



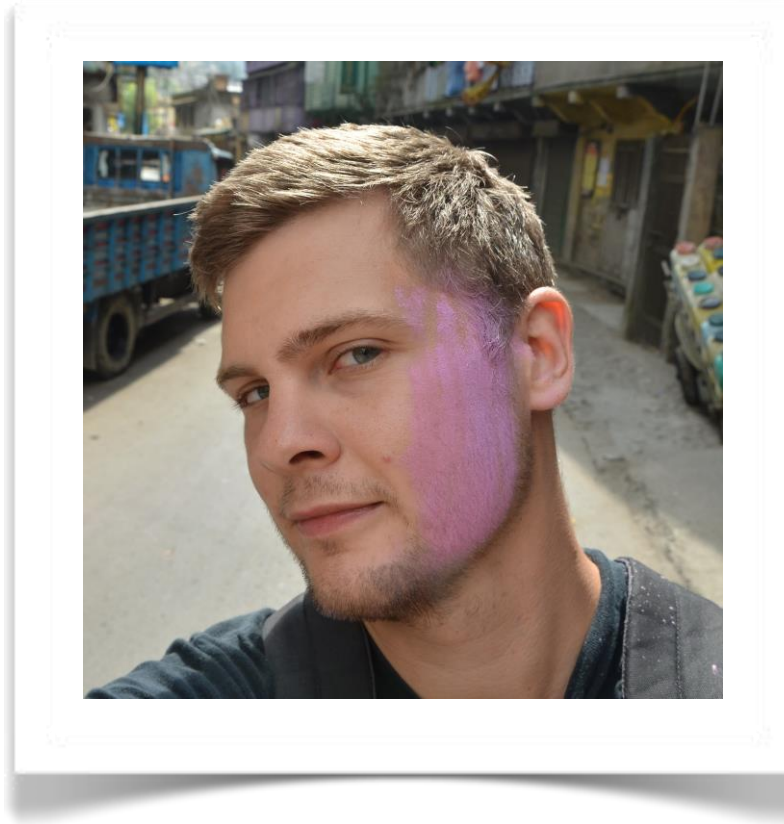
**Spotify®**

# Leadership @ Spotify

Agile 2014 Orlando



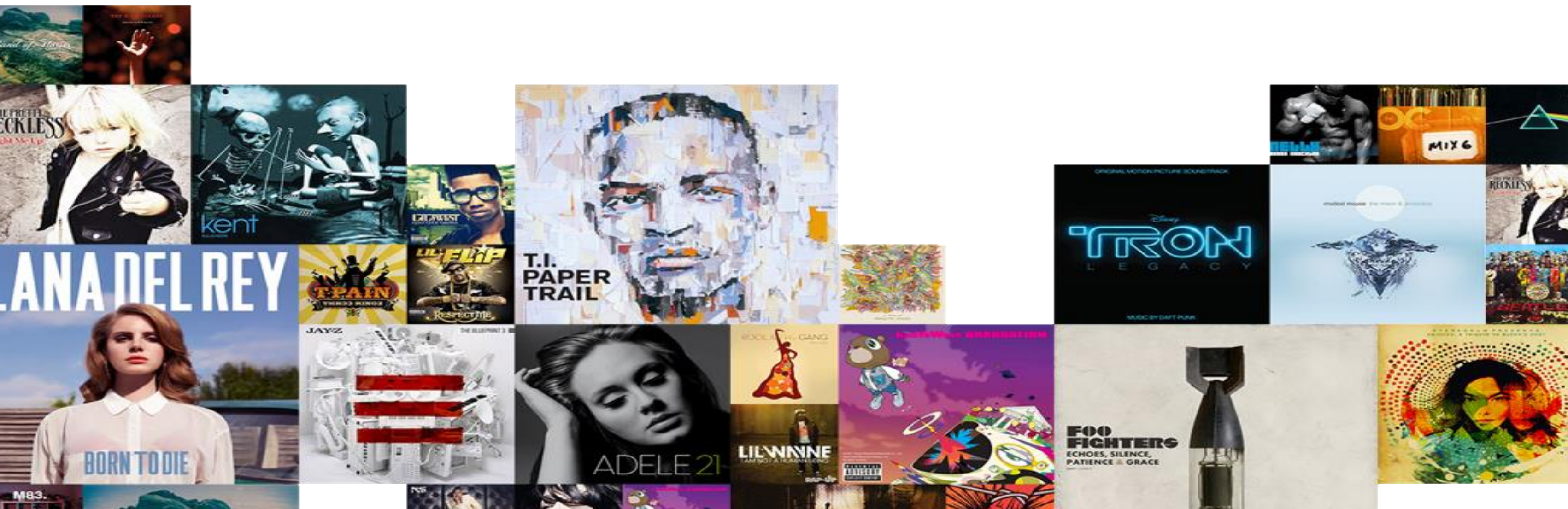
Kristian Lindwall  
@klindwall



Anders Ivarsson  
@anders\_ivarsson



# The right music for every moment





**40M** active users.

**10M** paying subscribers.

**1.5B** playlists.





**56 countries.**





San Fransisco

Boston  
New York

Gothenburg Stockholm

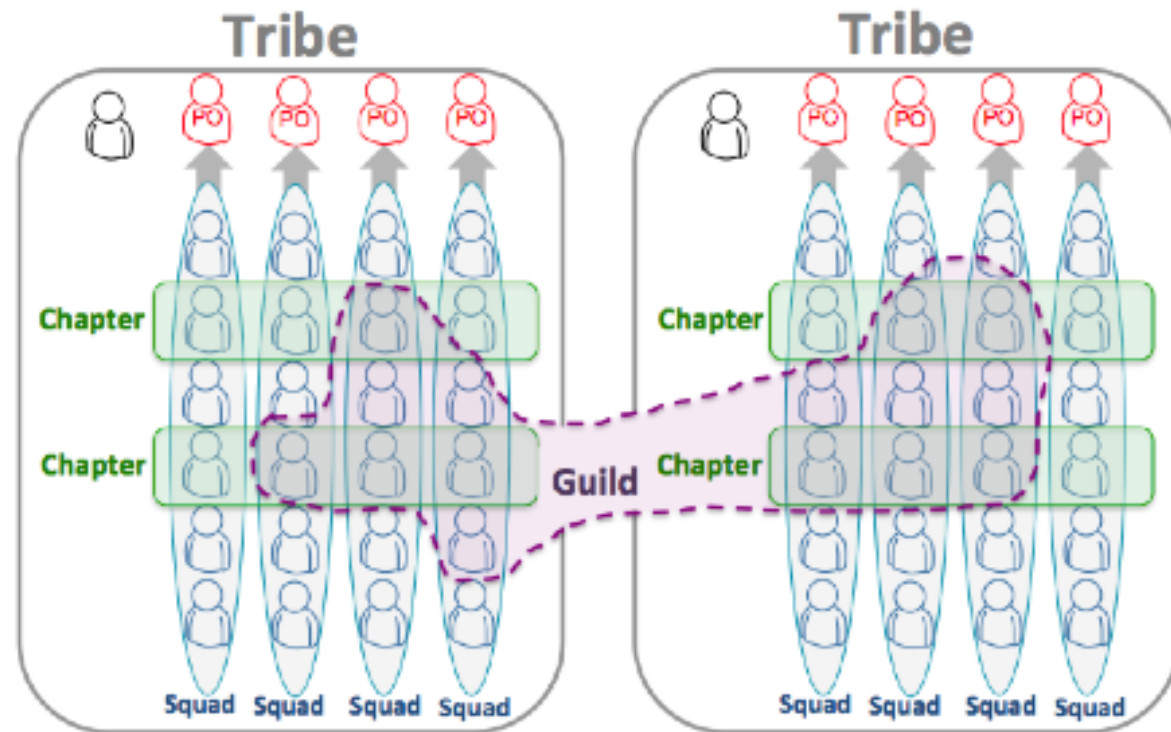
**5 dev sites.**  
**3 timezones.**  
**500+ devs.**



# Scaling Agile @ Spotify

## with Tribes, Squads, Chapters & Guilds

Henrik Kniberg & Anders Ivarsson  
Oct 2012



Dealing with multiple teams in a product development organization is always a challenge!

One of the most impressive examples we've seen so far is Spotify, which has kept an agile mindset despite having scaled to over 30 teams across 3 cities.

Spotify is a fascinating company that is transforming the music industry. The company has only existed 6 years and already has over 15 million active users and over 4 million paying. The product itself can be likened to "a magical music player in which you can instantly find and play every song in the world".

Alistair Cockburn (one of the founding fathers of agile software development) visited Spotify and said "Nice - I've been looking for someone to implement this matrix format since 1992 :) so it is really welcome to see."

# Scaling Agile @ Spotify

- Released oct 2012
- Presented at Agile 2013
- Many questions around leadership
- How do you make decisions?



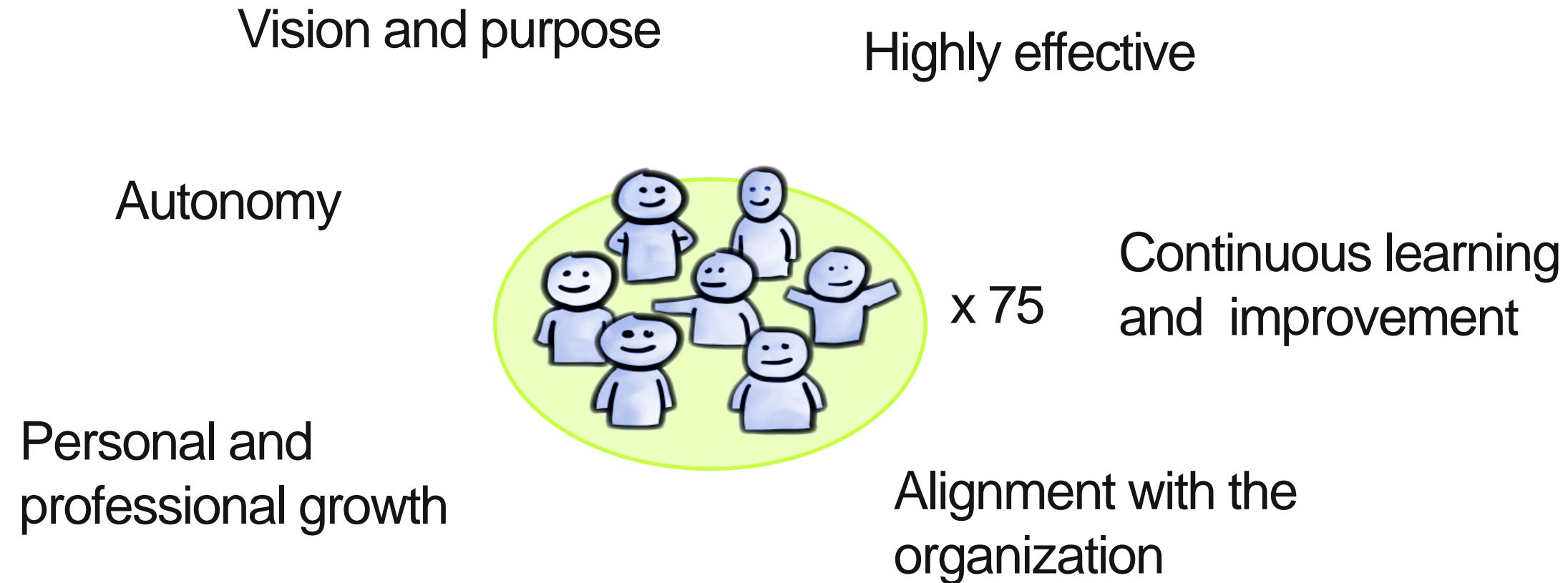


The best organizational structure.  
Ever.

No Problems Model <sup>TM</sup>

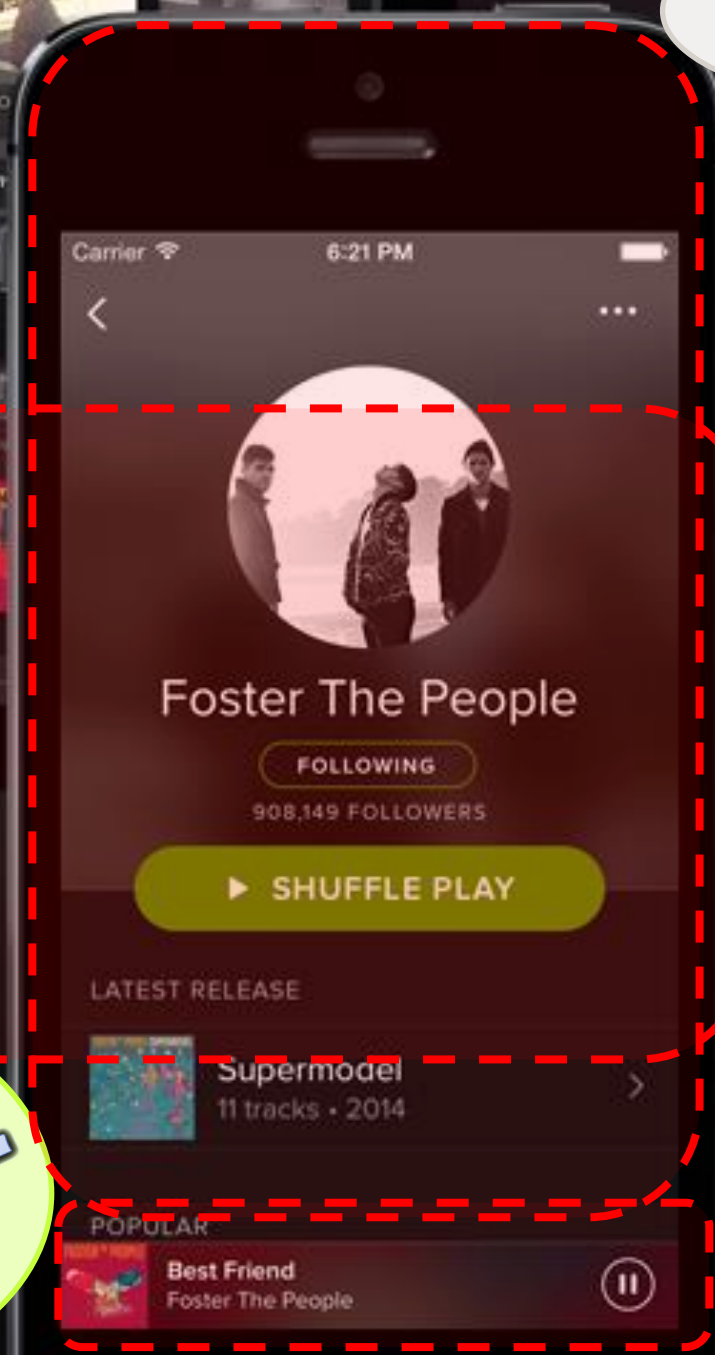
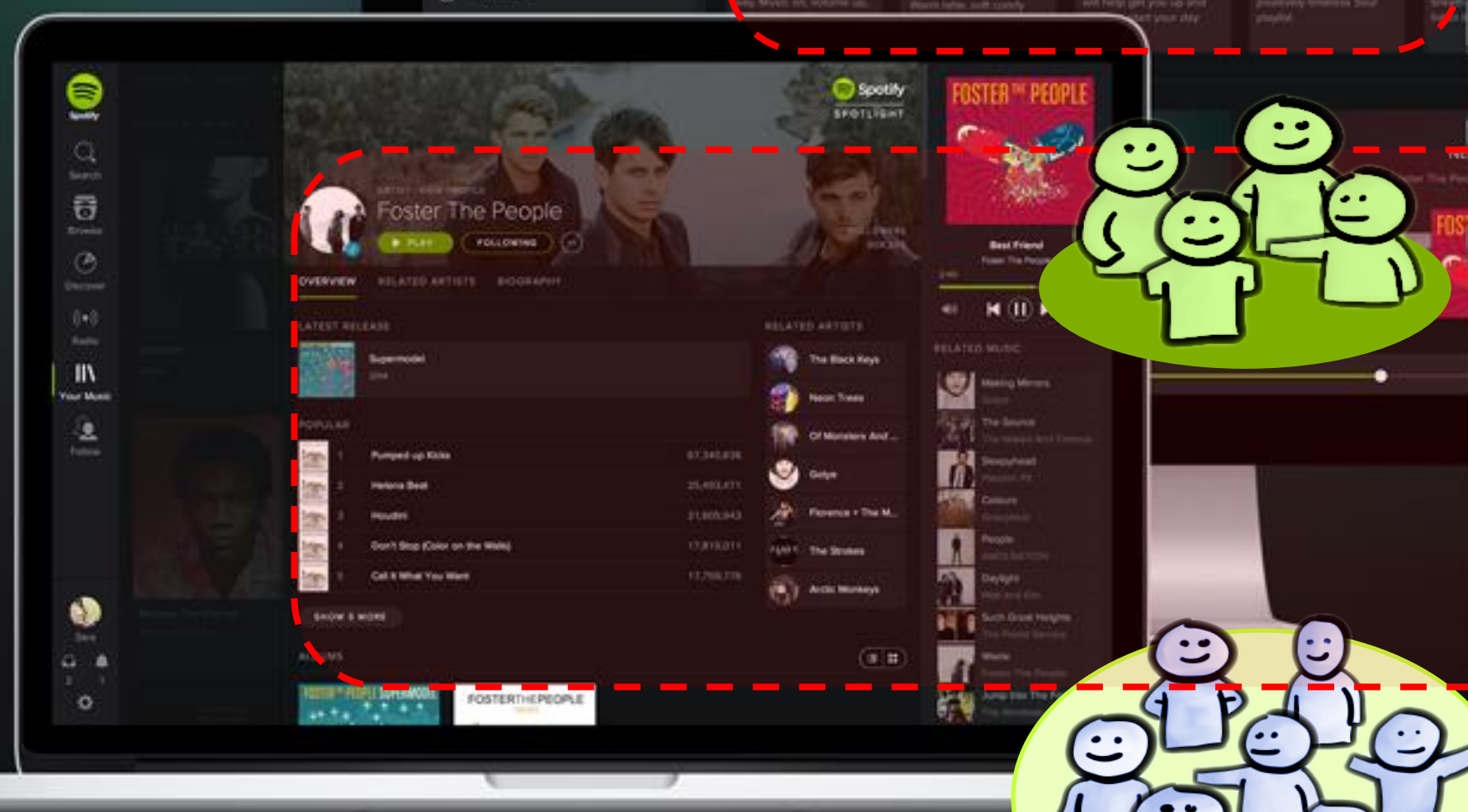
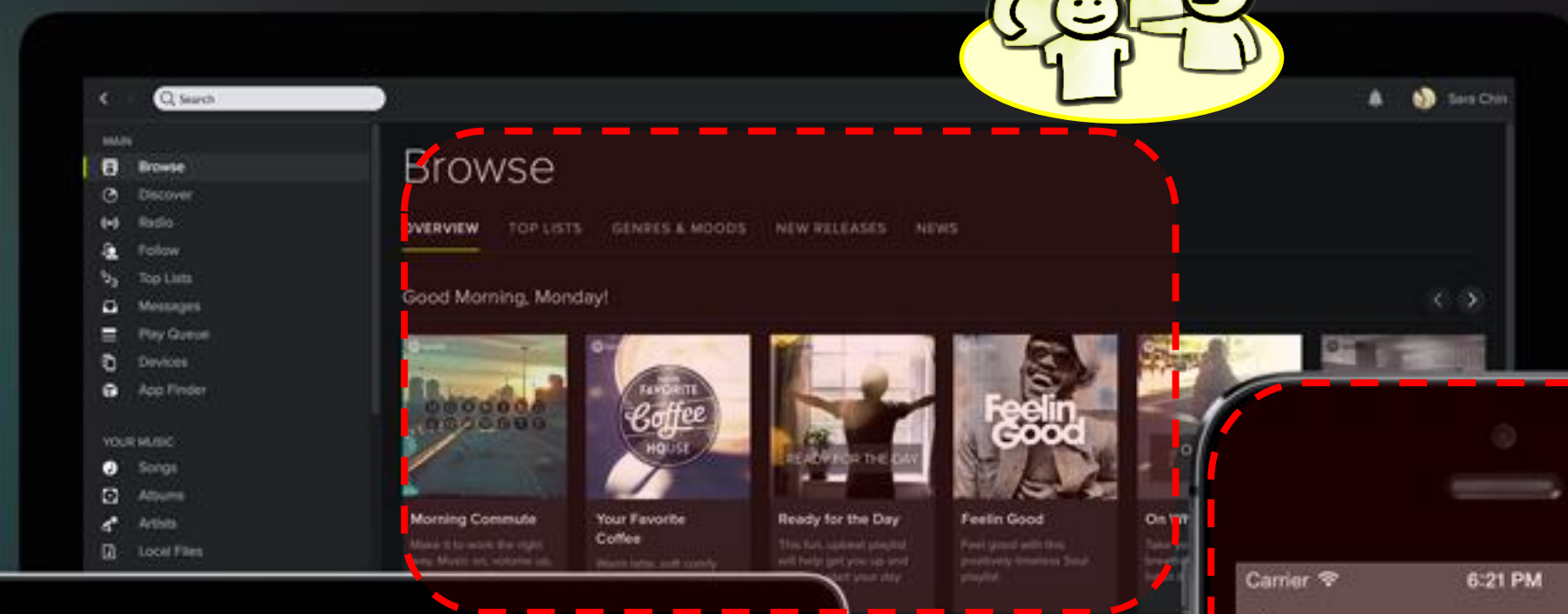
\* Everyone working at the right thing, at the  
right moment in the right way. No structure.  
No communication. No problems.





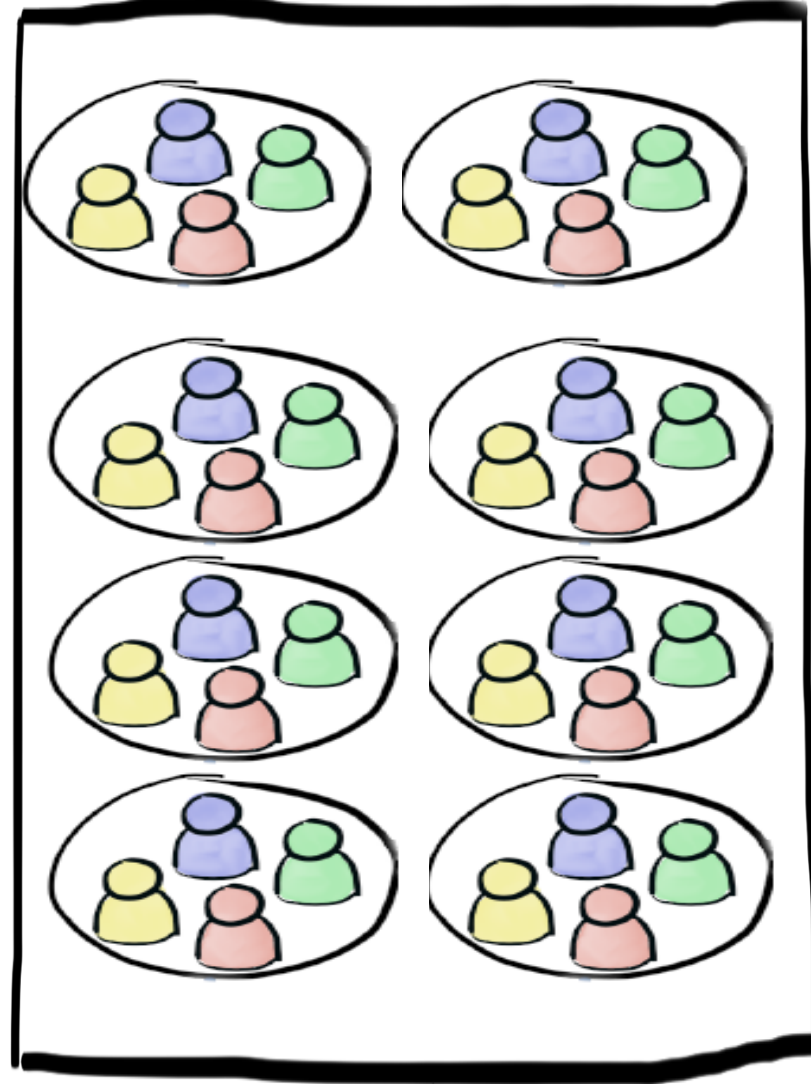
Any act in support of these is  
an act of leadership



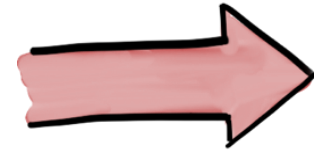




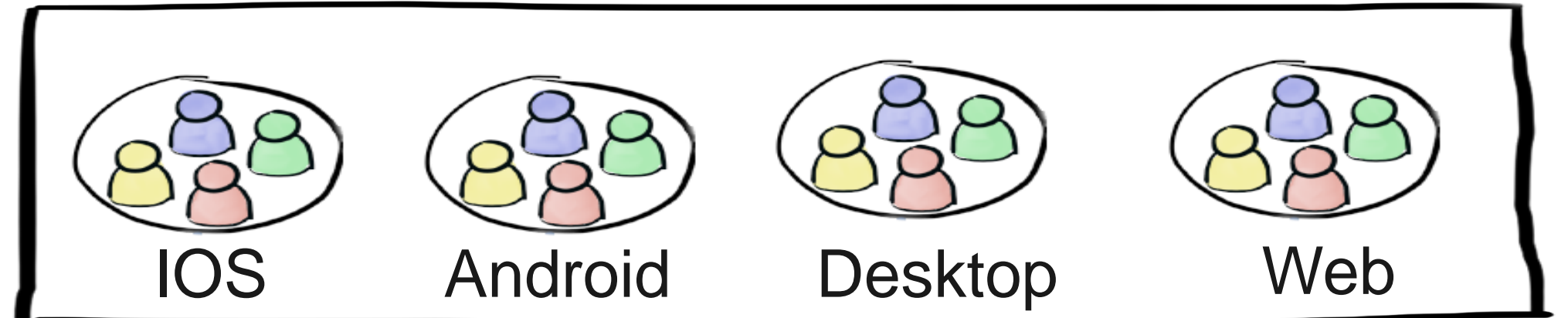
Infrastructure squads



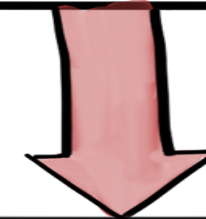
Enable & support



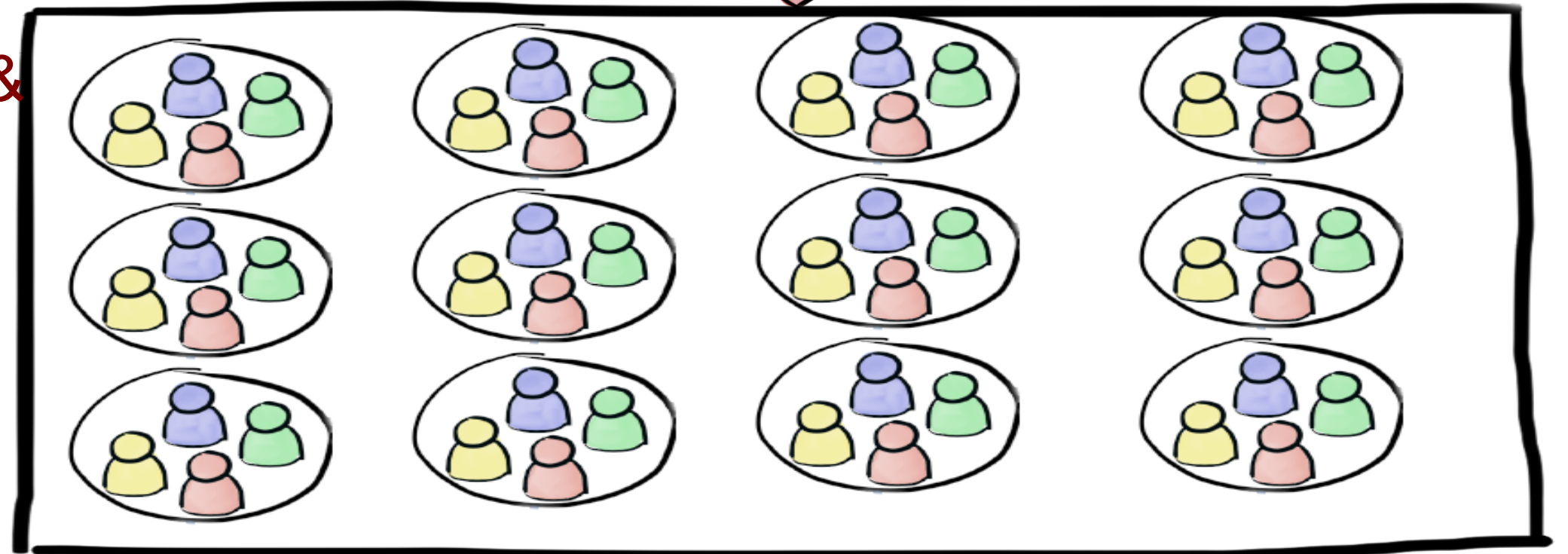
Container squads



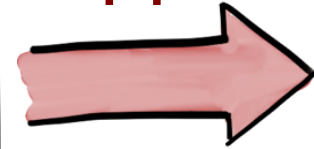
Enable & support



Feature squads

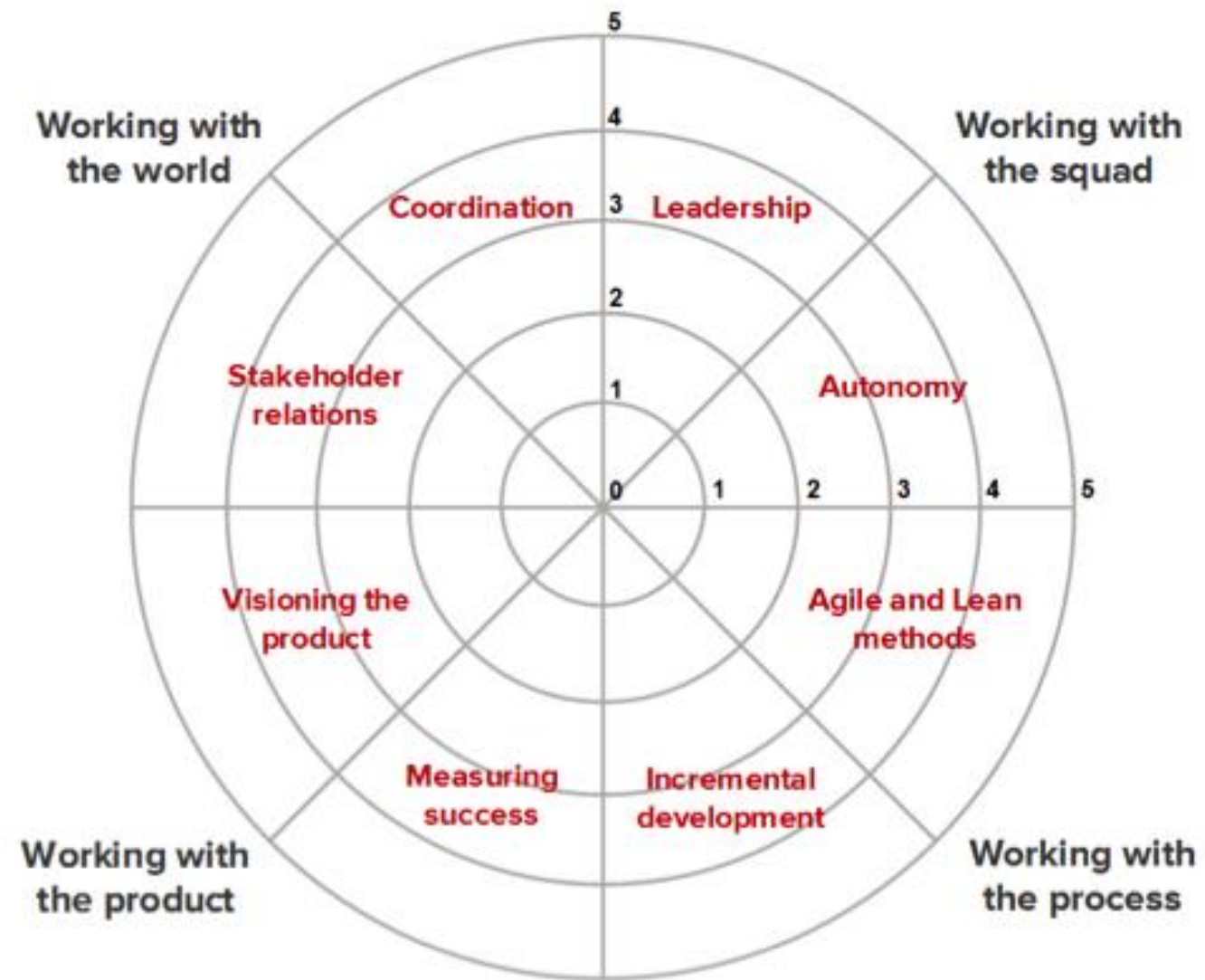
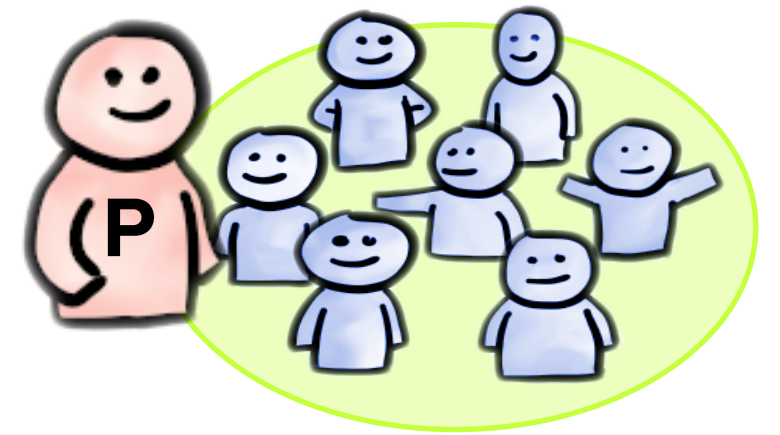


Enable & support



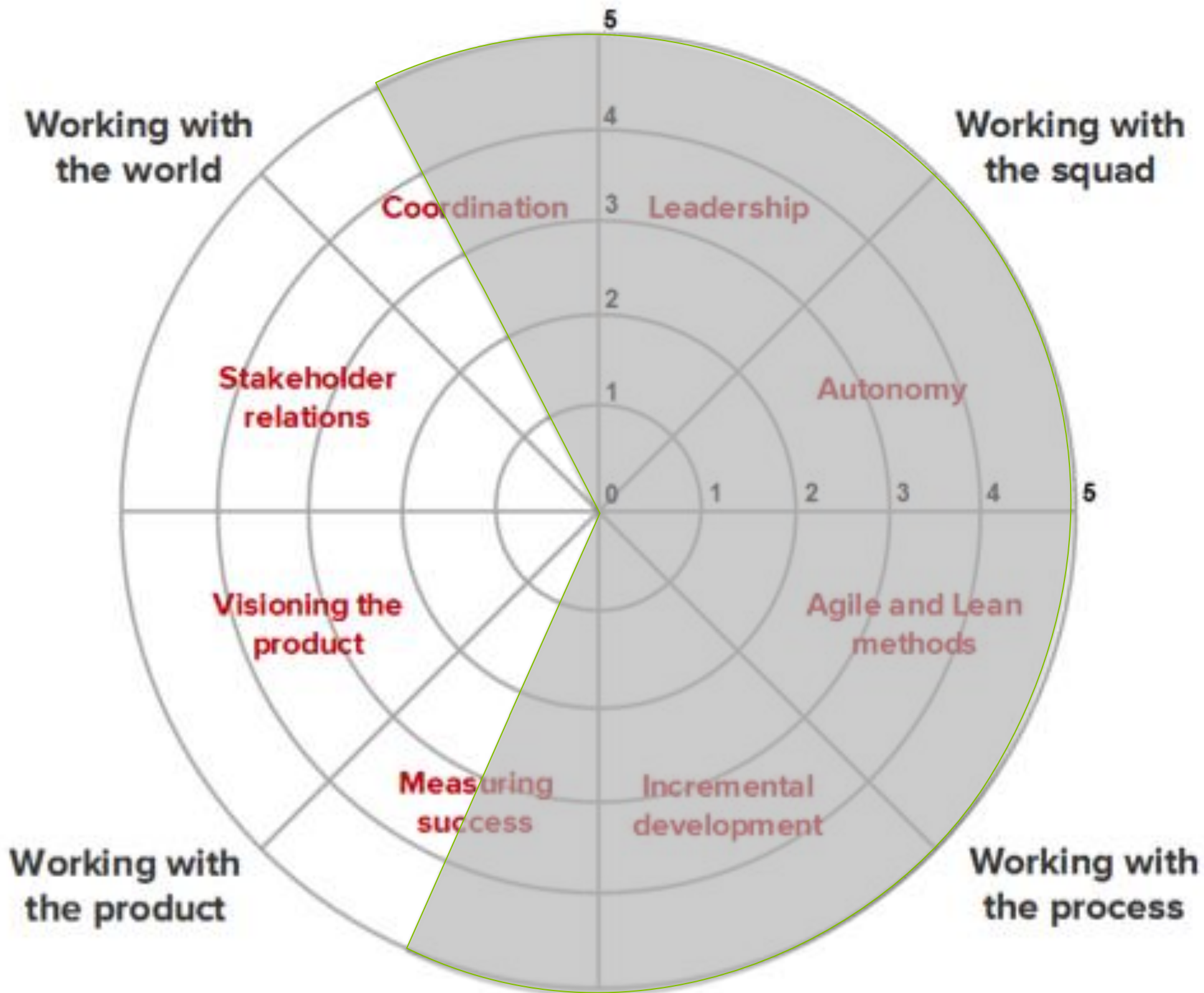
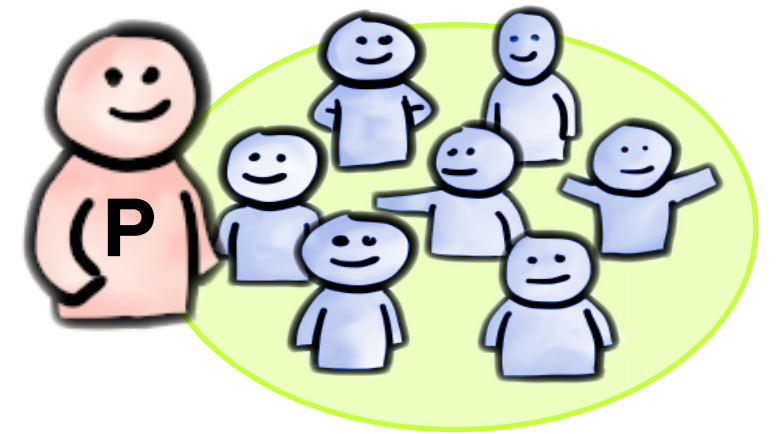


# Product Owner



- What do we do – and why?
- Product vision
- Squad mission
- Purpose
- Alignment
- Support the squad





## Product Owner framework

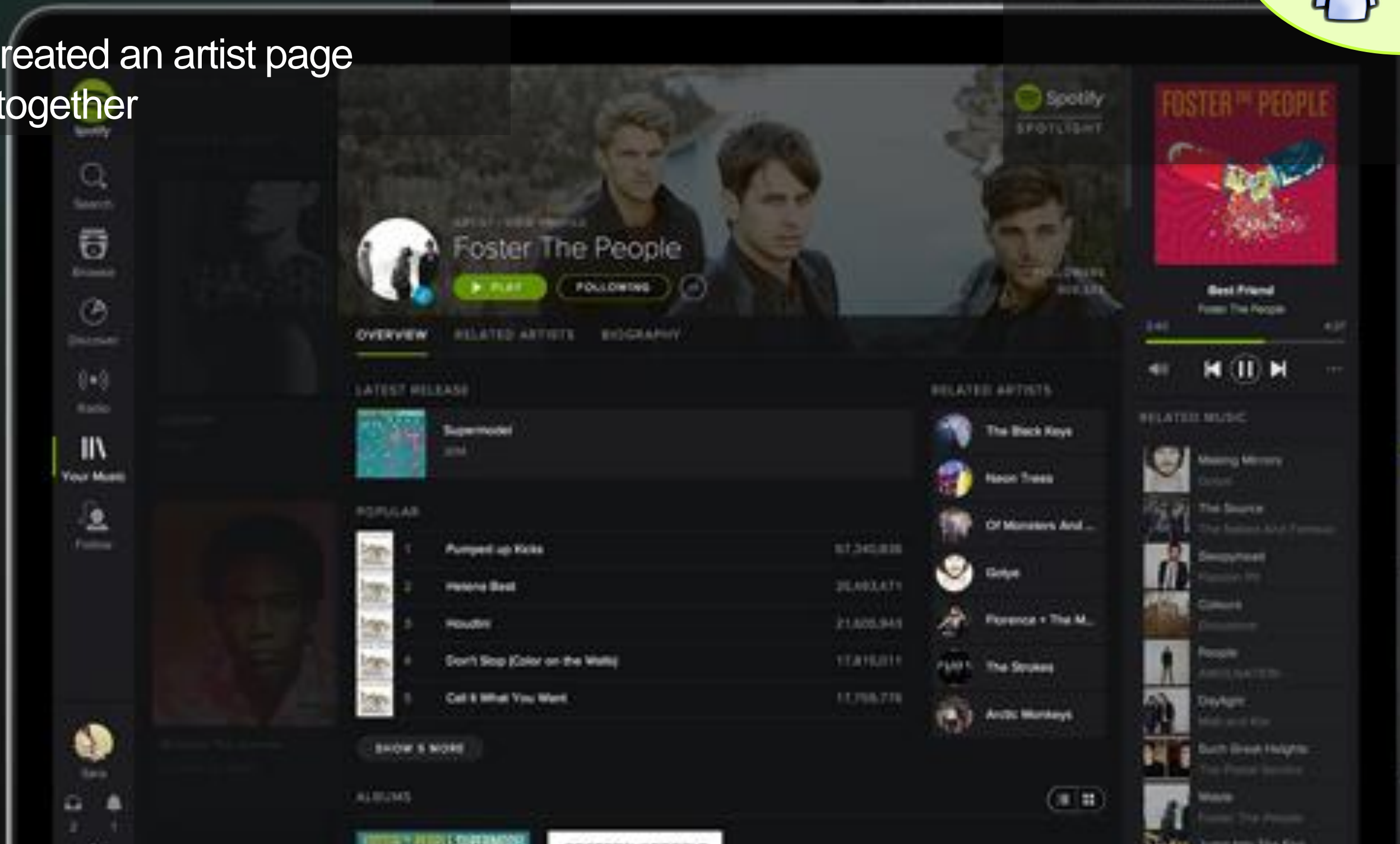
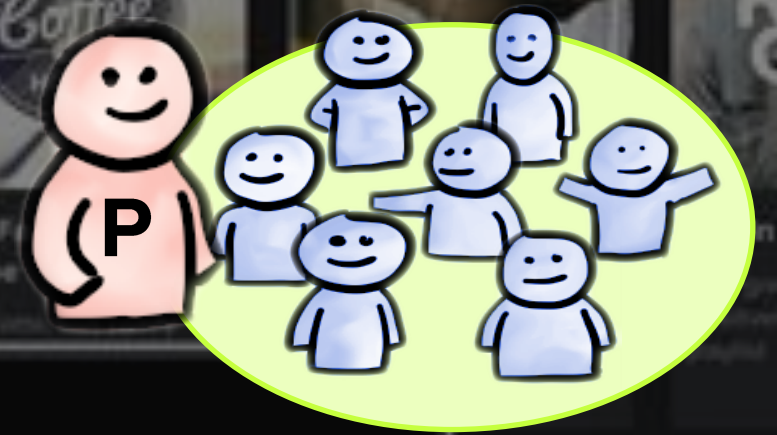
- Working with the world
- Working with the product
- Working with the squad
- Working with the process



# Product Owner

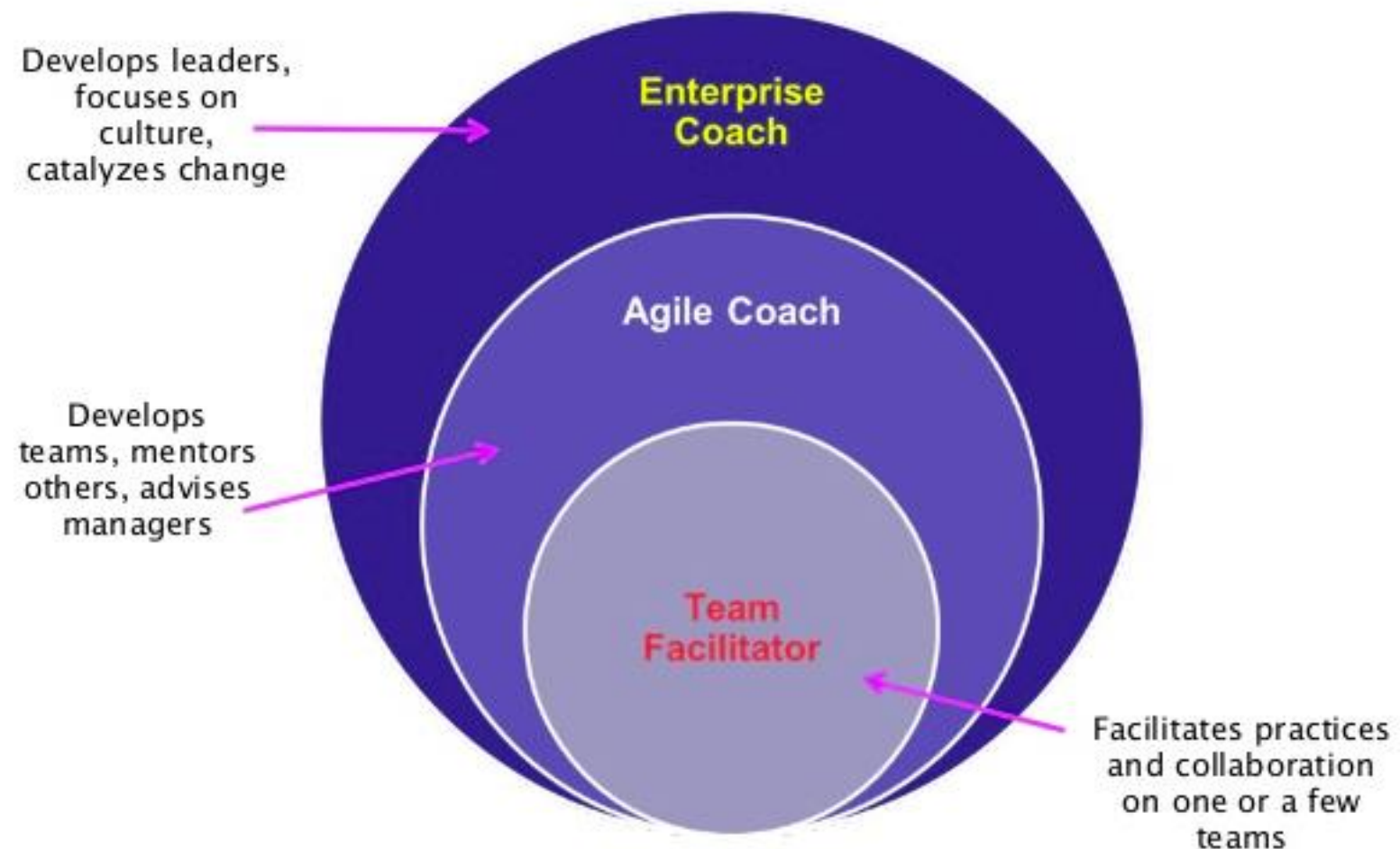
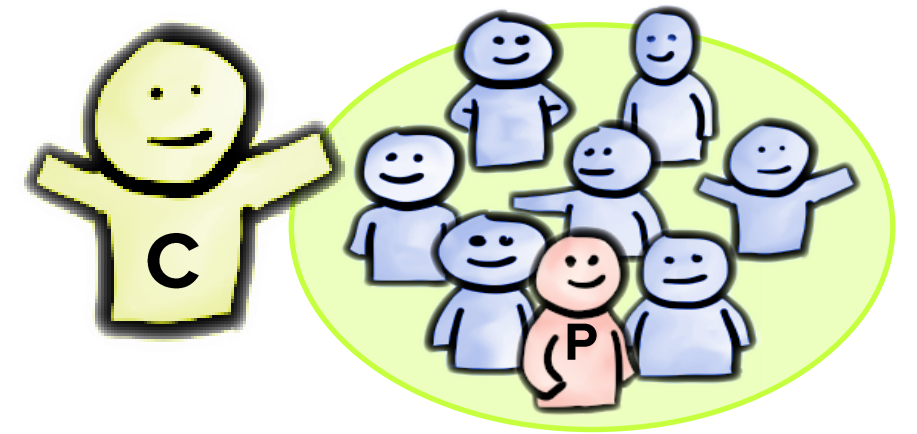
## Example

How we created an artist page  
roadmap together





# Agile Coach



- Grow high performing teams
  - Continuous improvement
  - Ways of working
  - Collaboration
  - Agile and lean practices and values
- 
- Grow a high performing organization

*Levels of agile coaching, The agile coaching institute*



Sprints

User stories

Scrum

XP

Planning Poker

# We definitely do the basics...

Story Points

Demos

Kanban

Stand ups

Retrospectives



# 1 on 1 coaching

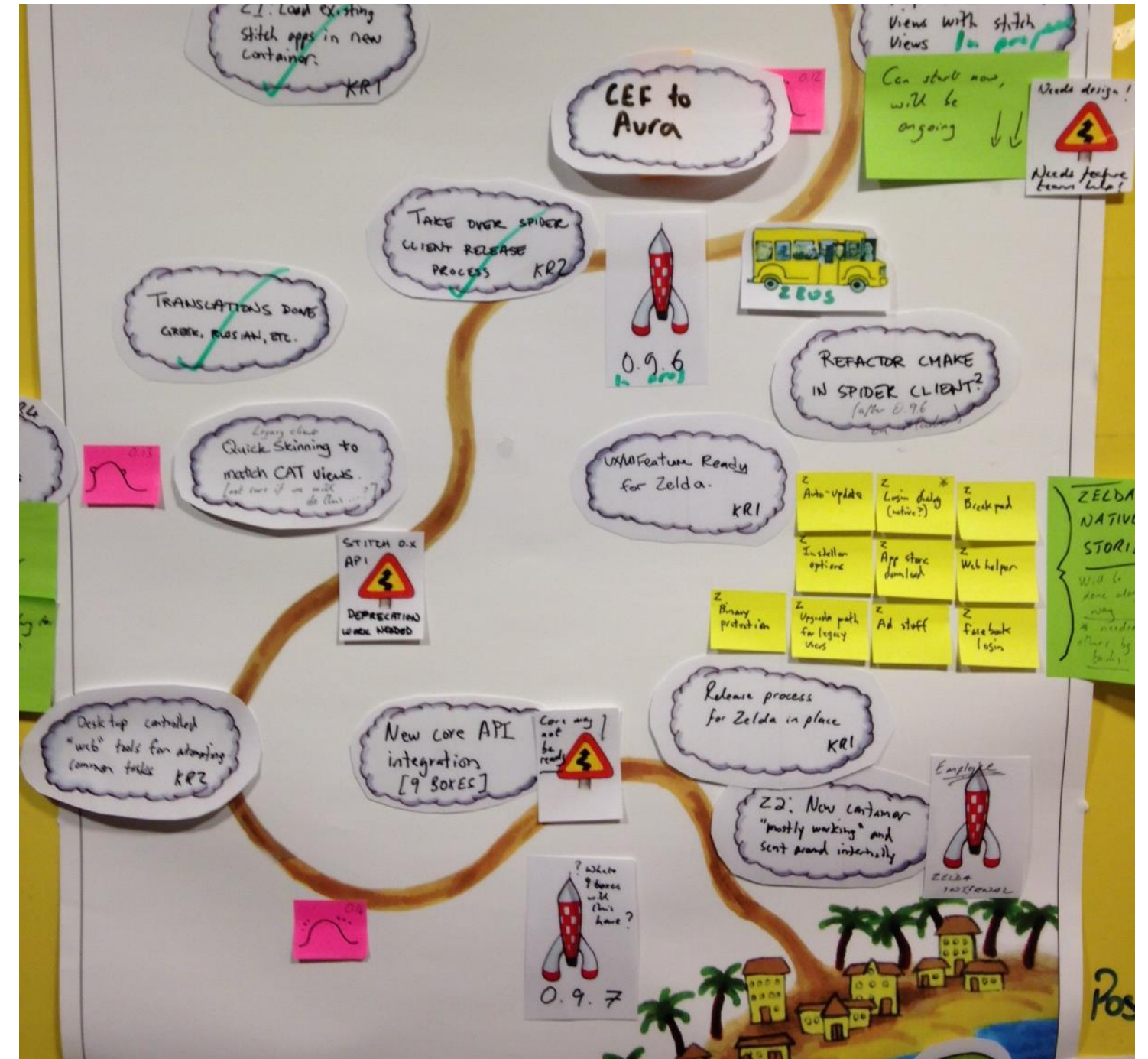
- Team members
- Product owners
- Chapter leads
- Senior management





# Squad roadmapping

- Group of coaches came up with new visualization
- Ran workshops with all squads in a tribe





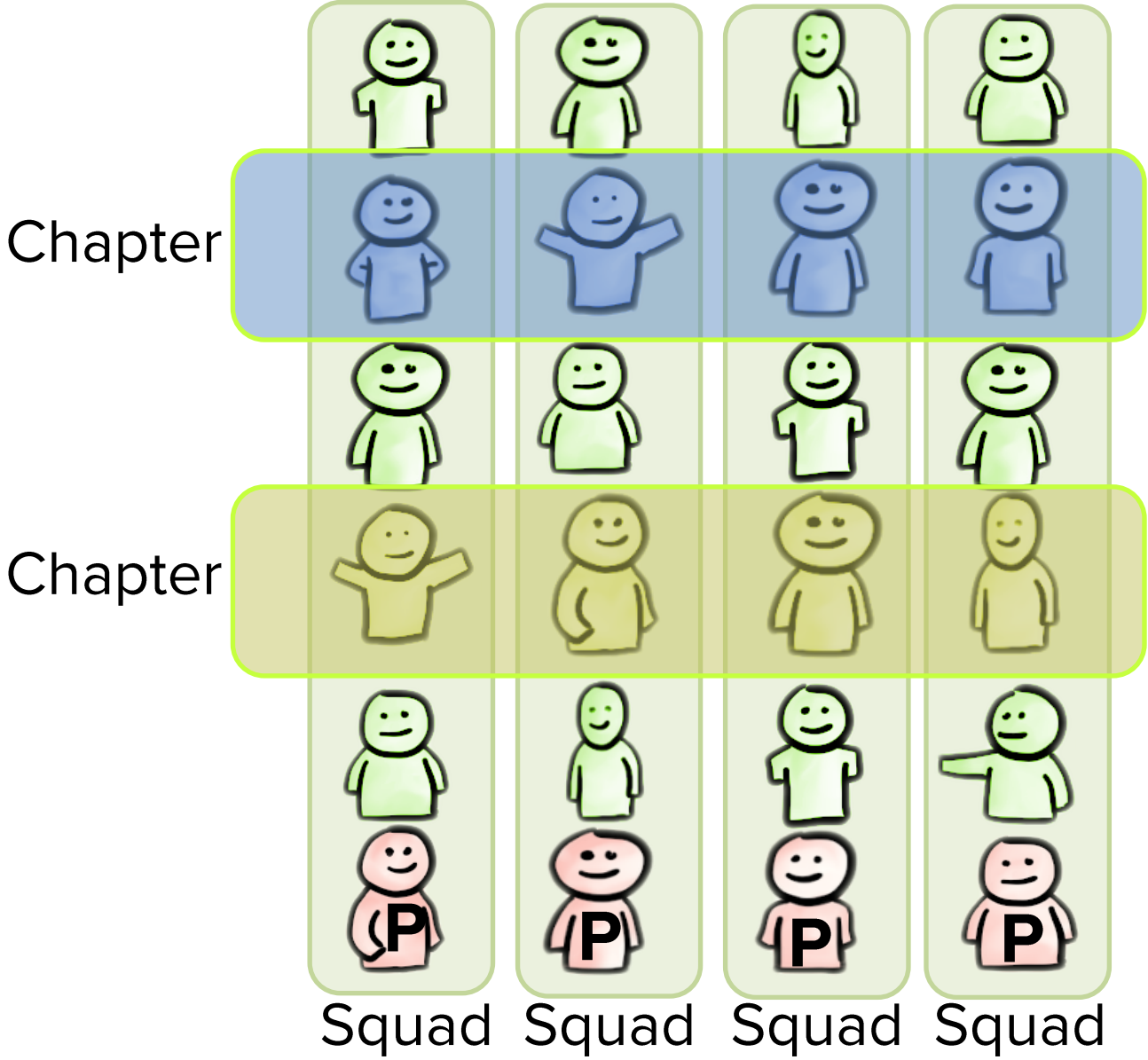
# Continuous delivery

- Maturity workshops
- Big cultural shift
- Cross tribe initiative

9 steps to heaven			
	Hell	Purgatory	Heaven
<b>Source / configuration management</b>	Team uses source control for code only. Highly manual to recreate any build or environment. May involve occult knowledge...	Most things are in source repository - but some things missing (configuration information, build steps), some manual steps required to fully install / build	Totally risk free and easy to rebuild any environment / executable on fresh machine from one repo. Fully automated
<b>Branching</b>	Anything goes - frequent merge hell	Some long lived feature branches, but we try to avoid this	We work off a stable Master using feature flags / Branch by Abstraction to avoid long lived feature branches
<b>Commit</b>	Commit size varies from small to enormous. Commits regularly abandoned	Mainly sensible size, but unpredictable. Discipline varies.	Small discrete commits, protected by feature flags when needed.
<b>Automated testing</b>	The team occasionally writes Unit Tests. Their status is not always known. Test automation seen as the responsibility of a Test Automater. Failing automated tests are ignored.	Unit tests run by post-commit hook. Test automation is valued, though not everyone contributes. Unit tests are expected, and coverage known. It takes a long time to run regression tests. Results of automated tests are unpredictable.	Test automation includes Unit Testing, Integration Testing, Acceptance Testing, Performance Testing. The level of automated tests gives us confidence to release. Very limited human checking required.
	We have a CI server, but it's slow and unreliable to build. Broken builds are common	The CI server builds nightly. Most builds succeed, but we don't always fix a broken build	Every commit to master is built on an Integration machine. Fixing a broken build stops all other work and the build is fixed as

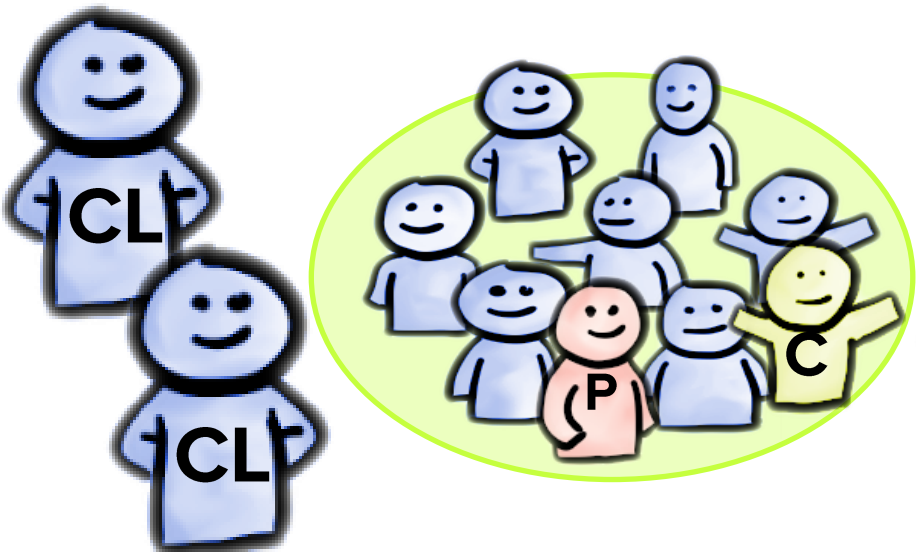
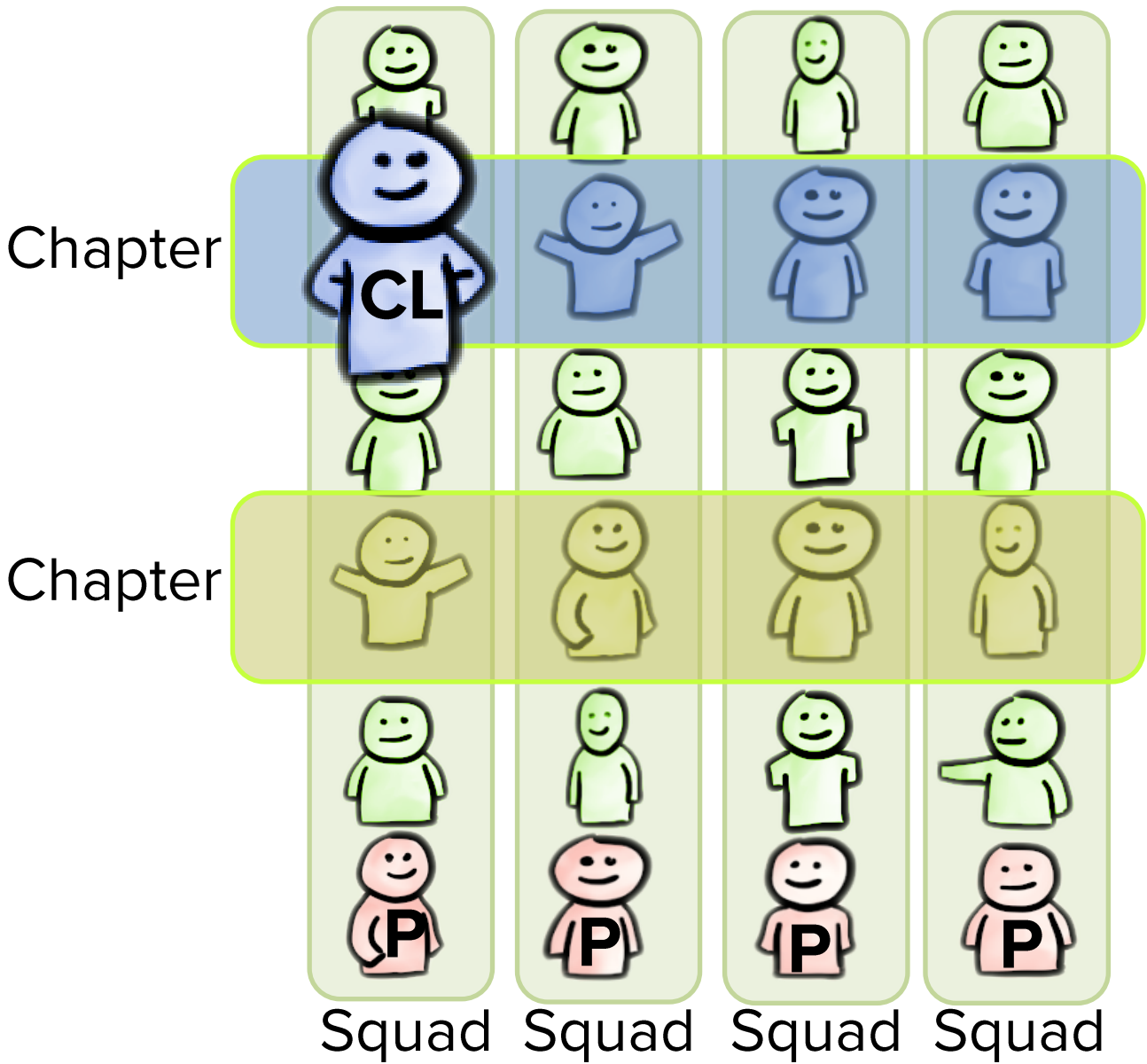


# Chapters





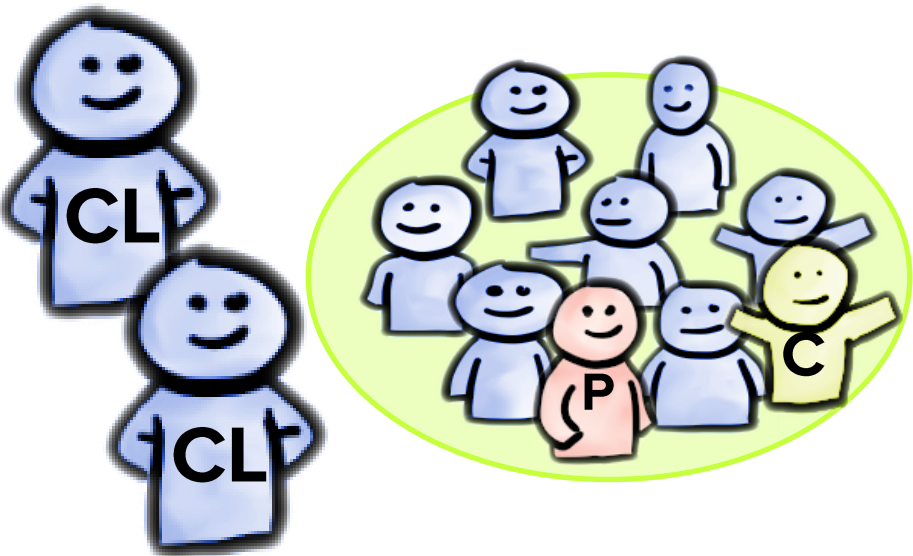
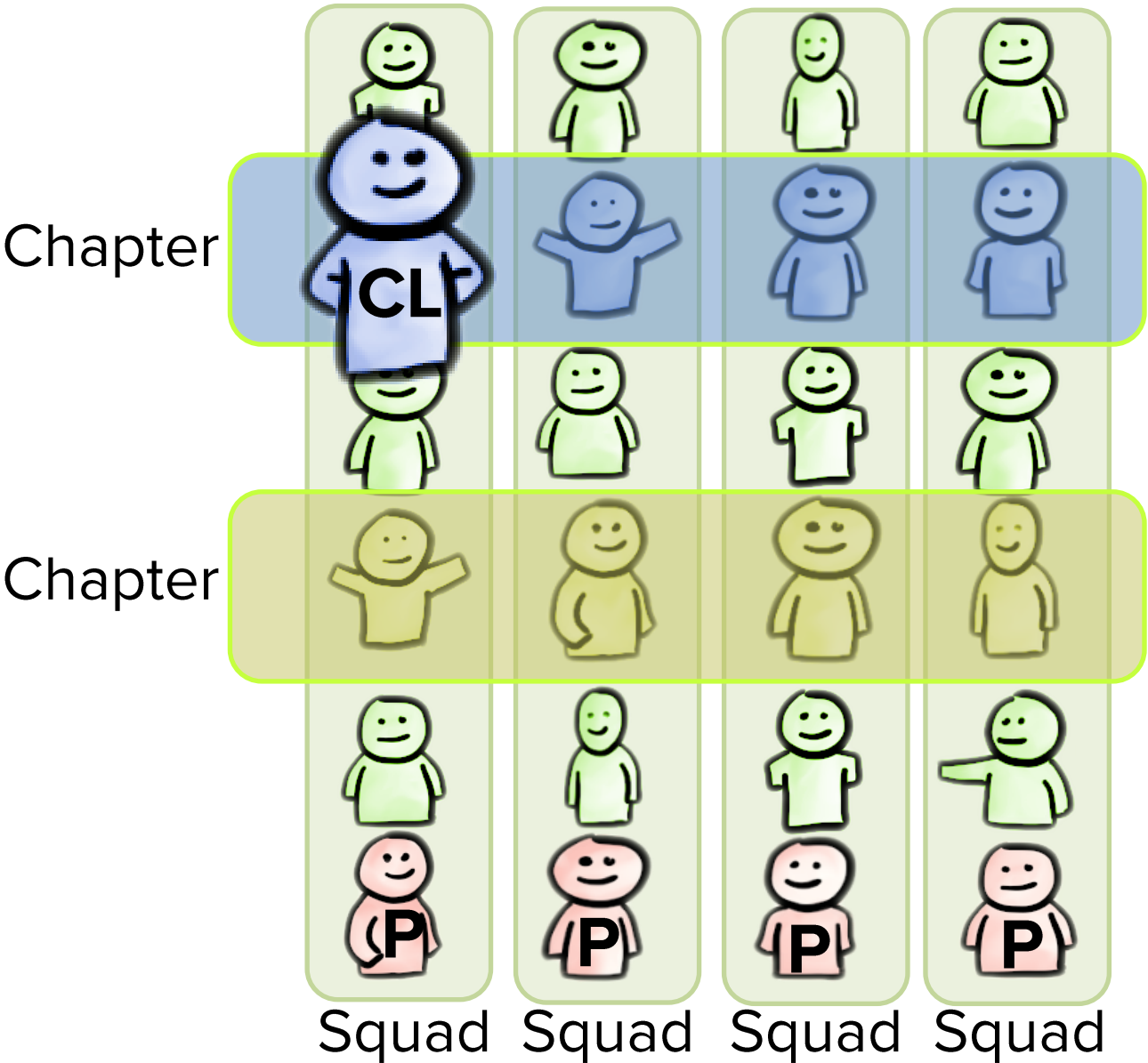
# Chapter Lead



- Grow individuals
- Grow engineering culture
- Support the squads
- Hire the right people



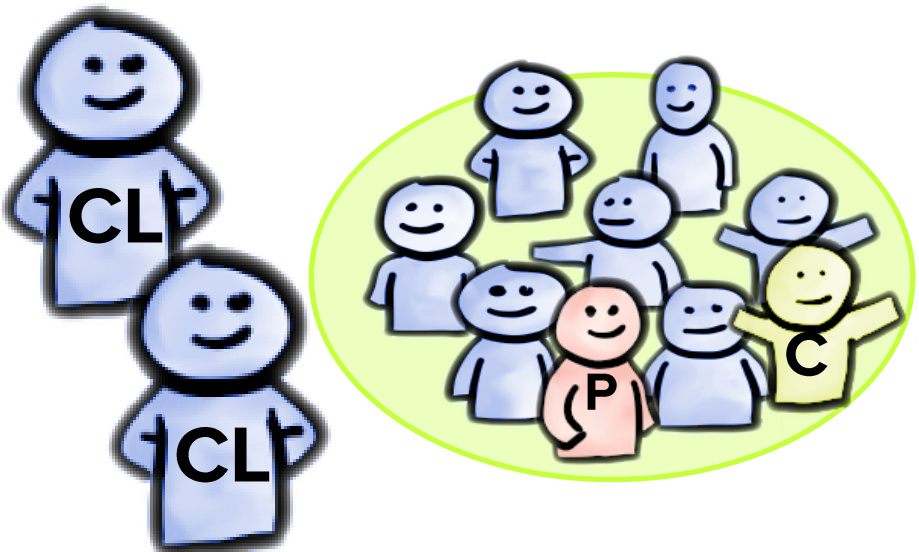
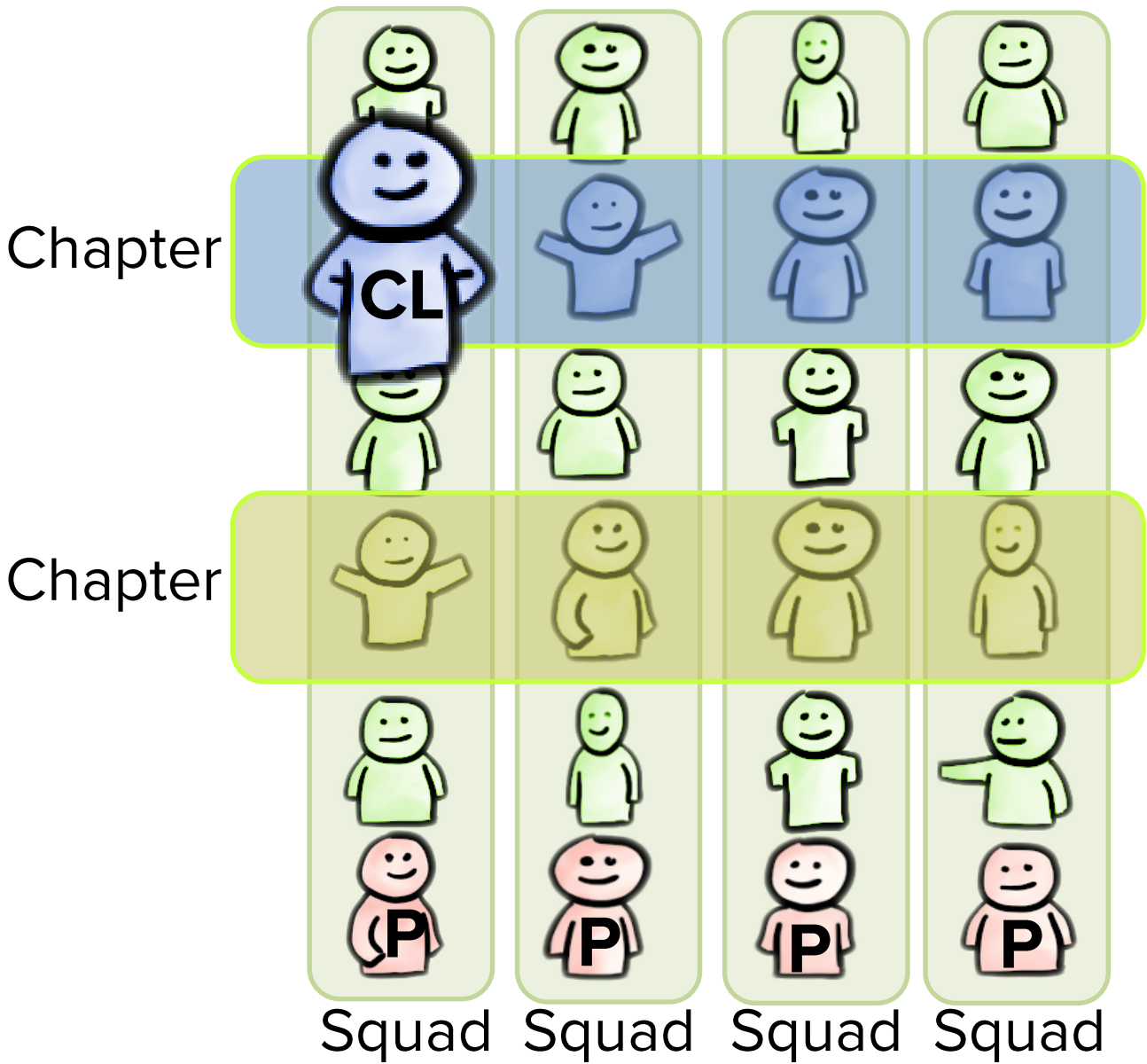
# Chapter Lead



## Examples

- Dev practices
- Tech backlog and tech debt
- Hiring

# Chapter Lead



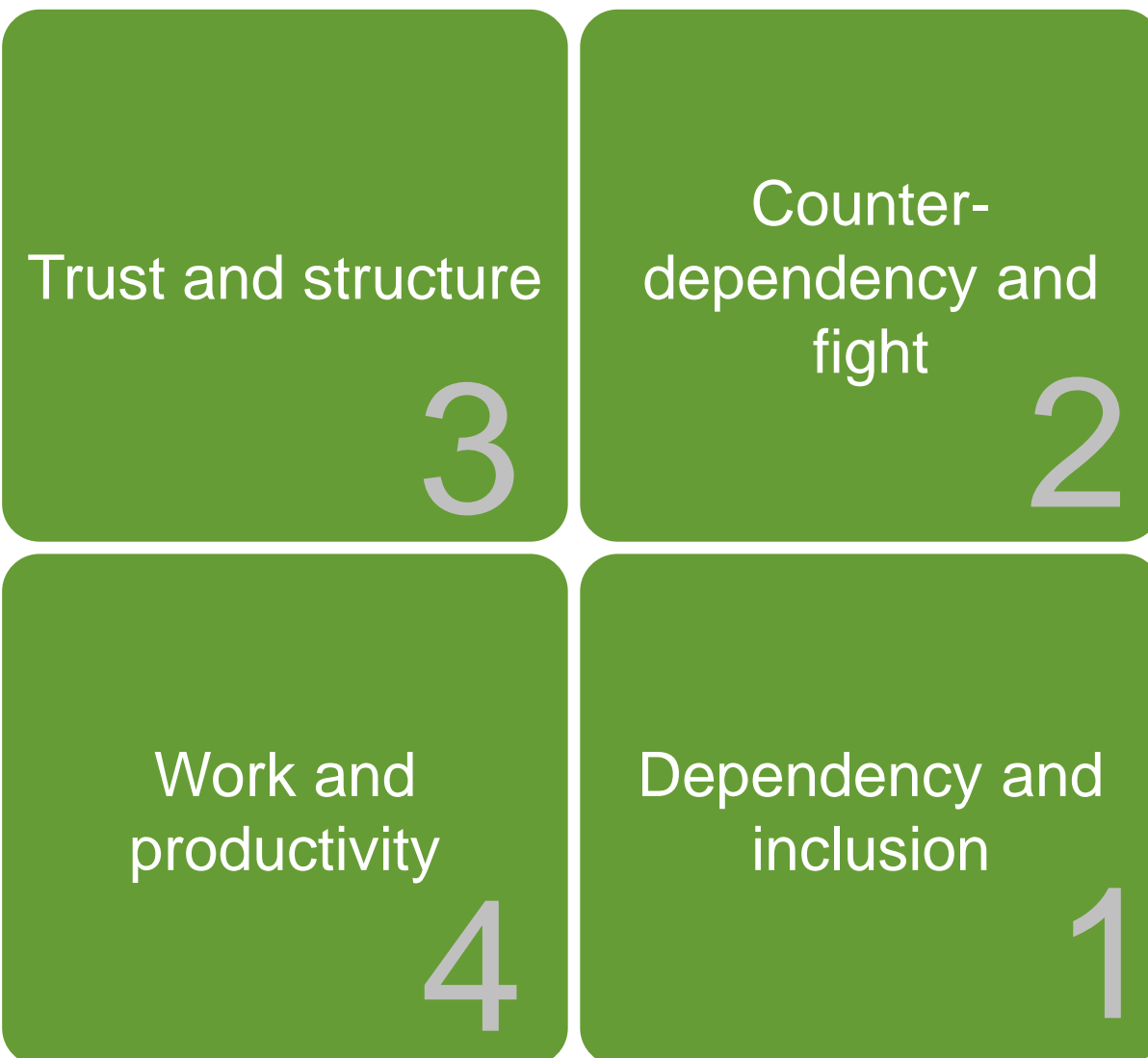
## Common patterns

- 50/50
- Prioritize squad
- Bumble-bee
- All-in



# High-performing squads

## TEAM DYNAMICS



## RESULT DRIVEN ACTIVITIES

*Lean & Agile methods*

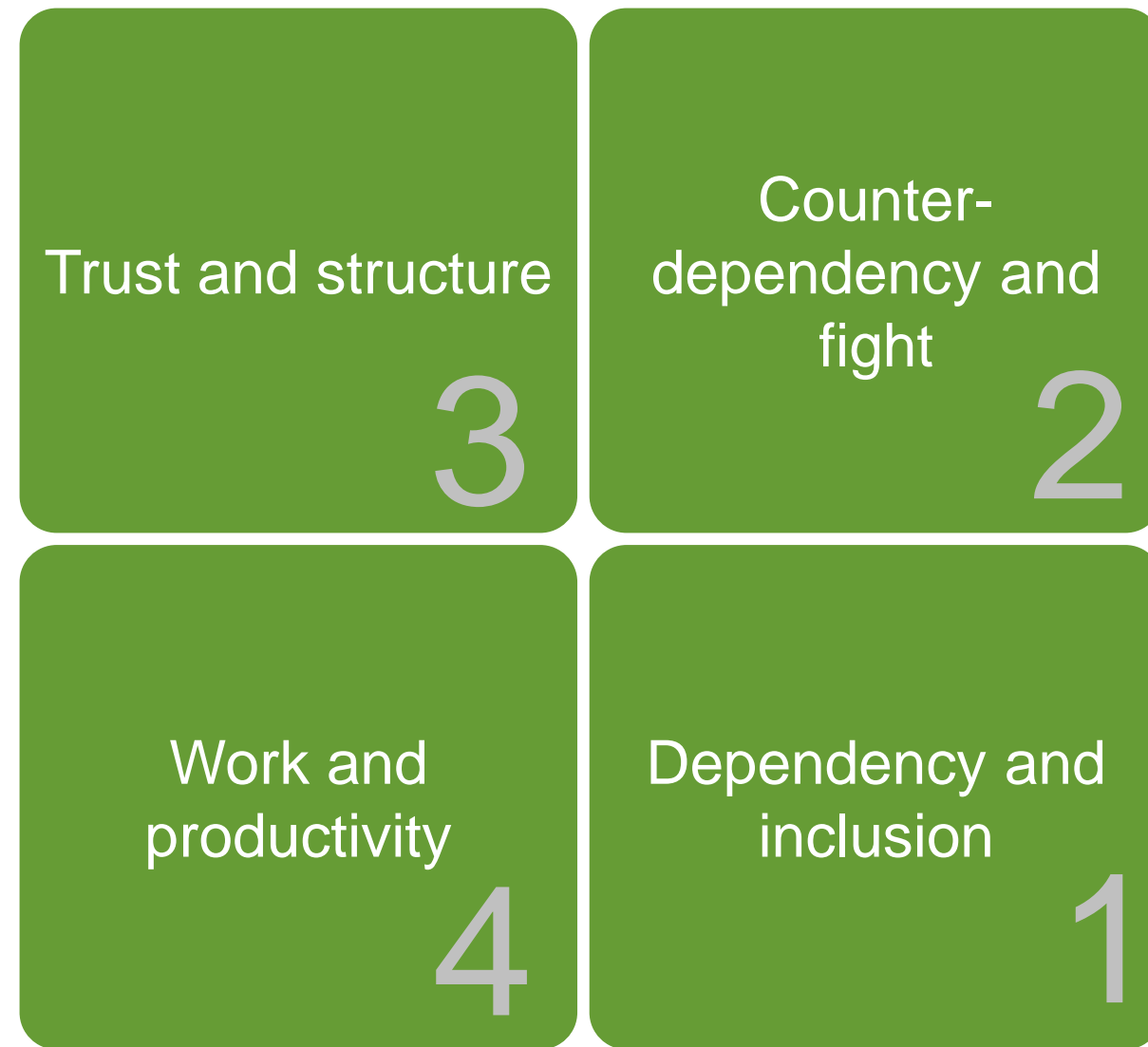
*Innovation*

*Quality* *Value*

*Speed*



# IMGD – Integrative Model of Group Development





## ***Characteristics:***



Looking for  
structure

Cautious

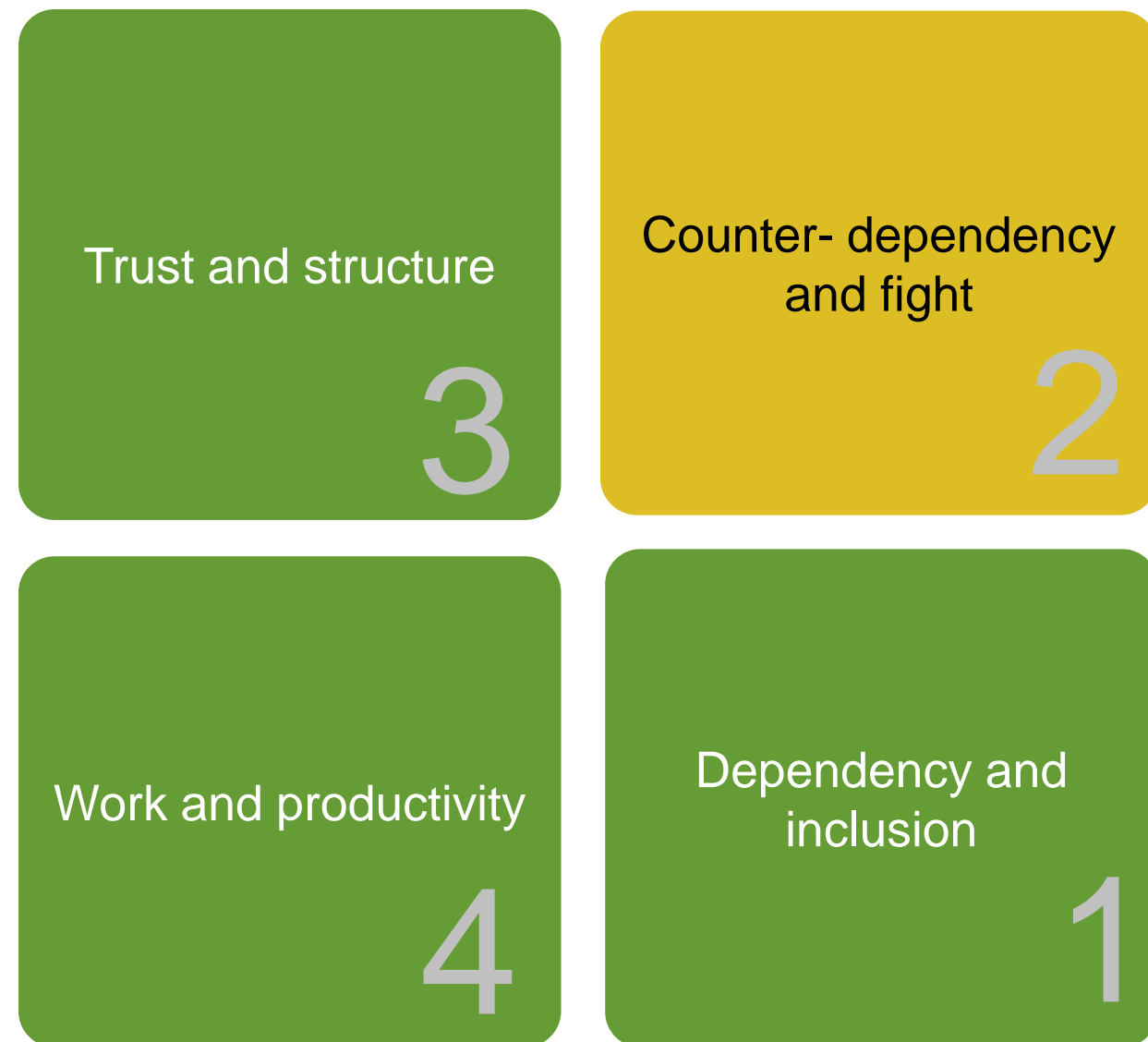
Polite

Avoiding  
conflict

Depending  
on leader

Endless  
discussions

## ***Characteristics:***



Questioning the leader and the goals

Conflicts

Competition

People profiling themselves

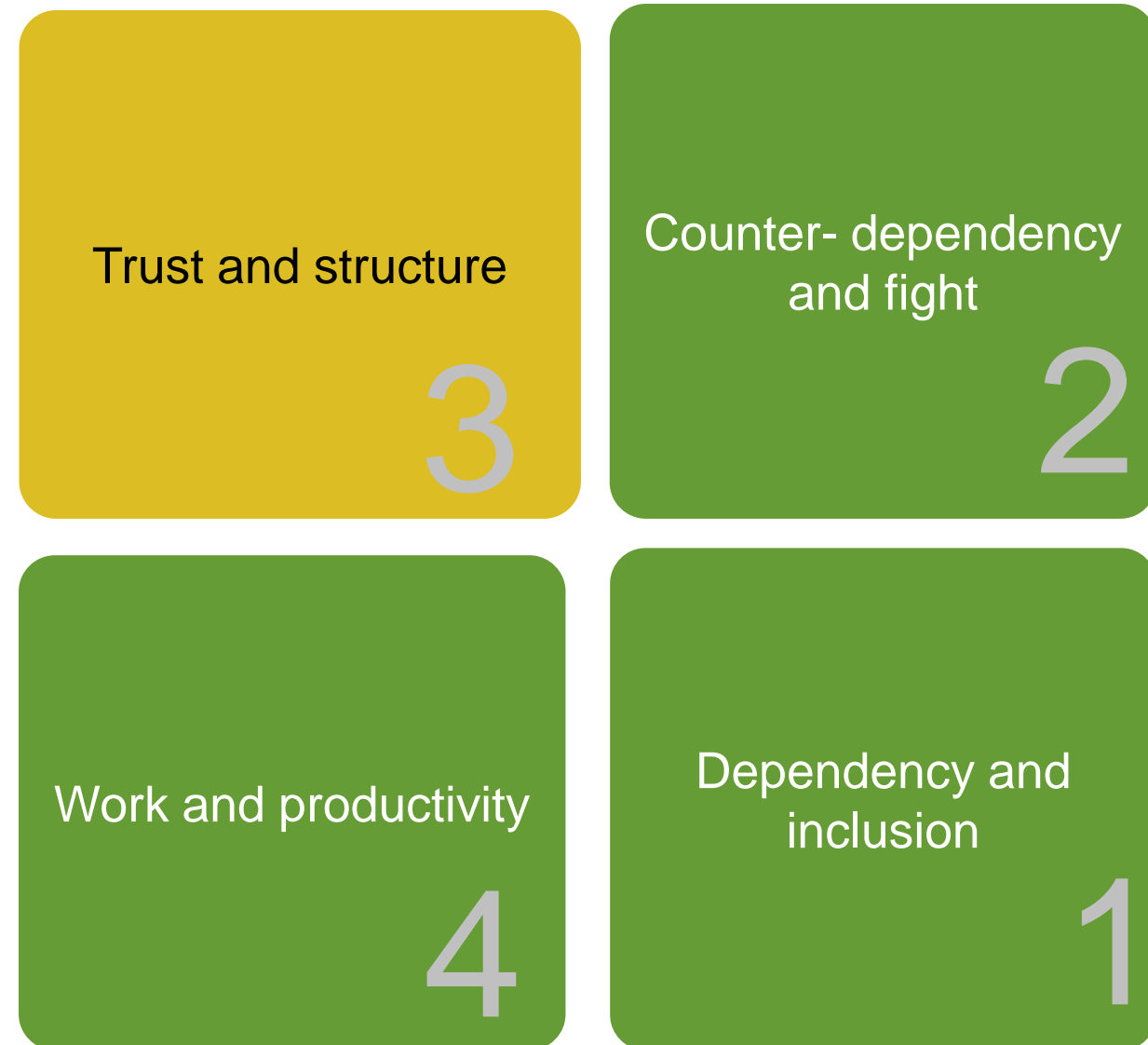
People don't listen to each other

People trying to convince of "their way"

Rather attack than support



## ***Characteristics:***



Clarity around goals

Conflicts handled

Listening

Openness

Consideration and care

Playfulness

## ***Characteristics:***

Trust and structure

3

Counter- dependency  
and fight

2

Work and productivity

4

Dependency and  
inclusion

1

Clarity around  
goals, roles and  
responsibility

Effective  
communication  
and handling of  
conflicts

Warmth and  
connection

Strong feedback  
culture

Engagement

Awareness of  
improvement  
possibilities



# High-performing squads

## TEAM DYNAMICS



## RESULT DRIVEN ACTIVITIES

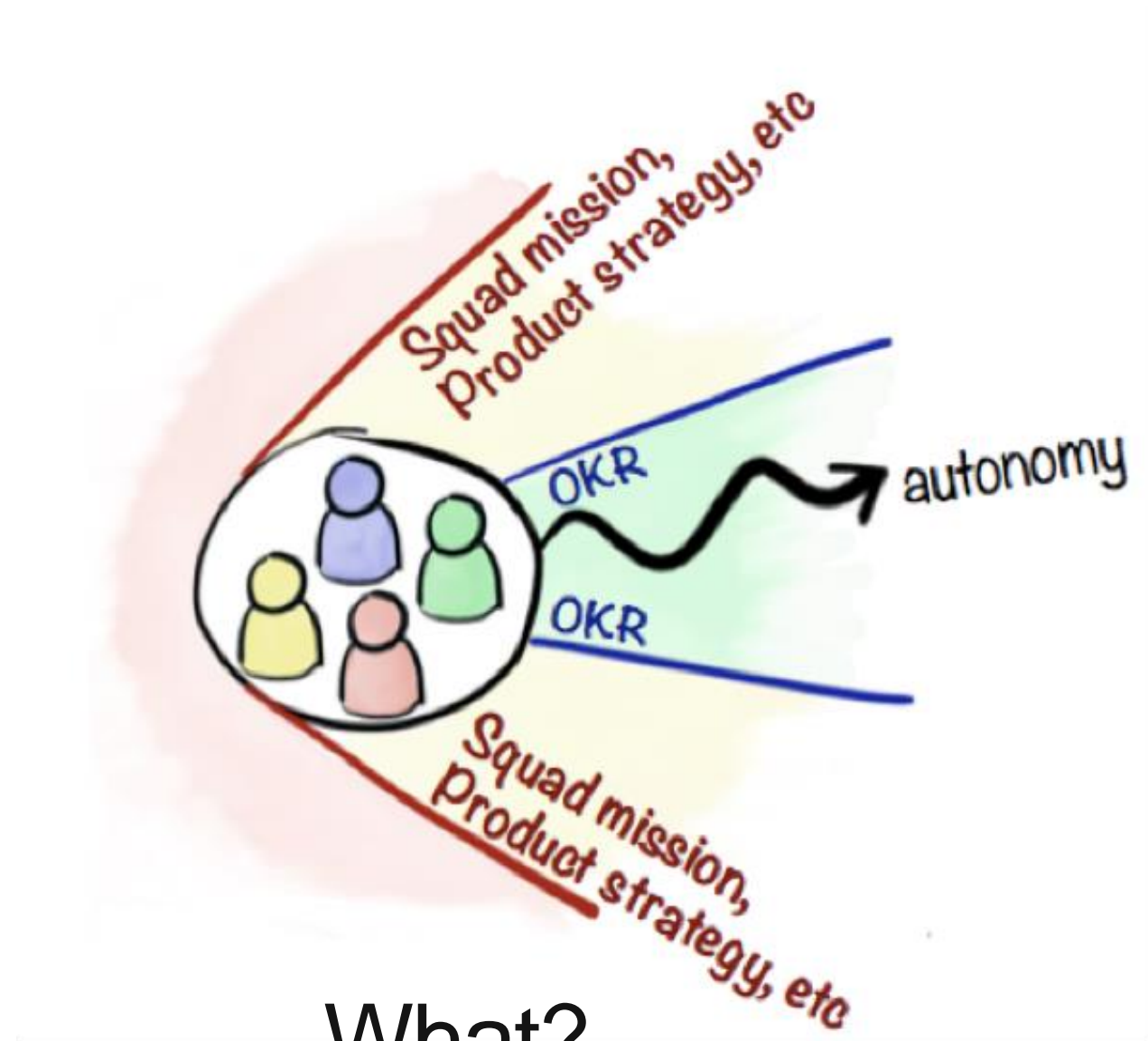
*Lean &  
Agile  
methods*

*Innovation*

*Quality* *Value*  
*Speed*



# Squad autonomy



# What?



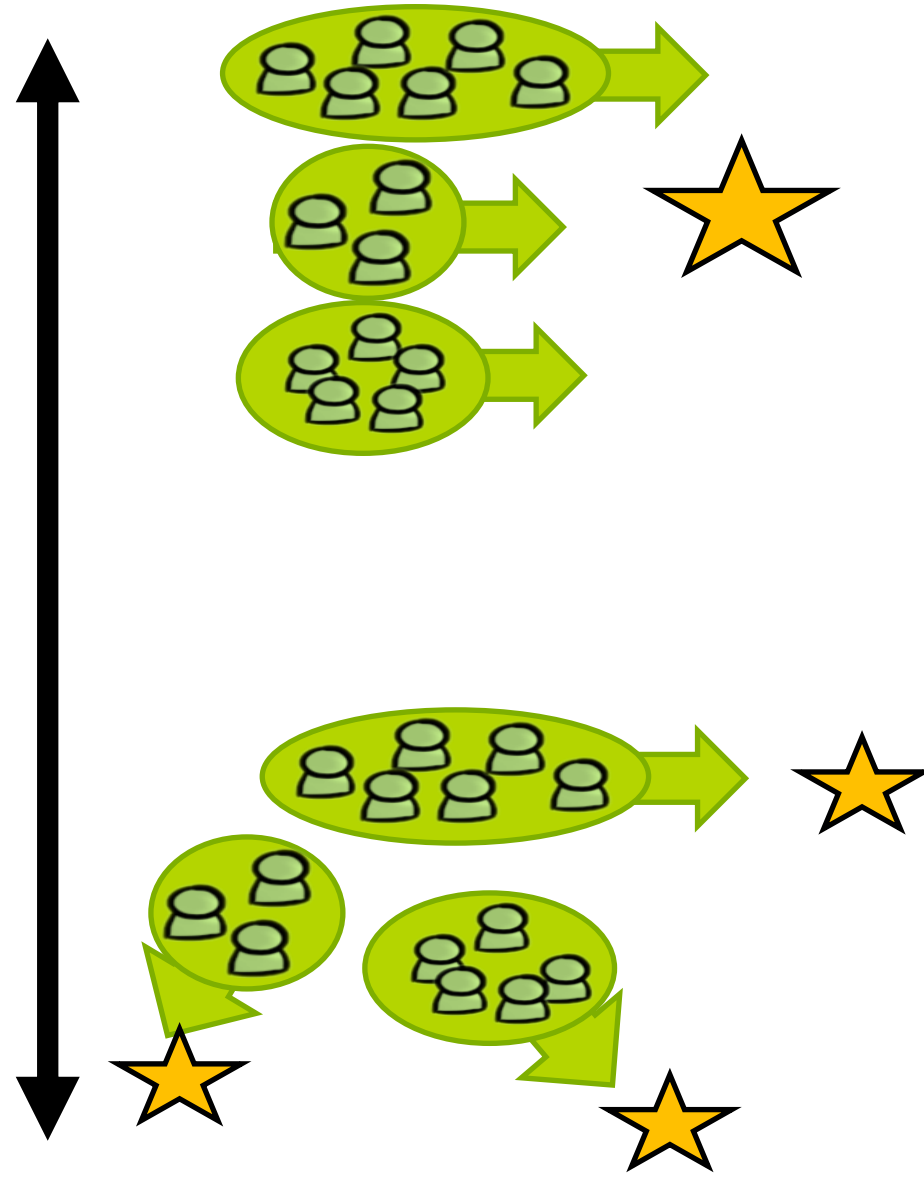
# How?



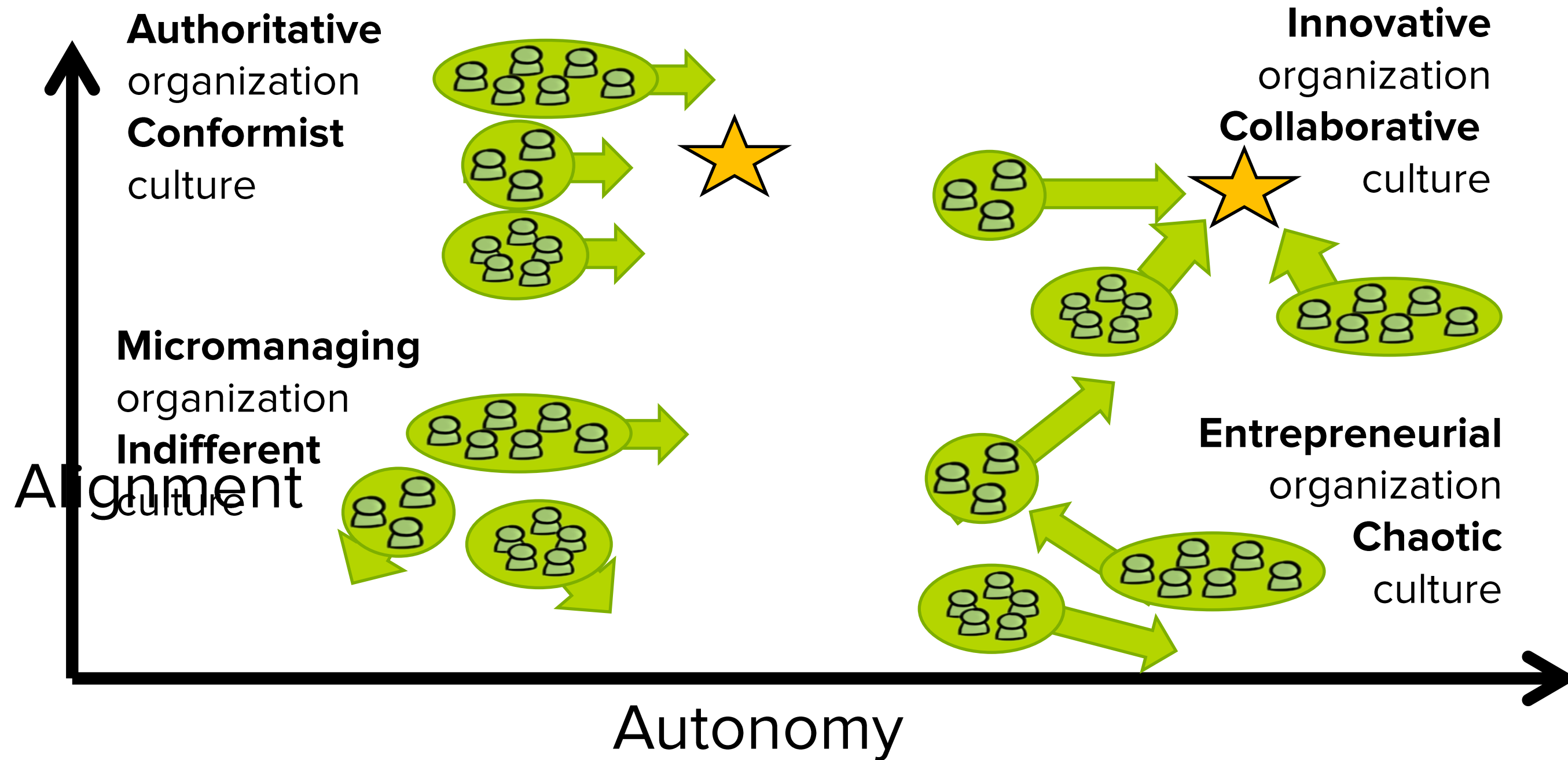
# Autonomy vs. Alignment

Alignment

Autonomy

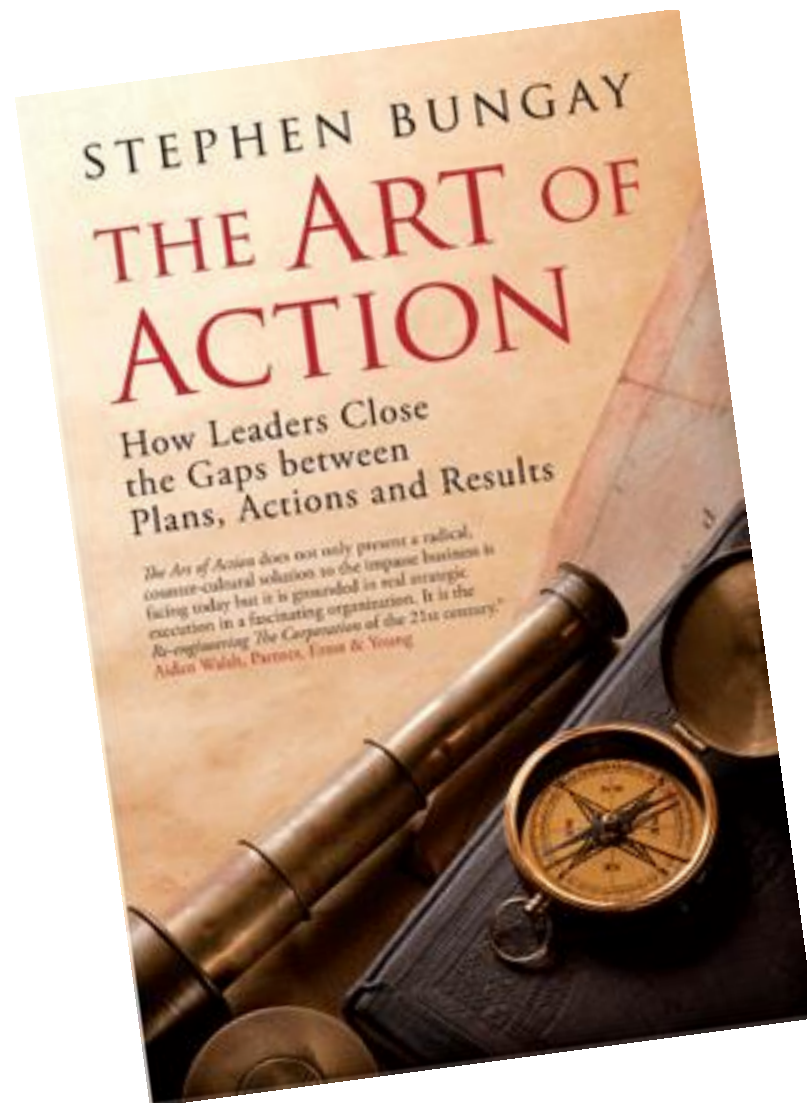


# Autonomy vs. Alignment





# Example: Ongoing experiment with Priorities



Help • 10

## 2012 Q3: Agile coach guild refined

Created by: **Anders Ivarsson**

5

### Objectives

CONNECTED WITH

+ Add people

- Anders Ivarsson
- Andreas Mattsson
- Daisy Pillbrow
- George McMonigle
- Ian Robbins
- Ingrid Franck

### Key Results

Period: Q3 2012 ▼

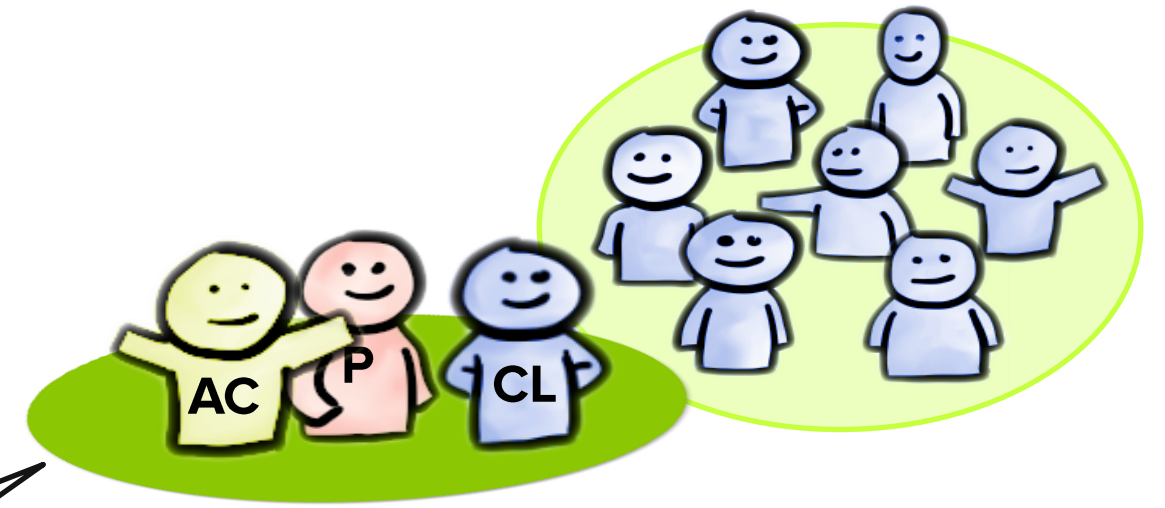
- Clear purpose and mission statement for agile coach guild published on the wiki **100%** ☐ Completed: Oct 16, 2012
- Click the [+] button to add a new action to this key result.
- Defined way of working - e.g. guild meetings, Popsagile group, squad/chapter/guild focus **20%** ☐ Sep 30, 2012
- Identify metrics for the agile coach guild to evaluate, that are related to company goals **20%** ☐ Sep 30, 2012

### Post update

Visible to people viewing this objective

Type @, then a person's name to notify them of your update.

# Squad support



How can we  
support our squad?

Product Owner

Chapter (Team) lead

Agile Coach



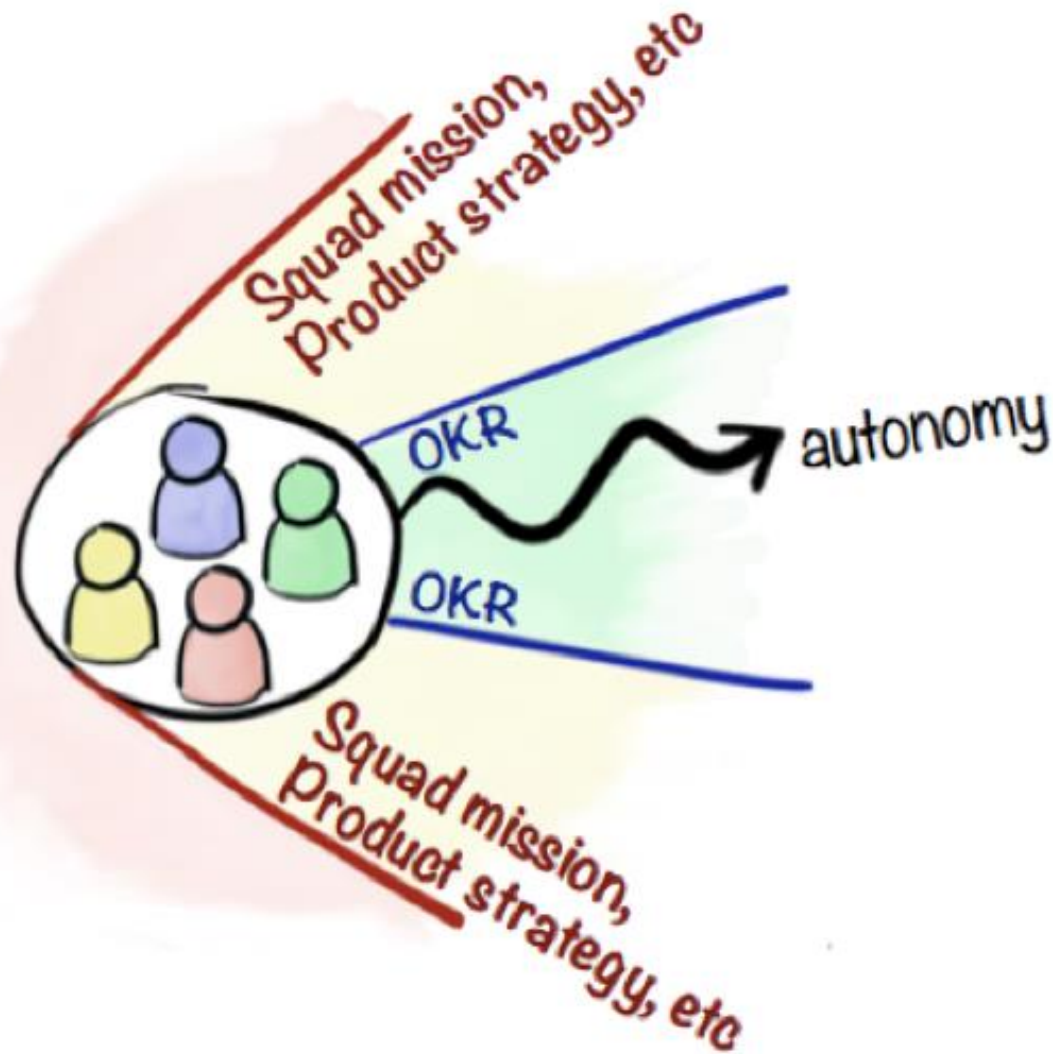
**"POTLAC"**



# Supporting squad autonomy



**Innovative**  
organization  
**Collaborative**  
culture



Be a good citizen in the  
Spotify **ecosystem**!  
Balance these:

1. Benefit Spotify as a whole
2. Serve the squad's mission and OKR:s

# Example: Dominating individuals in a squad



1. Give feedback
2. Improve team dynamics
3. Reinforce feedback culture
4. Nurture informal leadership



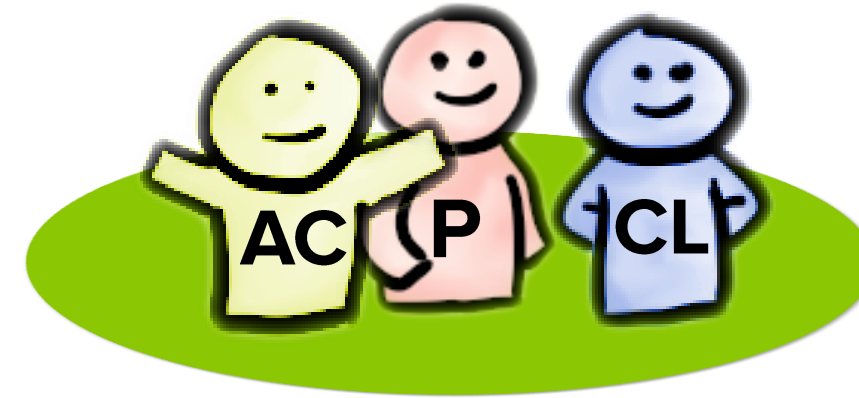
# Example:

## A squad is growing too big



1. Draft new missions
2. Facilitate squad split
3. Support individuals
4. Kick-off new squads

# Leadership teams



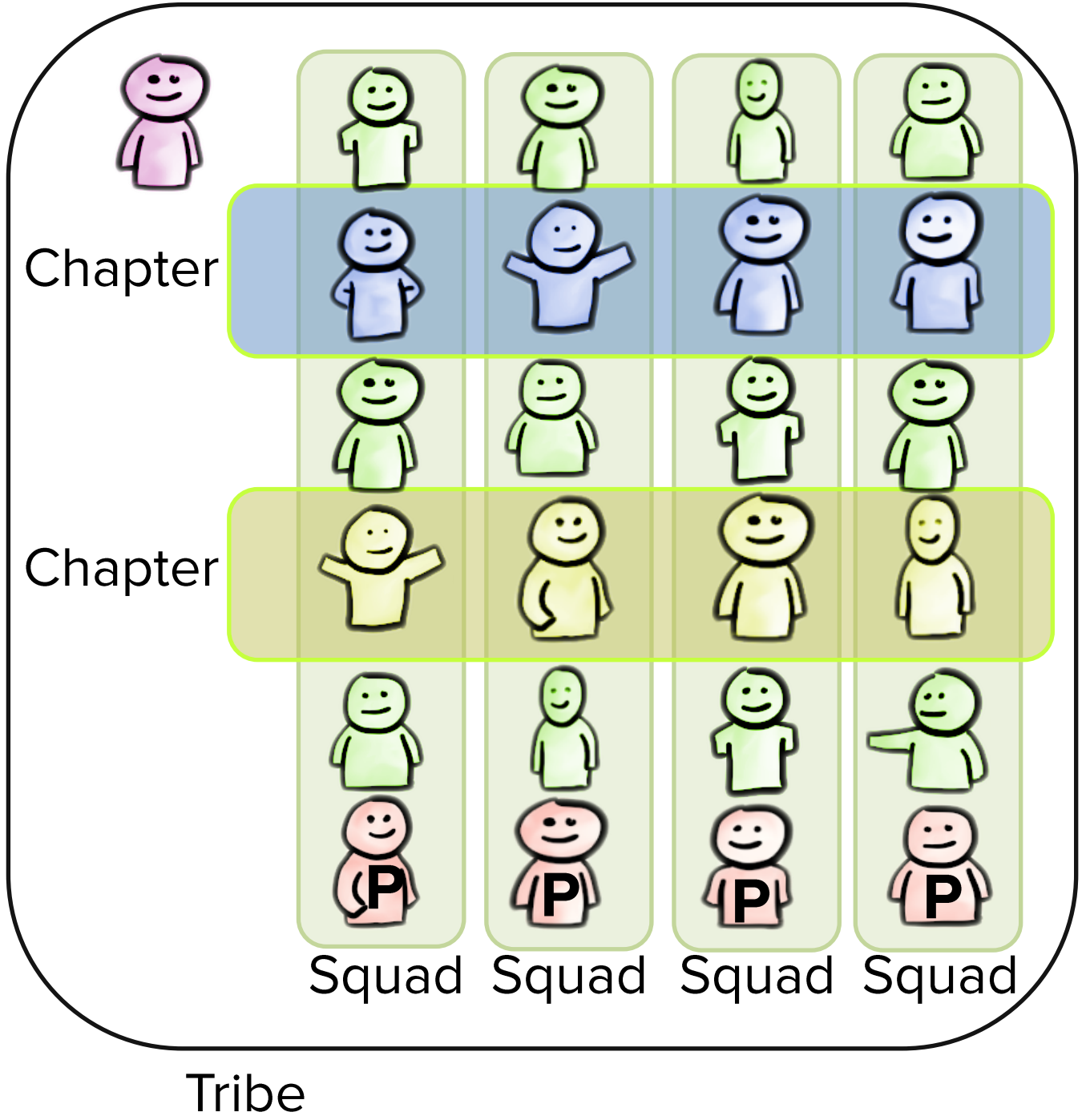
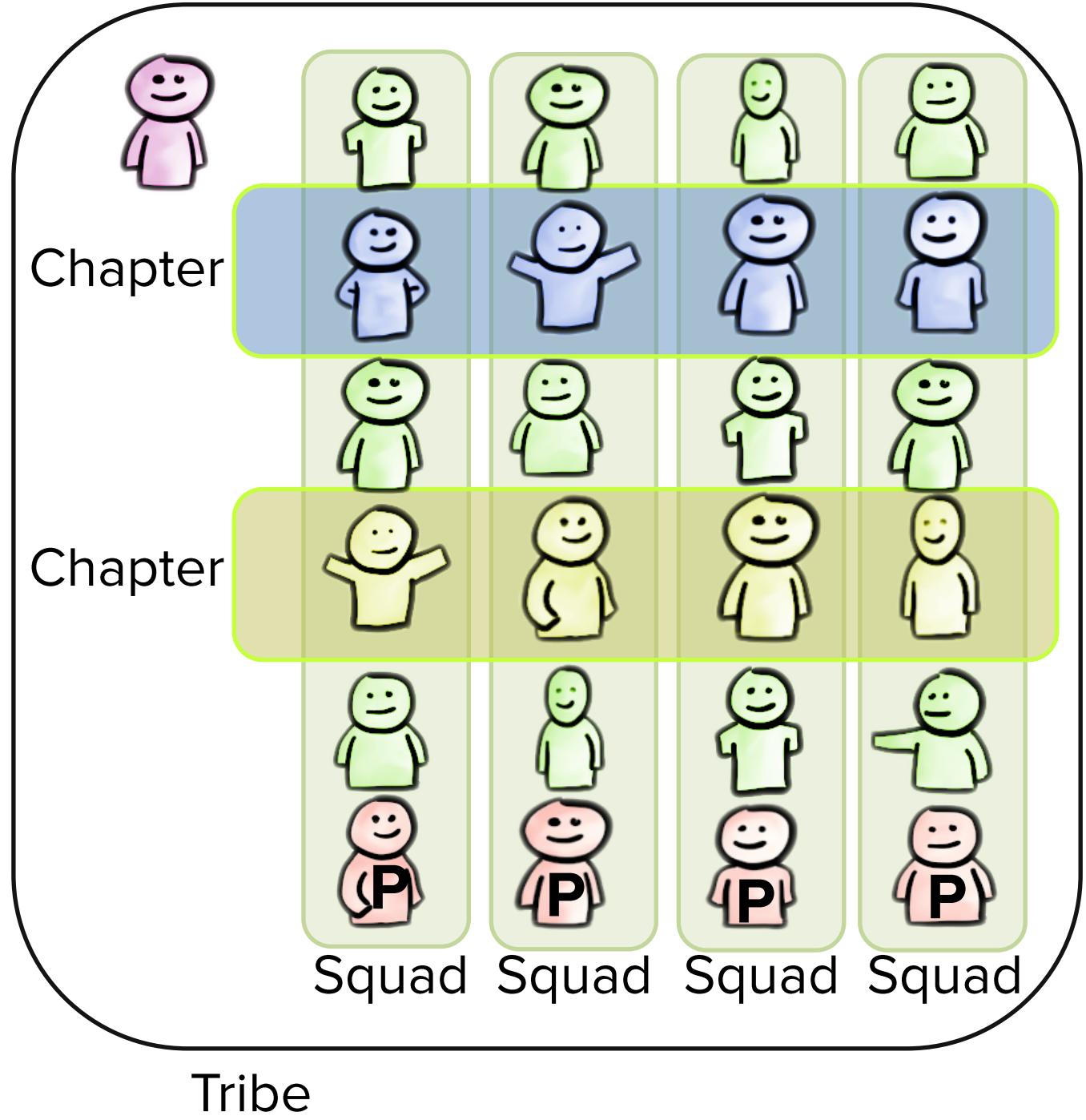


# Leadership – everyone!



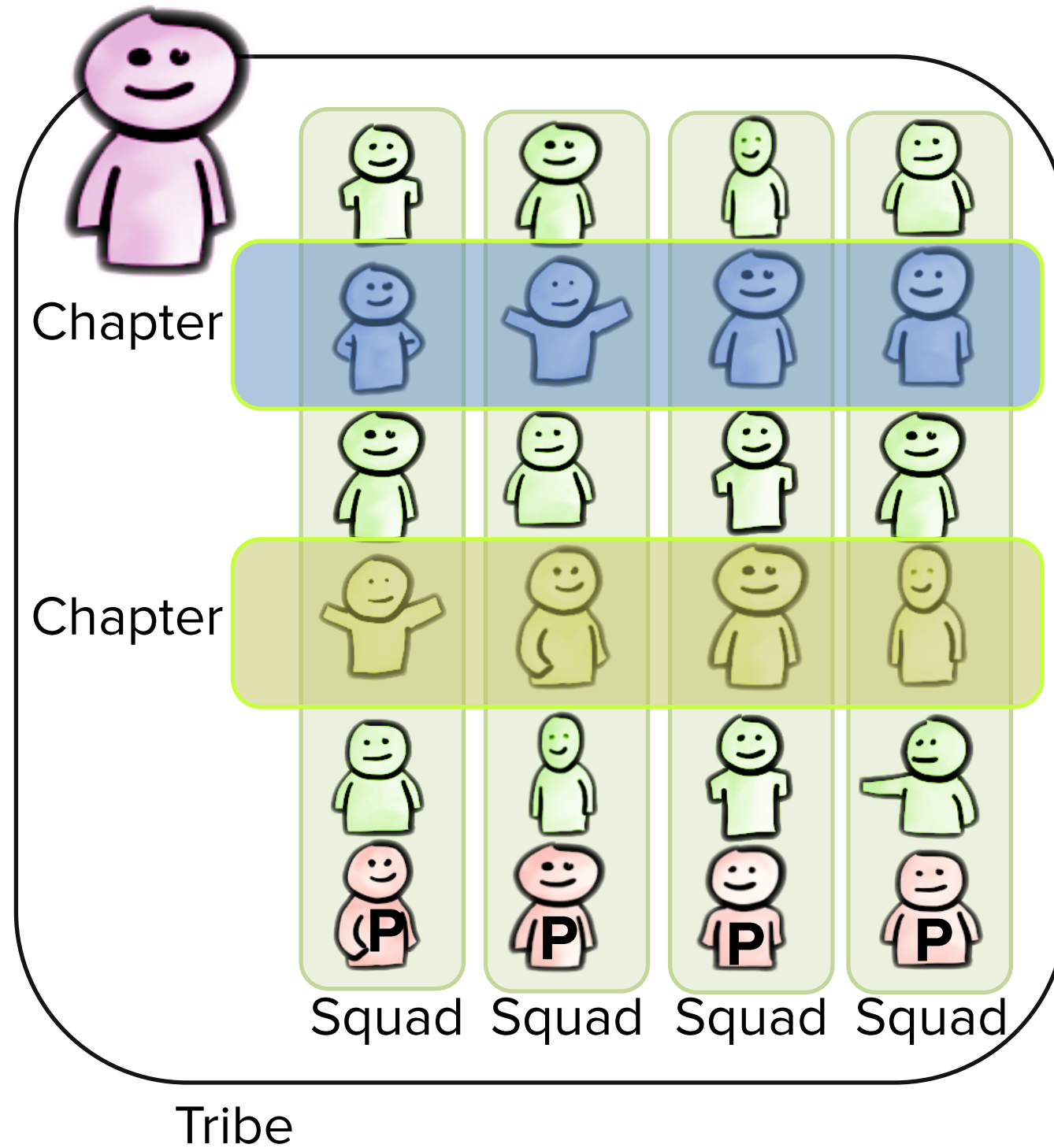
- Leadership is an act, not a role  
– everybody is a leader
- Informal leadership is the backbone of a strong squad
- Leadership guild  
¼ of the tech/prod organization

# Tribes



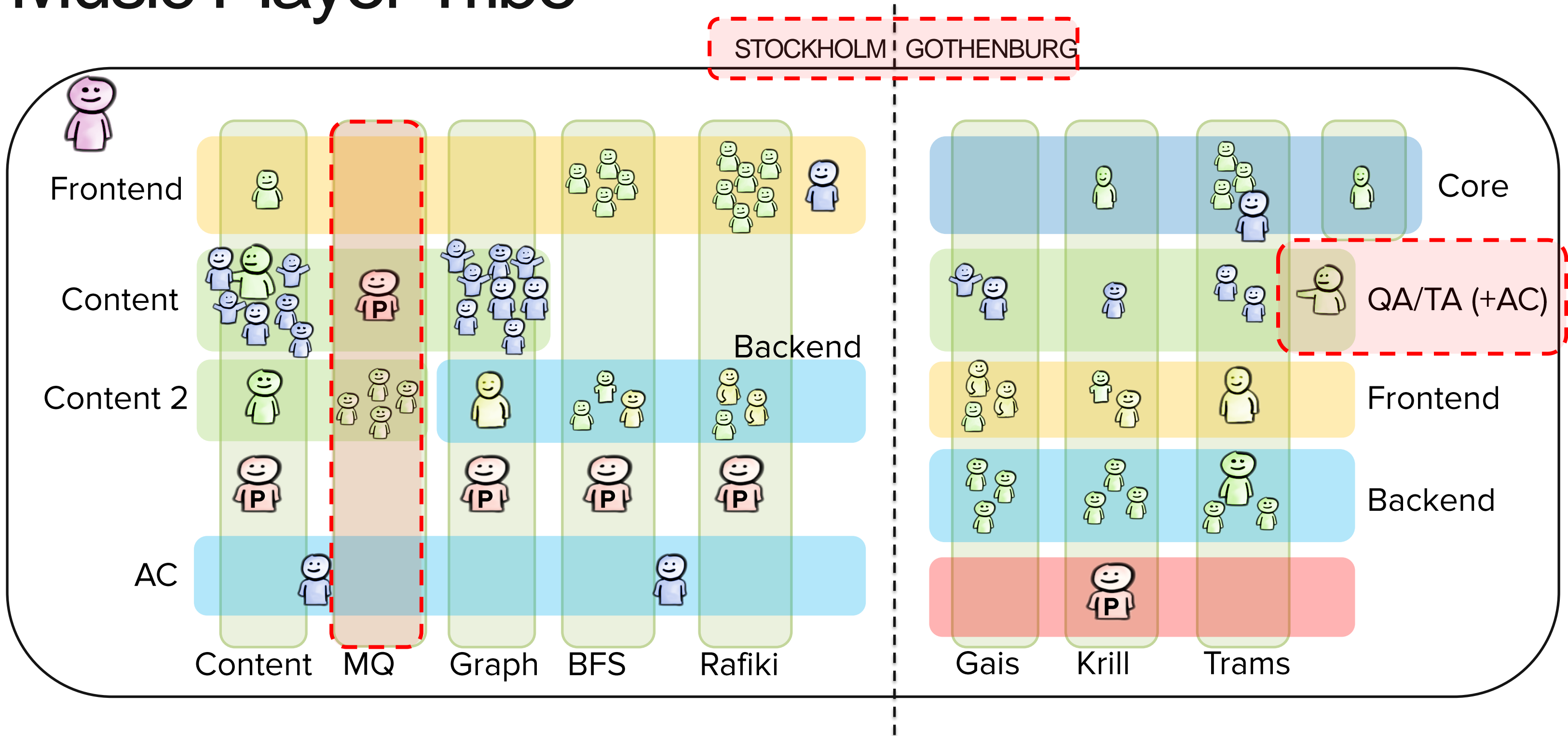


# Tribe Lead



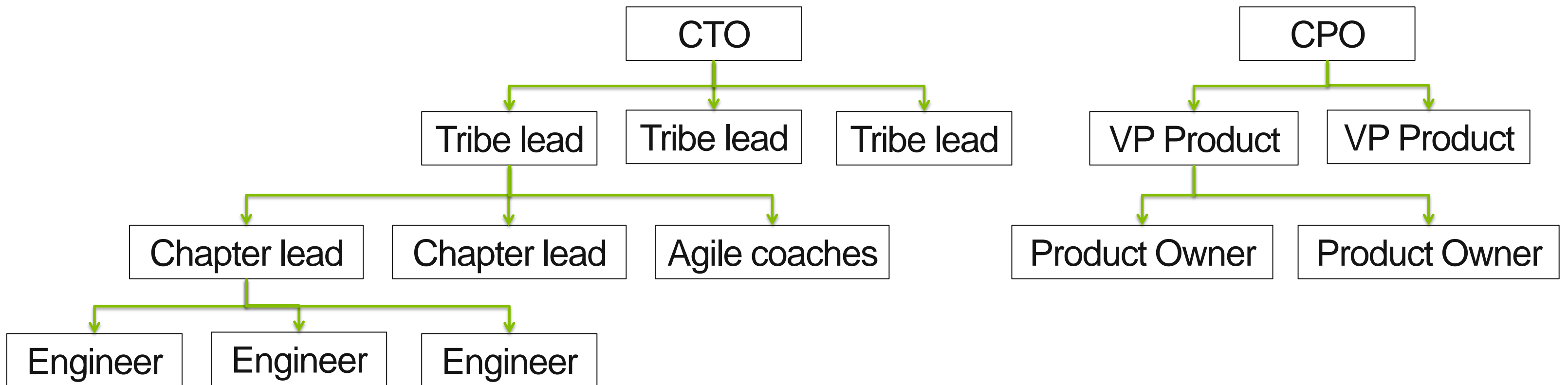
- High-performing tribe
- Grow strong chapter leads
- Coach the agile coaches
- Build a strong tribe leadership team

# Music Player Tribe

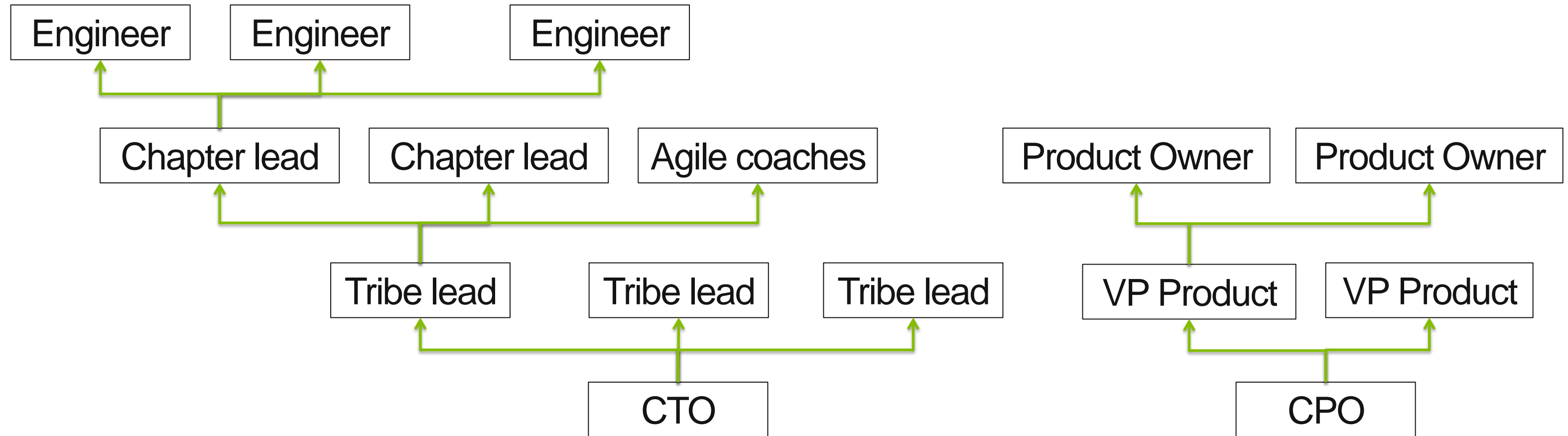




# The reporting structure

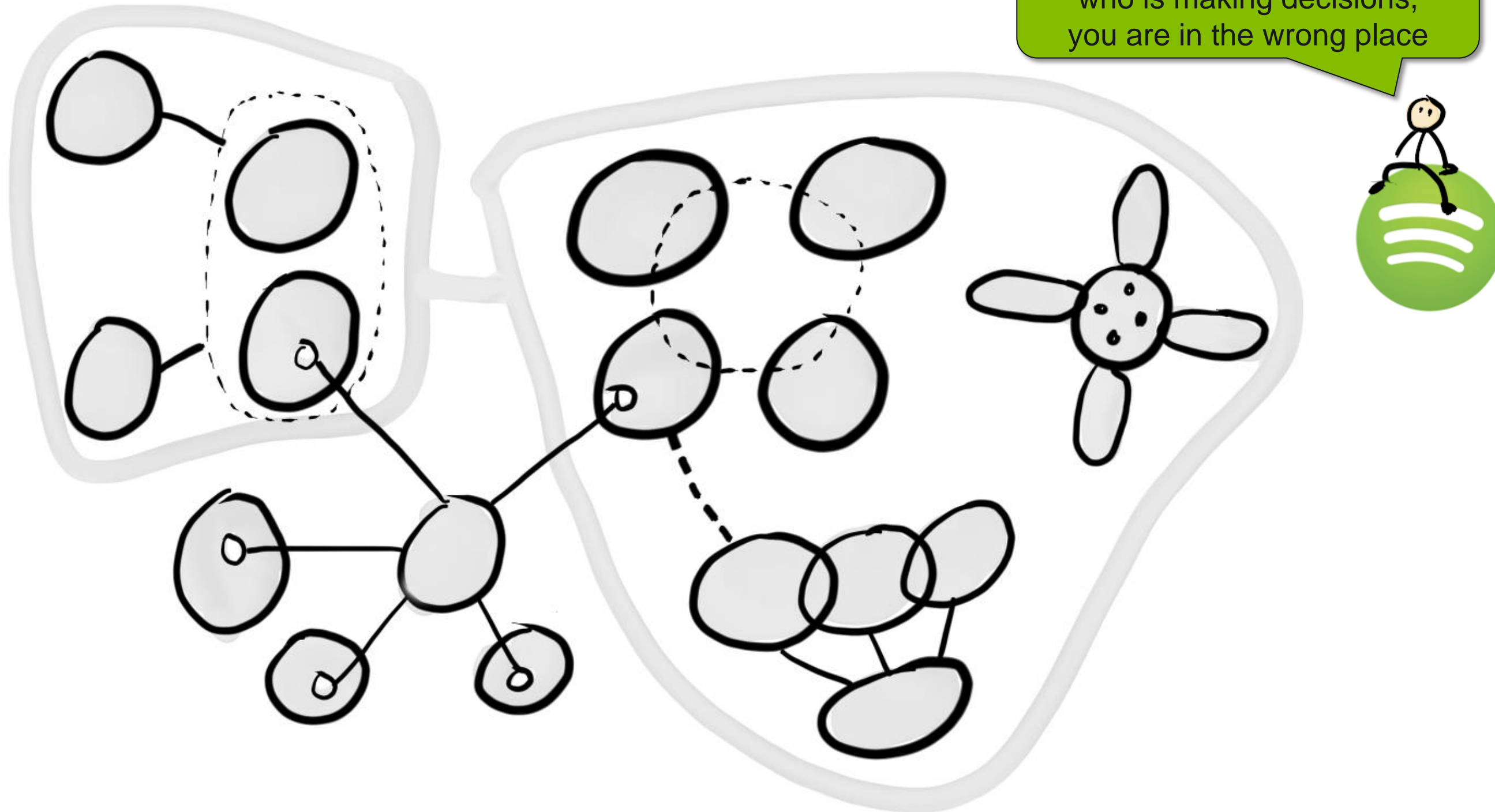


# The ~~reporting~~ structure supporting

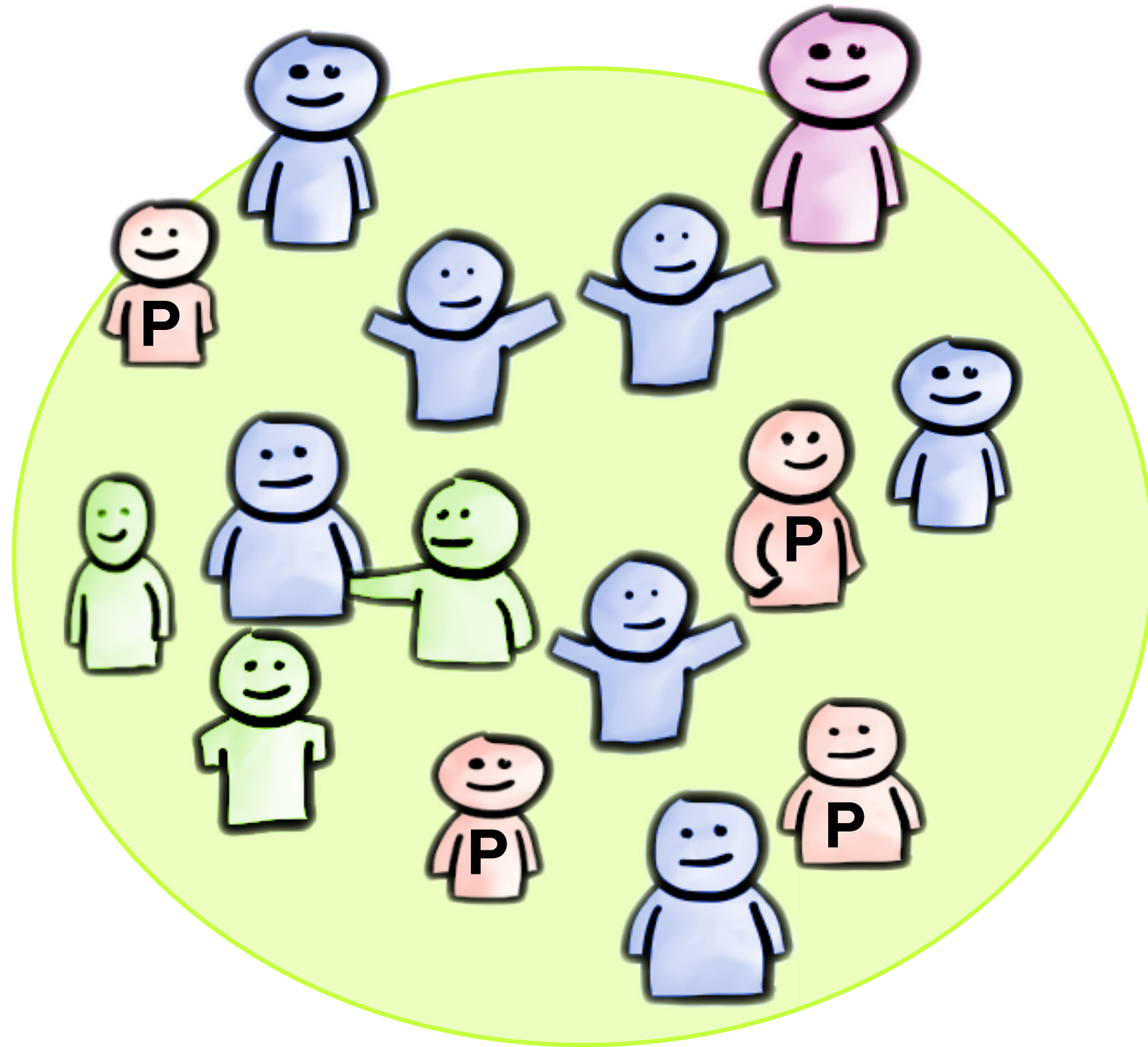




# Organic structure



# Tribe Leadership Team

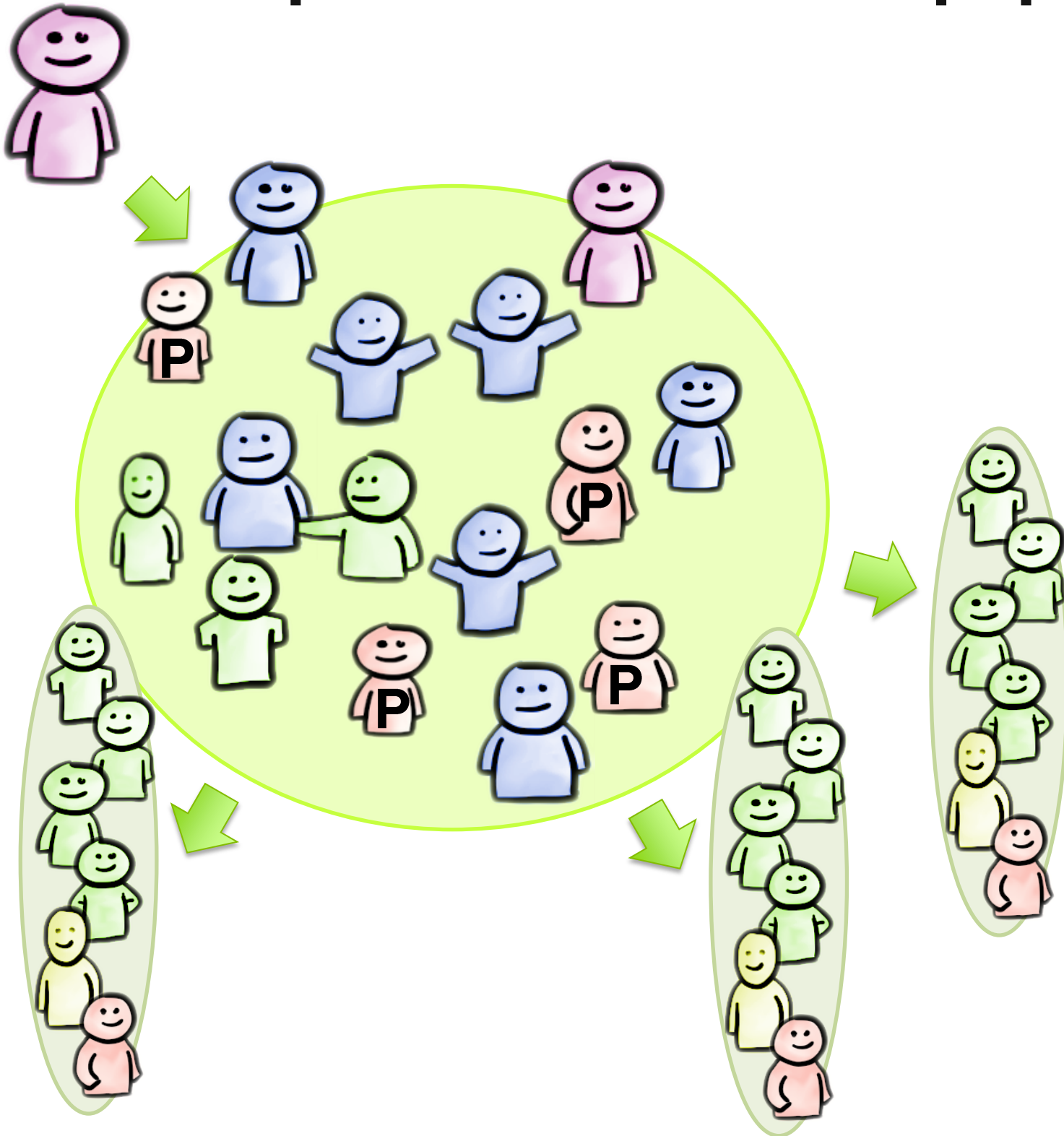


Tribe lead, CL, PO, AC

## Examples

- Recruitment / onboarding
- Grow high-performance teams
- Culture
- Tribe WIP
- Alignment

# Example: Leadership post-mortem



Pressure and stress spreads,  
and drives bad behaviors

Post-mortem – breakdown of leadership

Improvements

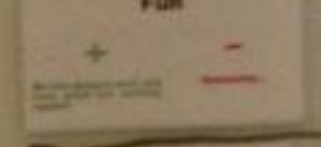
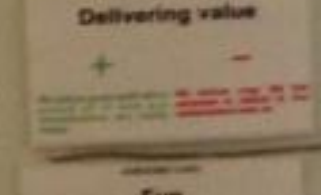
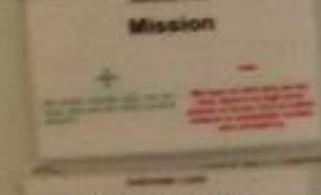
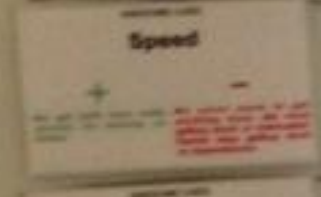
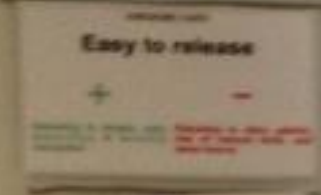
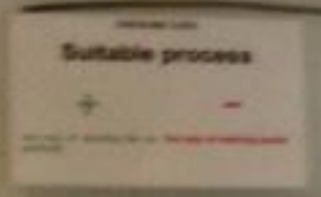
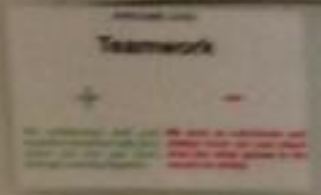
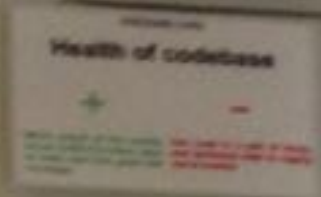
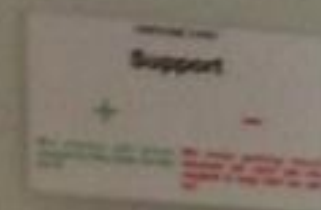
- Multiple stakeholder channels
- Share pressure within lead team
- Continuous delivery
- No Friday releases



How do we know how we're  
doing?

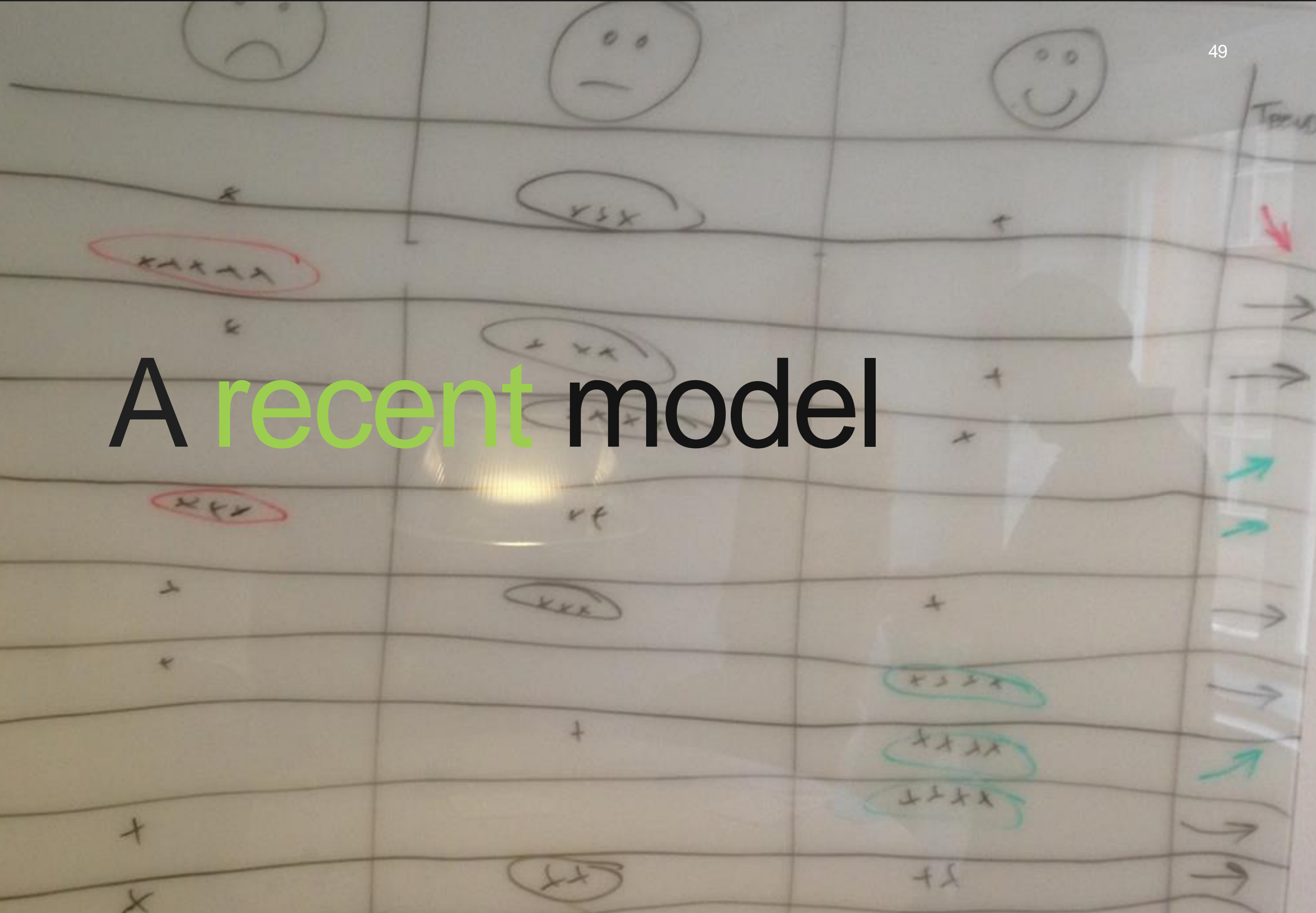


# A recent model



CI-  
TIDE

CI-  
TIDE





An Example...





# Maslow



- Self-actualization
- Self-esteem
- Belonging
- Safety
- Physiological

# Maslow



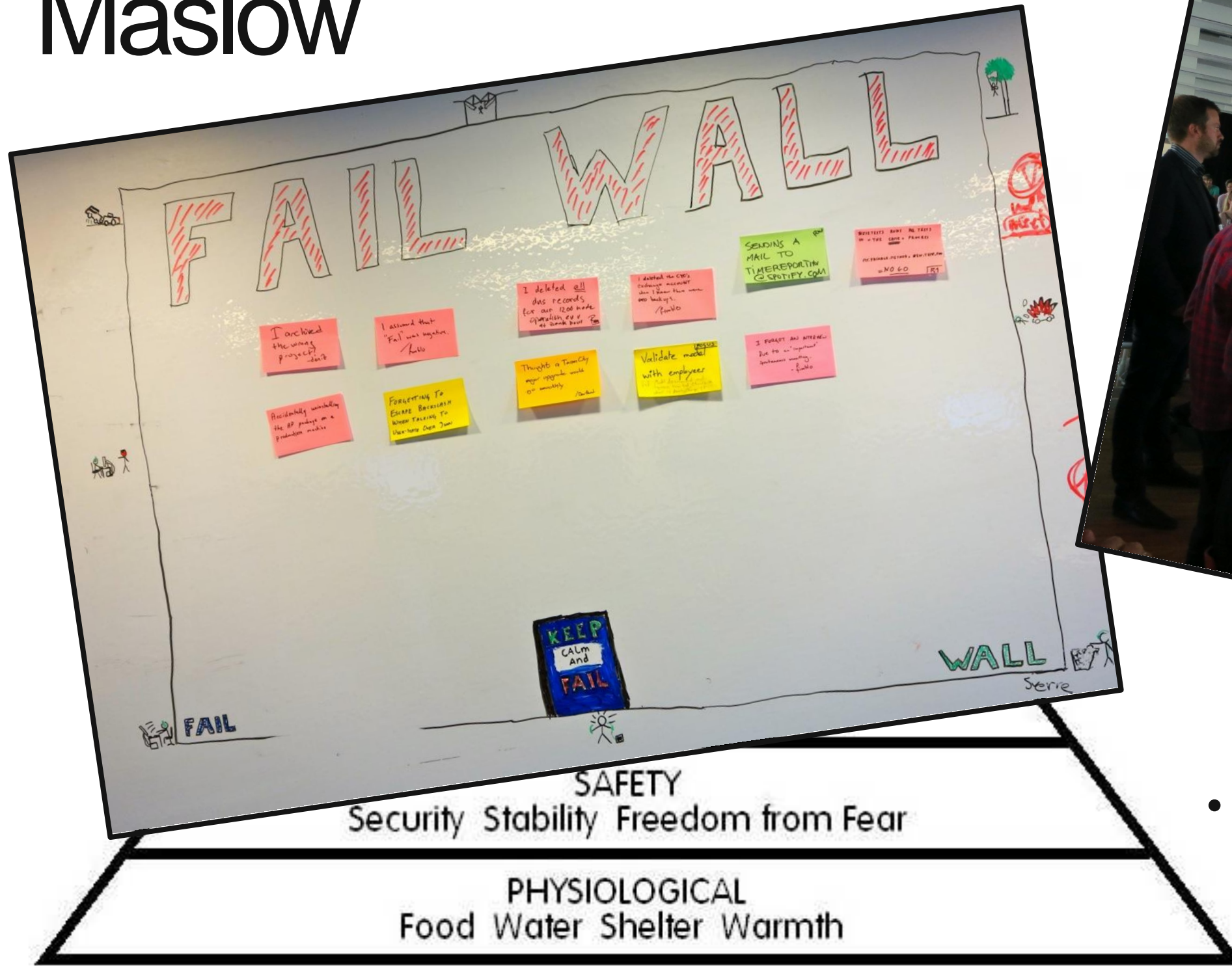
Physiological  
Food Water Shelter Warmth



- Physiological – kickass work environment



# Maslow



- Safety – no fear culture, embrace failure



# Maslow





# Maslow



- Self-actualization

- Self-esteem

are very important

BUT don't forget – no fish air

# Values → Culture → Org Structure

Empowerment → Autonomy → Squads

Trust → Servant leadership → Chapters

Collaboration → Teamwork → Teams on all levels





# Summary

Org structure < formal leaders < informal leadership < culture and values





# Thank you!



Anders Ivarsson  
@anders\_ivarsson



Kristian Lindwall  
@klindwall