Thank you for your patience.
The webinar will begin shortly.

Host: Julian Meissner
Manager, Tech & Transformation, Robert Walters Thailand
- Joined Robert Walters Thailand in 2016
- He brings with him more than three years of recruitment experience from Brussels where he focused on senior engineering management profiles
- Prior to joining Robert Walters, he was working for a tech startup which he led the product roadmap, project management and enterprise account management

Speaker: Paul Knapp
Chief Technology Officer, Kaidee
- 25 years of expertise working in technology across Asia, Australia, and Europe
- Has extensive knowledge in software engineering
- Prior to taking on the Chief Technology Officer role, he was Kaidee's Director of Engineering, where he worked on a highly scalable microservice architecture

ROBERT WALTERS

ROBERT WALTERS

Better products with cross-functional teams

Who we are

Kaidee is Thailand's biggest online marketplace.

Anyone can use our service to sell online, both new and used items.

The platform is available on mobile apps and the web. It makes it easy for anyone with a smartphone to buy and sell online.

Kaidee has more than 264 categories, and you can post and sell for free.

We want to improve people's lives by bringing buyers and sellers together.

Kaidee in numbers

➤ **30 million** Thais used Kaidee during 2019.

➤ **280 million** total visits with a monthly average of **24 million**.

➤ **1.85 million** items were sold valued at **100 billion baht** (**US$3.2 billion**).

➤ **15 million** downloads of our phone apps.

➤ **800,000** cars listed for sale.

➤ **400,000** motorcycles listed for sale.

➤ **700,000** properties listed for sale.

➤ **3 million** keyword searches a day.

Paul Knapp - Chief Technology Officer

➤ 25 years experience working in Technology

➤ Worked in Thailand, Australia, UK, Singapore and other countries in Asia.

➤ Software Engineering background.

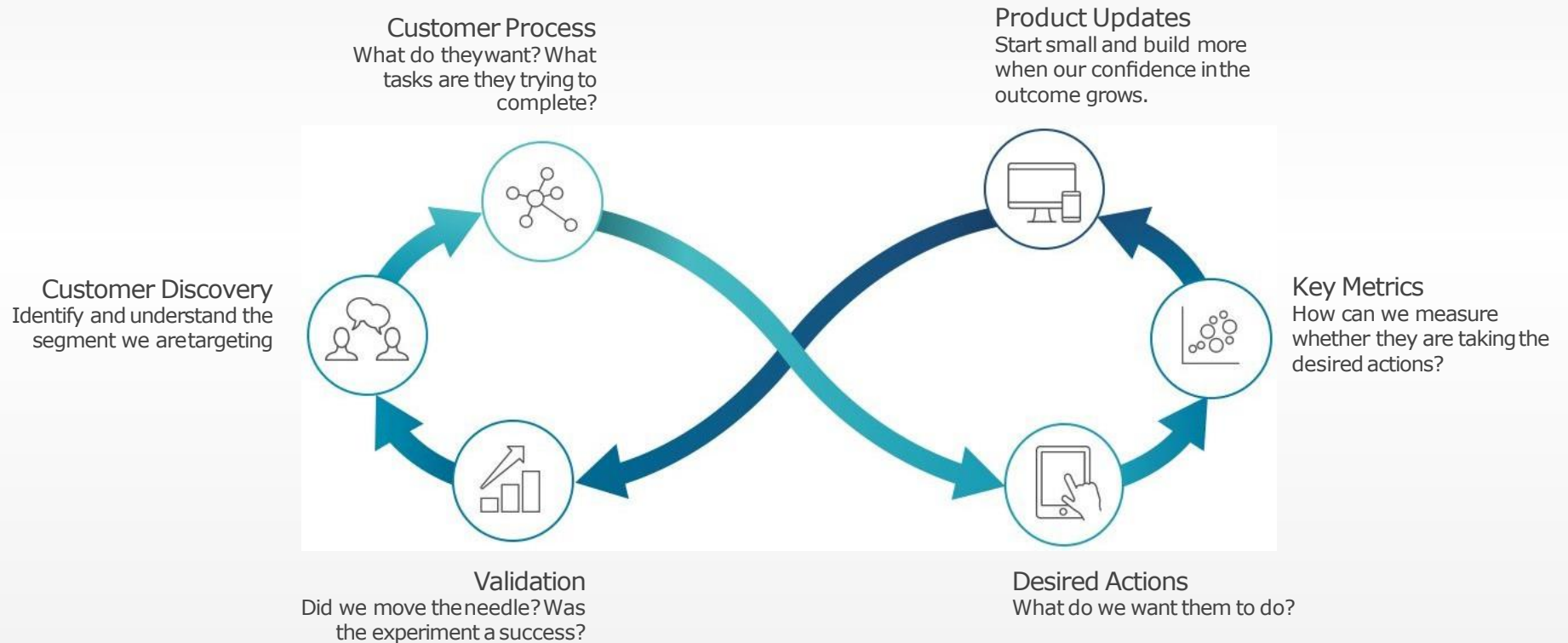➤ Managing 45 people across product, CX, data and engineers.

Allianz ⑪   Telstra
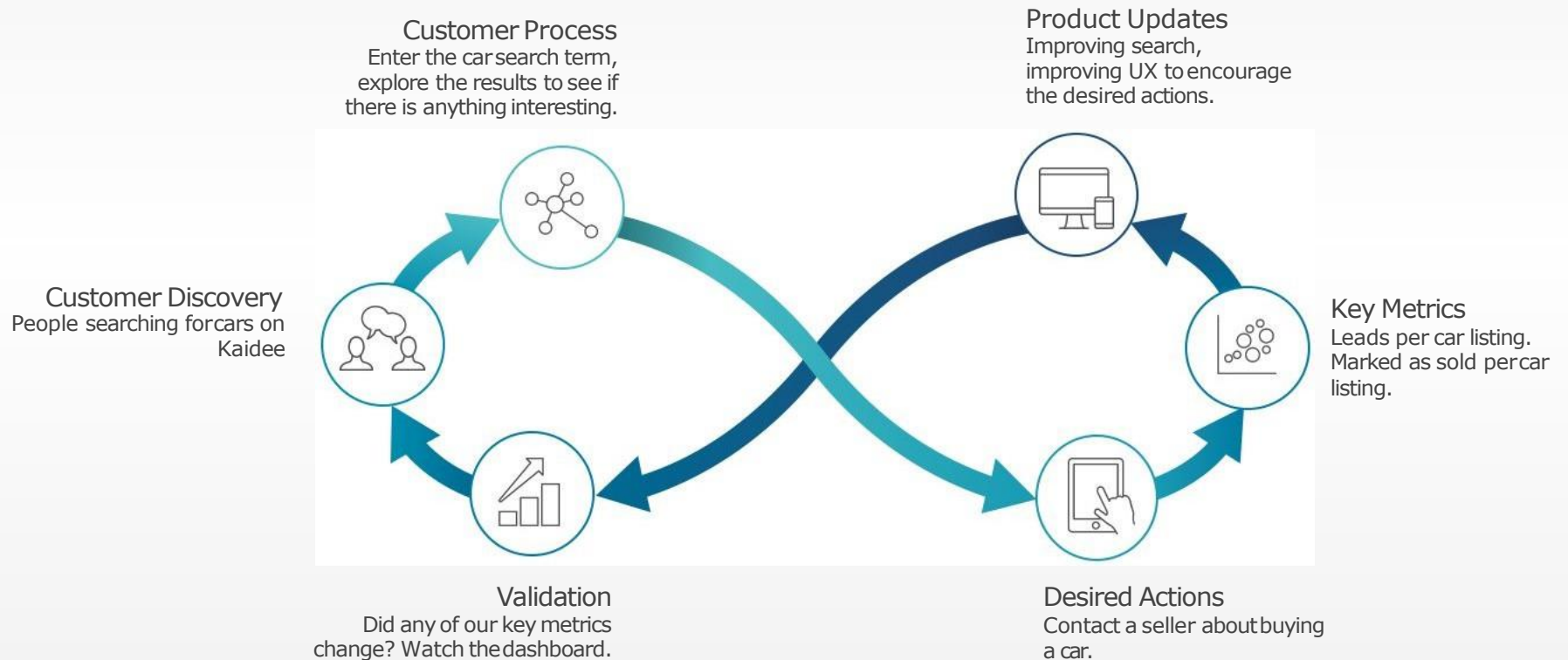
Westpac   citi

Deutsche Bank   accenture

Products, not projects

No end point, continuously improving our product forever

Customer Process
What do they want? What tasks are they trying to complete?

Product Updates
Start small and build more when our confidence in the outcome grows.

Customer Discovery
Identify and understand the segment we are targeting

Key Metrics
How can we measure whether they are taking the desired actions?

Validation
Did we move the needle? Was the experiment a success?

Desired Actions
What do we want them to do?

ROBERT WALTERS

A simple example

No end point, continuously improving our product forever

Customer Process
Enter the car search term, explore the results to see if there is anything interesting.

Product Updates
Improving search, improving UX to encourage the desired actions.

Customer Discovery
People searching for cars on Kaidee

Key Metrics
Leads per car listing. Marked as sold per car listing.

Validation
Did any of our key metrics change? Watch the dashboard.

Desired Actions
Contact a seller about buying a car.

Question

No end point, continuously improving our product forever

How should we organise our teams to achieve the results we want?

Typical way of organising teams

Everyone in their neat siloed box, sitting separately, working apart.

Business Teams

Product / Business Analysts

Software Engineers

Site Reliability Engineers

Things get thrown over the fence

Work is pushed down to the other team without care

Business Teams

Business/product ideas

Product / Business Analysts

Requirements

Software Engineers

Deployment requests

Site Reliability Engineers

Lots of room for misunderstandings

Little interest in the other team's point of view, needs or constraints

Tech is slow and useless!

These ideas are stupid!

These requirements suck!

This will break my system!

| Business Teams |
| Product / Business Analysts |
| Software Engineers |
| Site Reliability Engineers |

Business/product ideas

Requirements

Deployment requests

It encourages the worst of tribalist instincts

Business Team

Engineering Team

Product Team

A better way: Splitting into feature teams

Cross-functional teams focussed on delivering value to end customers.

Feature Team
Business Development
Marketing
Product
UX/UI
Web Engineers
Mobile Engineers
Back-end Engineers

A permanent team of people working together to deliver value to the end customer of the organisation.

Each team member brings skills for helping deliver value.

Skills can be shared across members. Plenty of scope for cross-learning.

If a skillset is missing, it can be learned by the team, or a new permanent member can be added.
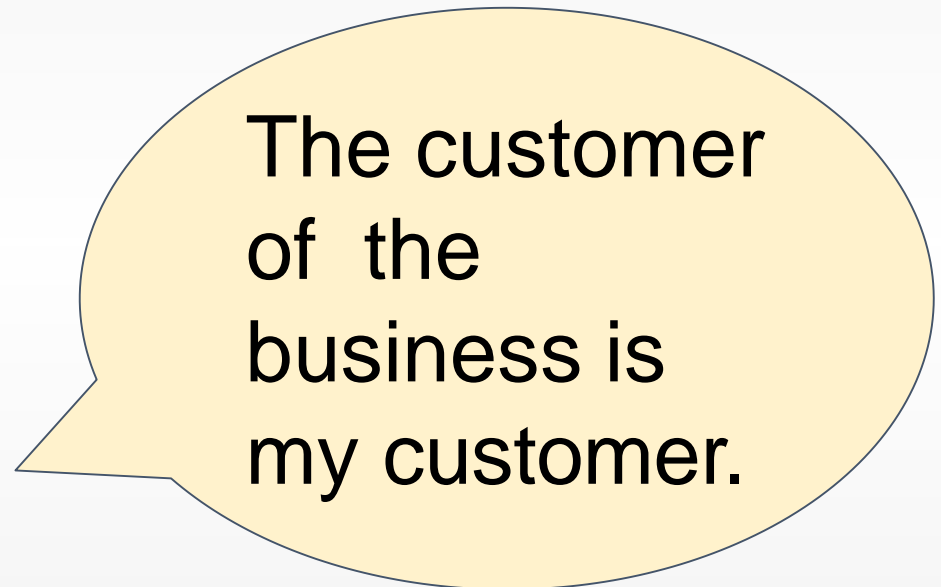
No more throwing problems over the fence.

No more blaming or misunderstanding other teams.

Who is the real customer?

Cross-functional teams focussed on delivering value to end customers.

Feature Team
Business Development
Marketing (still to come)
Product
UX/UI
Web Engineers
Mobile Engineers
Back-end Engineers

The customer of the business is my customer.

Splitting into feature teams

Cross-functional teams focussed on delivering value to end customers.

| Generalist	Cars	Property
Business Development	Business Development	Business Development
Marketing (still to come)	Marketing (still to come)	Marketing (still to come)
Product	Product	Product
UX/UI	UX/UI	UX/UI
Web Engineers	Web Engineers	Web Engineers
Mobile Engineers	Mobile Engineers	Mobile Engineers
Back-end Engineers	Back-end Engineers	Back-end Engineers

Site Reliability Engineers   Data Engineers
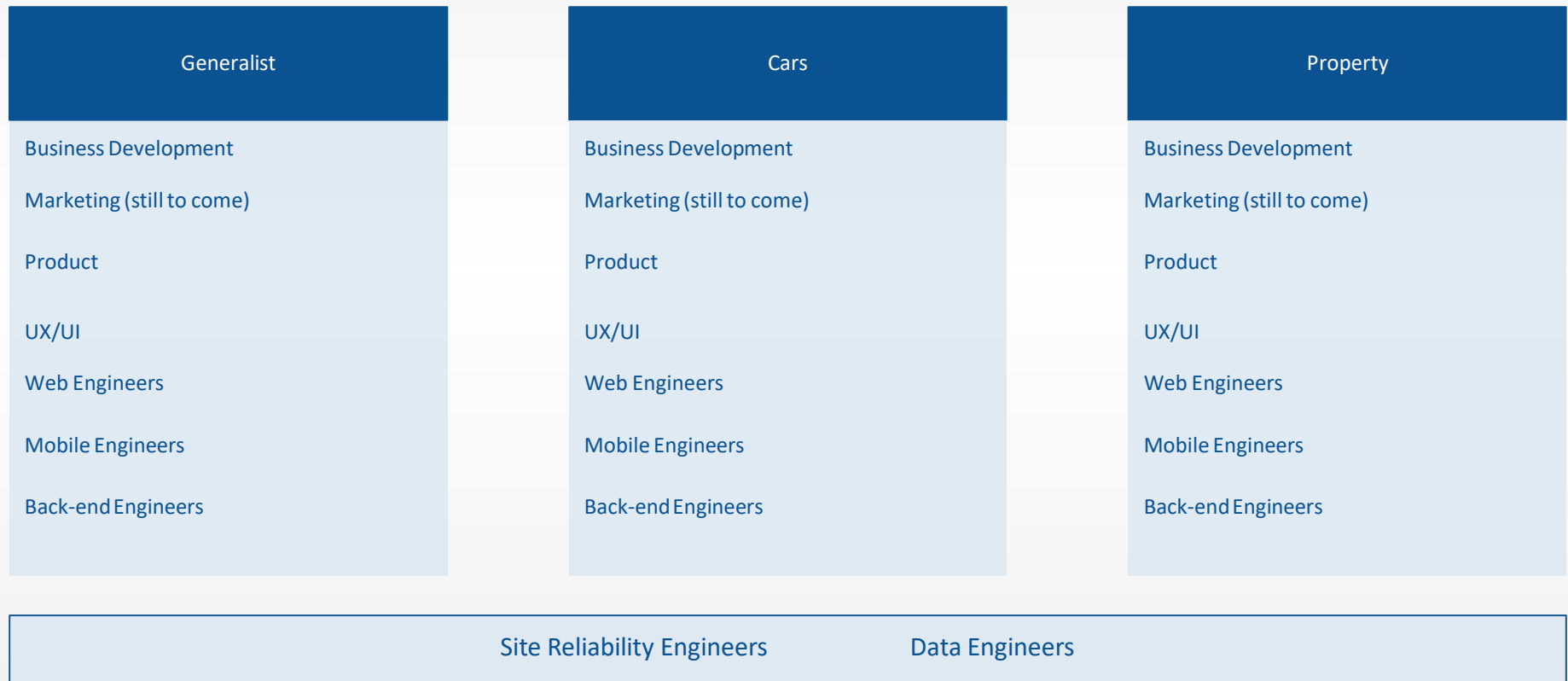
Team management

Functional Leads are for HR issues and mentoring. Otherwise teams should be self-managing and focussed on their end customer. Less functional leads is better.

	Generalist	Cars	Property
BD Functional Lead	Business Development	Business Development	Business Development
	Marketing	Marketing	Marketing
	Product	Product	Product
Product Functional Lead	UX/UI	UX/UI	UX/UI
	Web Engineers	Web Engineers	Web Engineers
	Mobile Engineers	Mobile Engineers	Mobile Engineers
Engineering Functional Lead	Back-end Engineers	Back-end Engineers	Back-end Engineers

Desk Layout

Functional teams sit together so they can work closely

Engineers

UI/UX

Product

Engineers

Marketing

Business Development
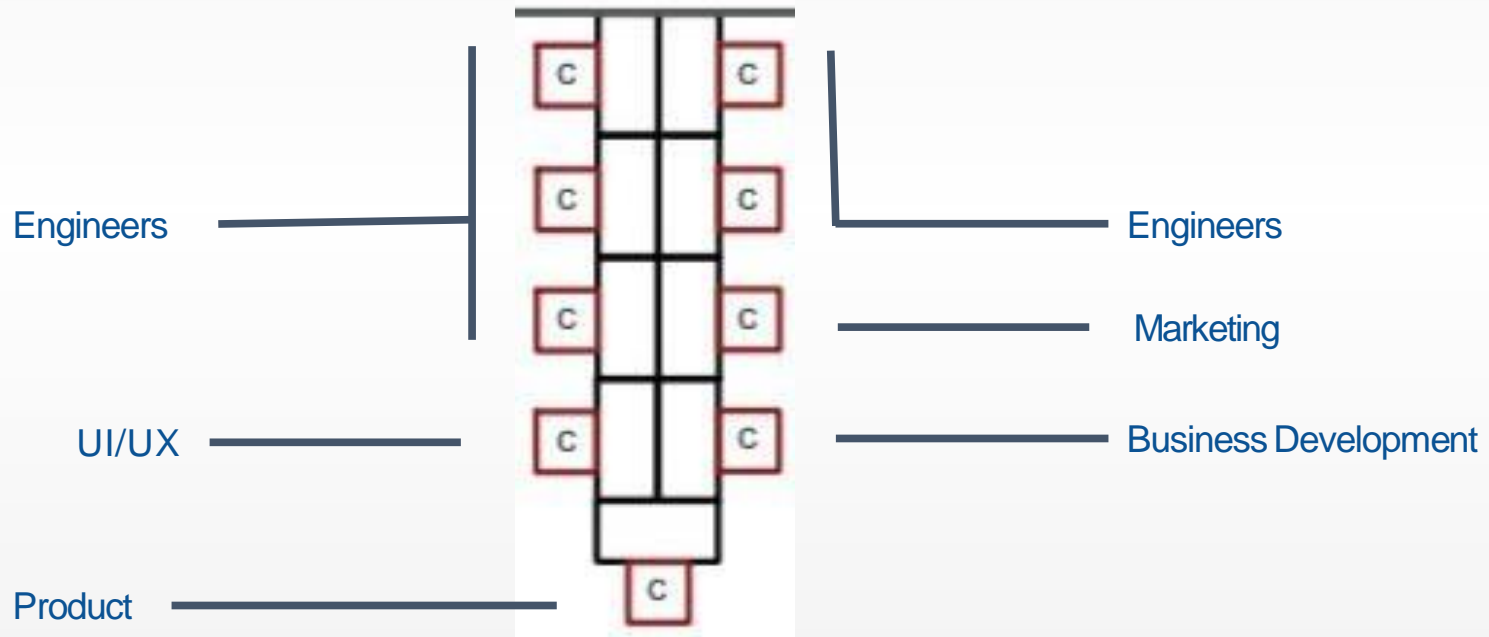
Roadmap Management

## Epics fed into the teams from the Product Roadmap
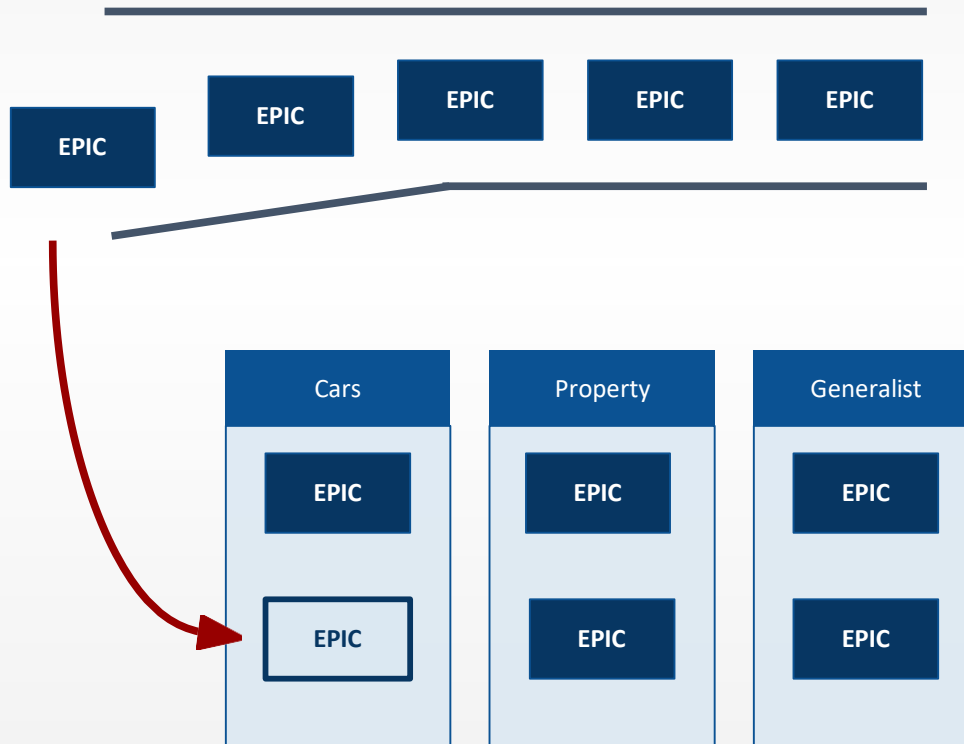
The overall Product Roadmap is managed by the Product and the internal sponsors and fed into the appropriate teams.

Getting a strict definition of which team is working on what is very challenging and can obstruct business goals. Better to be a bit flexible.

Sprint planning

Sprint planning based on the Less framework
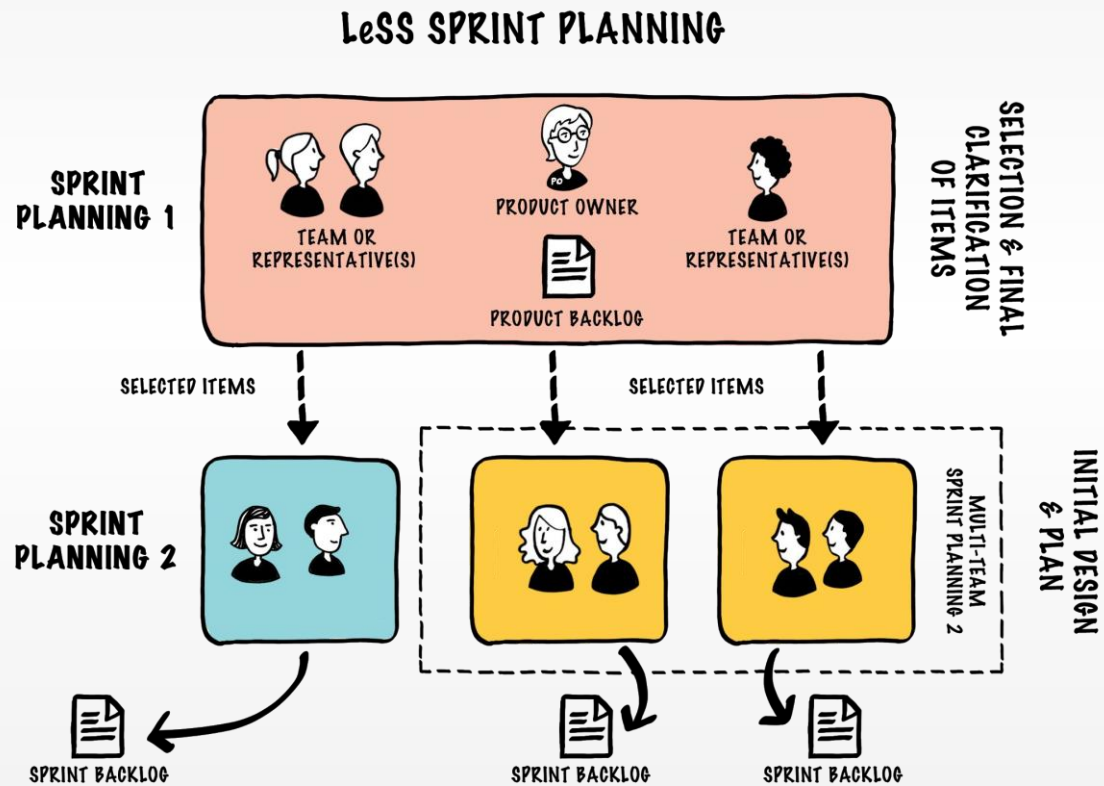
Side meetings

Making sure everyone is aligned

If necessary, side groups with representatives from the teams can break off into small meetings and discuss things that cut across teams. Such as:

➤ How to ensure work on shared components is aligned.

➤ Making sure UI/UX is consistent.
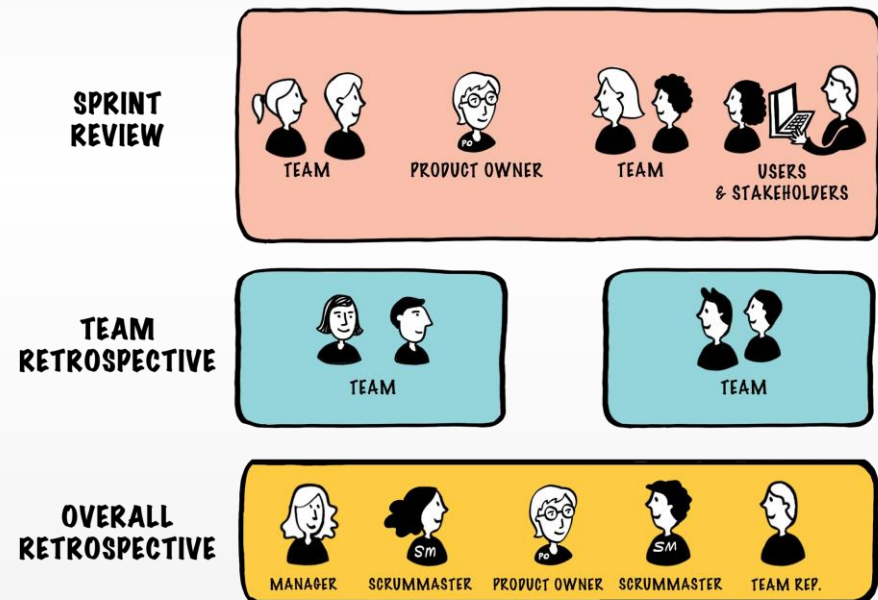
➤ Release schedules that don't conflict.

➤ etc.

Other meetings

Basically just Scrum

Follow standard Scrum events for other meetings. Having an overall retrospective is a good idea though.

➤ Daily stand-ups as normal and per team.

➤ Sprint review as normal, we do it with all teams.

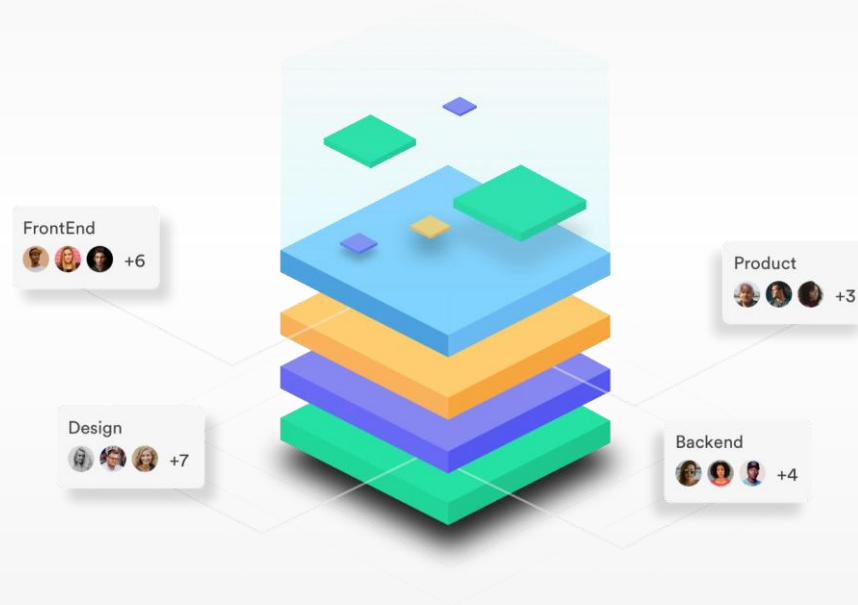➤ Try to keep meetings to a minimum, and focussed.

LeSS SPRINT REVIEW & RETROSPECTIVE

SPRINT REVIEW
TEAM — PRODUCT OWNER — TEAM — USERS & STAKEHOLDERS

TEAM RETROSPECTIVE
TEAM — TEAM

OVERALL RETROSPECTIVE
MANAGER — SCRUMMASTER — PRODUCT OWNER — SCRUMMASTER — TEAM REP.

http://less.works (cc) BY-ND

What about shared components?

This wasn't as big a problem as expected

If you are following good engineering practices, shared components are manageable.

- ➤ Designated "component owner" on one of the teams.

- ➤ Backwards compatibility.

- ➤ Small regular releases, no big bangs.

- ➤ Cross-team meetings to discuss upcoming changes.

- ➤ CD pipelines, automated tests.

- ➤ Open codebase with pull requests.

FrontEnd +6

Product +3

Design +7

Backend +4

ROBERT WALTERS

Challenges

➤ Deciding how to split the teams, this will depend on your business.

➤ Start small with only one team to experiment with, or go all in?

➤ Fear of change.

➤ Dealing with "my team is special and can't move out of its silo" syndrome.

➤ Making sure your engineering practices and infrastructure are ready.

➤ Avoiding the desire to create a "core team" which is just an excuse to keep silos.

➤ Keeping the teams together.

Further reading

https://less.works/
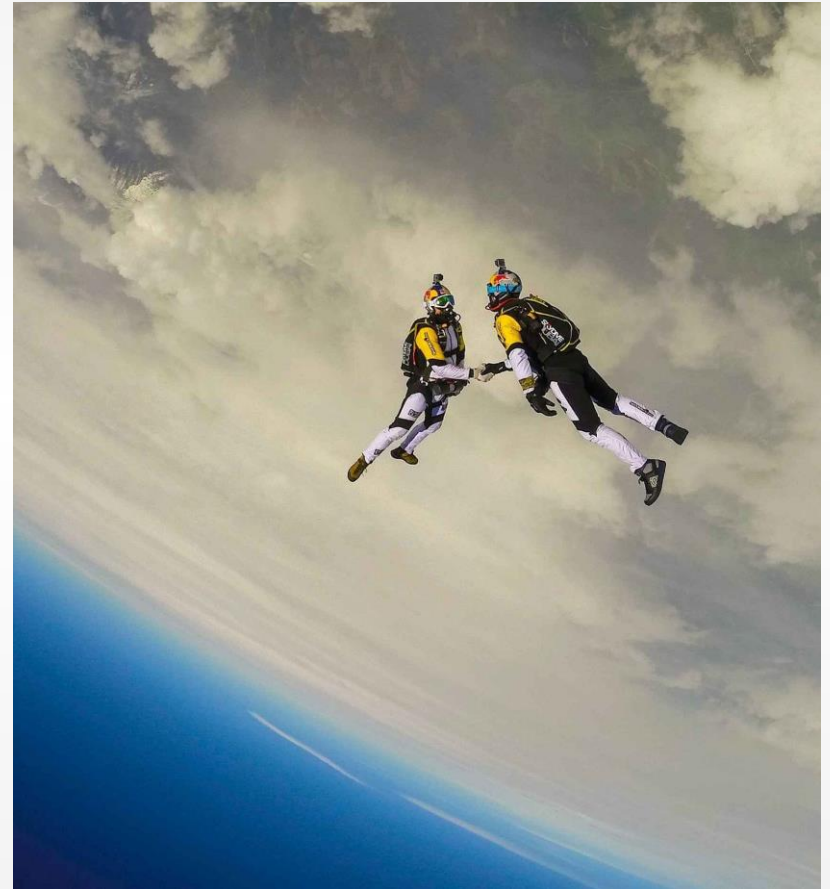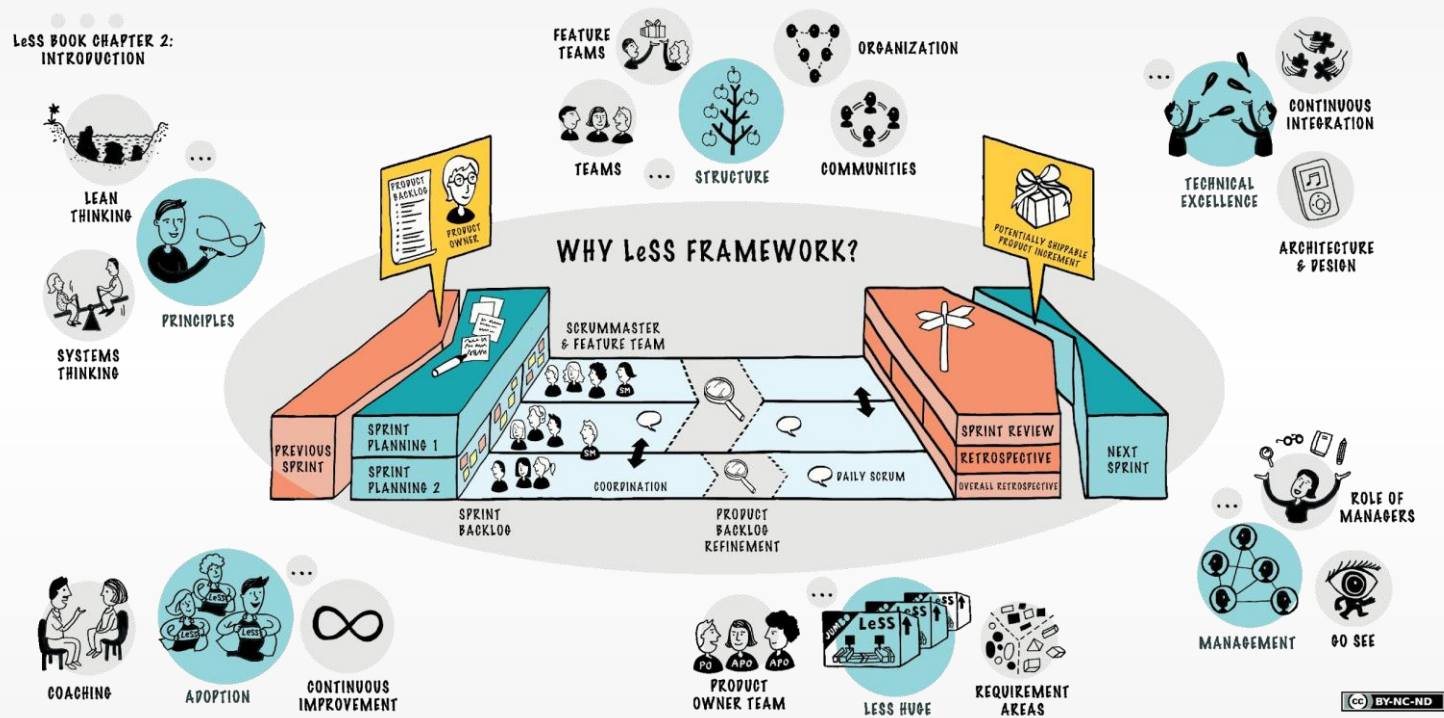
Questions?

Thank you for joining our webinar today.