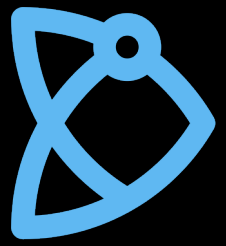
DAIRDUX

Intelligence in maintenance—

For whom, doing what?

Henrik Göthberg
henrik.gothberg@dairdux.com

Mikael Klingvall
mikael.klingvall@dairdux.com



“Agentic AI: Decisive, Operational AI Arrives In Business”

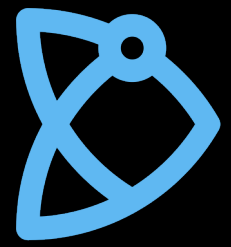
—CIO.com
(featured article,
30th of Aug 2024)

5 sec jobs	5 min jobs	5 hour jobs
ChatGPT 3.5	ChatGPT 4.o	ChatGPT Nxt

Sam Altman, Spring 2024

Agentic AI in Intelligent Maintenance

Whose intelligence?
Which decisions?
For whom, doing what?



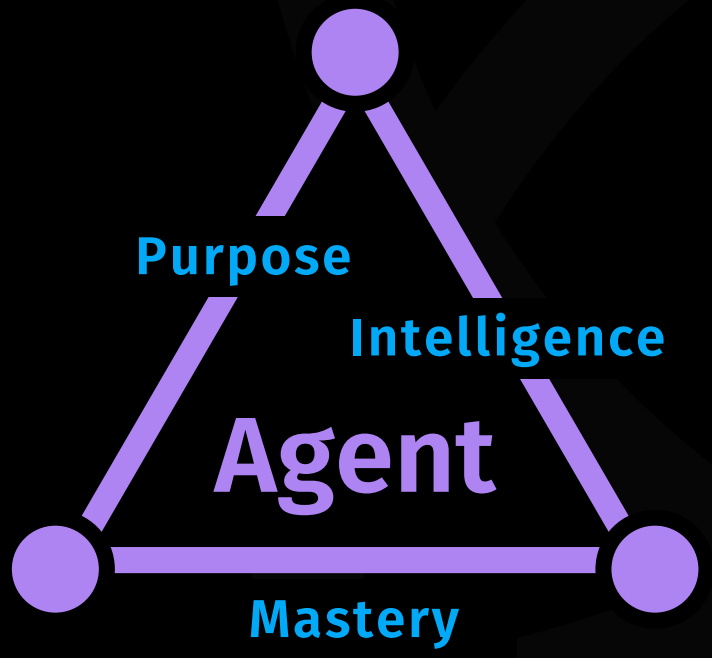
‘Agentic’ means “behaves like an agent”

Autonomy

(self-determination)

lets agents assert patterns

make decisions →



Agency

lets agents accomplish outcomes

← enact (sets of) patterns

Adaptability

lets agents adapt = learn and adopt patterns

rules
algos
functions
heuristics

‘Agentic’ means ‘behaves like an agent’, and agents are able to act, learn, and decide

- **Act: do and accomplish things (agency)**
- **Learn from mistakes made (feedback) and adjust its behaviour (adaptability)**
- **Decide** (about) **its future actions on its own (autonomy)**

Agentic AI is still in its early days,
but the writing is on the wall

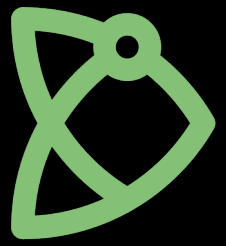
Agency Rules

...because it always comes down
to the outcomes accomplished

Agentic AI is the latest manifestation of something that started 75 years ago

Norbert Wiener, father of cybernetics, theorised that all intelligent behaviour was the result of feedback mechanisms

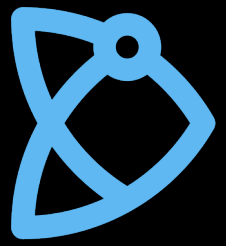
And that these mechanisms could possibly be simulated by machines



‘Agenticly’ speaking, intelligence emerges from ensembles of agents interacting, adjusting to each other

Each of them learns from the consequences of their actions feeding back to them as inputs for future actions

Intelligence is Agentic



Let’s take this idea into maintenance data–decision workflows

Maintenance operations orbit around data work and data-driven, knowledge-intensive decision-making

Do data work to augment decisions

Make decisions to augment data work

Maintenance operations orbit around data work and data-driven, knowledge-intensive decision-making

Data workers and decision-makers mutually adjust through feedback loops

Do data work to augment decisions

Make decisions to augment data work

Maintenance operations orbit around data work and data-driven, knowledge-intensive decision-making

Within domain

Data workers and decision-makers mutually adjust through feedback loops

Do data work to augment decisions

Make decisions to augment data work

Maintenance operations orbit around data work and data-driven, knowledge-intensive decision-making

A data–decision workflow extends to other entities to accomplish useful outcomes

Within domain

Data workers and decision-makers mutually adjust through feedback loops

Do data work to augment decisions

Make decisions to augment data work

Maintenance operations orbit around data work and data-driven, knowledge-intensive decision-making

A data–decision workflow leverages feedback loops to stay useful to other entities

A data–decision workflow extends to other entities to accomplish useful outcomes

Within domain

Data workers and decision-makers mutually adjust through feedback loops

Do data work to augment decisions

Make decisions to augment data work

Maintenance operations orbit around data work and data-driven, knowledge-intensive decision-making

Between domains

- A data–decision workflow leverages feedback loops to stay useful to other entities
- A data–decision workflow extends to other entities to accomplish useful outcomes

Within domain

- Data workers and decision-makers mutually adjust through feedback loops
- Do data work to augment decisions
- Make decisions to augment data work
- **Maintenance operations orbit around data work and data-driven, knowledge-intensive decision-making**

...are sociotechnical systems created and operated to accomplish outcomes for agents by agents through agents, agentic systems, and technical systems

data–decision workflows

...are sociotechnical systems created and operated to accomplish outcomes for agents by agents through agents, agentic systems, and technical systems

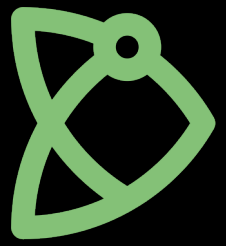
The ‘socio-’ part refers to agents trying to coordinate, cooperate, or co-act—in a web of feedback loops

data–decision workflows

…are sociotechnical systems created and operated to accomplish outcomes for agents by agents through agents, agentic systems, and technical systems

The ‘socio-’ part refers to agents trying to coordinate, cooperate, or co-act—in a web of feedback loops

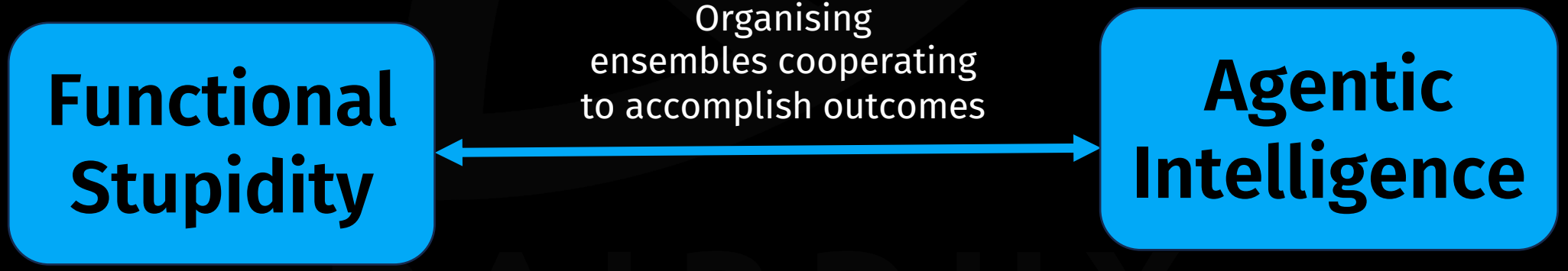
The ‘-technical’ part refers to those agents using tech/systems to get jobs done and accomplish outcomes



Managing *anything* sociotechnical benefits from explicitly taking an agentic perspective

BUT… how well have we organised for agentic intelligence on team, unit, and enterprise level to address the sociotechnics of our workflows?

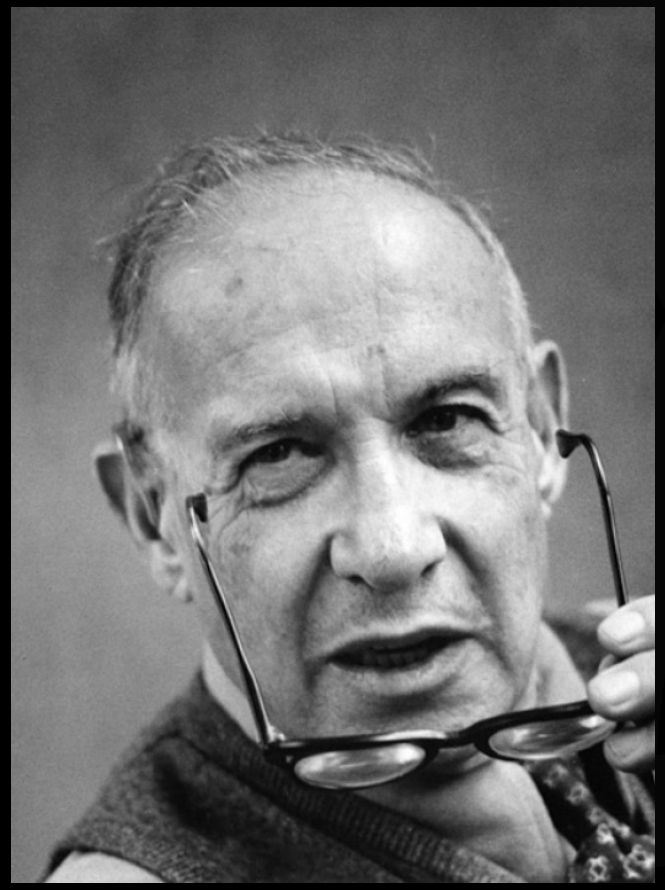
Have we even considered this as a design goal for our organisation and workflows?



Organising ‘ensemble cooperation’ poorly results in functionally stupid behaviour

Our ways of working can easily ‘make smart people do stupid things’ through unintended consequences

Alvesson & Spicer 2012 & 2016



Peter Drucker

“Only three things happen naturally in organizations:

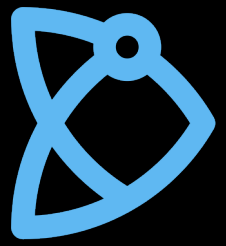
friction, confusion, and underperformance

Everything else requires leadership”

What, then, leads to agentic intelligence?

Domain-Driven Design, *Team Topologies*, and *Data Mesh* describe an

‘agentic’ separation of concens

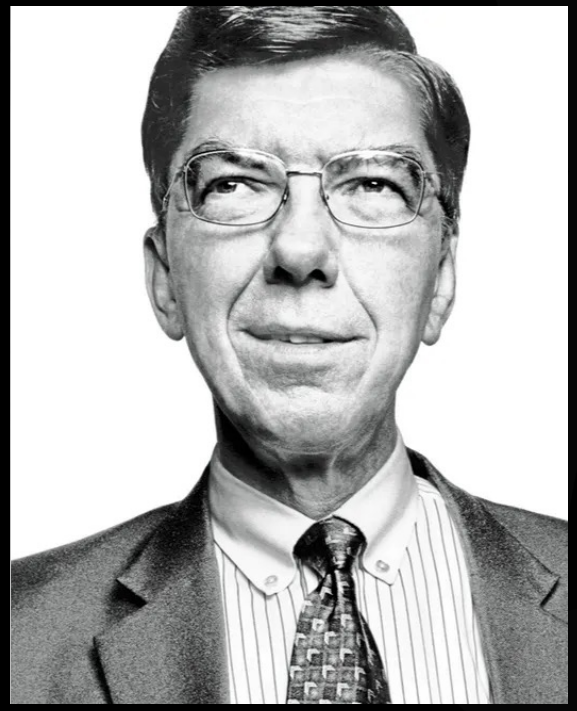


The books are brilliant

And we have tried to extract a heuristic that can be used recursively as organisational design pattern:

The ‘Boost’ Heuristic

The ‘Boost Heuristic’ also leans heavily on Christensen’s ‘Theory of Jobs To Be Done’ (JTBD)



Clayton Christensen

When people find themselves needing to get a job done,

they ‘employ’ products to get it done

The job, not the customer, is the unit of analysis

paraphrased from Clayton Christensen (2005), “Marketing Malpractice: The Cause and the Cure”

The Boost Heuristic + Agency Rules

The Boost heuristic: boost the reachers!

Other agents reaching for new ways to get their jobs done frames the job-to-be-done: the boost's purpose and the team's required mastery

The Boost Heuristic + Agency Rules

The Boost heuristic: boost the reachers!

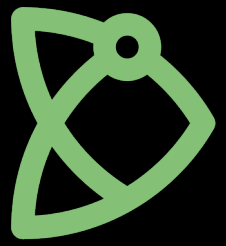
Other agents reaching for new ways to get their jobs done frames the job-to-be-done: the boost's purpose and the team's required mastery

Products are team-sized

The pivotal unit of action and agent of change: a team with high cohesion and low coupling—decompose any product until teams have clear purpose and workable cognitive loads

Product as interface

Productising boosts as co-created self-services enables reachers to get their jobs done without blockers, boosting their autonomy and their productivity



To increase intelligence in maintenance operations—lead with...

Agentic Separation of Concerns

instead of

Functional Division of Labour

The Boost heuristic

Agentic Separation of Concerns

‘Reachers’ are struggling to get jobs done within their existing workflows, **reaching for boosts**: new ways to get jobs done

A boost frames **team agency**: its purpose, job, and needed mastery

Functional Division of Labour

A self-absorbed focus on reducing the costs of your own operation means lack of focus on boosting reacher operations

The result is “friction, confusion, and underperformance”

Agency Rules: Products are team-sized

Agentic Separation of Concerns

Booster teams with agency and autonomy:

Vehicles for making a difference for other agents and achieving changes that stick

Functional Division of Labour

Functional units and workers:

Task machines with minimal agency and autonomy: silos with low cohesion and high coupling

Agency Rules: Product as interface

Agentic Separation of Concerns

Agents using **self-services** to get their jobs done reduces their couplings to a minimum

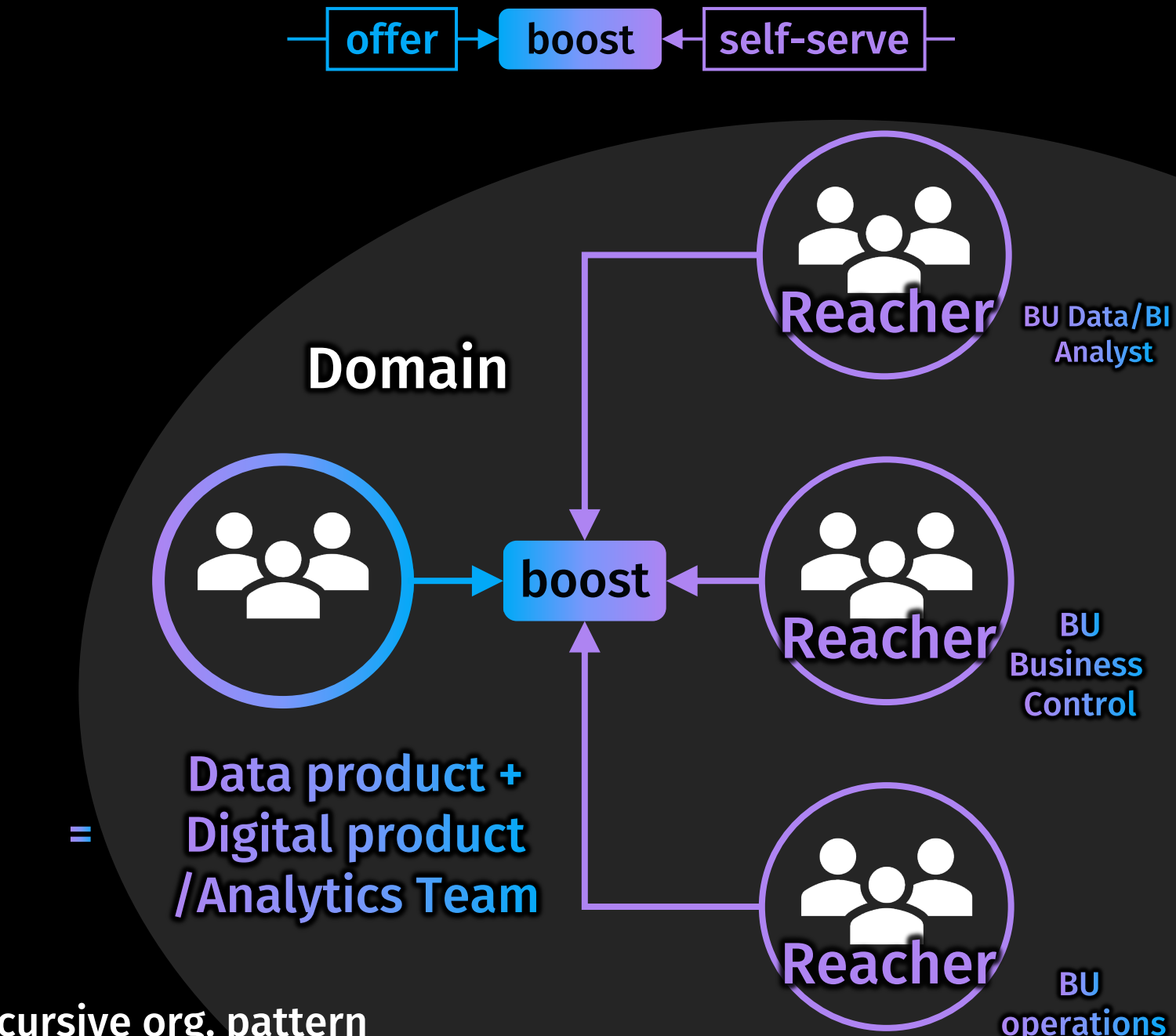
Productisation of couplings into self-service interfaces increase autonomy and, thus, productivity

Functional Division of Labour

High coupling impedes efforts to boost workflows and increases both the ‘blast radius’ of changes and team cognitive load

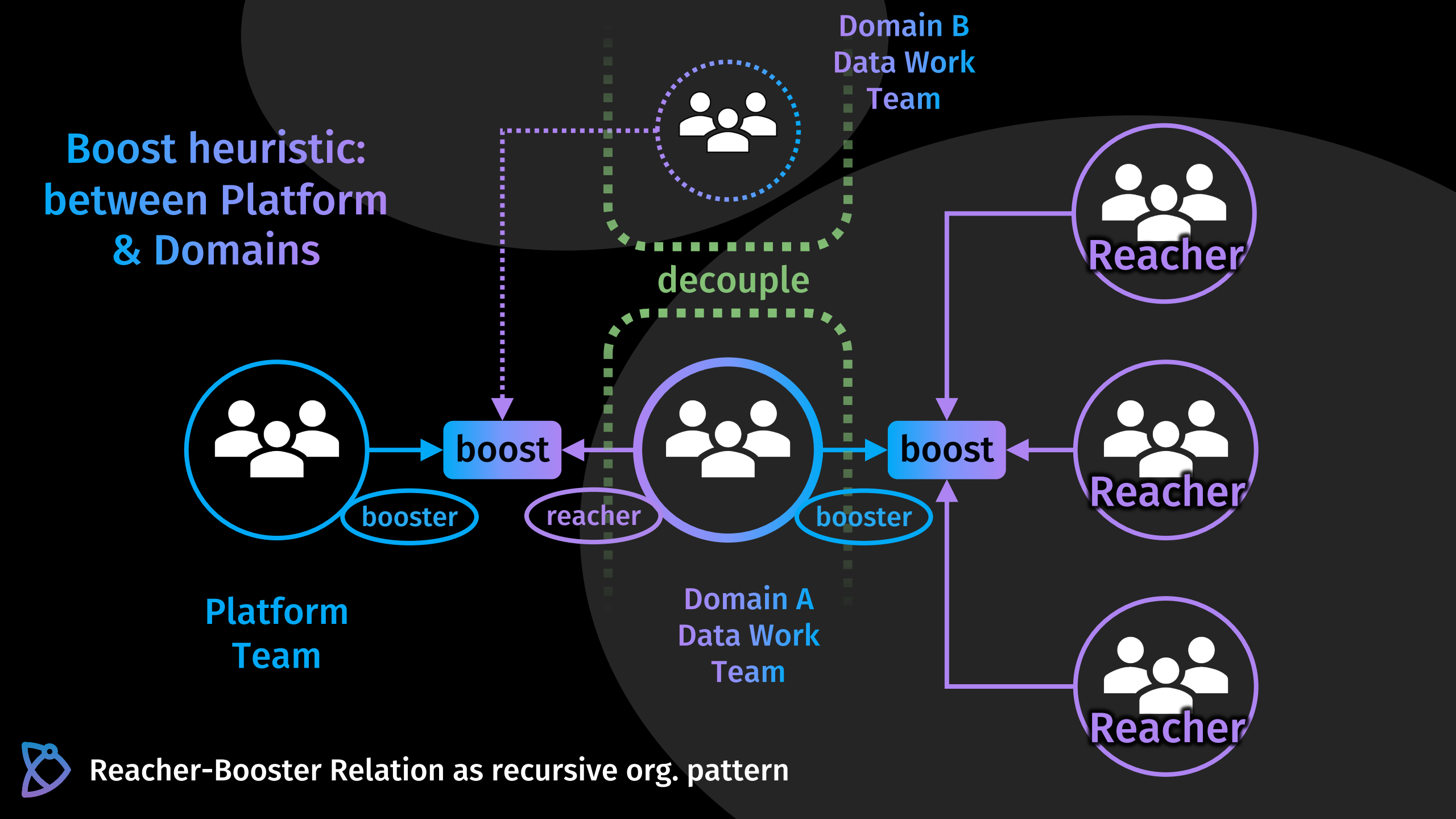
The only workaround is developing point solutions

Boost heuristic: within the domain



Domain Data Work Team = Data product + Digital product /Analytics Team

Reacher-Booster Relation as recursive org. pattern


Boost heuristic:
between Platform
& Domains
Domain B
Data Work
Team
decouple
Reacher
Reacher
Reacher
boost
boost
booster
reacher
booster
Platform
Team
Domain A
Data Work
Team
Reacher-Booster Relation as recursive org. pattern

Intelligent Maintenance must be smart about AI Hype

Companies are scrambling to formulate AI strategies and AI use-cases

These efforts tend to result in innovation theater: the appearance of keeping up with the AI revolution to look good to investors/owners while continuing to do ‘business as usual’

Employing *Agentic AI* in workflows and ignoring the sociotechnics will suffer unintended consequences

Neglecting *Agentic Separation of Concerns* comes with AI Safety risks and limits AI innovation speed