Managing Teams in software engineering

or

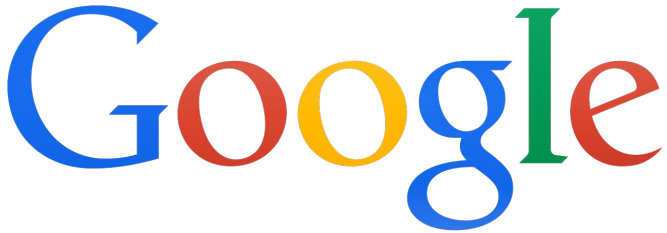
Some stuff about teams

This is a talk about lots of things

- Why we talk a lot about teams in technology and software development companies
- My experiences of ways of working and “why teams”
- Some ways that we use of making teams work
- Showing you the “management” tools used in companies
- Offer some thoughts about teamwork and trust
- I love making slides like this
- But I know you’ve already stopped listening to me
- I’m probably only on the 4th line by now
- You’ve already finished the slide
- I’ll be right with you

I hope you take something useful away from this

Even if that's just keywords to search for later



Who am i?

Hi, I'm Pieter!

this is me

yeap, that's a glitter mohawk

I am very proud of it!

About me-as-human

- I have been a developer, an architect, a (technical) PO and a team manager.
- I will take any excuse to use a whiteboard.
- I love the high level design work, but I get lost in implementation detail when under pressure.
- I have witnessed Conway's Law in action, and performed the Reverse Conway Maneuver.
- Flattery will basically get you everywhere.

About me-as-human

- I have been a developer, an architect, a (technical) PO and a team manager.
- I will take any excuse to use a whiteboard.
- I love the high level design work, but I get lost in implementation detail when under pressure.
- I have witnessed Conway's Law in action, and performed the Reverse Conway Maneuver.
- **Flattery will basically get you everywhere.**

I've seen some things ...

Team models I've seen

"Lone wolf"

One person, one code base, one problem.

Team models I've seen

"Specialist Team"

University of Kent

University of Essex

ultraSOC

insensys

smoothwall
The Web You Want

Totaljobs

jobsite

StepStone UK

Multi-disciplinary, individual-goal-driven team.

Team models I've seen

University of Kent

ultraSOC

University of Essex

insensys

"Agile teams"

smoothwall
The Web You Want

Totaljobs

jobsite

StepStone UK

Multi-disciplinary, shared-goal-driven team

Team models I've seen

No team

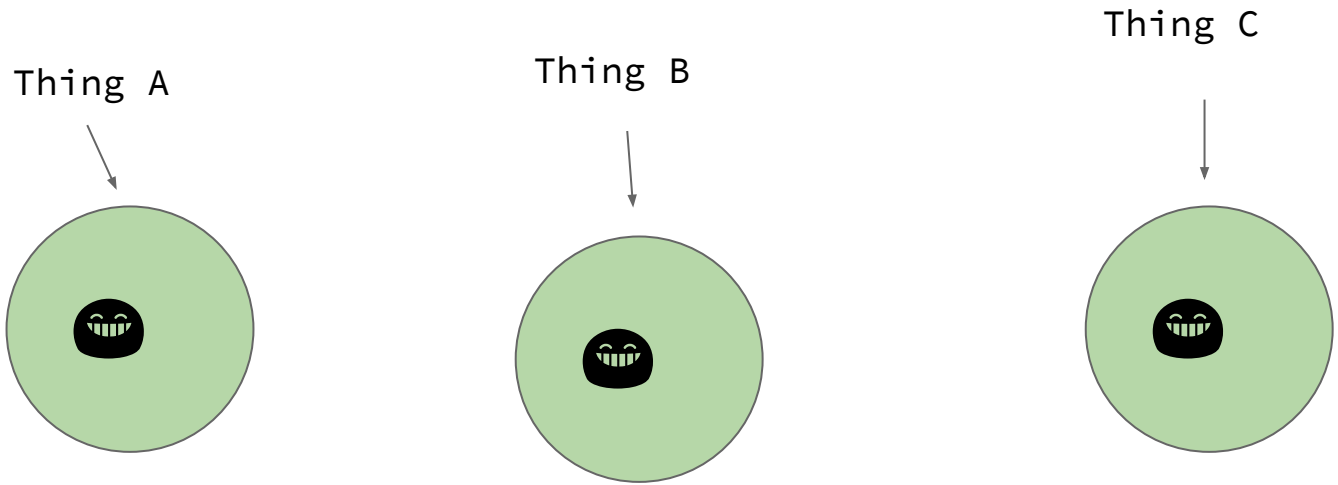
One big-ish 5-9 people team

Lots of 6-people teams

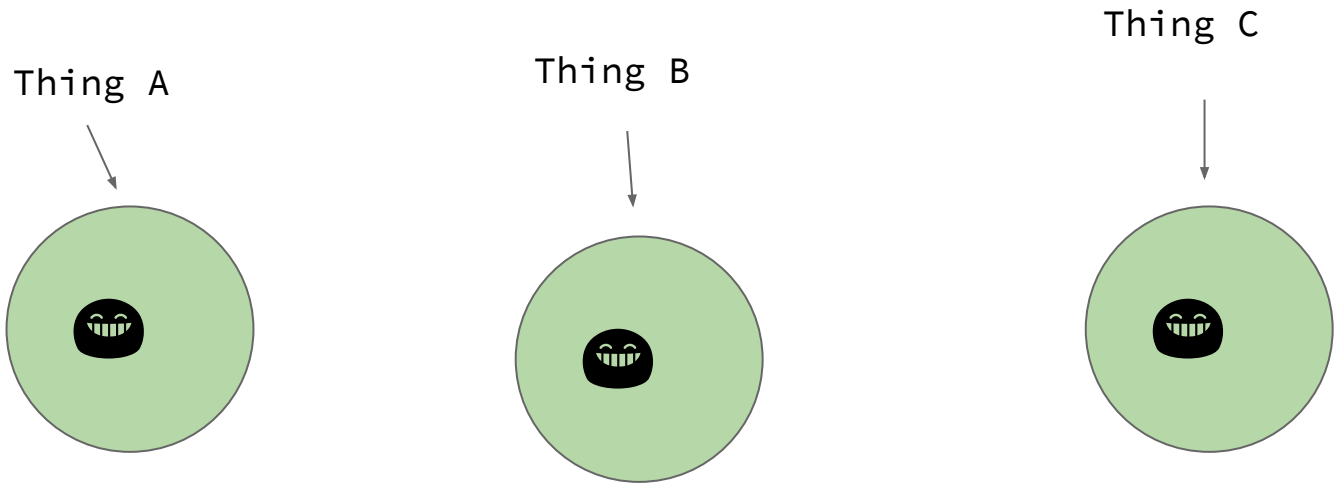
Why bother with groups of people?

Because we want to get more work done.

More people == more things!



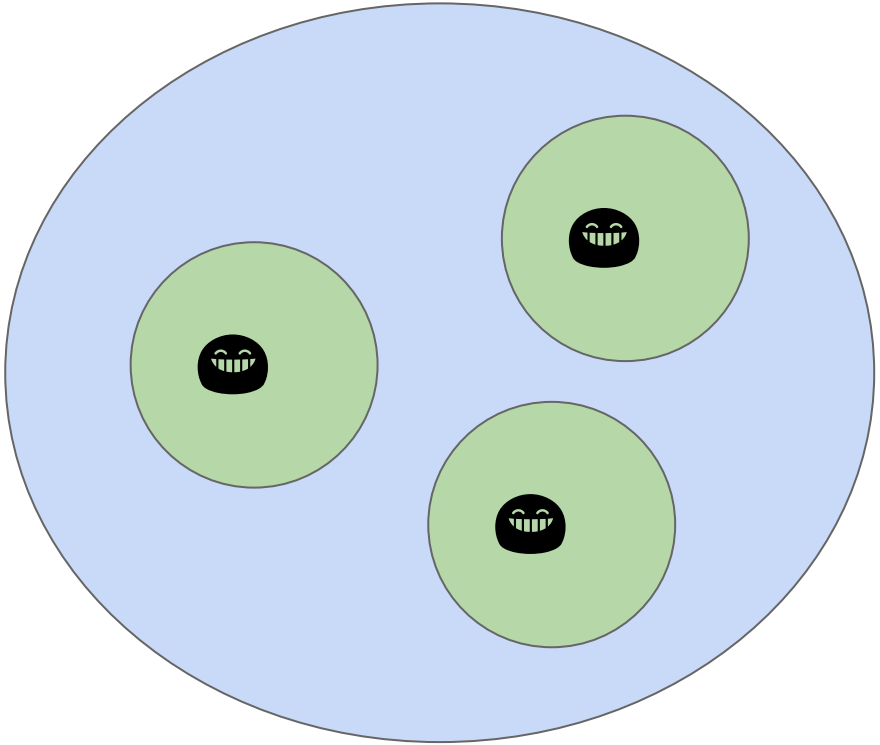
More people != team



Team == more consistency

Todo:

1. Thing A
2. Thing B
3. Thing C



So let's talk about teams!

TEAMS ARE HARD WORK . . .

“I worked in a team once.
I am now sad.”

People

~~TEAMS~~ ARE HARD WORK . . .

Synchronising brains can be hard work.
It means talking to each other.
A lot.

A quick bit of maths

More work = teams

Cost of teams = communication

Therefore:

Cost of (more work) = communication

Communications = latency cost

1 brain -> Processor cache

2 brains -> RAM

7 +/- 3 brains -> HDD

Lots of brains -> Network

Communications = latency cost

1 brain -> Processor cache

2 brains -> RAM

7 +/- 3 brains -> HDD

Lots of brains -> Network

Communications = latency cost

1 brain -> Processor cache

2 brains -> RAM

7 +/- 3 brains -> HDD

Lots of brains -> Network

Communications = latency cost

1 brain -> Processor cache

2 brains -> RAM

7 +/- 3 brains -> HDD

Lots of brains -> Network

Can't we all just work on something different?

Failures of splitting a team

- Low variety of ideas, therefore poor solution
- Low awareness of external factors (like other people's time or changes)
- Not being held to account (code review, TDD, documentation)

Okay, teams might be good

Benefits of agile teams

1. More diversity == more ideas
2. Knowledge sharing
3. Resilient against changes

Benefits of agile teams

1. More ideas
2. Sharing knowledge is cheaper -> more resilience
3. Resilient against changes

Benefits of agile teams

1. More ideas
2. Knowledge sharing
3. Resilient against changes -> consistent team output

Benefits of agile teams (for companies)

1. More ideas
2. Knowledge sharing
3. Resilient against changes ->
   consistent team output ->
4. Higher predictability

Resilience is a function of scale

1 brain -> Processor cache

2 brains -> RAM

7 +/- 3 brains -> HDD <- non-volatile system

Lots of brains -> Network

Agile teams are typically ...

1. Multi-disciplinary
2. Based around a shared goal
3. Made of T-shaped people

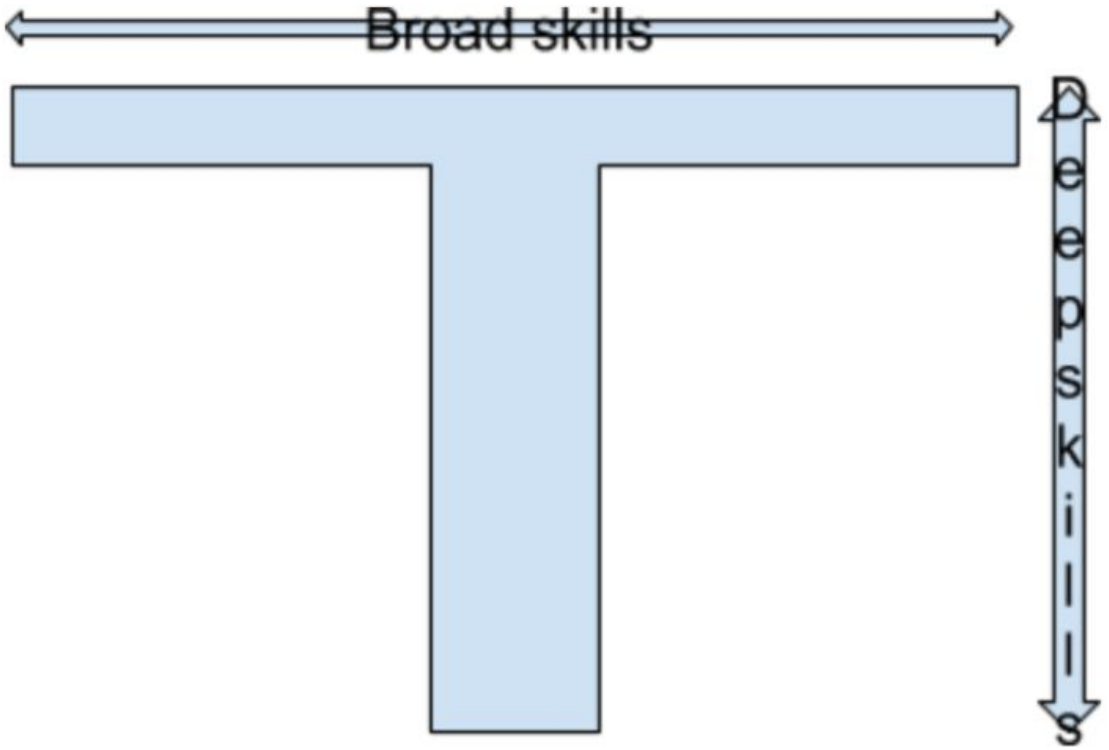
Agile teams are typically ...

1. Multi-disciplinary
2. Based around a shared goal
3. Made of T-shaped people

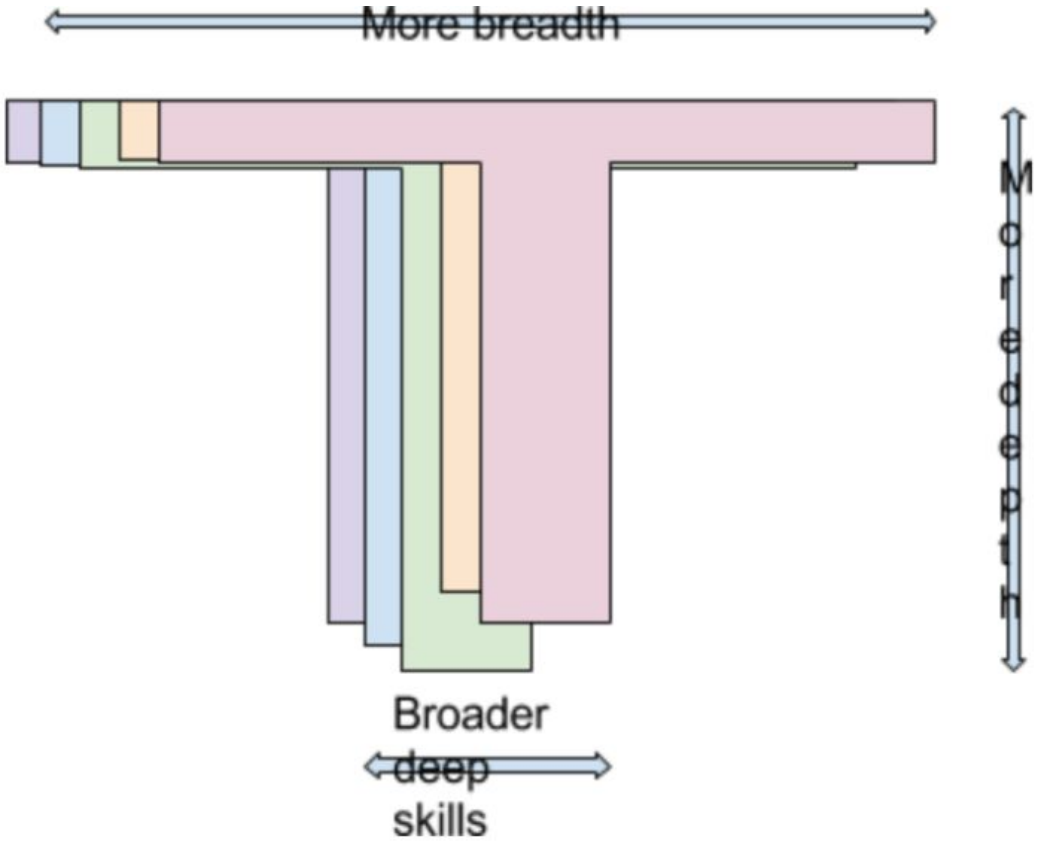
Agile teams are typically ...

1. Multi-disciplinary
2. Based around a shared goal
3. Made of T-shaped people

T-shaped person



T-shaped team



How do we make teams?

We grow them.

That means ...

Creating the right environment

Seeding the team

Allowing time to grow

Care & feeding

RIGHT
ENVIRONMENT
==
CLEAR GOAL

Why am I here?

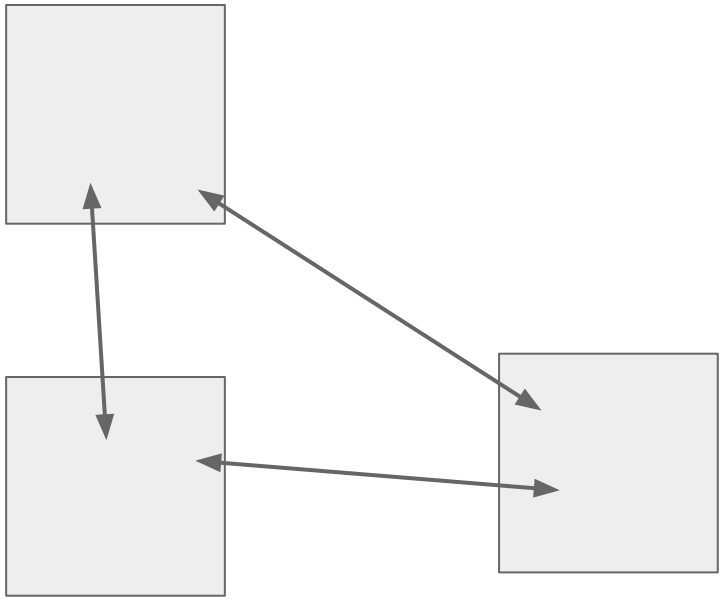
Let's talk about software architecture

"Organizations which design systems ... are constrained to produce designs which are copies of the communication structures of these organizations."

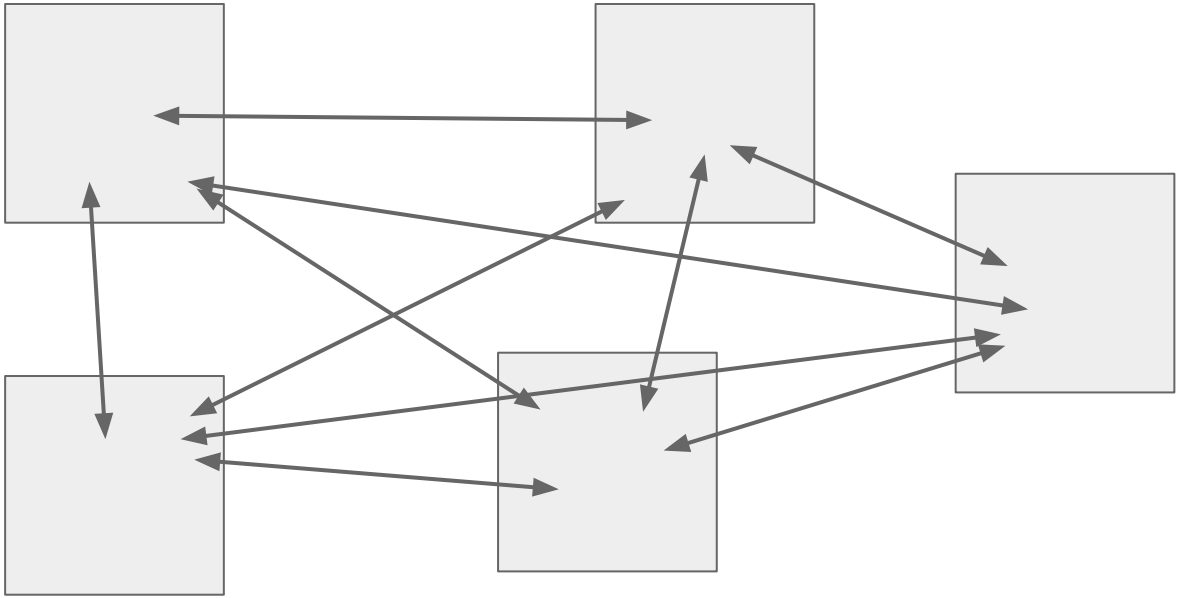
- M. Conway, 1967

How does software grow?

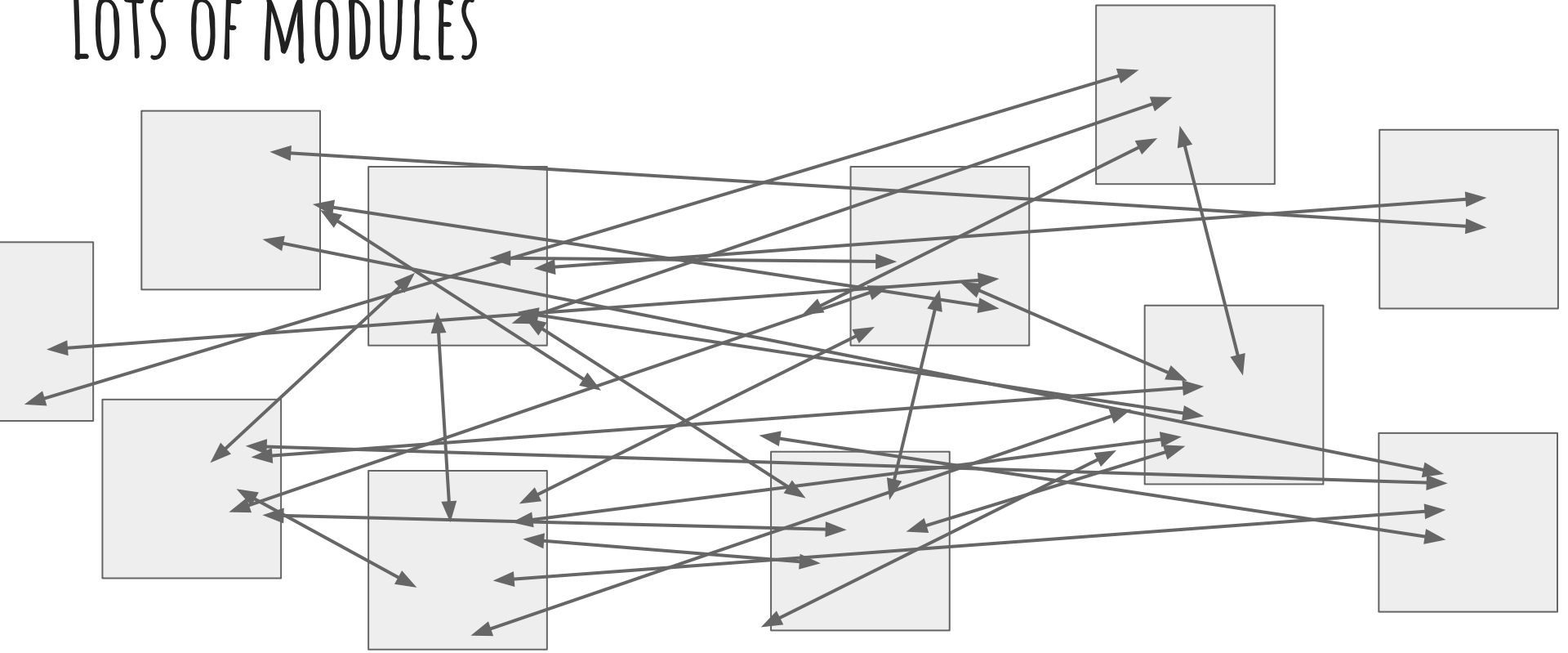
Few modules



Some modules



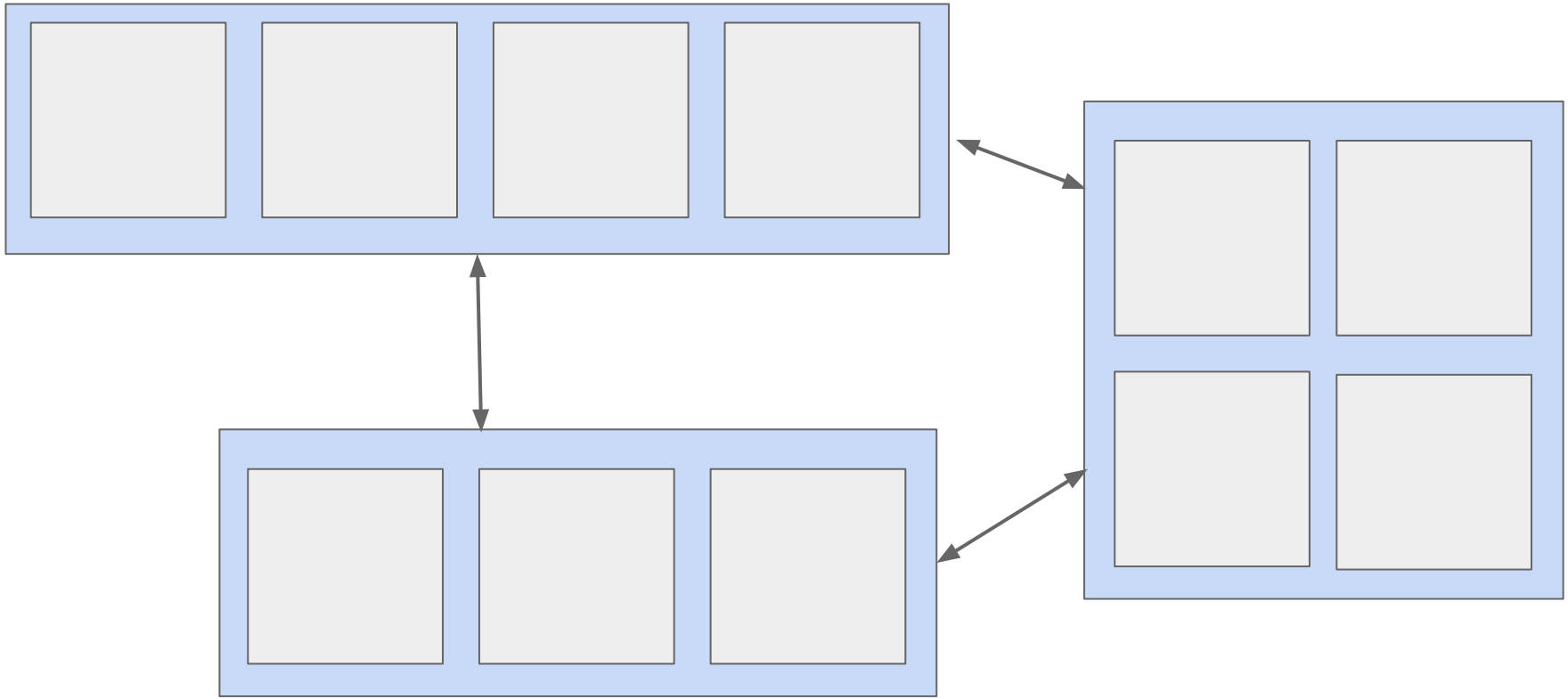
Lots of modules



How software grows

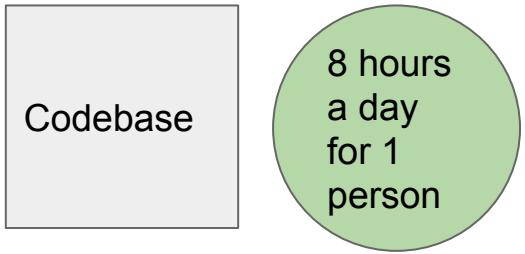
You seem like
a fun guy ...

Abstraction

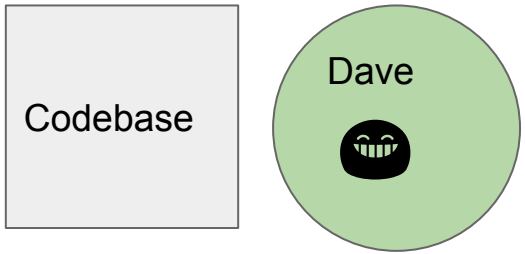


How do teams grow?
(and how is this related to a clear goal?)

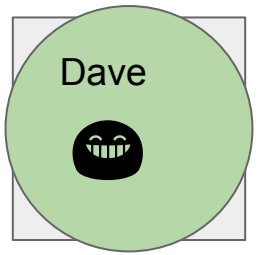
Small codebase, one person



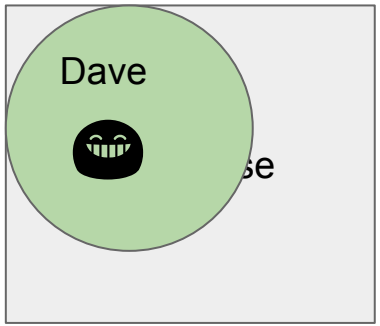
Small codebase, one person



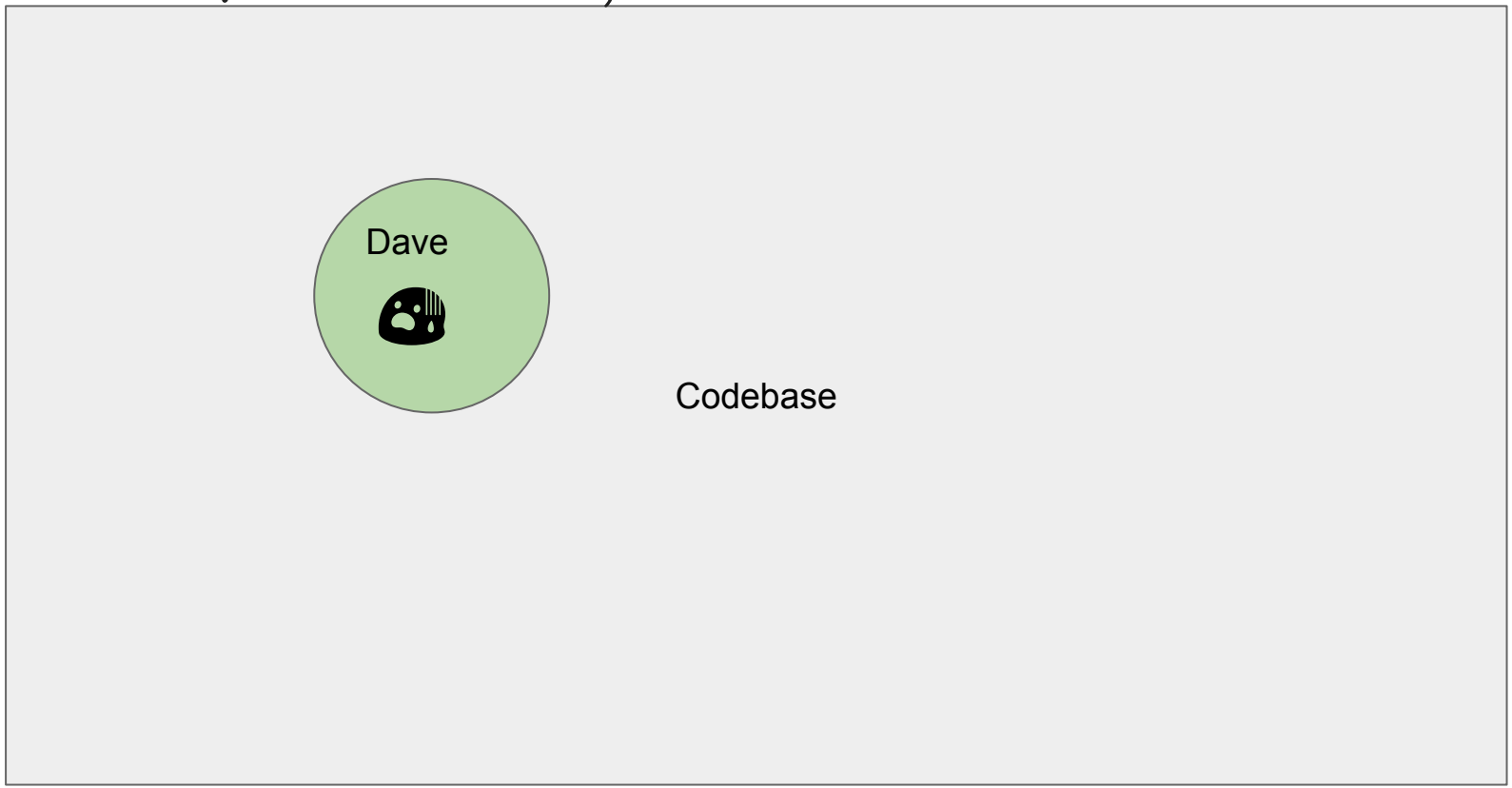
Small codebase, one person



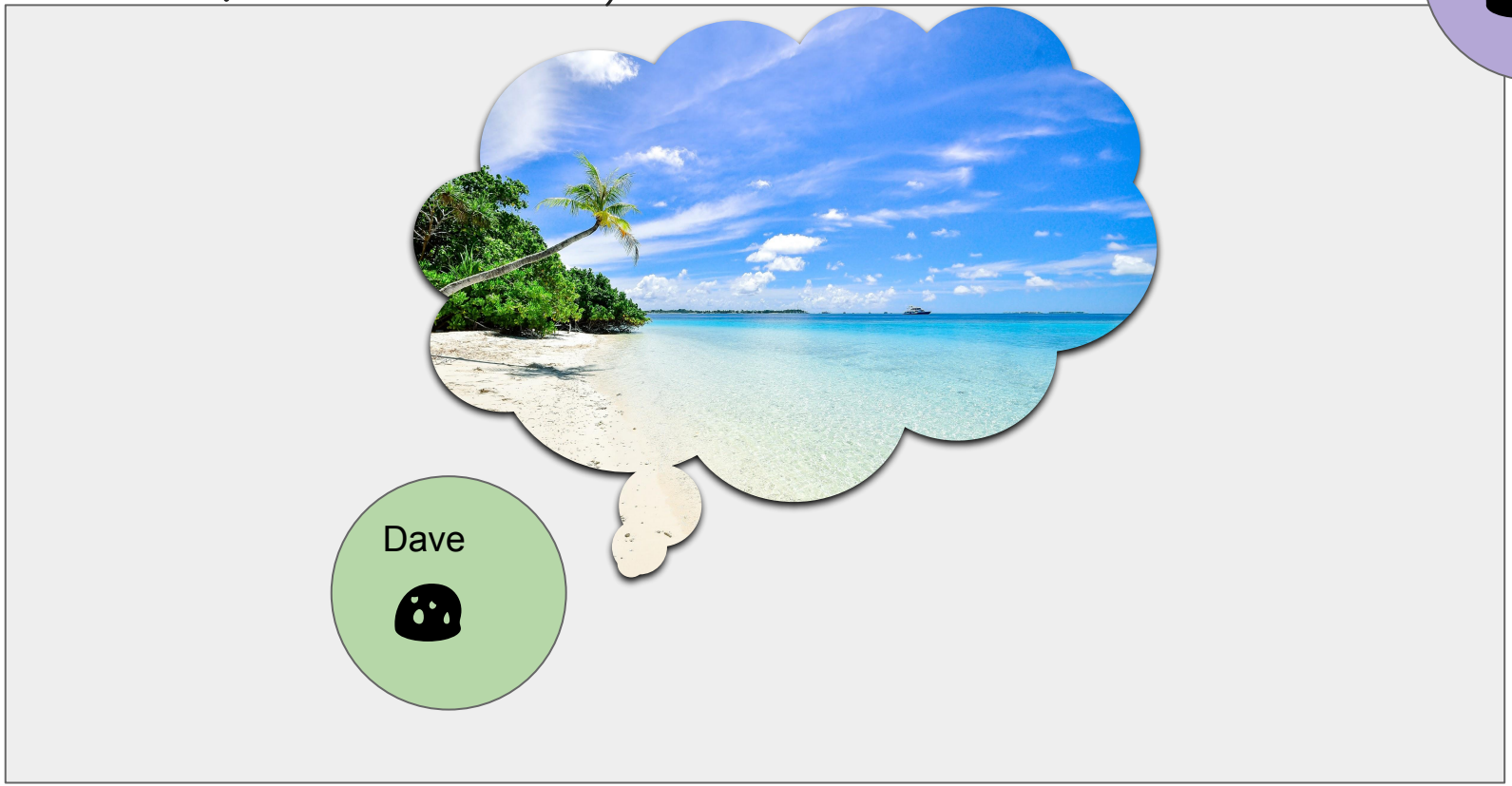
Small codebase, one person



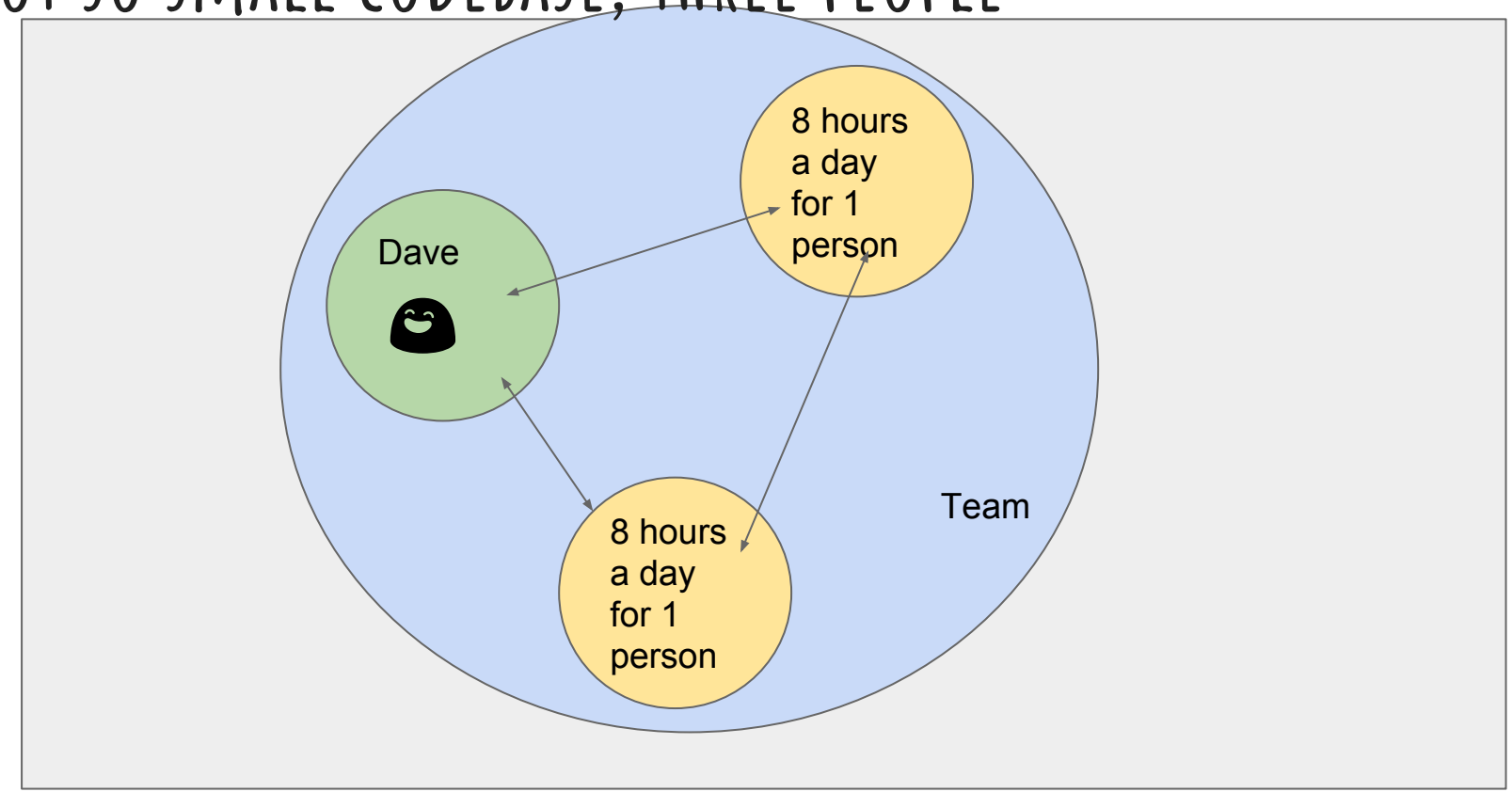
Not so small codebase, one person



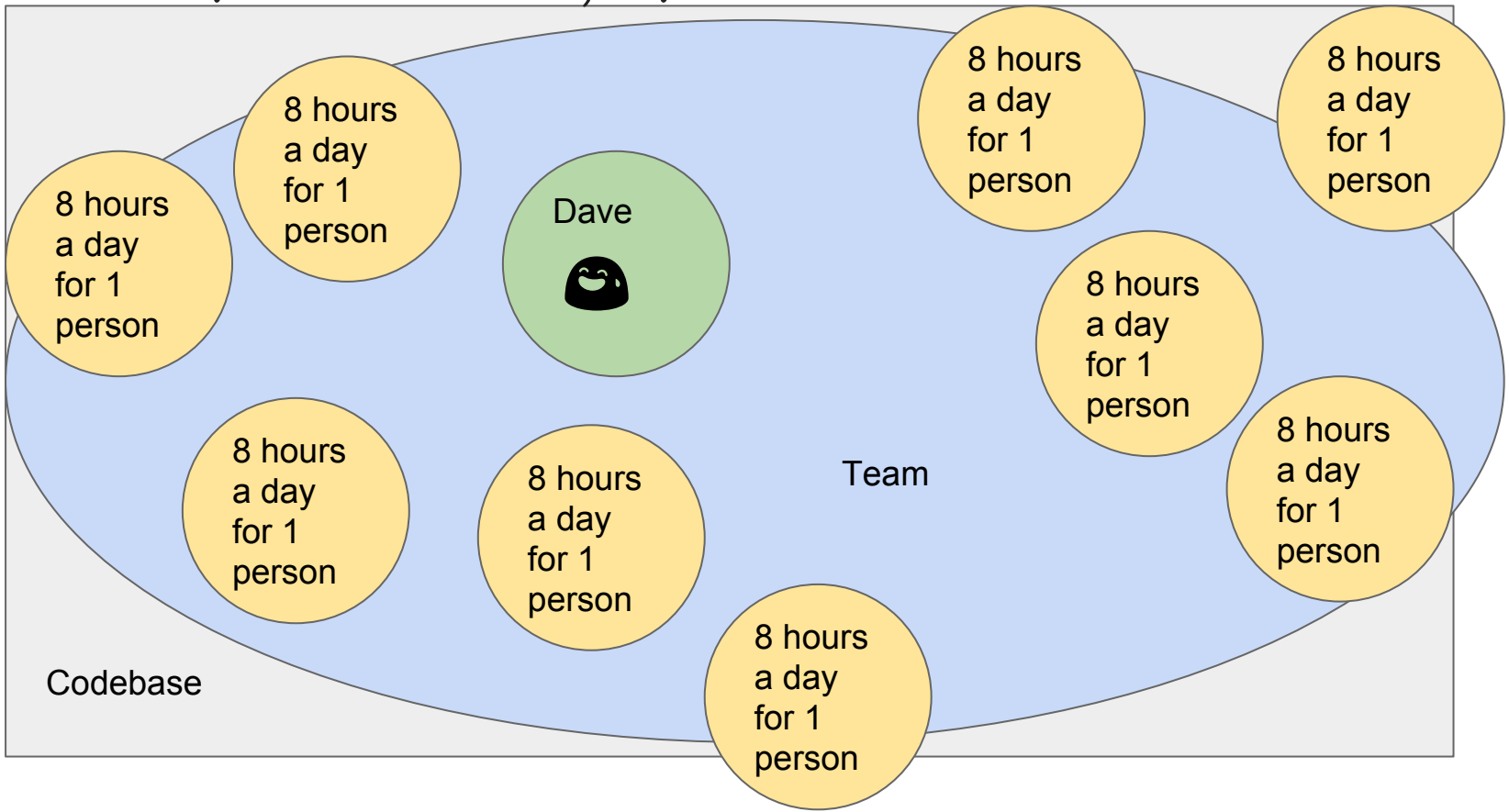
Not so small codebase, one person



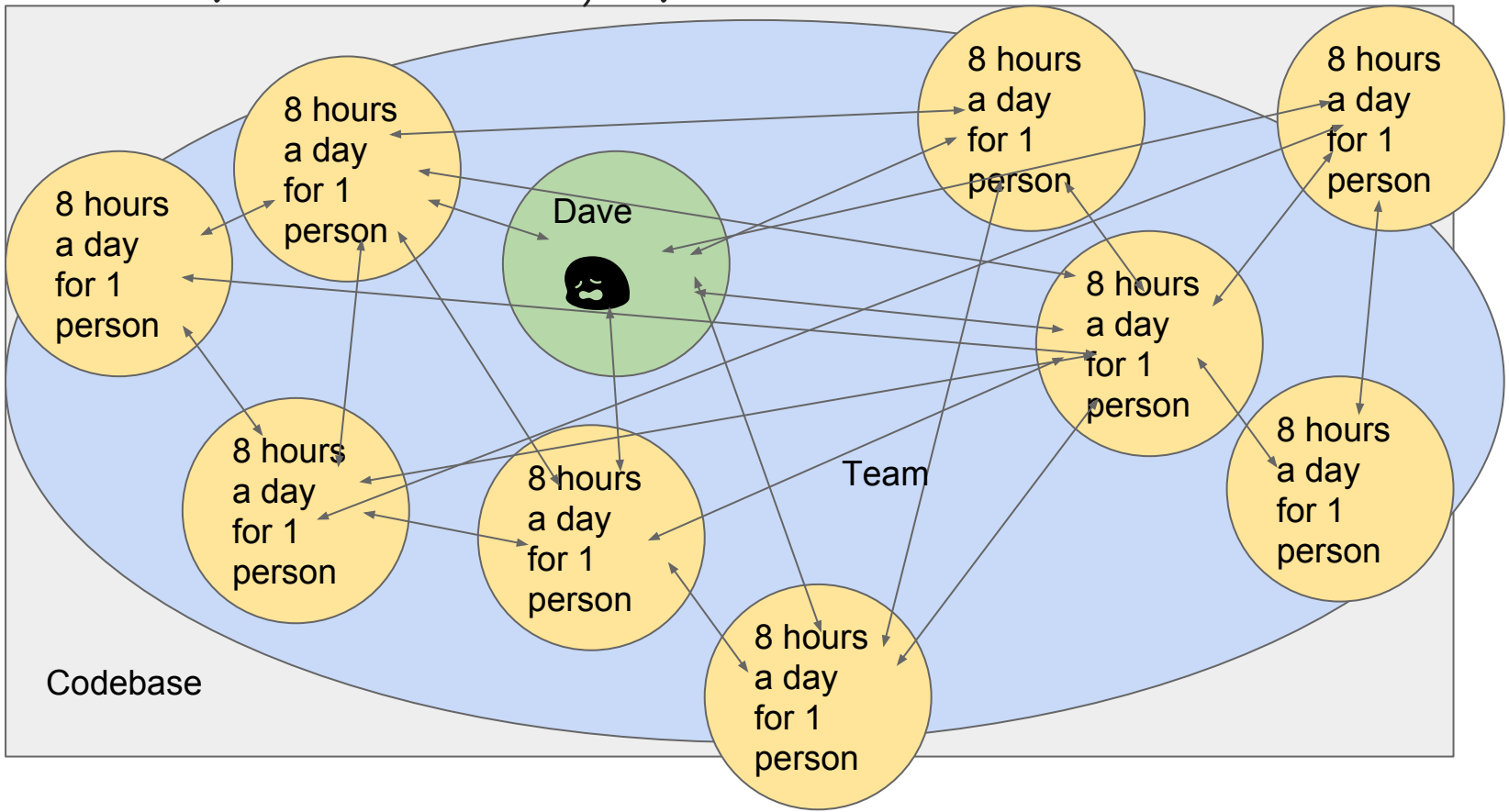
Not so small codebase, three people



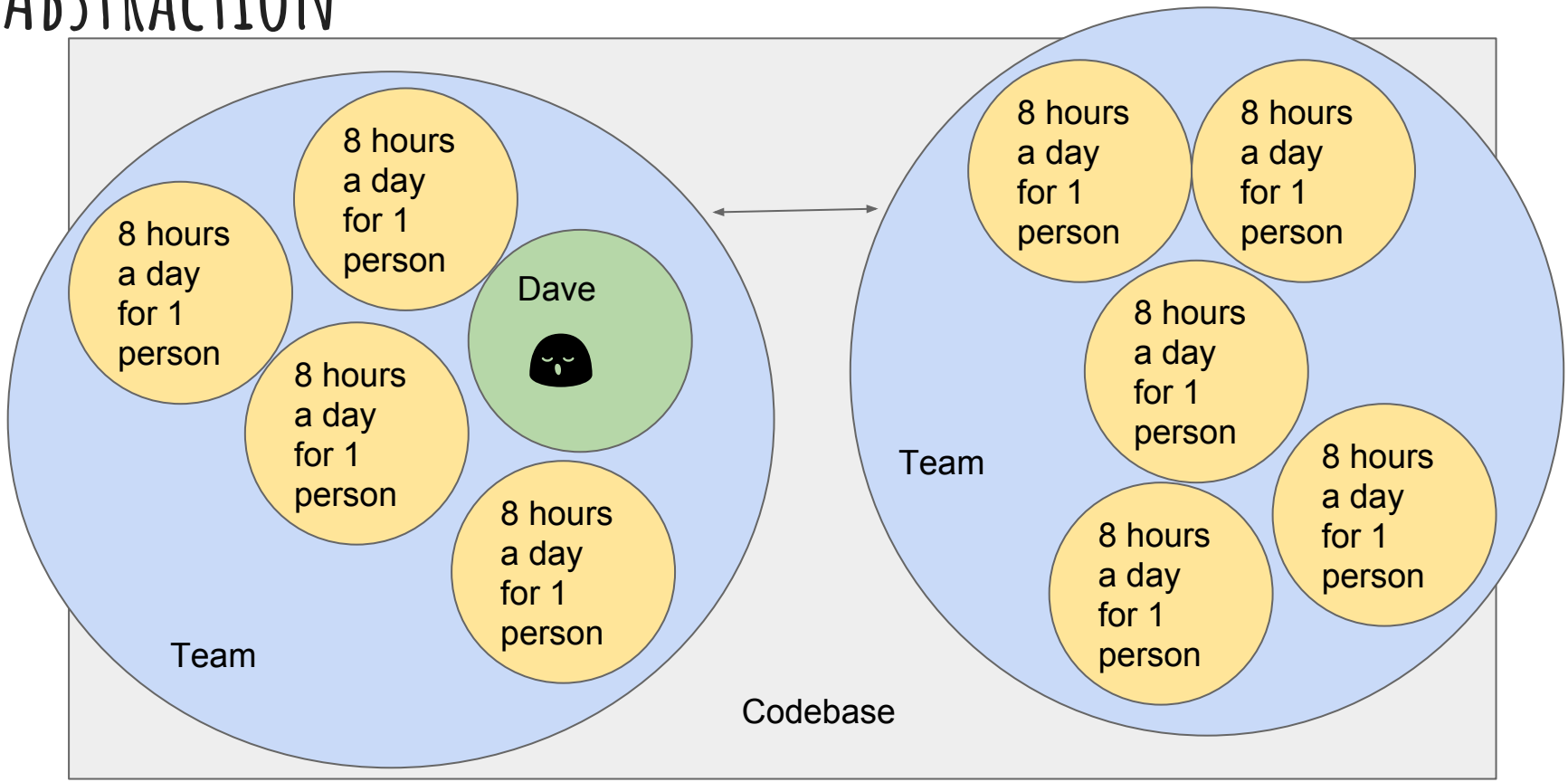
Not so small codebase, more people



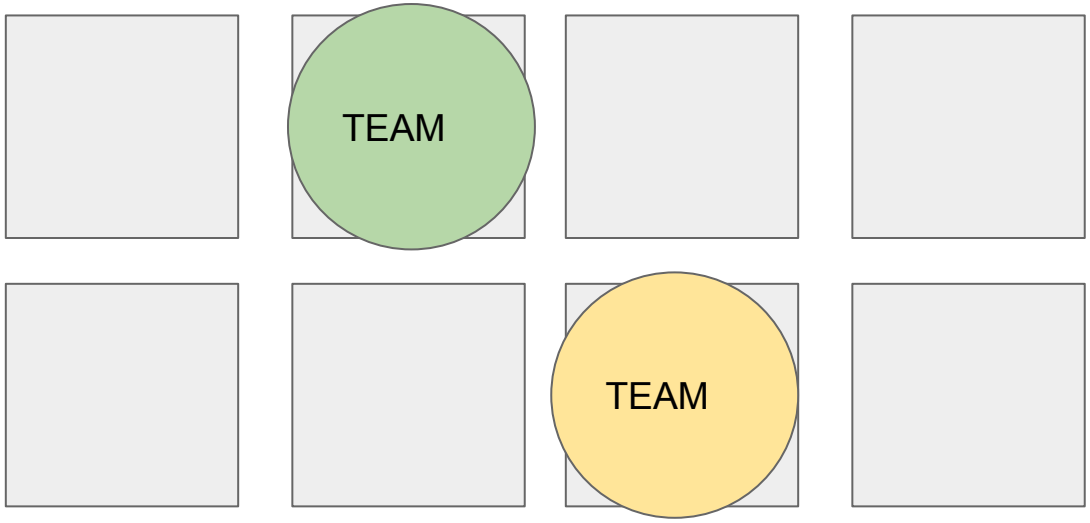
Not so small codebase, more people



Abstraction



Refactoring around teams

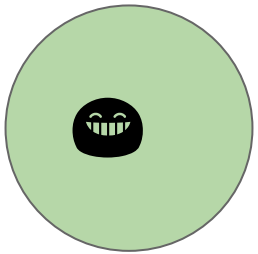


Remember Conway's Law?

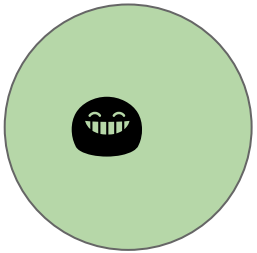
You ship your org chart.

Team ownership offers direction

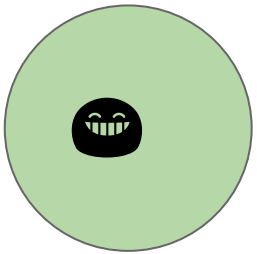
What happens if you don't have ownership?



Havant

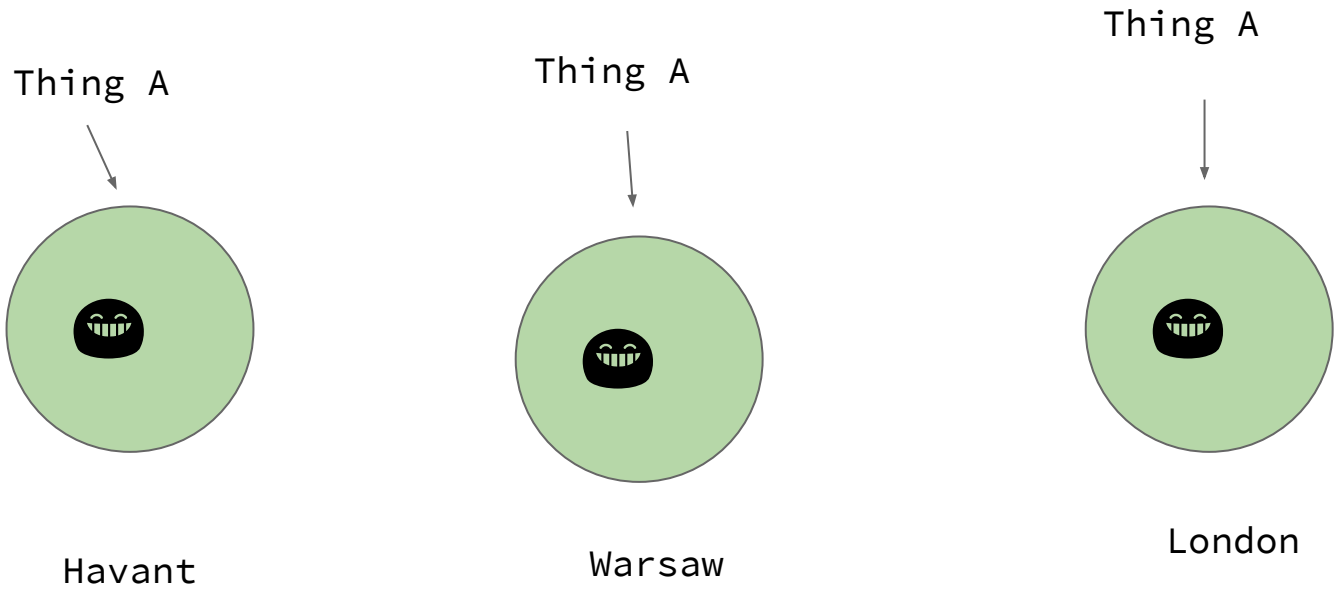


Warsaw

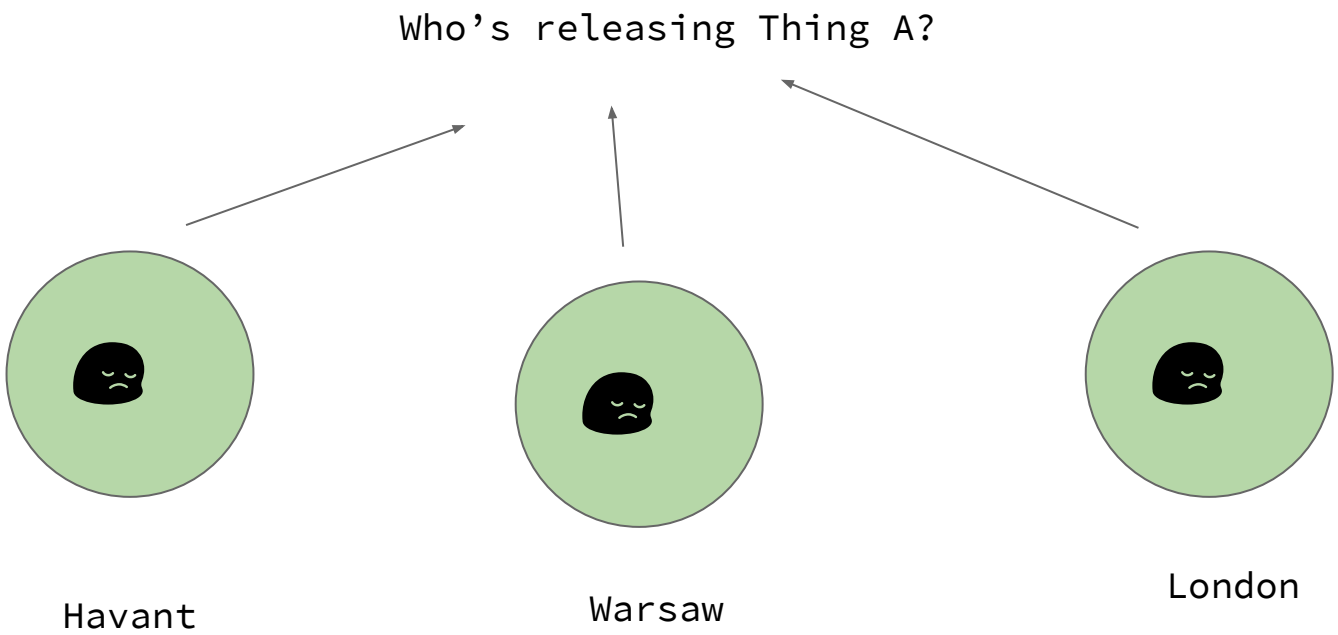


London

Easy to make a change ...



… BUT NOT EASY TO OWN IT.



Todo:

[x] Clear goal / ownership

[ ] Seed the team

[ ] Allow time to grow

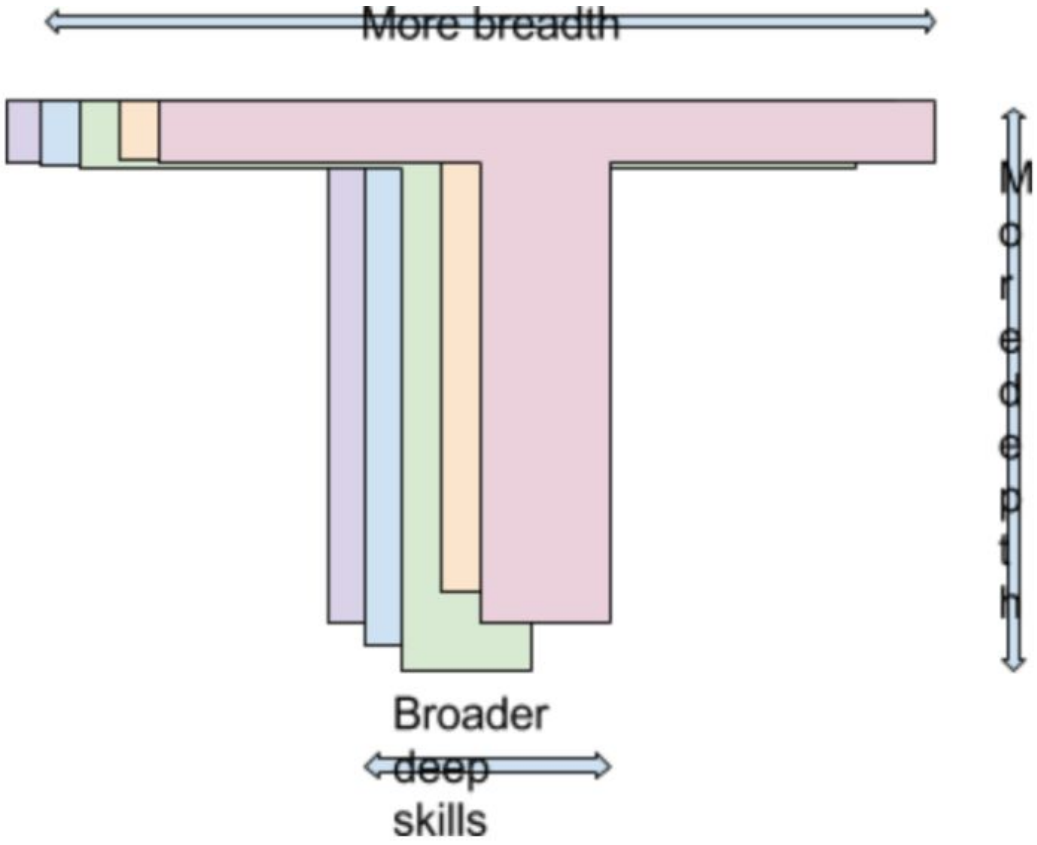
[ ] Care & feeding

Seed the team

==

Hire people!

T-shaped team



Desirable skills & behaviours

Skills I want people to have:

Commercial awareness, technical ability, good written and spoken communication, leadership

Behaviours I want people to exhibit:

Respectful, reliable, enthusiastic, proactive, pragmatic, a team player, committed, motivated, able to learn, able to give & receive feedback

Desirable skills & behaviours

Skills are taught

Commercial awareness, technical ability, good written and spoken communication, leadership

Behaviours are learned

Respectful, reliable, enthusiastic, proactive, pragmatic, a team player, committed, motivated, able to learn, able to give & receive feedback

I will take someone with good behaviours and the ability to learn over someone with excellent technical skills who's a jerk. every time.

Don't be a brilliant jerk.

Todo:

[x] Clear goal

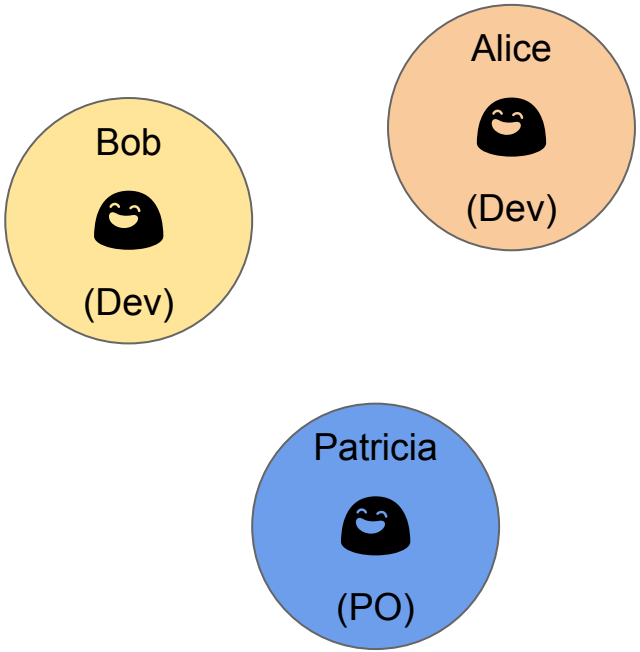
[x] Hire people!

[ ] Allow time to grow

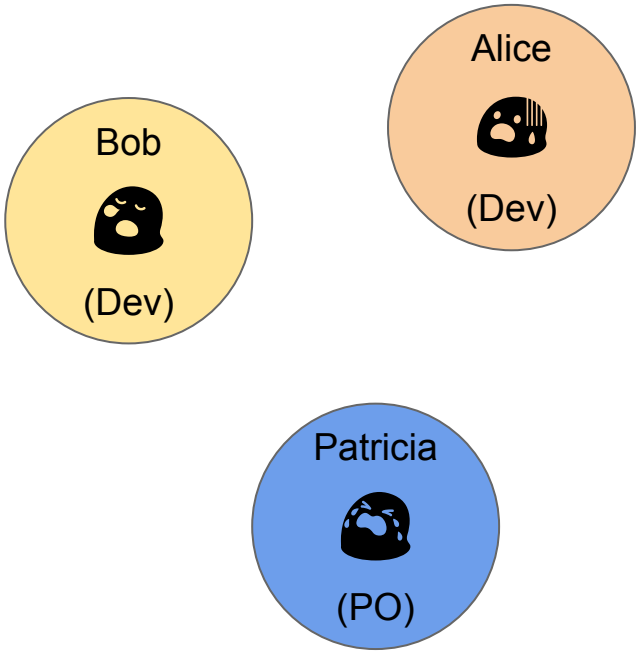
[ ] Care & feeding

Allowing time to grow (Have good team practices)

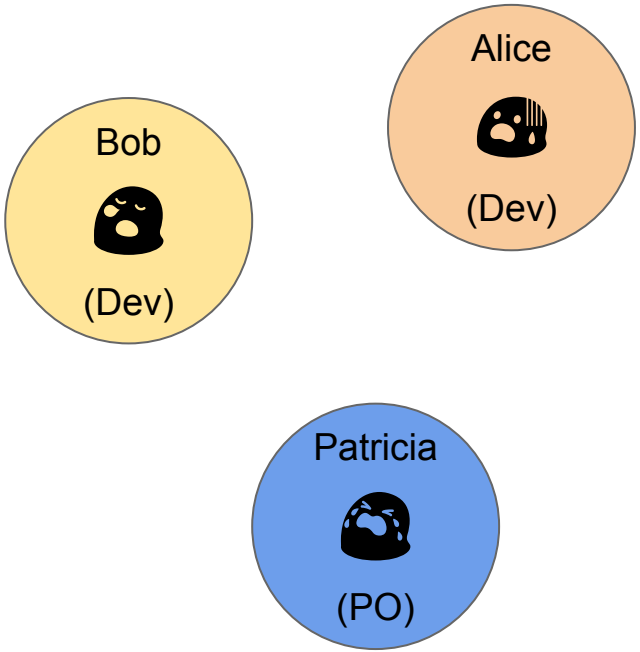
Once upon a time, in Team Endeavour



Once upon a time, in Team Endeavour

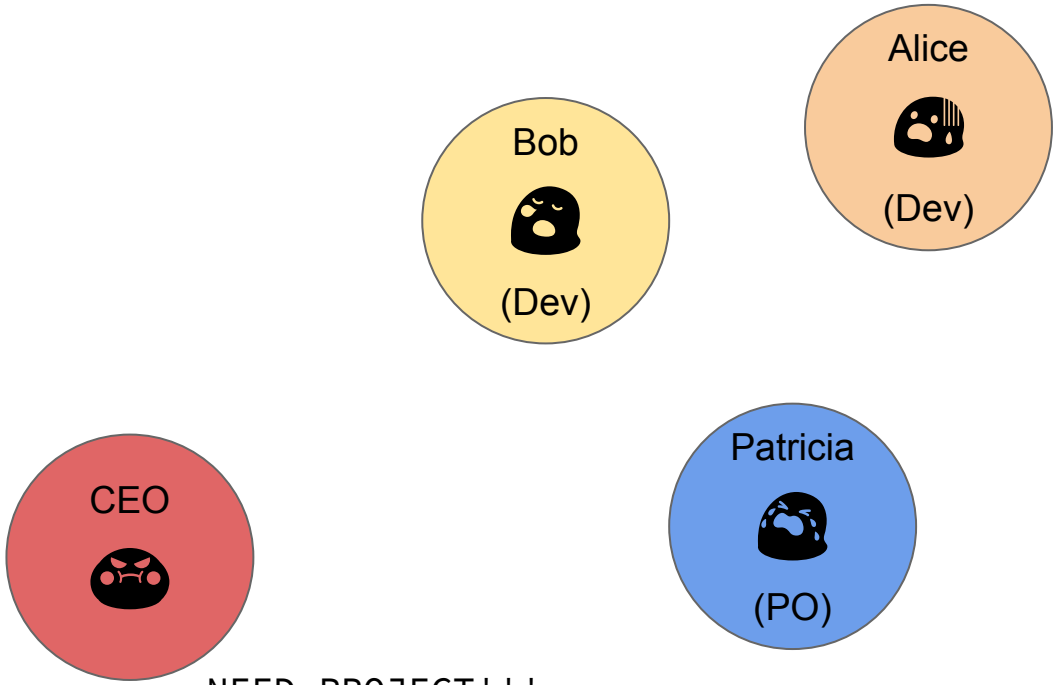


Once upon a time, in Team Endeavour



- Wanted to succeed
- Overestimated their capacity
- Burning out trying to do it all
- Generating more flex time
- Not using holiday
- Causing more stress
- Delivering slowly
- Poor meetings

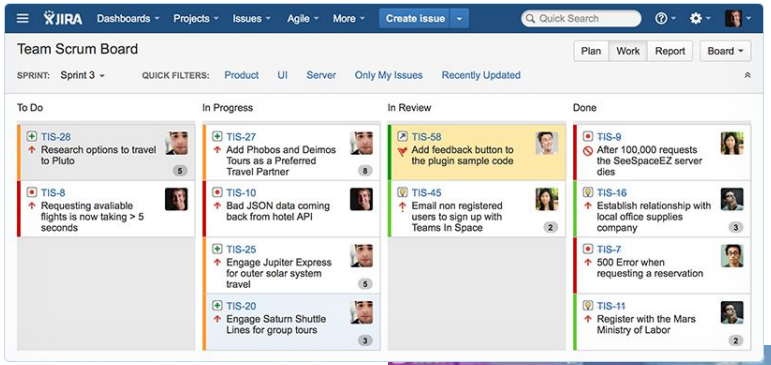
Once upon a time, in Team Endeavour



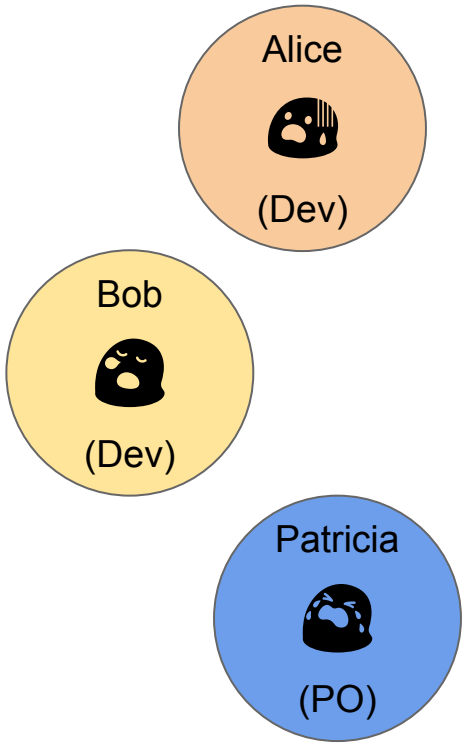
NEED PROJECT!!!

- Wanted to succeed
- Overestimated their capacity
- Burning out trying to do it all
- Generating more flex
- Not using holiday
- Causing more stress
- Delivering slowly
- Poor meetings

Keep the board clean

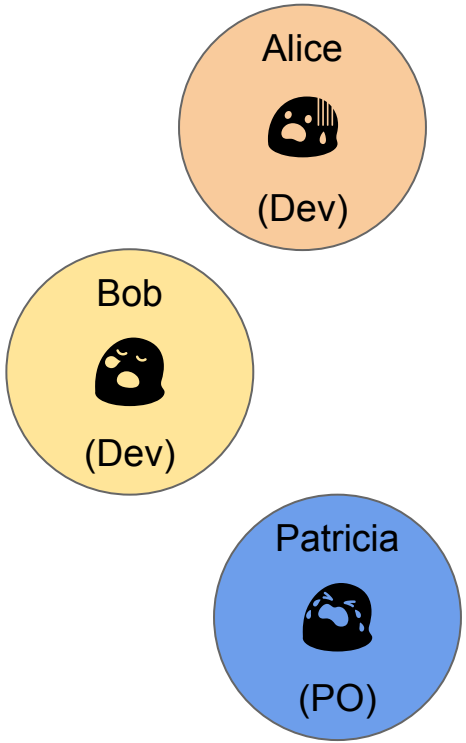


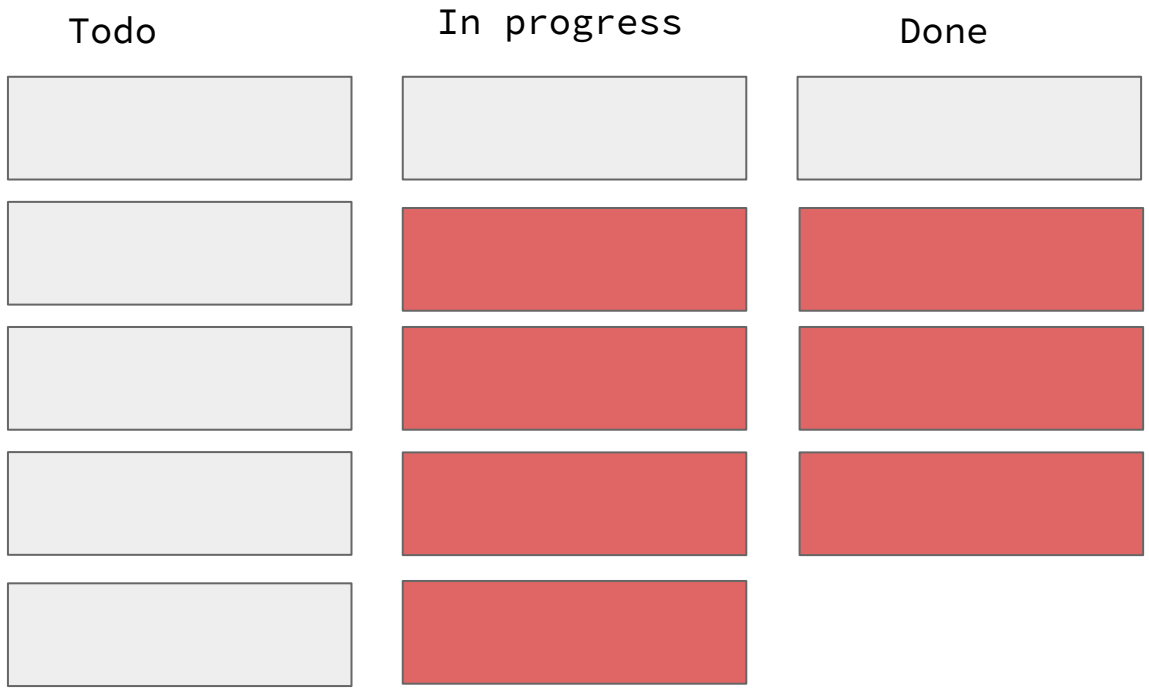
Keep the board clean



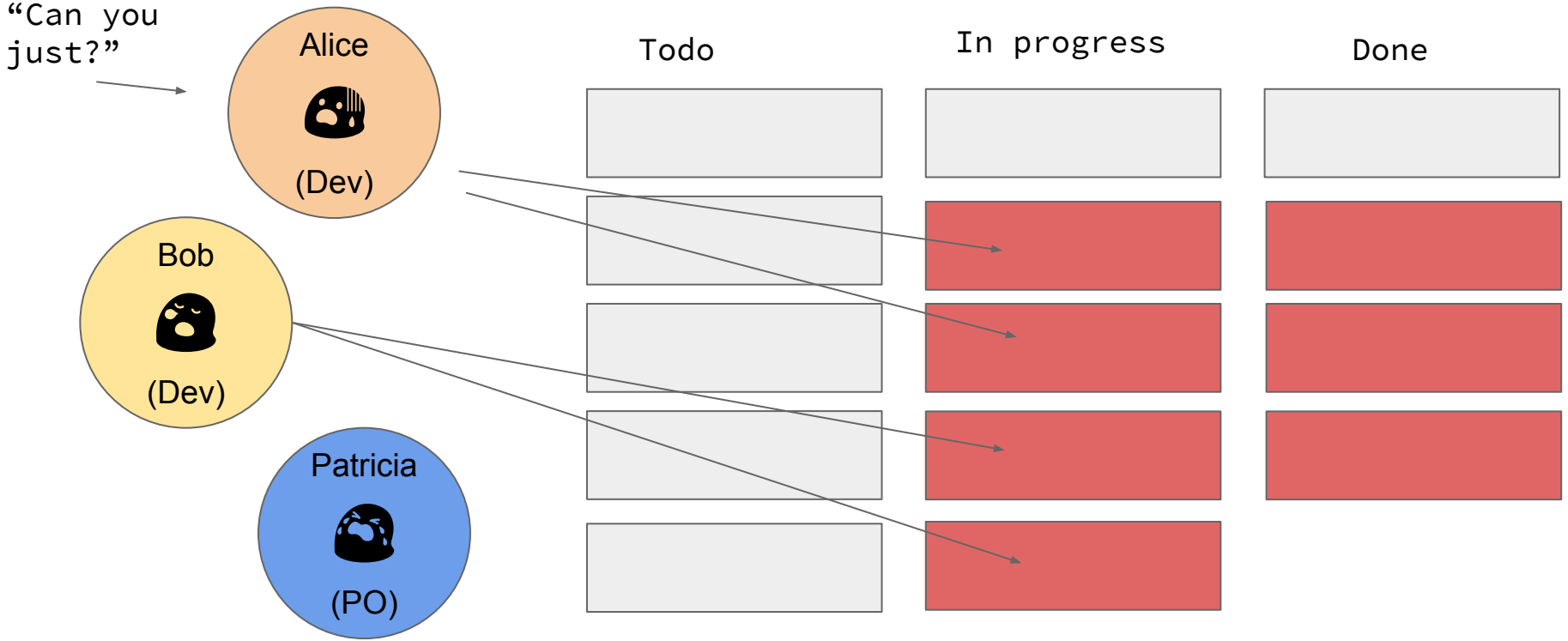
Todo	In progress	Done

Keep the board clean





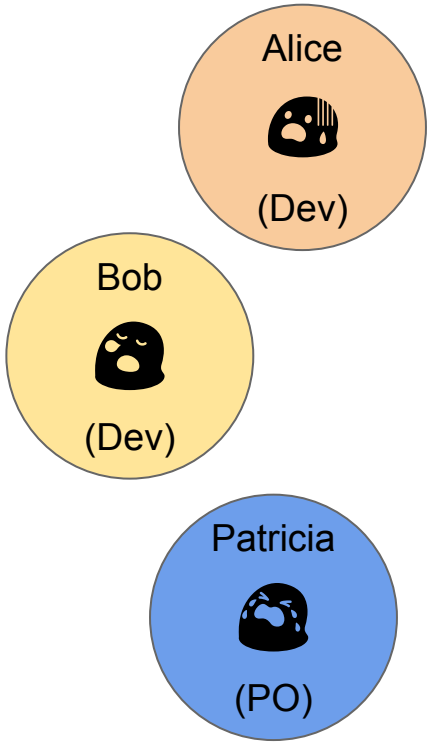
Keep the board clean

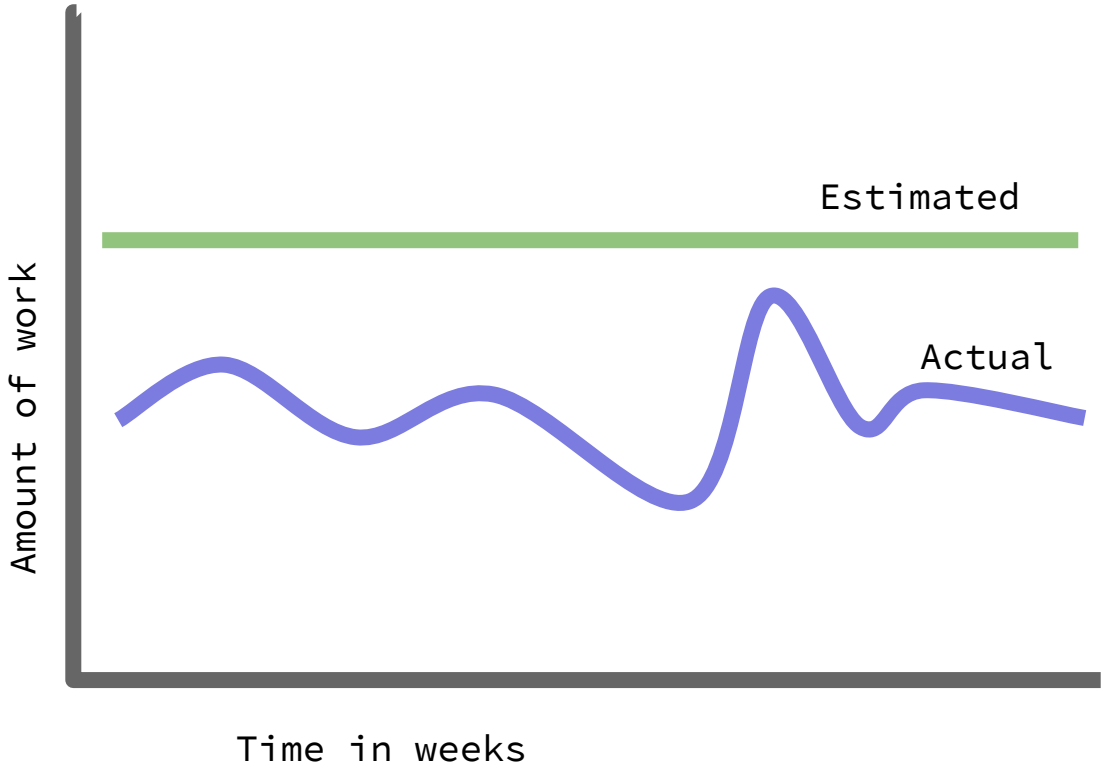


Estimate based on reality

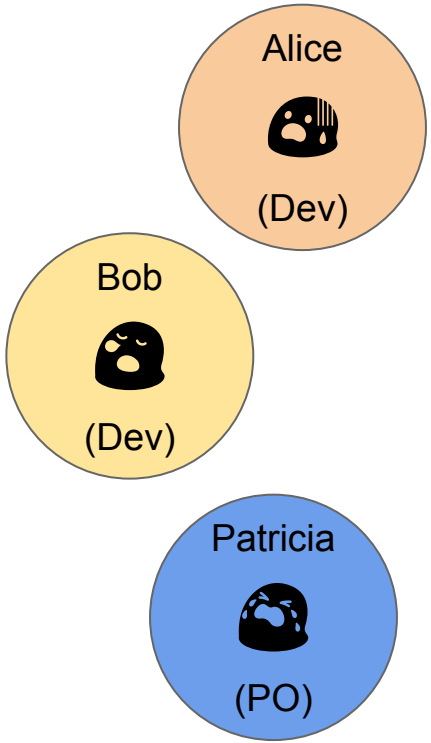
1 sprint = 25 points (based on **real historic data**)

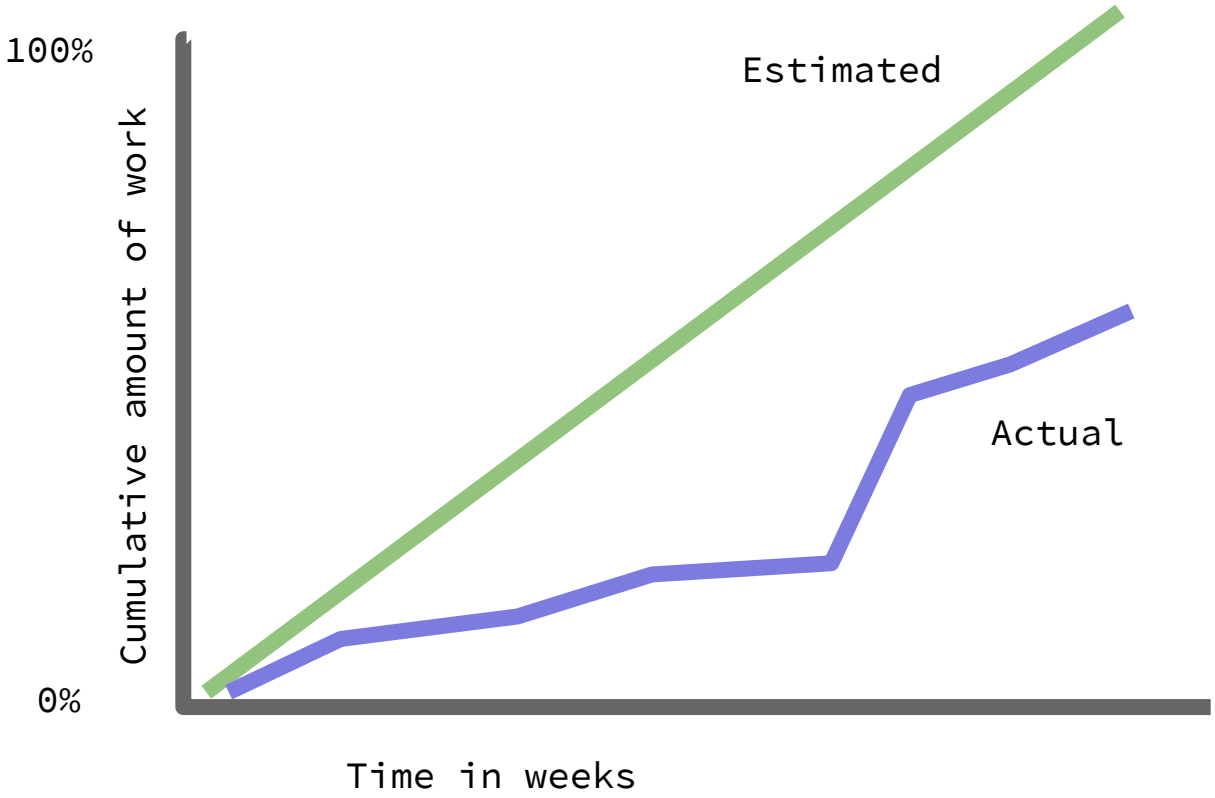
Estimate based on reality



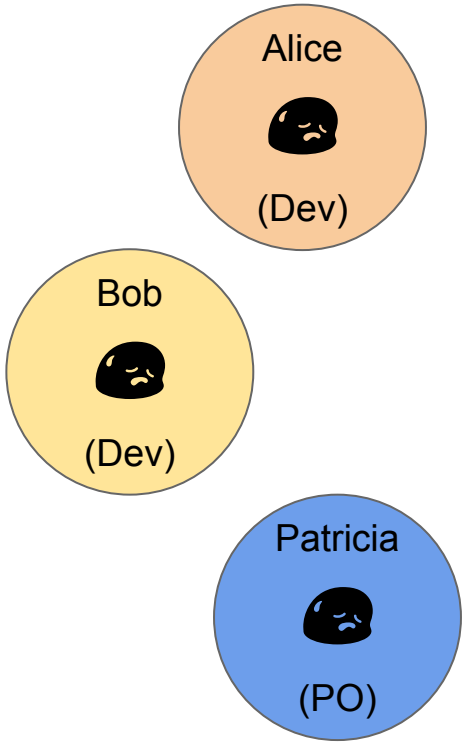


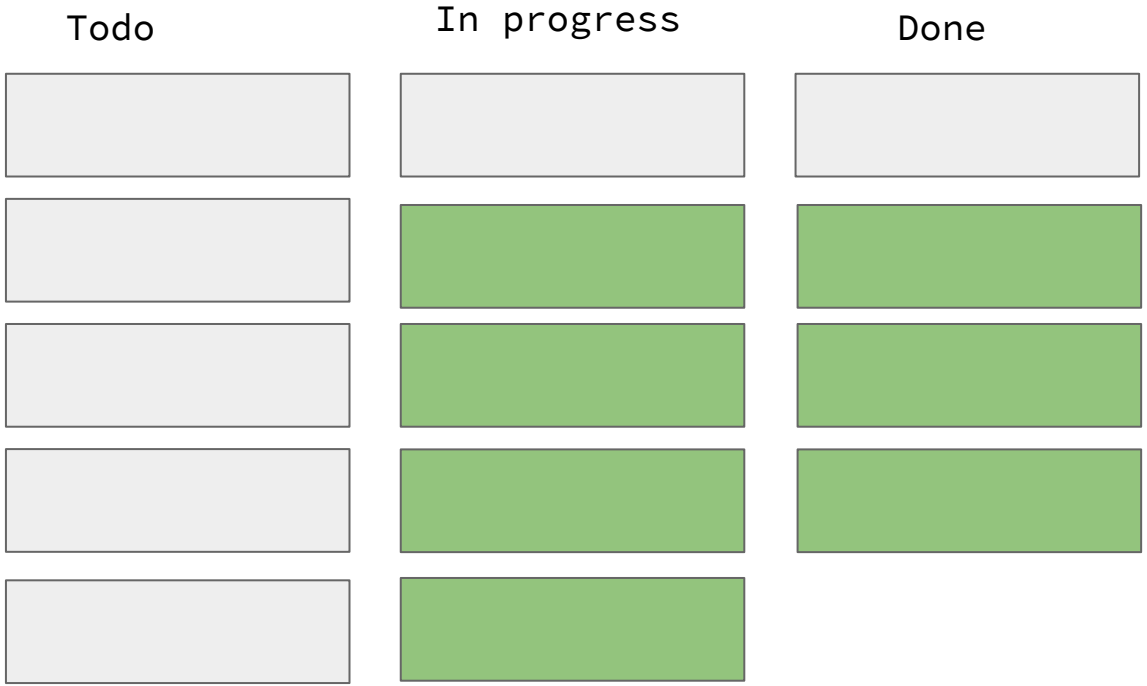
Estimate based on reality





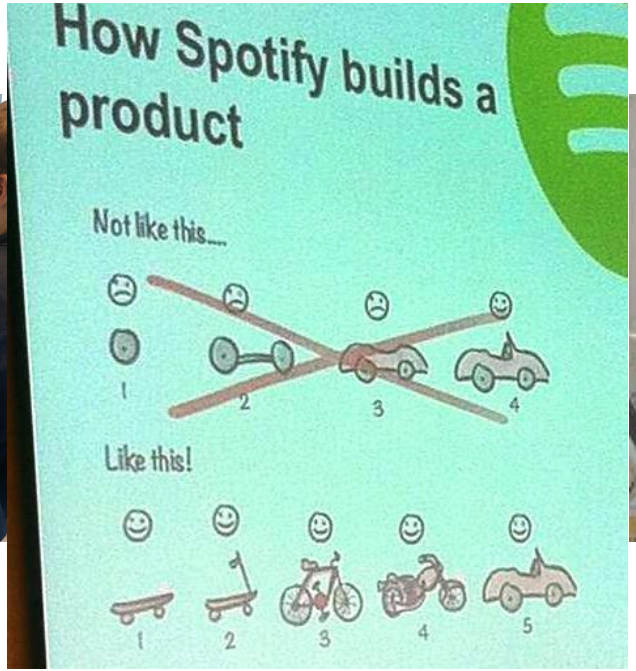
Estimate based on reality





Create focus

Sprint review

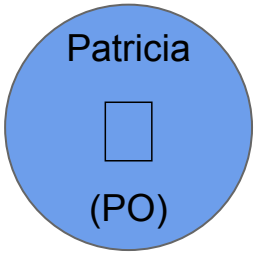


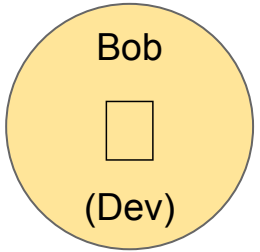
Sprint planning

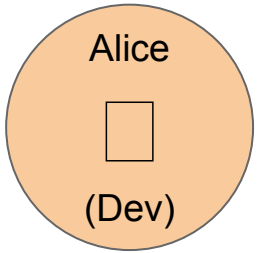
Agile build mindset

Create focus

1. Sprint review
2. Sprint planning
3. Agile mindset - always working stuff!



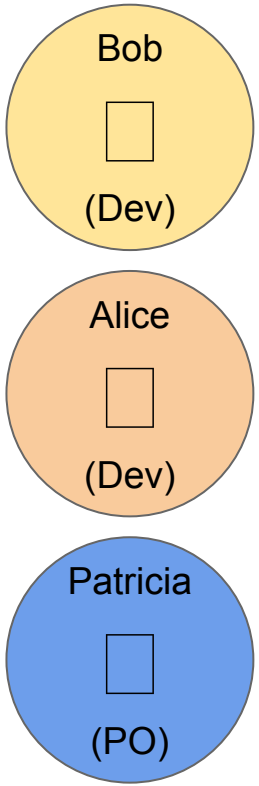




Learn from failures

Retrospective

Learn from failures



Good

- Stuff
- Things

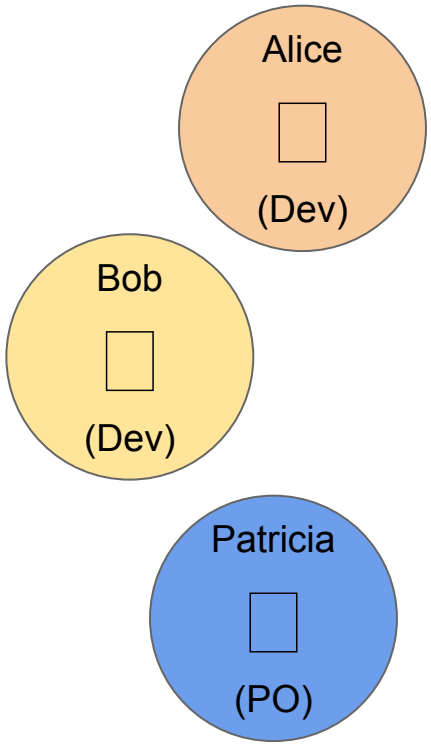
Not good

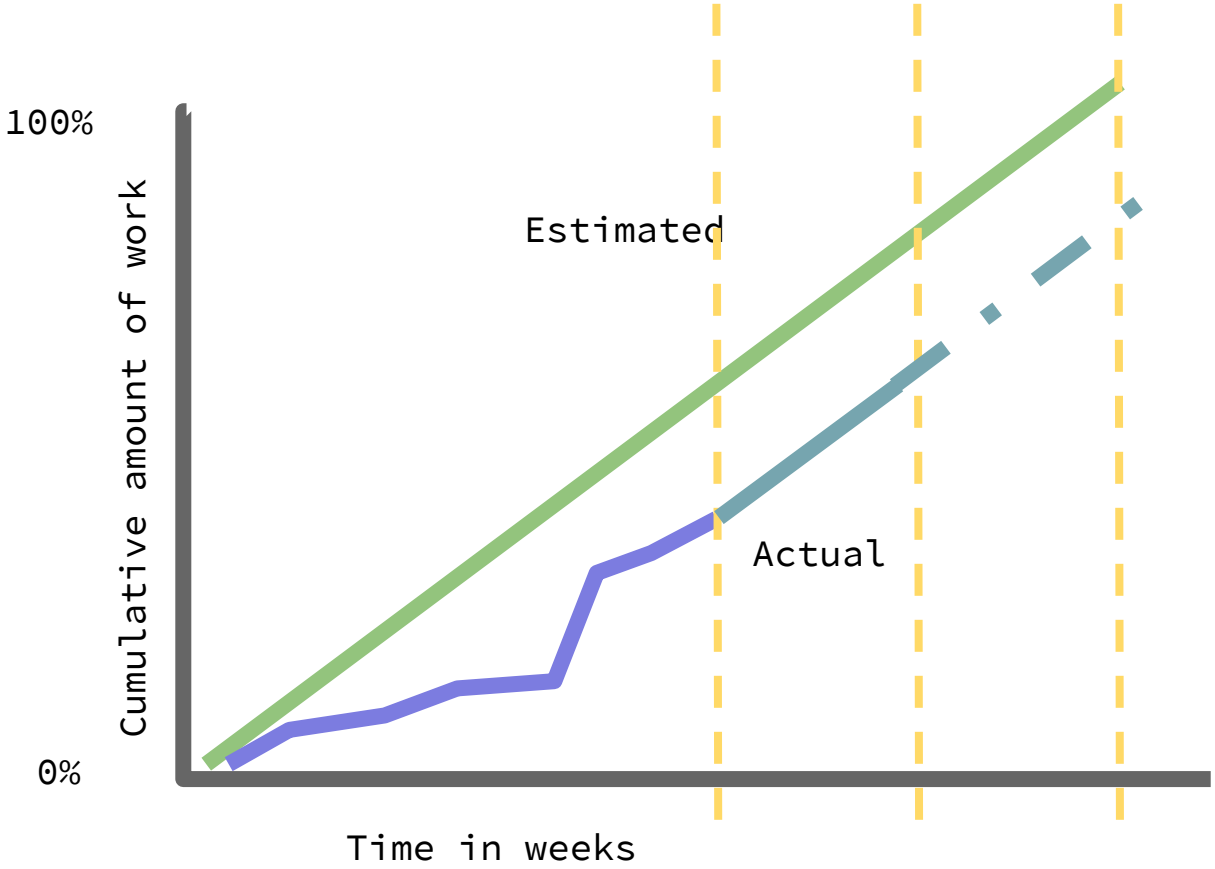
- Goals could be clearer

Check in often

Daily standup

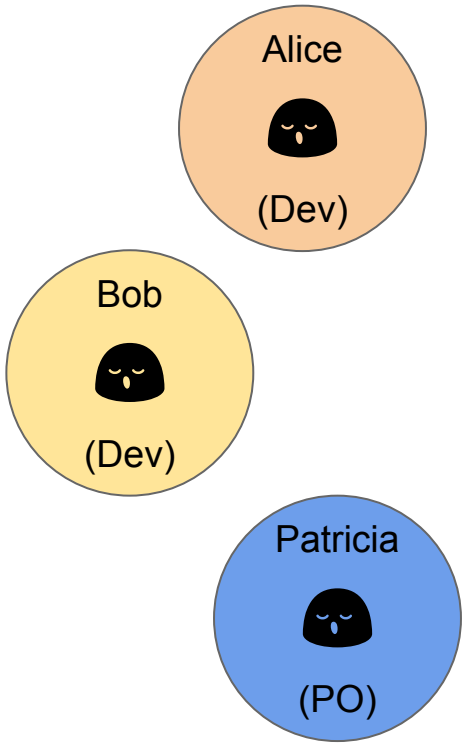
Check in often

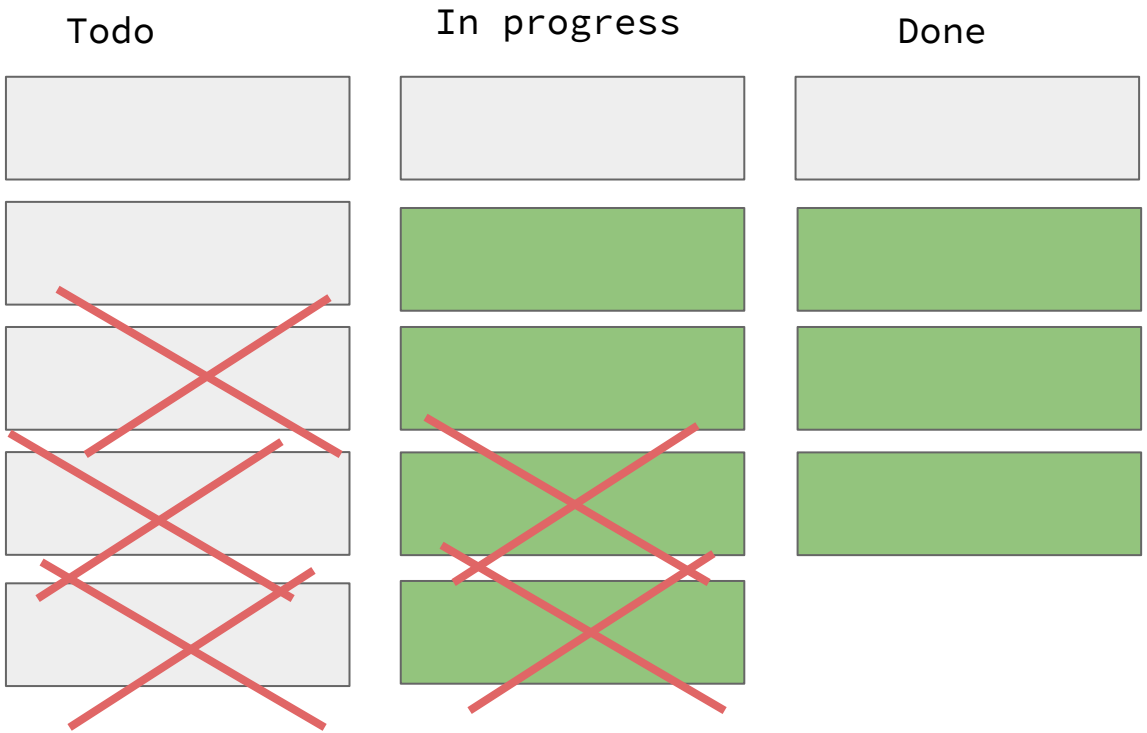




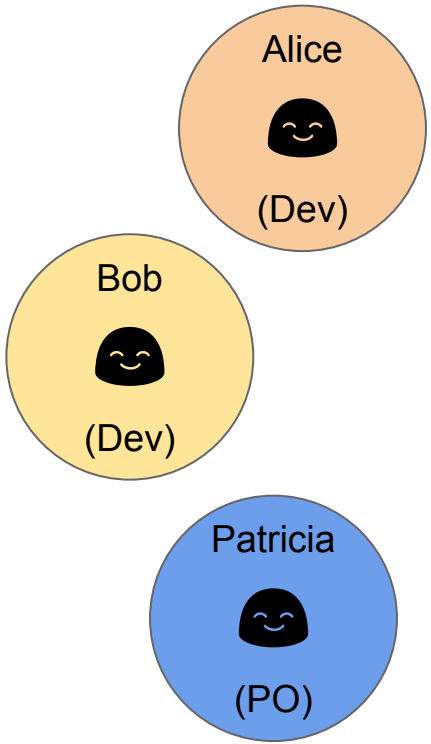
Reduce scope

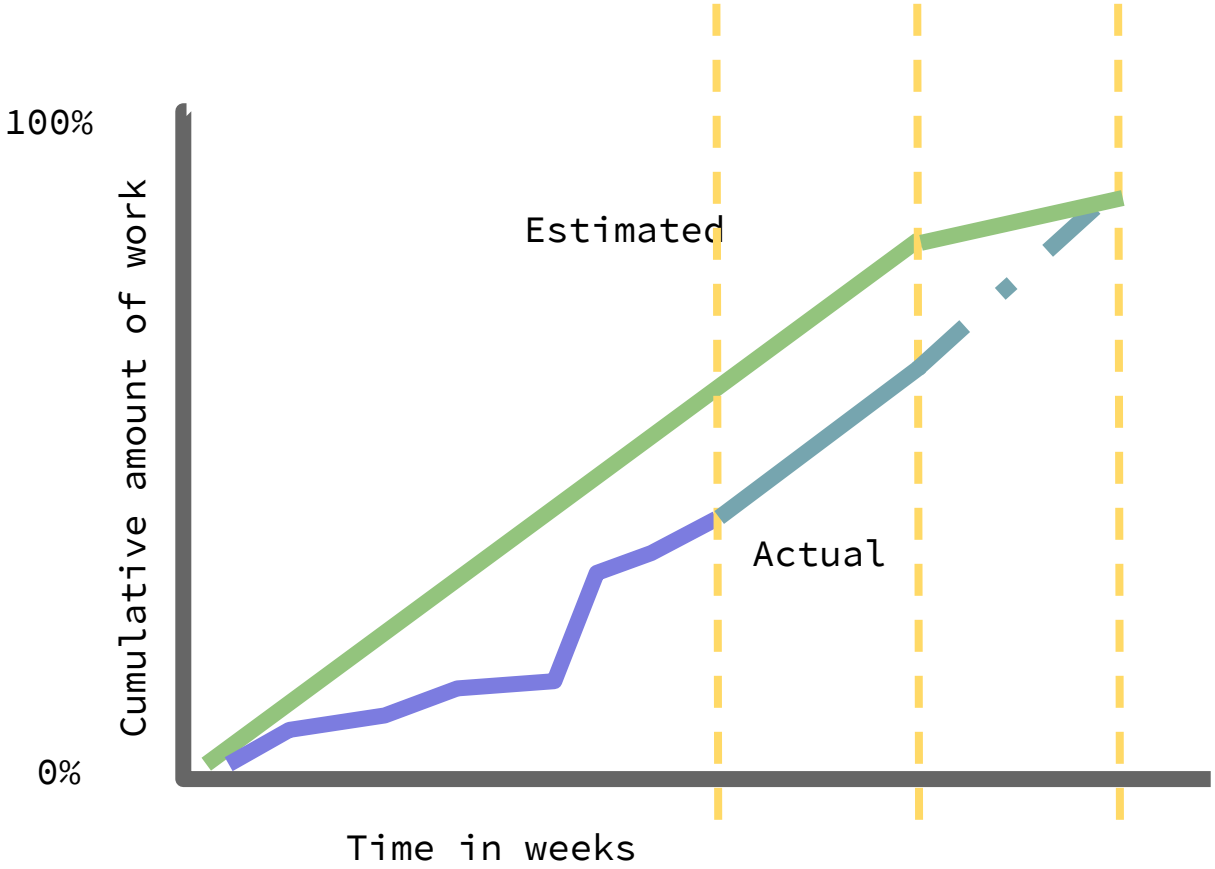
Reduce scope



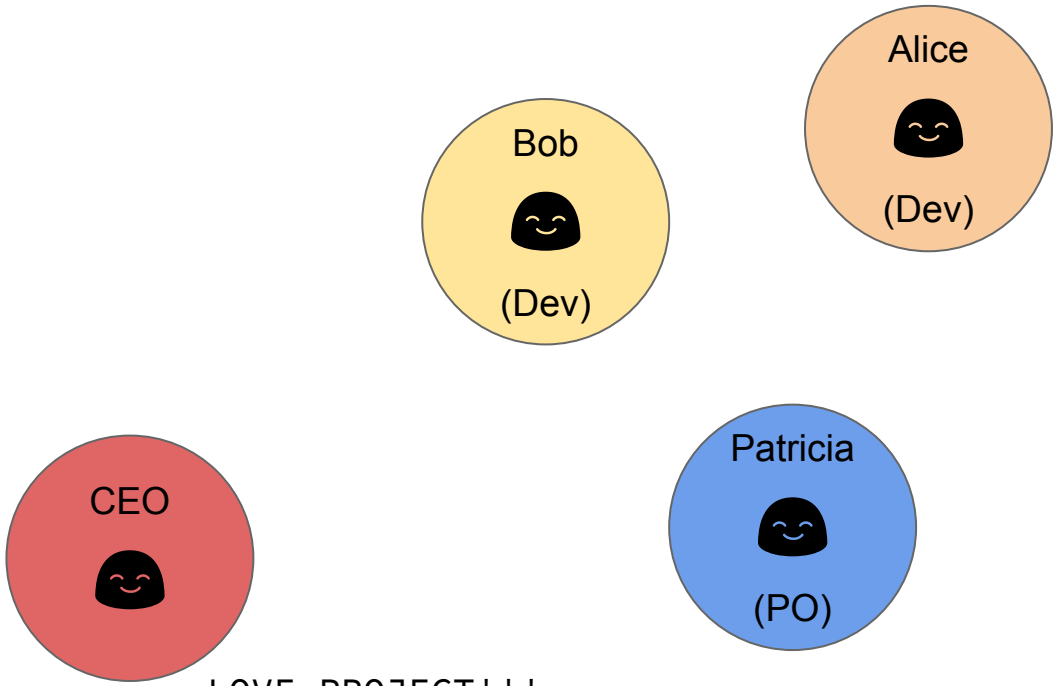


Reduce scope





Once upon a time, in Team Endeavour



- Went from failing to succeeding
- Delivered a project the business had written off
- The team were heroes of that release

Todo:

[x] Clear goal

[x] Hire people

[x] Team practices

[ ] Care & feeding

Care and feeding (The manager's stack)

Objectives

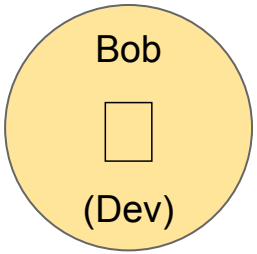
IF YOU LOOK INTO MY RÉSUMÉ
YOU'LL NOTICE I HAVE CAUGHT THE RED DOT.

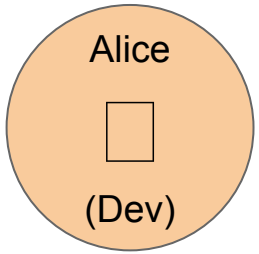
goals
i haz dem

Appraisals

1-1s

Remember Alice & Bob?





Good

- Delivering
- Making progress

Not good

- Alice doesn’t trust me

"Human Resources"

Can get complicated ...

... but is really simple.

HR is about evidence

HR audit trails

Positive: Promotions - titles, job description, salary

Neutral: Absence recording - sickness, holidays

Negative: Disciplinary - unauthorised absence / lateness, misconduct

HR audit trails

Positive: Promotions - titles, job description, salary

Neutral: Absence recording - sickness, holidays

Negative: Disciplinary - unauthorised absence / lateness, misconduct

HR audit trails

Positive: Promotions - titles, job description, salary

Neutral: Absence recording - sickness, holidays

Negative: Disciplinary - unauthorised absence / lateness, misconduct

HR audit trails

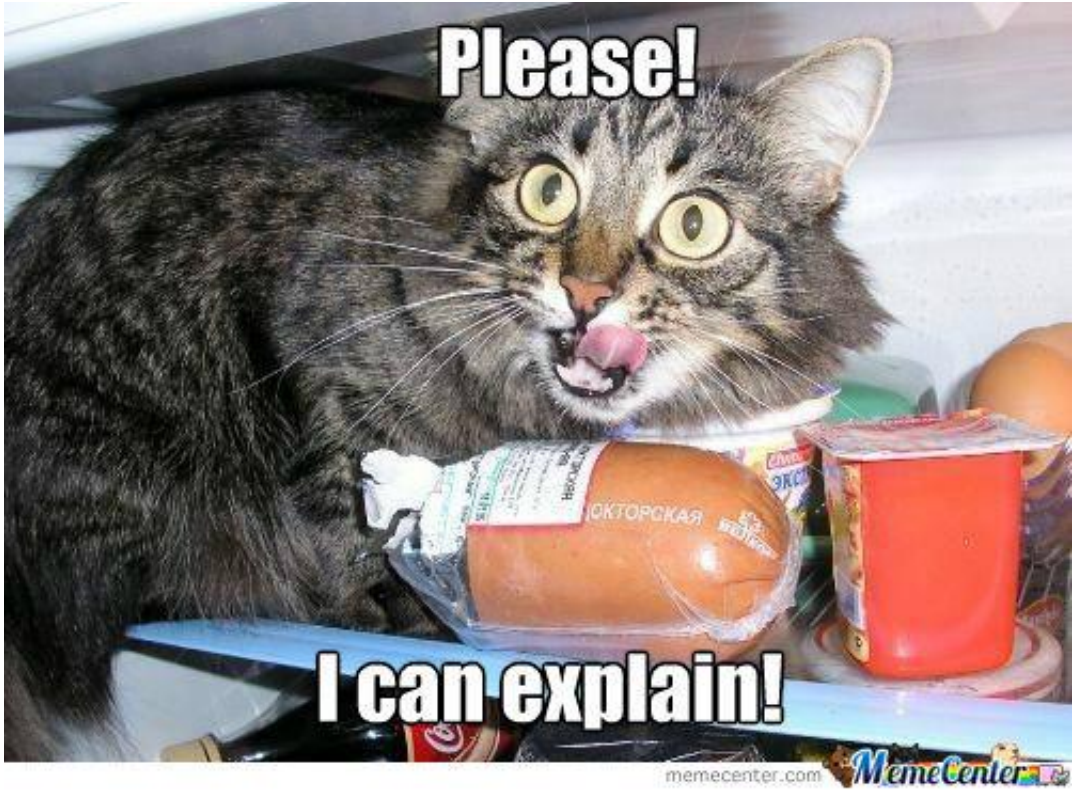
Positive: Promotions - titles, job description, salary

Neutral: Absence recording - sickness, holidays

Negative: Disciplinary - **inappropriate behaviour**

Performance management

Setting context



Todo:

[x] Clear goal

[x] Hire people

[x] Team practices

[x] Manager tools & HR processes

So are teams just processes?

Teamwork is about relationships

~~TEAM~~WORK IS ABOUT RELATIONSHIPS

Trust

Creating trust

- Do what you say you would do
- Keep people updated
- Admit when you’ve got it wrong

Be realistic

Todo:

- [x] Clear goal
- [x] Hire people
- [x] Team practices
- [x] Manager tools & HR processes
- [x] Talk about teams
- [ ] Questions?

Questions?

FRACTAL, YOU SCALE UP PEOPLE, THEN YOU SCALING UP # OF TEAMS, THEN YOU SCALE UP GROUPS OF TEAMS ... IT'S ALL THE SAME PROBLEM, JUST DIFFERENT

Credits

T-shaped people diagrams
https://www.adventureswithagile.com/2017/07/12/heres-thing-t-shaped-people/

Lots of google image searches