TEAM TOPOLOGIES
by Matthew Skelton and Manuel Pais

TEAM AS THE MEANS OF DELIVERY

Team assignments
First draft of the architecture

Inverse Conway manoeuvre
Organize teams to match the architecture you want

- Not all communication / collaboration is good
- Restrict communication between teams
- Focus communication between specific teams

> *"Disbanding high-performing teams is worse than vandalism: it is corporate psychopathy."*
> — Allan Kelly, Project Myopia

TEAM FIRST-THINKING

5-9 **Dunbar's number**
Seven-to-nine MAX
> Trust will break down

Owns the Software
"Continuity of care"
No shared ownership

Embrace Diversity
Produce more creative solutions

Use Small, Long-Lived Teams
As the Standard
Autonomous

Minimize Team Cognitive Load
Total amount of mental effort used in the working memory
Use good boundaries

Reward the Whole Team
Not individuals

TEAM TOPOLOGIES THAT WORK FOR FAST FLOW

STREAM-ALIGNED TEAM

Team aligned to a single valuable business **stream of work**

Product or service
Set of features
User Journey
User Journey
User Persona

Primary type in an organization
(80/90 %)

- Work on the **full spectrum** of delivery
- Requires clarity of purpose and responsibility

"Purpose of the other fundamental team topologies is to reduce the burden on the stream-aligned teams."

ENABLING TEAM

Help stream-aligned teams **acquire missing capabilities**

Not a permanent dependency

HIRE SPECIALIST

Composed of specialists
In a given technical or product domain

Collaborative nature
Focus on stream-aligne teams problems first
Not the solutions per se

"Do not exist to fix problems that arise from poor practices, prioritization choices, or code quality within stream-aligned teams."

COMPLICATED SUBSYSTEM TEAM

Reduce cognitive load of stream-aligned teams that needs to **use the complicated subsystem**

Responsible for building / maintaining
A part of the system
That depends heavily on specialist knowledge

Examples : Video processing codec, Mathematical model, Real-time trade, Reconciliation algorithm, Face-recognition, ...

"Prioritizes and delivers upcoming work [...] respecting the needs of the stream-aligned teams that use the complicated subsystem."

PLATFORM TEAM

Provide internal services to reduce cognitive load of stream-aligned teams

Treat services as products
Reliable / Usable
Fit for purpose

Thick platform
Combination of several inner platform teams
Providing a myriad of services

VS

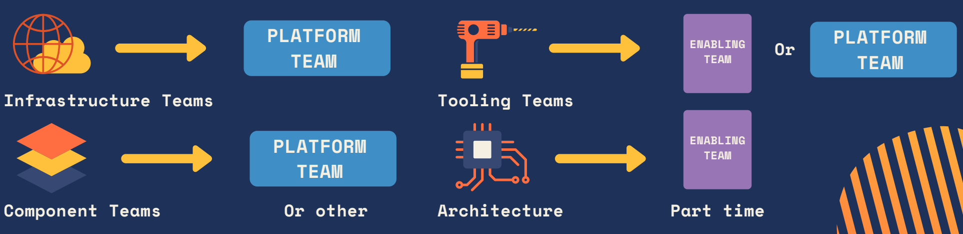
Thin platform
Could simply be a layer on top of a vendor-provided solution

Provision new server instance
Provide tools for access management

"A digital platform is a foundation of self-service APIs, tools, services, knowledge and support which are arranged as a compelling internal product."

Convert Common Team Types to the Fundamental Team Topologies

> *"Most organizations would see major gains in effectiveness by mapping each of their teams to one of the four fundamental topologies [...] to adopt the purpose and behavior patterns of that topology.".*"



Split with FRACTURE PLANES

Software boundaries
Natural Seam
Allowing the system to be split easily

User Personas
Technology
Change Cadence
Regulatory Compliance
Team Location
Risk
Performance Isolation
Business Domain Bounded Context

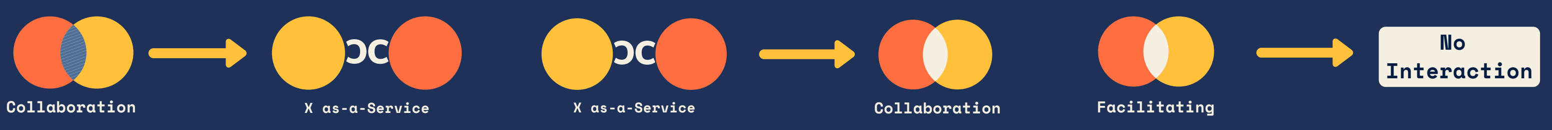
EVOLVING TEAM INTERACTIONS FOR INNOVATION AND RAPID DELIVERY

3 INTERACTION MODES

"Well-Defined Interactions Are Key to Effective Teams"

Interaction patterns per topology	Collaboration 2 teams work together On a shared goal During discovery of new technology or approaches	X as-a-Service (XAAS) 1 team consumes something Provided by another team Such as an API, a tool, or a full software product	Facilitating 1 team facilitates another team Learning / adopting new approach (usually an enabling team)
STREAM-ALIGNED TEAM	Typical	Typical	Occasional
ENABLING TEAM	Occasional		Typical
COMPLICATED SUBSYSTEM TEAM	Occasional	Typical	
PLATFORM TEAM	Occasional	Typical	

EVOLUTIONARY PATTERNS



Teams should ask
What kind of interaction should we have with this other team ?
Should we be collaborating closely with the other team?
Should we be expecting or providing a service?
Or should we be expecting or providing facilitation?

How to get started ?

1. Start with the Team
2. Identify Suitable Streams of Change
3. Identify a Thinnest Viable Platform (services needed)
4. Identify Capability Gaps (Team Coaching, Mentoring,...)
5. Share and Practice Different Interaction Modes
   Explain Principles behind New Ways of Working

Team Topologies alone : not enough
IN ADDITION

Healthy organizational culture
Supports professional development of individuals and teams
Safe to speak
Learn continuously

Good engineering practices
Test-first development
Focus on continuous delivery / operability
Pairing / mobbing for code review ...

Healthy funding / financial practices
Avoiding the pernicious effects of a CapEx/OpEx
Avoiding project-driven deadlines and large-batch budgeting
Allocating training budgets to teams or groups rather than individuals

Clarity of business vision
With horizons at human-relevant timescales
Clear reasoning behind the priorities

#sharingiscaring

BY YOAN THIRION @YOT88