

# Org Topologies from the trenches

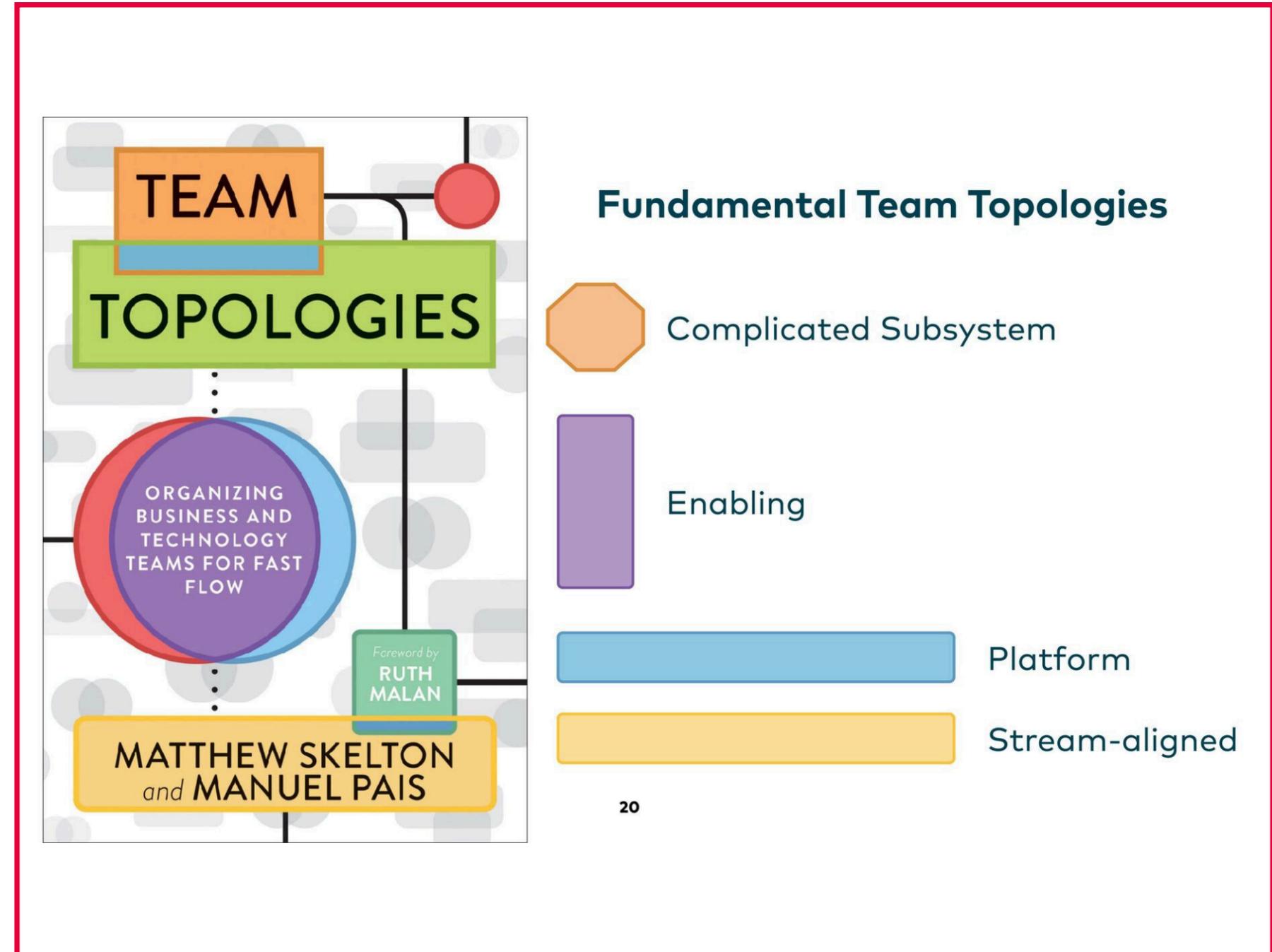
**Michael Voorhaen** - Product Manager @ Prottime

Flowcon 2026

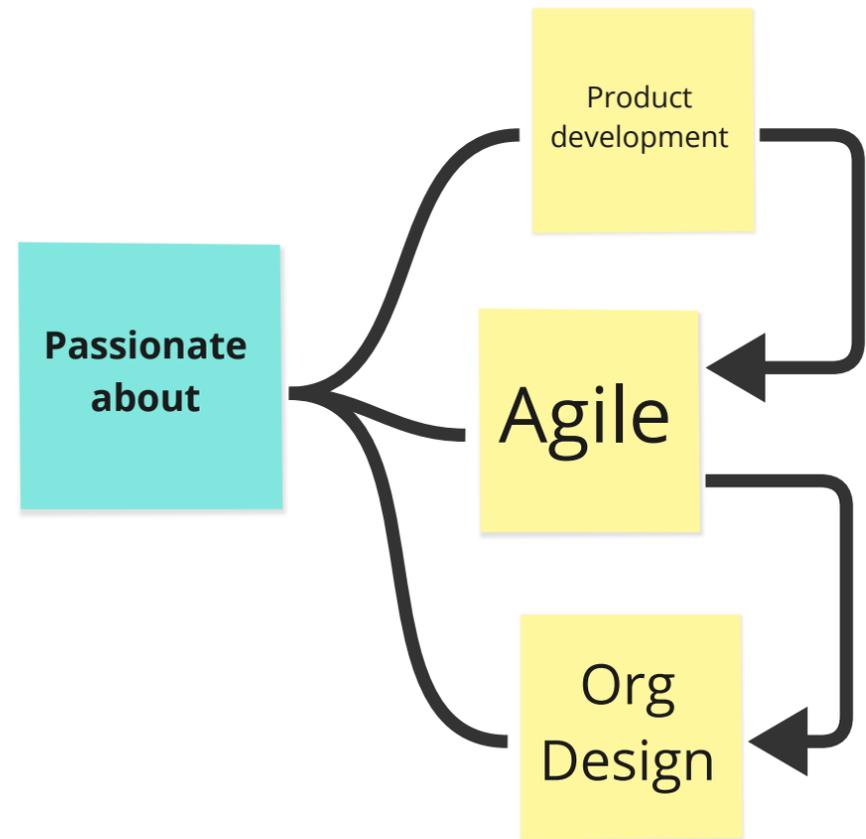
# THIS



# NOT THIS



# About Me



**Product Manager**  
Planning  
@Protime



**Product Owner**  
@Strobbo

Extensions of the Optimized Link-State Routing Protocol for Quality-of-Service and Multi-Level Routing

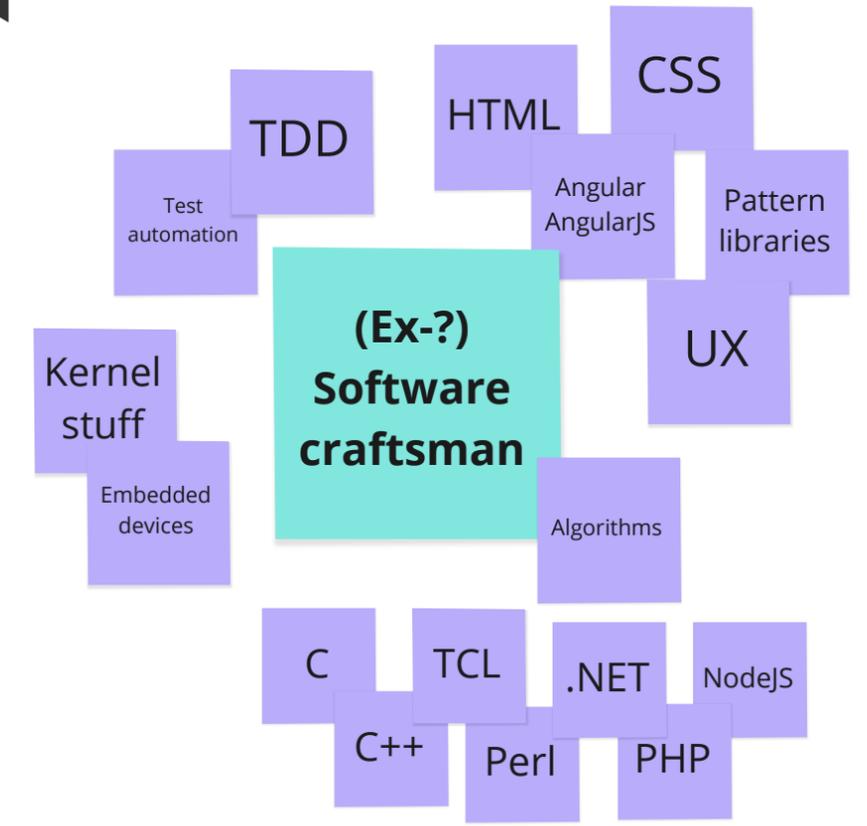
Proefschrift voorgelegd tot het behalen van de graad van Doctor in de Wetenschappen aan de Universiteit Antwerpen te verdedigen door  
Michael Voorhaen



**Software scientist**

Promotor: Prof. dr. C. Blondia

Universiteit Antwerpen



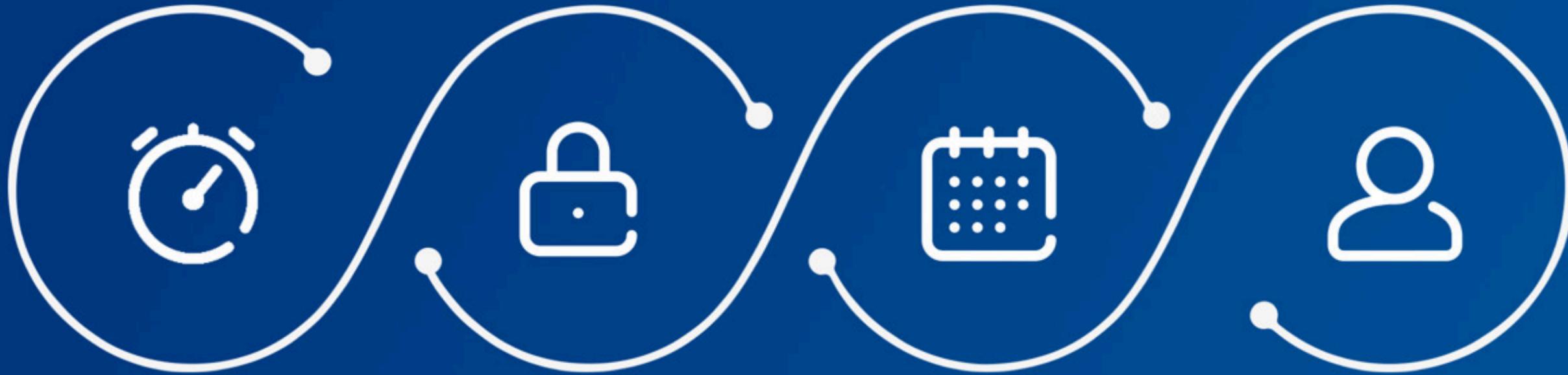


# Time is valuable.

Make time valuable



# Protime provides efficient solutions to optimize the time of your human capital.



## Time registration

Time & Attendance  
Self Service  
Wage Processing  
Cost Accounting

## Access

Access Control  
Visitor Registration  
Hardware

## Planning

Team Planning  
Job Planning

## Services

Extended Services  
Training Center  
Data Services  
Optimization

# Protime today



**8**  
offices



**Once in a  
lifetime  
employer**



**580**  
employees

Since  
**1995**



**10-15%**  
growth



**68**  
mio

**40**  
**Premium**  
countries

**30+**  
**Years**  
of  
experience

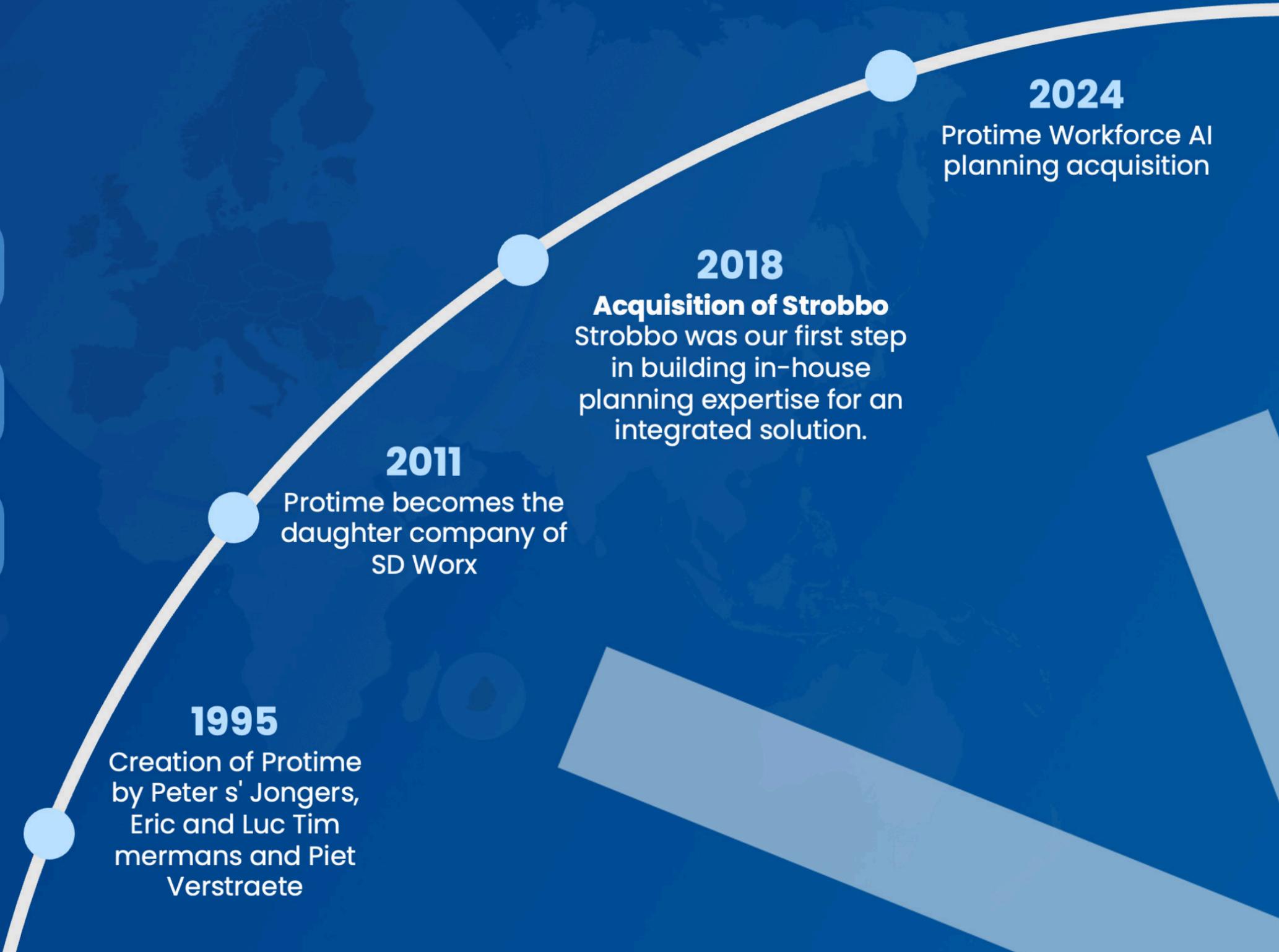
# Our story

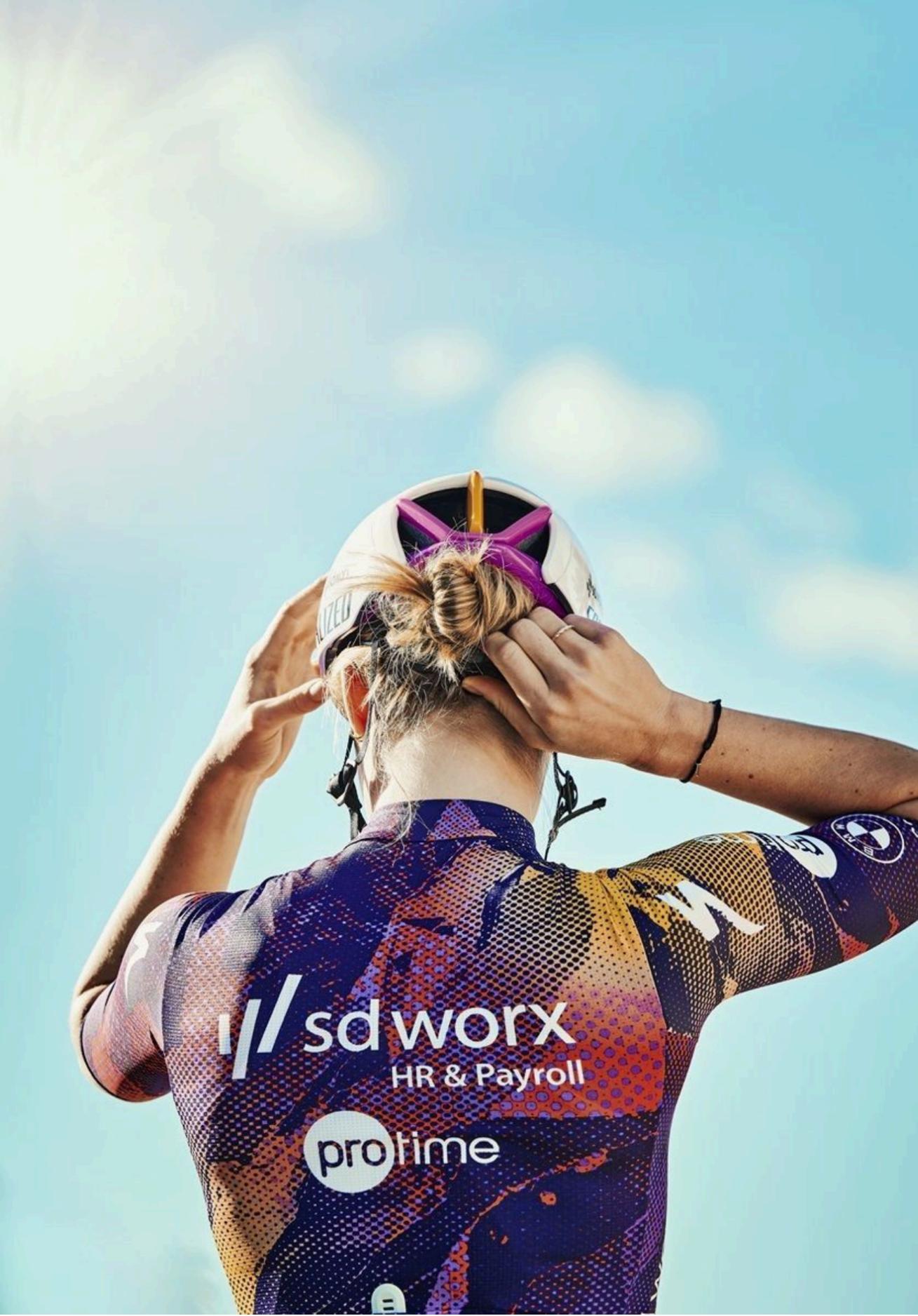
## Why choose for Protime?

Market leader in Europe

Local expertise

Innovative tech company





strobbo

Planification

	lu	ma	me	je	ve
Réserve	+			+	
Compteur					
Cuisine		+			
Strobbo	+			+	
Cuisine			+		

Envoyer les heures de travail

Disponibilité supplémentaire

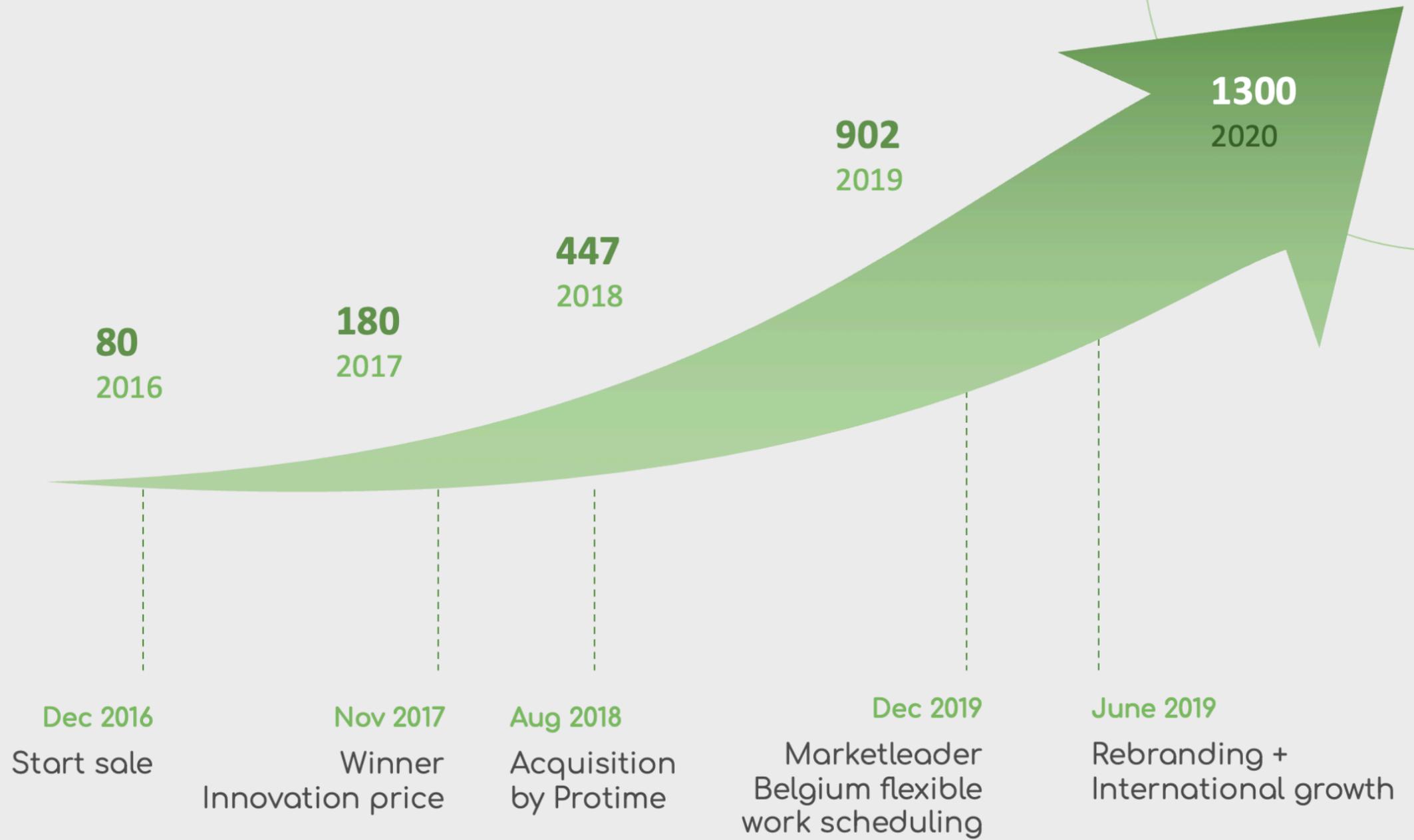
Illustration of a person sitting on a laptop, holding a smartphone, with various data charts and a calendar displayed on the devices.

Strobbo wordt door meer dan 2000 bedrijven gebruikt



Koppel de tools die je al gebruikt





# Founders

Bert

Nick





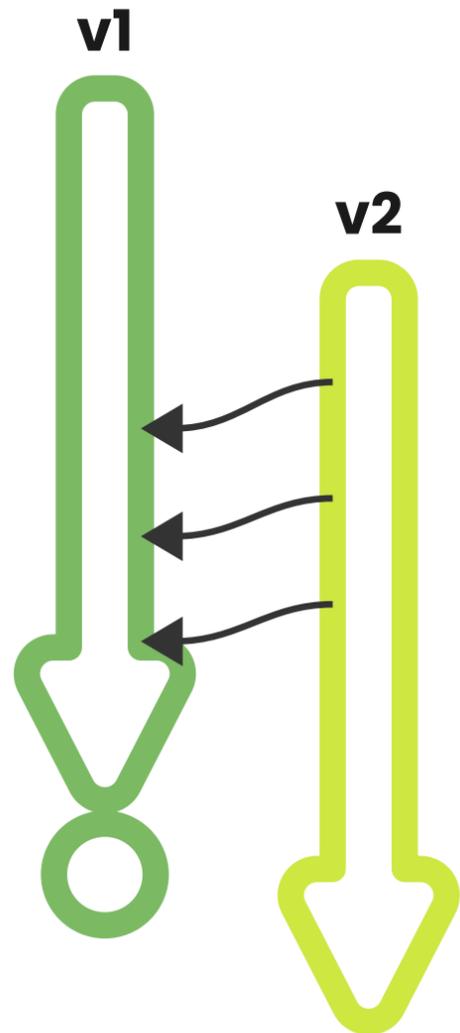


- Grew fast → from 2 devs to 3 devs to a 10+ dev team
- Mostly junior devs

# Struggles

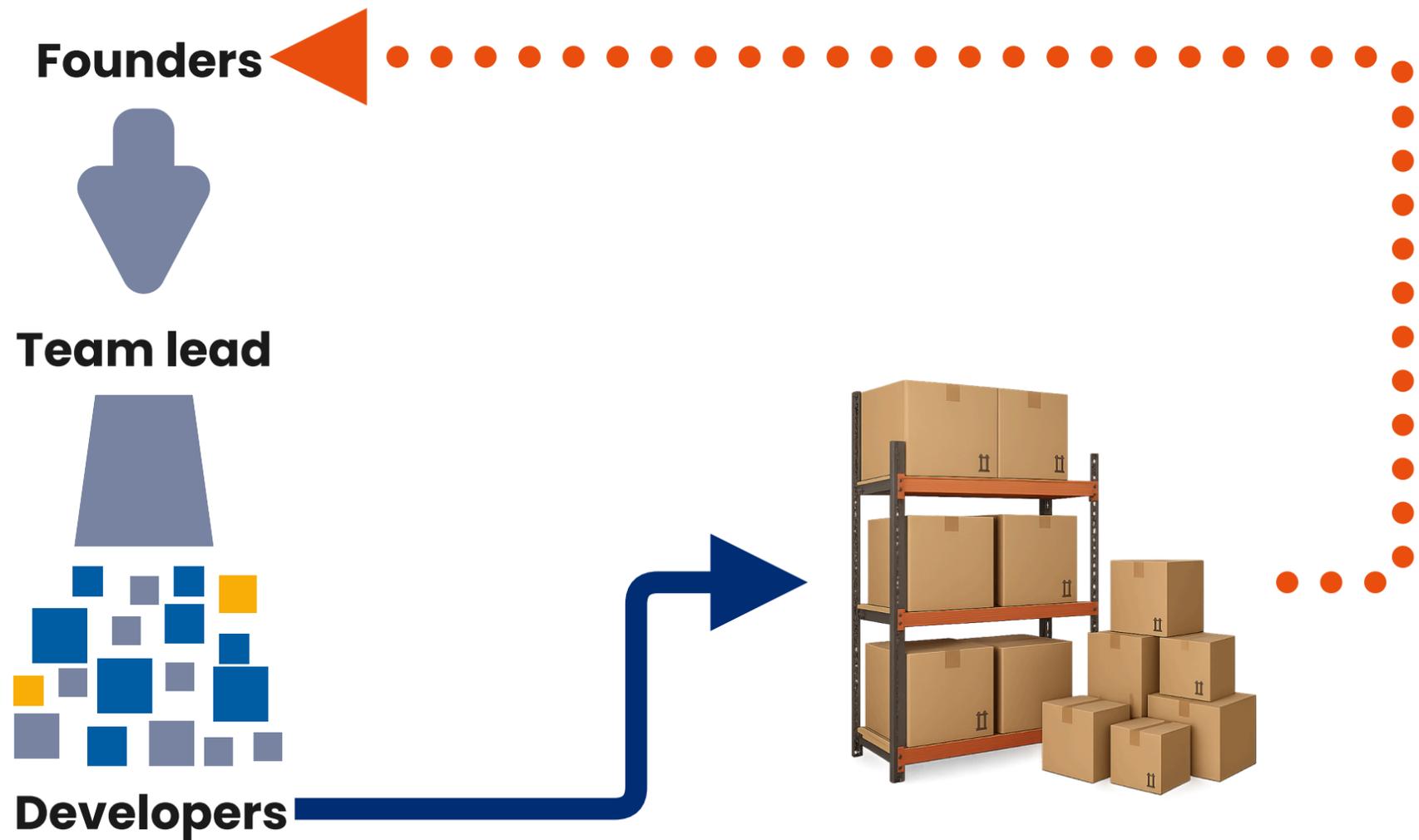
Building a new version of Strobbo alongside the old one with a new team.

The new team could not release without the old one.



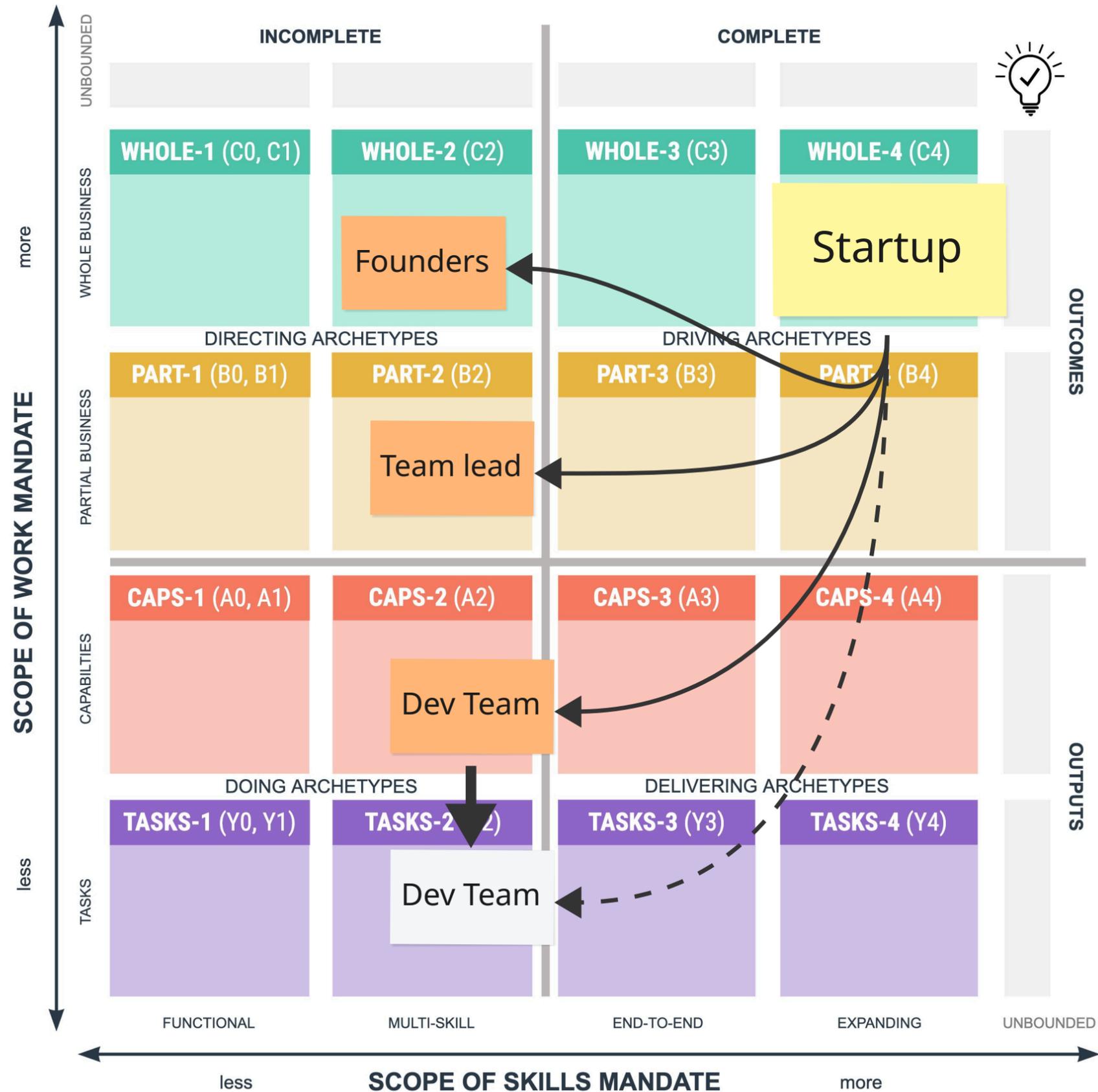
Work was presented fragmented and very detailed to developers.

Devs were identifying as front-end and back-end.



What the team delivered got stuck waiting for feedback and approval from the founders.

When we released they were large chunks with high impact.



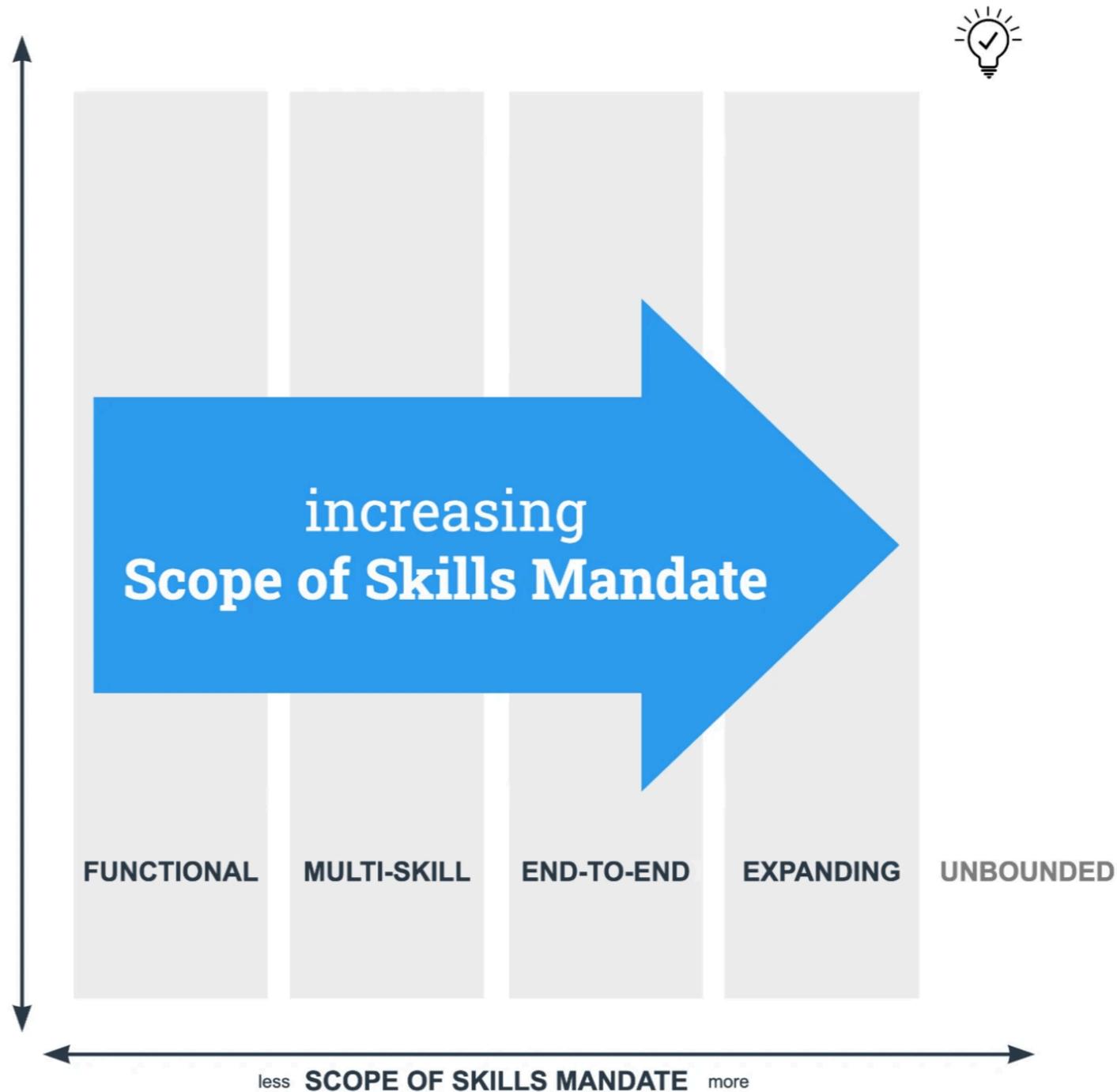
- We started fast – full ownership, end-to-end
- As we scaled, delivery slowed down
- Working on features and drifting to tasks
- Relying heavily on coordination from team lead and founders

How the OT map works

How we found our way back using MADE

How we travelled beyond

Where we are going next



### Left (Functional):

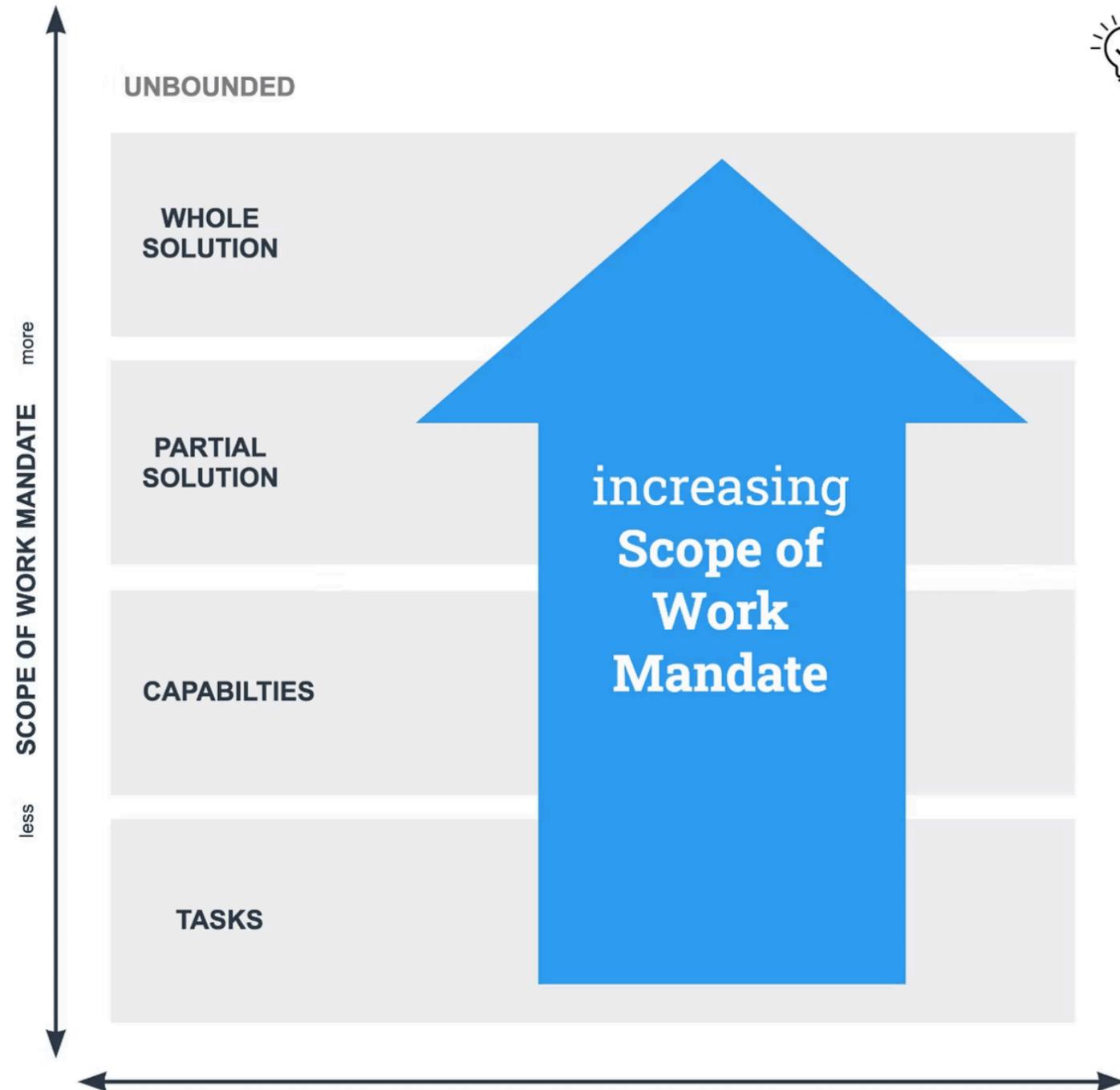
- Narrow specialization
- Strong dependencies
- Fragmented work

### Middle (Multi-skill / End-to-end):

- Broader capabilities
- Reduced handovers
- More complete work

### Right (Expanding / Unbounded):

- Continuous multi-learning
- High adaptability
- Independent value creation



## Tasks

- Execute assigned work
- No ownership of outcomes

## Capabilities

- Deliver functions
- Still fragmented

## Partial Solution

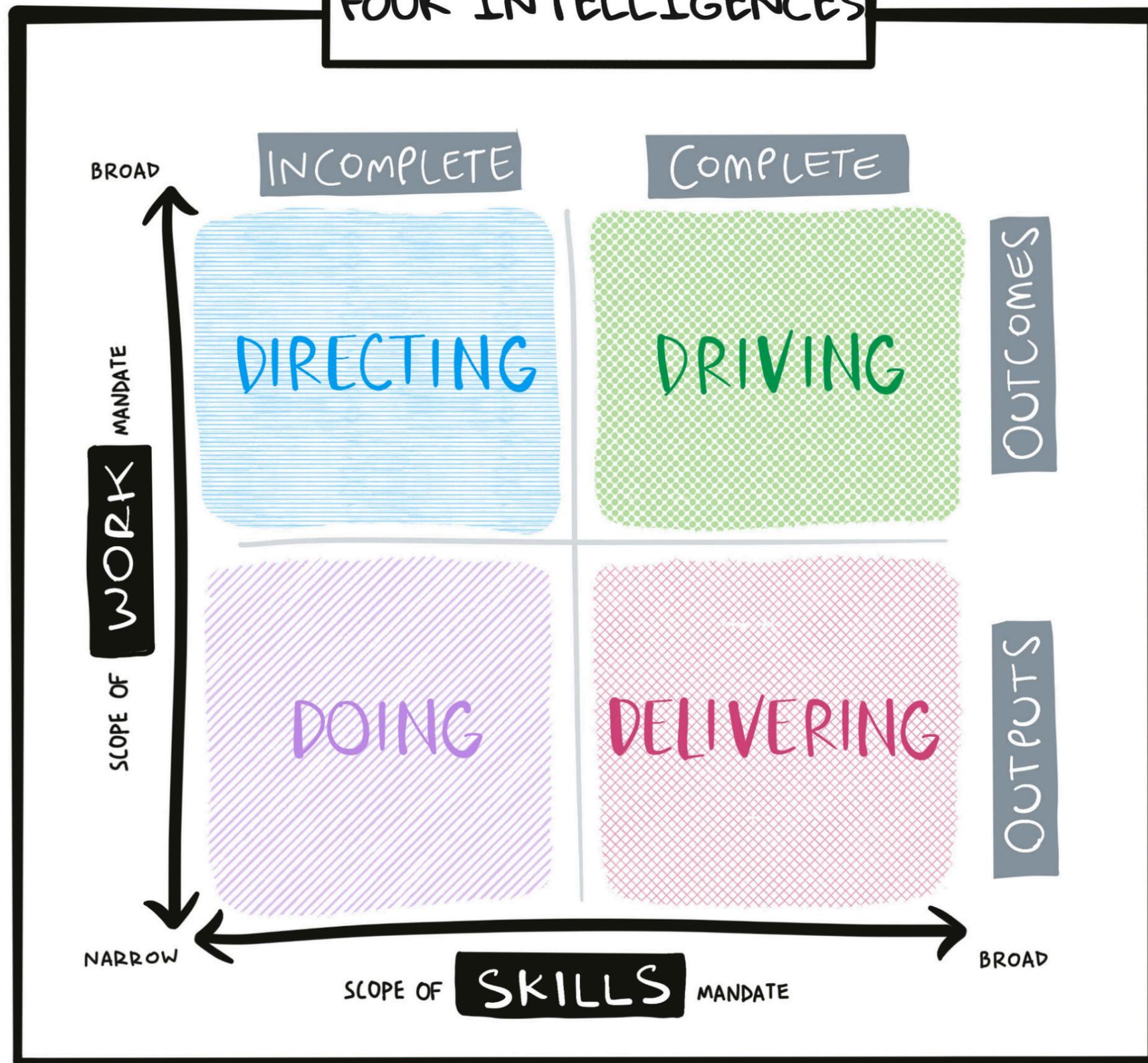
- Contribute to value
- Dependencies remain

## Whole Solution

- Own end-to-end outcomes
- Deliver real value

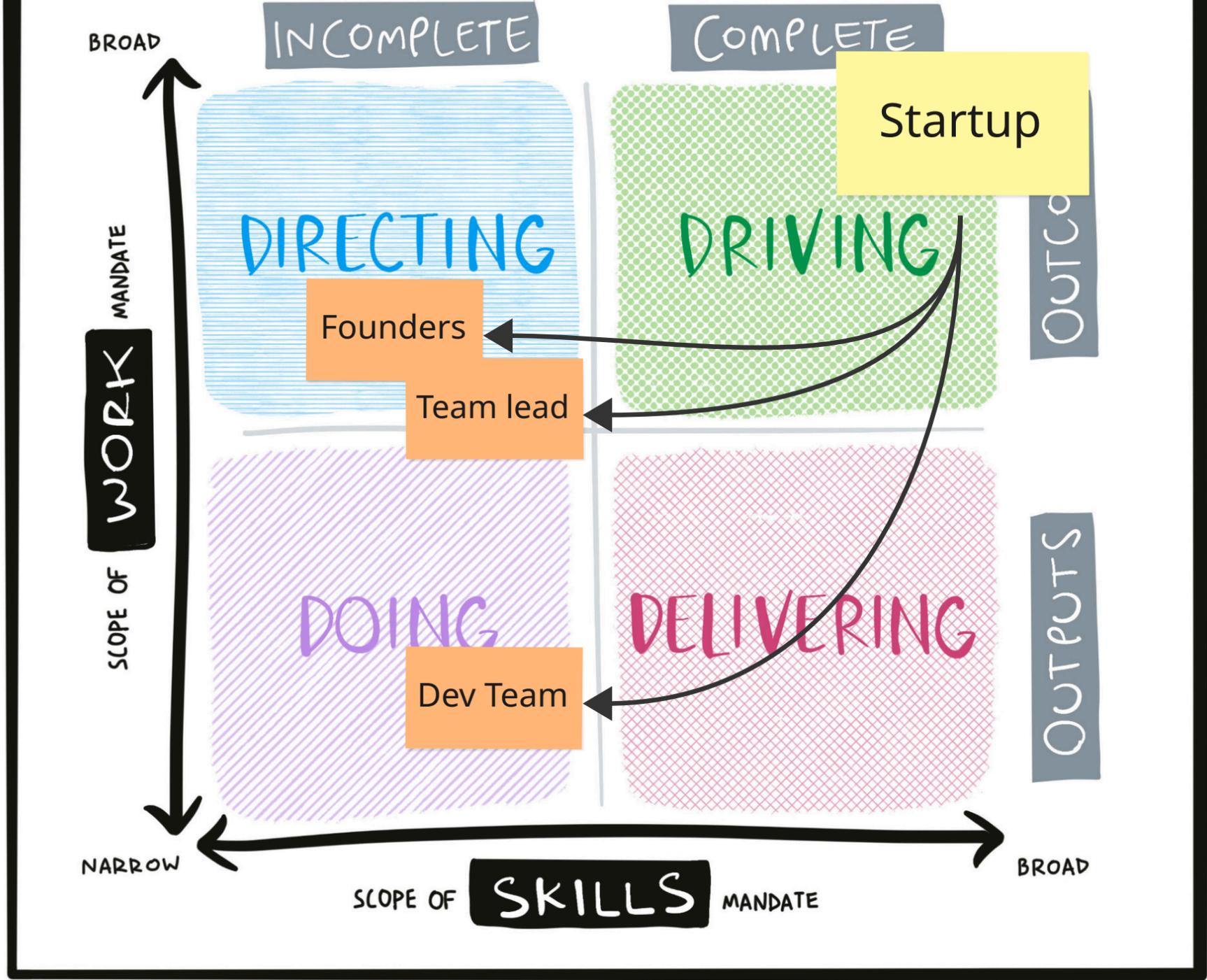


# FOUR INTELLIGENCES

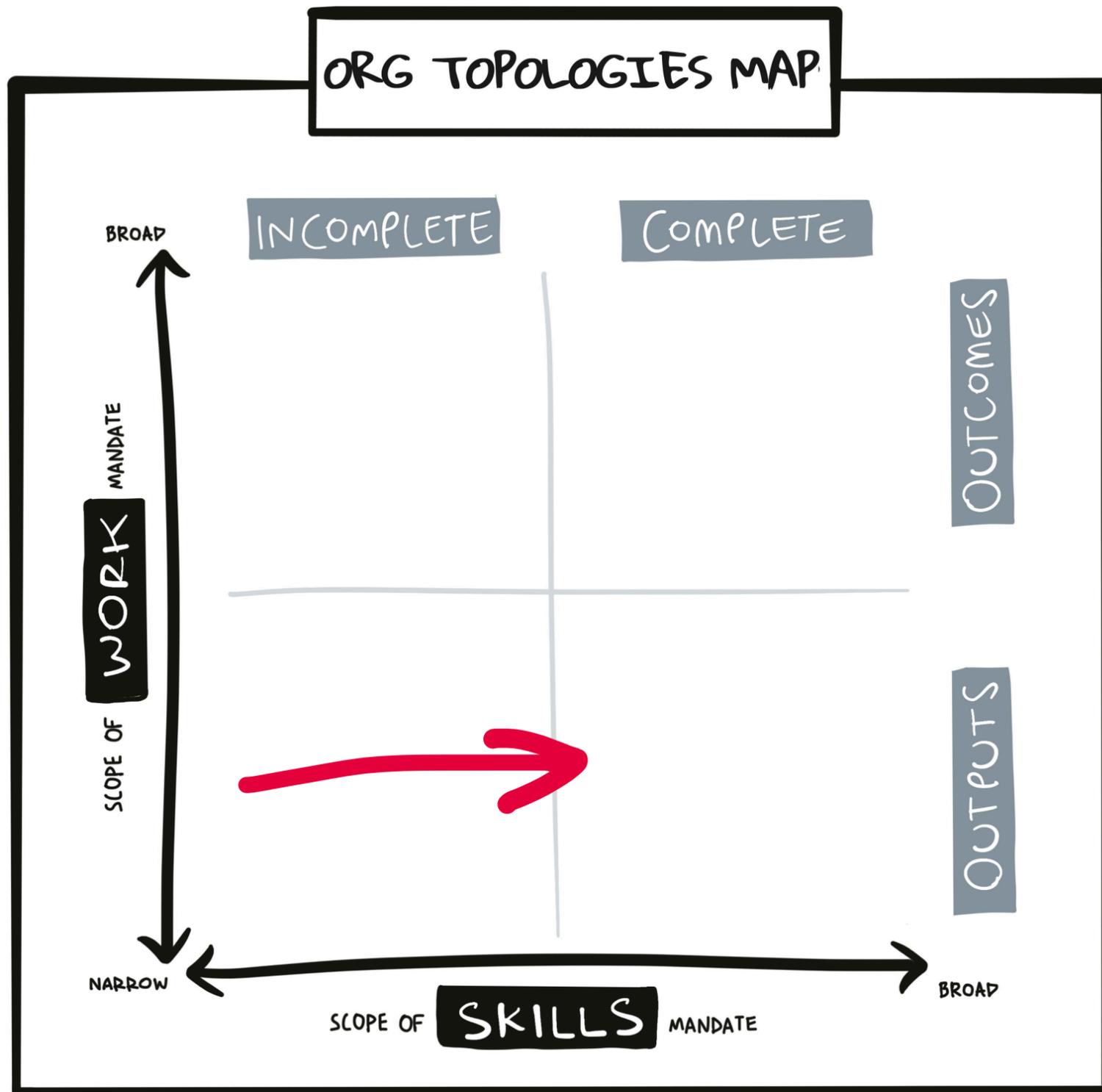


- Doing → executing tasks
- Delivering → producing outputs
- Directing → coordinating work
- Driving → owning outcomes

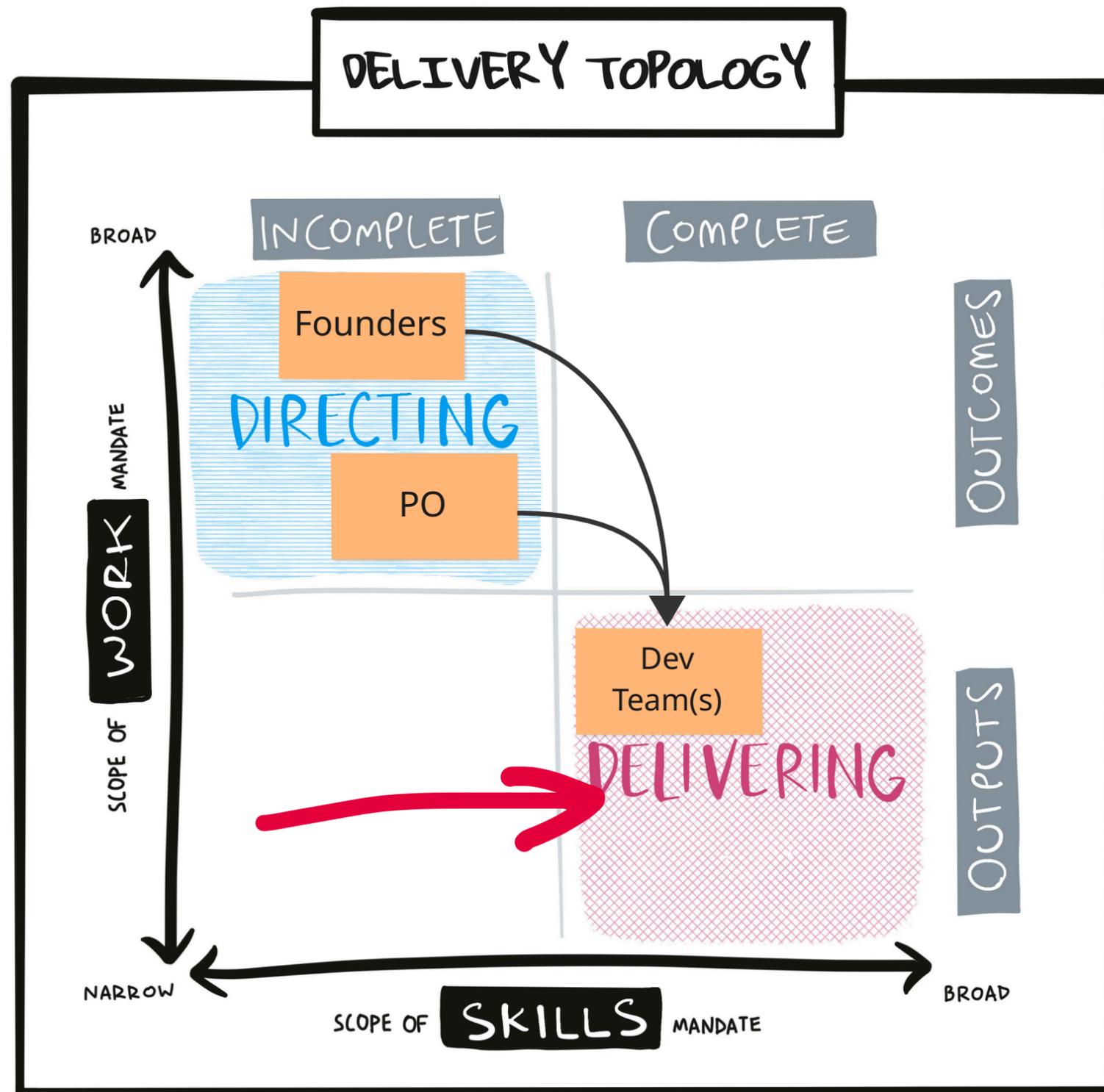
# FOUR INTELLIGENCES



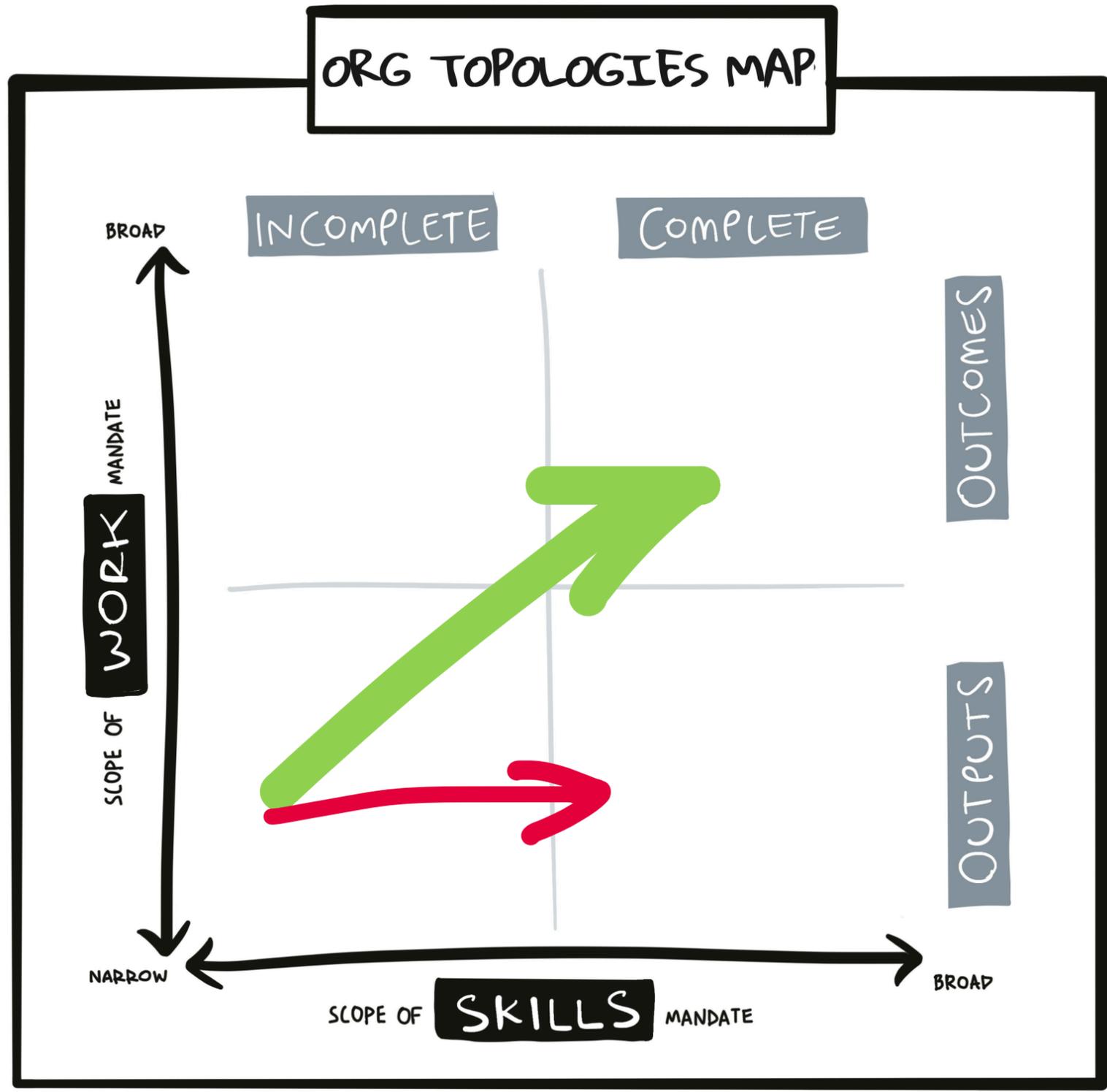
# **Designing our ideal org**

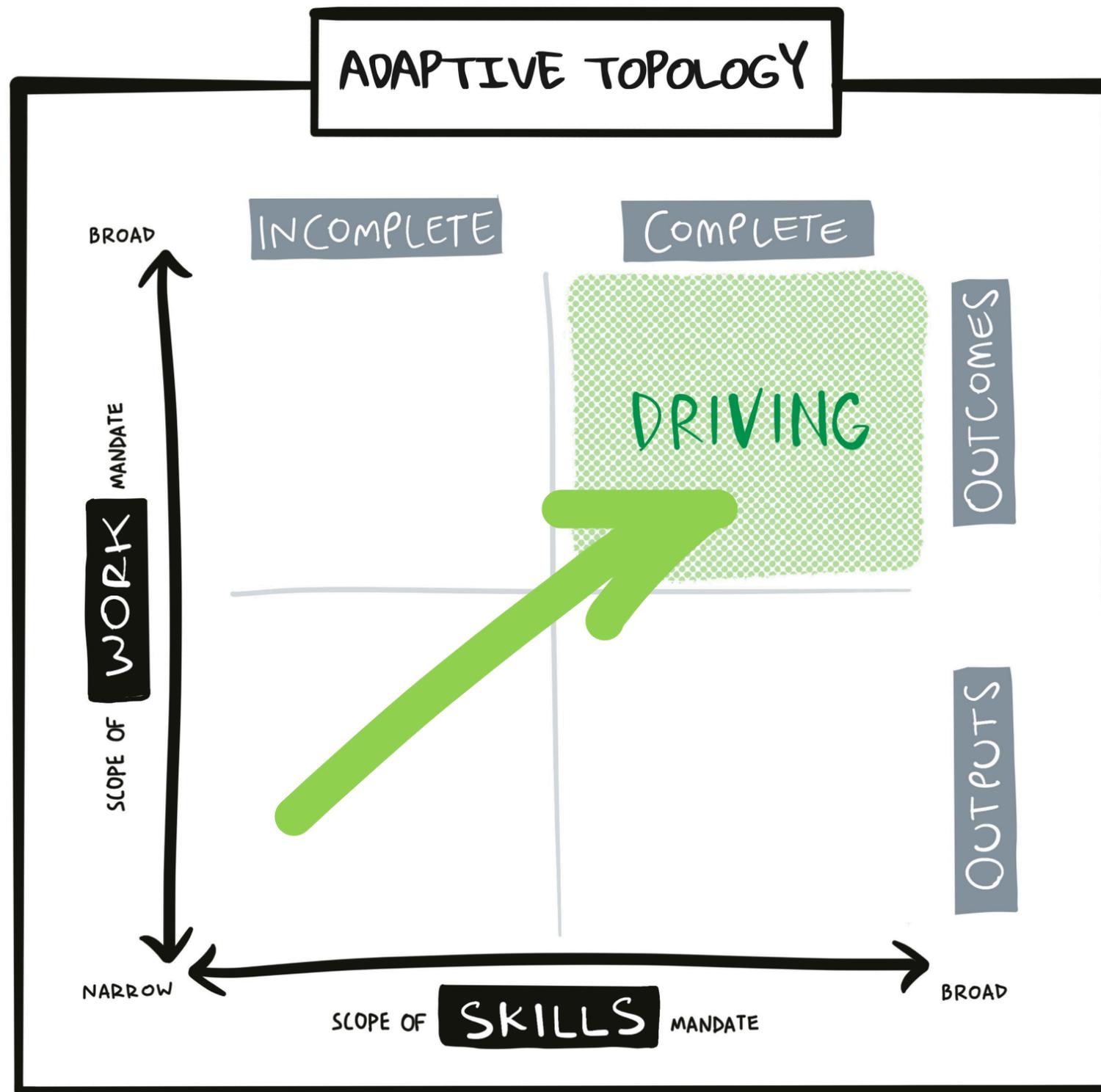


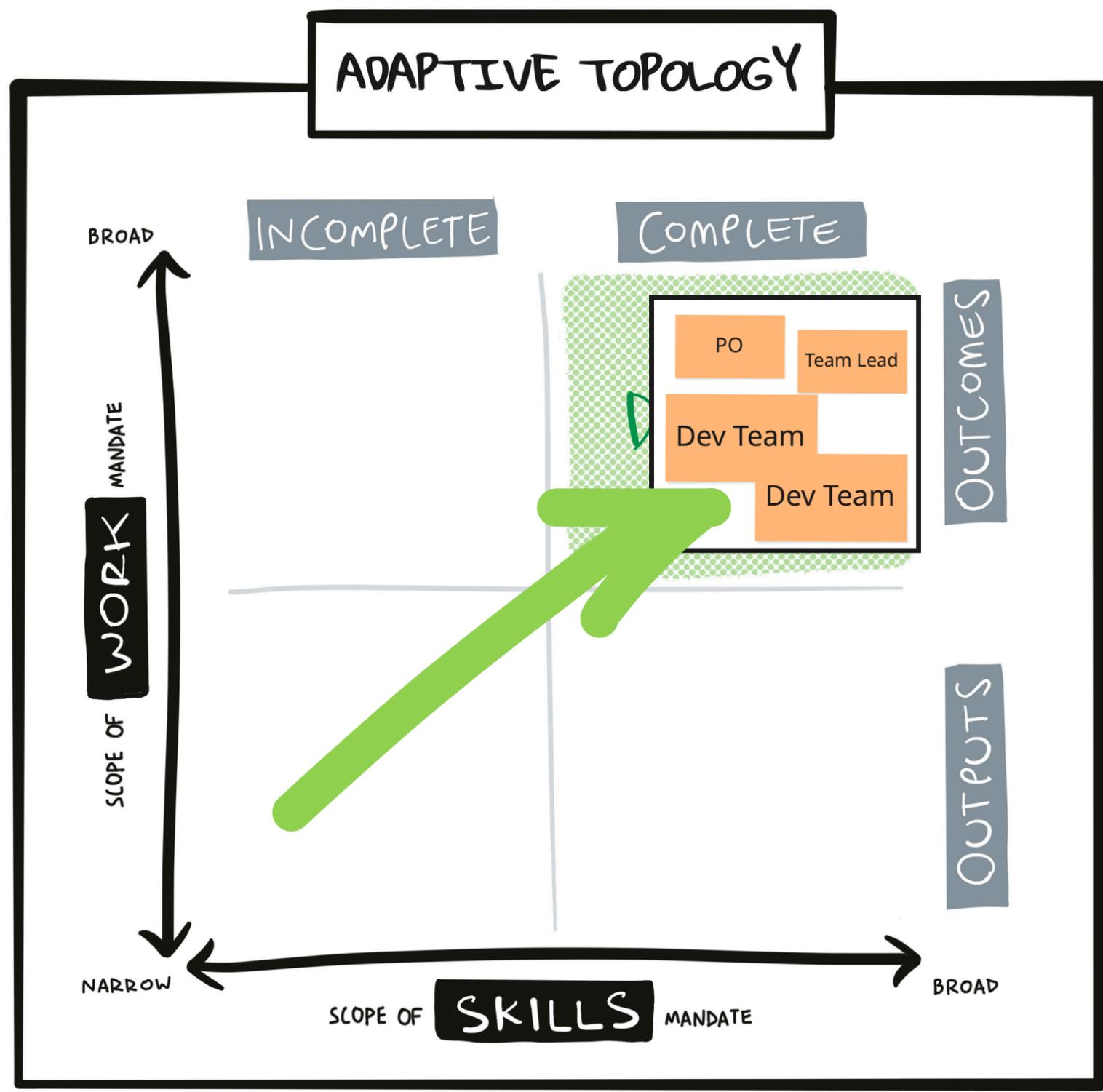
- We initially thought we needed to grow the skills of our teams and that would be enough.
- We focused on this for too long actually.
- But we weren't fixing the root cause with this.

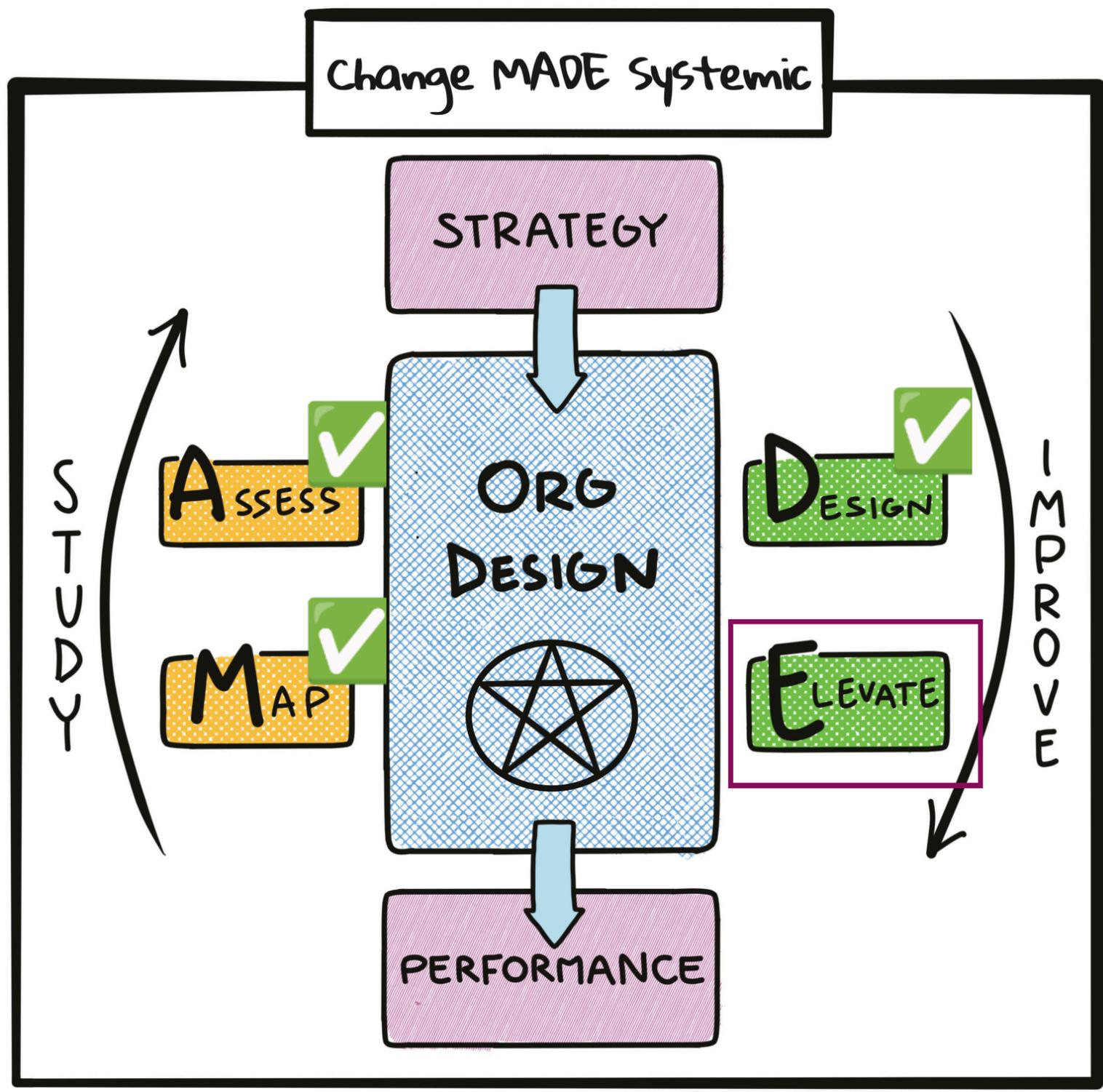


- And we also realized that we were not interested in continuously directing the teams
- We wanted more
- We wanted make a bigger impact









# **Elevating Katas**

# Sprint Review



**What will you do in the breakout?**



You will be helping *Salma* from "Papa Luigi's" create a planning that follows the rules.

This breakout will feature the following:

**Mobile - Tenant switch**

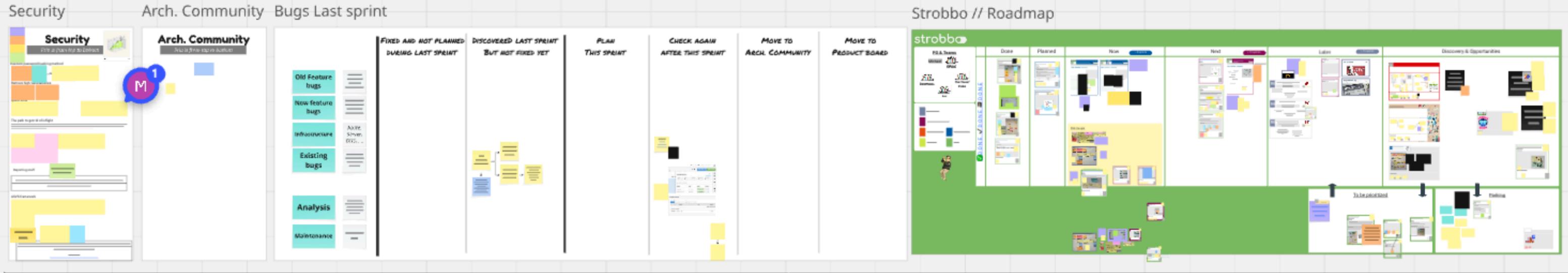
- Allows the user to switch to a different tenant
  - Only get push notifications from you active tenant
- Consultants can also switch and enter a new reason

**Breakout**

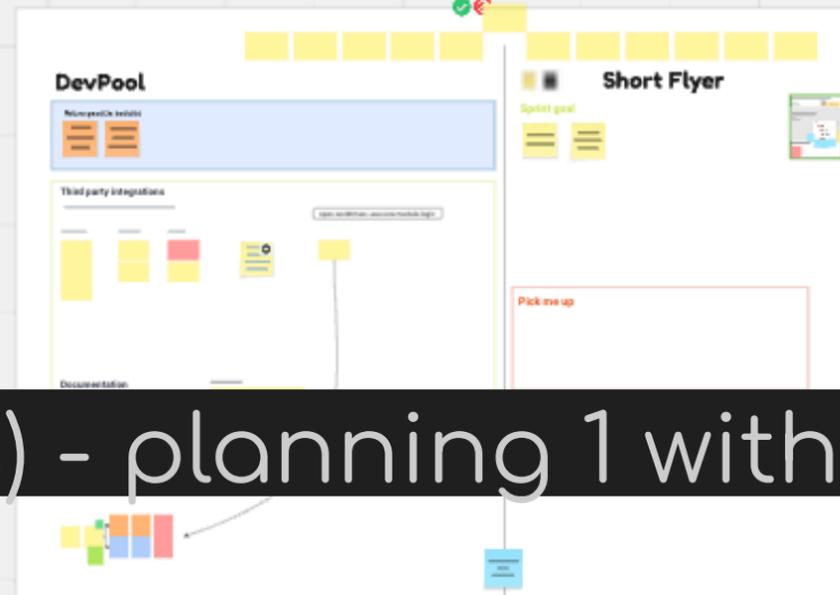
- Bazaar-style sprint reviews
- → transparency & whole-product focus

# Planning as an Elevating Kata

Planning 1



Planning 2



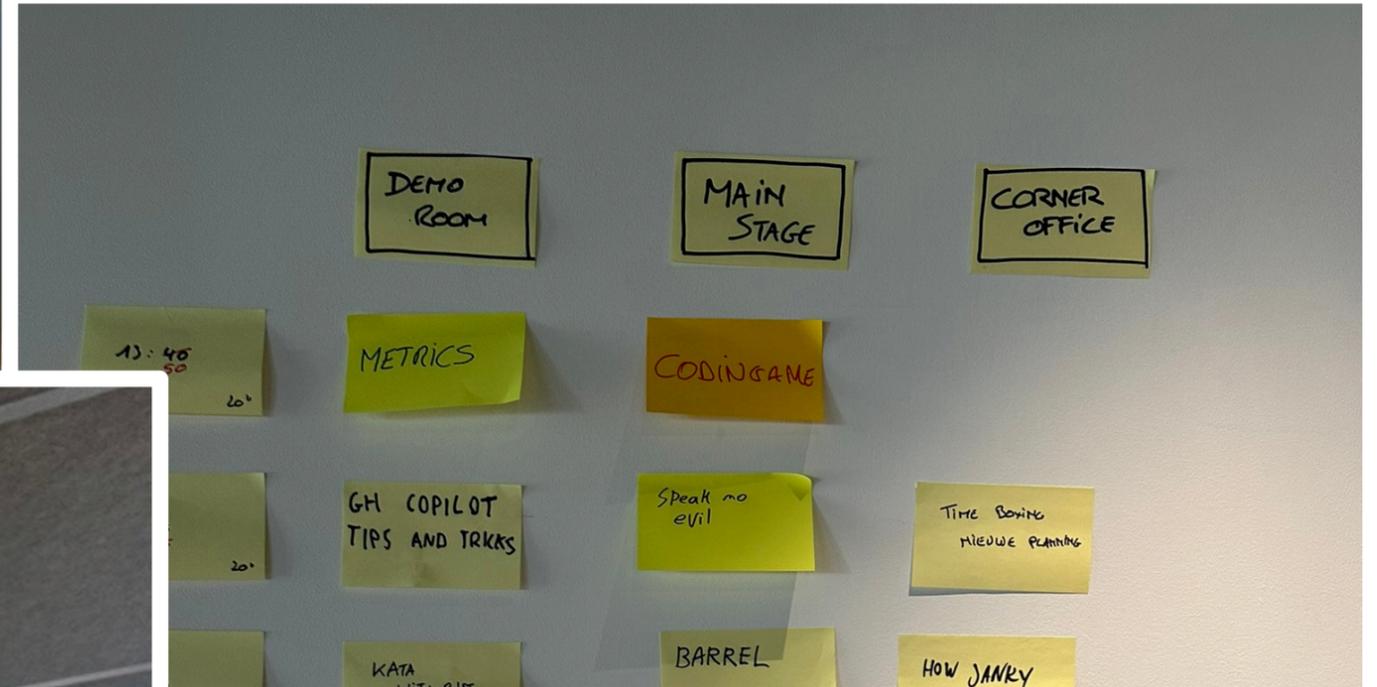
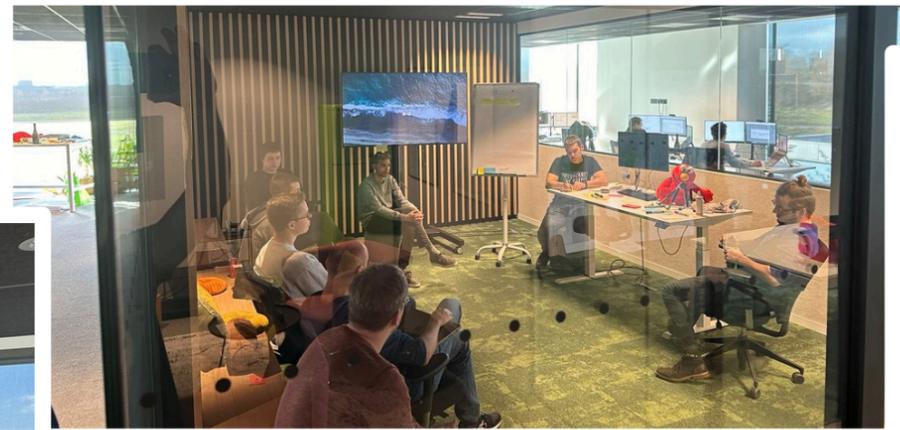
- Planning 1 & 2 (like LeSS) - planning 1 with the entire team
- On-site
- Built out routines to help us stay on track
- What do we think are the most meaningful sprint goals over pulling in work items

# Refinement as an Elevating Kata

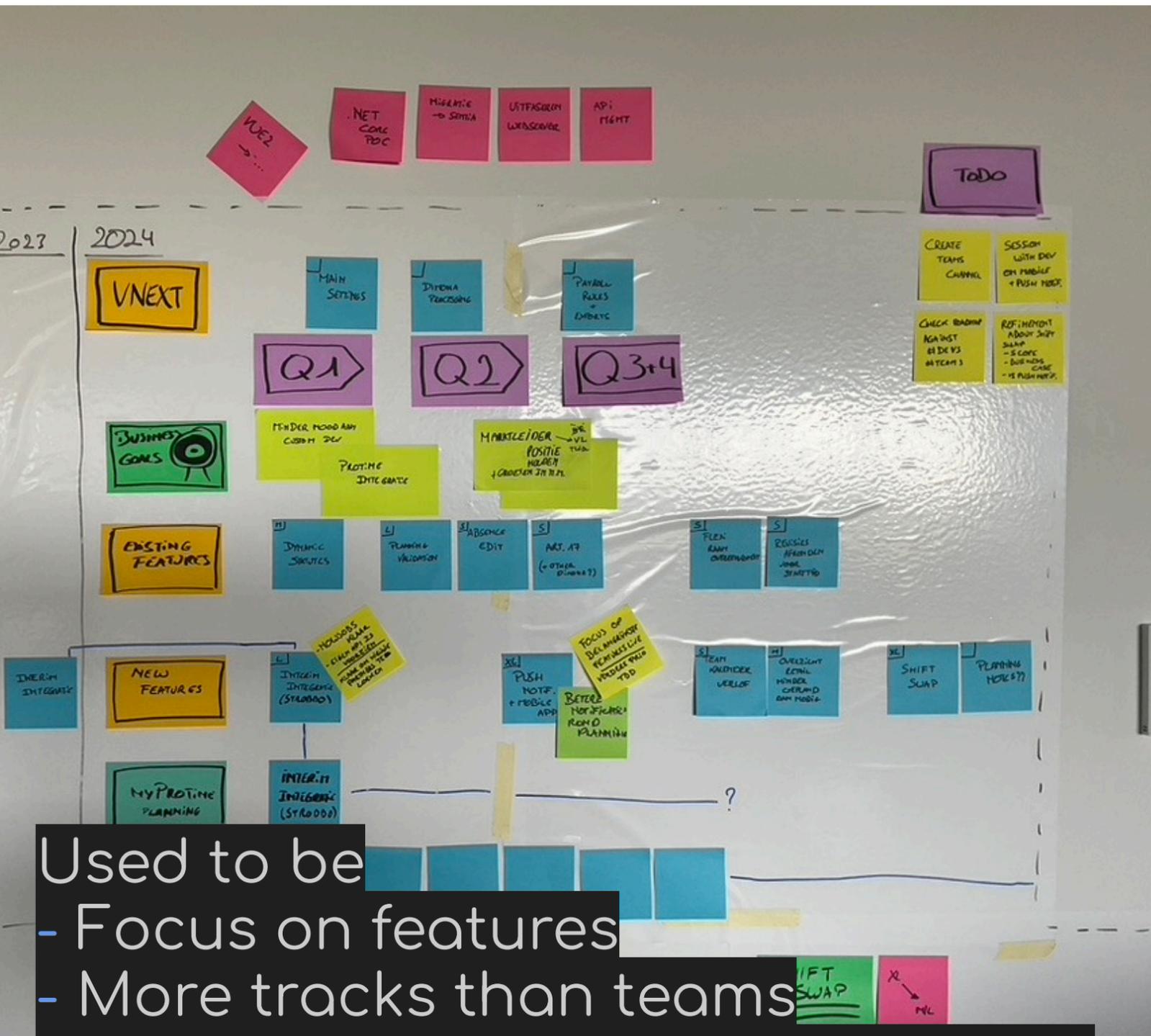


- Cross-team and involving internal stakeholders
- Evolved from
  - understanding stories → working on outcomes → product discovery
- Early on combined this with learning and knowledge transfer
  - understanding features, our domain, etc.
- Now: joint understanding through discussion

# Dev Days



# Roadmap / Backlog



Used to be

- Focus on features
- More tracks than teams
- Unclear what was most important
- Busy with too many things at once

0

**Leftovers of 2023 goals**

- vNext breakout rollout
- Phase out reporting 1.0

#Clients on vNext  
#Timeclocks migrated

#Customers still using reporting 1.0

Stability vNext

7 of 4 sprints

1

**Existing planning sync customers are migrated to external contracts**

- All **live** with **planning sync**
- Availabilities can be synced from **app**
- depending on status**

**live** **Pilot live?**

Integrations Stability

6 of 4 sprints

2

**Deeper integration with SD Worx Payroll**

- Extra fields with employee sync
- ...

# customers still using old sync  
# customers using the new sync

Deals won because of this

vNext Market leader

Now sprints

3

Goals focusing on outcomes

Clear order in which to proceed

WIP limit = the amount of teams

# Bets

- Estimates were associated with frustration and mistrust.
- But we still need a way to "control our clock"
- Introduced bets as a middle ground
  - Instead of "How long will it take?"
  - I started asking "What can I get in this timeframe?"
- As a PO you can think about investments



6 sprints

**Unit:** 1 sprint goal for 1 team

18

Default planning

*Protects and strengthens our market leader position in BE*



3 sprints

**Enable planners to define recurring, multi-week default plannings for employees**

**Allow planners to load default schedules directly into a selected planning period**

**Ensure planners can modify default schedules without disrupting current schedules**

## Bets

- Estimates were associated with frustration and mistrust.
- But we still need a way to "control our clock"
- Introduced bets as a middle ground
  - Instead of "How long will it take?"
  - I started asking "What can I get in this timeframe?"
- As a PO you can think about investments



6 sprints

**Unit:** 1 sprint goal for 1 team

Bets have become a conversation starter.

"What is most important to deliver first?"

"Which tech debt can we clean up?"

"Are we still on track? Do we need to make decisions?"

"Do I need to invest more time?"

# Bets

8

## Dynamic statutes and rule-based planning validation

*Protects our market leader position in BE  
Additional configurability for international markets*



**Dynamic statutes**

Combines well with incremental delivery

- Sprint 1: disable and hide old static statutes
- Sprint 2: add your own statutes
- Sprint 3: move some settings to dynamic statutes to give more options to users
- Sprint 5: release new planning validation with new functionality in beta next to the old validation
- Sprint 6, 7, 8, 9, 10: release additional rules and roll out to everyone
- **Sprint 11: invest in blocking validation rules**

## Dynamic Statutes - Update

3 statutes created during last sprint

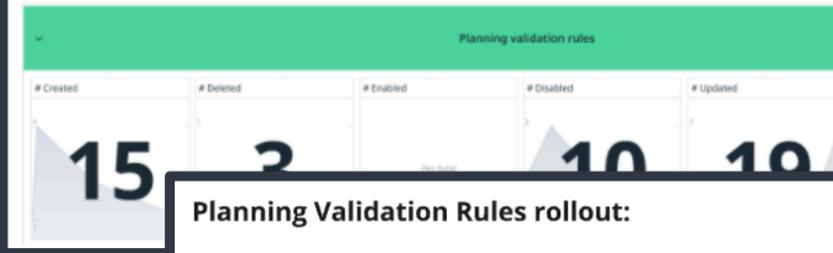
### Updated fields

name	18
hasToSignAllowedAbsence	14
translations.nl	13
translations.fr	9

## Planning Validation Rules Updates:

Some new widgets:

- Avg. # errors per sending of worktimes
- Avg. # warnings per sending of worktimes
- Total # of errors bypassed per sending of worktimes
- Total # of warnings per sending of worktimes

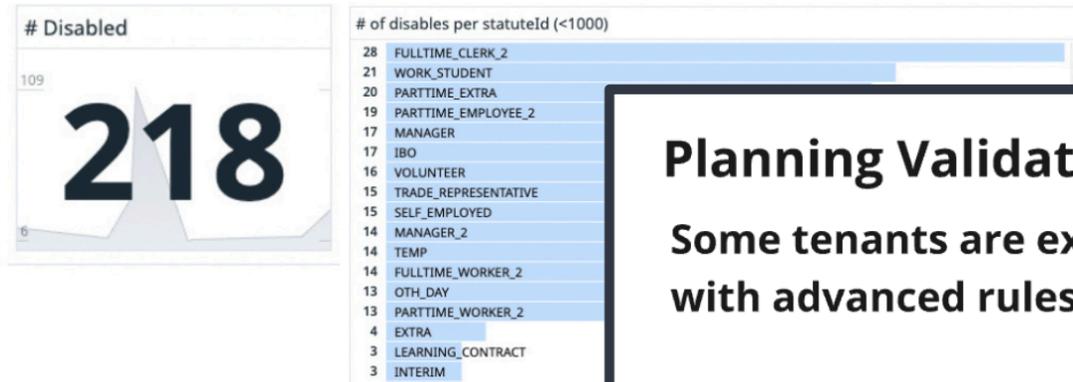


## Planning Validation Rules rollout:

Currently: 10/07/2025  
**60%** → **100%**

## Dynamic Statutes Update

200+ statutes disabled during last sprint



## Planning Validation Rules Update

Some tenants are experimenting with advanced rules



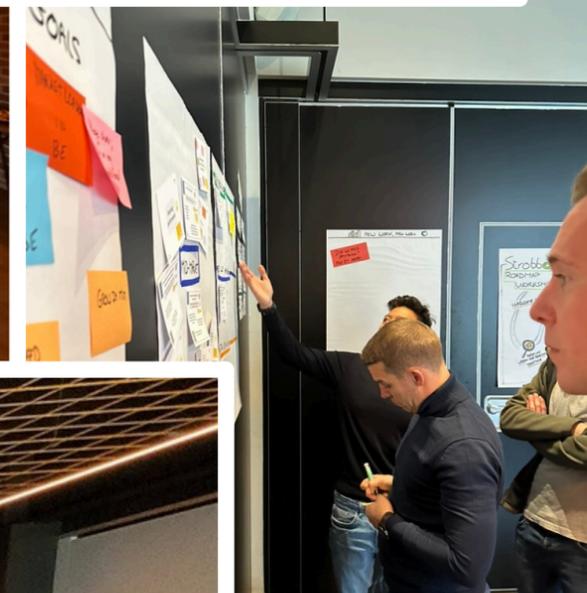
Some examples:

- **Students** must be planned for at least **4 days** per **month**
- **All employees** must be planned for at least **32 hours** per **1 week**
- **Part time employees** must be planned for at least **24 hours** per **1 week**
- ...

## Blocking Rules Update

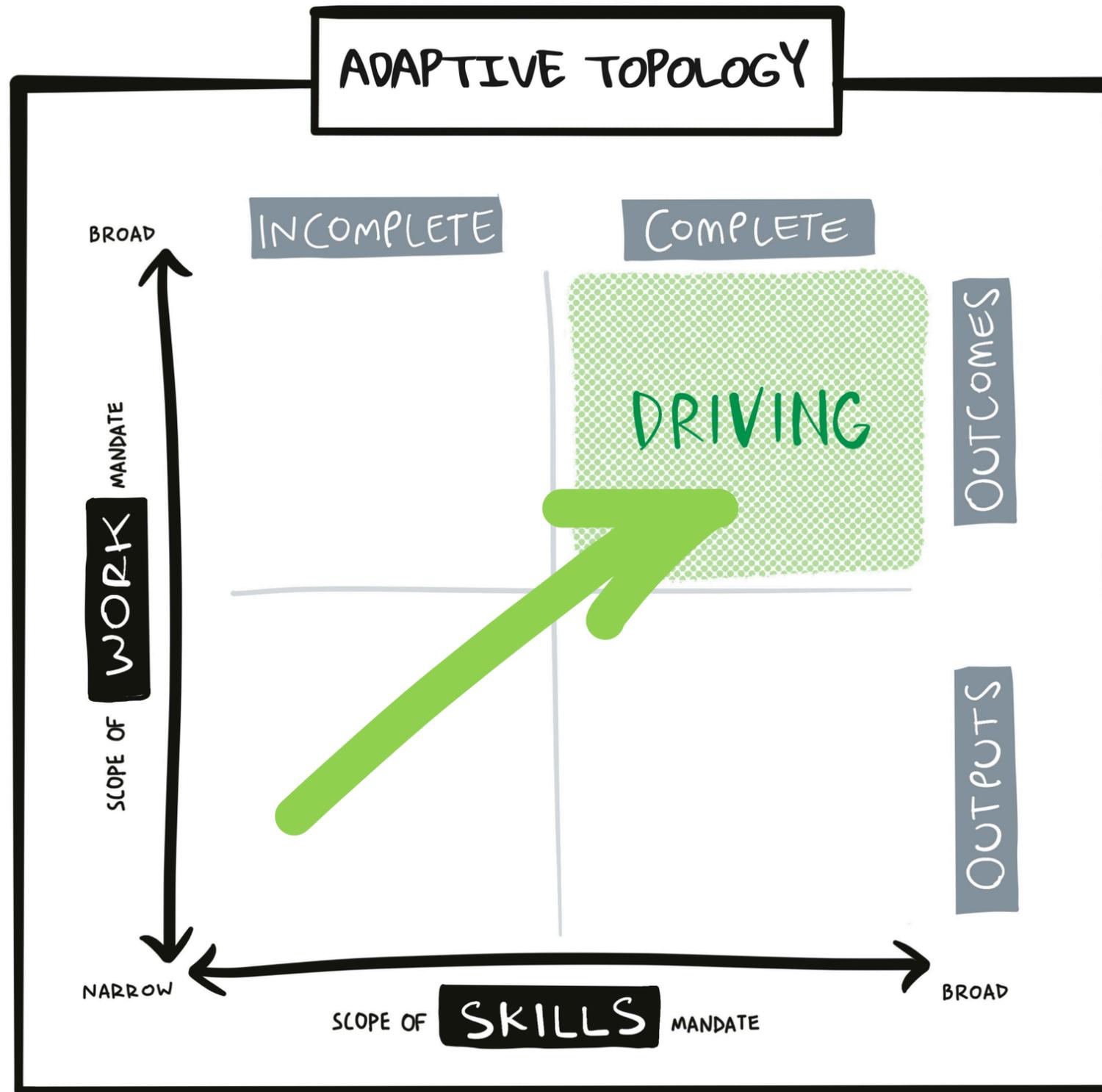
- 🚫 **53 rules** have been marked as **blocking** across **12 tenants**
- 🔒 **13 tenants** only allow **admins** to overrule blocking errors
- ⚠️ **Blocking errors** have been **overruled 17 times** across all tenants

# Roadmap workshop



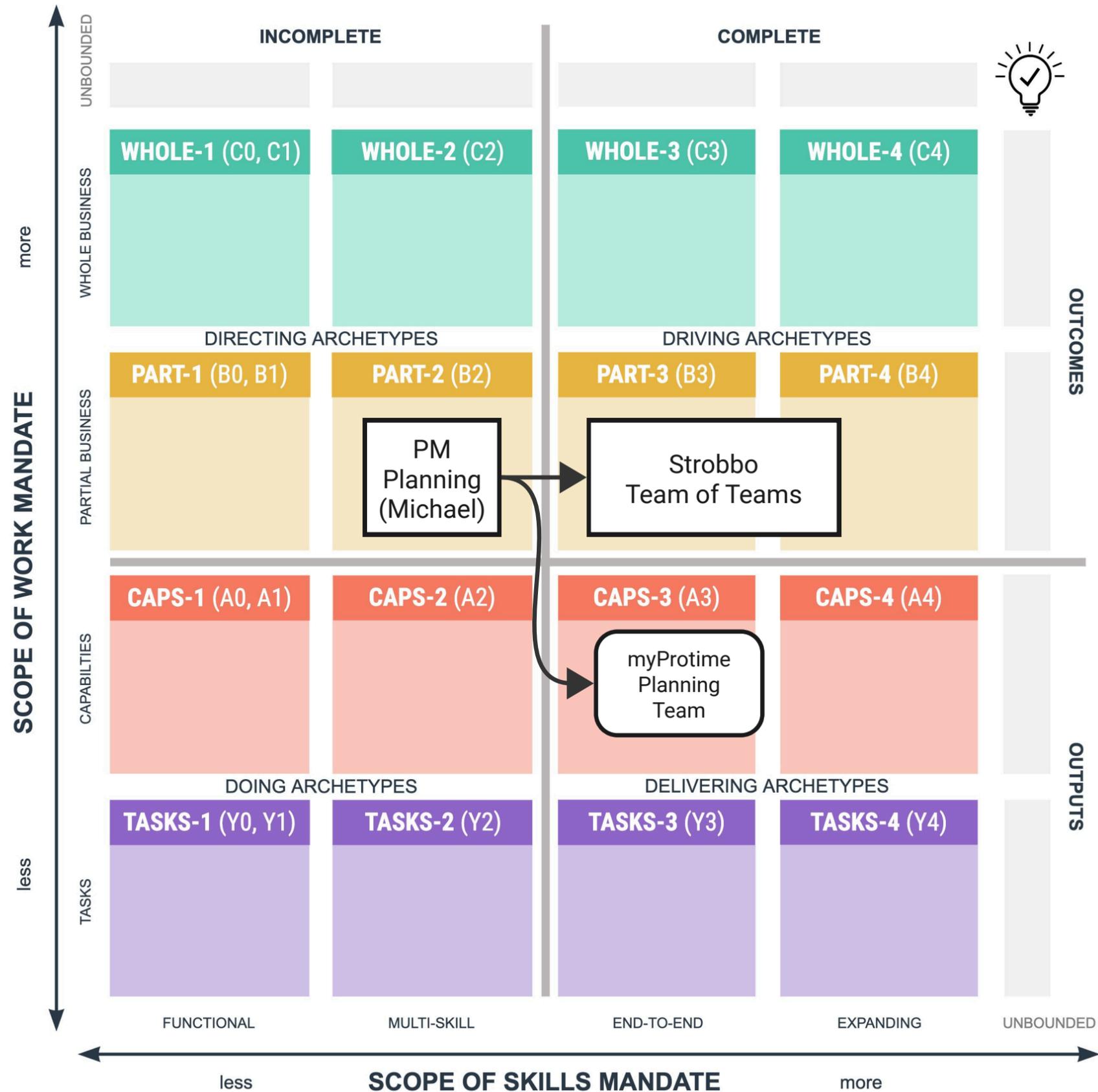
**Going to 2 Products**

What do you need to integrate the  
planning engine of Strobbo in  
myProtime?

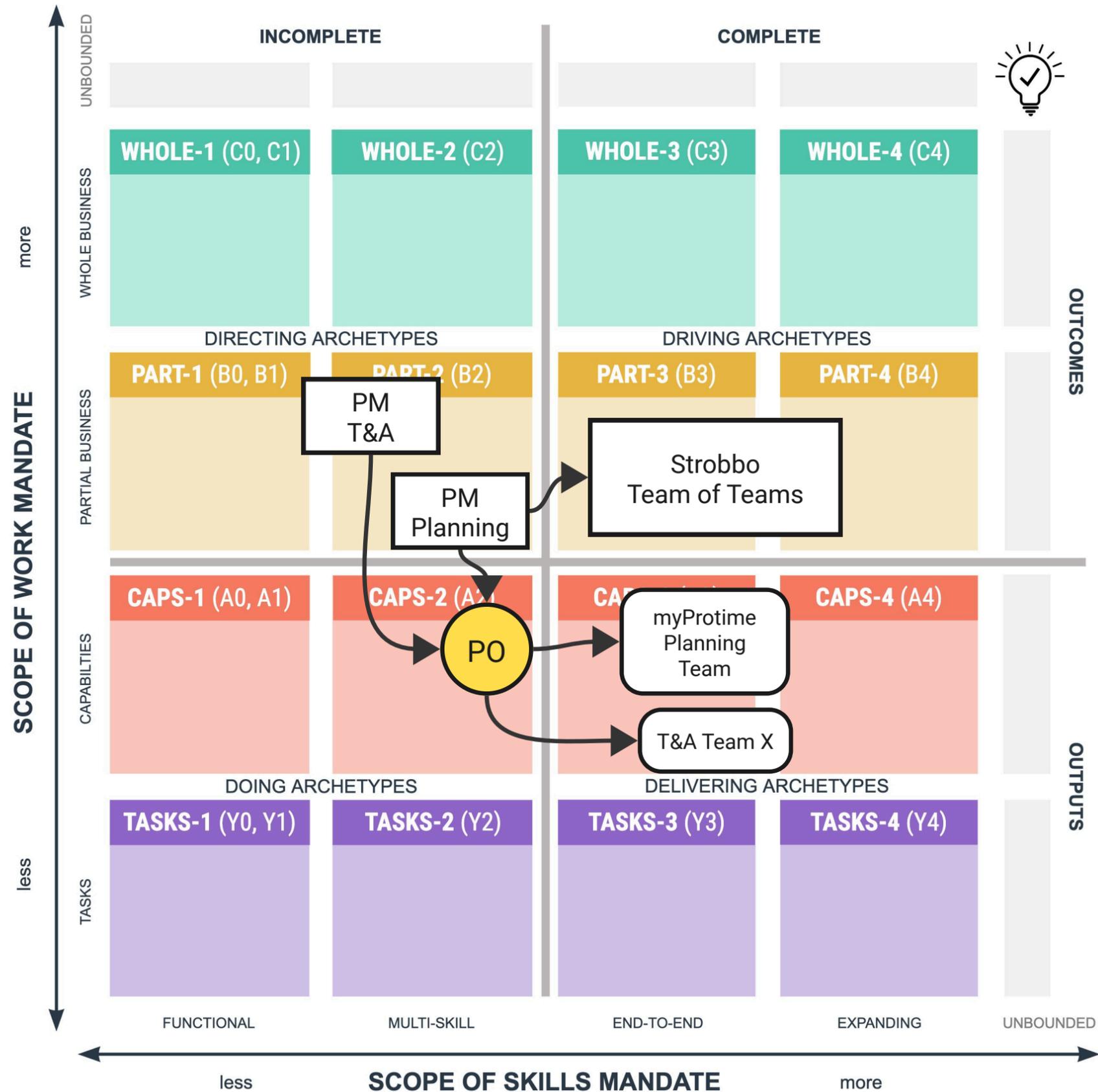


Strobbo

myProtime  
Planning



- Created a team dedicated to work on the new integrated product
- All skills needed to build this end-to-end
- I shifted to a PM position internally, but I was still the PO for Strobbo
- myProtime Planning had it's own PO mainly because we felt we needed someone there who could really work with the team on a day-to-day basis
- Shared Sprint Review / Planning
- Separate refinements

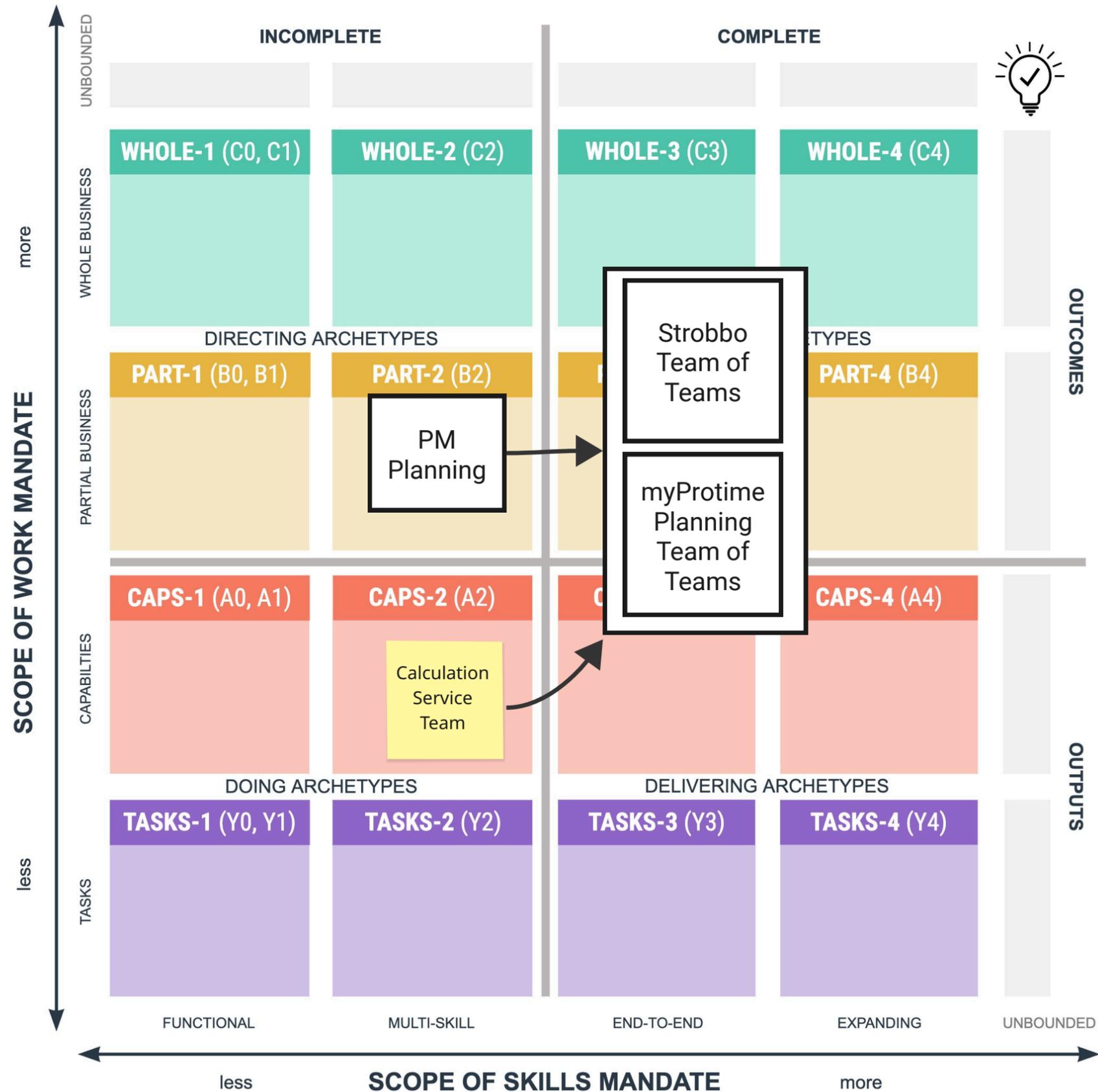


## Some pain points:

- Double role as PM / PO
- PO of myProtime planning responsible for 2 teams
  - in different products
  - with different PMs

Good idea, but we bumped into some parts of the organization we were not able to make changes to

**After 2 years**



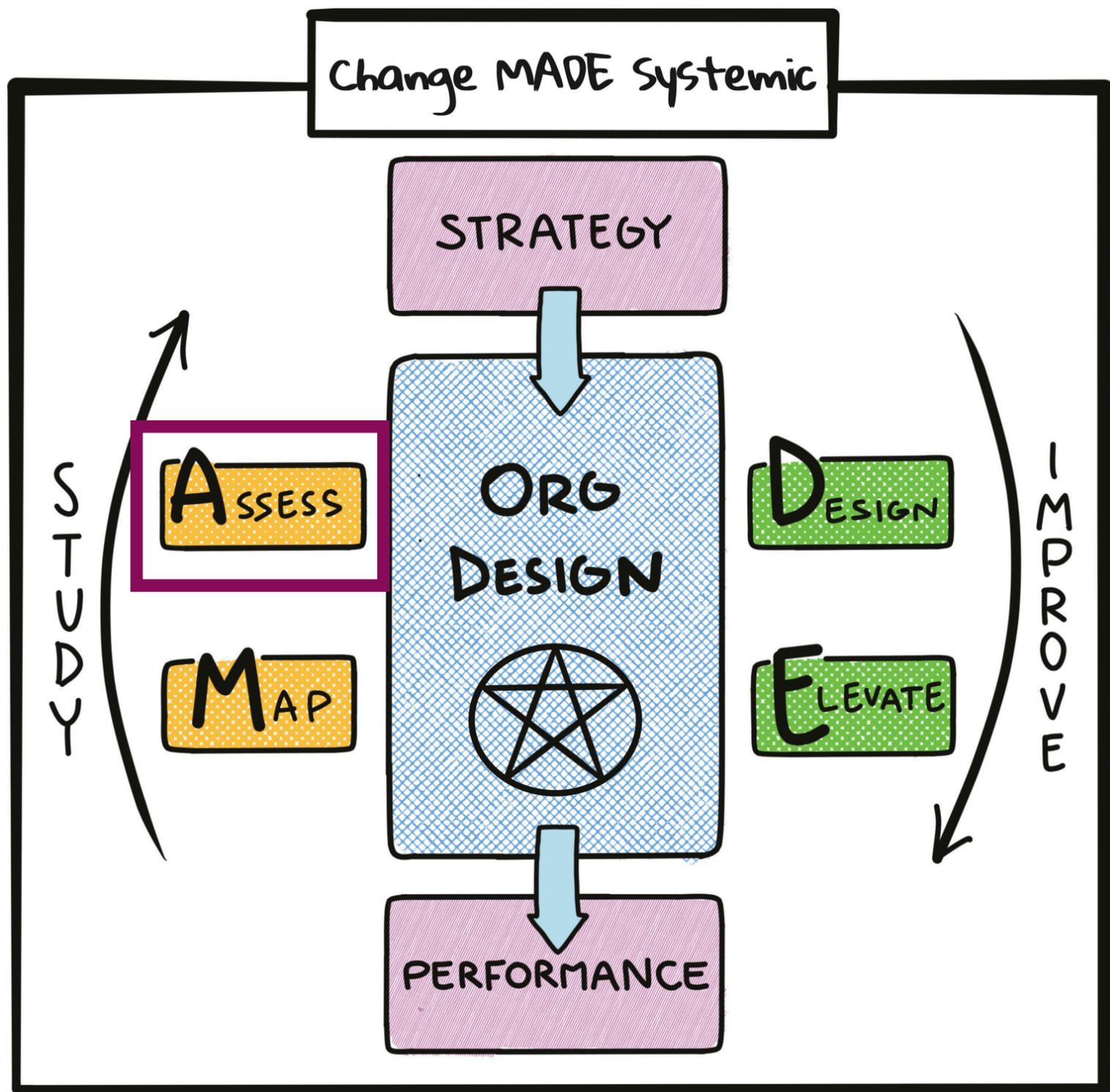
- Grown to 2 full team-of-teams
  - Strobbo 2
  - myP Planning 2
- Each ToT has a dedicated PO
- Split of Planning Calculation Service to a dedicated team to create focus at the cost of some more overhead to integrate

But also these elevating katas:

- Shared learning day once a month
- Shared sprint review
- Sprint planning per product, but with quick check in before we split
- Refinements per product, but lot's of people join both
- Travellers to share knowledge
- POs are encouraged to come up with goals that can deliver value for both products and that teams can collaborate on

**What now?**

AI of course ...



*(focusing just on Strobbo for a moment)*

Teams feel oversized

We are delivering more value **and at the same time** reducing tech debt

Goals have become smaller and better split

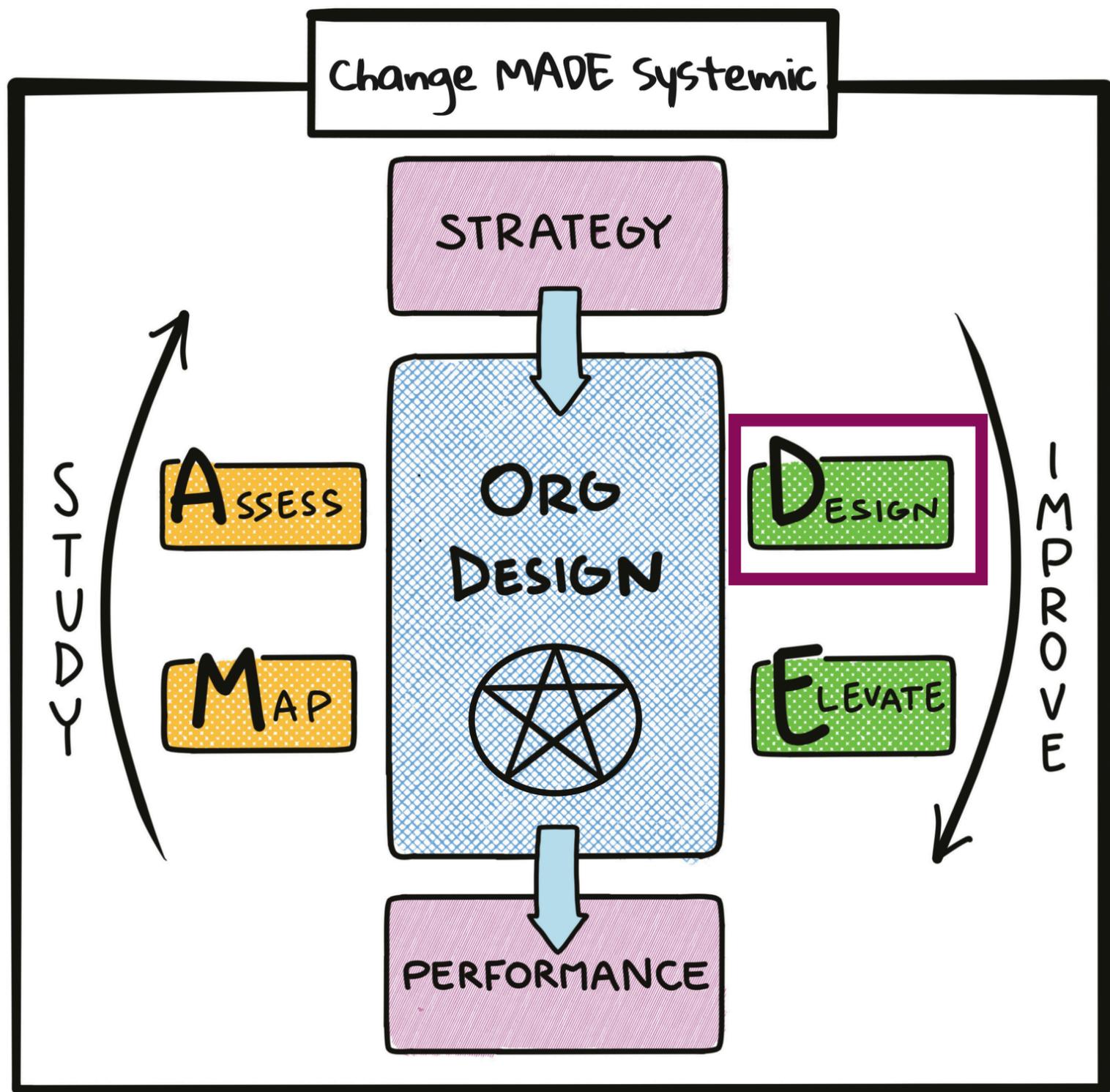
*(focusing just on Strobbo for a moment)*

Teams feel oversized

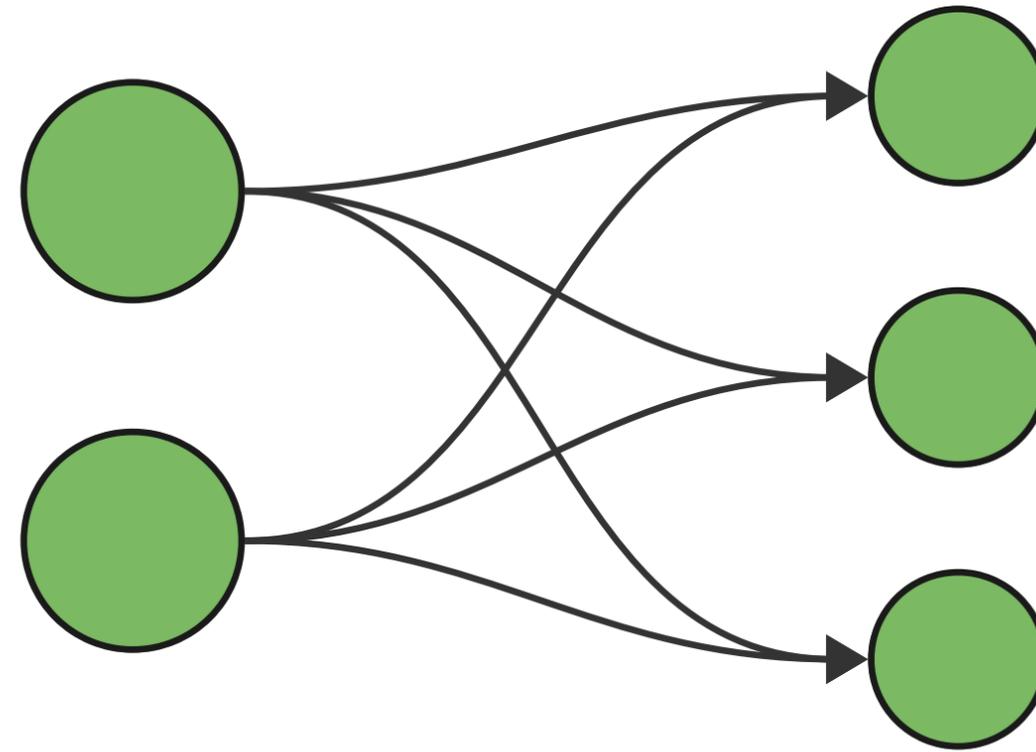
Delivering more value **and at the same time** reducing tech debt

Goals have become smaller and better split

**Hypothesis**: *smaller teams would be a better fit*



Strobbo  
Team of  
Teams



**Experiment:** *Switching from 2 larger to 3 smaller teams*

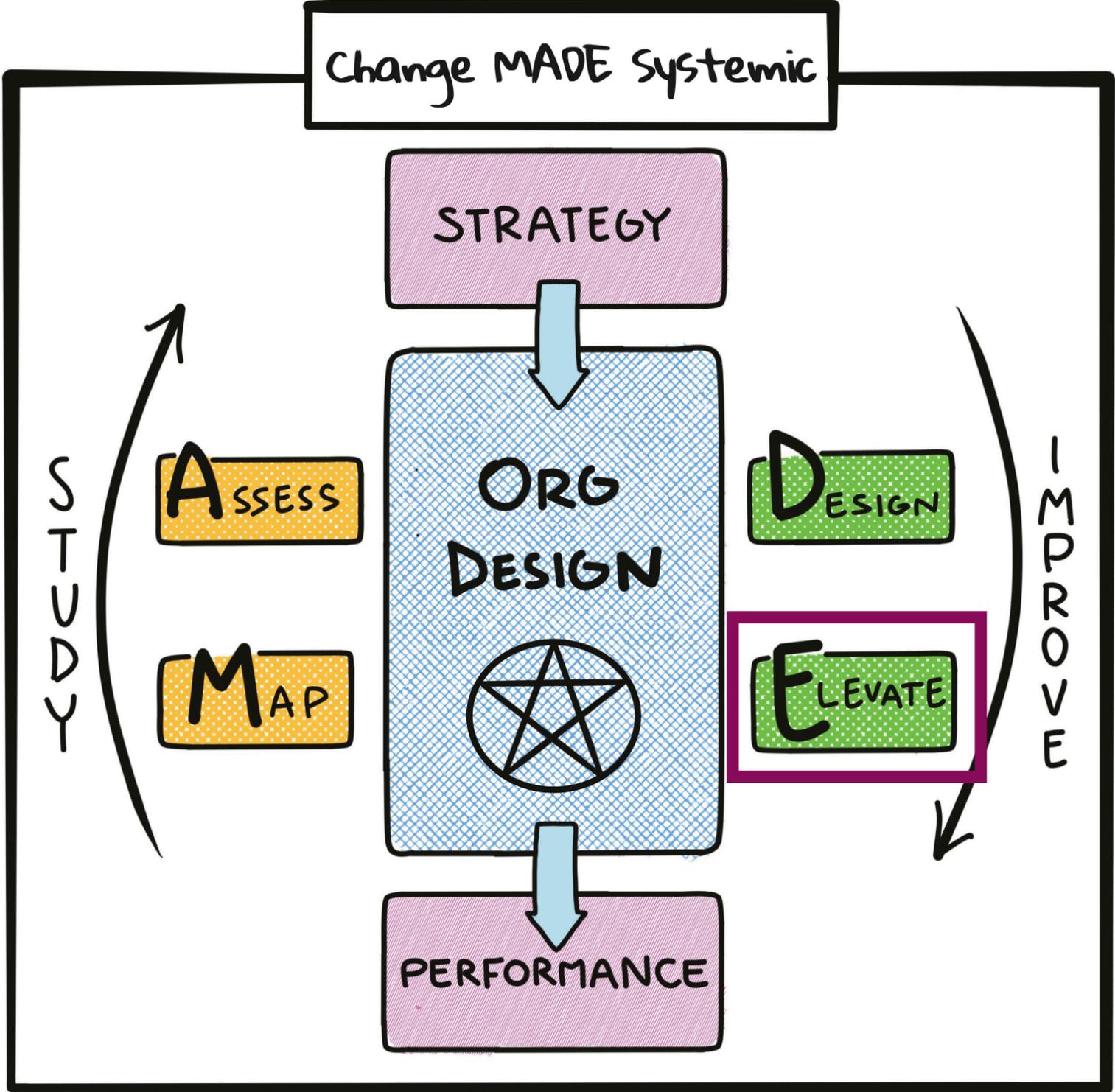
**Goal:** *each smaller team is as effective as the larger teams*

**Positive outcome:** *deliver more value with the same amount of people*

Small change on the map, real impact

More flexibility: **1–3 goals at once**

Needs **ready-to-pick goals**



More conscious approach to requests: frame the problem before shaping solutions

Rapid shaping with fast feedback using sketches, prototypes (Lovable), and agentic POCs

Validate ideas early before committing to build

Evolve refinement to support parallel topics and ensure ready-to-pick work

Move toward agentic delivery beyond small coding tasks

**Is this going to work?**

**Is this going to work?**

We don't know.

**Is this going to work?**

We don't know.

But we are confident.

- **Output / throughput** (more tickets, more story points, more releases)
- **Predictability** (hitting dates, low variance, fewer surprises)
- **Local efficiency** (utilization, minimizing context switching, “stay in your lane”)
- **Quality** (low defects, reliability, clean architecture)
- **Risk reduction / compliance** (controls, approvals, auditability)
- **Cost** (run cost, cloud spend, vendor spend)
- **Speed to learn** (fast feedback, experiments, short cycles)
- **Customer outcomes** (retention, conversion, NPS, reduced time-to-value)
- **Alignment / coordination** (fewer conflicts across teams, smoother dependencies)
- **Developer experience** (tooling, cognitive load, onboarding time)

- **Output / throughput** (more tickets, more story points, more releases)
- **Predictability** (hitting dates, low variance, fewer surprises)
- **Local efficiency** (utilization, minimizing context switching, “stay in your lane”)
- **Quality** (low defects, reliability, clean architecture)
- **Risk reduction / compliance** (controls, approvals, auditability)
- **Cost** (run cost, cloud spend, vendor spend)
- **Speed to learn** (fast feedback, experiments, short cycles)
- **Customer outcomes** (retention, conversion, NPS, reduced time-to-value)
- **Alignment / coordination** (fewer conflicts across teams, smoother dependencies)
- **Developer experience** (tooling, cognitive load, onboarding time)

# **Personal conclusions**

PM's and PO's should care about org design  
*Actually everyone should care about org design*

It defines your ability to create customer value

Outcomes require the right system, not more effort

Work on finding structural fixes, not local optimizations

Think about what you want to optimize for, then dig deeper

You are never done with this

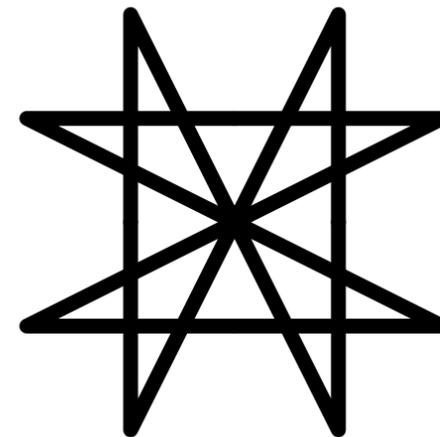
Work on your leadership skills



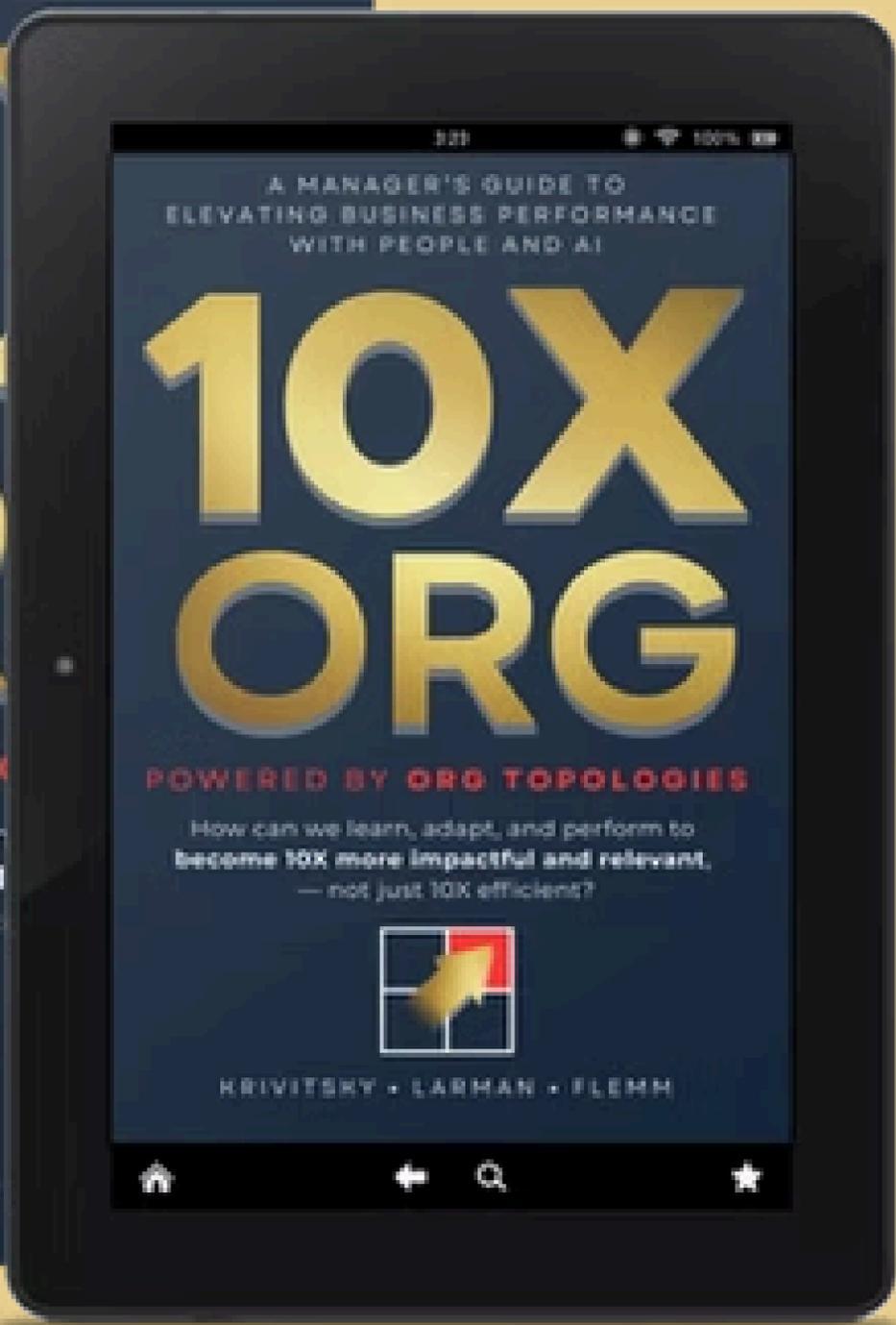
LinkedIn



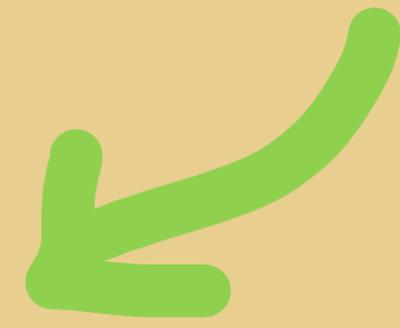
<https://www.linkedin.com/in/michaelvoorhaen>



<https://blog.rooster.solutions/>



strobbo   
use case included



PRESENTED BY ORG TOPOLOGIES

# Org Design Summit 2026

How do we lead the agentic shift  
with people and AI?



**Thank you!**