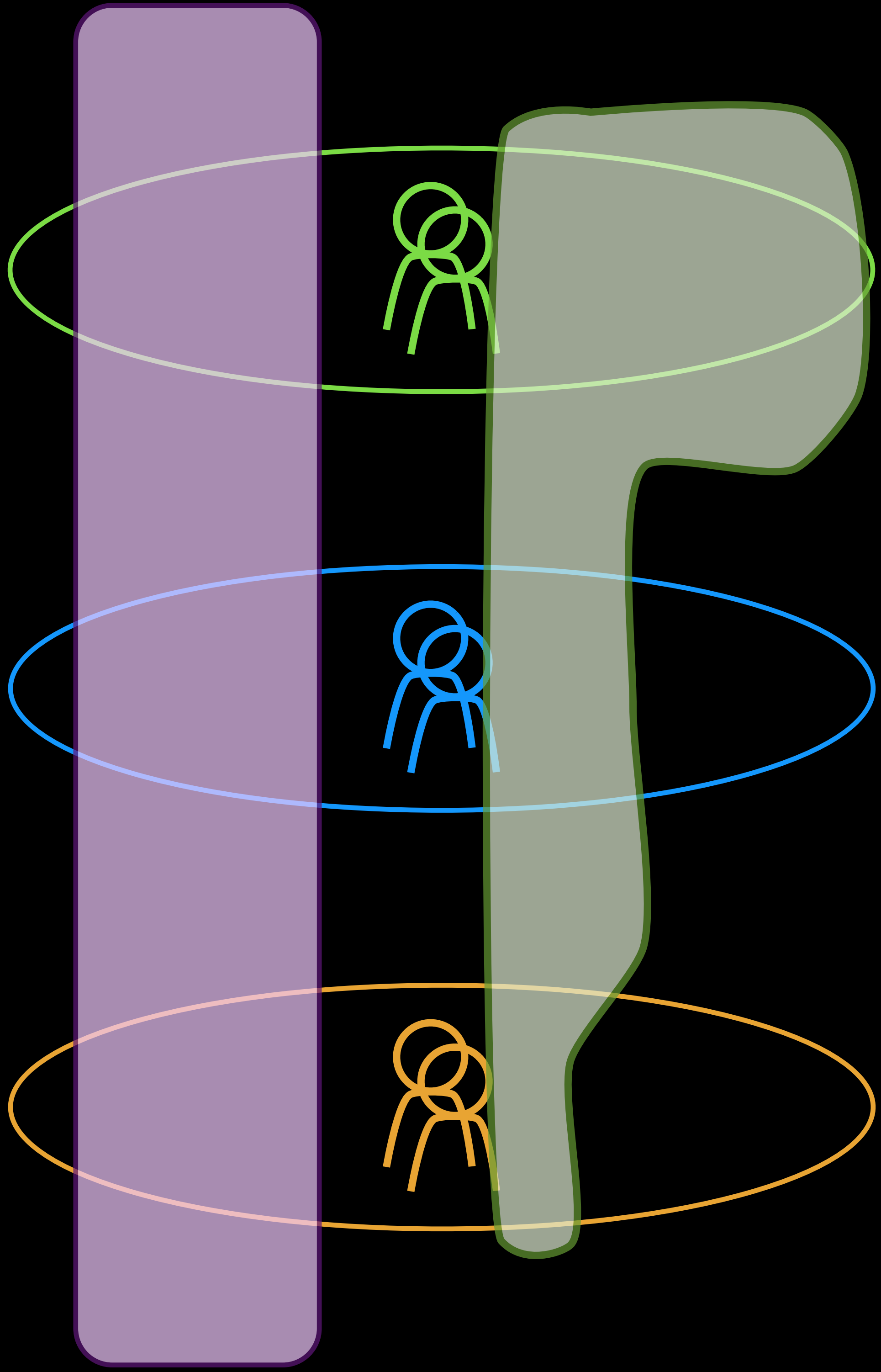
User story 1
User story 2
User story 3
User story 4
User story 5
User story 6
User story 7
..

Product

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?



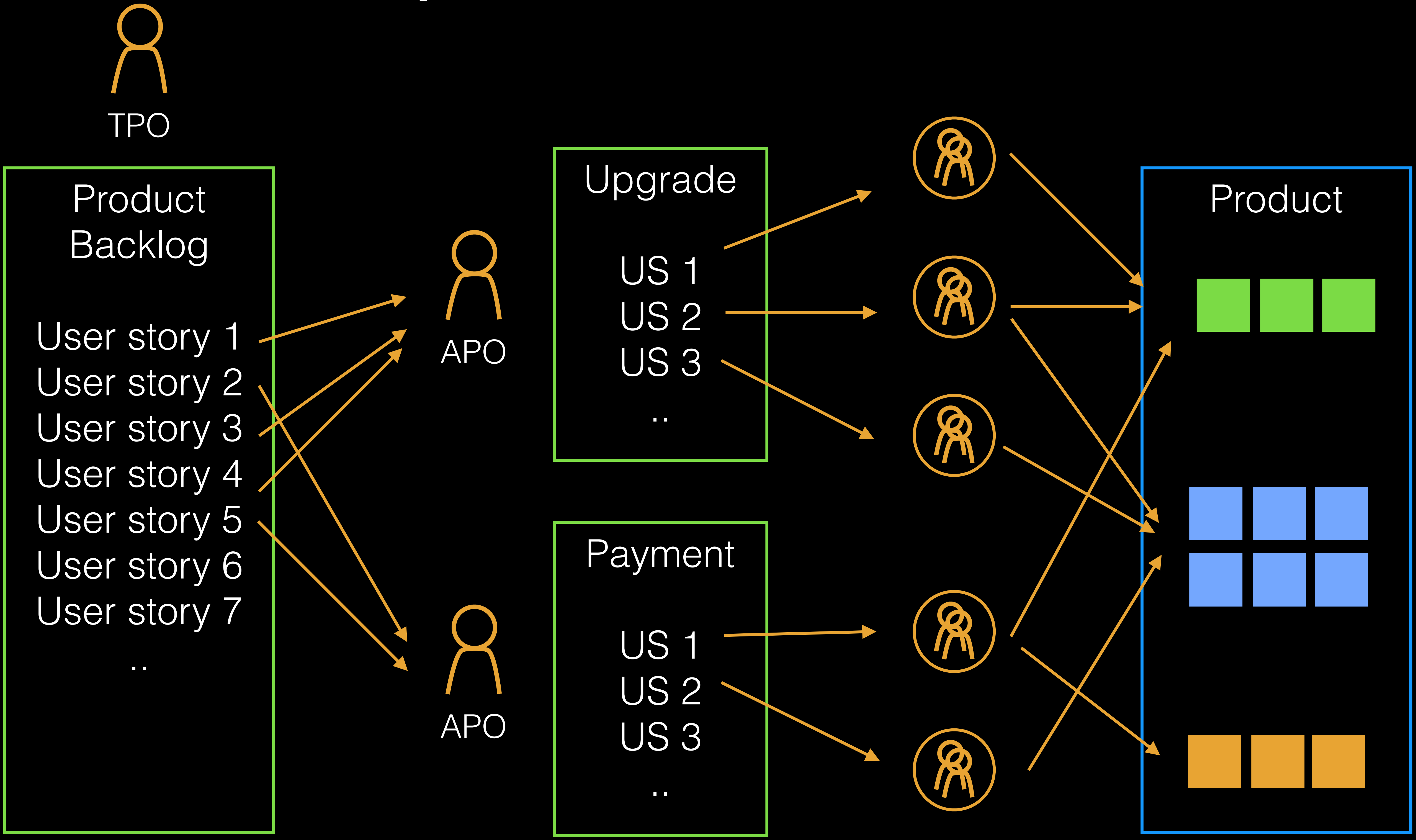
For even larger products

- 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


TPO

Product Backlog

User story 1
User story 2
User story 3
User story 4
User story 5
User story 6
User story 7
..

APO

APO

Upgrade

US 1
US 2
US 3
..

Payment

US 1
US 2
US 3
..

Product

Case study: Agile transformation

- 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

Case study: Spotify

- Kniberg & Ivarsson, 2012, “Scaling Agile @ Spotify with Tribes, Squads, Chapters & Guilds”

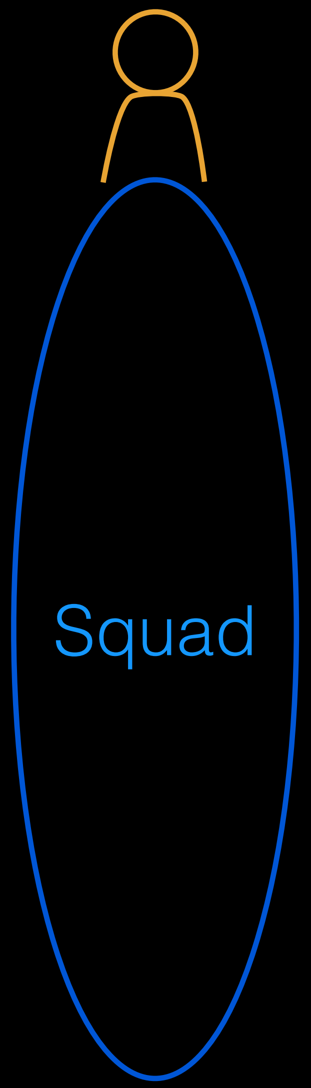
Spotify

- Since 2008
- 6 million paying subscribers
- 24 million active users
- 20 million songs
- Available in 28 countries

- 30 agile teams across 3 cities (2012)

Development

- Development is done in Squads
- A Squad is a self-organizing, cross-functional, team, that use Scrum, Kanban or other Agile methods
- A Squad works with one Product Owner (PO)
- Every Squad has a long-term mission and responsibility (e.g. build and improve mobile app experience, provide payment solutions)
- Encouraged to use MVP (minimum viable product)
- Squads become experts in their area!

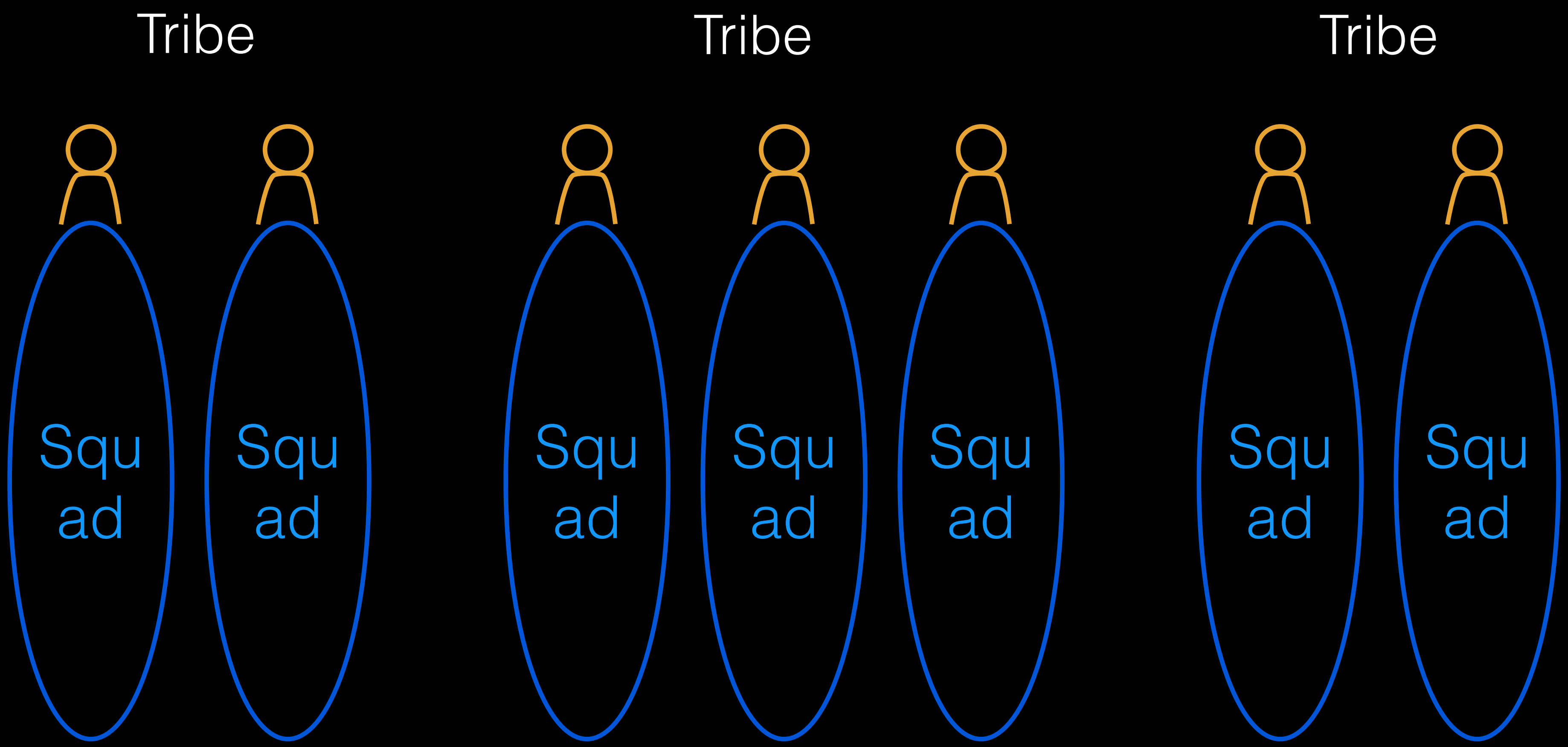


Squads, Roadmap and Coaching

- Each Squad is encouraged to spend 10% of development time on "hack days"
- There is no Squad leader, true self-organization
- Product owner only prioritizes the backlog, does not tell Squads what to do
- Products owners align their backlogs and present Spotify roadmap
- Agile coaches available to support Squads with Sprint Planning, Retrospectives
- Quarterly surveys with each Squad

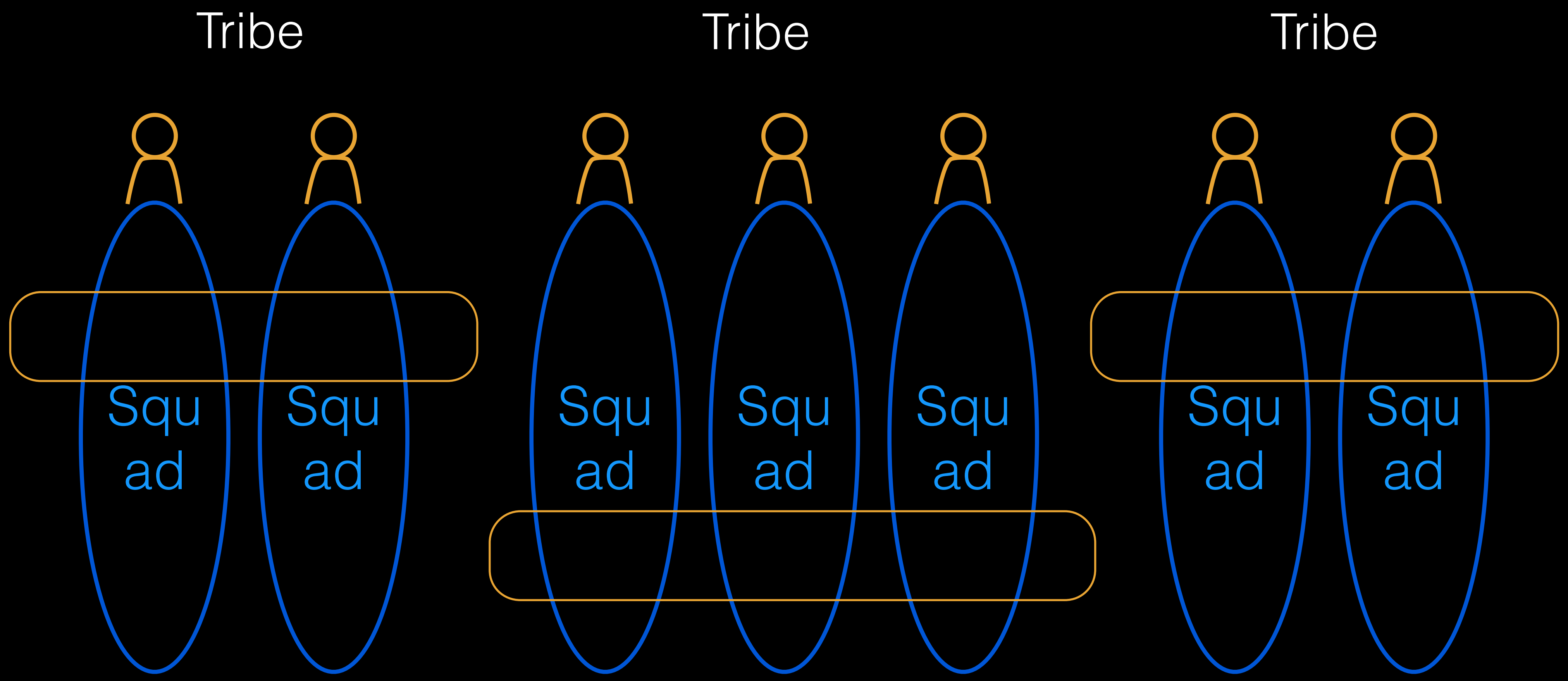
Tribes

- Collection of Squads that work in similar areas
- Sit together, has a Tribe leader, has Gatherings



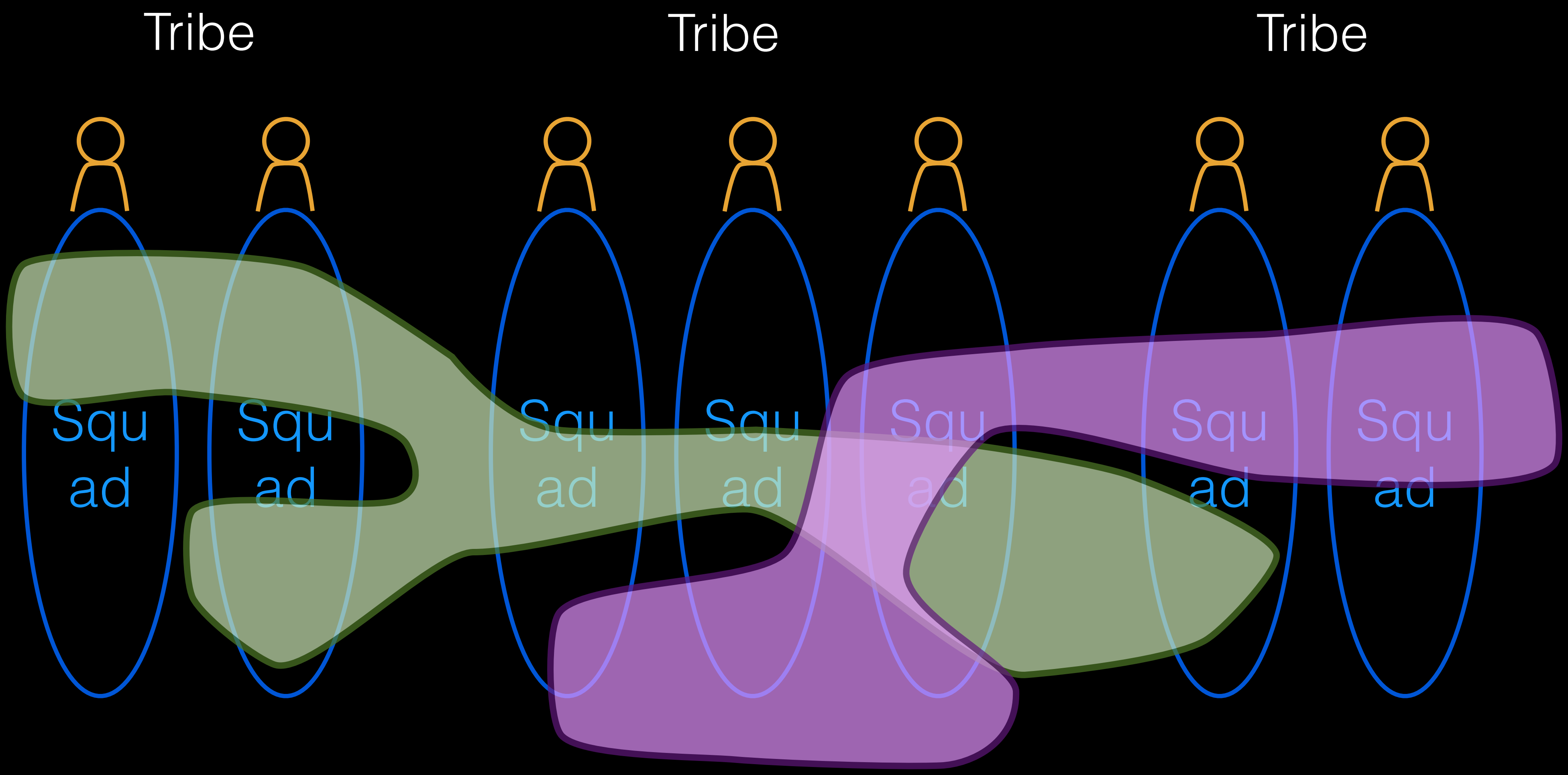
Chapters

- Chapter is your family within a Tribe of the people with the same competences as you
- Meets regularly to discuss the area and challenges
- Chapter leader is the line manager, responsible for people development, salary etc.



Guilds

- Similar to a community of practice, that spans the whole company
- E.g. web technology guild, tester guild, agile coaches guild



Architecture

- 100 distinct systems that can be deployed separately
- Squads are feature teams and can edit any part of the system
- System Owners
  - One person or one pair that keeps track of quality, performance, technical debt, release process, scalability for a system
  - A system maps approximately to the tribe areas
- Chief Architect
  - Handles high level architecture issues across multiple systems

Spotify

- Case study was written in 2012
- Spotify had grown from 30 to 250 people
- So far the model had scaled well!