

Internalized Liabilities of Organizations

... obvious ~~but invisible~~...

Approach for this presentation

Simple reason:

This is about relating the three complex
and complicated fields of Finance,
Systems Thinking and IT.

1. High level walk through, staying at the surface
⇒ You need to trust me, with quite some things ...

2. Deep(er) Dive into Details
⇒ Substantiating it

😊 Scratch your head, play with it, challenge it, ...

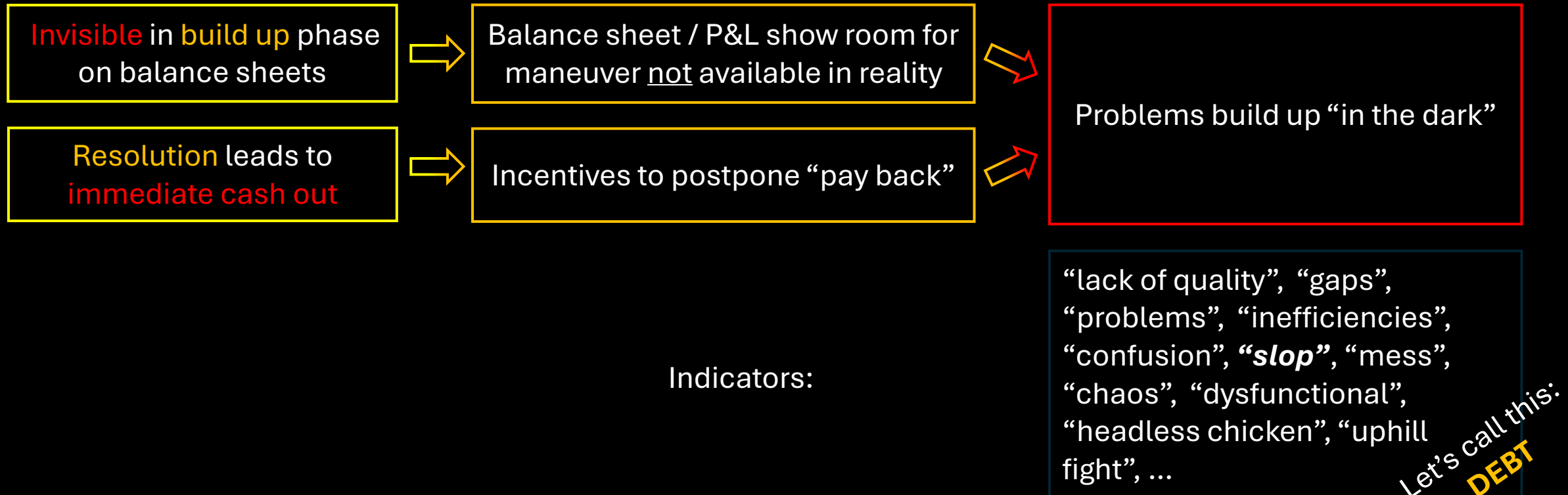
Afterwards

3. Use Cases

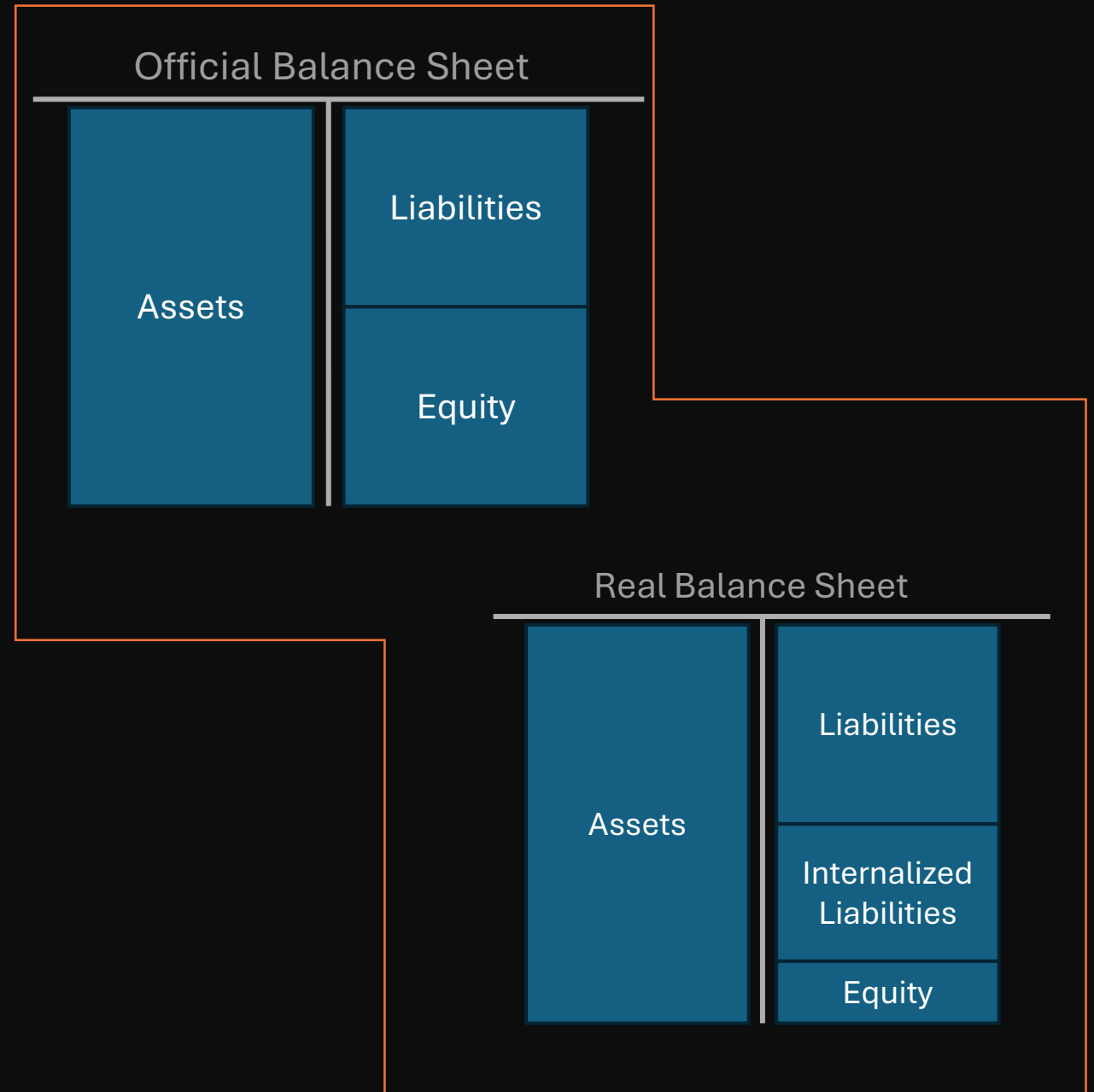
4. Conjectures & Implications

Property of Internalized Liabilities

obligations in the future
! but no allowance for building provisions !



Internalized
Liabilities are not
accounted for
under “traditional”
accounting
regimes



How to address this?

Systems Thinking is the only way out (that I am aware of)

Components

Internalized Liabilities of Organizations

Strategic Debt		
Organizational Debt		
Process Debt		
Technical Debt	Functional Debt	Data (Quality) Debt

This can be operationalized

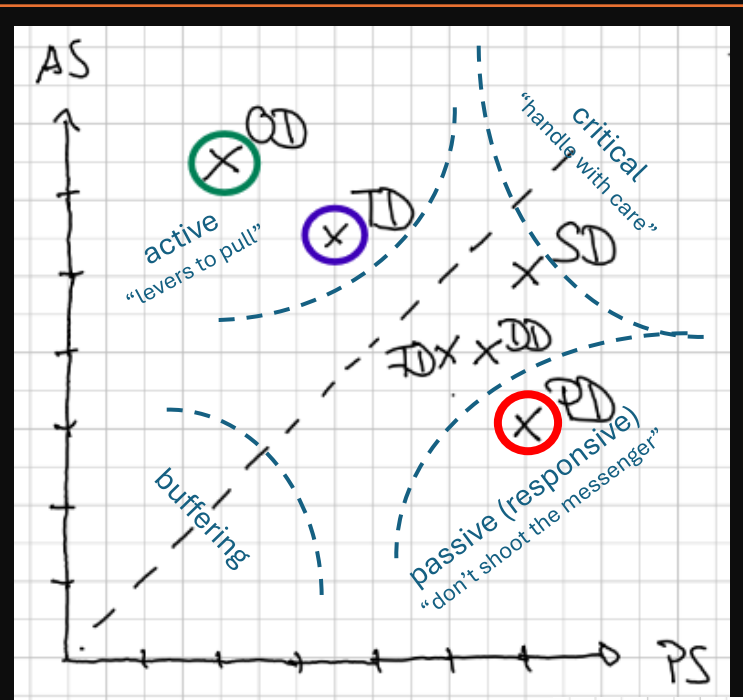
Major Claim: This model is complete!

in the sense that everything else finds its place on the balance sheet somewhere else – or is not material

A first flavor of systems thinking

Sensitivity Map

Nice and neat application of “fuzzy logic”



Influence Matrix

(aka “paper computer”)

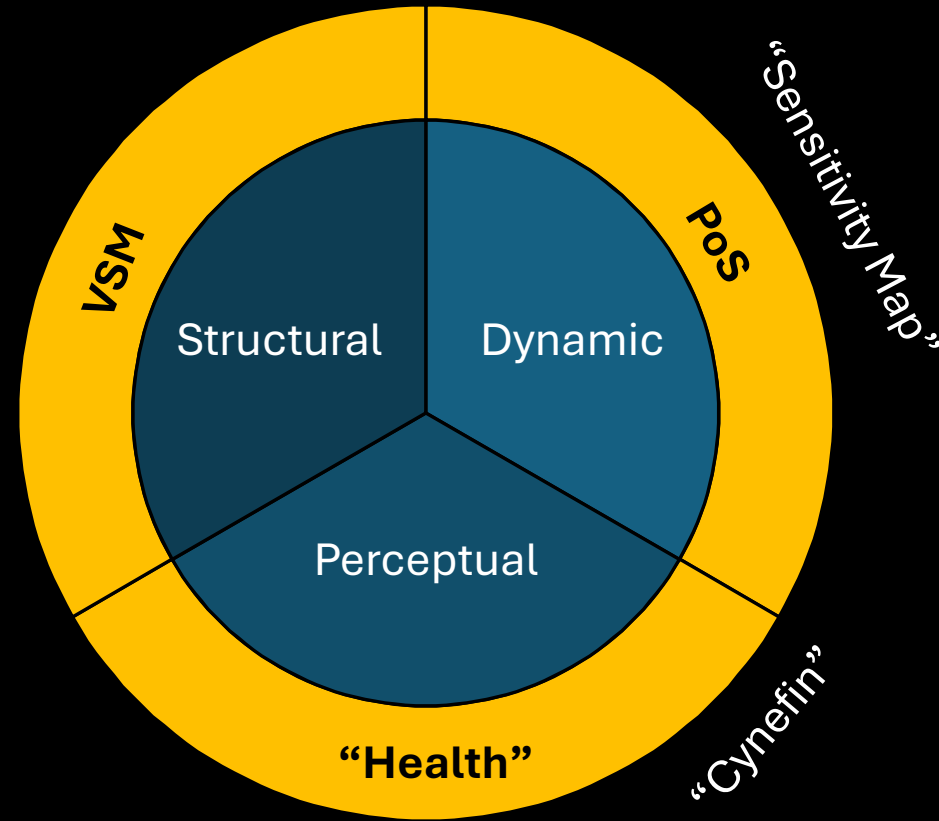
Basically, a causal loop diagram with numbers representing the strength of the arrow.

Disclaimer:

My personal, average, influence matrix of what I have seen so far after two iterations of polishing
Yours will differ and specific situations will differ as well

Impact on →		AS					
of	TD	FD	DD	PD	OD	SD	Σ
TD	2	3	2	1	0	3	11
FD	1	0	2	3	0	2	8
DD	0	2	1	3	0	2	8
PD	1	1	2	0	0	2	6
OD	2	3	2	3	1	2	13
SD	1	1	2	2	3	1	10
PS Σ	7	10	11	12	4	12	

Complexity



Internalized Liabilities (based on VSM and PoS) are “Complexity Complete” in this sense

First Tests & Use Cases

(“Safe to Fail Experiments”)

Change Process Business Unit & IT	An interface absorbing complexity
Target Setting Process	Three questions to my team: • Which targets are missing? • Which impediments are in our way? • What would help with, achieving our targets?

End of First Iteration

What is The Right to Exist?

... for Systems and Complexity in Organizations ...

Organizations applying the approaches should be better off compared to

- Peers not doing this
- The time before the organization applied it

This means:

- Anecdotal evidence, story telling is helpful
- But not enough

THE benchmark are the Financial Statements

- Mid-term results need to show this
- “Short term evidence is just randomness”
- “Long term we are all dead”

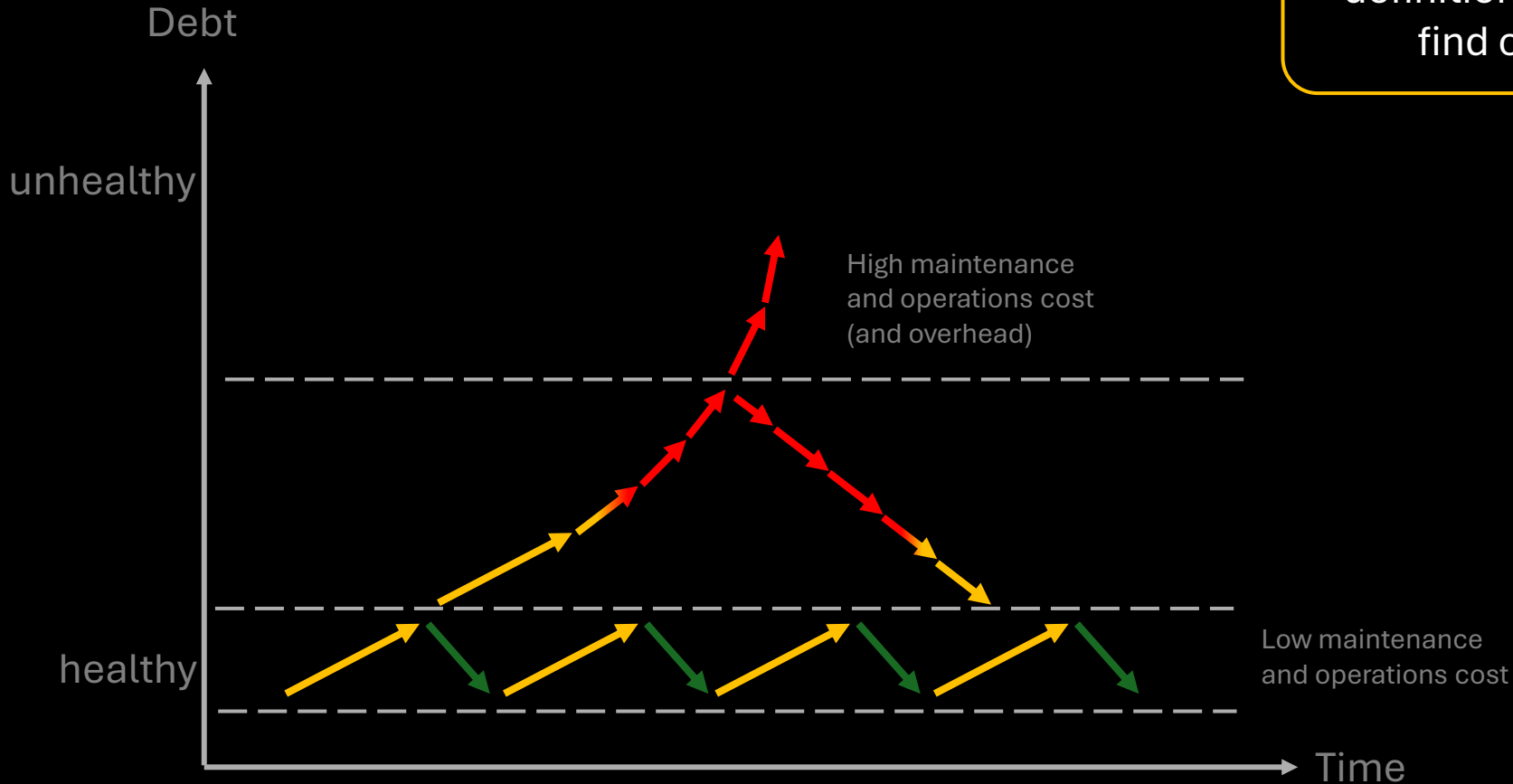
Is there something we can leverage, copy or steal from?

- IT did something on “**Technical Debt**” since 1992 ...¹⁾
- Detecting a link between IT / software development and Finance

¹⁾ Ward Cunningham

Technical Debt

(of IT-systems)



Worth noticing: Carola Lilienthal (WPS) had this impressive visualization idea

These **root-cause** definitions you will find online

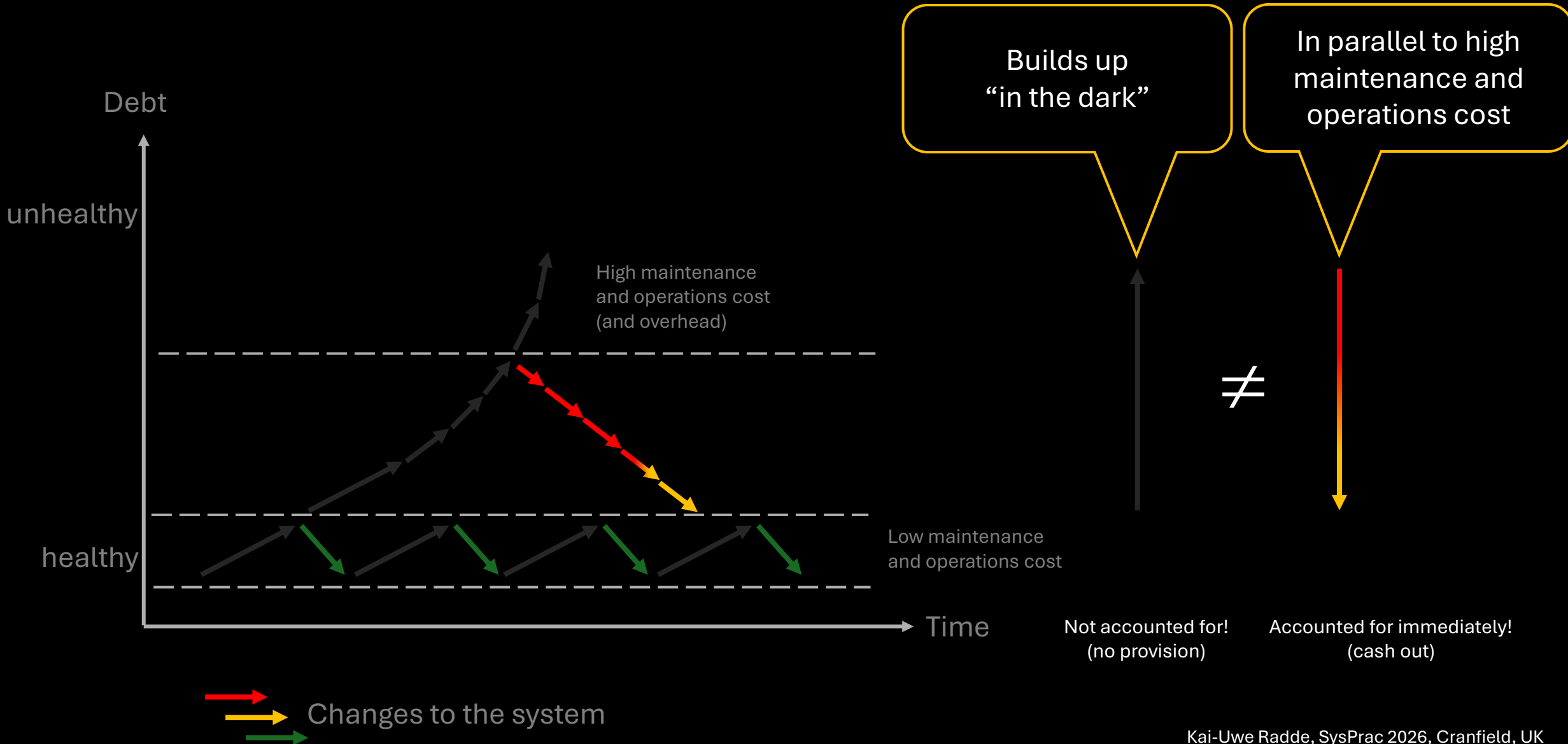
This **forward-looking** definition is simple and more helpful

=

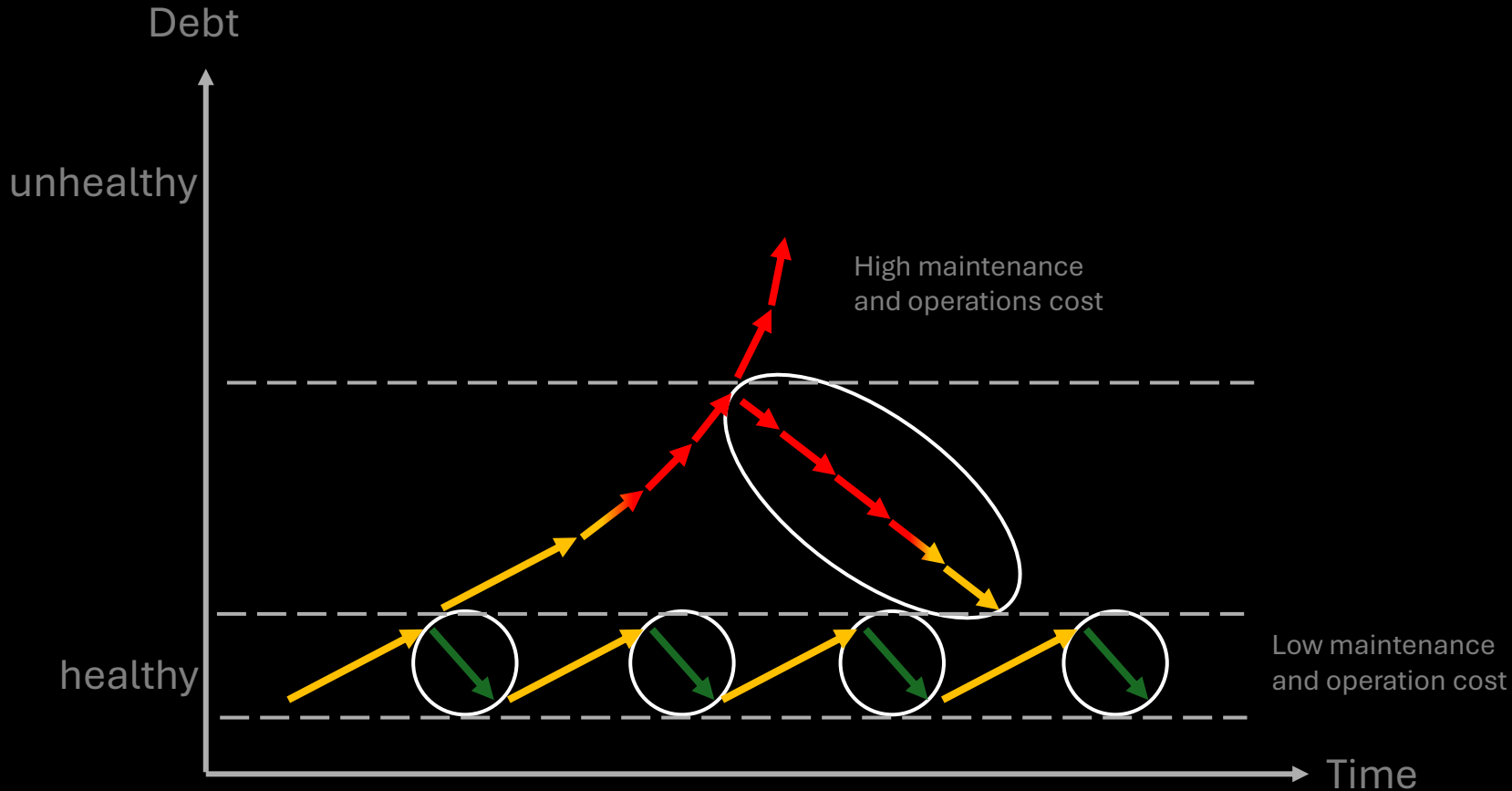
Not accounted for!
(no provision)

Accounted for immediately!
(cash out)

Technical Debt in the Finance Function



Defining Technical Debt



Worth noticing: Carola Lilienthal (WPS) had this impressive visualization idea

Definition:

- (1) changes
- (2) targeting a healthy IT state,
- (3) that are not yet implemented

Important:

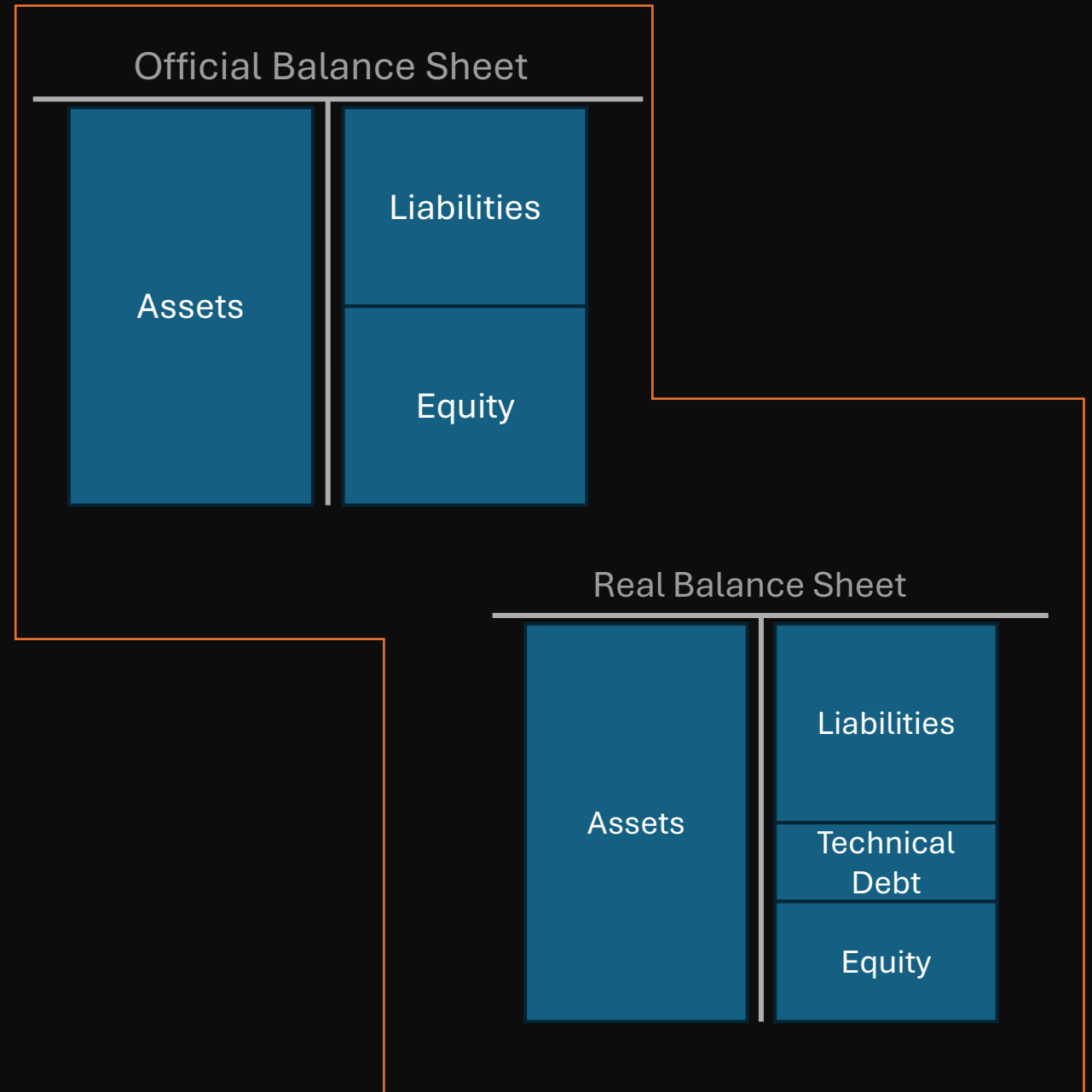
Technical Debt is the minimum investment an organization needs, to (just) stay at its desired IT delivery level.

Even more important:

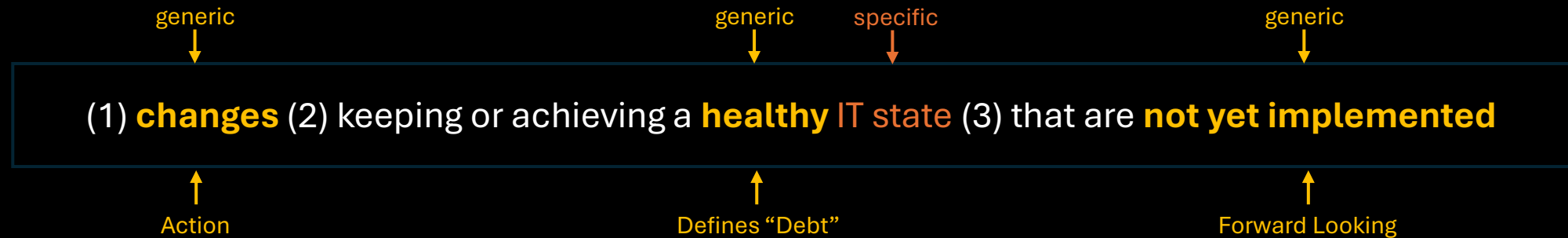
Technical Debt is useful

- To gain speed
- If kept under control

Technical Debt is
not accounted for
under “traditional”
accounting
regimes



The “Eureka Moment”



Internalized Liabilities of Organizations

Liability	Consistent Definitions	Model of Choice / Health Criteria
Strategic Debt	(1) changes (2) keeping or achieving a healthy strategic position, (3) that are not yet implemented	Patterns of Strategy / Fit
Organizational Debt	(1) changes (2) keeping or achieving a healthy organization, (3) that are not yet implemented	VSM / 6 subcomponents
Process Debt	(1) changes (2) keeping or achieving a healthy end-to-end process, (3) that are not yet implemented	... / buffers, cycle times, quality, effort
Data (Quality) Debt	(1) changes (2) keeping or achieving healthy data quality, (3) that are not yet applied	... / Objective & Subjective completeness, validity, uniqueness, credibility, utility, interpretability
Functional Debt	(1) changes (2) keeping or achieving a healthy level of functionalities i.e., usability (3) that are not yet implemented	... / ... Individually assessed per IT system
Technical Debt	(1) changes (2) keeping or achieving a healthy IT state (3) that are not yet implemented	

Components

Internalized Liabilities of Organizations

Strategic Debt		
Organizational Debt		
Process Debt		
Technical Debt	Functional Debt	Data (Quality) Debt

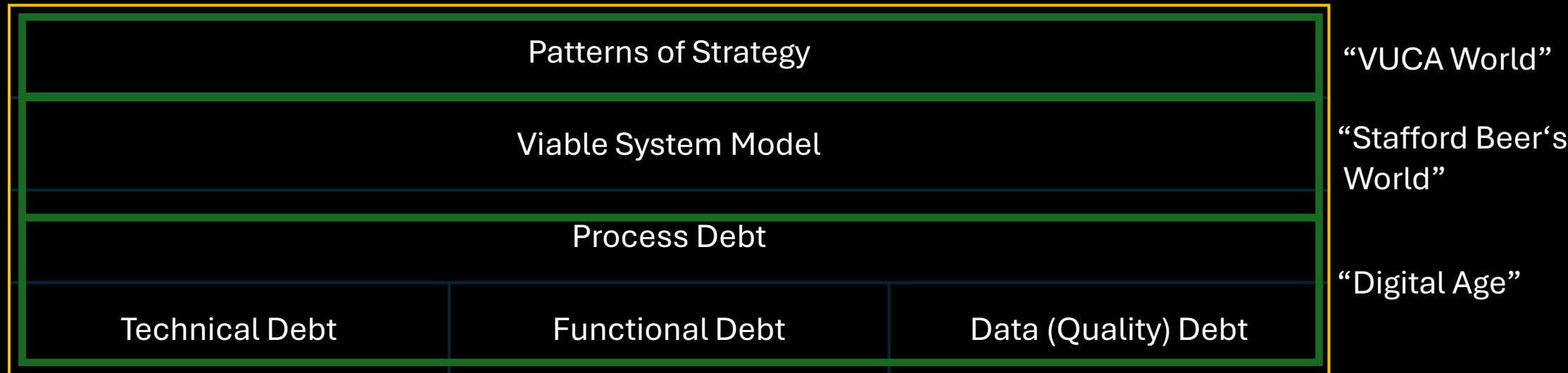
This can be operationalized

Major Claim: This model is complete!

in the sense that everything else finds its place on the balance sheet somewhere else – or is not material

VSM at the core of the whole approach

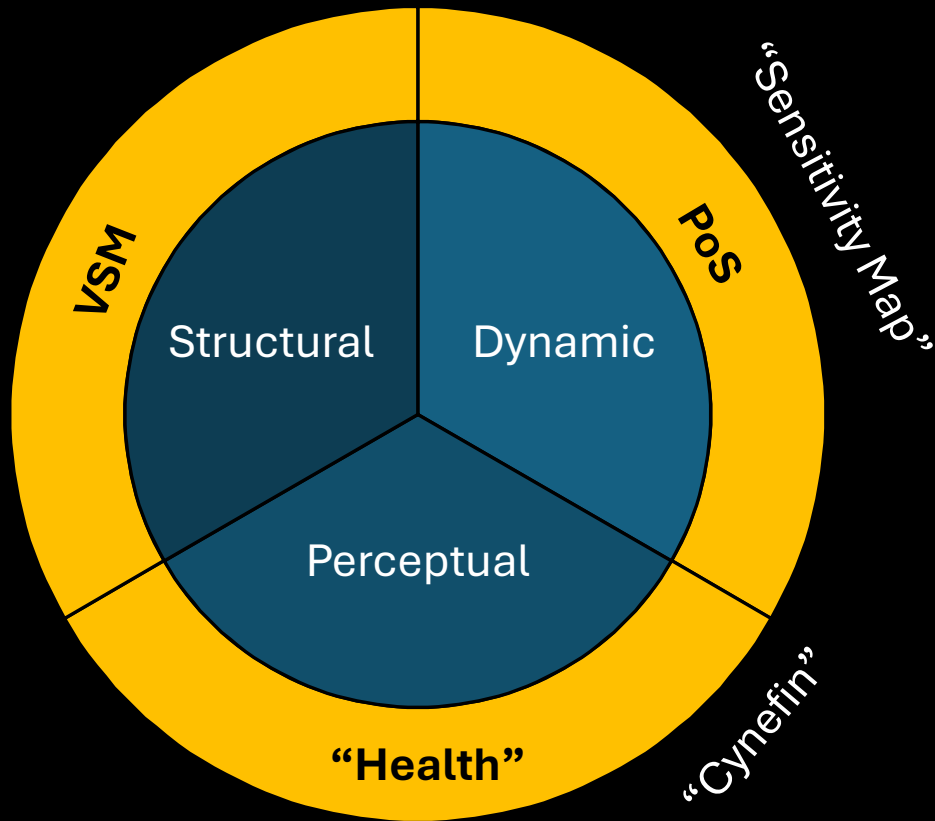
Internalized Liabilities of Organizations – Overview of Models



Major Claim: This model is complete!

in the sense that everything else finds its place on the balance sheet somewhere else – or is not material

Complexity



“Viable” does not mean **“Healthy”**

“Health” is a perception and perceptions can change i.e., they have their own **“dynamics”**

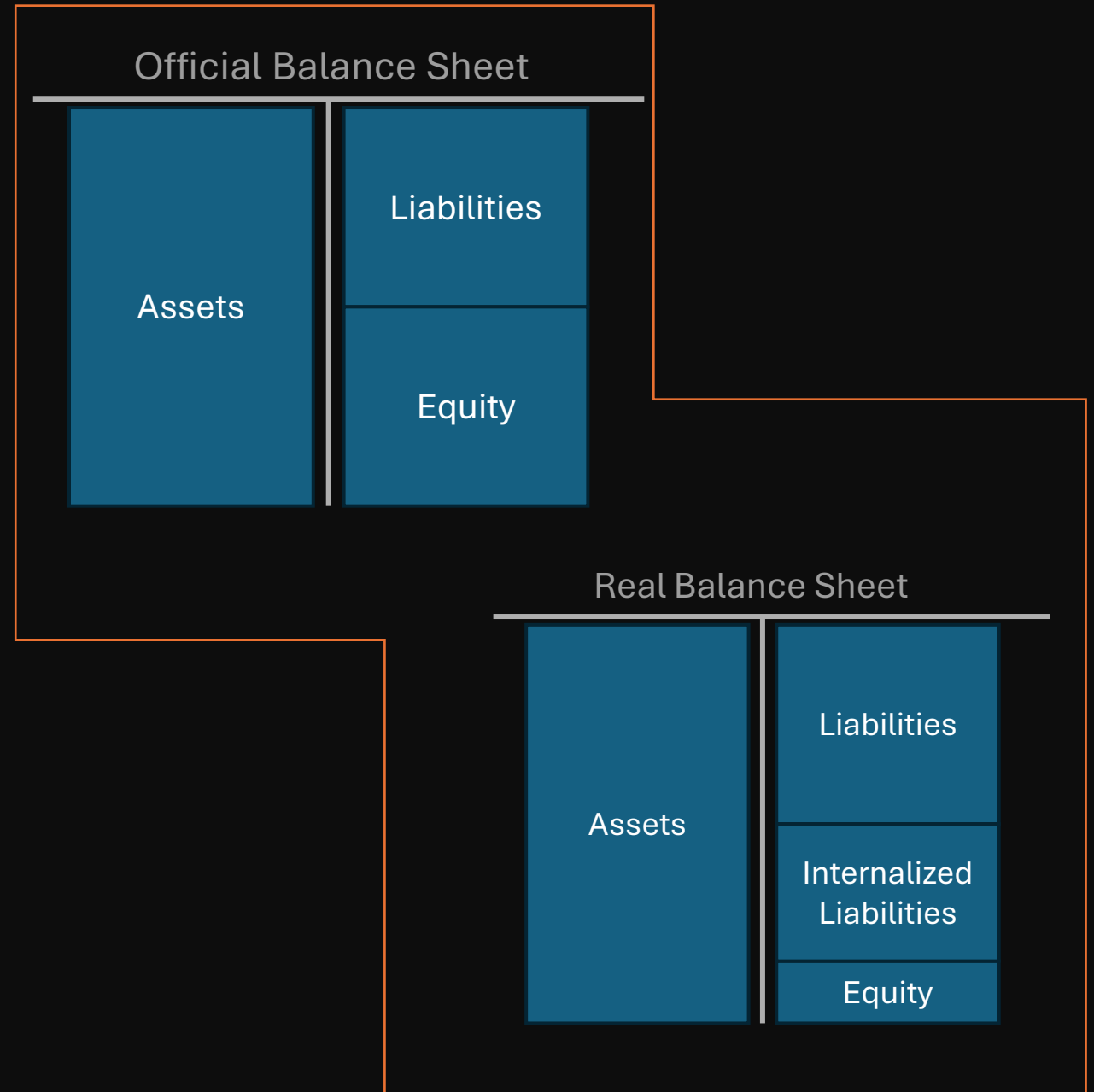
Internalized Liabilities (based on VSM and PoS) are “Complexity Complete”

Cynefin Framework



Source: www.thecynefin.co

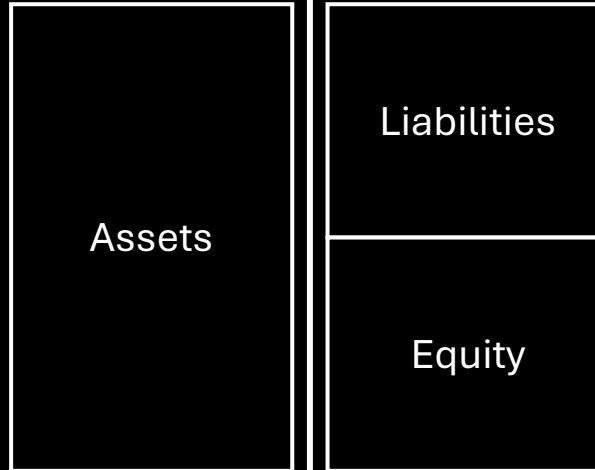
Internalized
Liabilities are not
accounted for
under “traditional”
accounting
regimes



Why does this work?

~14th century:

Modern Accounting Started

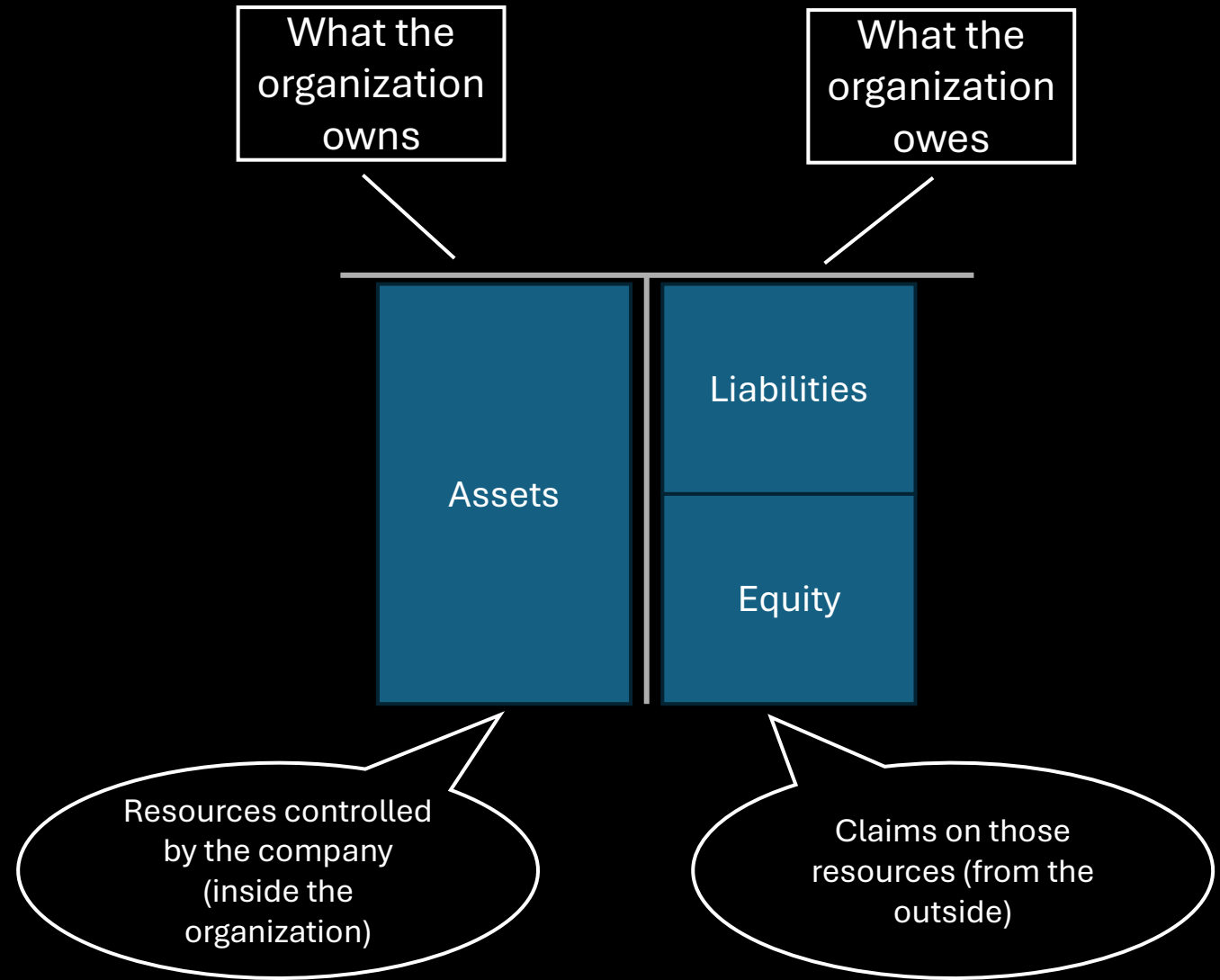


$$\text{Assets} = \text{Liabilities} + \text{Equity}$$

With roots back to Al-Khwarizmi (Persia)
813-833: "Al-Jabr" The Compendious Book on
Calculation by Completion and Balancing



The balance sheet describes the **boundary** of an organization and relationships between **inside** and **outside**



Just read the “grammar of systems” with a balance sheet in mind ...

Summary – Internalized Liabilities

- Obligations without allowance for building provisions
- Can be described via Technical Debt, ..., Strategic Debt
- A Financial view on the systemic “frictional costs” of running an organization
- Provide a “Management Language” for it

Open questions:

- Can this be modeled differently?
 - Probably yes – but I don’t know how
 - And if yes – it needs to be a “systemic approach”
- Why am I detecting this? (only now)
 - I am just putting existing material together

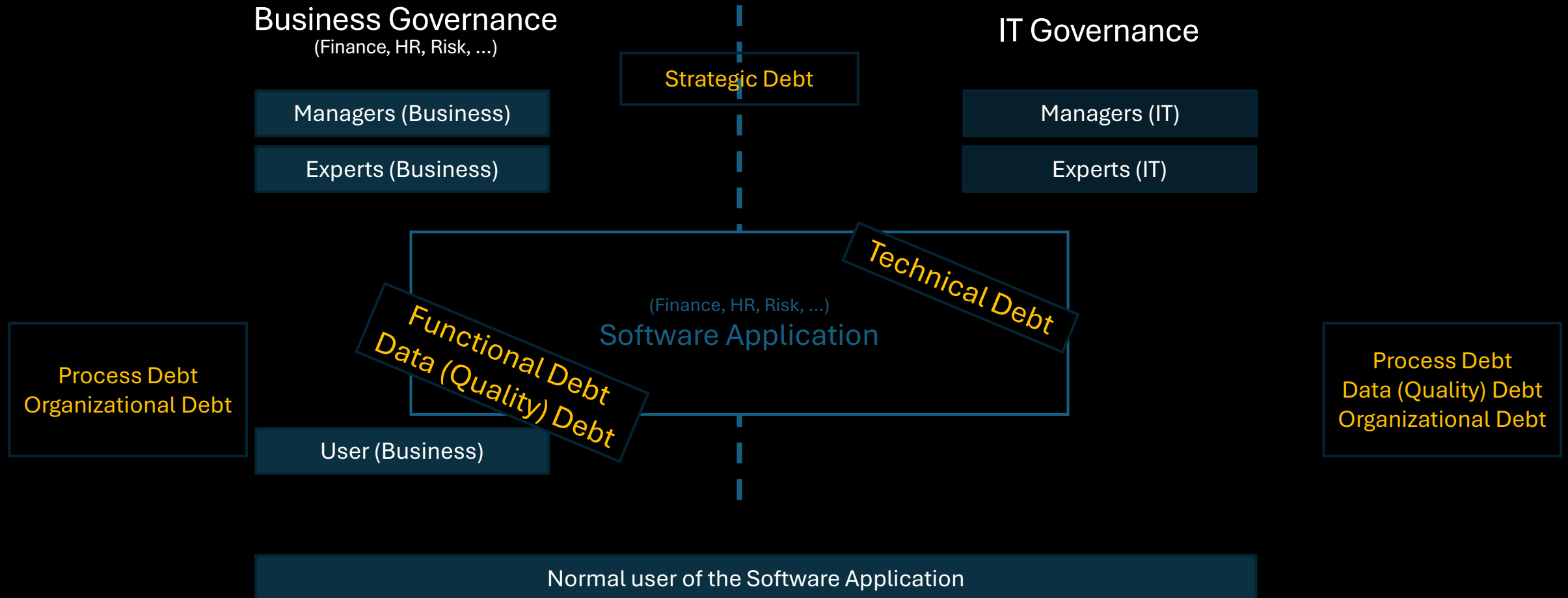
End of Second Iteration

Use Case 1

Change Process Business Unit & IT

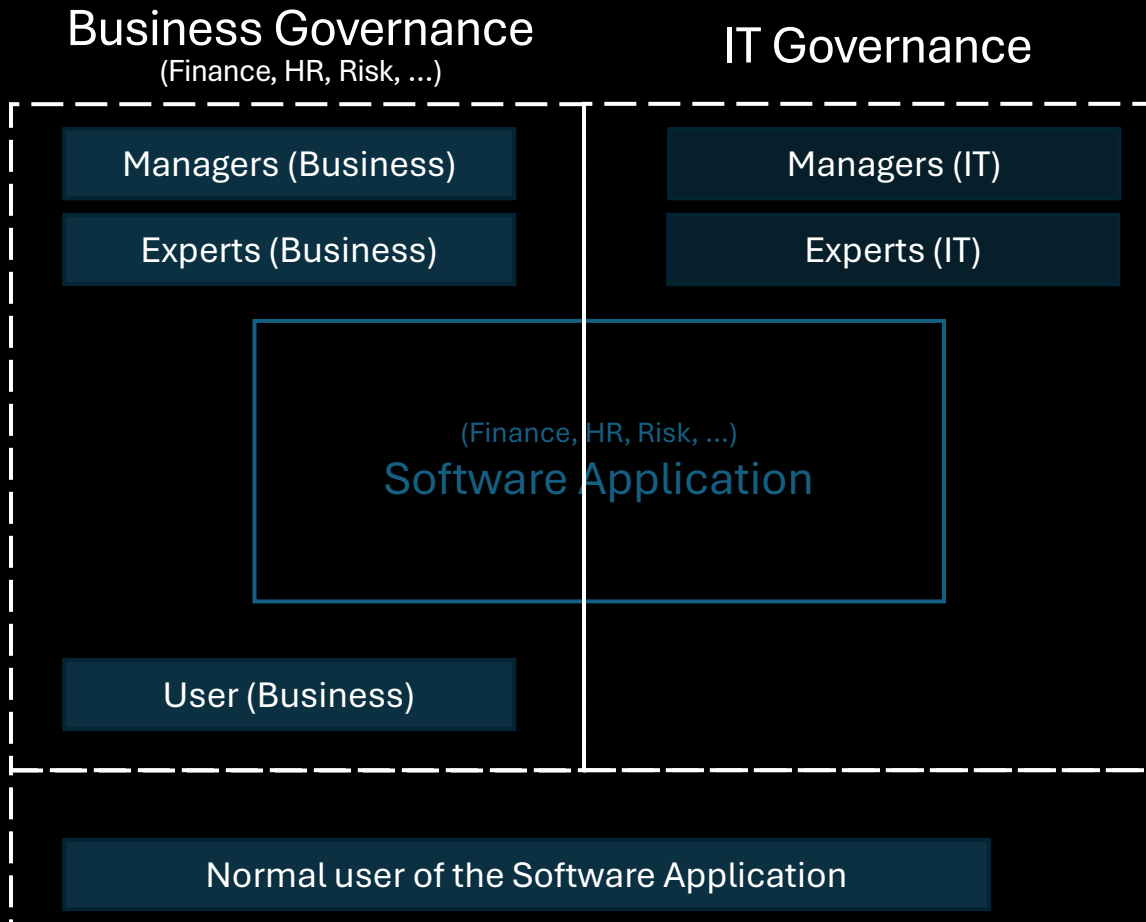
Change Process – Business Unit & IT

How to handle this complexity (in a good way)?

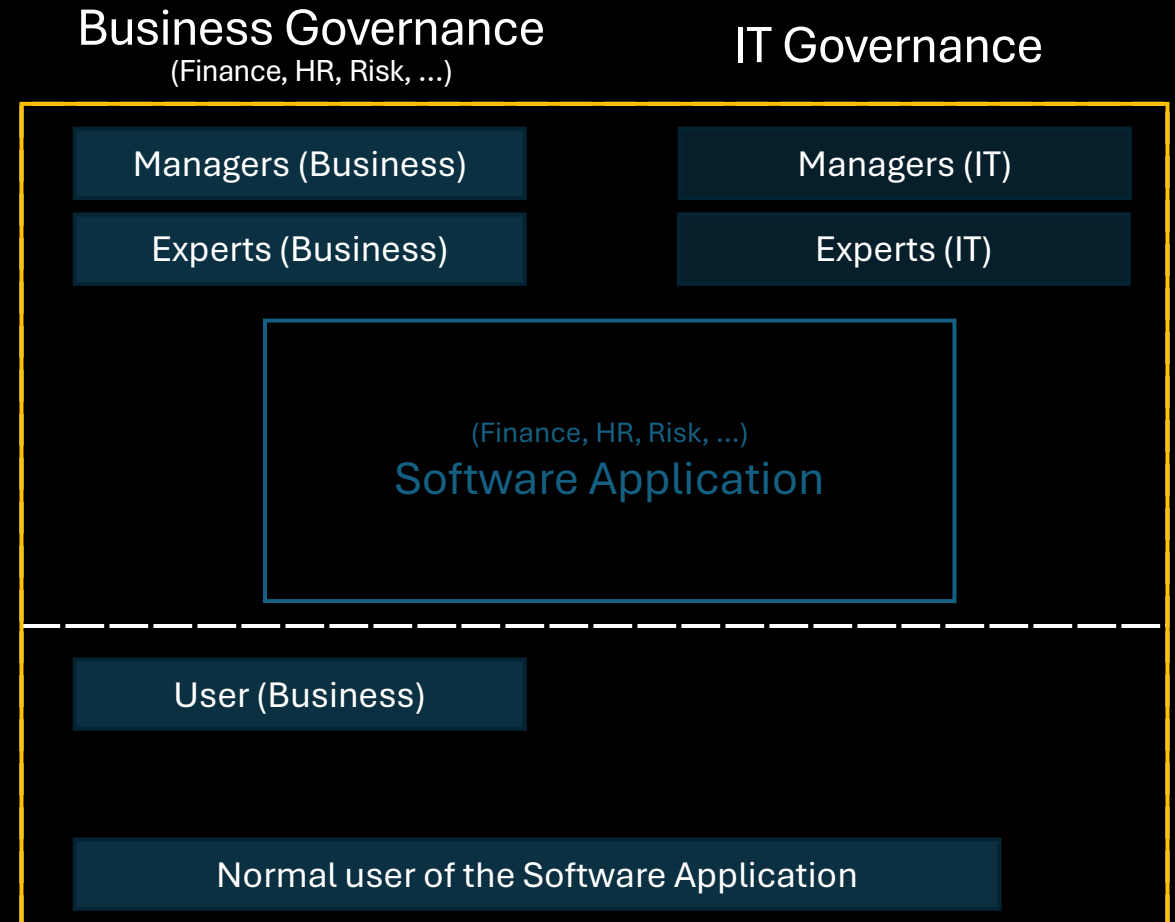


Alternatives

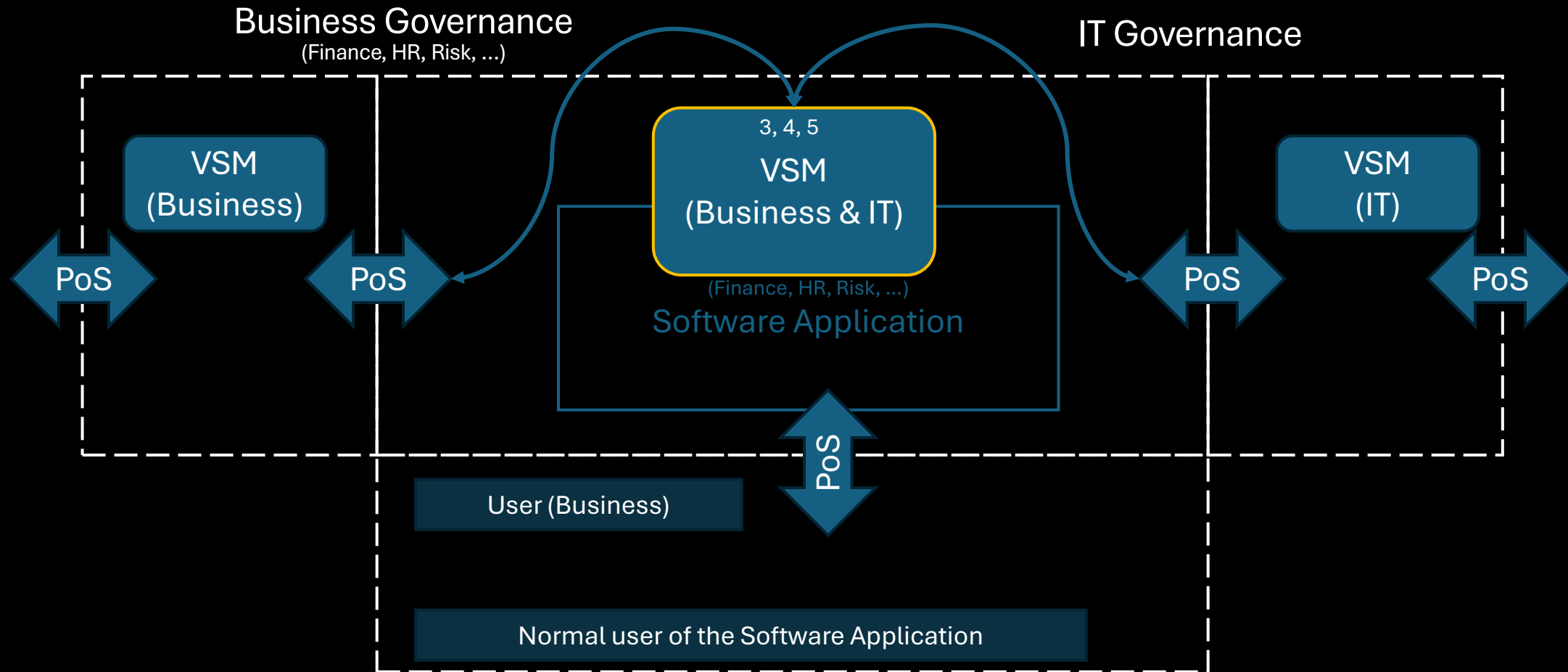
(A) OrgChart Focus



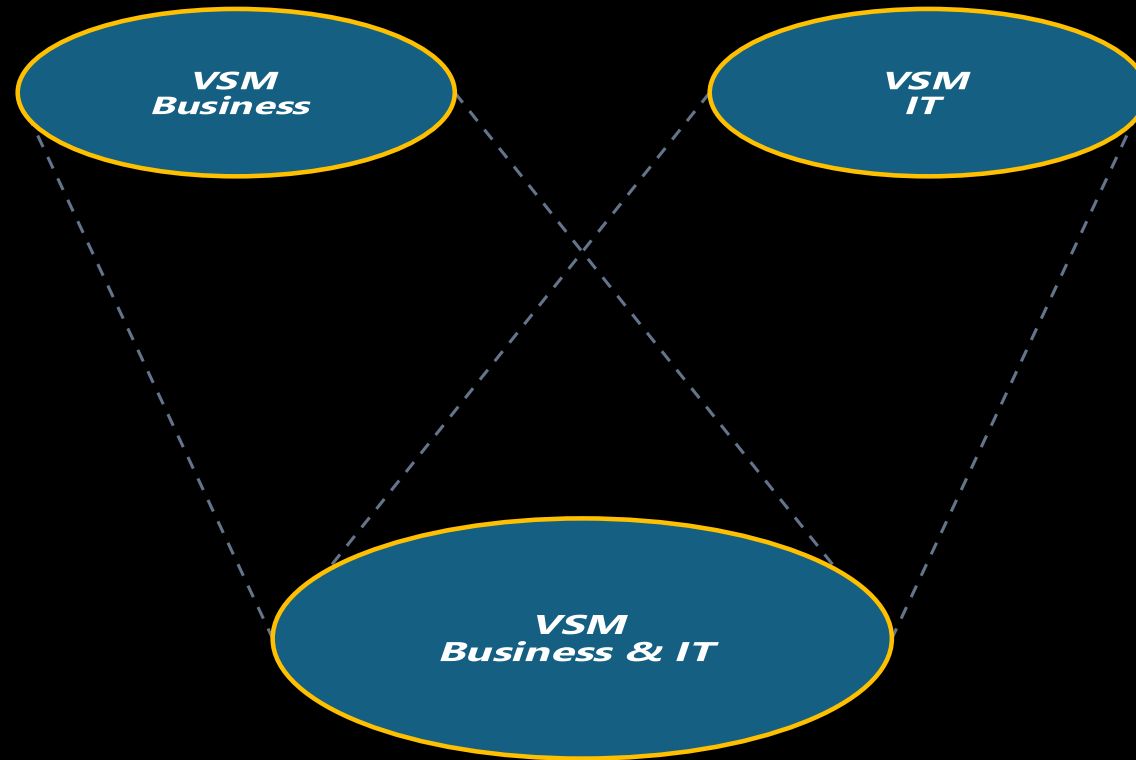
(B) User Focus



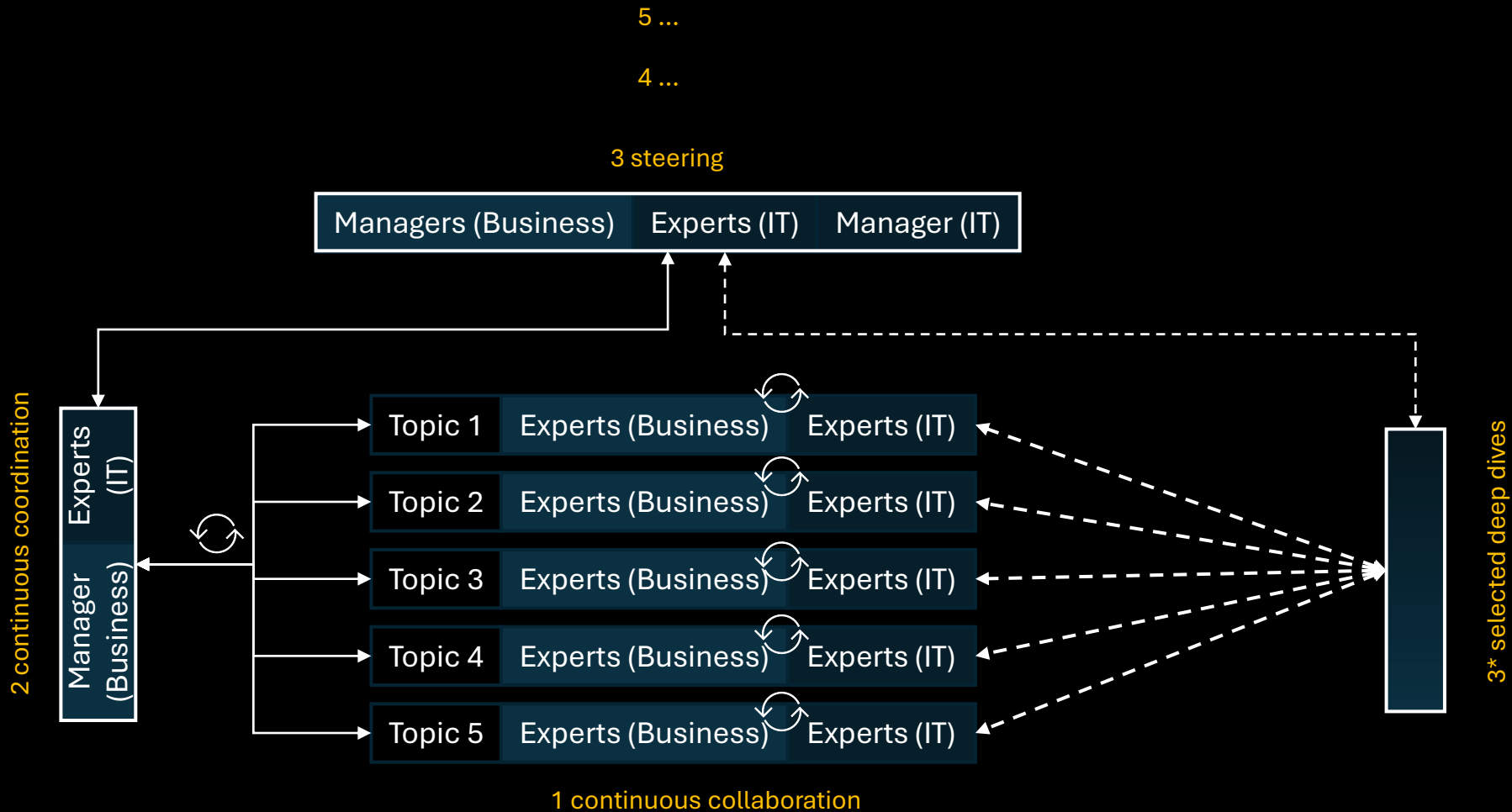
(B) User Focus – modeling w/VSM



(B) User Focus – modeling w/VSM



(B) VSM (Business & IT)



1, 2, 3, 3*, 4, 5 : Notation of the “Viable System Model” (VSM)

Use Case 2

Target Setting Process

Target Setting Process

Three questions to my team:

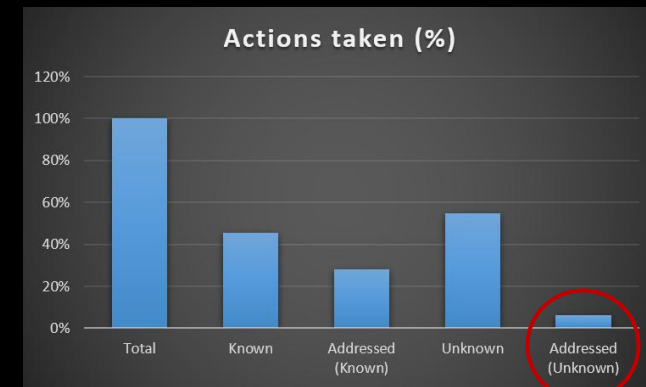
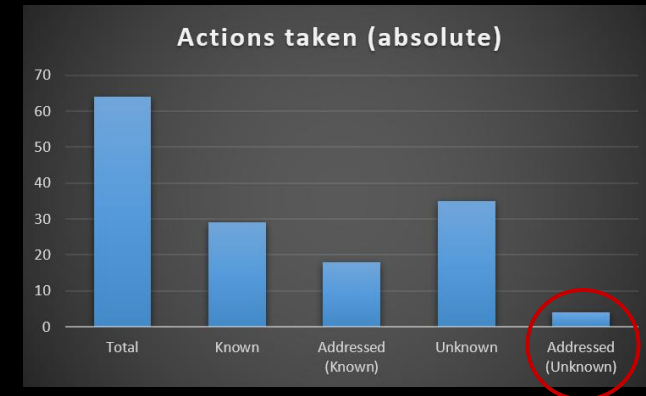
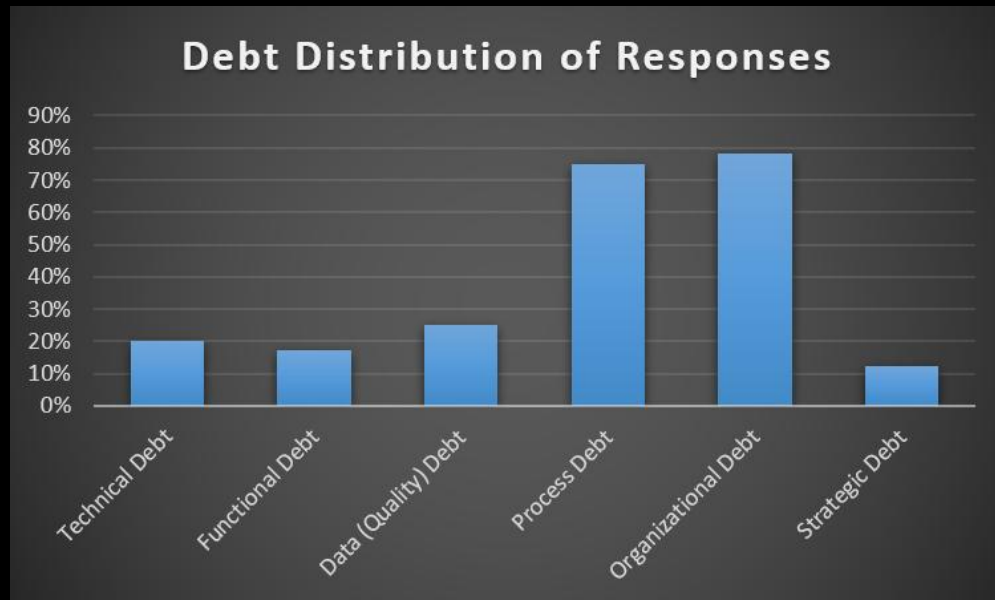
- Which targets are missing?
- Which impediments are in our way?
- What would help with, achieving our targets?

Expectation:

The answers should give valuable insights into
Internalized Liabilities

Early Results

(No statistical validity, just a “try run”)



Conjectures and Implications

Conjecture

Internalized Liabilities > 0 (always)

implies that “AI takes over” will not happen and Elon Musk’s “world of abundance” is impossible 😊 at least for most of us

Worth mentioning:

Internalized Liabilities = 0 is something that can be checked, AI has a target now