



og en fryktløs endringsreise



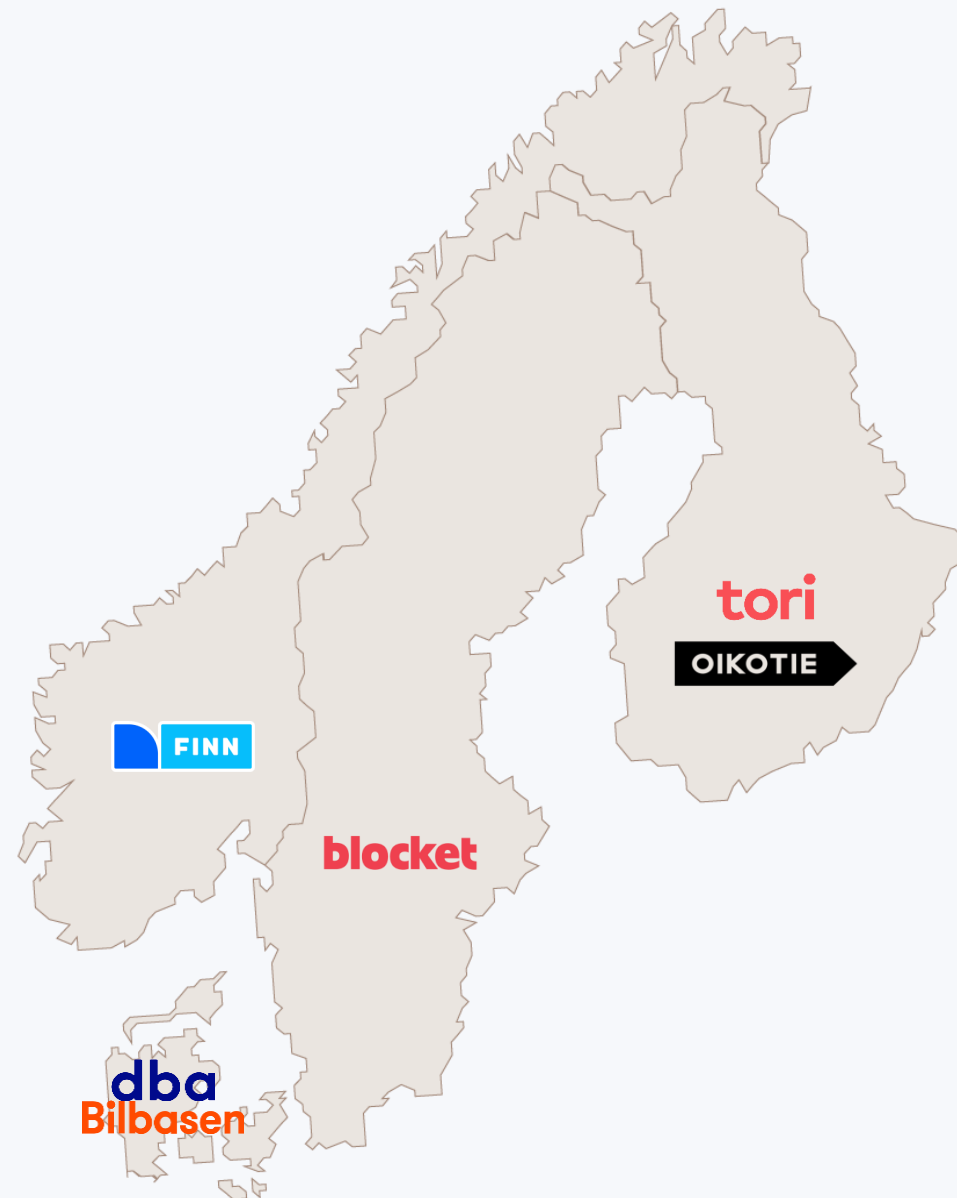
Fryktløs endringsreise?



Konsolidering
Vertikalisering
Plattformisering



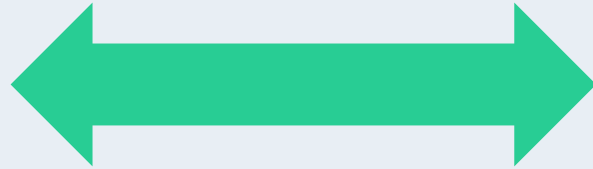
Konsolidering
Vertikalisering
Platformisering







Omorganisering



Selskapsfusjon









blocket



tori
OIKOTIE



dba
Bilbasen





☒ Annonce





- ☒ Annonse
- ☒ Kjøpekontrakt
- ☒ Forsikring
- ☒ Betaling
- ☒ Eierskifte
- ☒ Omregistrering



<input checked="" type="checkbox"/>	
<input checked="" type="checkbox"/>	
<input checked="" type="checkbox"/>	
<input checked="" type="checkbox"/>	
<input checked="" type="checkbox"/>	



<input checked="" type="checkbox"/>	
<input checked="" type="checkbox"/>	
<input checked="" type="checkbox"/>	
<input checked="" type="checkbox"/>	
<input checked="" type="checkbox"/>	

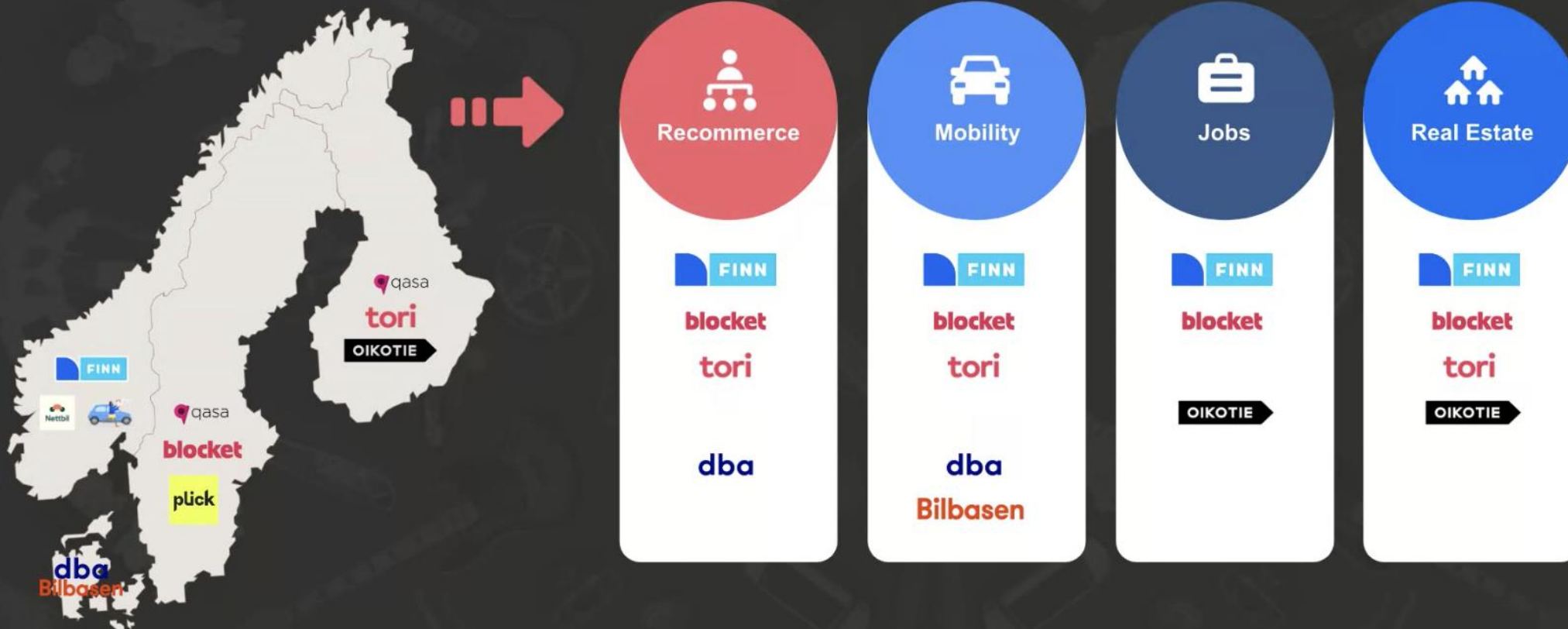


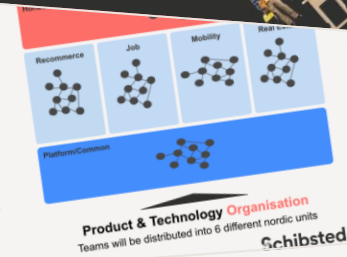
<input checked="" type="checkbox"/>	
<input checked="" type="checkbox"/>	
<input checked="" type="checkbox"/>	
<input checked="" type="checkbox"/>	
<input checked="" type="checkbox"/>	



<input checked="" type="checkbox"/>	
<input checked="" type="checkbox"/>	
<input checked="" type="checkbox"/>	
<input checked="" type="checkbox"/>	
<input checked="" type="checkbox"/>	

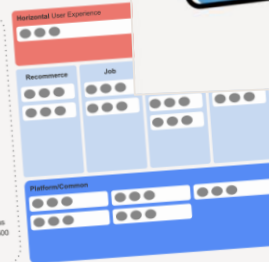
Verticals, not countries





Organisational building blocks

- Product Team**
The cross functional Product Team is the fundamental unit - the atom. Dedicated members from all needed disciplines working together to reach common goals and continuously deliver value to our customers and users. Size: 5-15 (optimal 7-8)
- Extended Product Team**
Product Teams have an extended, stable network with expertise in the organisation. This enables knowledge sharing, alignment, collaboration and an efficient end to end delivery of value to our users/customers.
- Area**
Teams working on same domain/customer journey/business objectives are organised together in areas. Size: 50-150
- Platform / Verticals / Horizontal**
A vertical, platform and horizontal unit consists of several Areas working together delivering on overall strategies. Size: up to 500



NMP Product Development Playbook



Topics
1. Direction
2. Product Development Principles
3. Architecture Principles
4. Goal Steering Framework
5. Product Development structure
7. Role descriptions
8. Collaboration
9. Communities
11. Other things - Dictionary - Inspiration

Conway's law

In 1967, computer scientist Mel Conway wrote an article whose thesis became known as Conway's Law. "Organizations which design systems are constrained to produce designs which are copies of the communication structures of those organizations."⁶

A sort of "revival" of Conway's law took place around 2015, when microservices architectures were on the rise. In particular, James Lewis, Technical Director at Thoughtworks, and others came up with the idea of applying an "inverse Conway maneuver" (or reverse Conway maneuver), whereby an organization focuses on organizing team structures to match the architecture they want the system to exhibit rather than expecting teams to follow a mandated architecture design.¹⁰

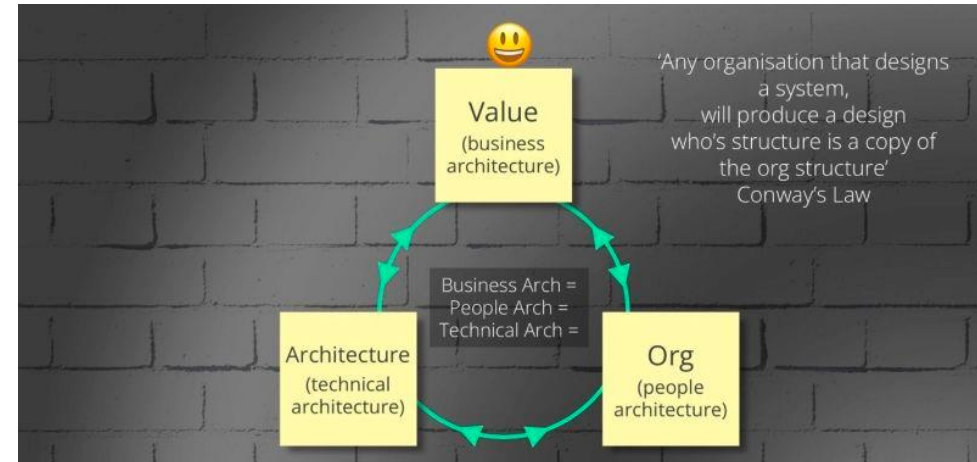
Team structures must match the required software architecture or risk producing unintended designs.

Conway's law

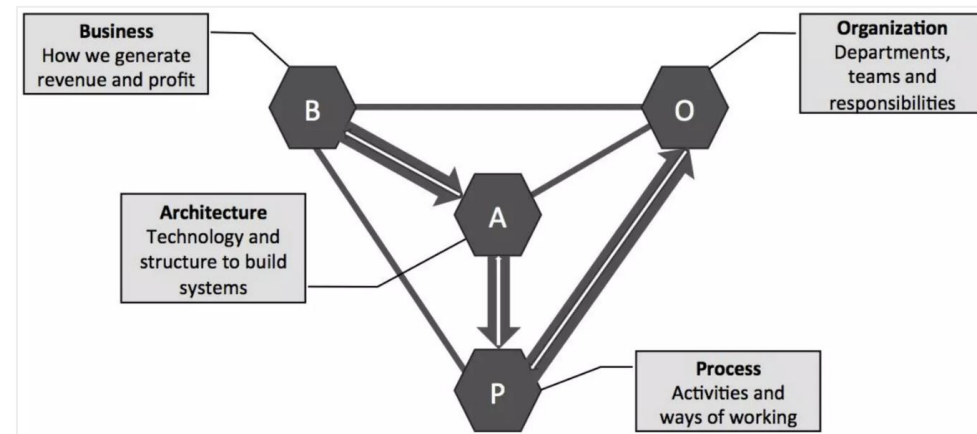
Conway's Law asserts that organizations are constrained to produce application designs which are copies of their communication structures. This often leads to unintended friction points.

Inverse Conway Maneuver

The 'Inverse Conway Maneuver' recommends evolving your team and organizational structure to promote your desired architecture. Ideally your technology architecture will display isomorphism with your business architecture.



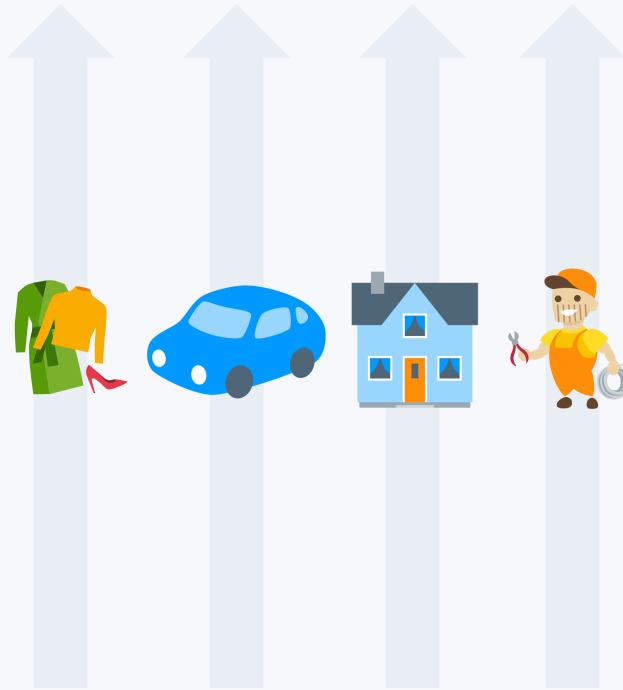
Antipattern: org design without consideration of the flow of value or the technical architecture.
Sooner Safer Happier: Antipatterns and Patterns for Business Agility by Jonathan Smart



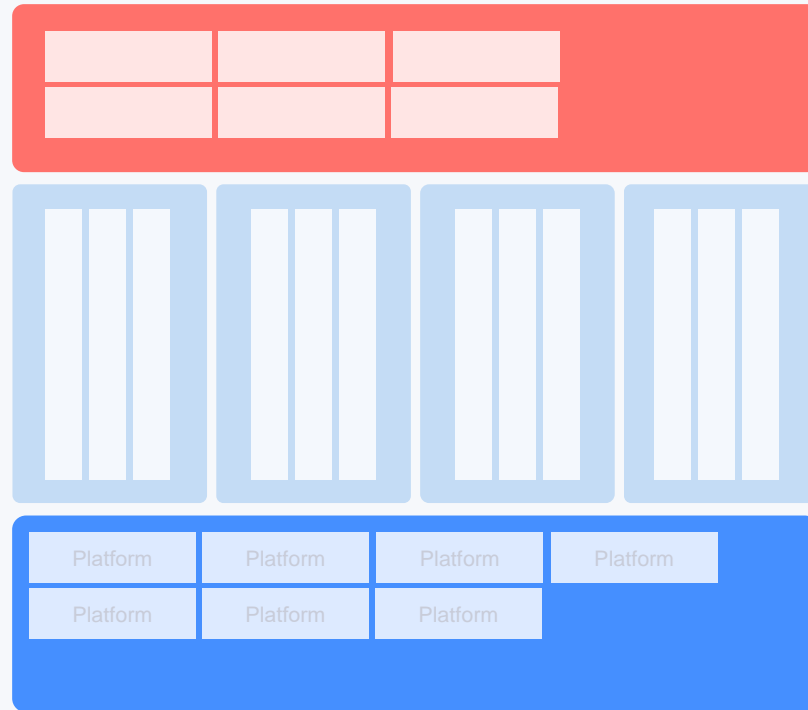
Structure Eats Strategy
BAPO model: Jan Bosh



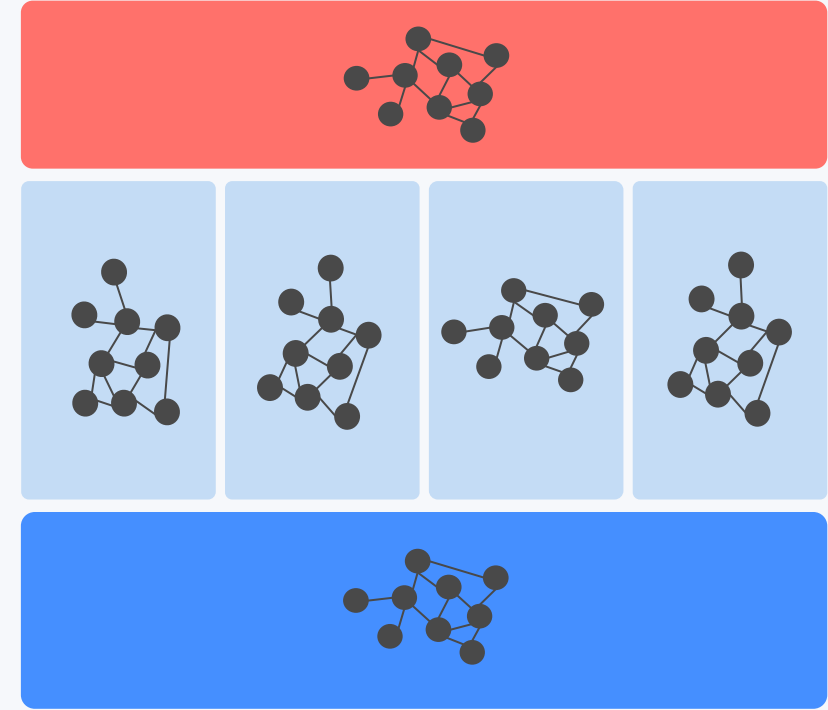
Verdi + Arkitektur + Organisasjon



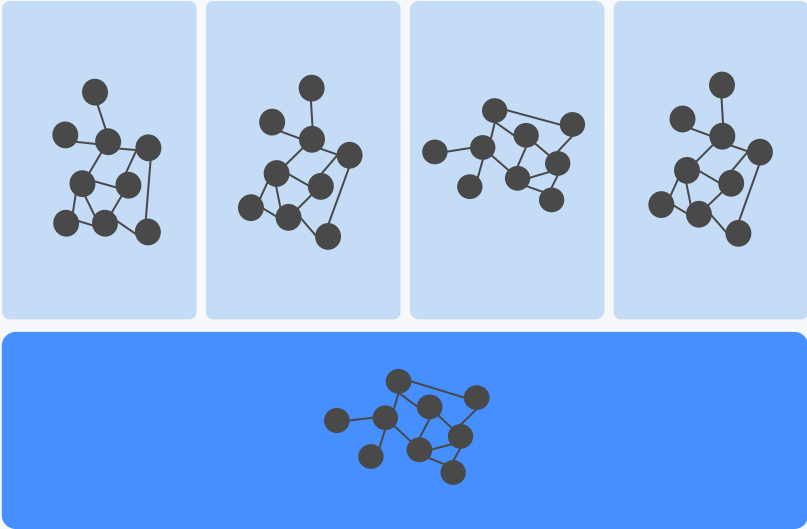
Business Architecture

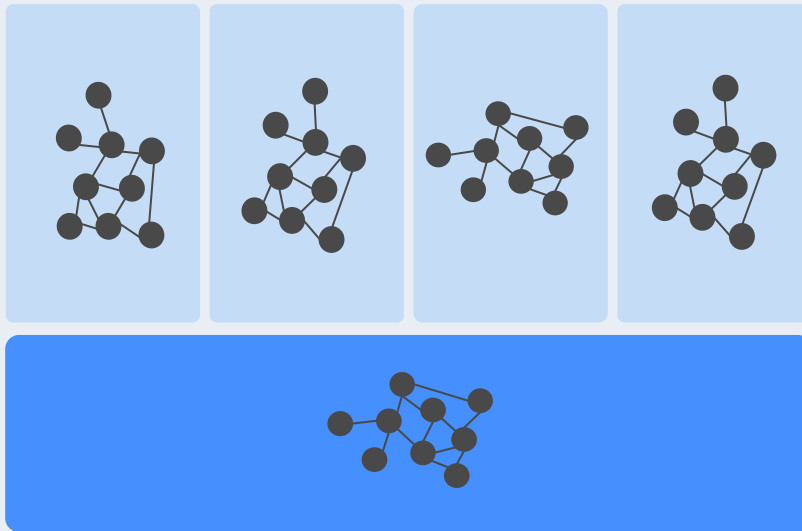


Technical Architecture

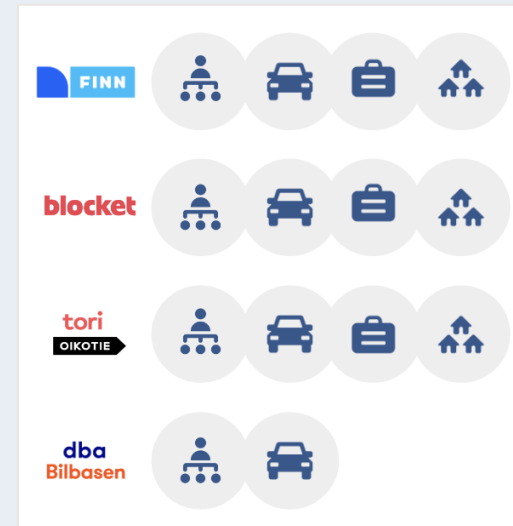


Product & Technology Organisation





Product & Technology Organisation



Current state



- ☒
- ☒
- ☒
- ☒
- ☒



- ☒
- ☒
- ☒
- ☒
- ☒

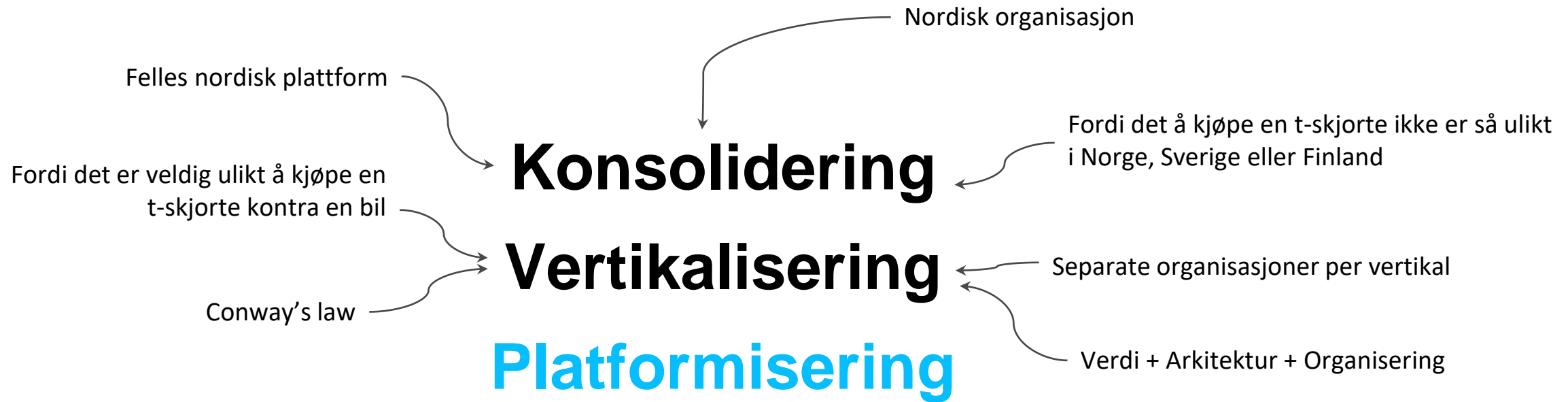


- ☒
- ☒
- ☒
- ☒
- ☒



- ☒
- ☒
- ☒
- ☒
- ☒







Verticals,

tailored to our users jobs-to-be-done,
independent of each other.

Platform (as a Product),

Anything generic that helps **accelerate**
development, configurable, optional and
self-served



Team Topologies

Key concepts

FOUR FUNDAMENTAL TOPOLOGIES

- **Stream-aligned team**: aligned to a flow of work from (usually) a single customer segment of the business domain
- **Enabling team**: helps a Stream-aligned team overcome obstacles. Also detects missing capabilities
- **Complicated Subsystem team**: works on complex mathematics/calculation/technical challenges
- **Platform team**: a grouping of other teams that provides a compelling internal product to access by multiple Stream-aligned teams

THREE TEAM INTERACTION MODES

- **Collaboration**: working together to discover new things (APIs, practices)
- **X-as-a-Service**: one team provides something "as a Service"
- **Facilitation**: one team helps and guides another

Organizations developing and running non-trivial software systems today need to optimize their teams for a safe and rapid flow of change, strongly informed by how

Team Topologies

Four fundamental team topologies



A "stream" is the continuous flow of work aligned to a business domain or organizational capability. Continuous flow requires clarity of purpose and responsibility so that multiple teams can coexist, each with their own flow of work.

A stream-aligned team is a team aligned to a single valuable stream of work; this might be a single product, service, a single set of features, a single user journey, a single user persona. Further, the team is empowered to build and deliver customer or user value as quickly, safely, and independently as possible, without requiring hand-offs to other teams to perform parts of the work.



The purpose of a platform team is to enable stream-aligned teams to deliver work with substantial autonomy. The stream-aligned team maintains full ownership of building, running, and fixing their application in production. The platform team provides internal services to reduce the cognitive load that would be required from stream-aligned teams to develop these underlying services.

Team Topologies

Interaction modes

Three Interaction Modes



Advantages	Disadvantages
<ul style="list-style-type: none">• Rapid innovation and discovery• Fewer hand-offs	<ul style="list-style-type: none">• Wide, shared responsibility for each team• More detail/context needed between teams, leading to higher cognitive load• Possible reduced output during collaboration compared to before
Constraint: A team should use collaboration mode with, at most, one other team at a time. A team should not use collaboration with more than one team at the same time.	
Typical Uses: Stream-aligned teams working with complicated-subsystem teams; stream-aligned teams working with platform teams; complicated-subsystem teams working with platform teams	

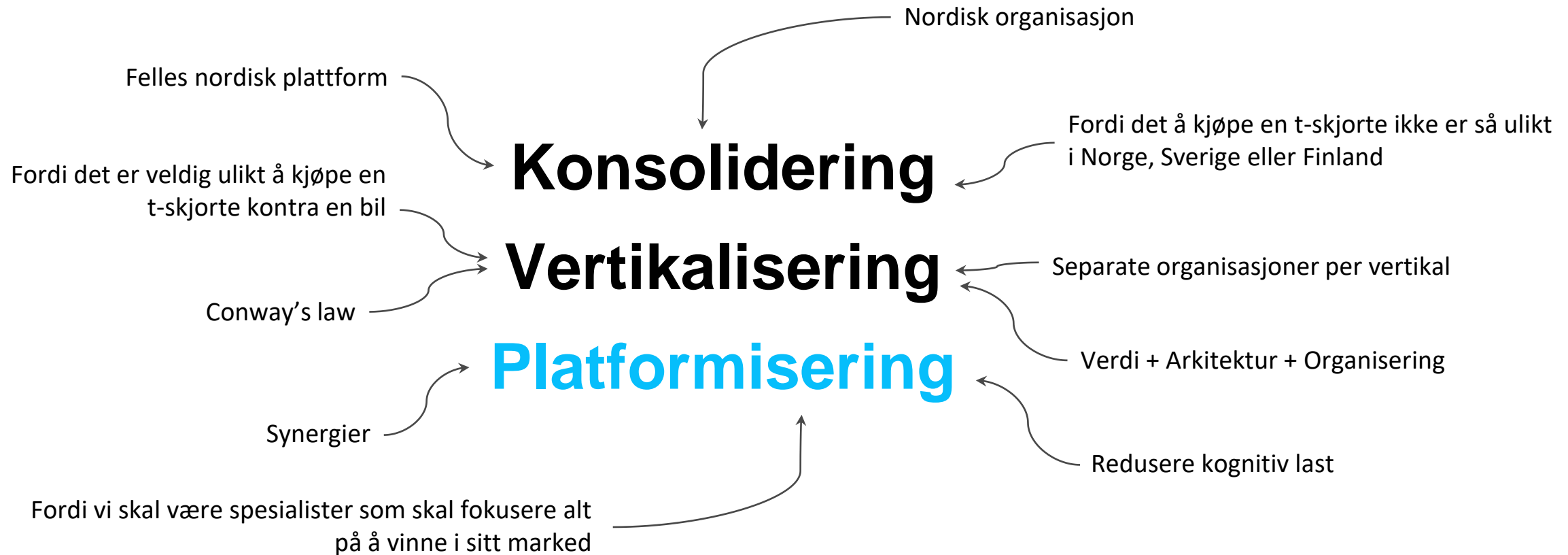
Collaboration mode

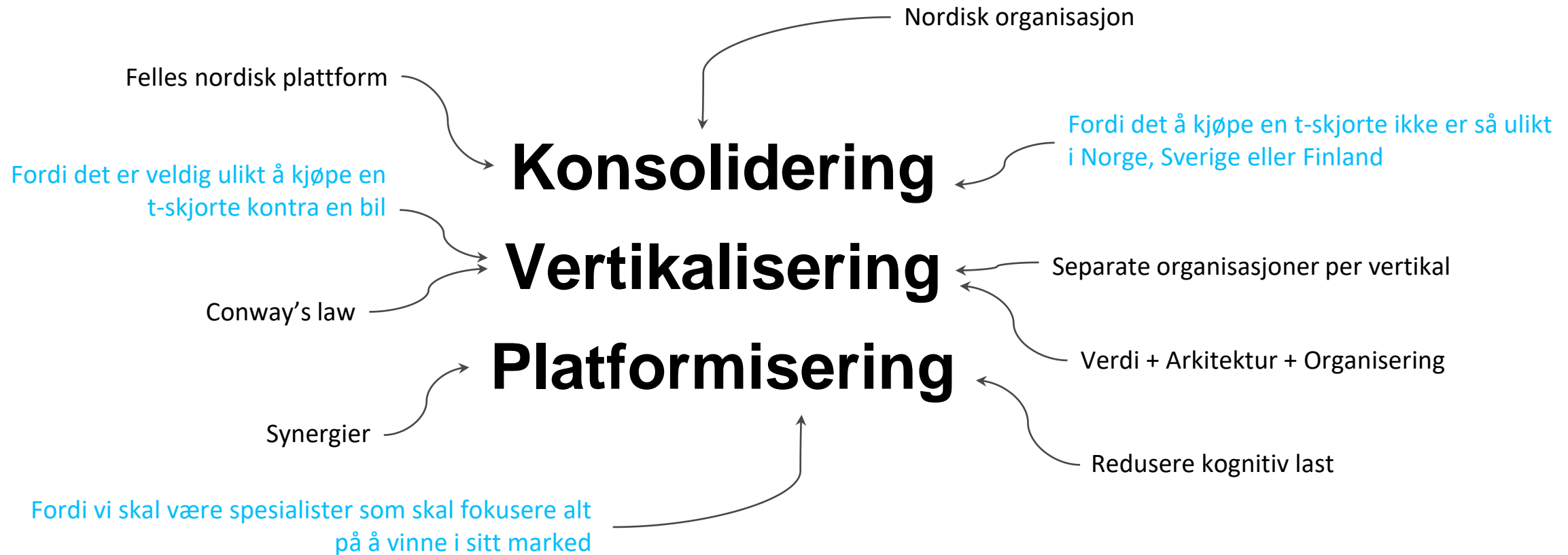
Advantages	Disadvantages
<ul style="list-style-type: none">• Clarity of ownership with clear responsibility boundaries• Reduced detail/context needed between teams, so cognitive load is limited	<ul style="list-style-type: none">• Slower innovation of the boundary or API• Danger of reduced flow if the boundary or API is not effective
Constraint: A team should expect to use the X-as-a-Service interaction with many other teams simultaneously, whether consuming or providing a service.	
Typical Uses: Stream-aligned teams and complicated-subsystem teams consuming Platform-as-a-Service from a platform team; stream-aligned teams and complicated-subsystem teams consuming a component or library as a service from a complicated-subsystem team.	

X-as-a-Service

Advantages	Disadvantages
<ul style="list-style-type: none">• Unblocking of stream-aligned teams to increase flow• Detection of gaps and misaligned capabilities or features in components and platforms	<ul style="list-style-type: none">• Requires experienced staff to not work on "building" or "running" things• The interaction may be unfamiliar or strange to one or both teams involved in facilitation
Constraint: A team should expect to use the facilitating interaction mode with a small number of other teams simultaneously, whether consuming or providing the facilitation.	
Typical Uses: An enabling team helping a stream-aligned, complicated-subsystem, or platform team; or a stream-aligned, complicated-subsystem, or platform team helping a stream-aligned team.	

Facilitating mode







Takk for meg!

Jostein Emmerhoff
Schibsted Nordic Marketplaces



blocket

dba

tori

Bilbasen

OIKOTIE