

**DASHLANE**

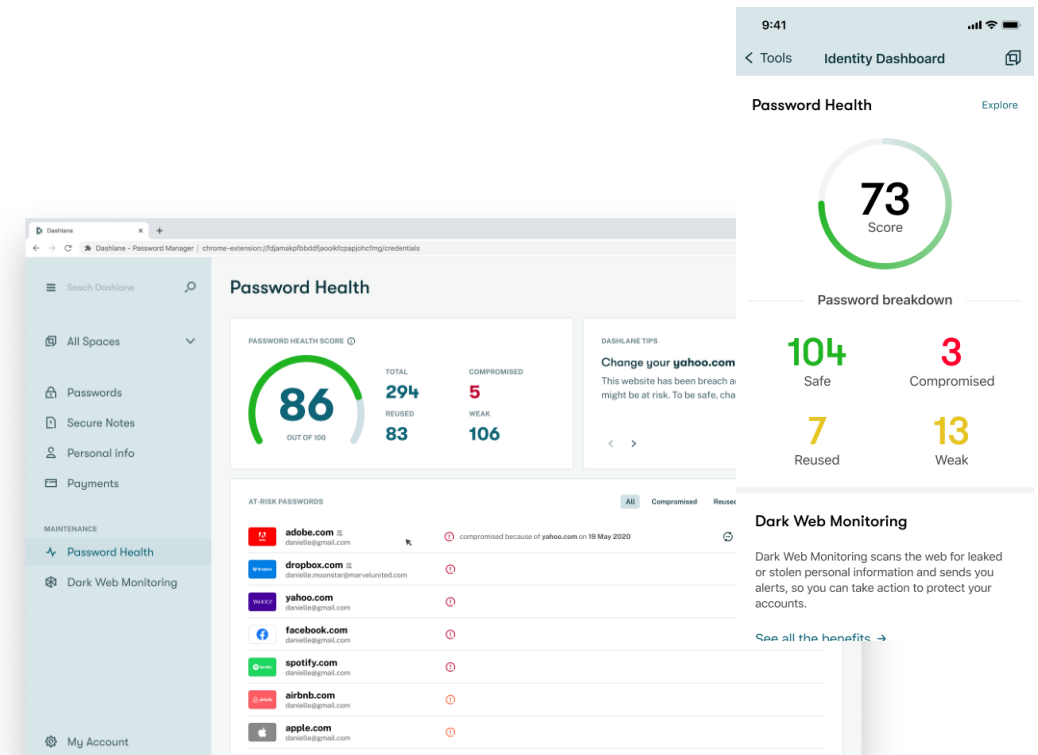
# The Dashlane Triple Track

*Looking for the right Organization*



# Frédéric RIVAIN CTO of Dashlane

We build a Password Manager to help you  
**manage your identity and payments in a simple  
and secure way everywhere**



## A bit of context

Founded in 2009 by **Bernard Liautaud** and 3 French students from Ecole Centrale

~250 employees in Paris, Lisbon and New York

Consumer product (B2C) + Enterprise offer (B2B)

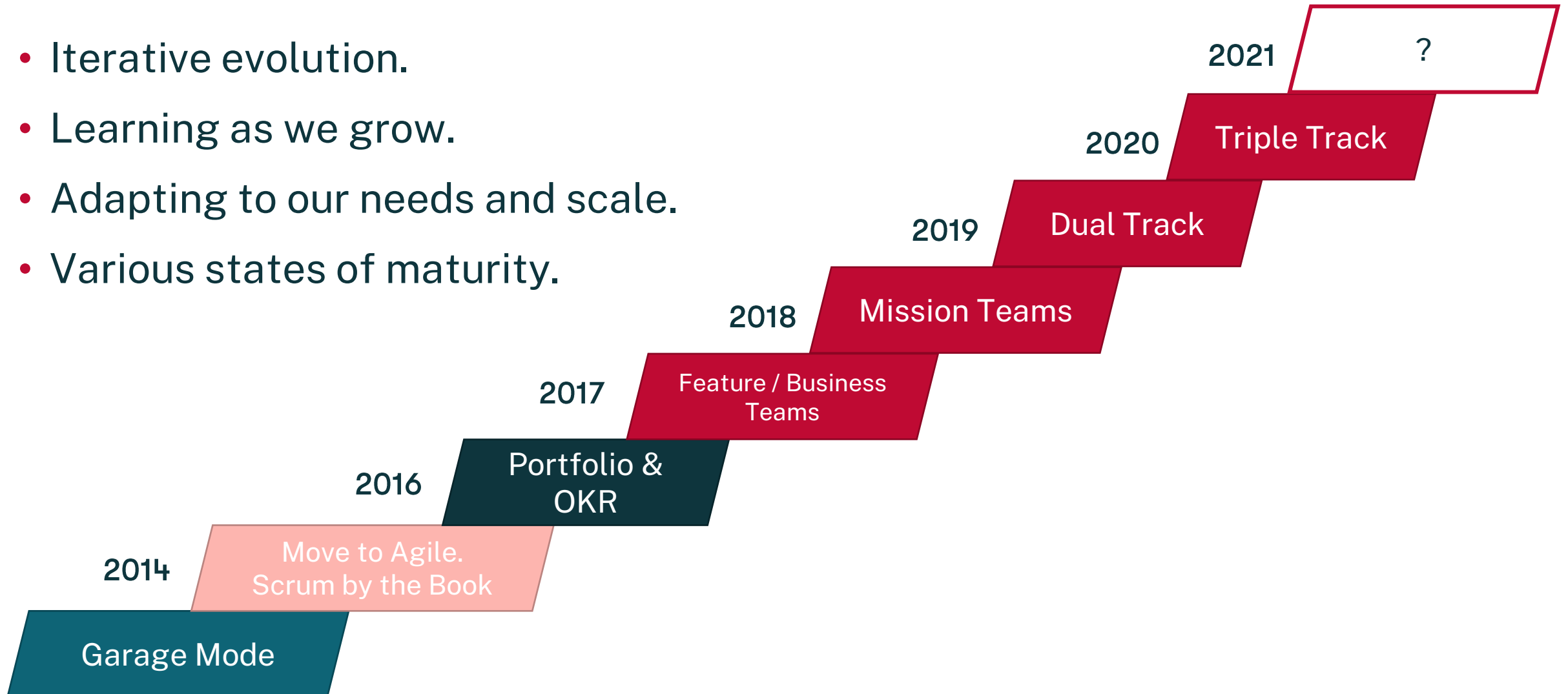
~15 “product & engineering” teams

100+ Engineering Team

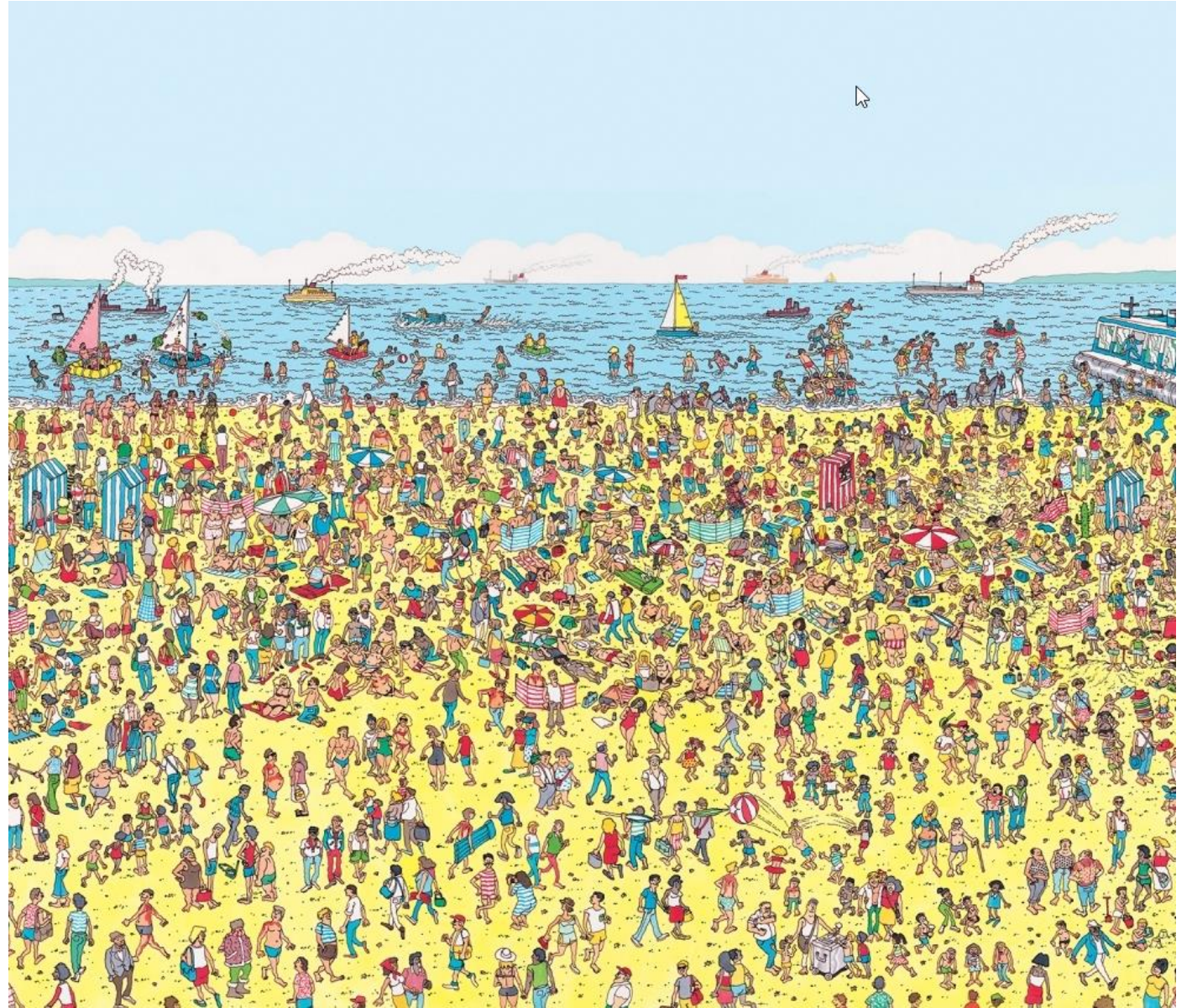


# Our Agile Story

- Iterative evolution.
- Learning as we grow.
- Adapting to our needs and scale.
- Various states of maturity.



# Looking for the right organization





# Half-Baked Agility\*



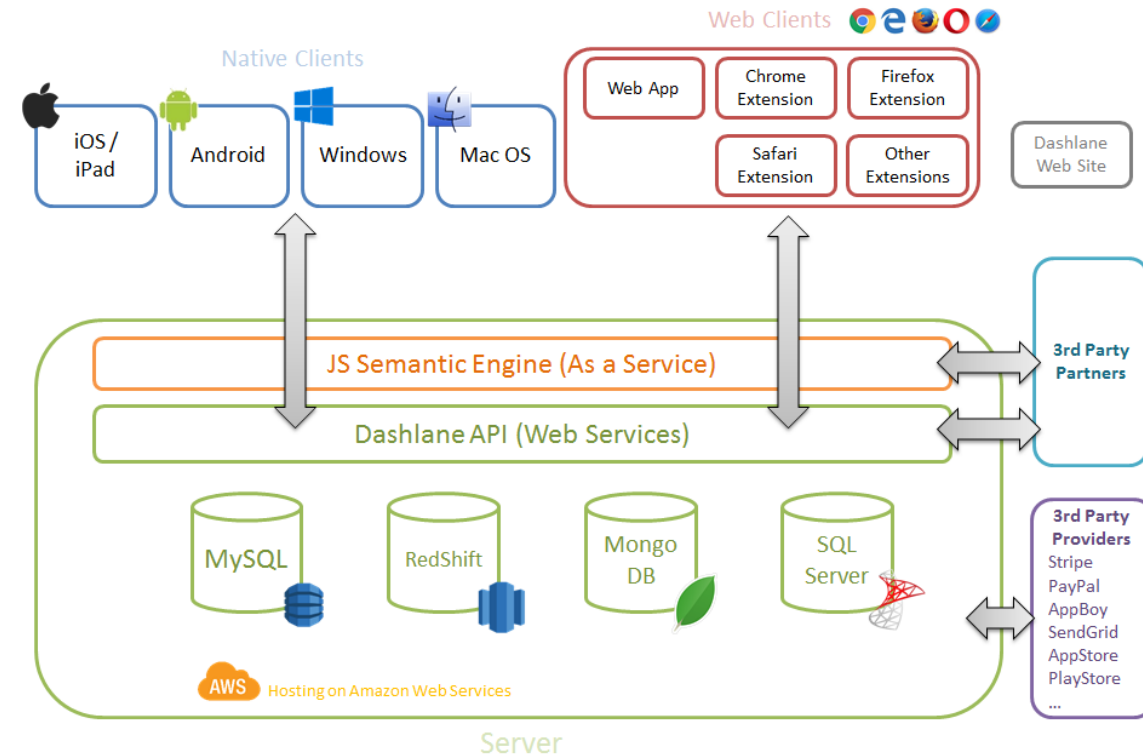
\* As quoted from [Felipe Castro](#)

# Half-Baked Agility\*



# Legacy Platform Teams

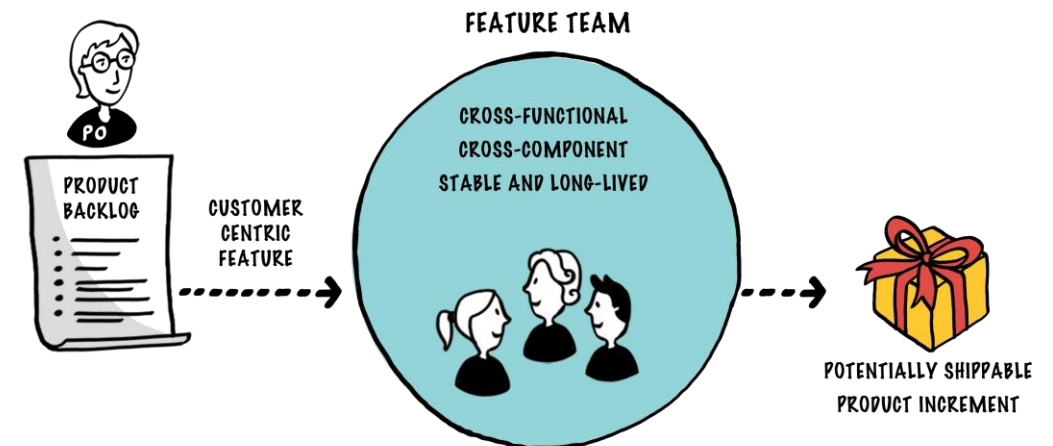
- Originally, platform tech teams:
  - Desktop, iOS, Android, Web, Server, Semantic Engine
- Works well for small teams. With one line of business.
- Starts hurting as you grow the team and as you diversify:
  - Synchronization issues between platforms
  - Inconsistency in product
- Technical investment and Business work mixed within platform teams



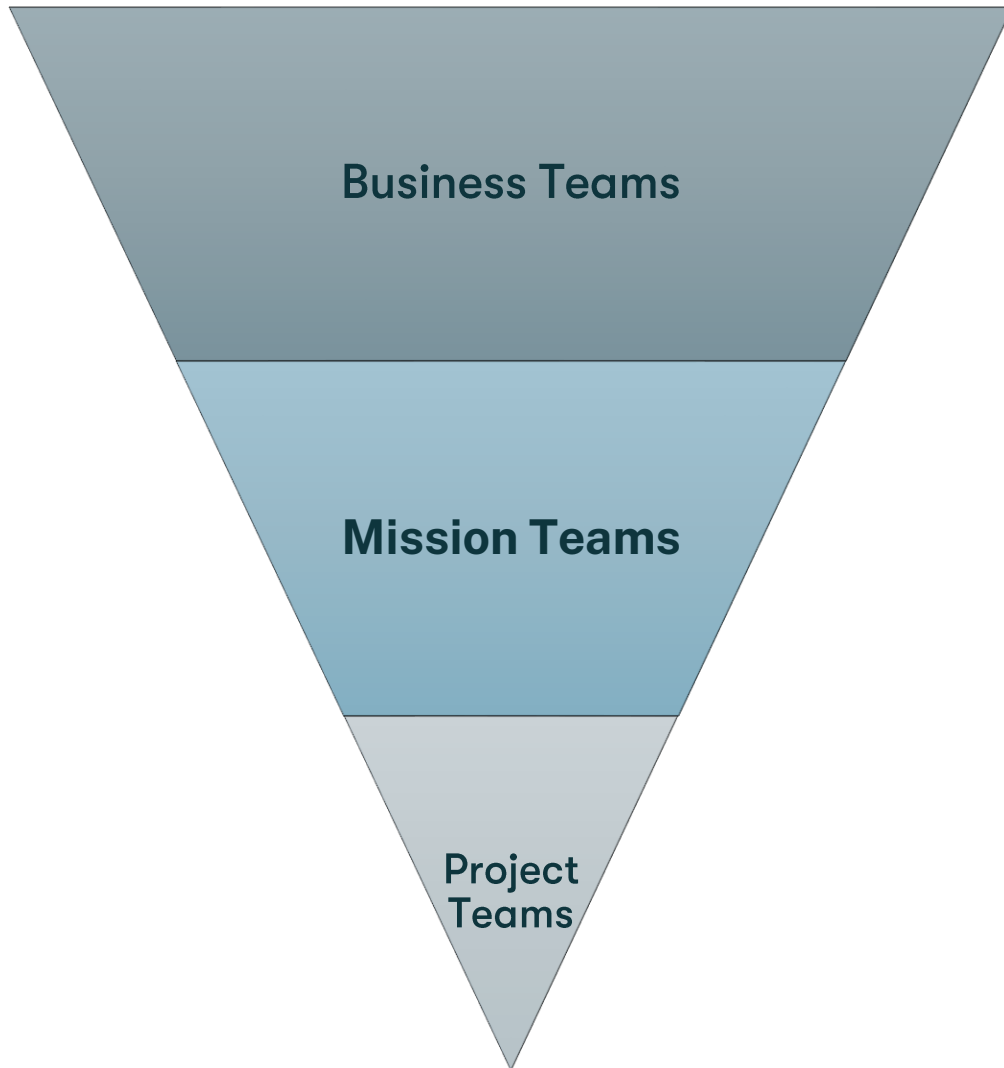


# Transitioning to Business Teams

- Inspired by the Feature Teams model (a la [Spotify](#))
- Cross-functional teams including:
  - ➔ Product, Development, QA + Design, Analytics, Product Marketing, User Support
- « Mini Startup » inside the company, with end-to-end responsibility on their scope.
- Business focus
  - Acquisition
  - Conversion
  - Retention
  - 2 focused on B2B
  - 1 for Partnerships
  - 1 for our semantic engine



# Transitioning again to Mission Teams



## **“Increase retention”**

Too many ideas, no filtering lens for Product  
No clear sense of when to stop and do something else  
Lagging indicator-focused

## **“Get more passwords”**

Lots of room for creativity within a boundary  
Success is clear  
Leading and lagging indicators

## **“Build feature X”**

No room for ideas  
Success is delivery not results  
Leading indicator-focused

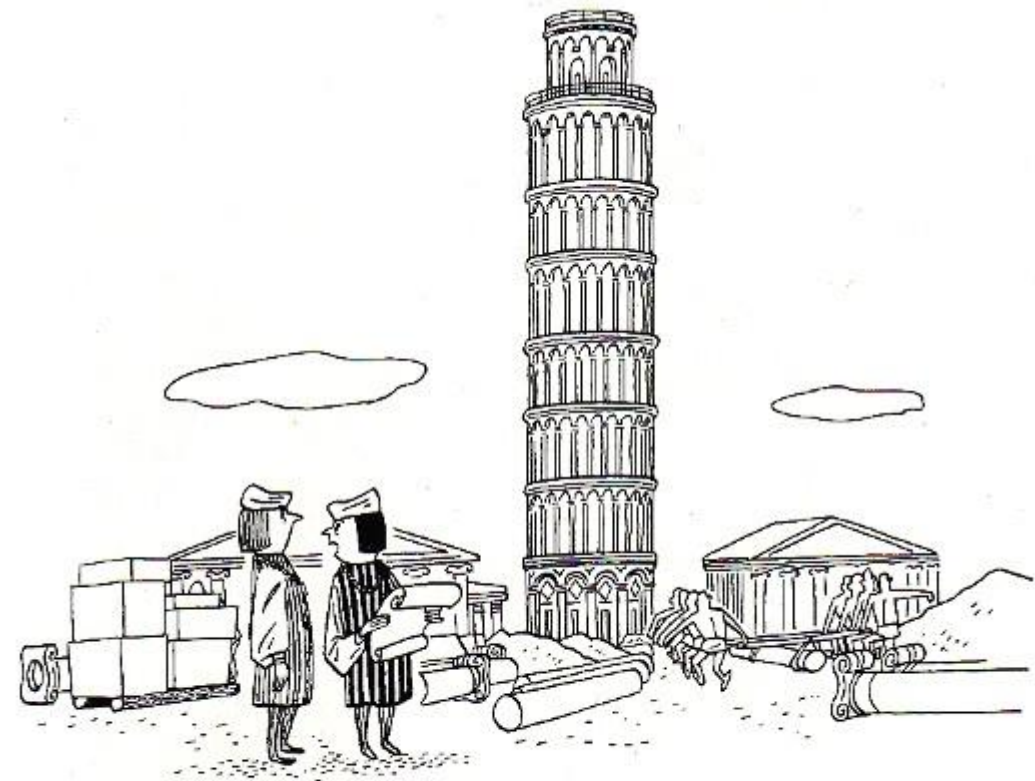
# Cross-functional Teams

- Cross-platform teams, with dedicated resources and skills, based on Missions
- Small teams of 1 Product Manager, 2 to 6 engineers, 1 UX designer, 1 QA.
- A double organization:
  1. **Mission Teams**
  2. « Platforms » communities of practice

	Mission Team 1	Mission Team 2	Mission Team 3	...	Mission Team N
Product Manager	x	x	x		x
Scrum Master	x	x	x		x
QA	x	x	x		x
Server	x	x			
iOS	x	xx			
Android	x	xx			
Windows	x		xx		
Web	x		xxx		
UX Design	x	x	x		x
Analytics	x	x	x		x
User Support	x	x	x		x

PLATFORMS

# Half baked technical investment



Wilkinson

This Week

**"I skimmed a little on the foundation, but  
no one'll ever know it."**

# Managing Technical investment poorly

- Testing different approaches for technical investment
  - 10 % of the sprint
  - First 2 days each sprint (20%)
  - 1 week after each sprint (33%)
- Platform communities come back together to work on their platform



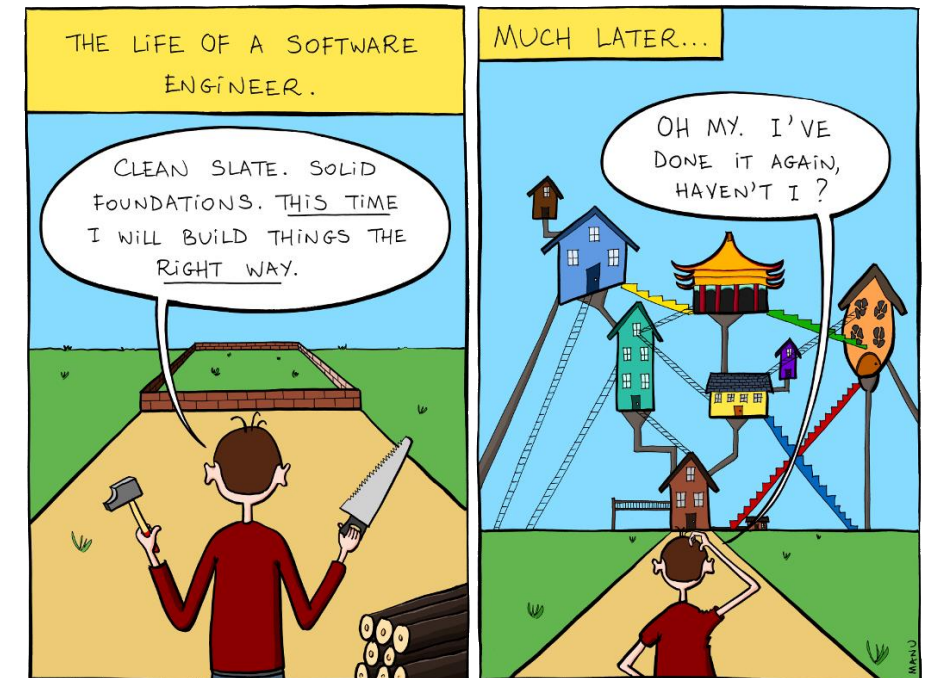
# Technical investment

## → Pros:

- Easier prioritization
- Help tightening the platform communities

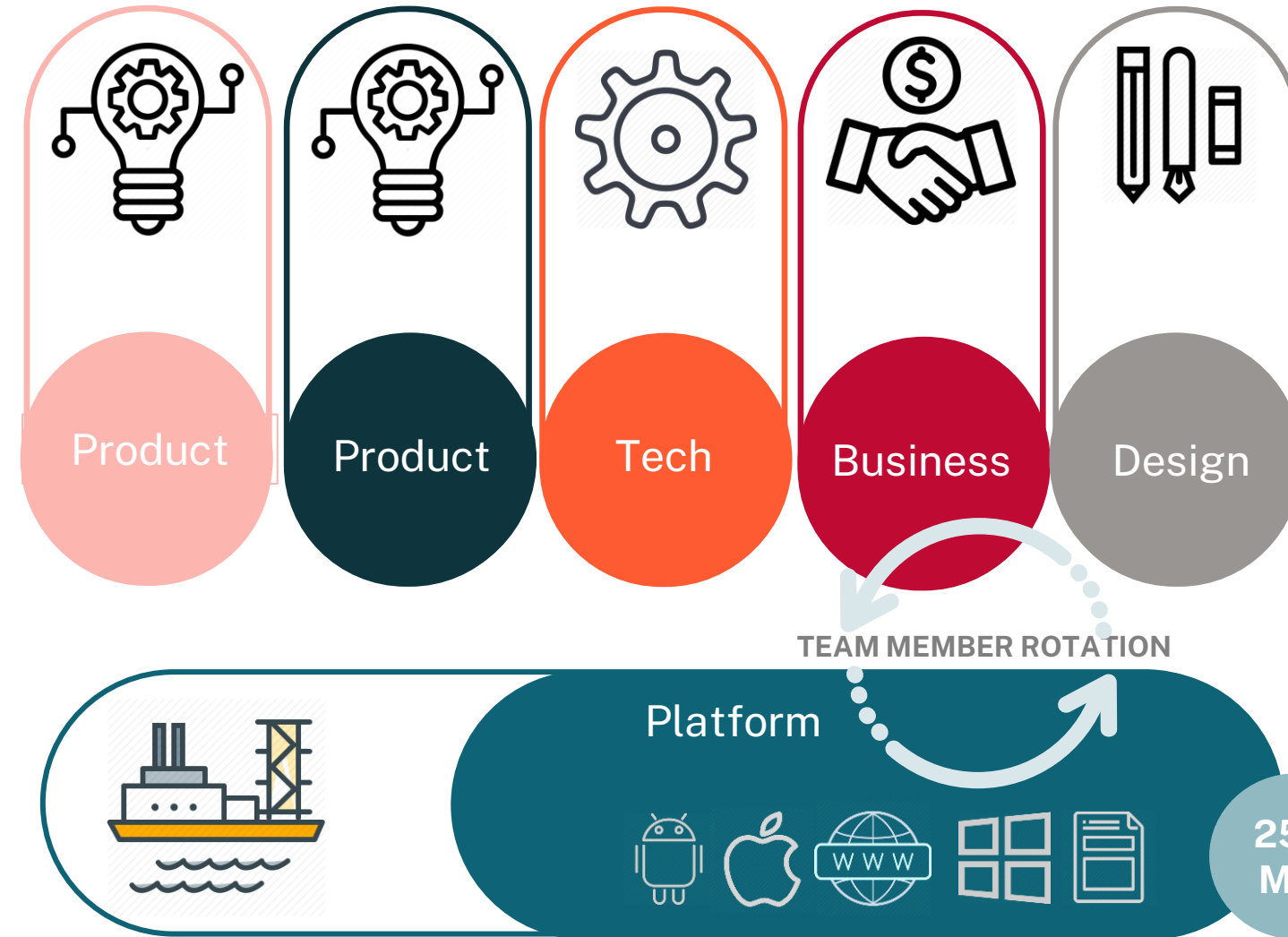
## → Cons:

- Constant context switching
- Difficult to work on long term technical projects
- Both ways overflow





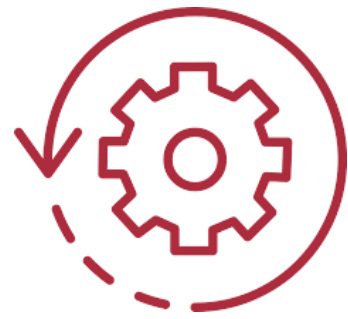
# Dual Track Organization for Mission Teams & Platforms



- **Minimum of 25%** of total platform resource allocation
- Team member **rotation**



# How do Platform Squads work?



- Platform Squads are similar to normal mission teams
- **High autonomy** regarding the team processes: sprints, kanban
- Bi-weekly reviews
- The **product owner is a Tech Lead**
- They are in charge of:
  - production monitoring
  - release coordination
  - refactors, optimizations, platform bug-fixing...
  - Tooling and cross-platform tech projects
  - communication points for the rest of the organization (User Support, Marketing,...) - Team Captain role

# Platform Strategy

- Aligned to a yearly Engineering Strategy and OKRs
- Quarterly technical roadmaps
  - Built by the teams
  - Must contain technical OKRs
  - Presented to the executive team
- Evaluate for impact

## Q3 2019 Platform Roadmaps

Created by Guillaume Maron, last modified just a moment ago

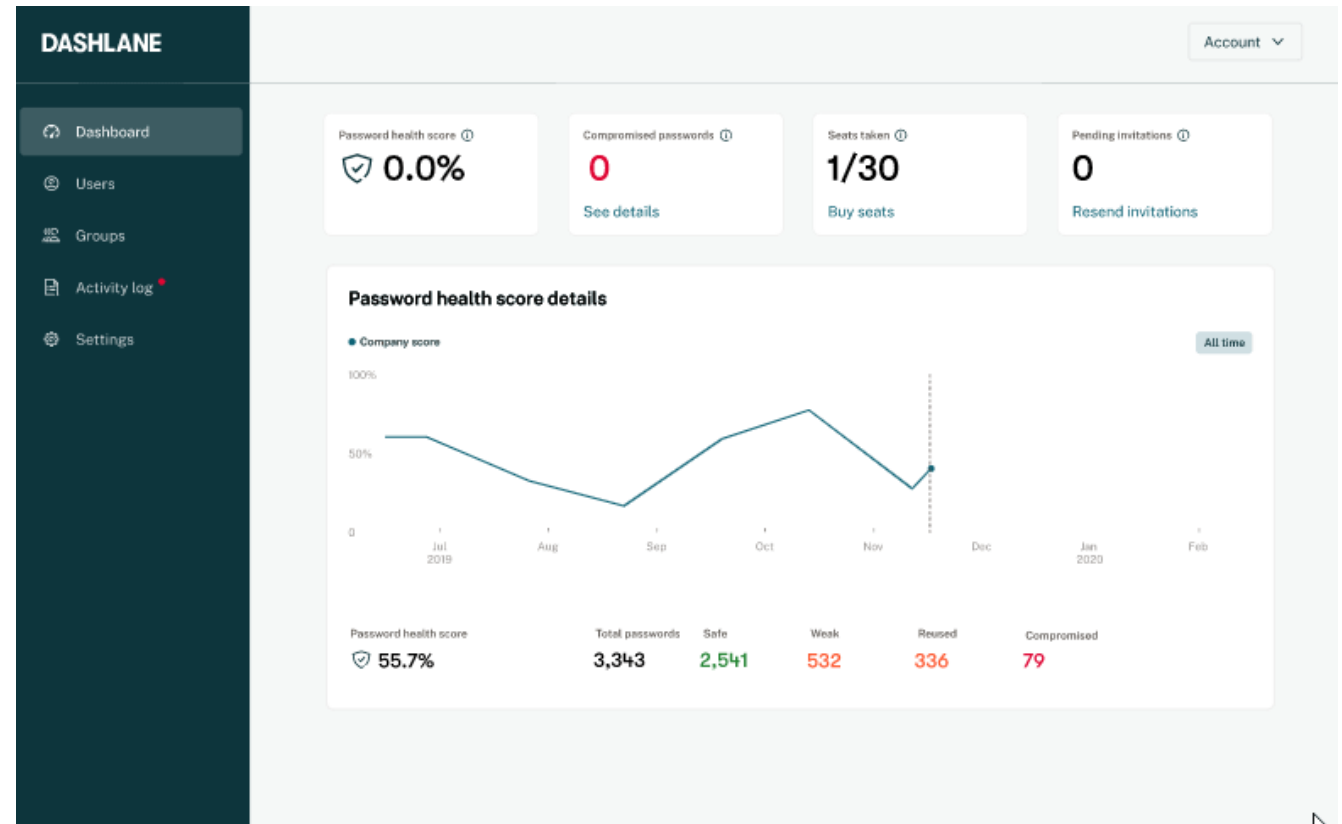
### Android

Main Quarterly Goal: Increasing app's performance		
Area of focus	Application performance	CI
KPI	<ul style="list-style-type: none"><li>Reducing RAM usage by 20% <b>FAILED</b></li><li>Reducing number of crashes by 30% <b>DONE</b></li></ul>	
Actions	<ul style="list-style-type: none"><li>Switch from lib XYZ to lib ABC <b>FAILED</b></li><li>Investigate top 10 crashes . <b>DONE</b></li></ul>	<ul style="list-style-type: none"><li>Integrate tool XYZ into the release pipeline</li></ul>

### iOS

Main Quarterly Goal: ...
--------------------------

# Feature Ownership



# Core Teams: feature ownership and quality



- Long-living (1+ year of existence).
- Cross-functional staffing
- Own a Core Area of the Dashlane **feature set**
- Responsible for the quality of their feature area (bug-fixing, performance,...)
- Deliver new features and solutions in that area. They define customer goals and metrics they want to achieve.

## Core Areas

IT Admin

Sharing & Collaboration

Autofill Engine & Experience

Auth & Sync

Protect the User

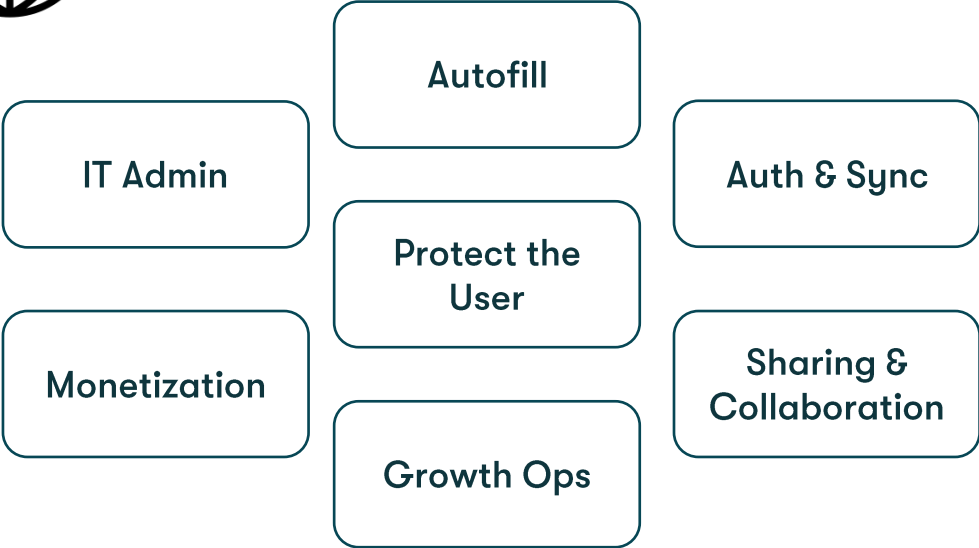
Growth Ops

Monetization

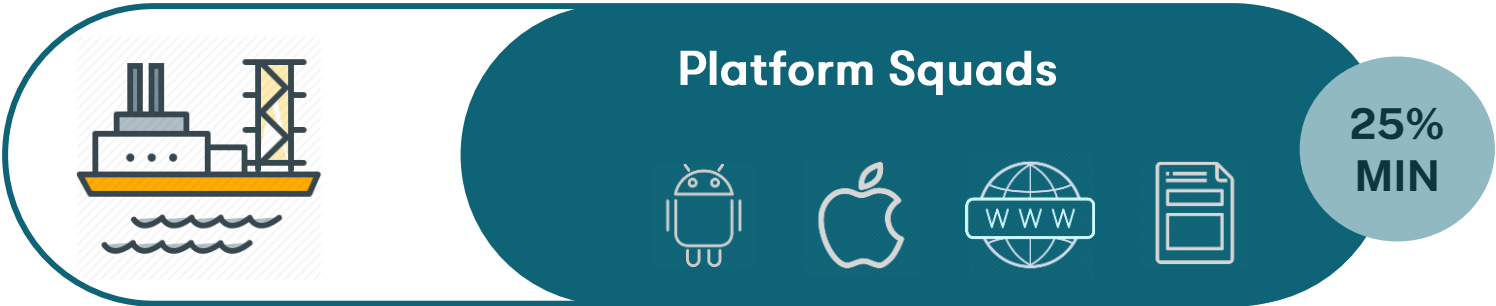
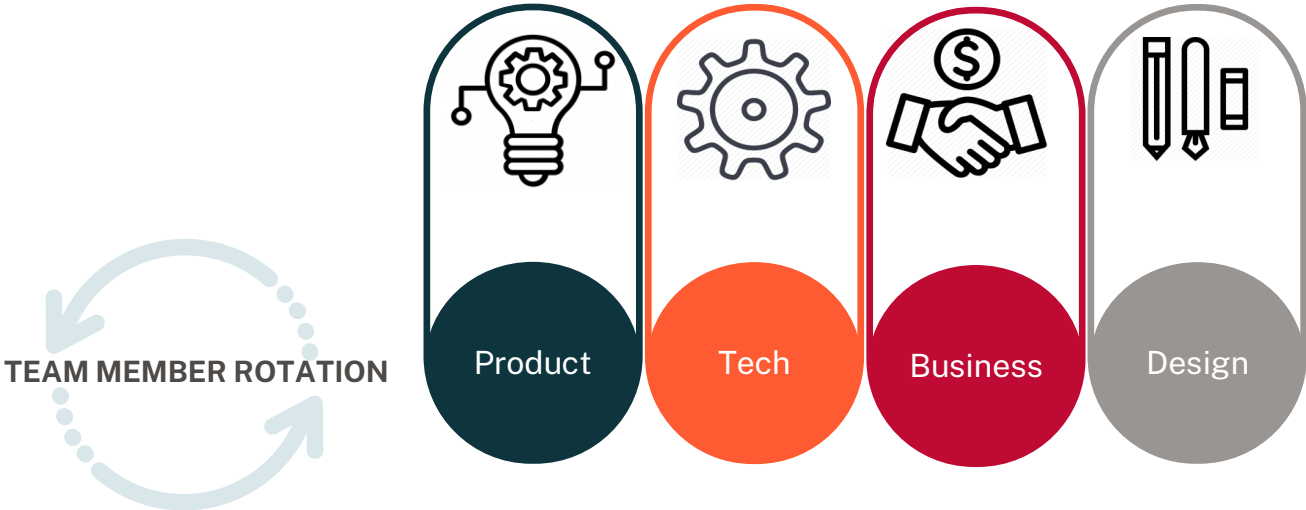
# Triple Track Organization



## Core Teams

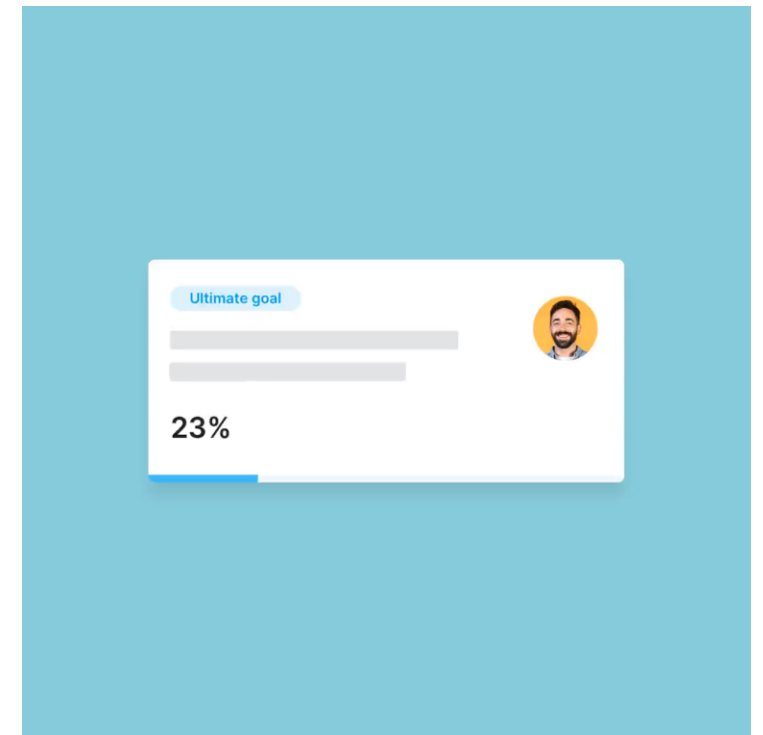
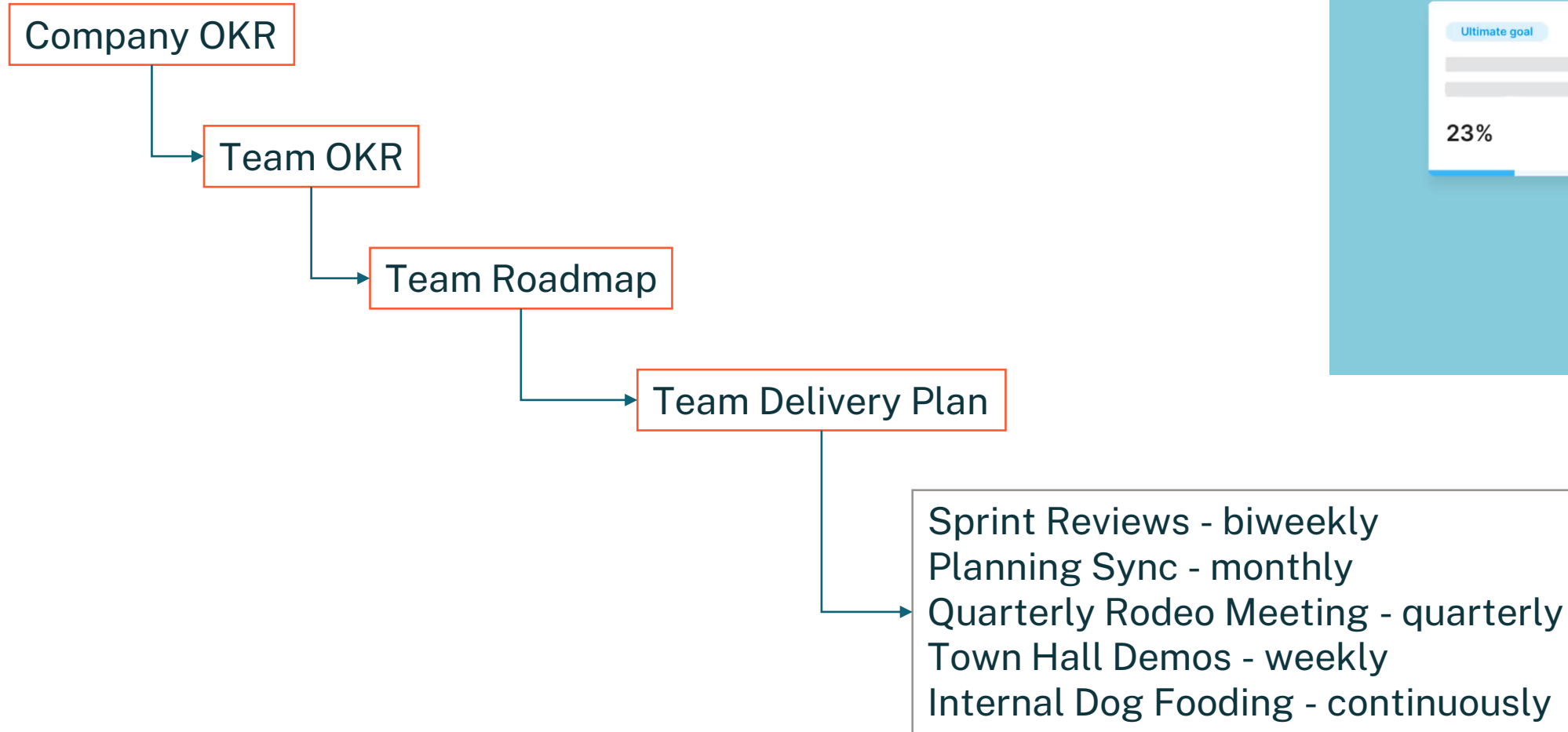


## Mission Teams





# Alignment & Synchronization



# Early Learnings

## Pros

- Clearer, more focused organization
- Permanent tech investment, allows for longer-term projects
- Identified people supporting the dev community and ecosystem (with tooling, release capability, production monitoring)
- Forcing function to never drop the ball on technical needs

## Cons

- Need to be a certain team size
- Risk of under-staffing (remains a staffing challenge). Avoid spreading yourself too thin
- Tendency of silo. So important to rotate people.

## 4 Key Take Aways

1. **Find the right mix between** tech and business
2. **Focus** and avoid context-switching
3. **Experiment** all the time with your organization. Aim for **learning**.
4. **Assess for impact**, not for delivery.



*Get 6 months premium and try  
Dashlane  
with the code **BUSINESSAGILITY2021***

