

# Optimizing for a Fast Flow of Value w/ Adaptive, Socio-Technical Systems

Susanne Kaiser

Independent Tech Consultant

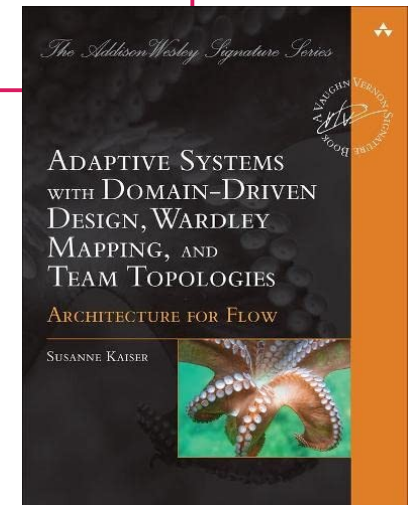
@suksr



## Warning: May cause information overload

This presentation contains a substantial amount of content. Please be advised that the sheer volume of information may be overwhelming.

At the end of the presentation, a downloadable link to the complete slide deck will be provided for your reference. Additionally, for those interested in an in-depth exploration of the topics covered, I am in the process of writing a book that delves deeper into the subject matter.



# Challenges of Building Systems

Building the **right** thing

How aligned is our solution to business / user needs?

Have we understood the problem?

Do we share the same common understanding?

Effectiveness

Building the **thing** right

How efficient are our engineering practices?

How fast can we deliver changes?

How easy and fast can we change and adapt?

Efficiency

# Challenges of Building Systems

Building the **right** thing

How aligned is our solution to business / user needs?

Have we understood the problem?

Do we share the same common understanding?

Effectiveness

“Doing the wrong thing right is not nearly as good as doing the right thing wrong”

Dr. Russell Ackoff

Building the thing **right**

How efficient are our engineering practices?

How fast can we deliver changes?

How easy and fast can we change and adapt?

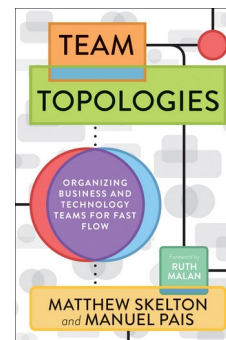
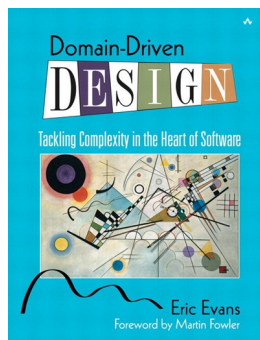
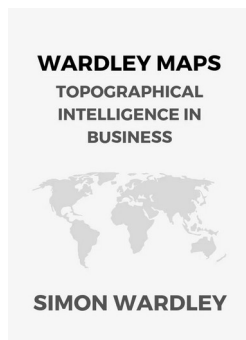
Efficiency

“Crap delivered quickly (and sustainably) is still crap, (unless we learn something, and act on that learning..)”

John Cutler

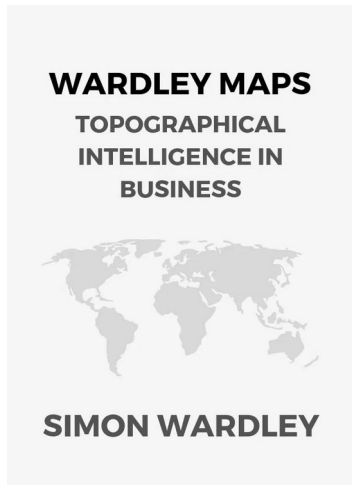
Building the right thing right

Building adaptive,  
socio-technical systems  
optimized for a fast  
flow of value & feedback

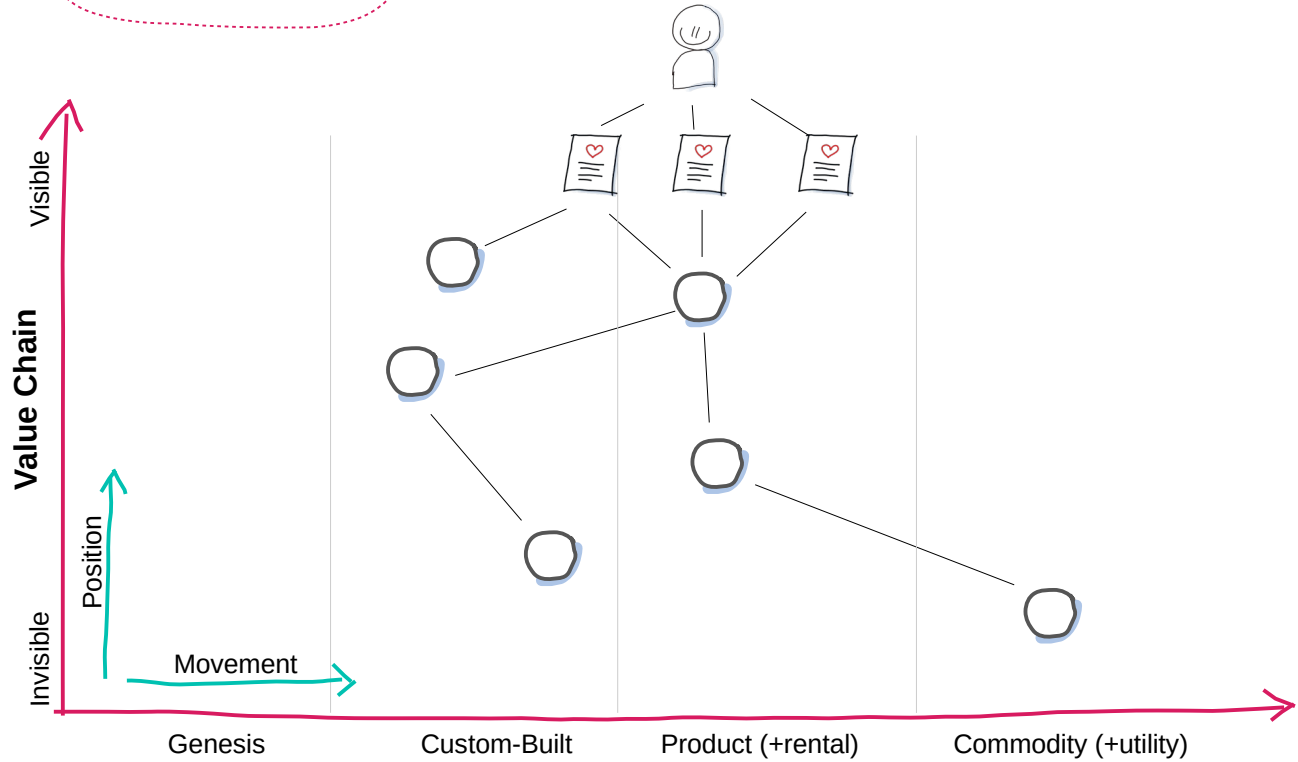


Architecture for Flow

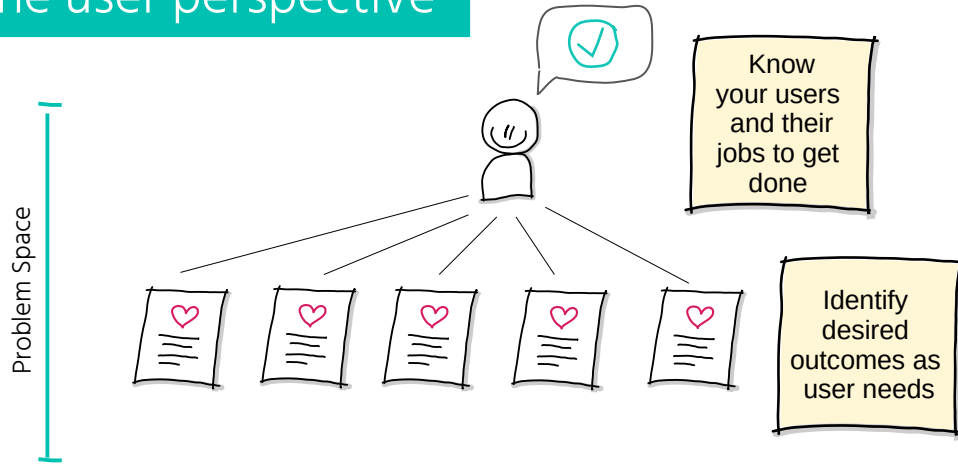
Visualising  
the business  
landscape



# Wardley Map



# Starting from the user perspective



Building the  
right thing

“People buy products and services to get a job done.”

Clayton Christensen

# Starting from the user perspective

Facilitate remote learning

Online School



Problem Space

Teachers



Know your users and their jobs to get done



Prepare learning material



Assess student understanding



Track progress



Facilitate communication



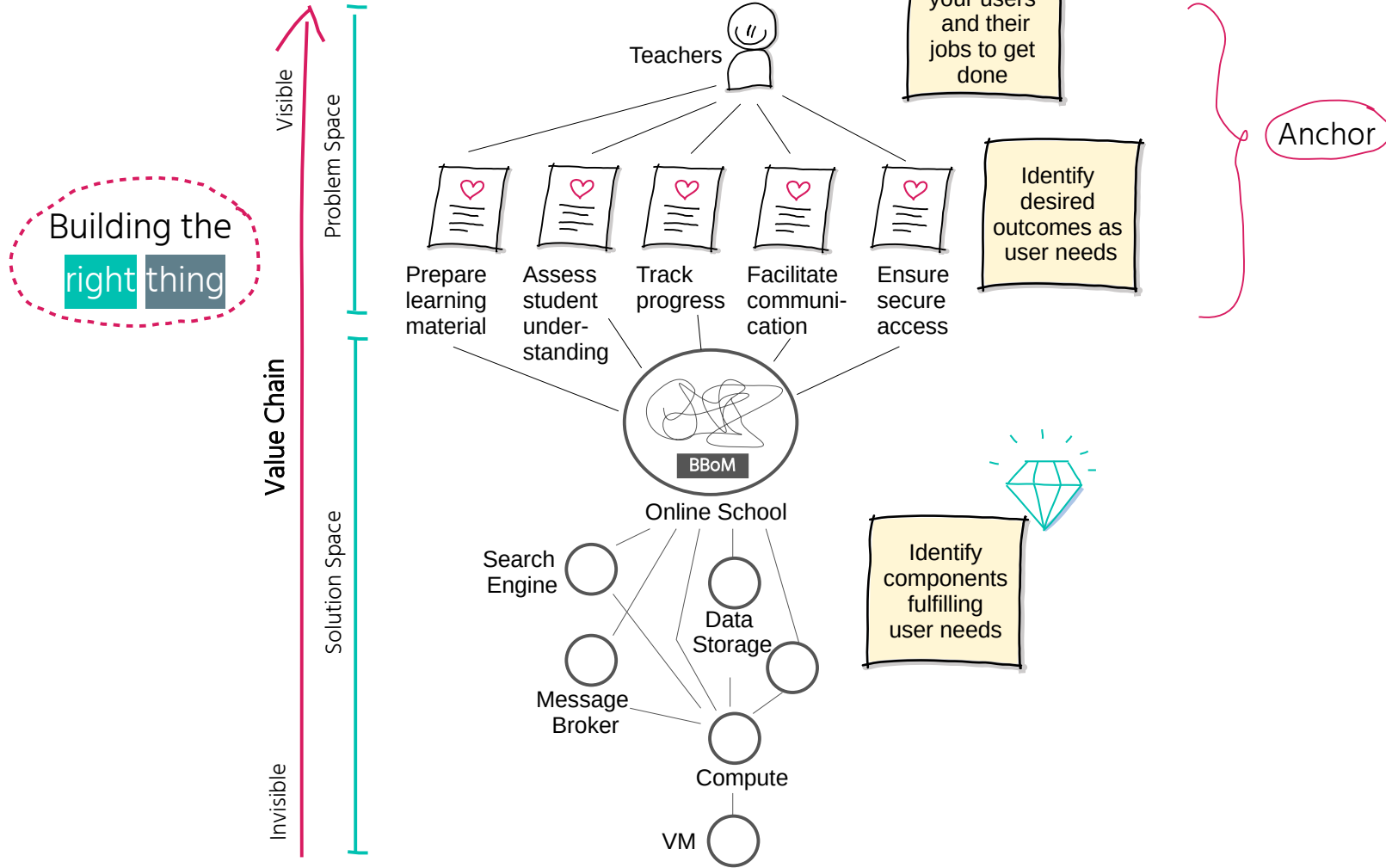
Ensure secure access

Identify desired outcomes as user needs

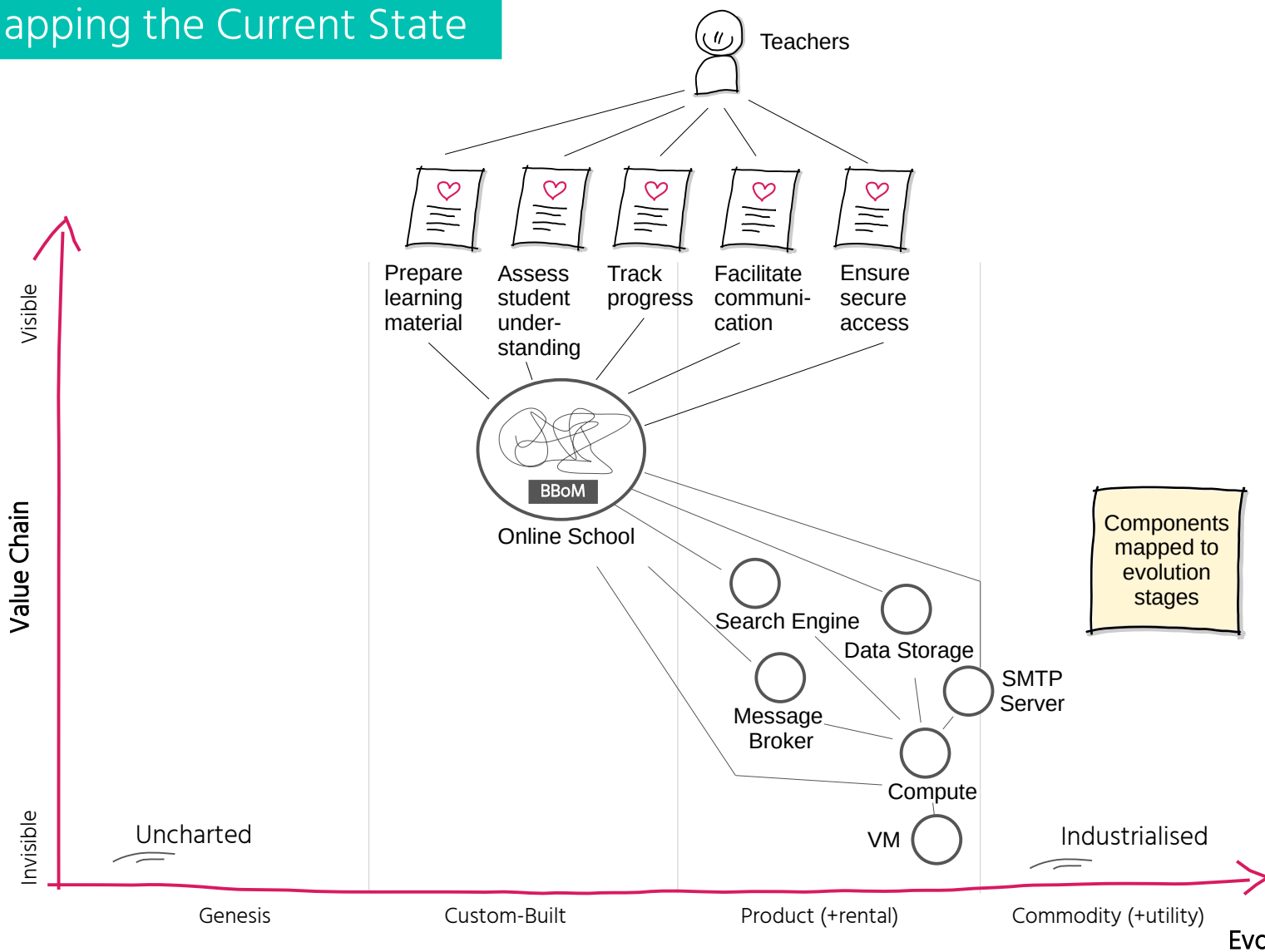
Building the right thing



# Understanding the value chain



# Mapping the Current State

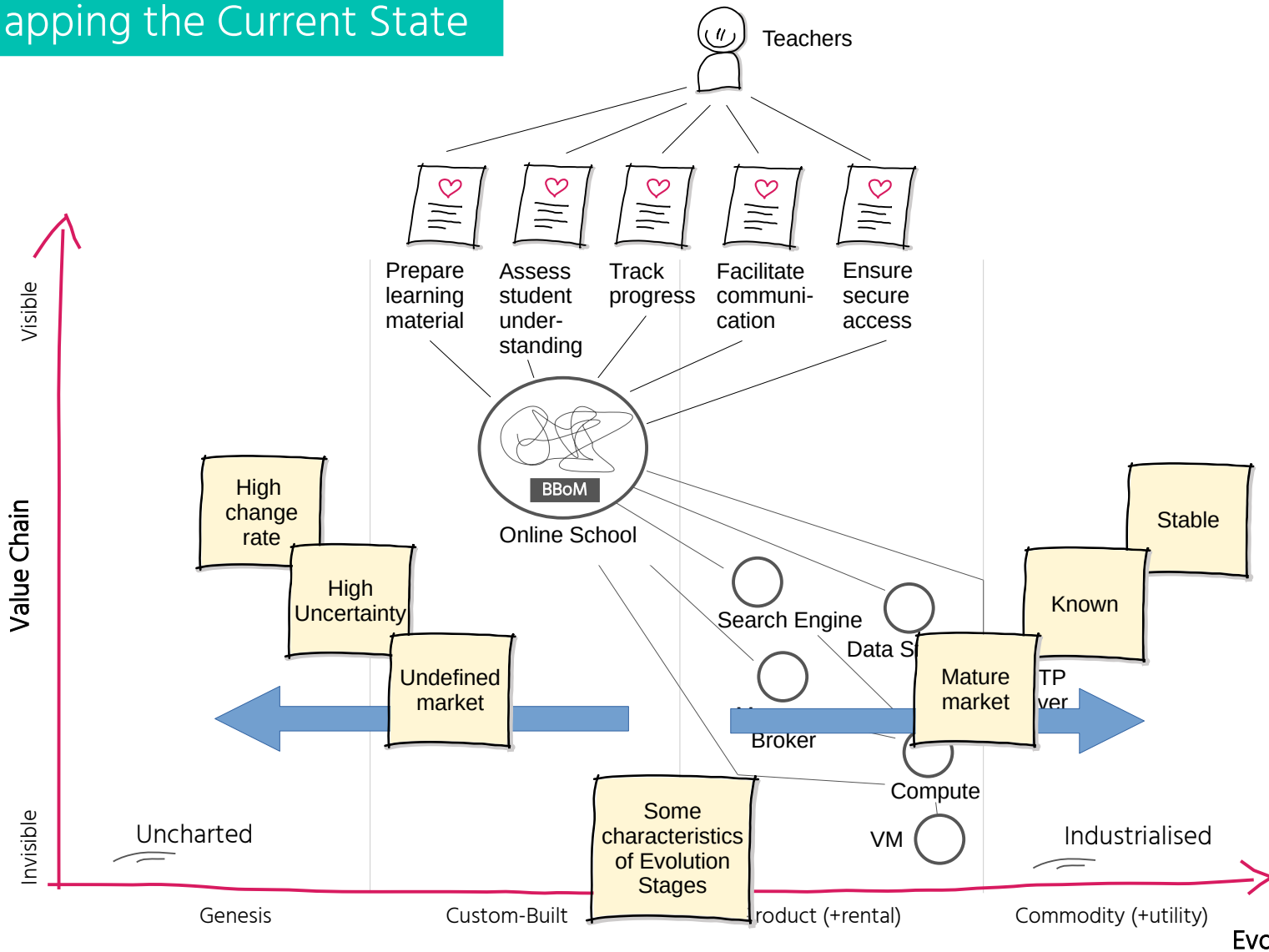


**WARDLEY MAPS**  
TOPOGRAPHICAL  
INTELLIGENCE IN  
BUSINESS



**SIMON WARDLEY**

# Mapping the Current State

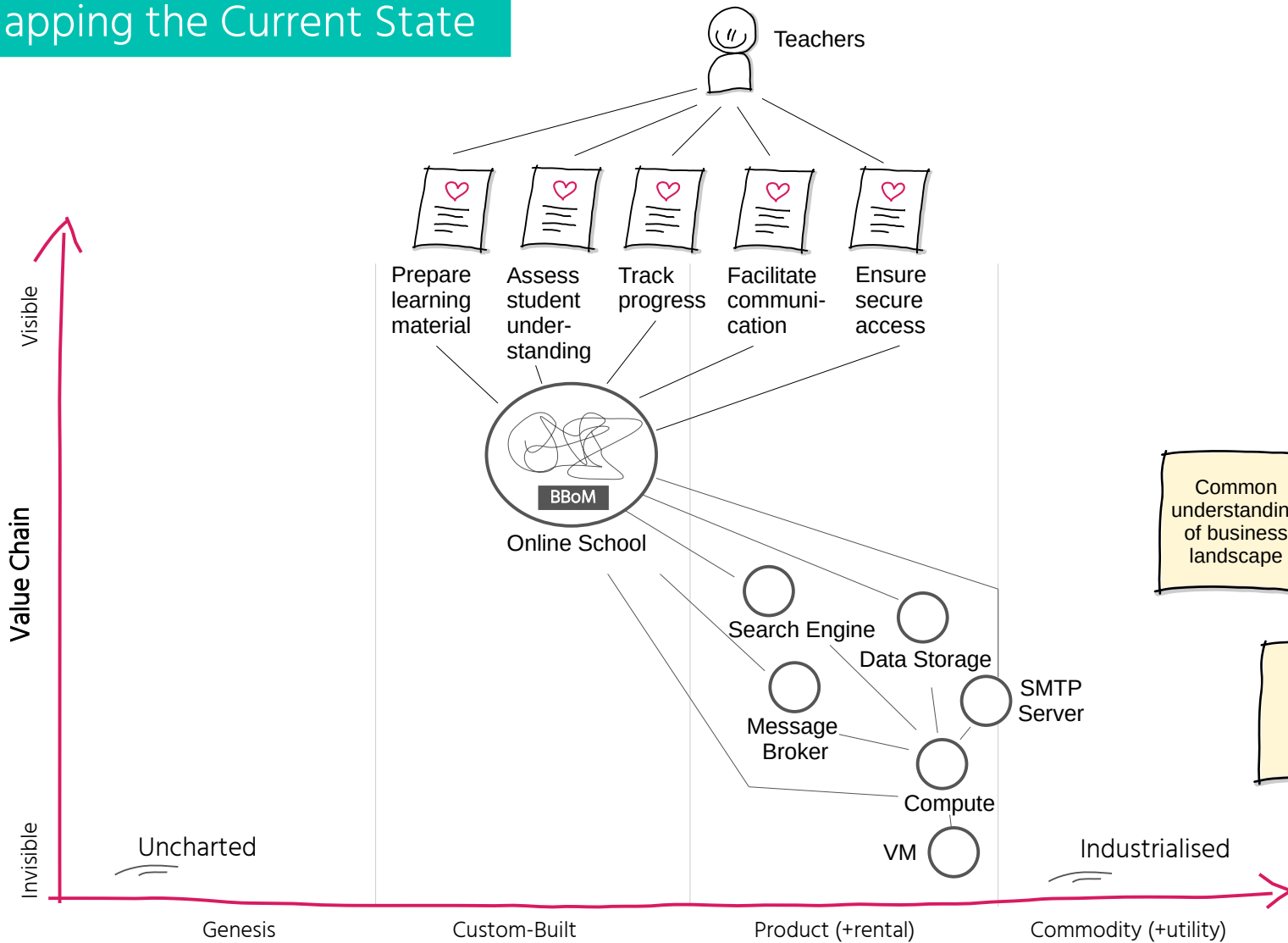


**WARDLEY MAPS**  
TOPOGRAPHICAL  
INTELLIGENCE IN  
BUSINESS



**SIMON WARDLEY**

# Mapping the Current State



**WARDLEY MAPS**  
TOPOGRAPHICAL  
INTELLIGENCE IN  
BUSINESS



**SIMON WARDLEY**

Common understanding of business landscape

Where to invest?

What to evolve?

# Assessing Current Flow of Change

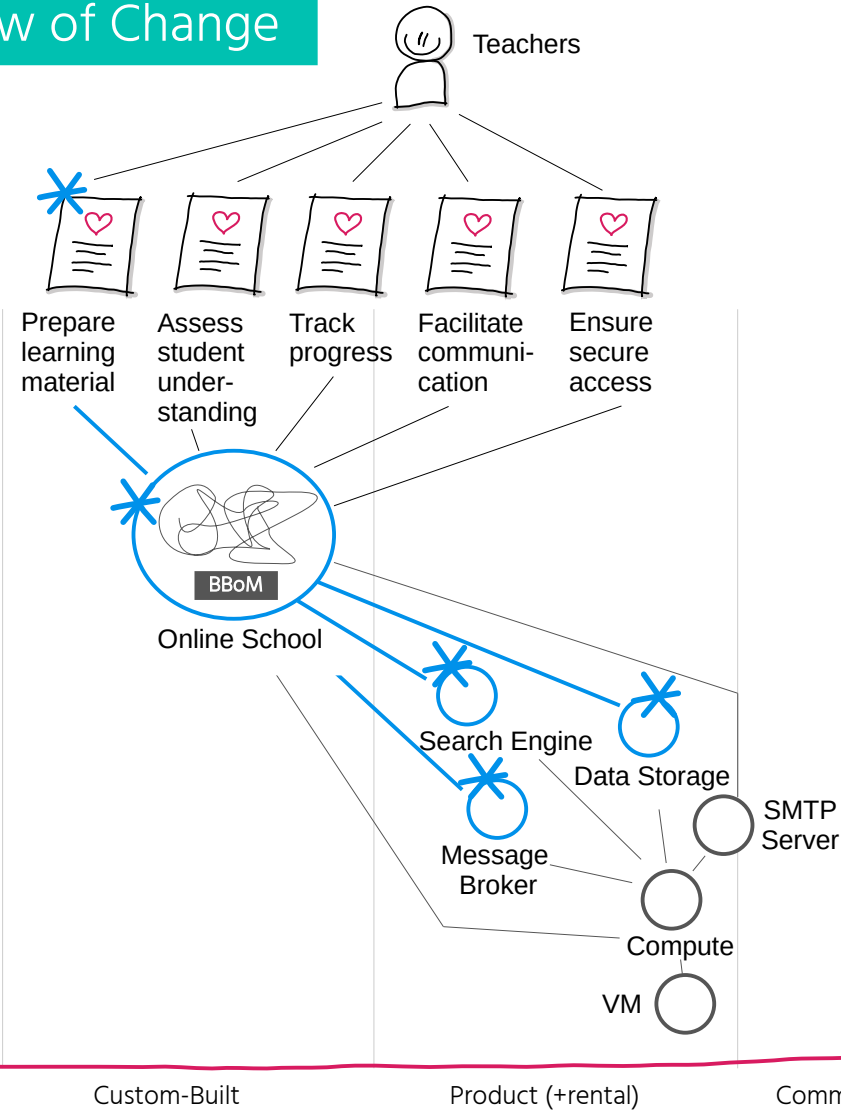
Building the thing right

Visible

Value Chain

Invisible

Assessing how a change would flow through our system



Genesis

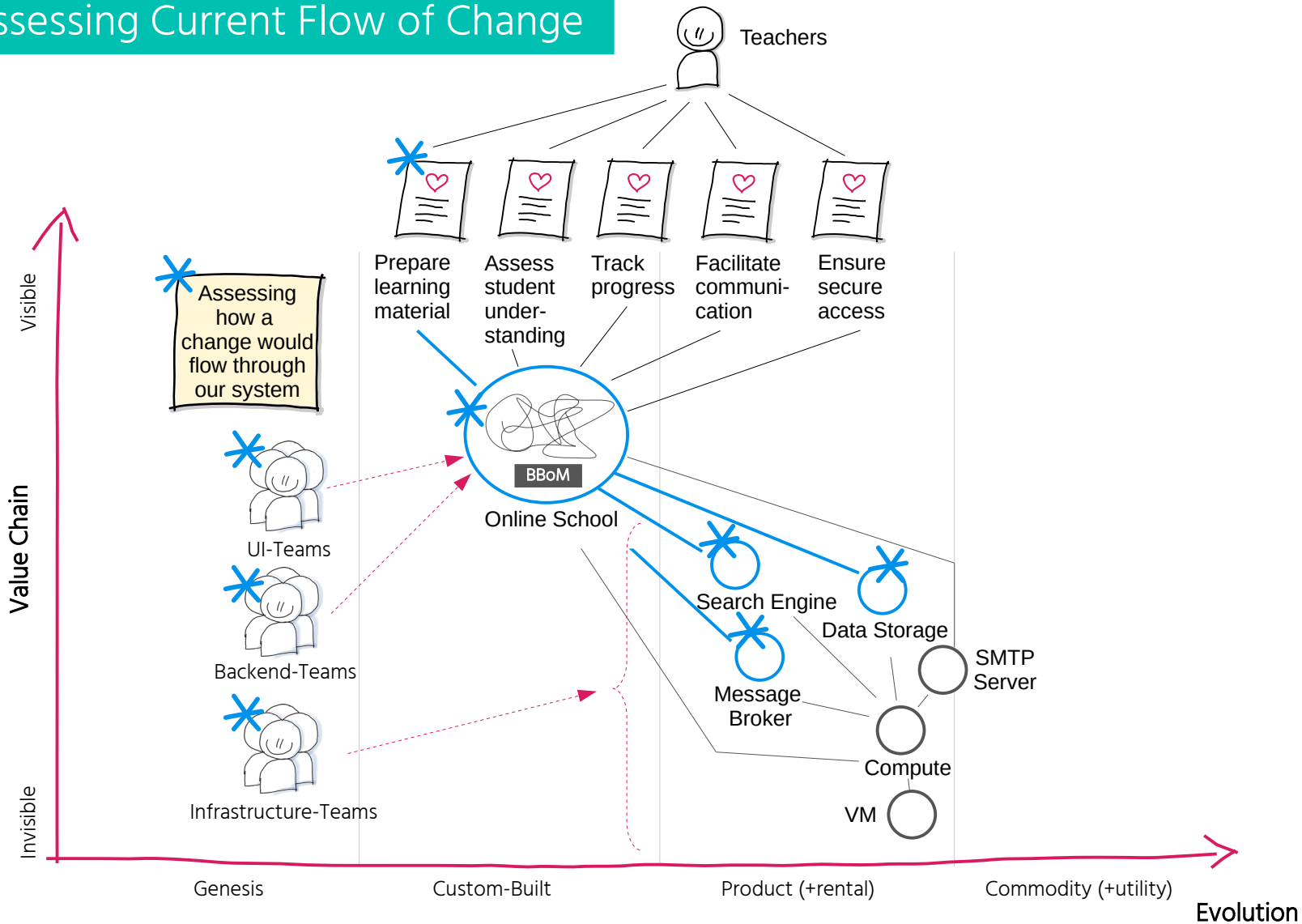
Custom-Built

Product (+rental)

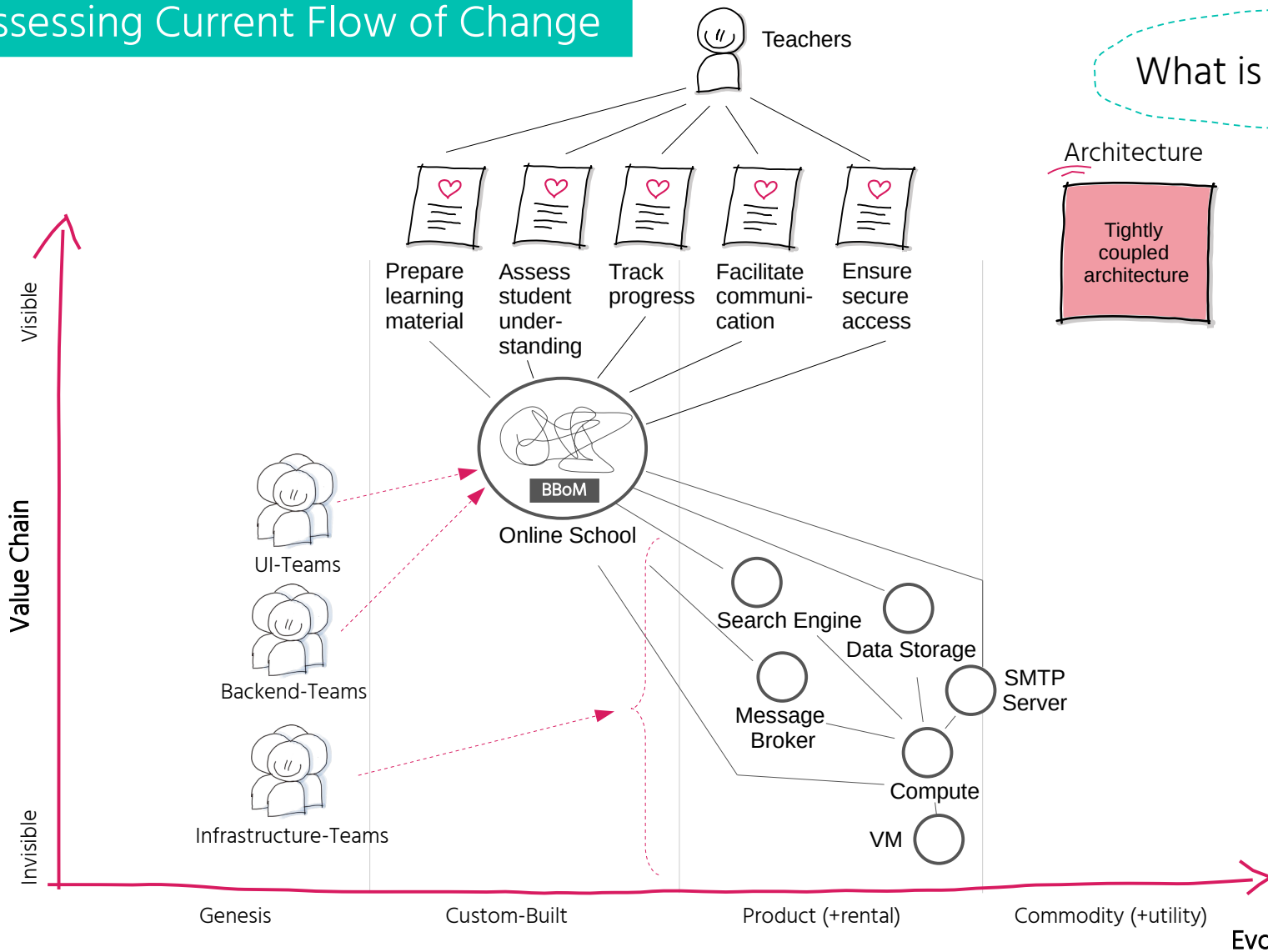
Commodity (+utility)

Evolution

# Assessing Current Flow of Change

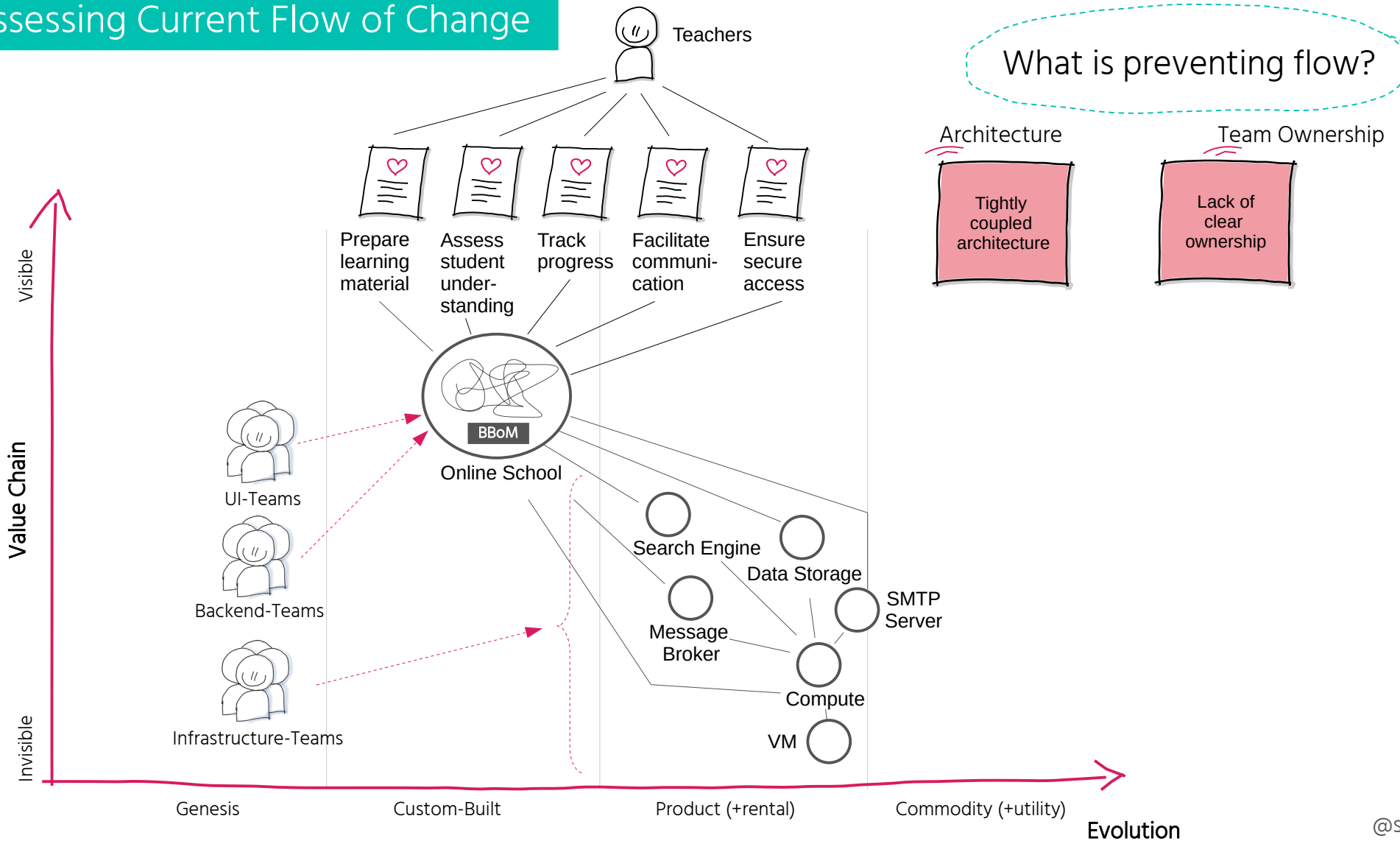


# Assessing Current Flow of Change



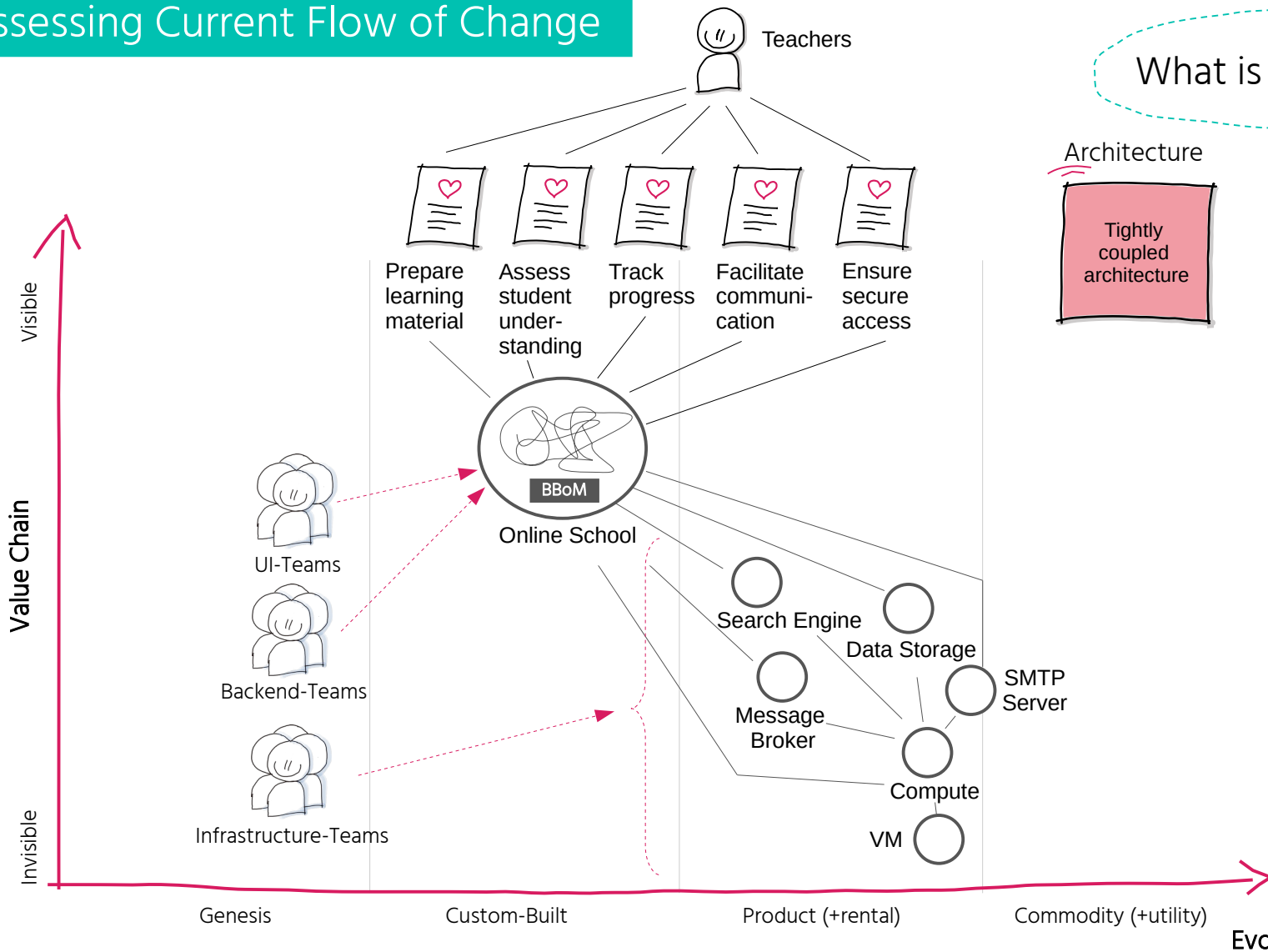
What is preventing flow?

# Assessing Current Flow of Change





# Assessing Current Flow of Change



What is preventing flow?

Architecture

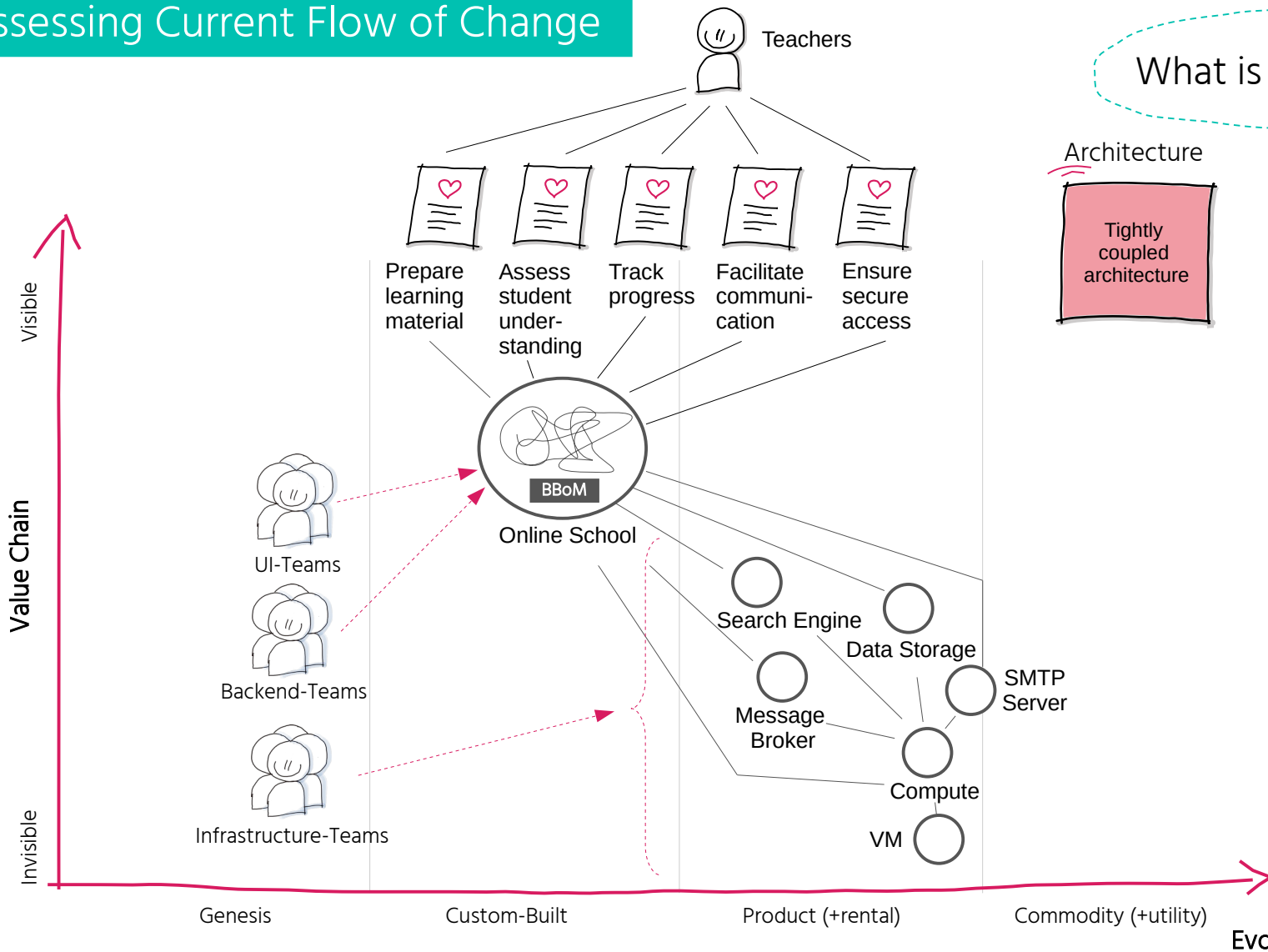
Tightly coupled architecture

Team Ownership

Lack of clear ownership

Teams own too many/large parts of the system

# Assessing Current Flow of Change



What is preventing flow?

Architecture

Tightly coupled architecture

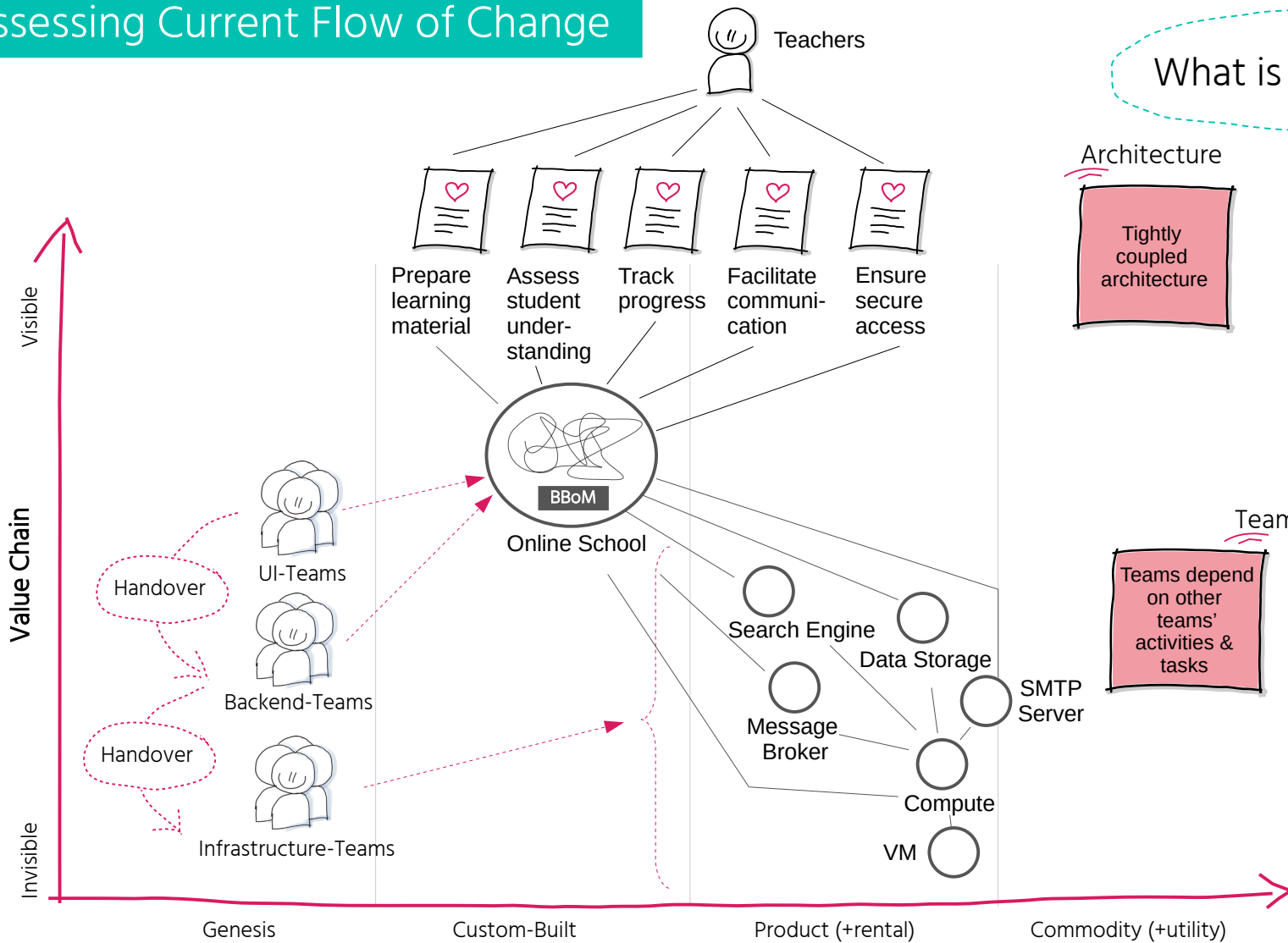
Team Ownership

Lack of clear ownership

Teams own too many/large parts of the system

High team cognitive load

# Assessing Current Flow of Change



What is preventing flow?

Architecture

Tightly coupled architecture

Team Ownership

Lack of clear ownership

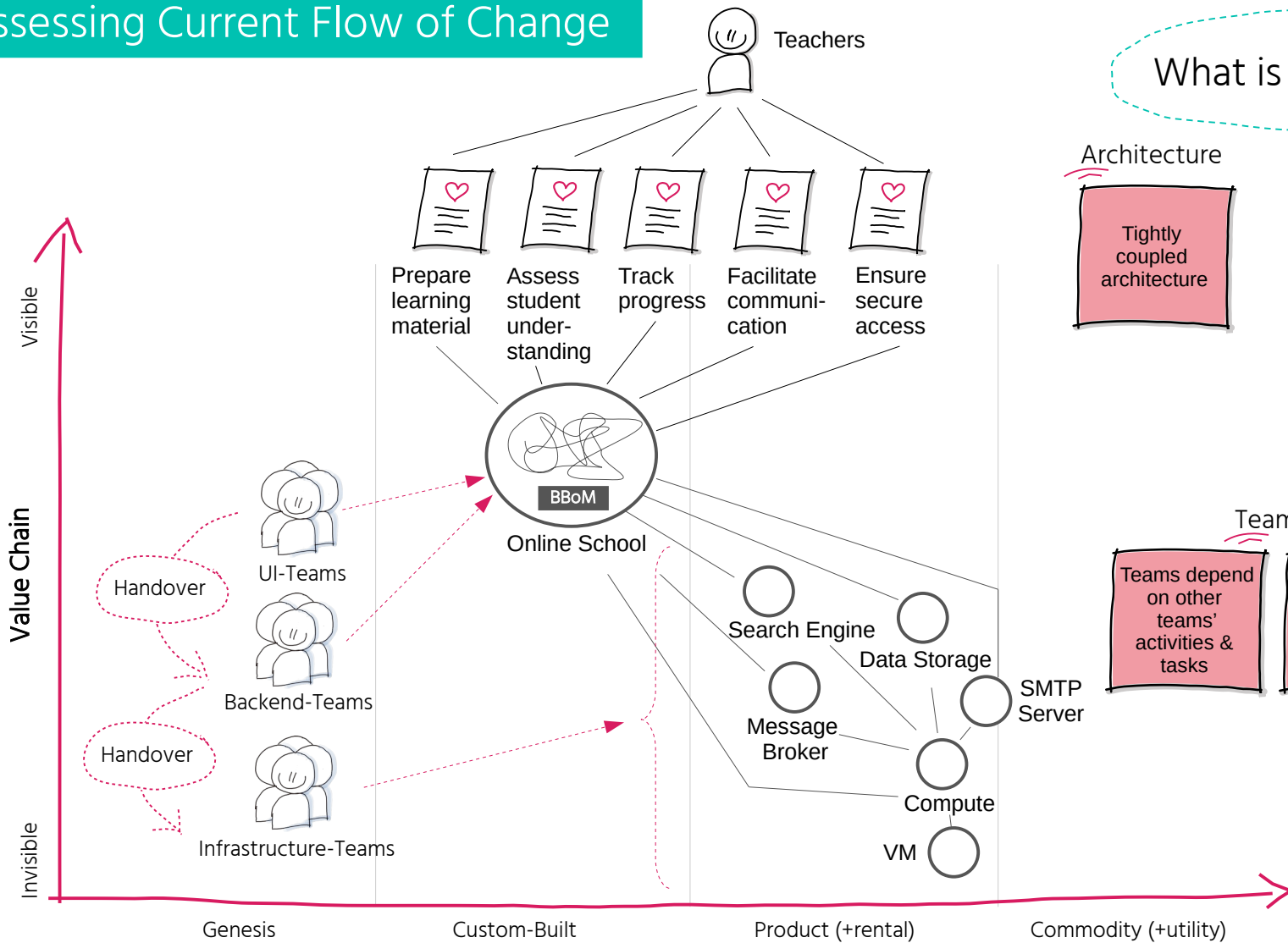
Teams own too many/large parts of the system

High team cognitive load

Team Dependencies

Teams depend on other teams' activities & tasks

# Assessing Current Flow of Change



## What is preventing flow?

### Architecture

Tightly coupled architecture

### Team Ownership

Lack of clear ownership

Teams own too many/large parts of the system

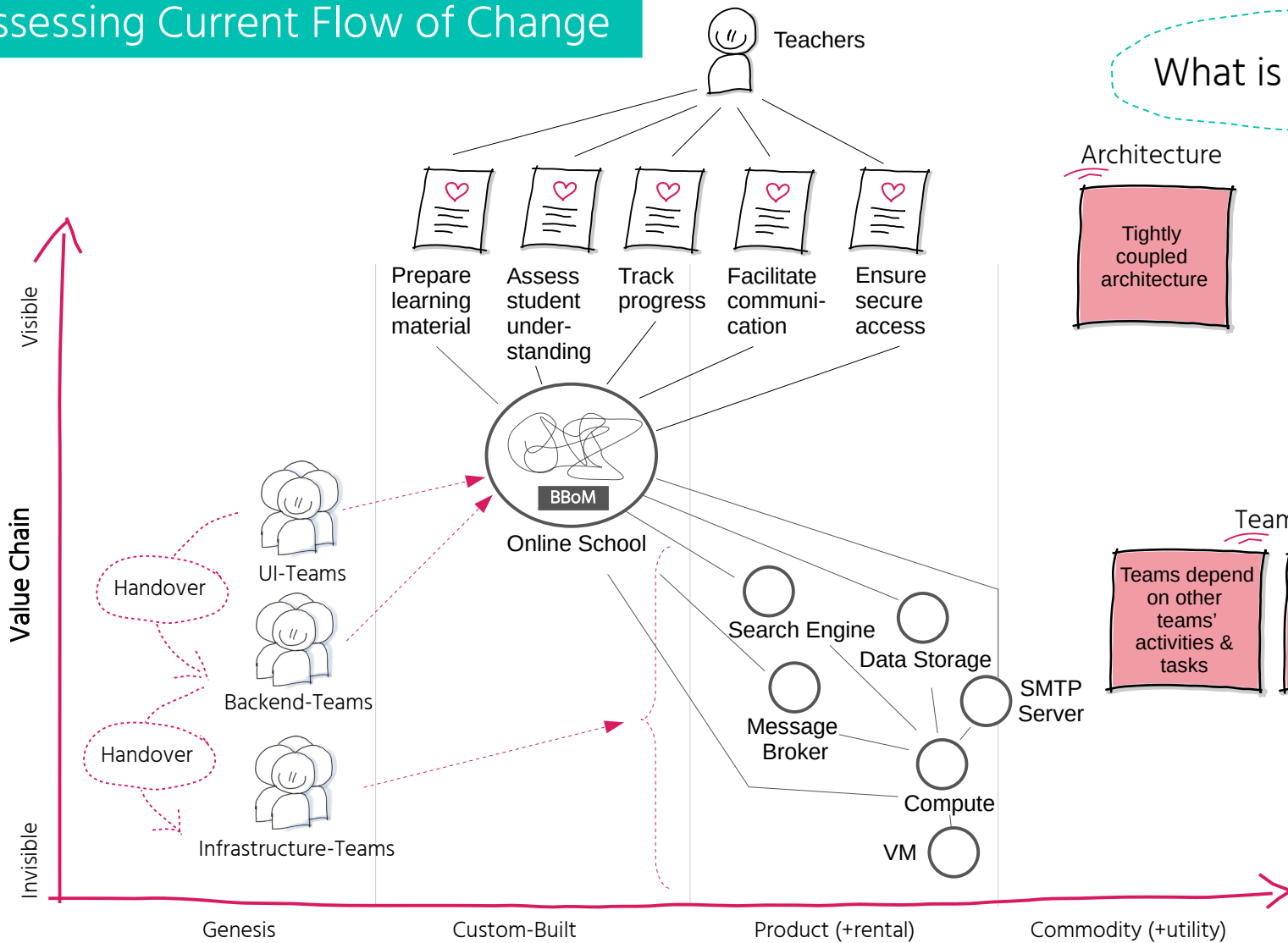
High team cognitive load

### Team Dependencies

Teams depend on other teams' activities & tasks

Teams depend on other people's specific expertise

# Assessing Current Flow of Change



## What is preventing flow?

### Architecture

Tightly coupled architecture

### Team Ownership

Lack of clear ownership

Teams own too many/large parts of the system

High team cognitive load

### Team Dependencies

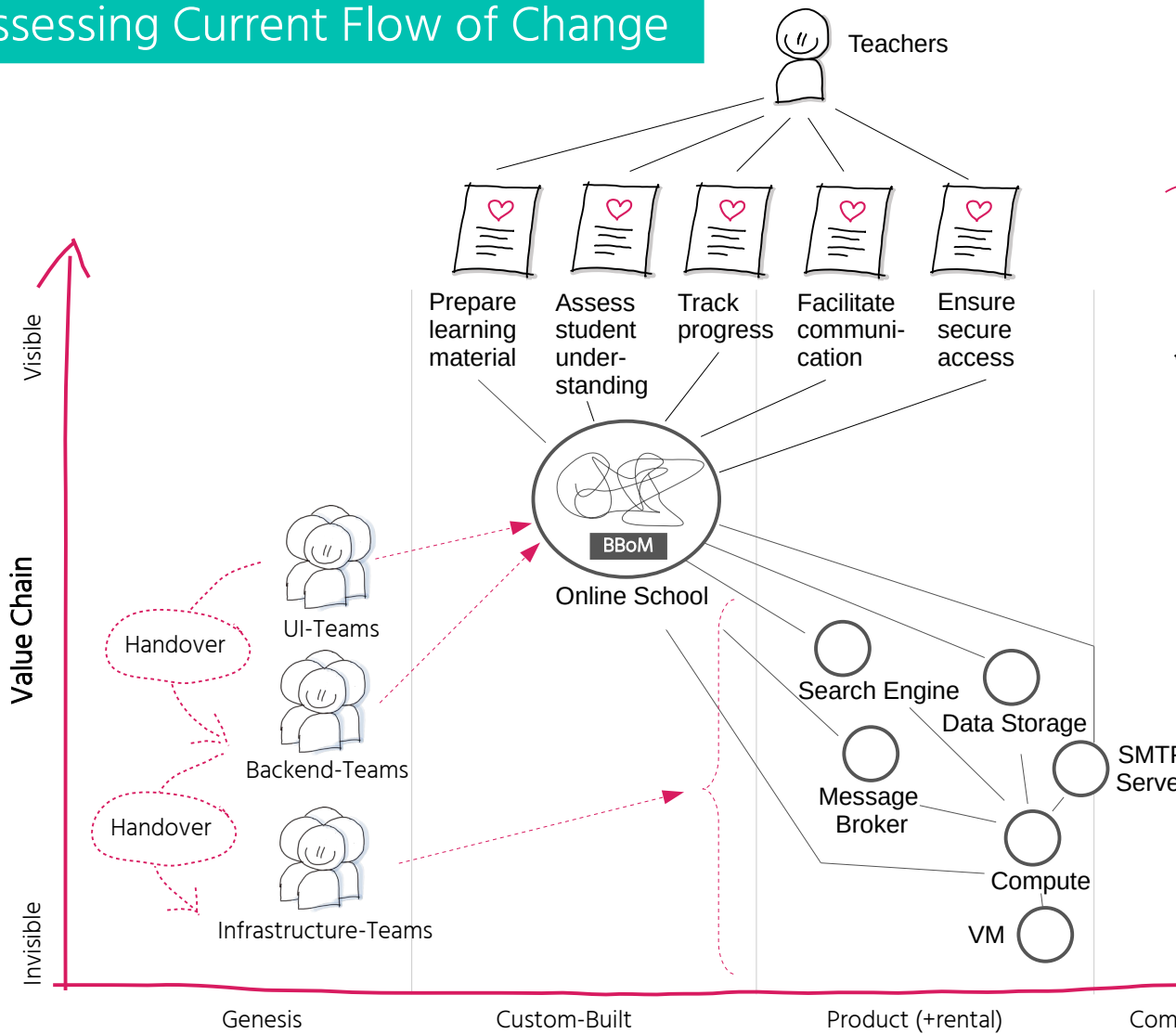
Teams depend on other teams' activities & tasks

Teams depend on other people's specific expertise

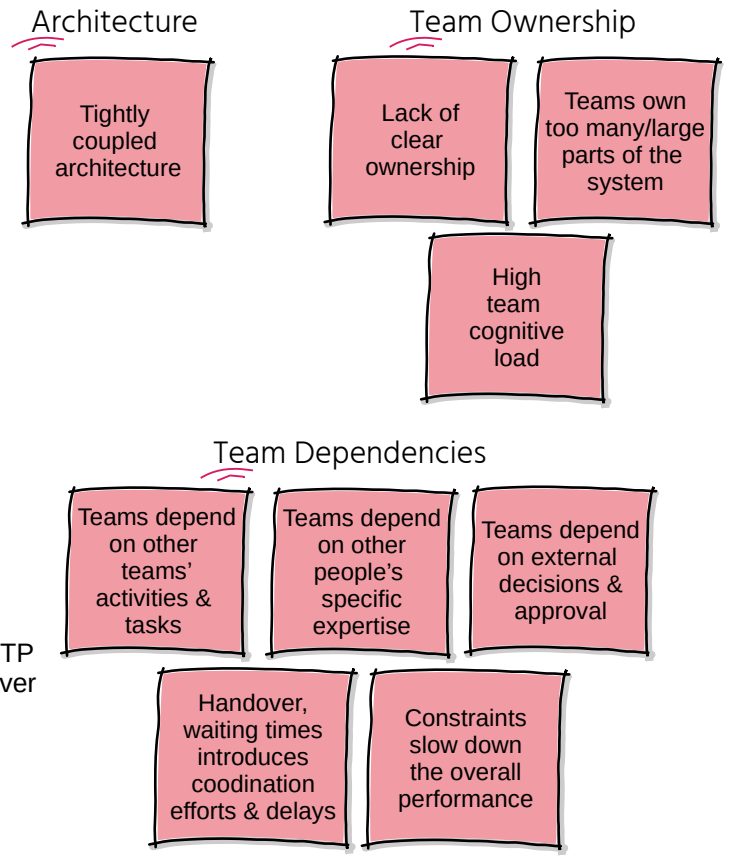
Teams depend on external decisions & approval

Evolution

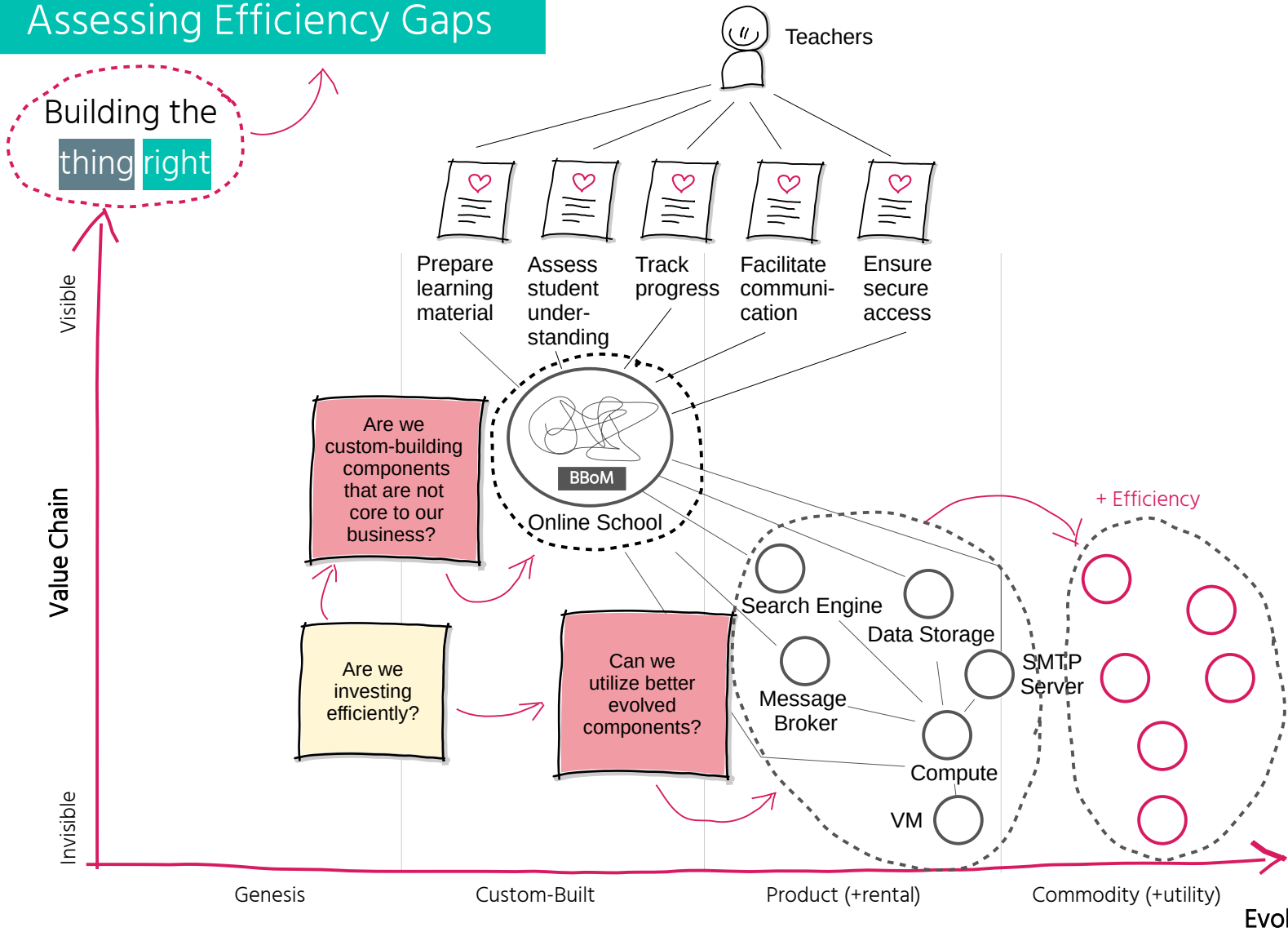
# Assessing Current Flow of Change



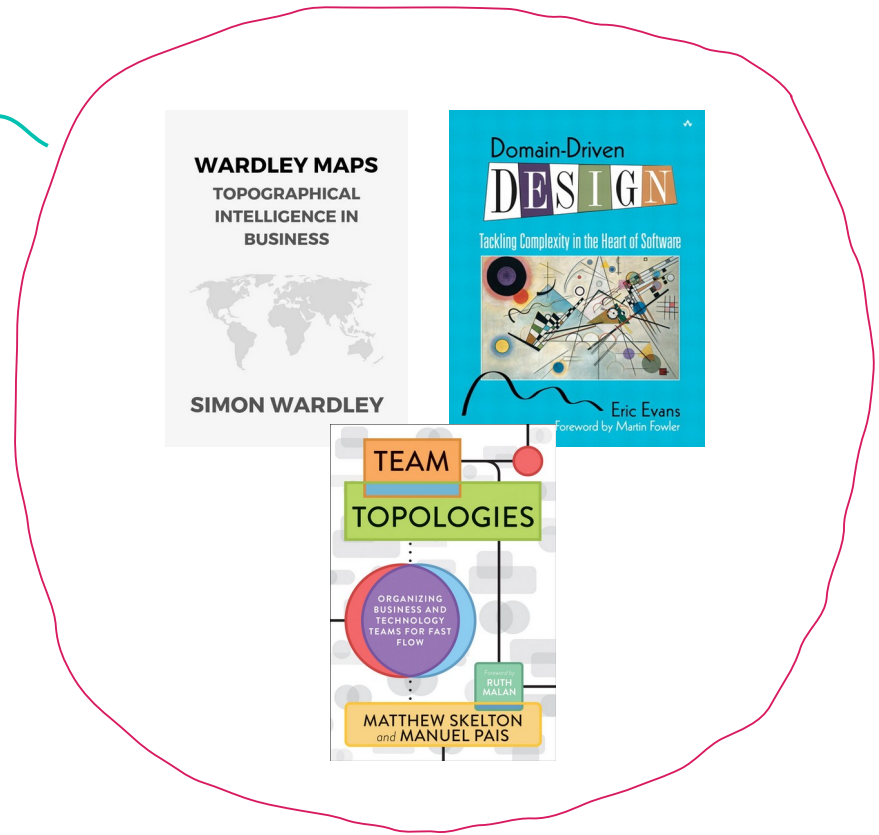
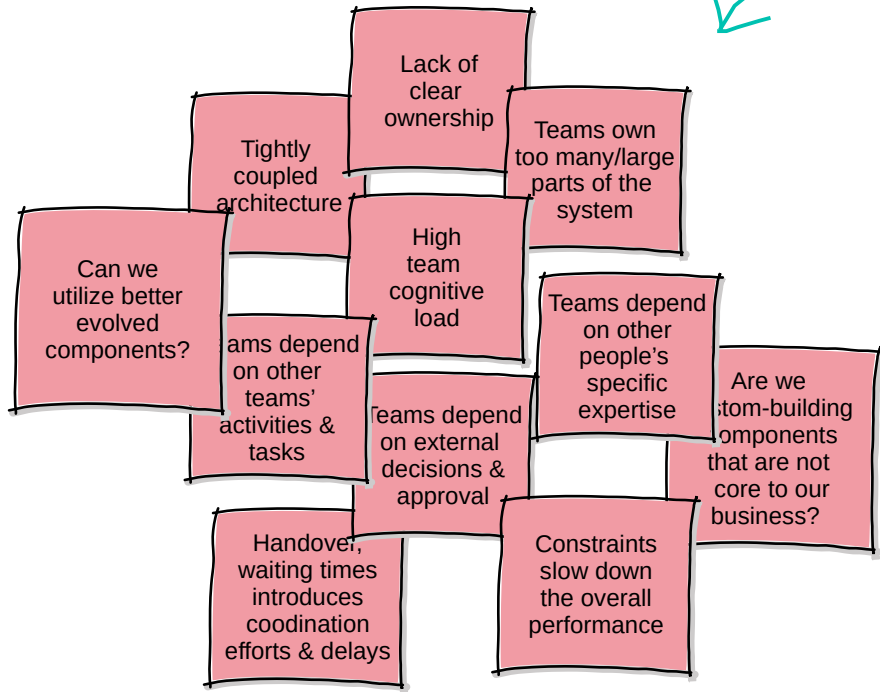
## What is preventing flow?



# Assessing Efficiency Gaps



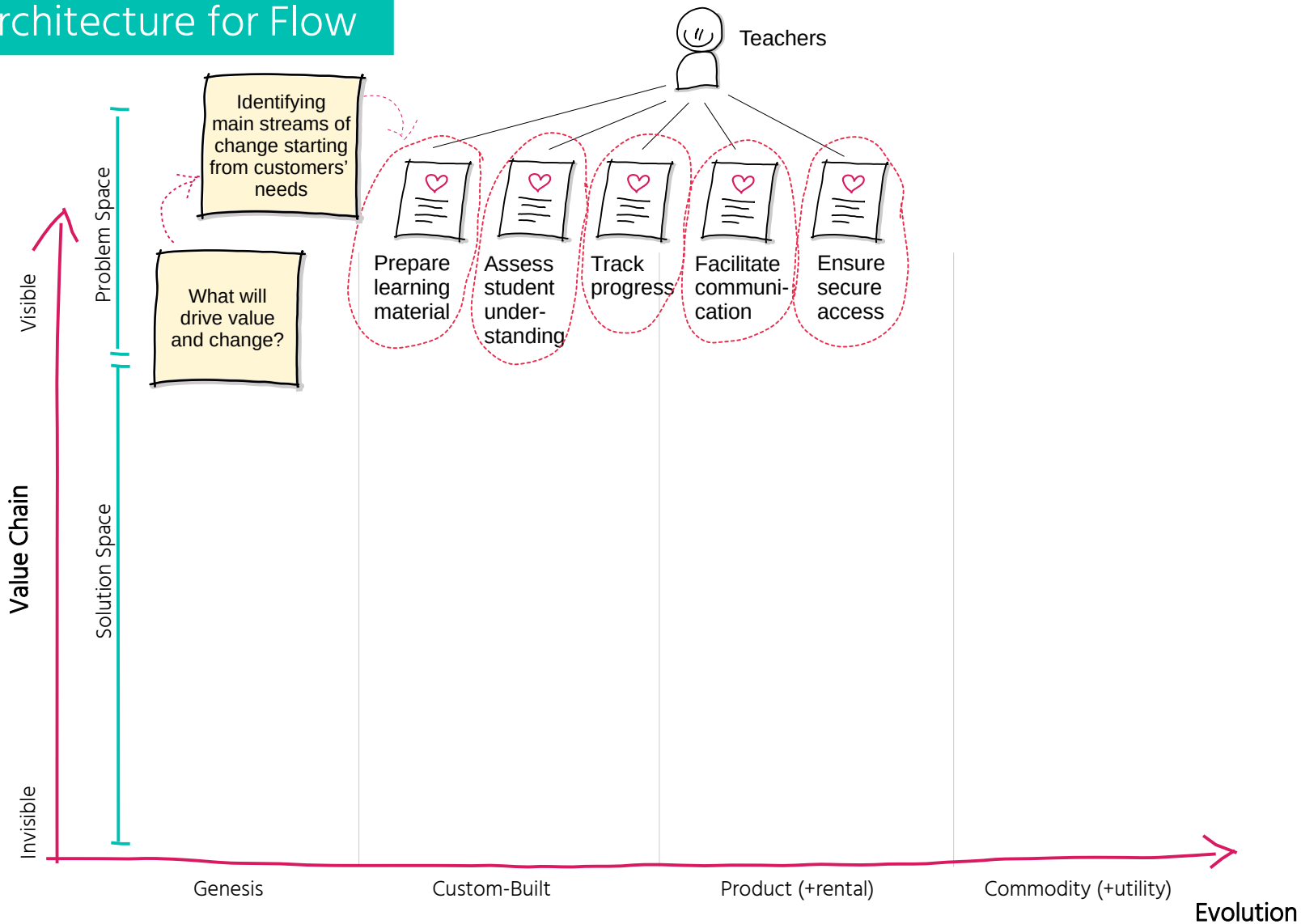
unlock



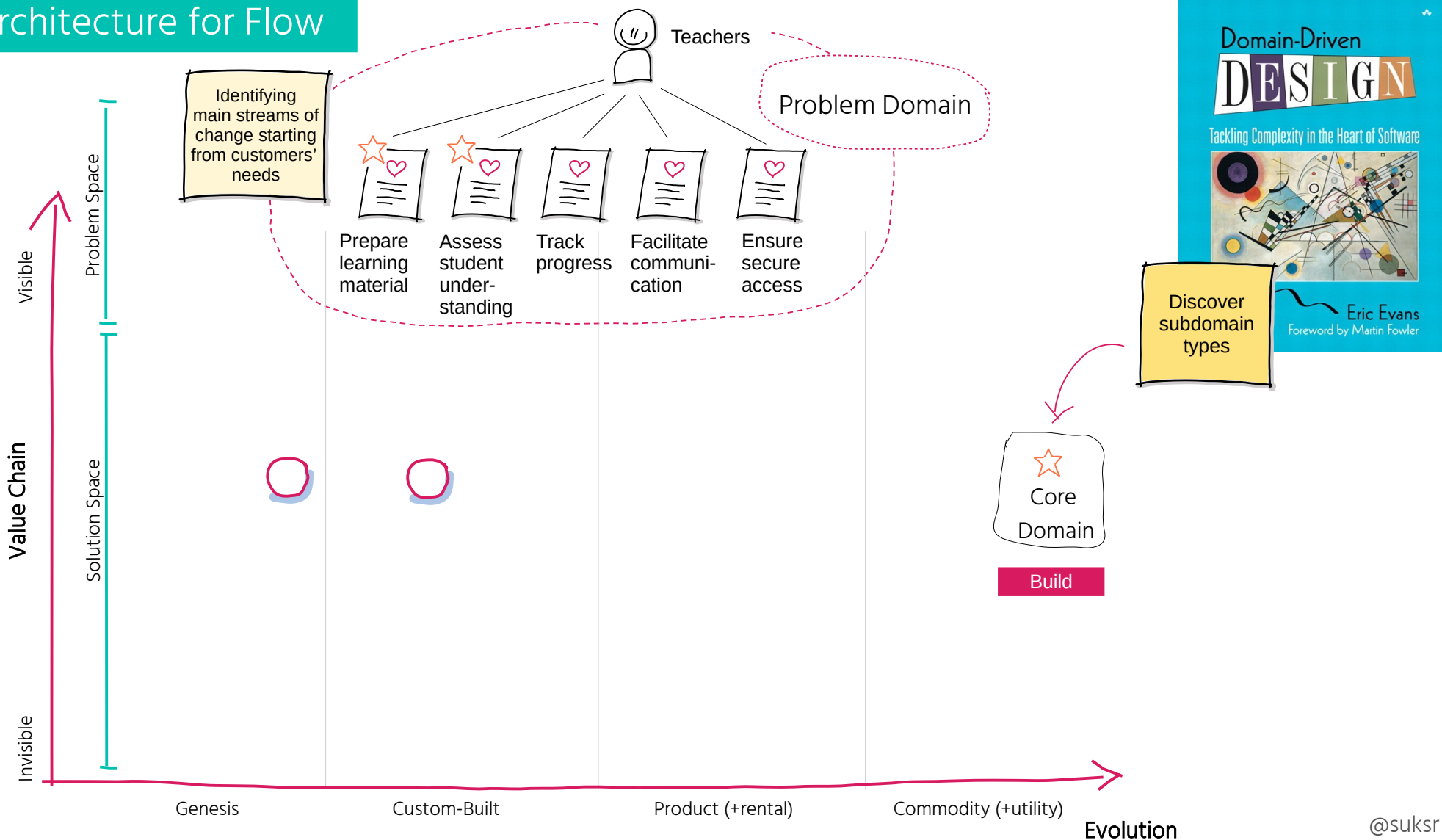
Architecture for Flow



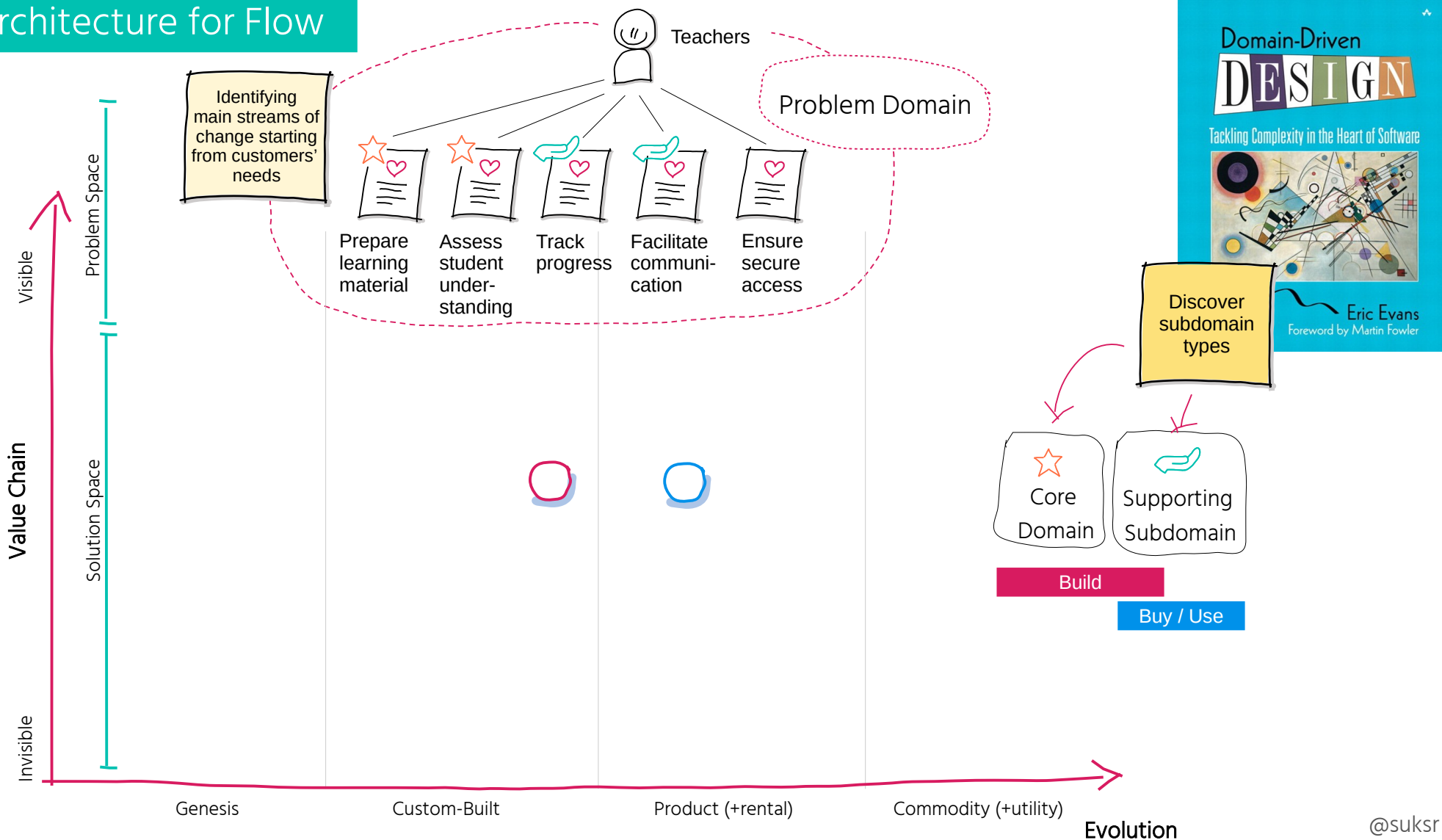
# Architecture for Flow



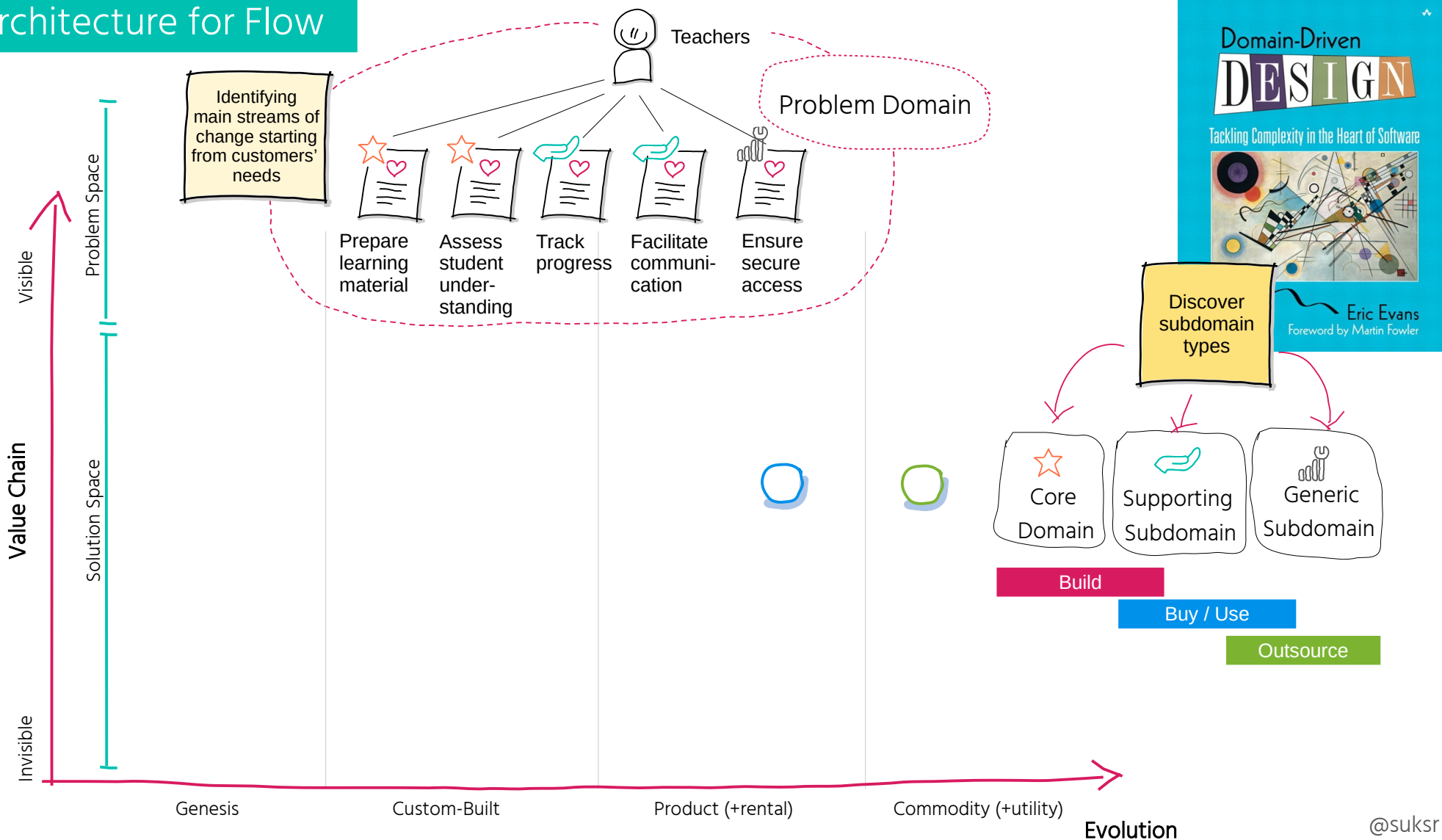
# Architecture for Flow



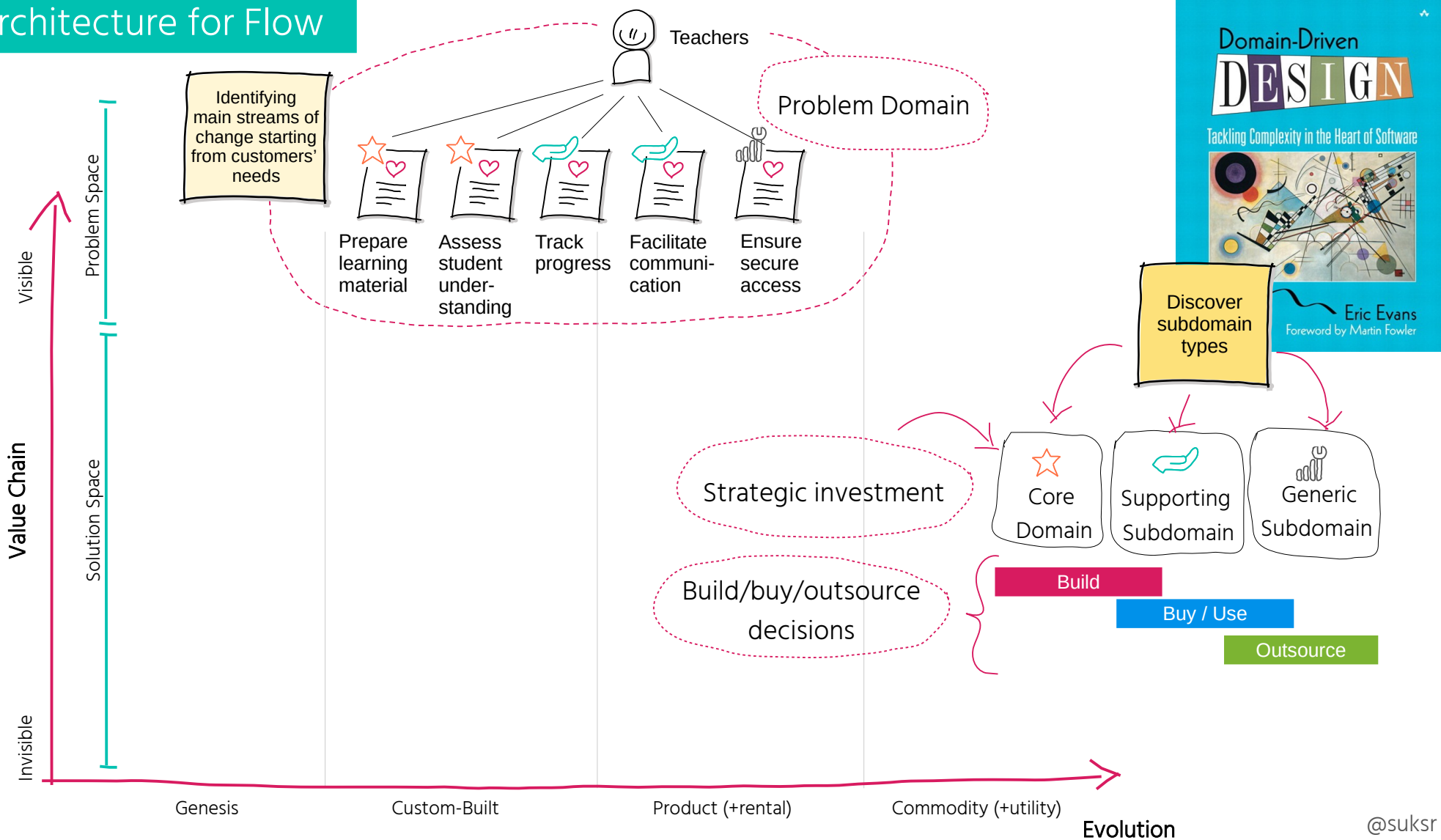
# Architecture for Flow



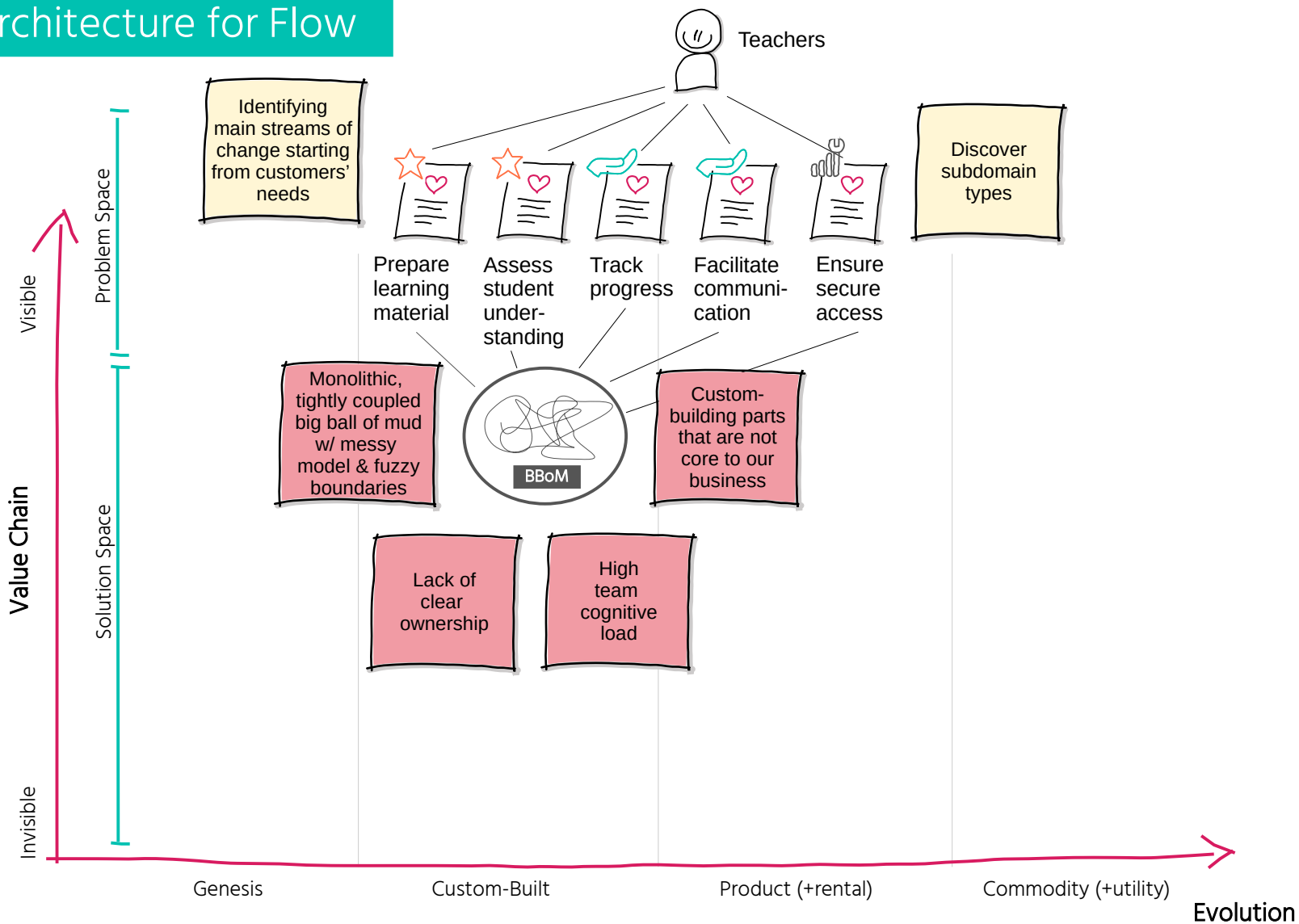
# Architecture for Flow



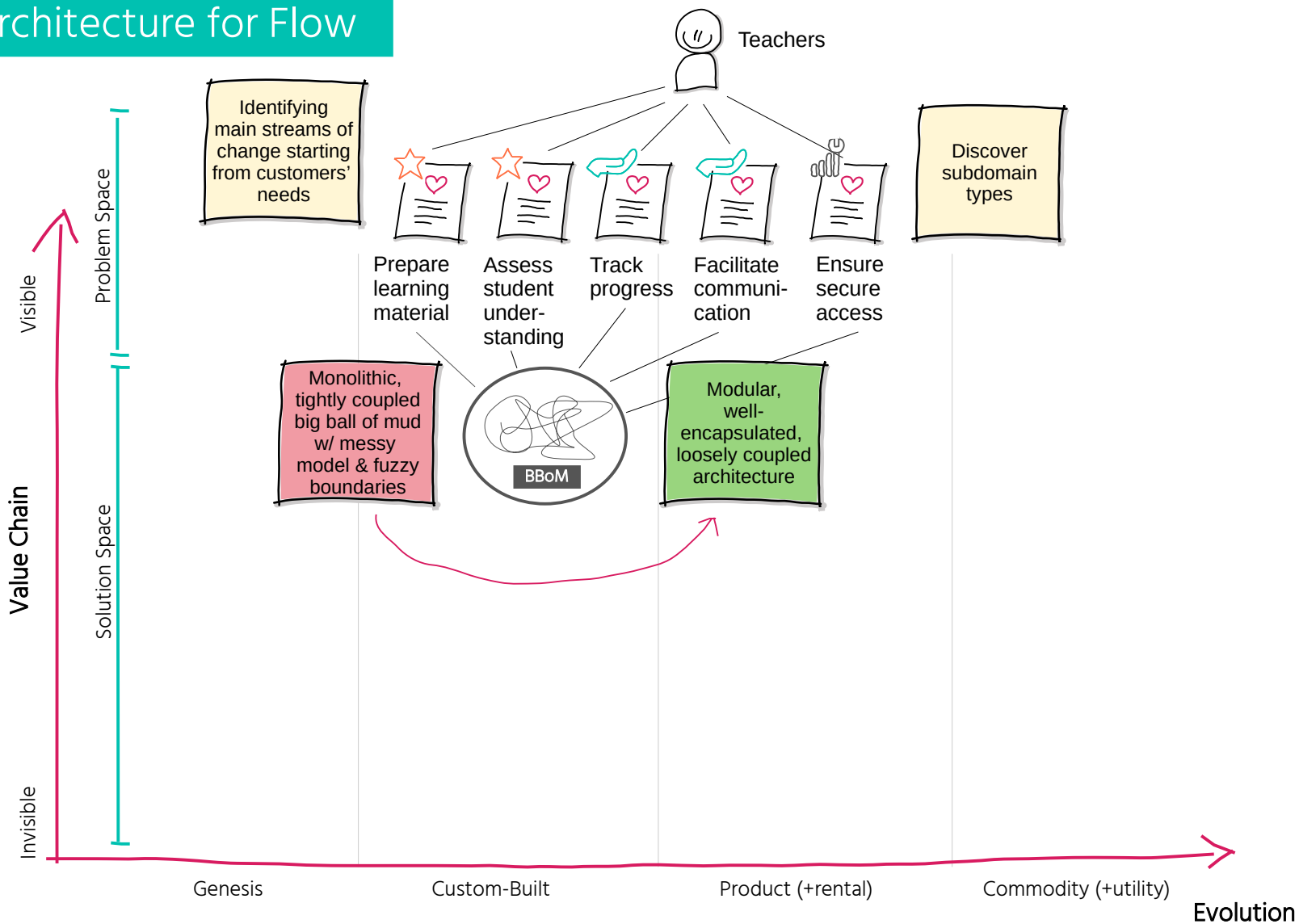
# Architecture for Flow



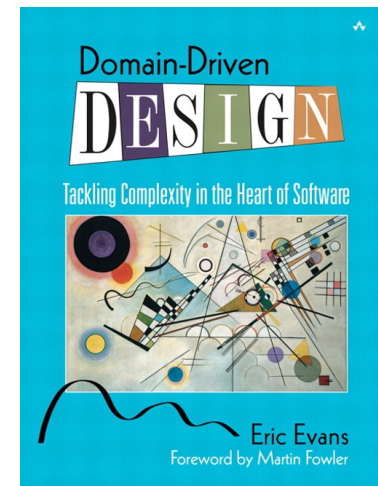
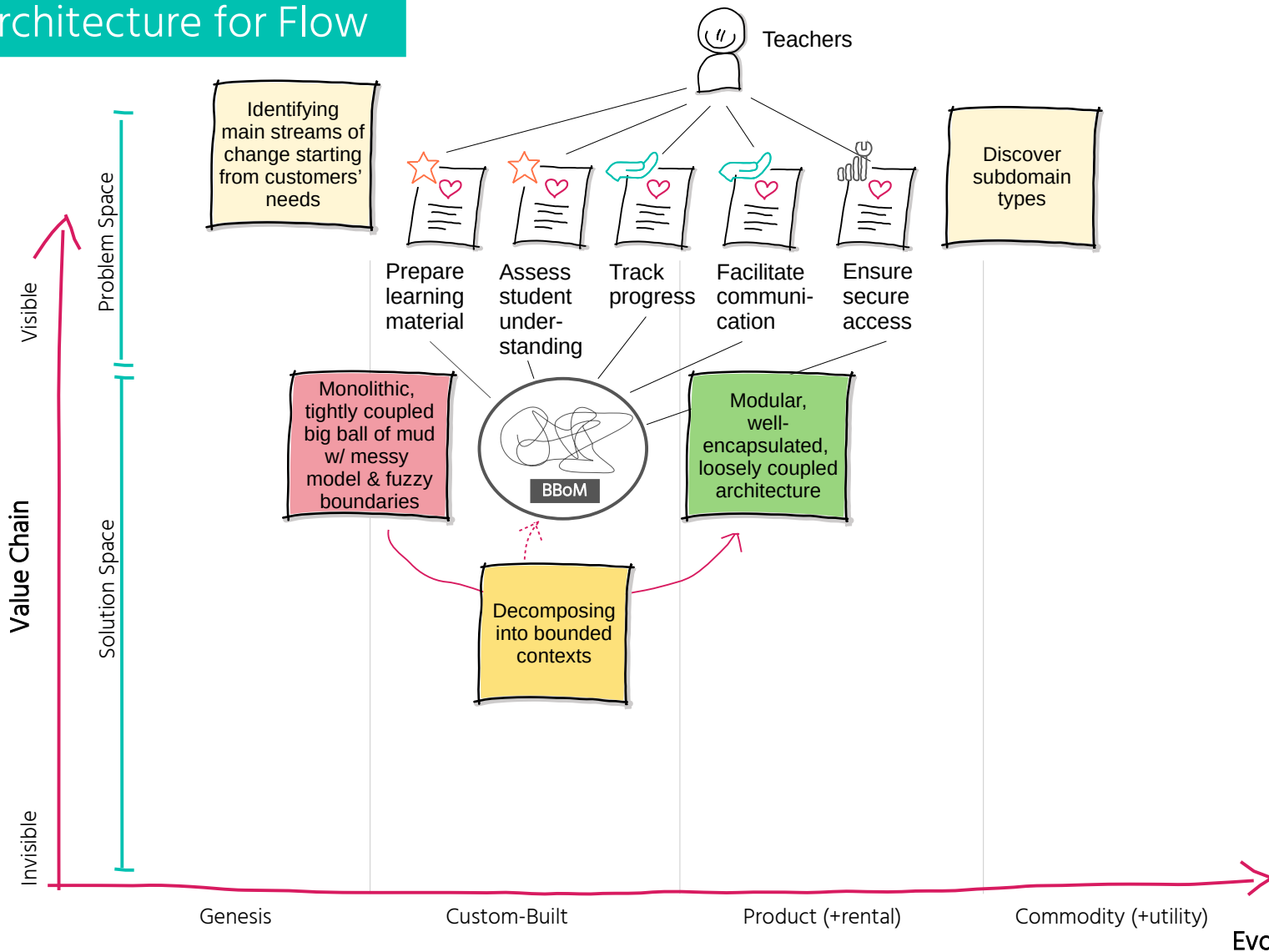
# Architecture for Flow



# Architecture for Flow

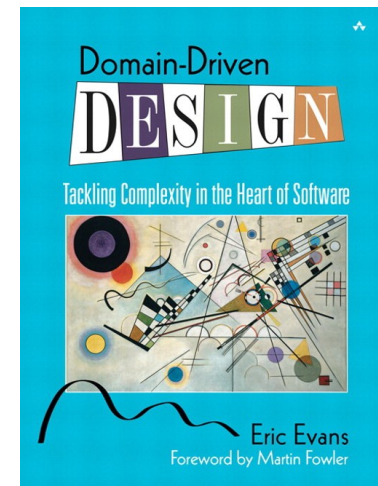
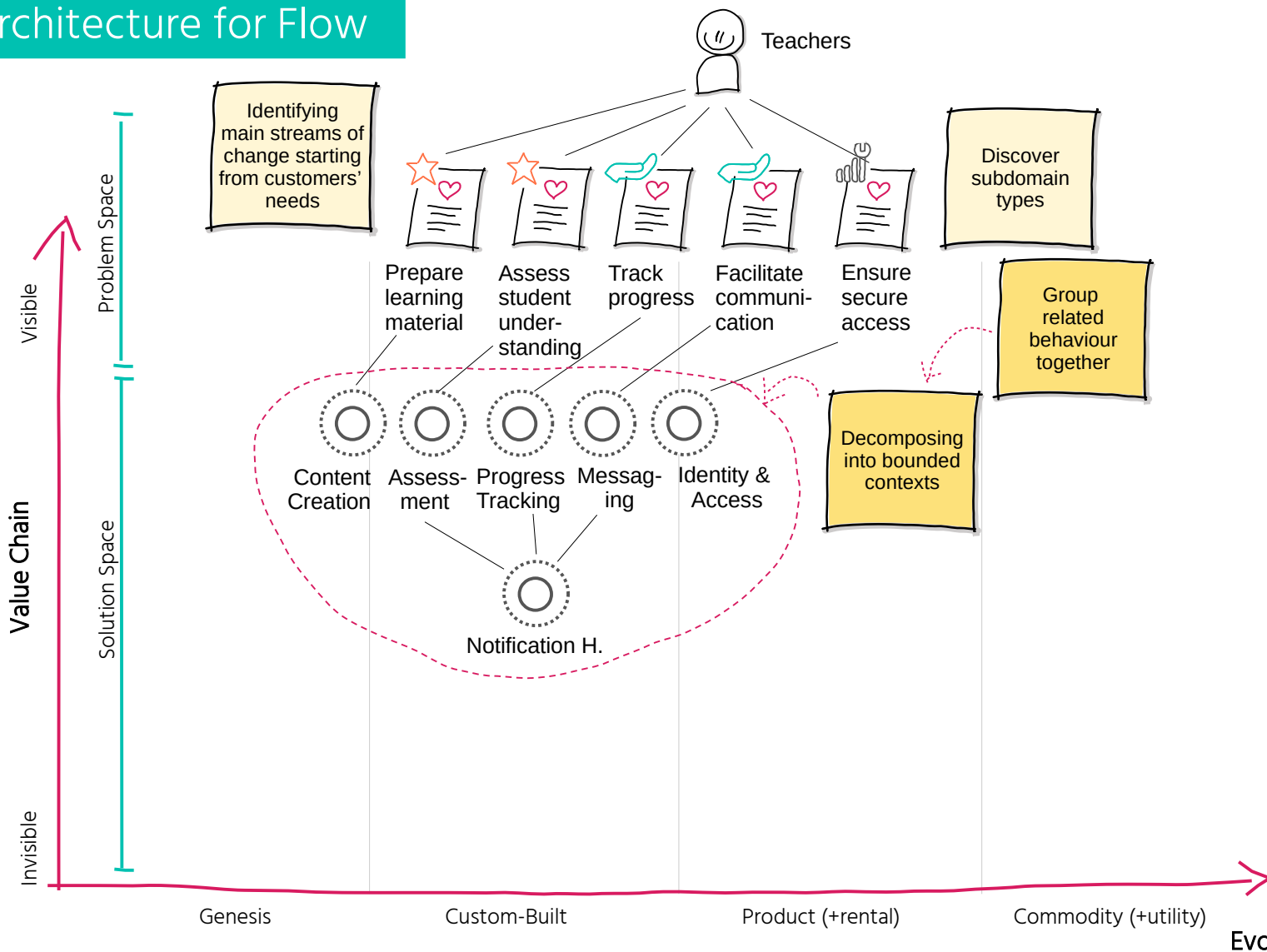


# Architecture for Flow

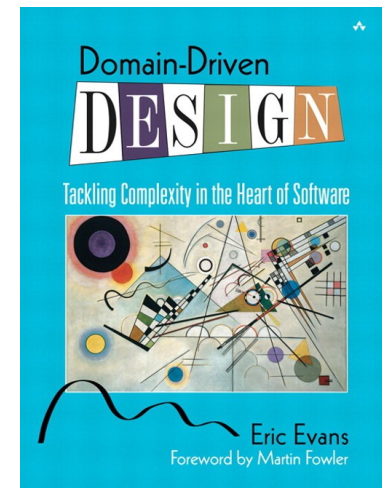
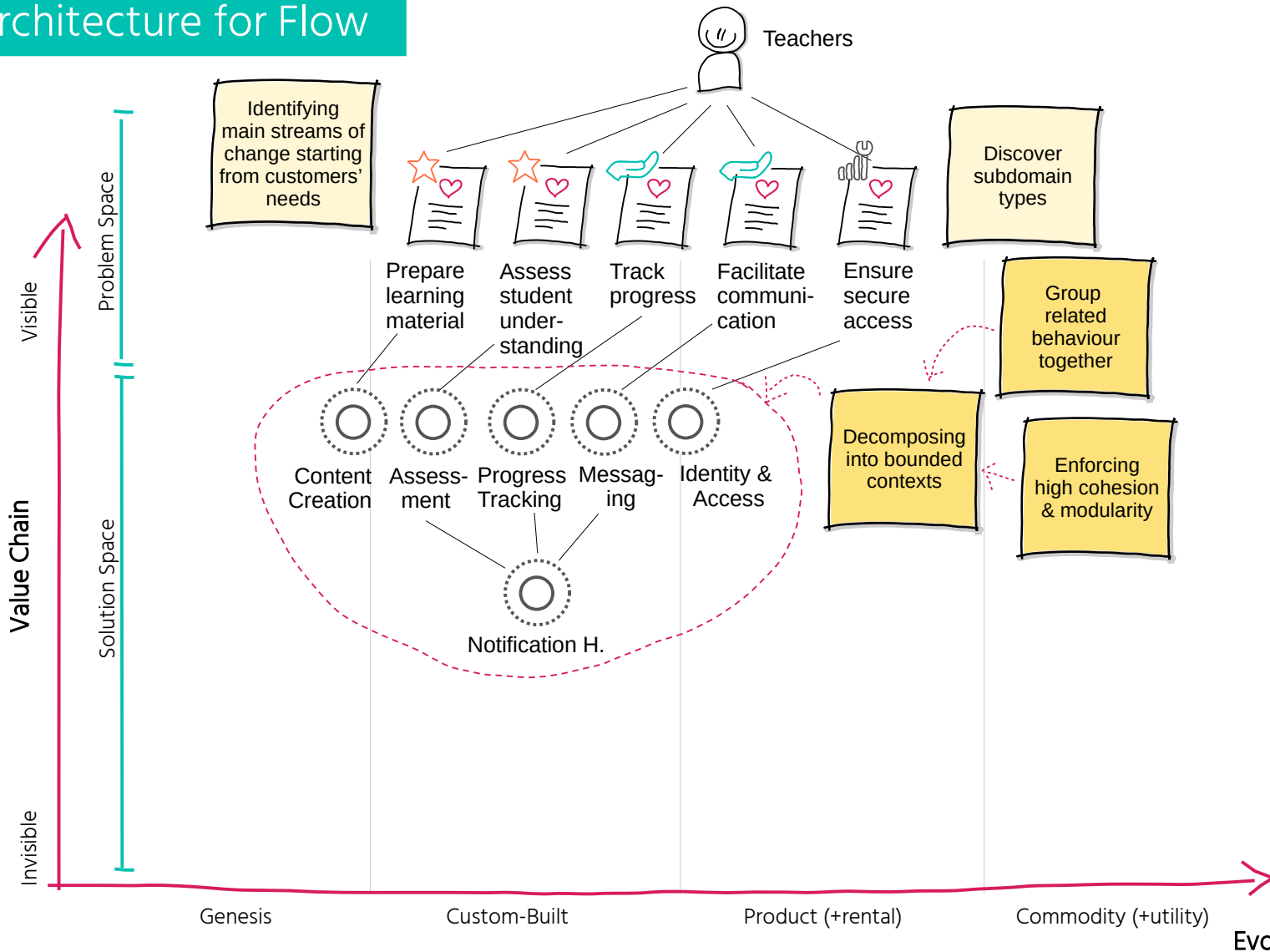




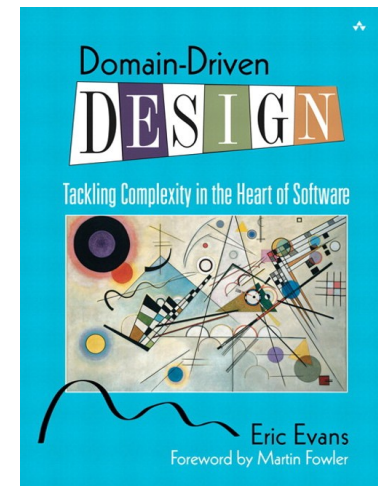
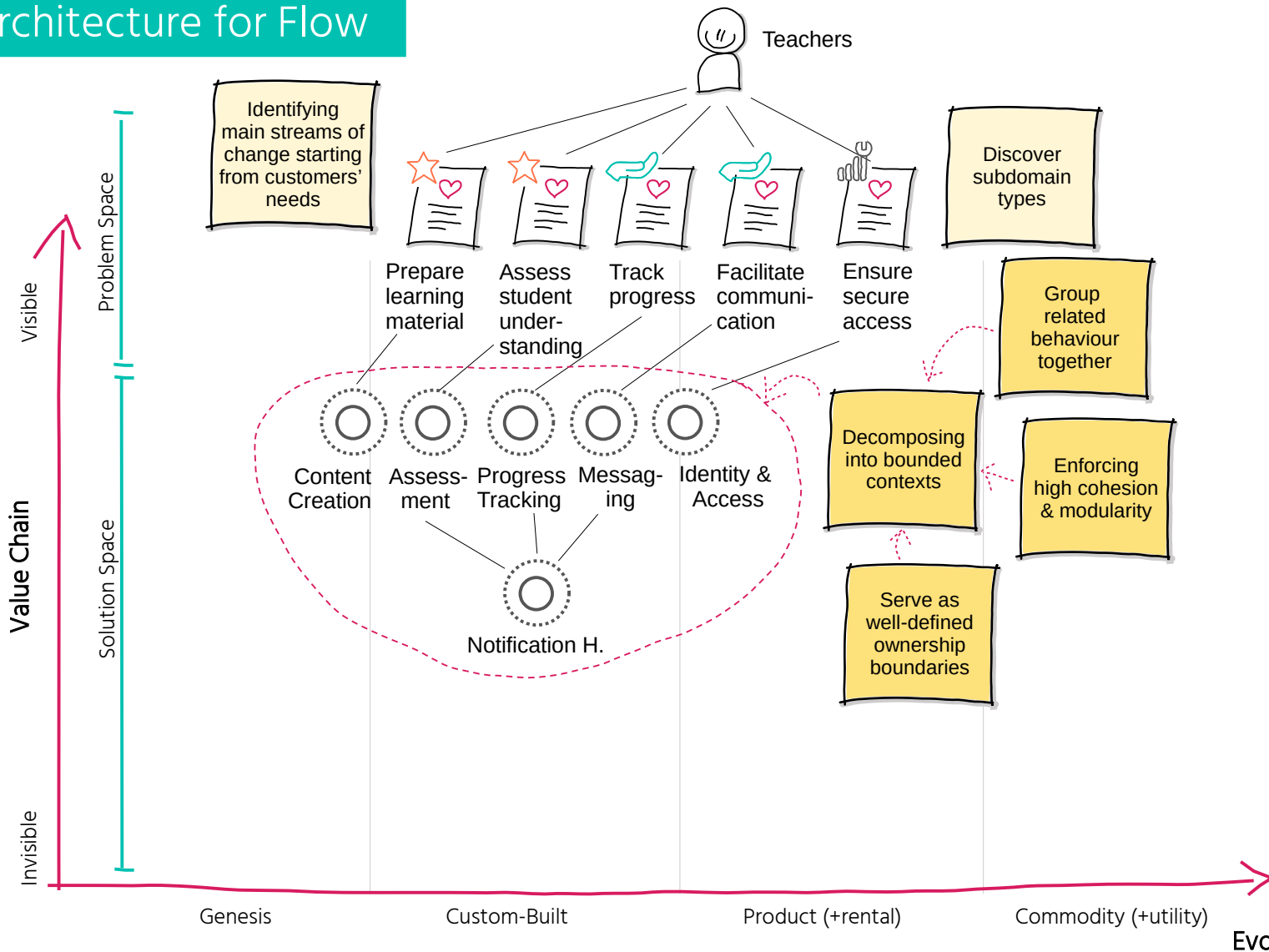
# Architecture for Flow



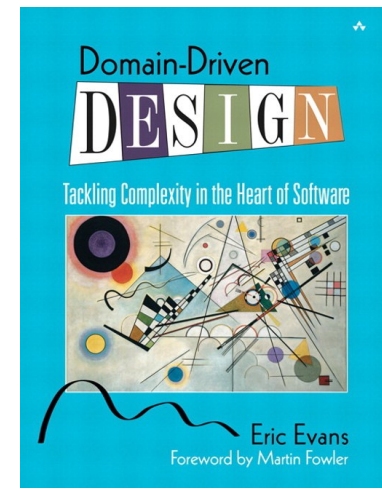
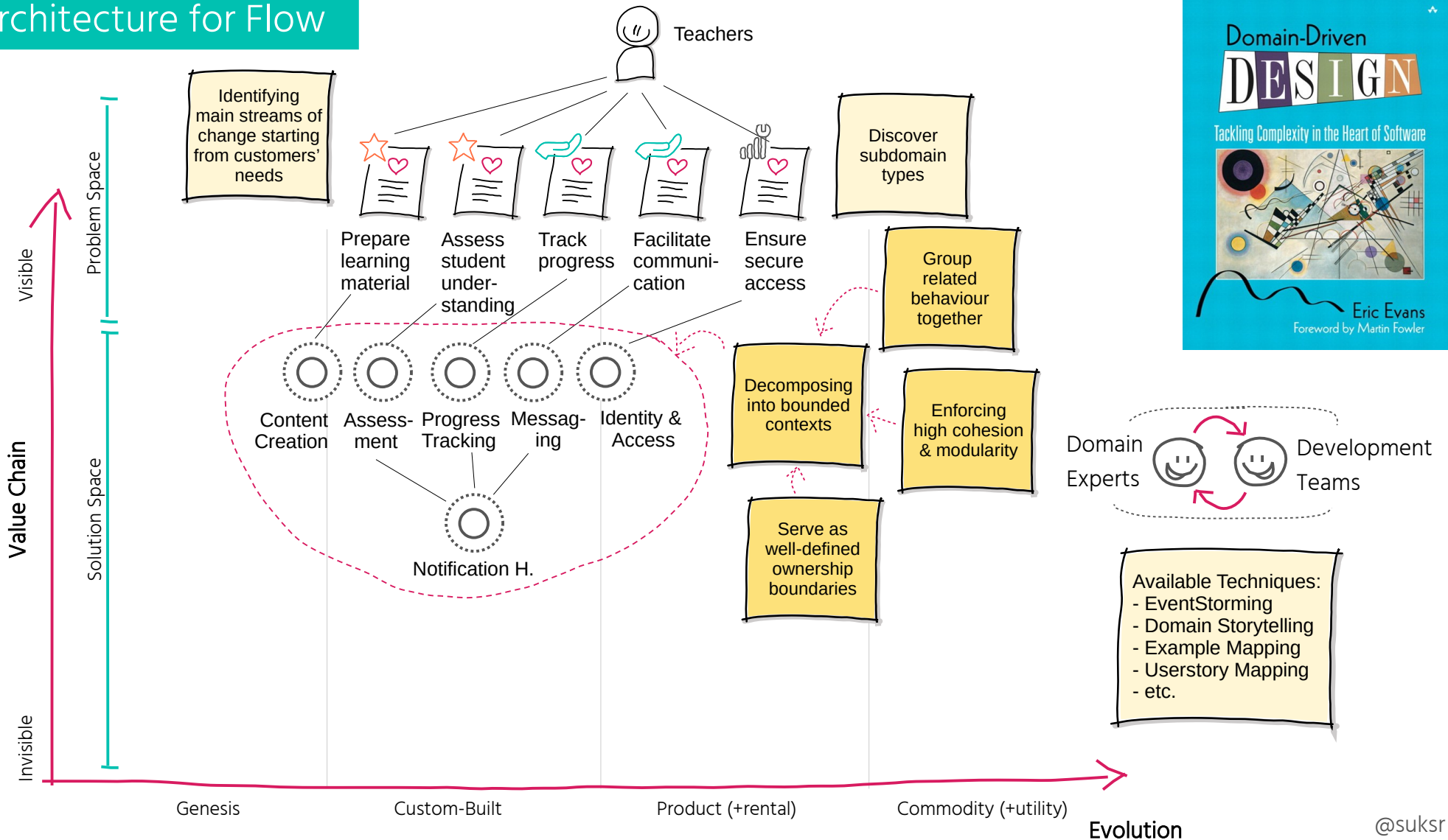
# Architecture for Flow



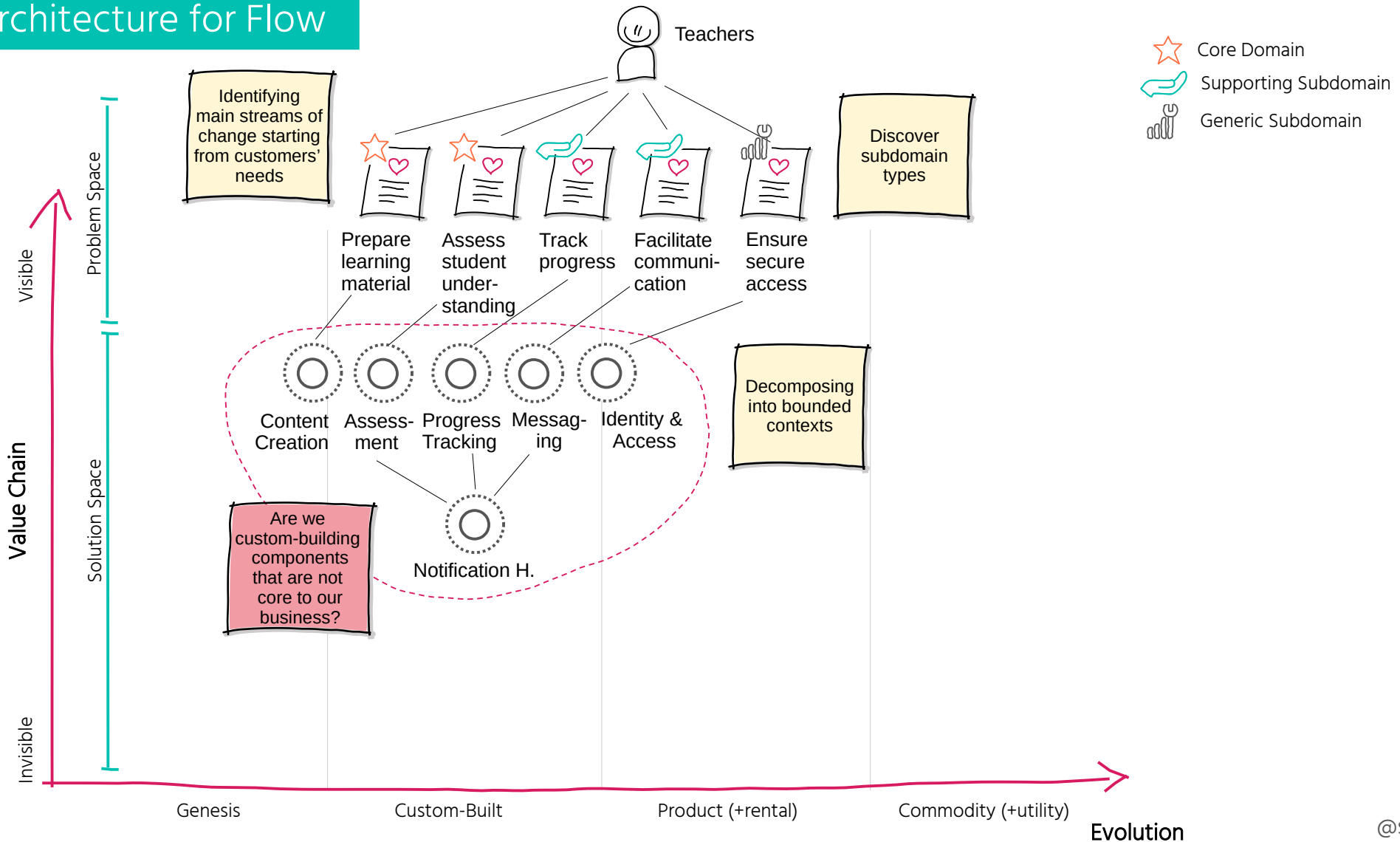
# Architecture for Flow



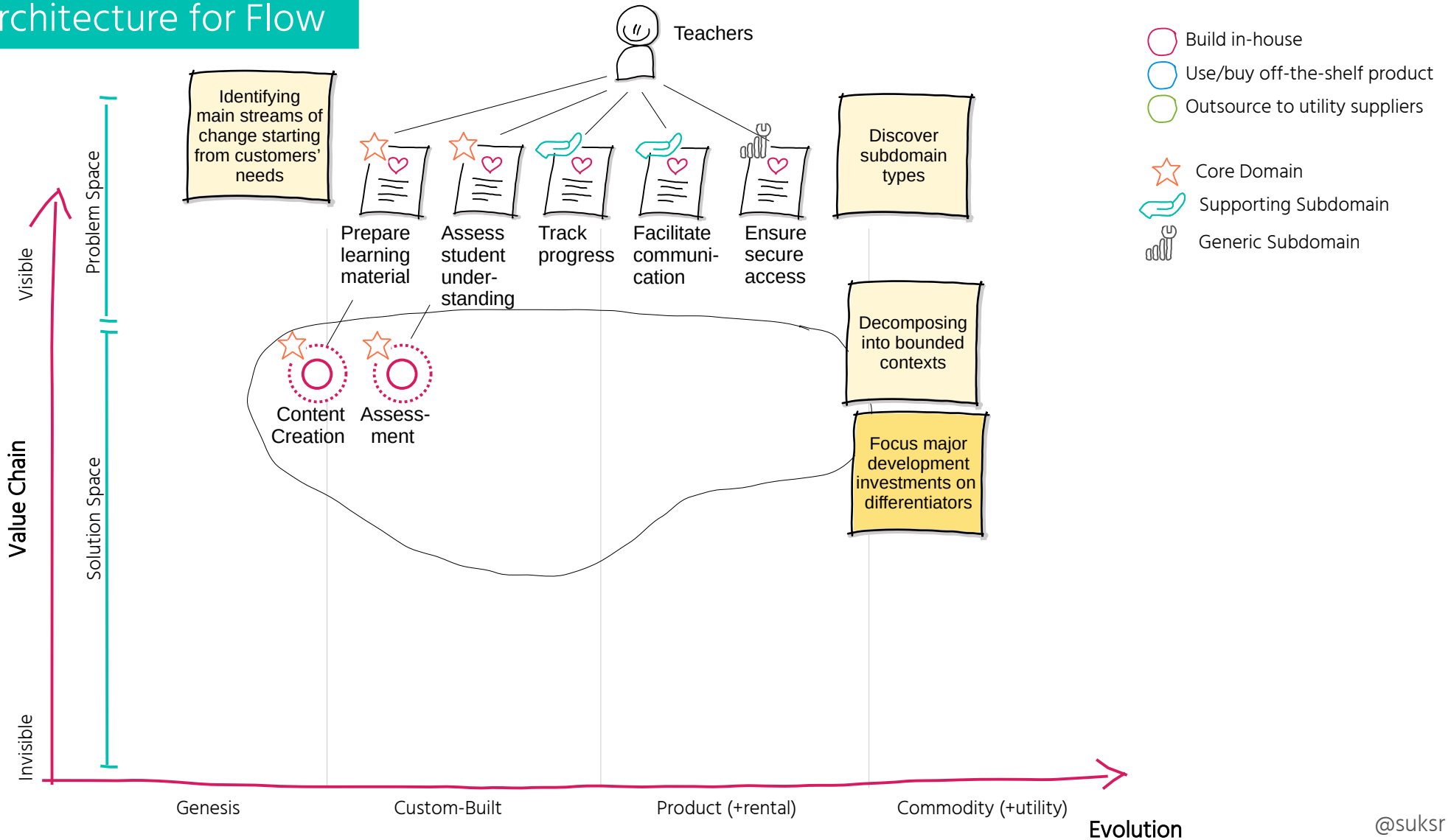
# Architecture for Flow



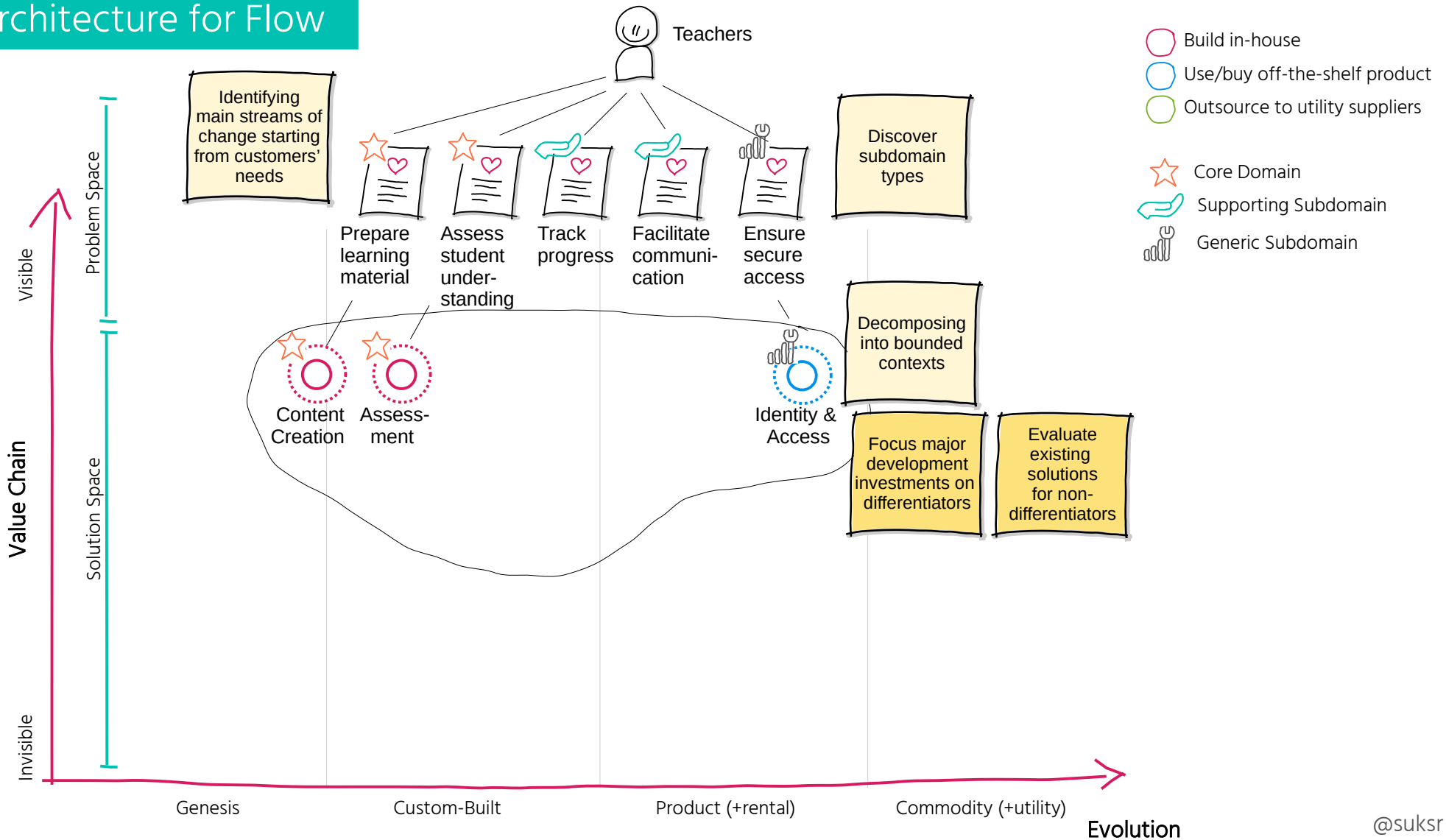
# Architecture for Flow



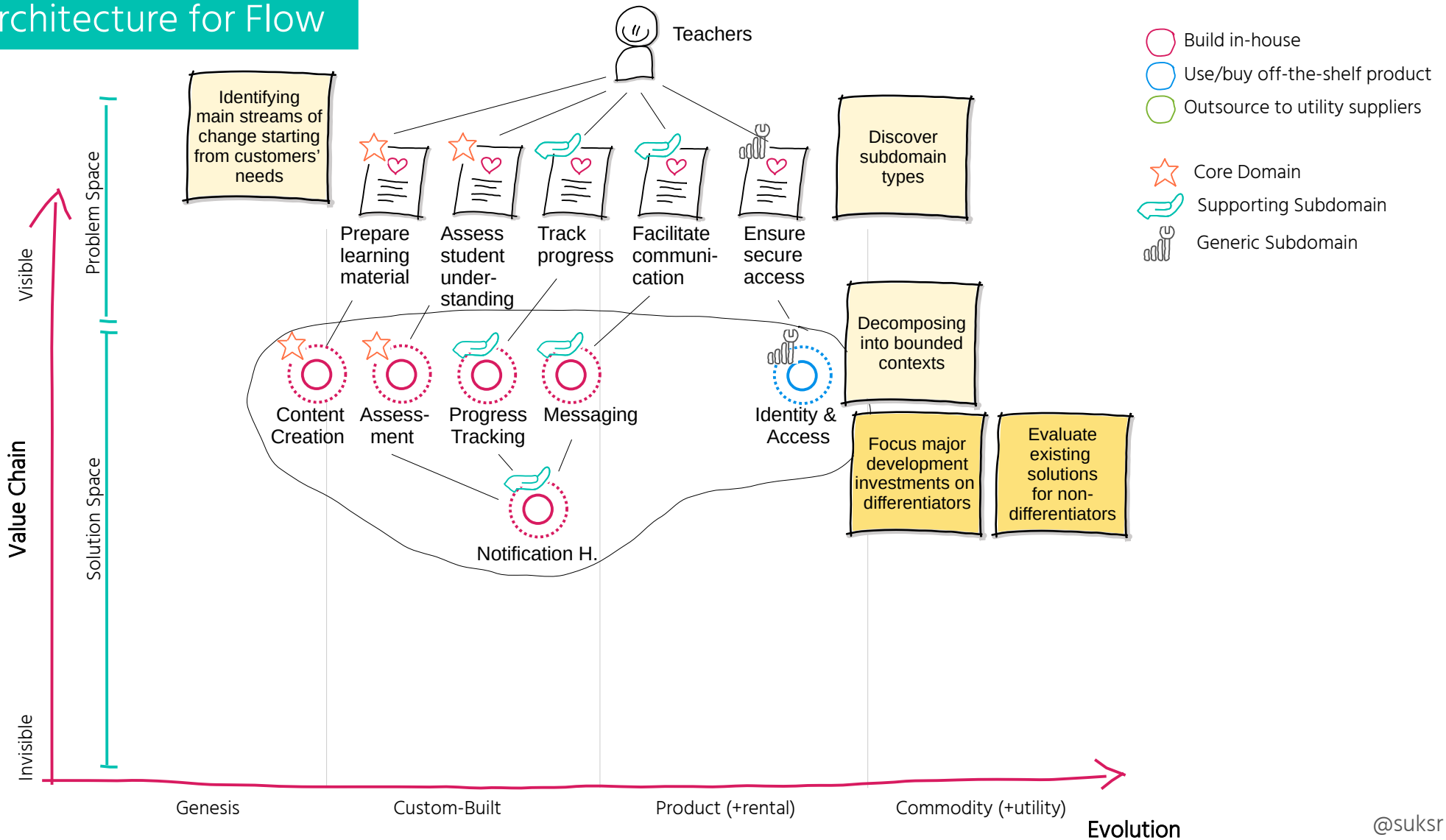
# Architecture for Flow



# Architecture for Flow

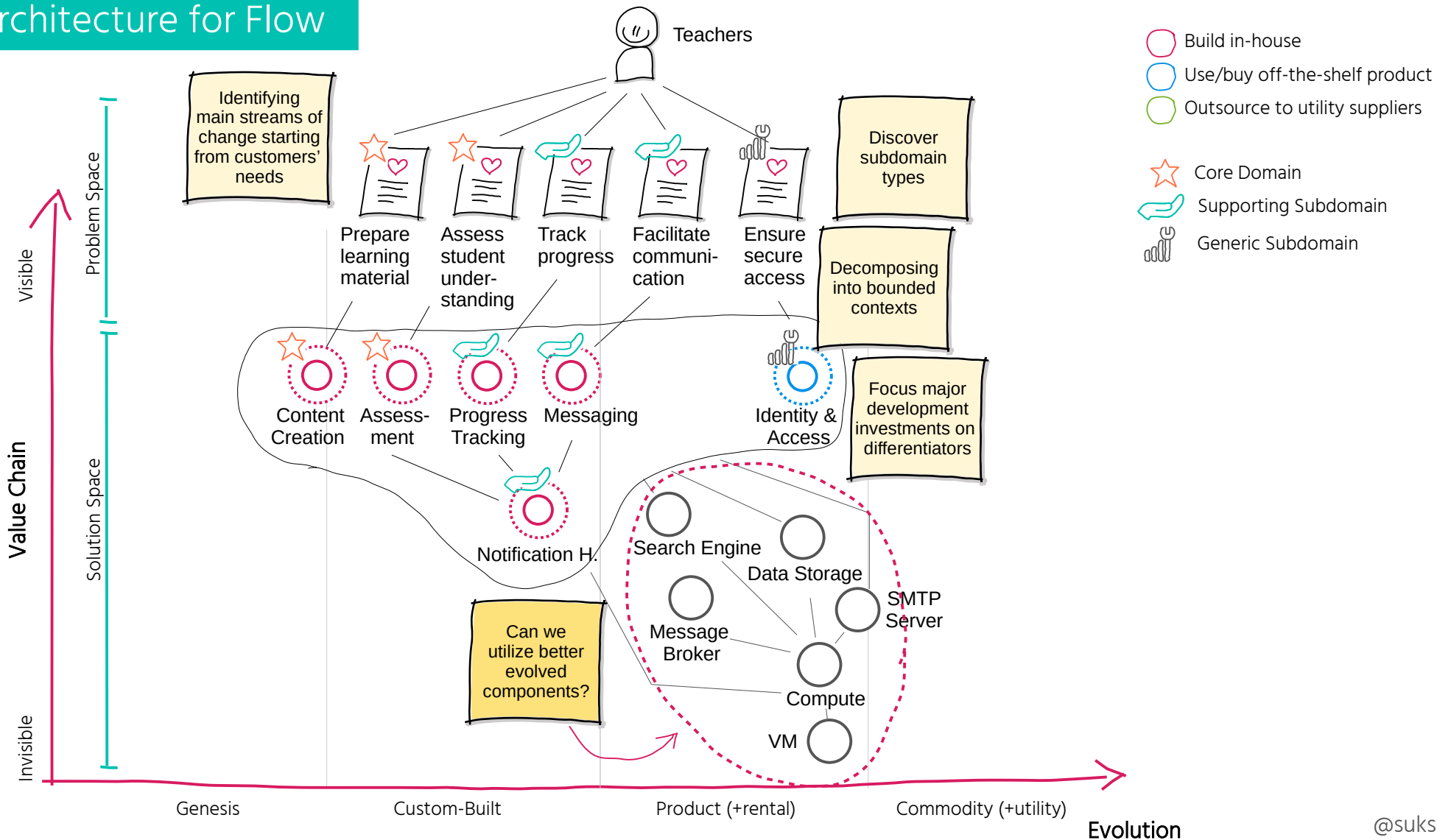


# Architecture for Flow

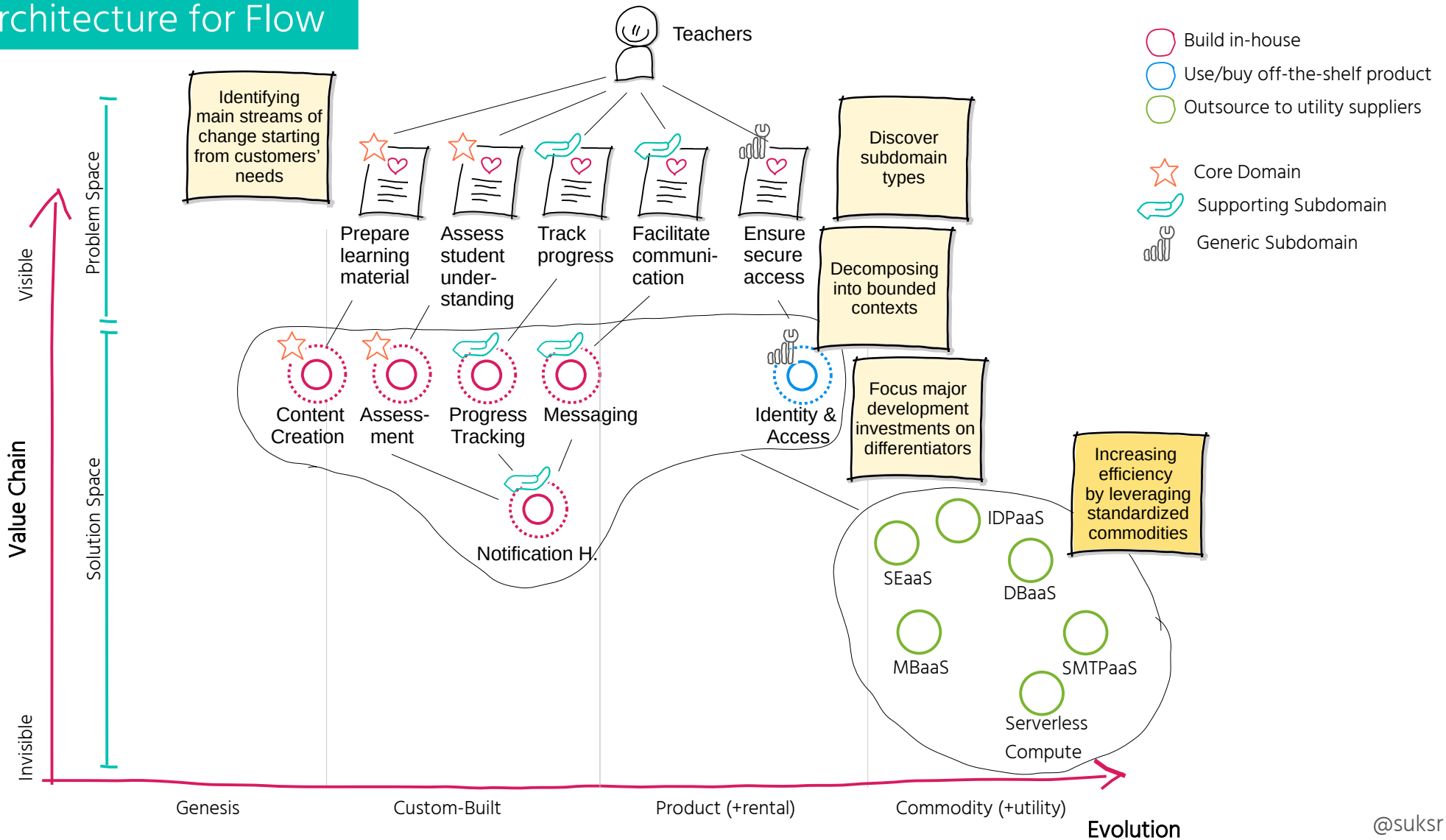




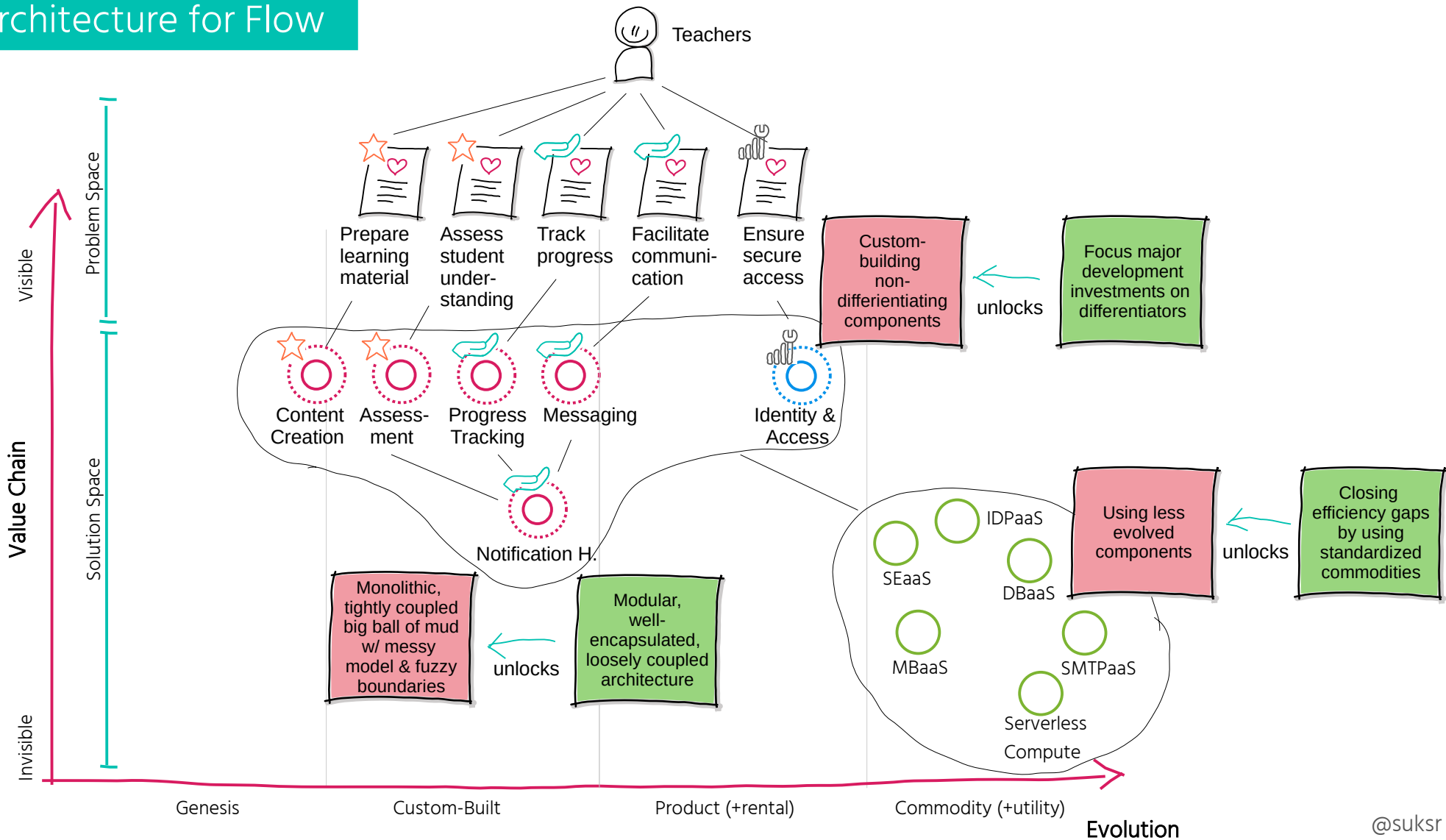
# Architecture for Flow



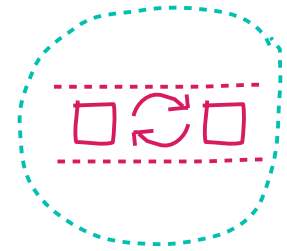
# Architecture for Flow



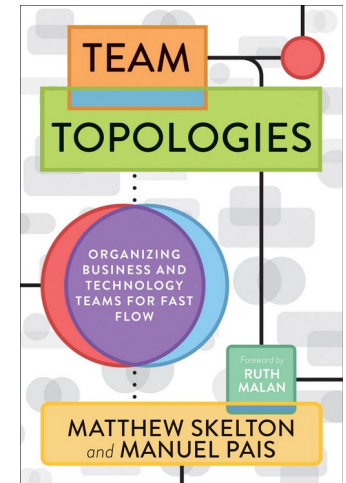
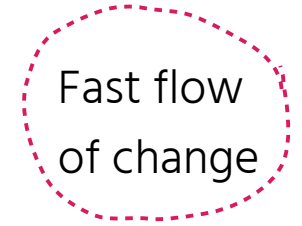
# Architecture for Flow



# Four Team Types of Team Topologies



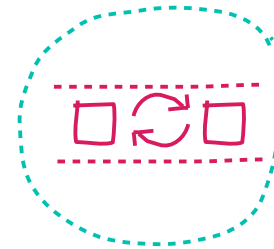
Stream-aligned team



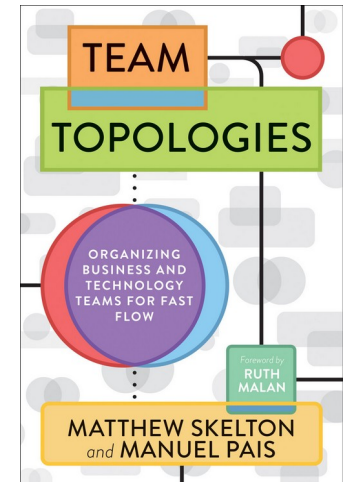
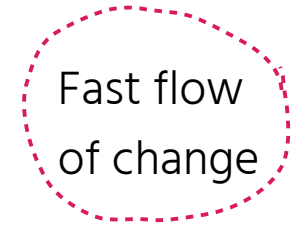
# Four Team Types of Team Topologies



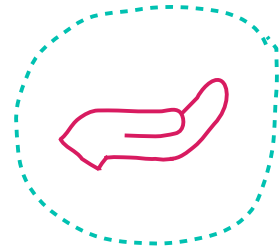
Platform team



Stream-aligned team



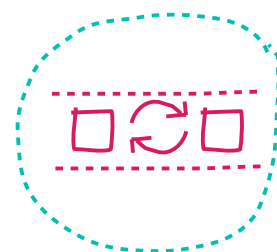
# Four Team Types of Team Topologies



Enabling  
team



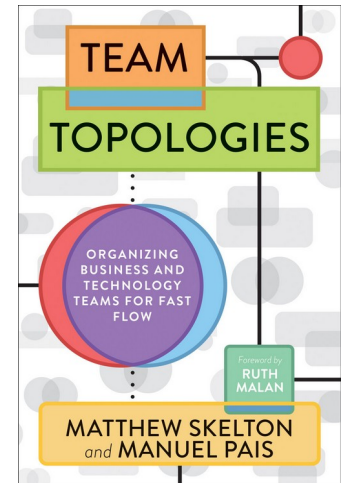
Platform  
team



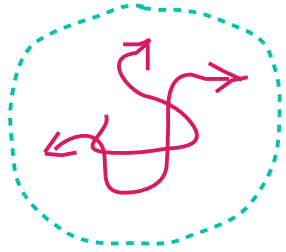
Stream-aligned  
team



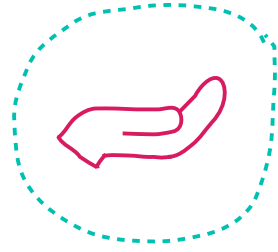
Fast flow  
of change



# Four Team Types of Team Topologies



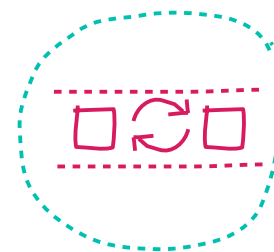
Complicated  
subsystem team



Enabling  
team



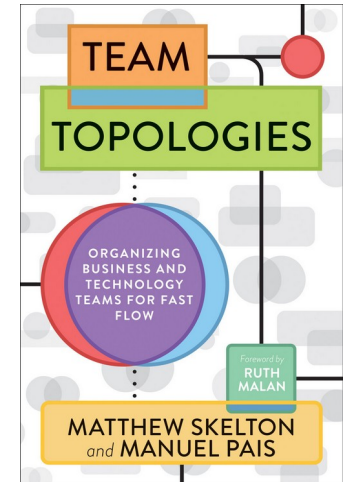
Platform  
team



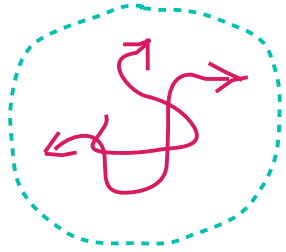
Stream-aligned  
team



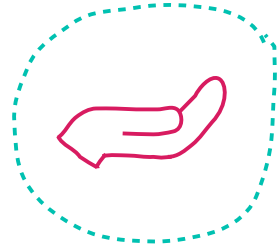
Fast flow  
of change



# Four Team Types of Team Topologies



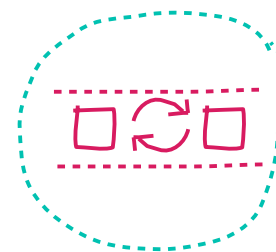
Complicated  
subsystem team



Enabling  
team



Platform  
team



Stream-aligned  
team

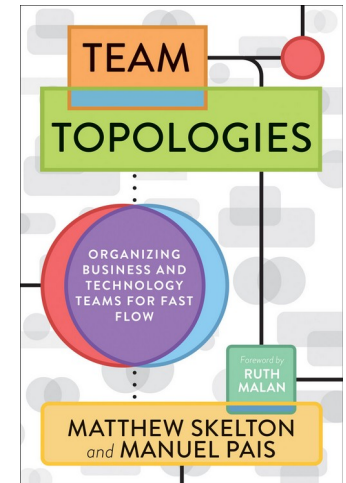


Fast flow  
of change



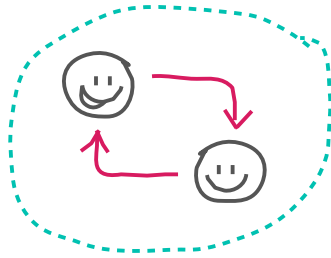
Increasing autonomy

Reducing cognitive load





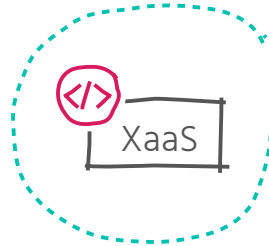
# Three Interaction Modes



Collaboration



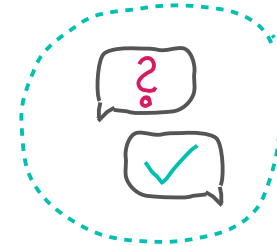
Rapid discovery



X-as-a-Service



Predictable  
delivery

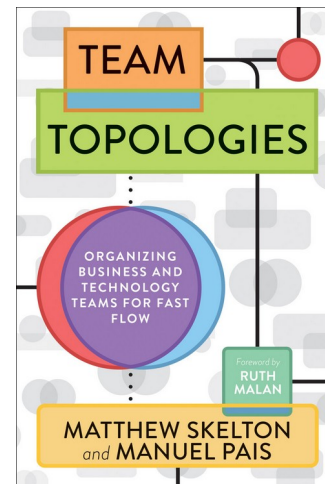
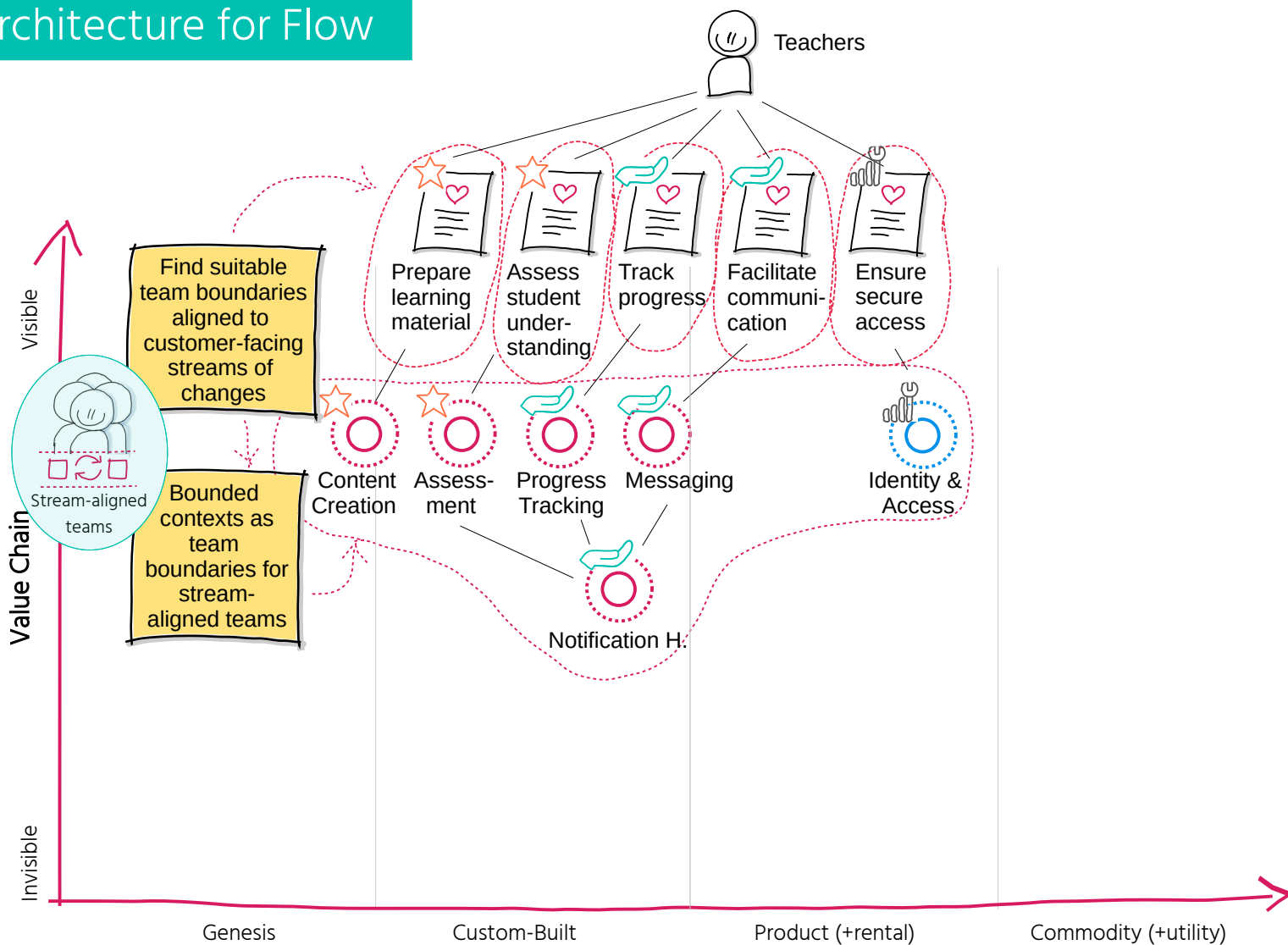


Facilitating

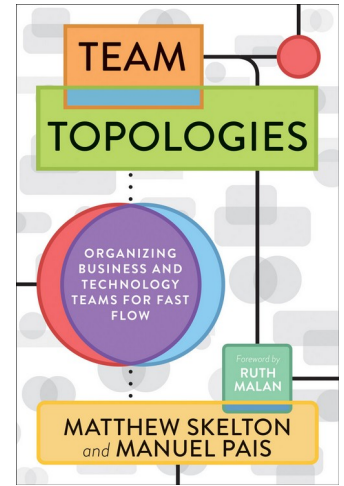
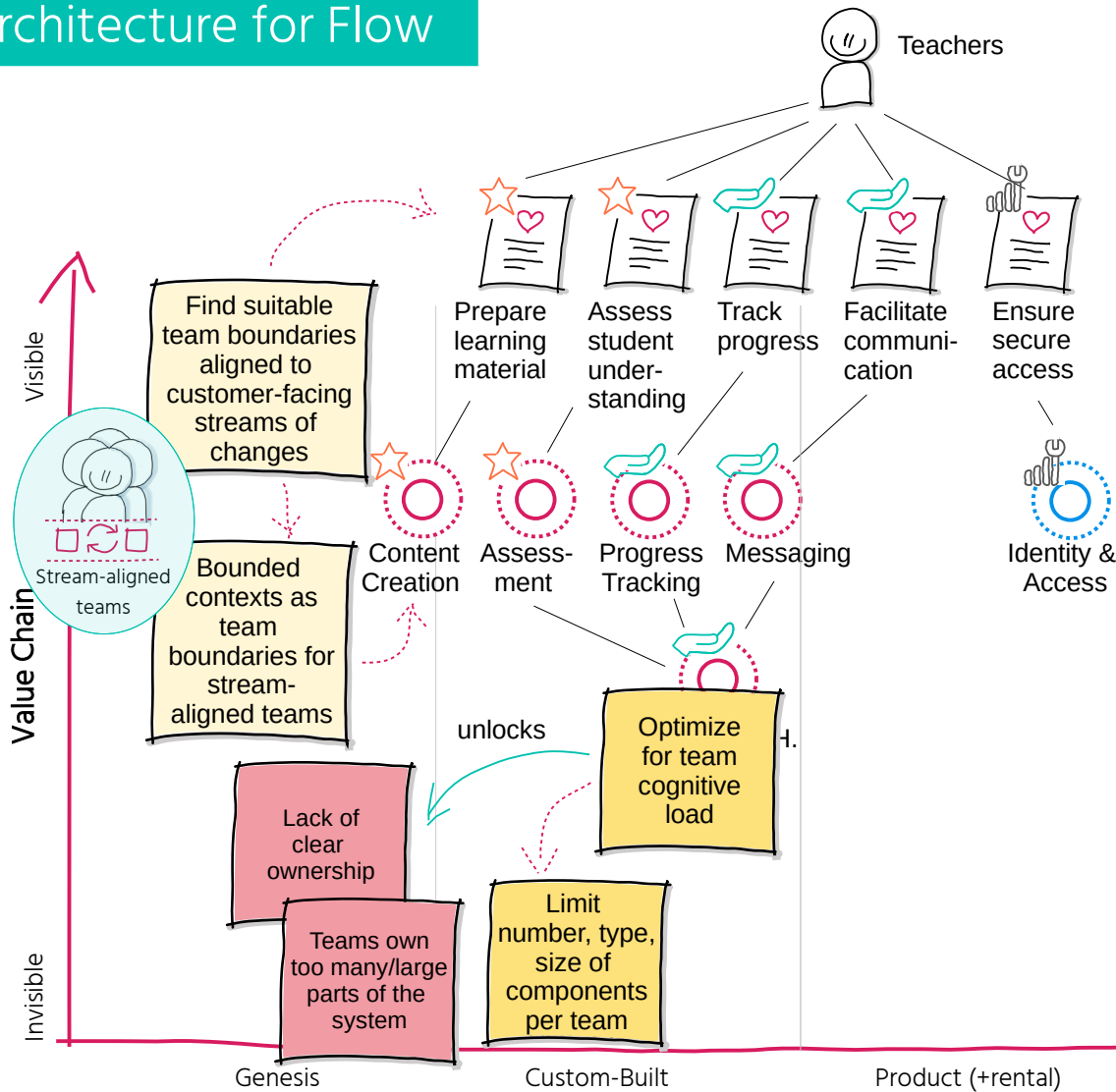


Active help

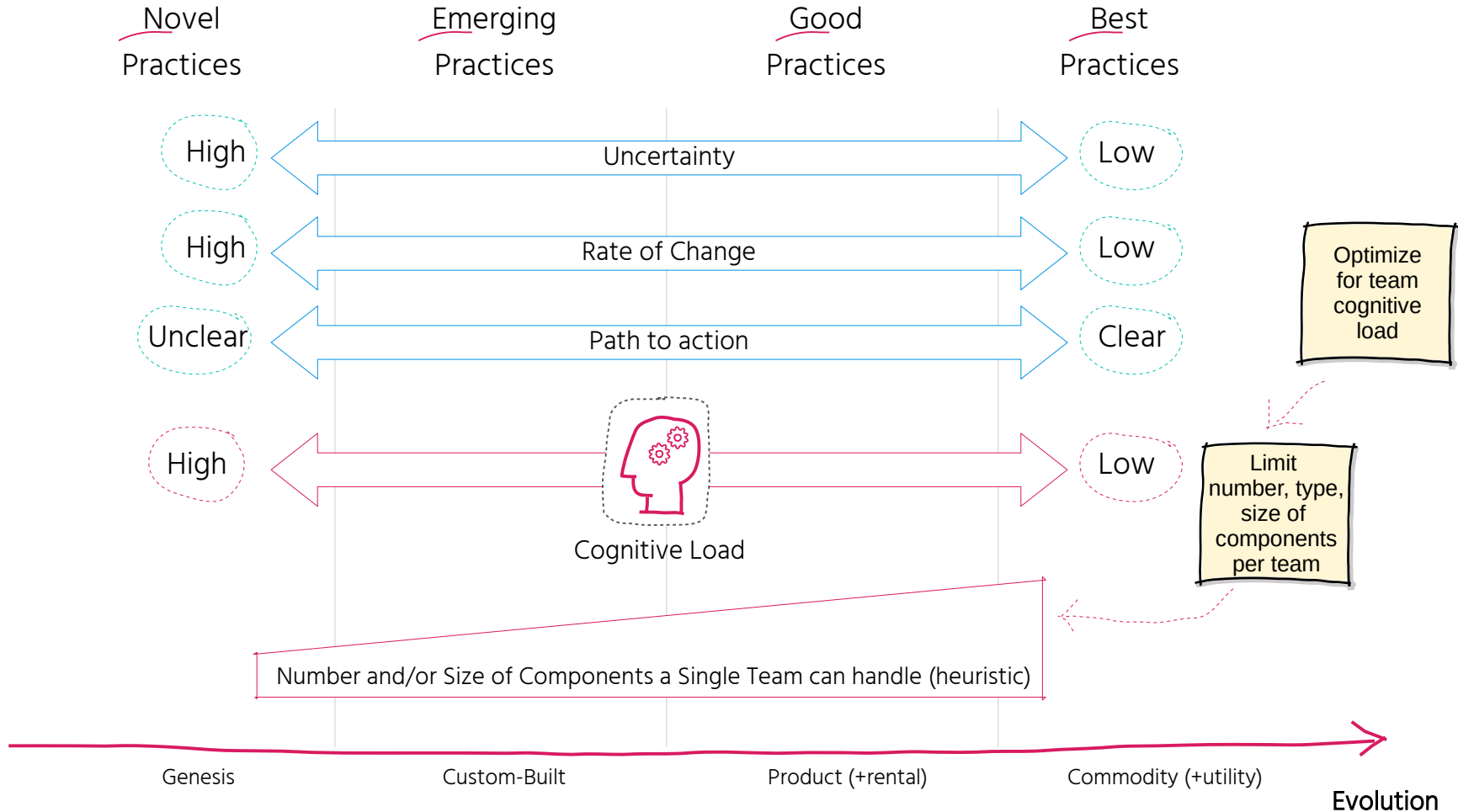
# Architecture for Flow



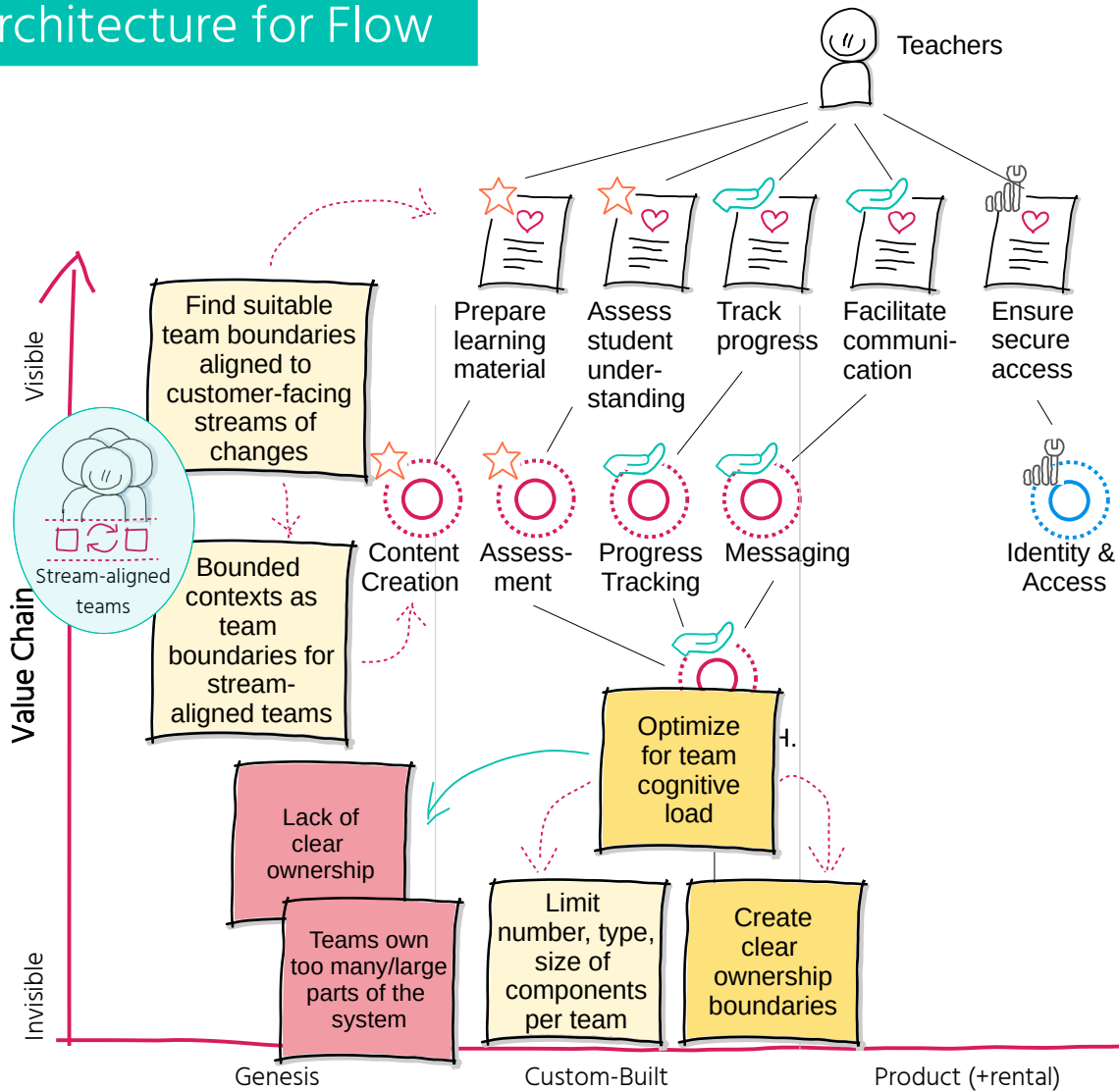
# Architecture for Flow



# Architecture For Flow



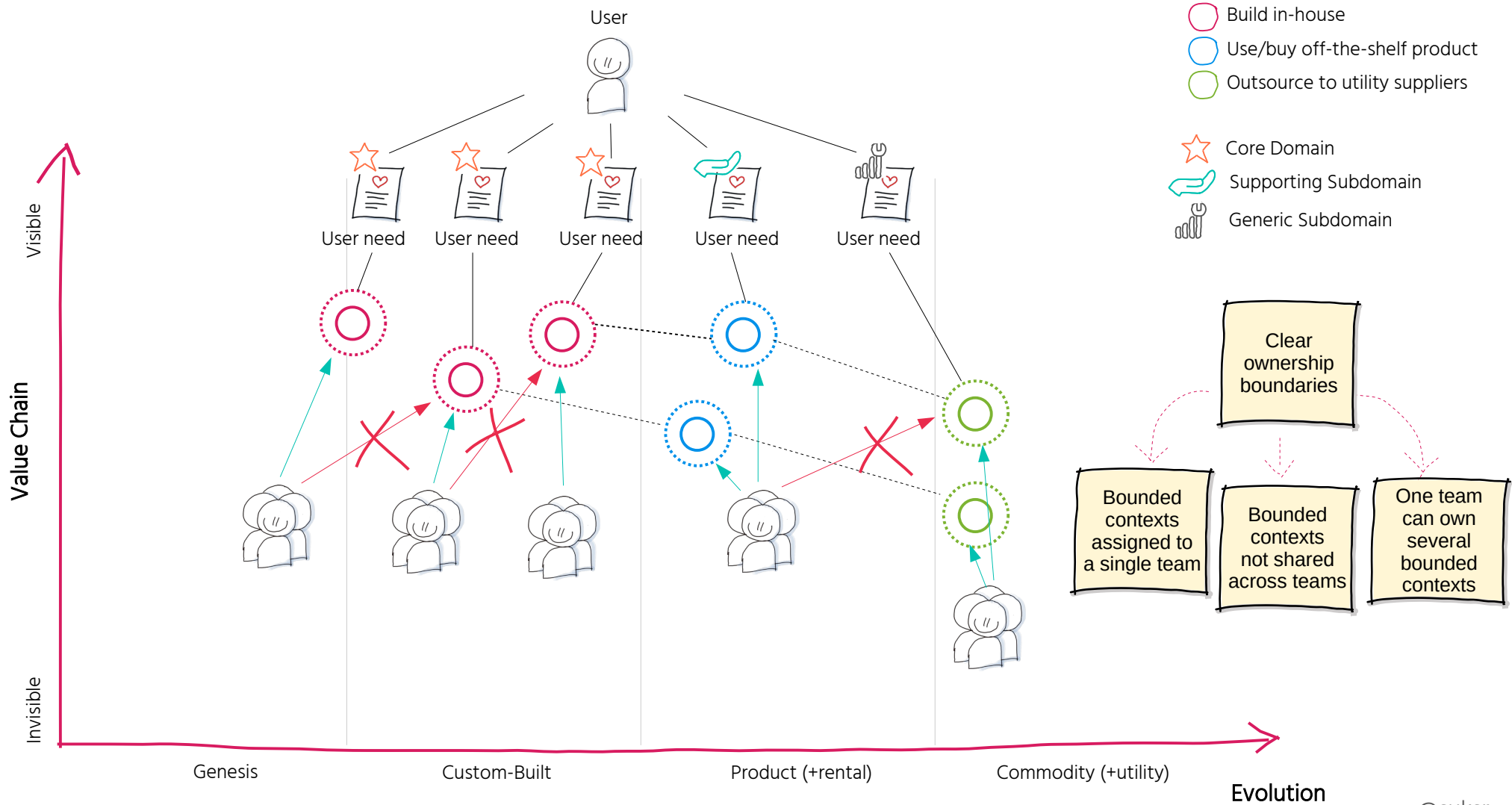
# Architecture for Flow



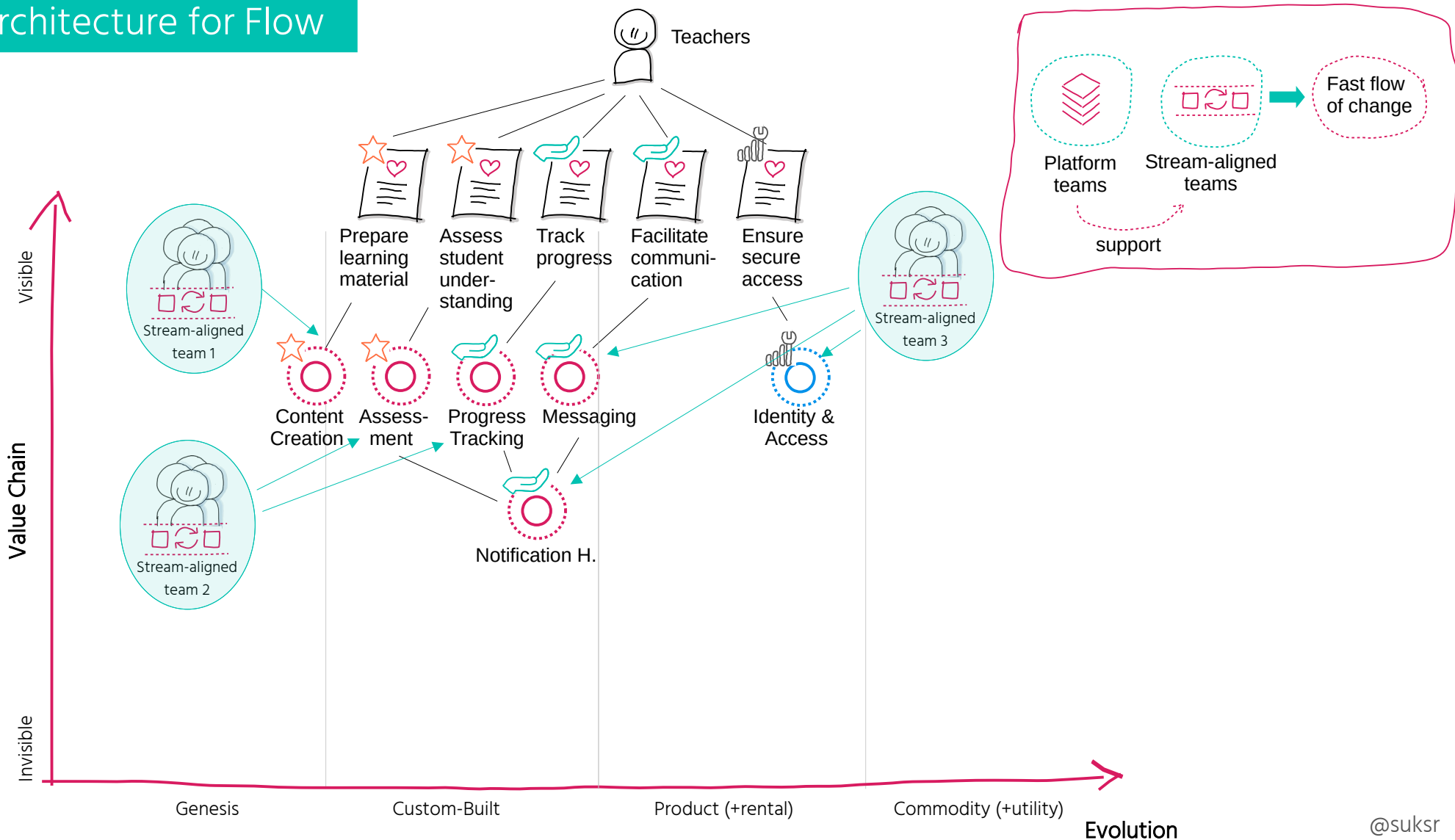
← unlocks blocker

Evolution

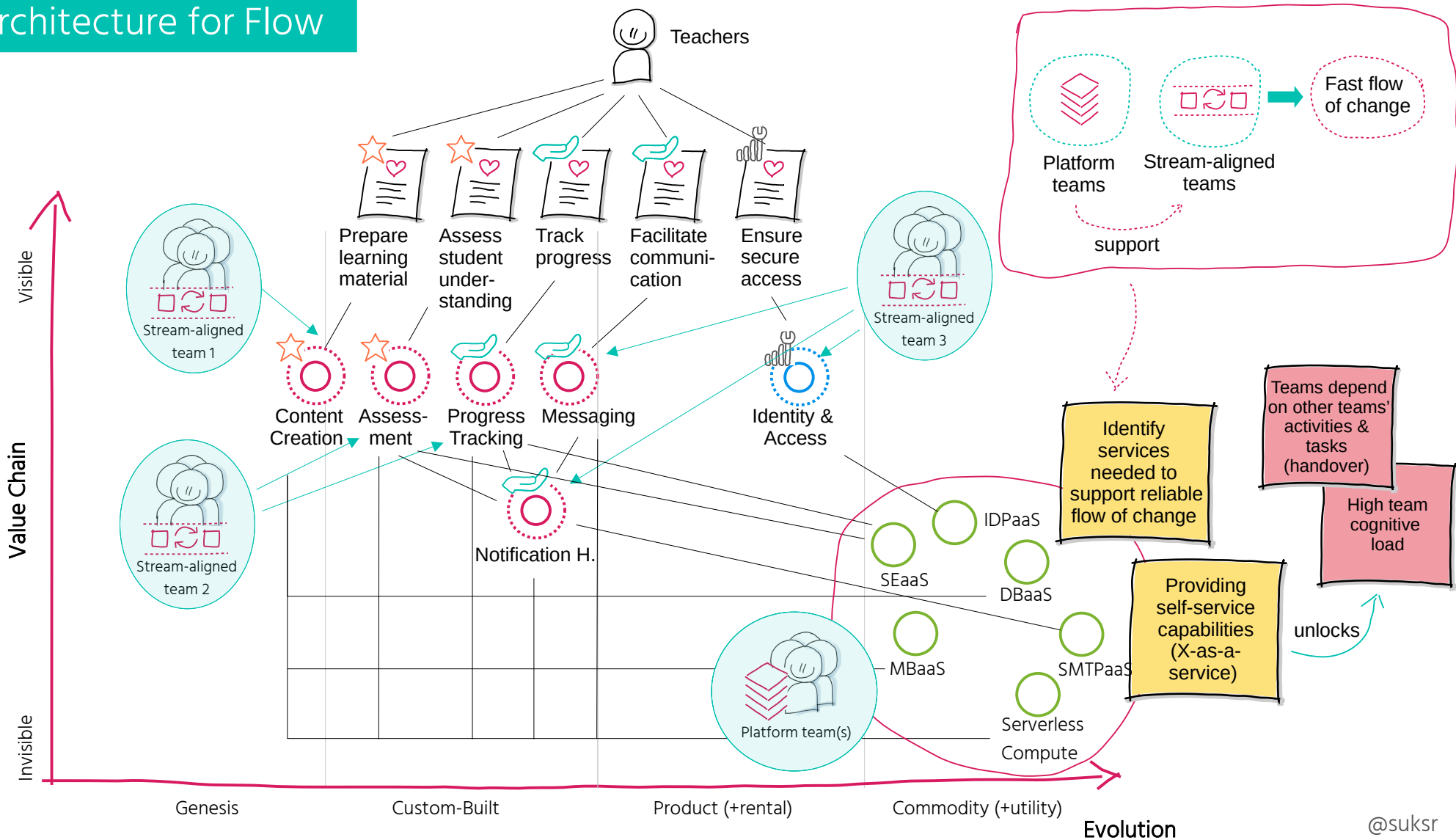
@suksr



# Architecture for Flow

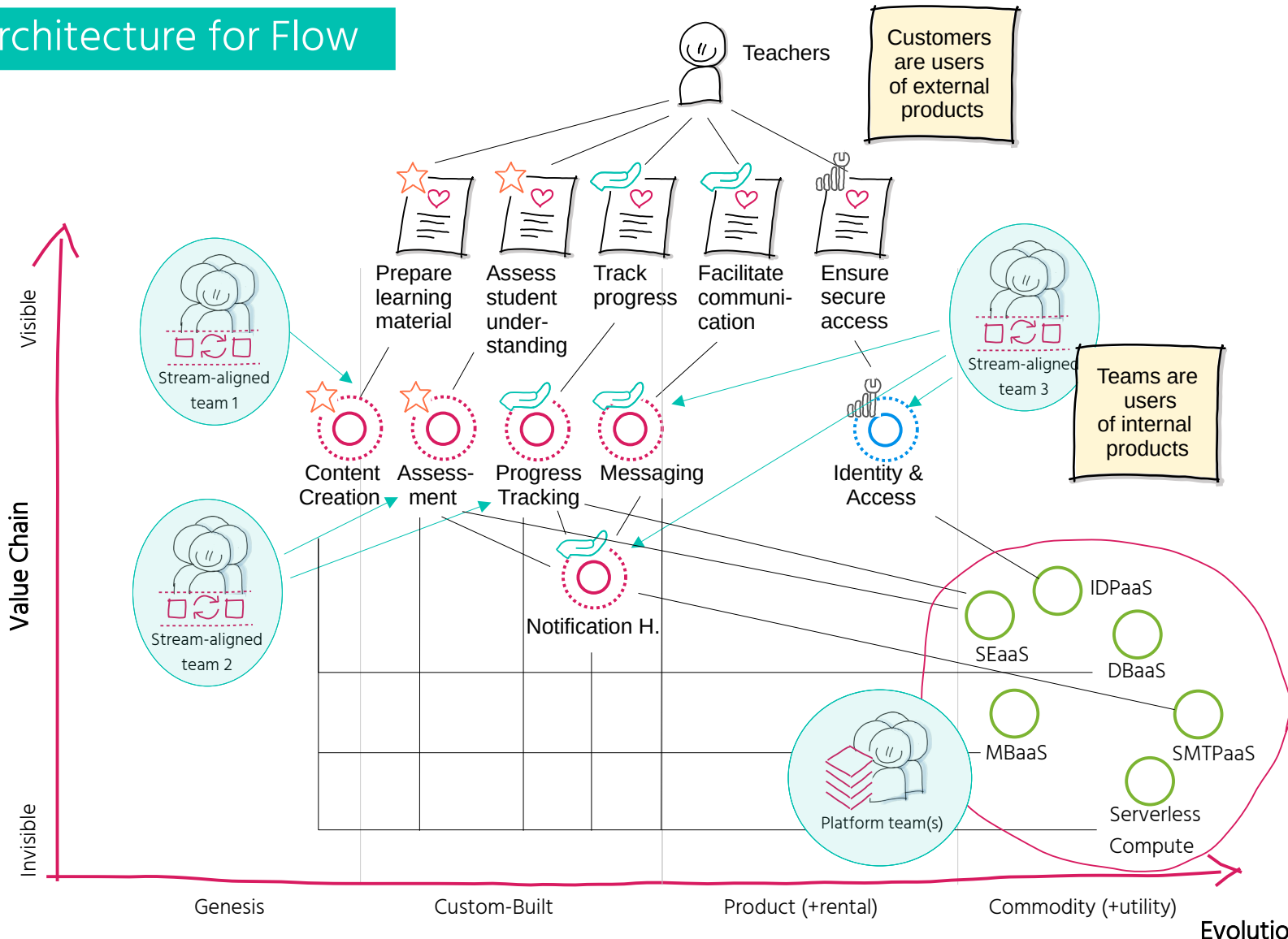


# Architecture for Flow

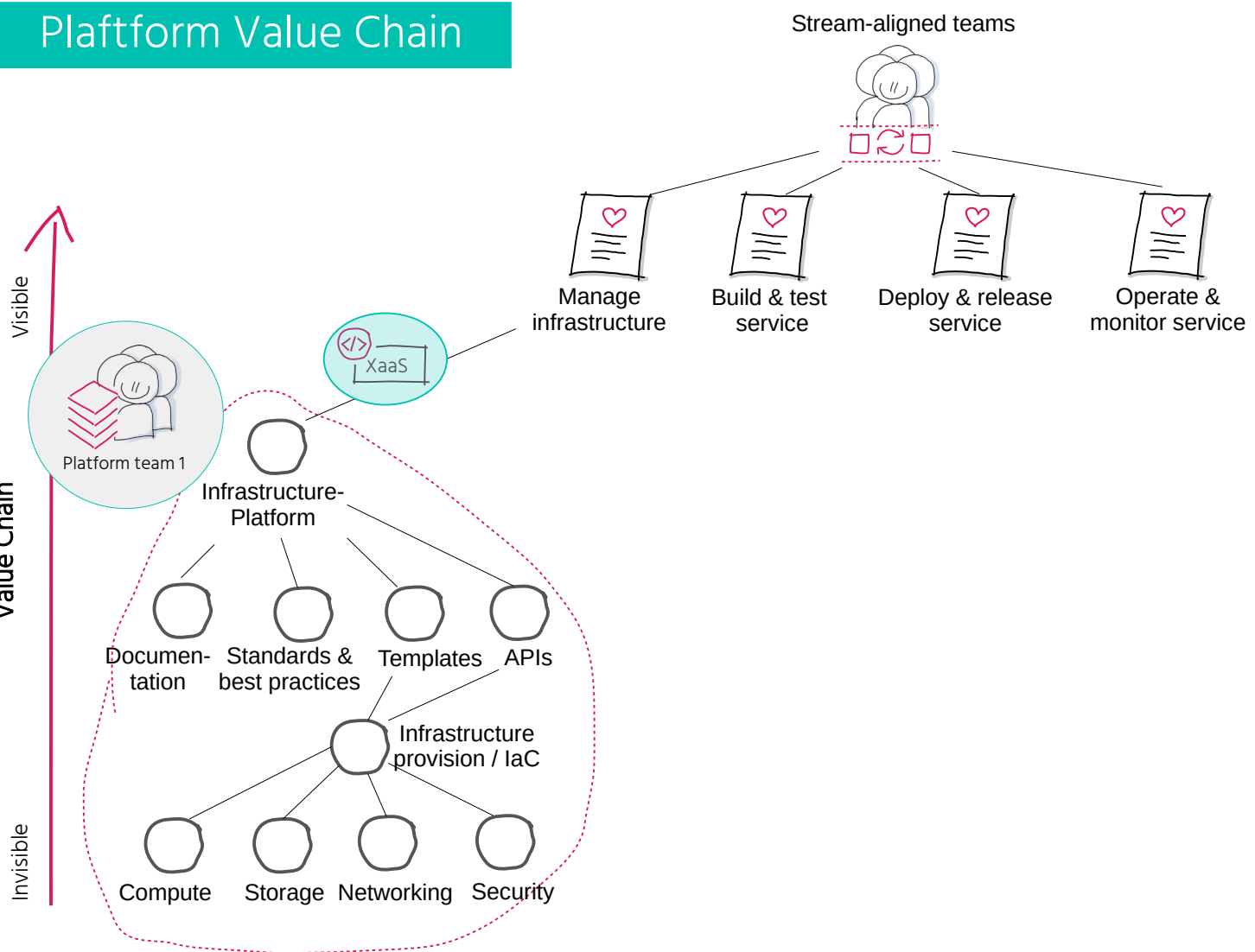




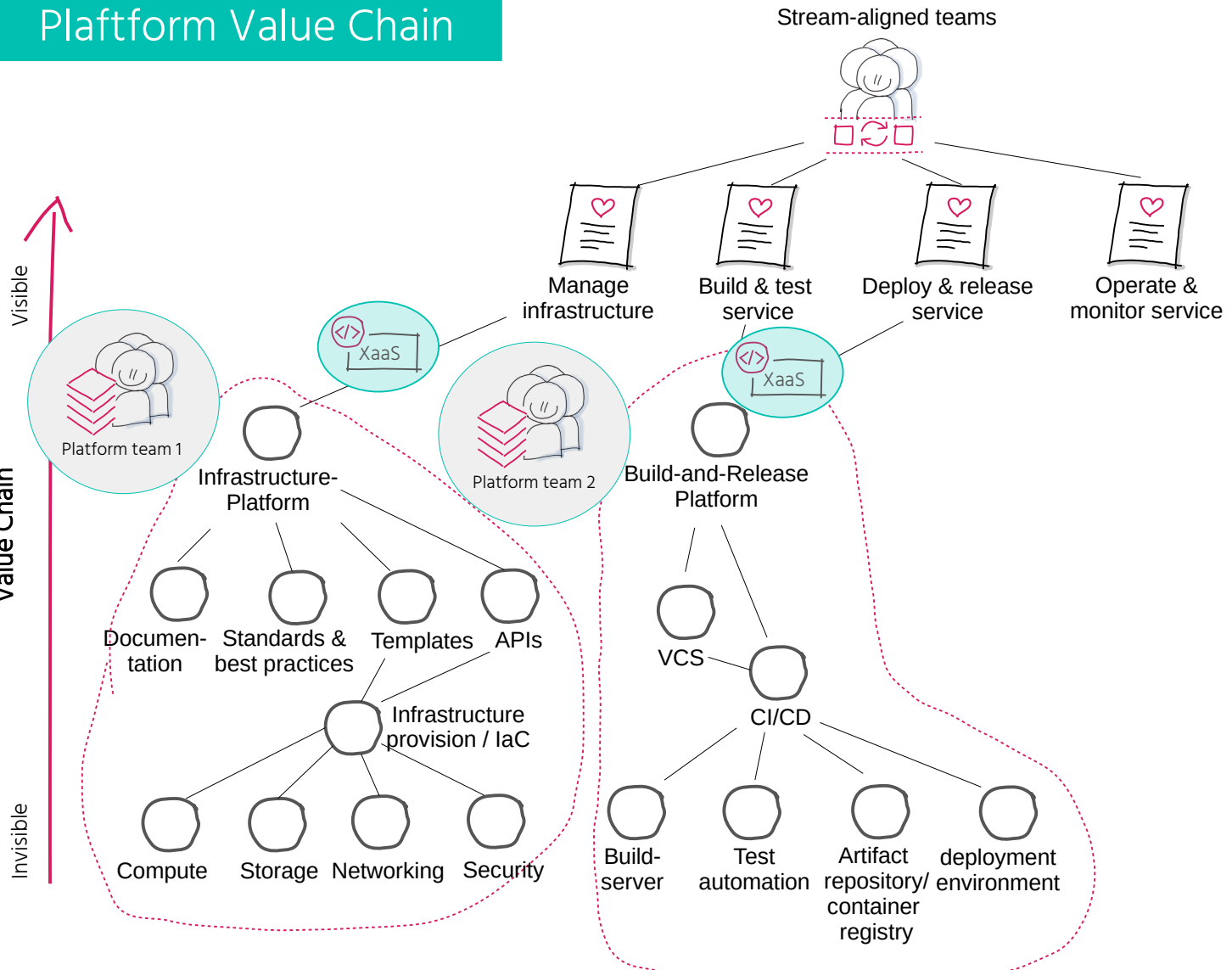
# Architecture for Flow



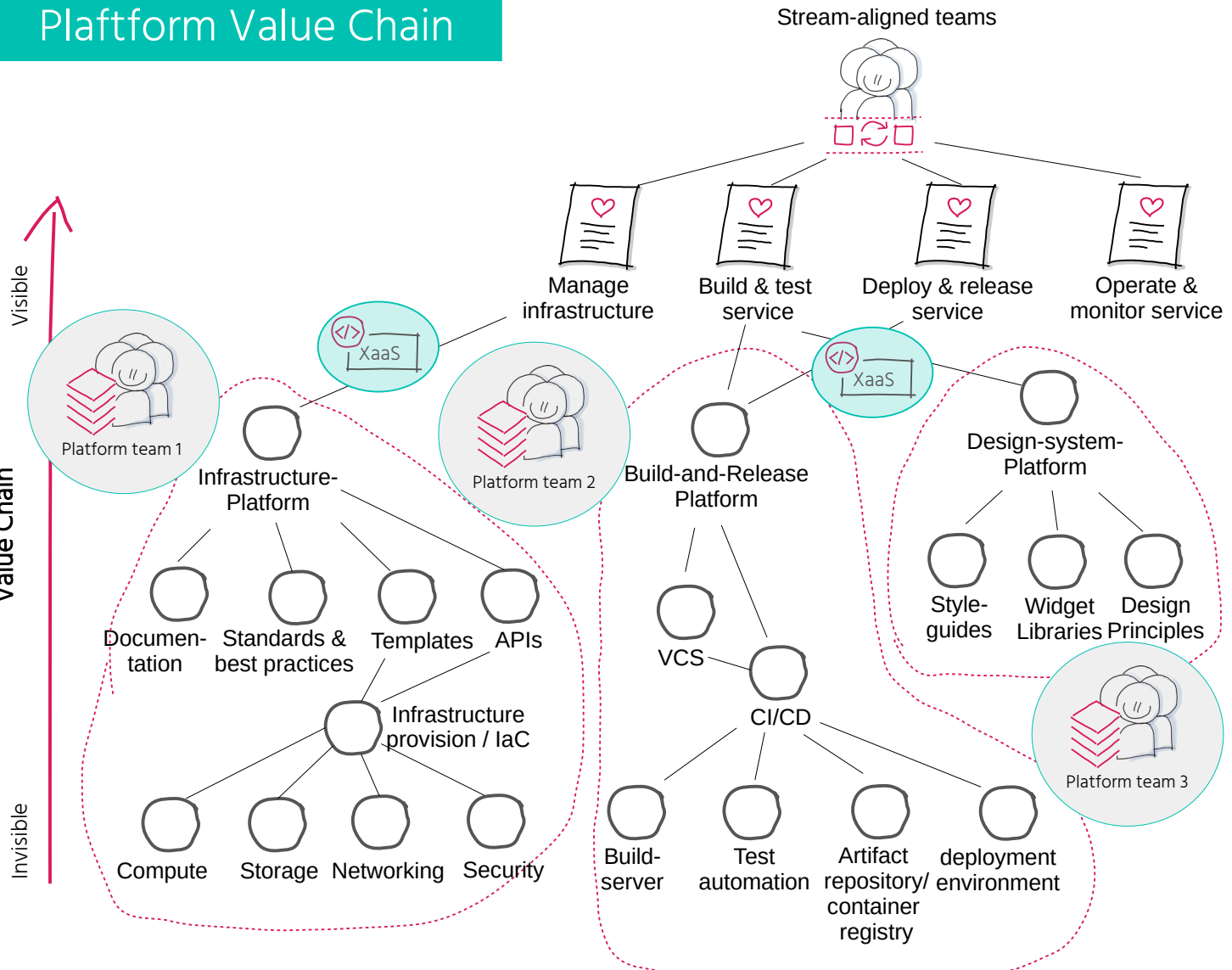
# Platform Value Chain



# Platform Value Chain

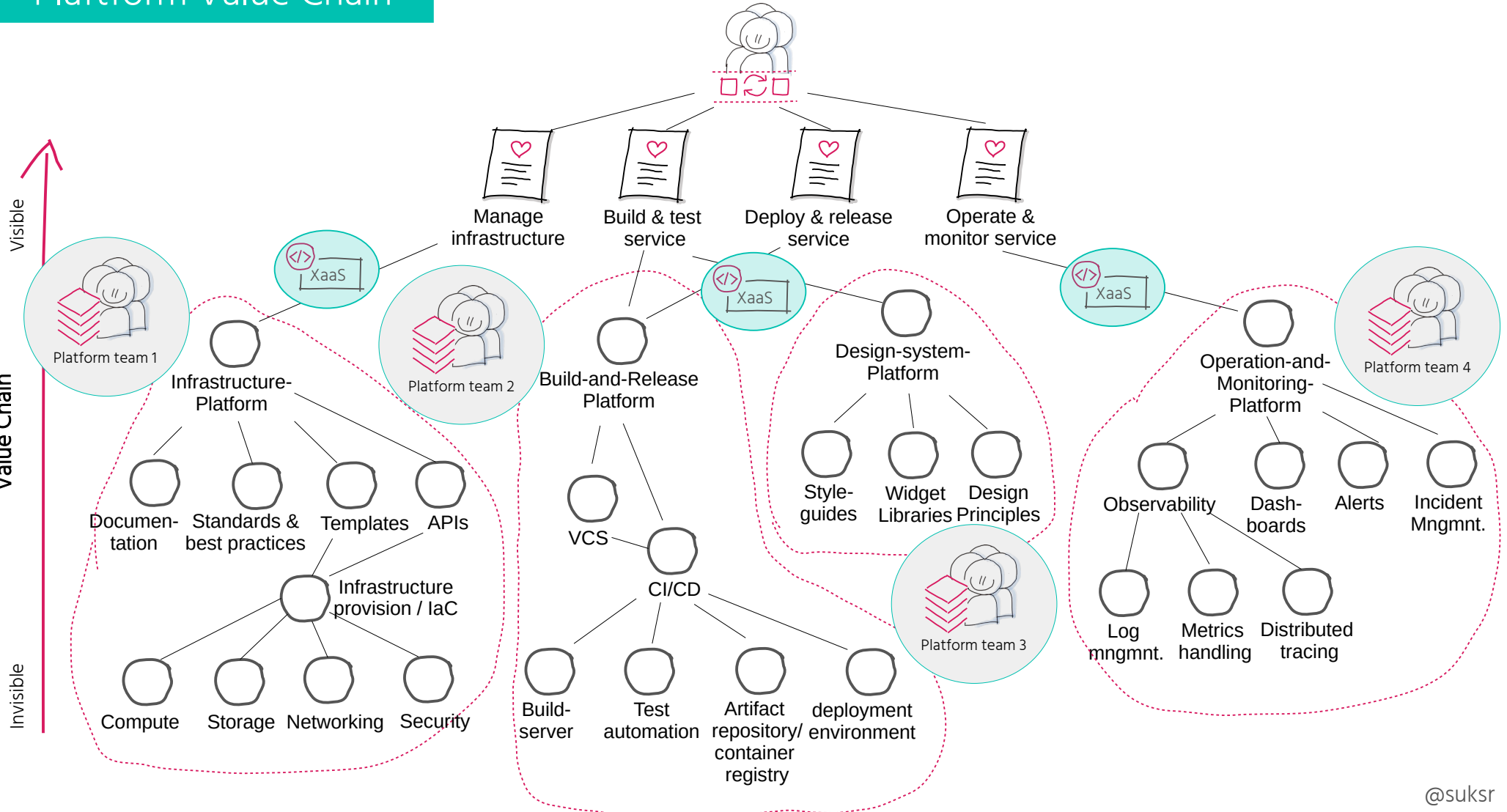


# Platform Value Chain

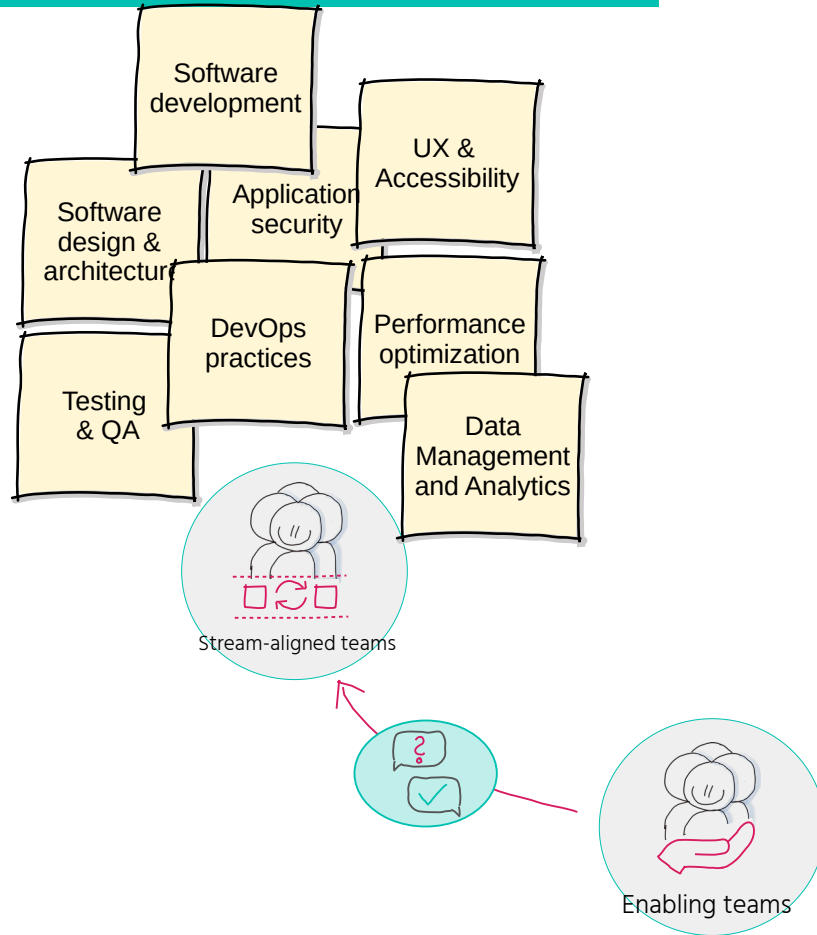


# Platform Value Chain

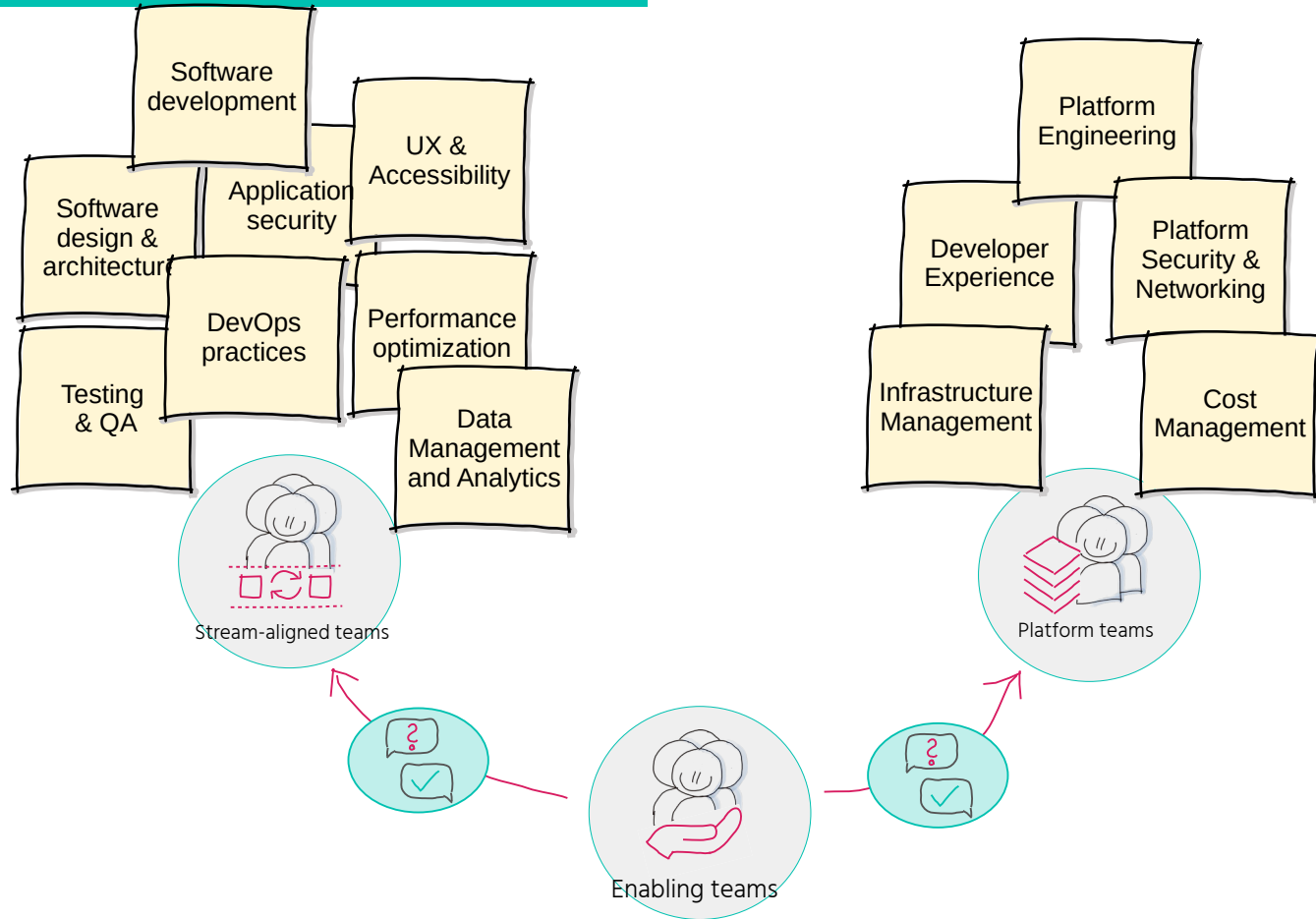
Stream-aligned teams



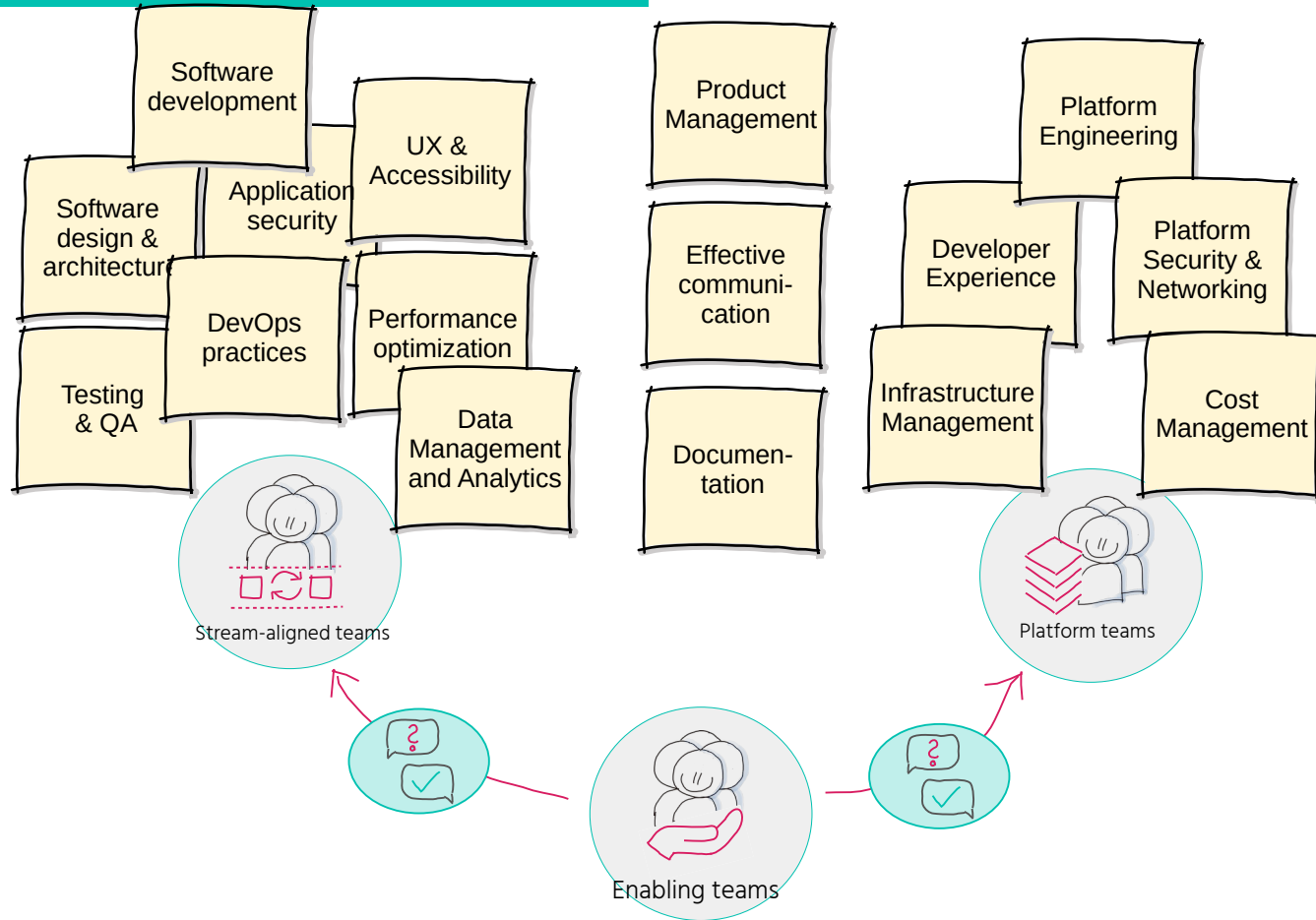
# Upskilling Teams on Missing Capabilities



# Upskilling Teams on Missing Capabilities

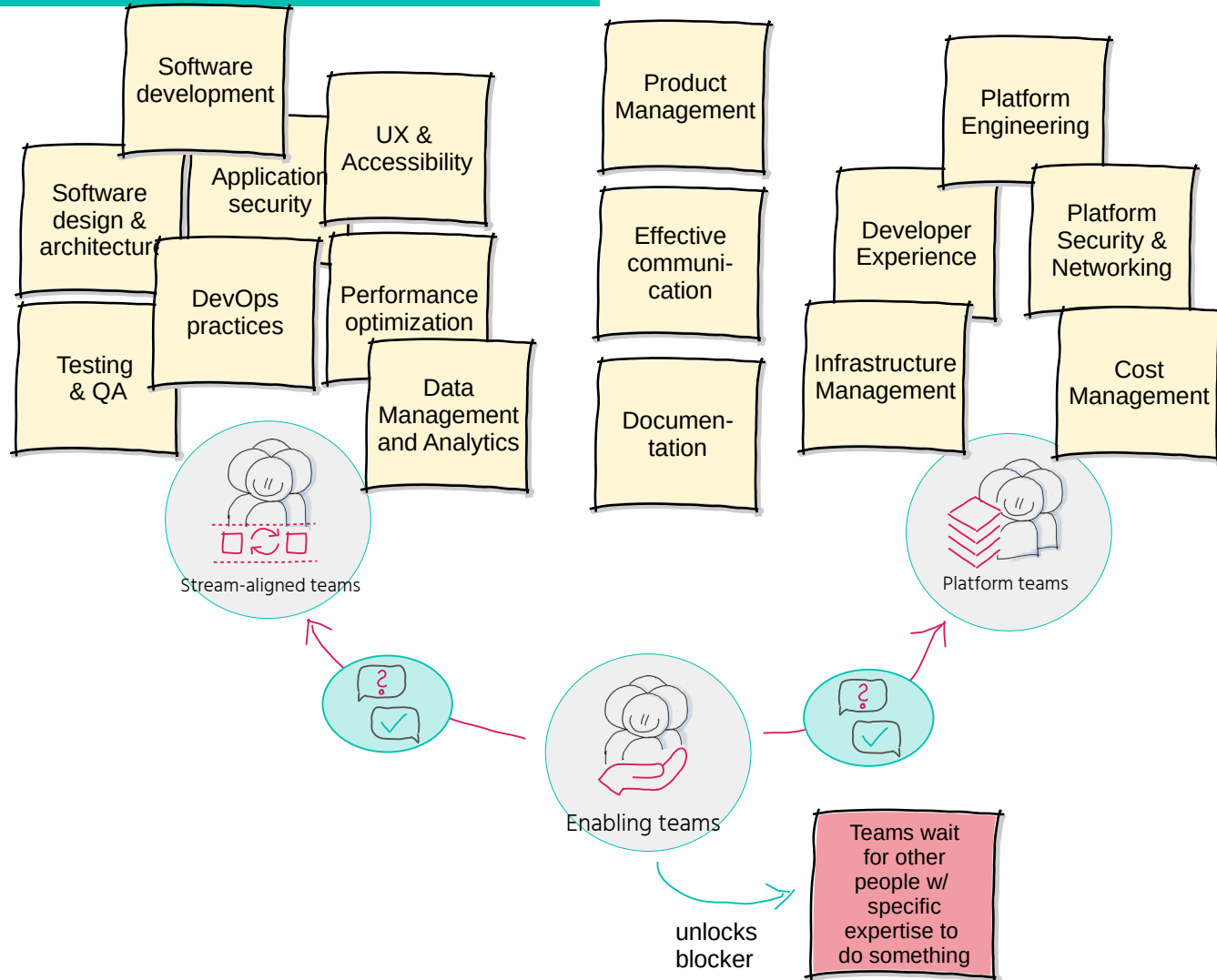


# Upskilling Teams on Missing Capabilities





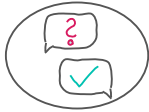
# Upskilling Teams on Missing Capabilities



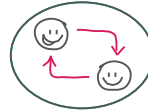
# Continuous Shared Learning & Experimentation

Adaptive to complex situations w/ unknown unknowns

Enabling teams as internal mentor, coaches to upskill teams

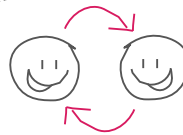


Collaboration for rapid discovery



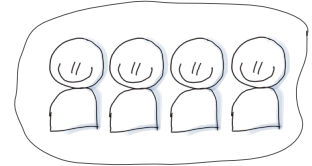
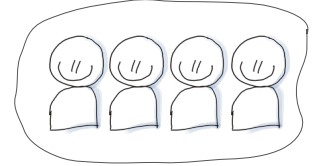
Collaboration for gaining domain knowledge

Domain Experts

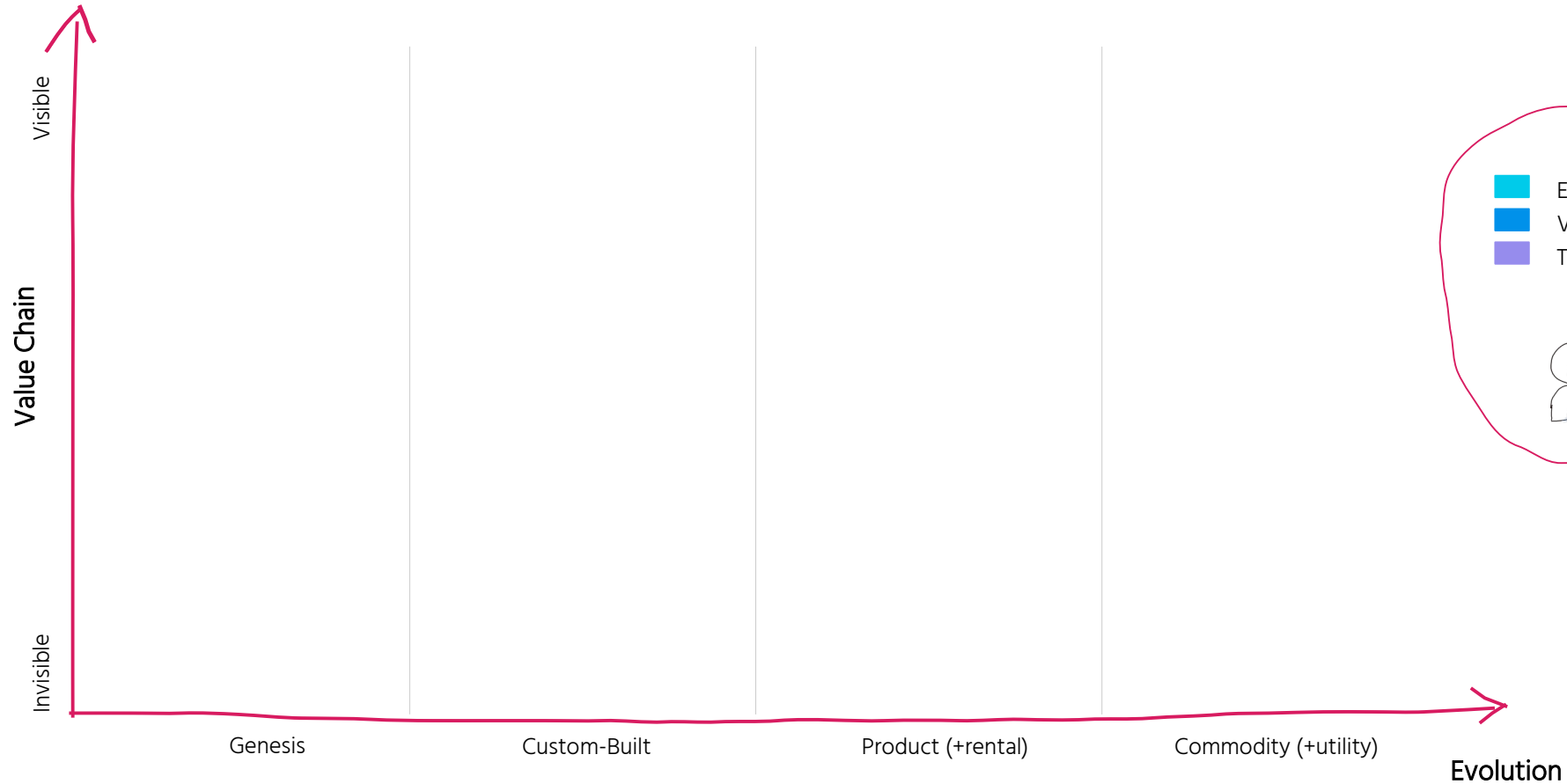


Development Teams

Communities of Practice



# A Mix of Mindsets per Team

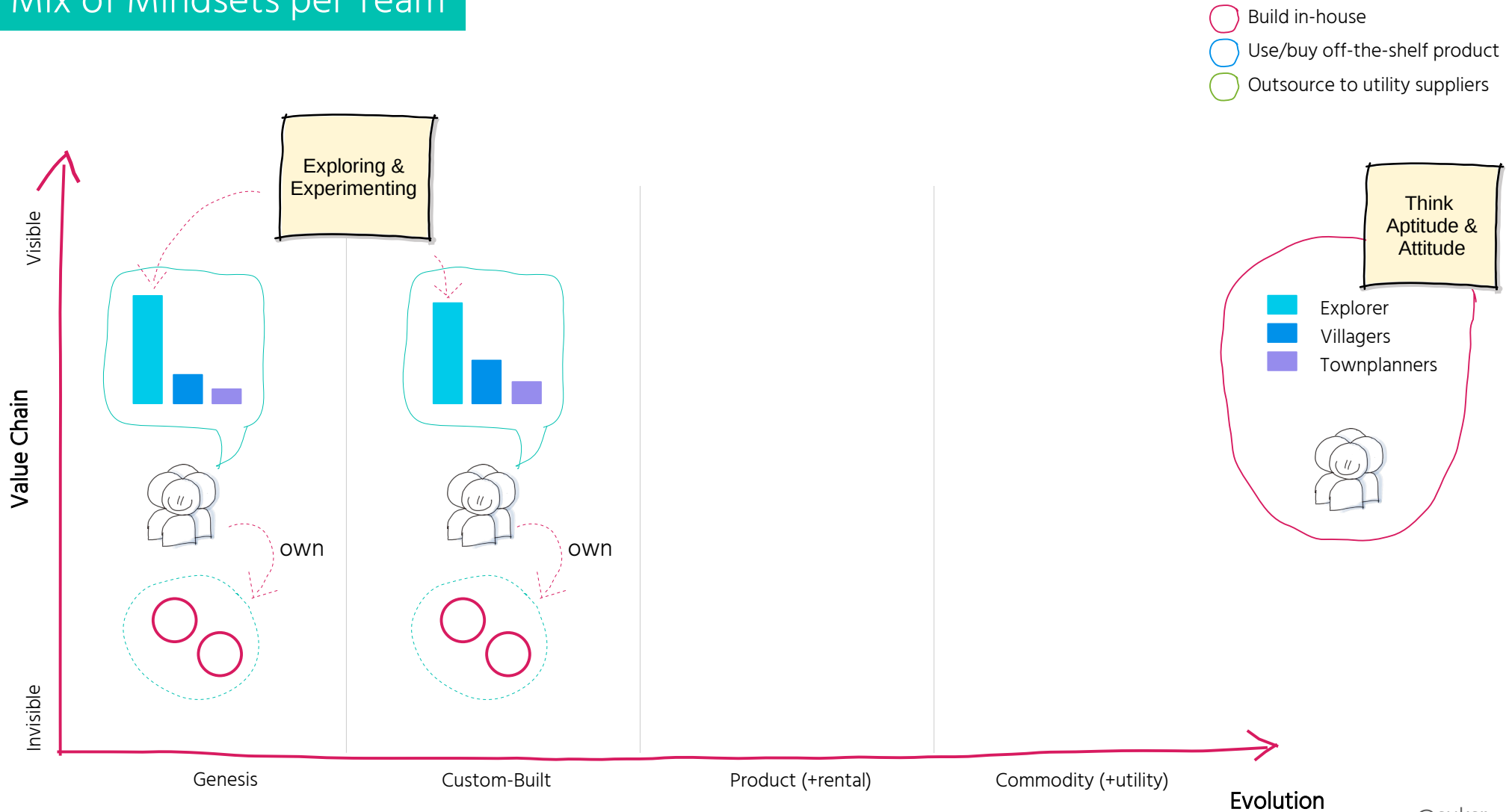


- Build in-house
- Use/buy off-the-shelf product
- Outsource to utility suppliers

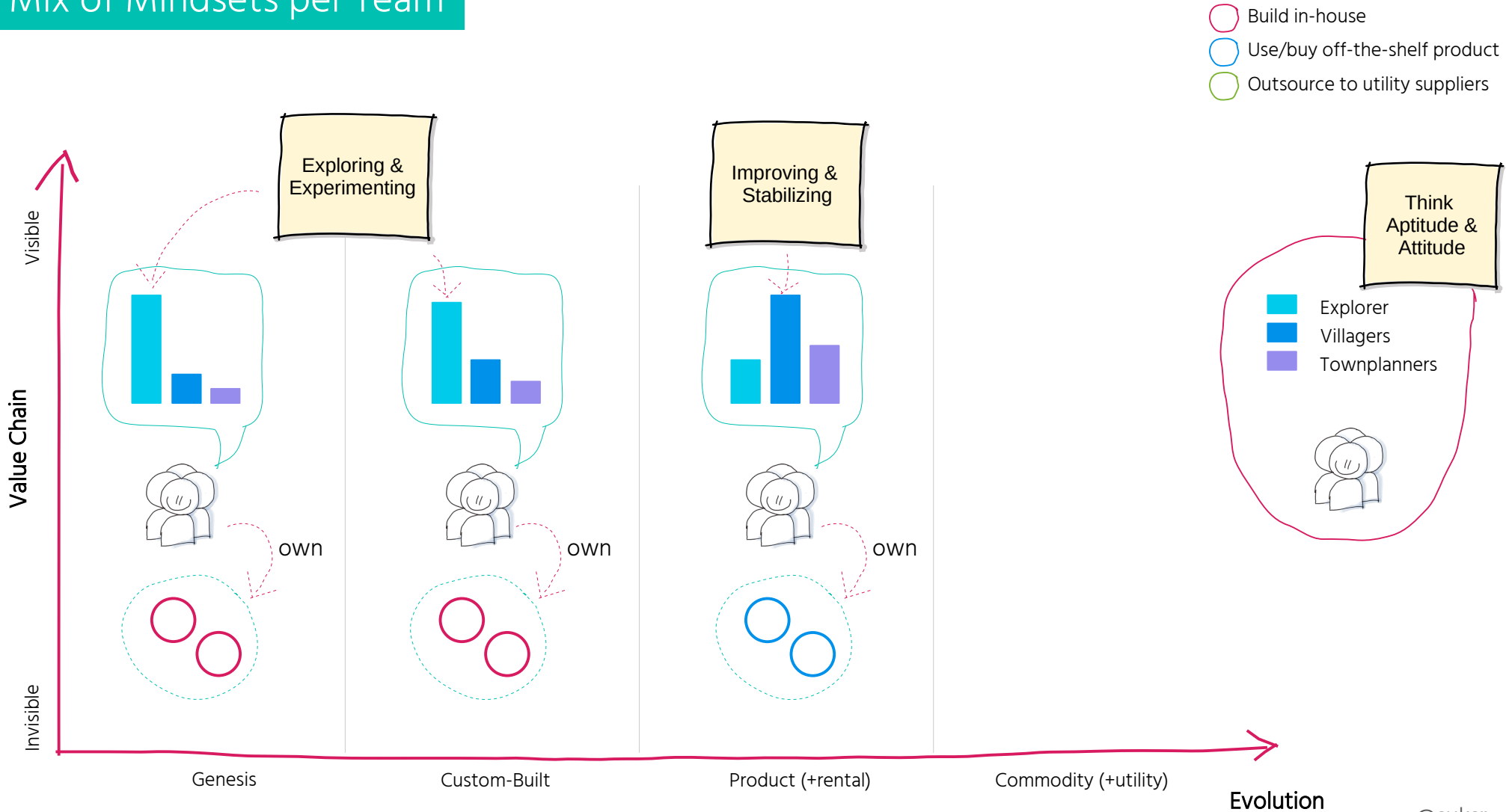
Think Aptitude & Attitude

- Explorer
- Villagers
- Townplanners

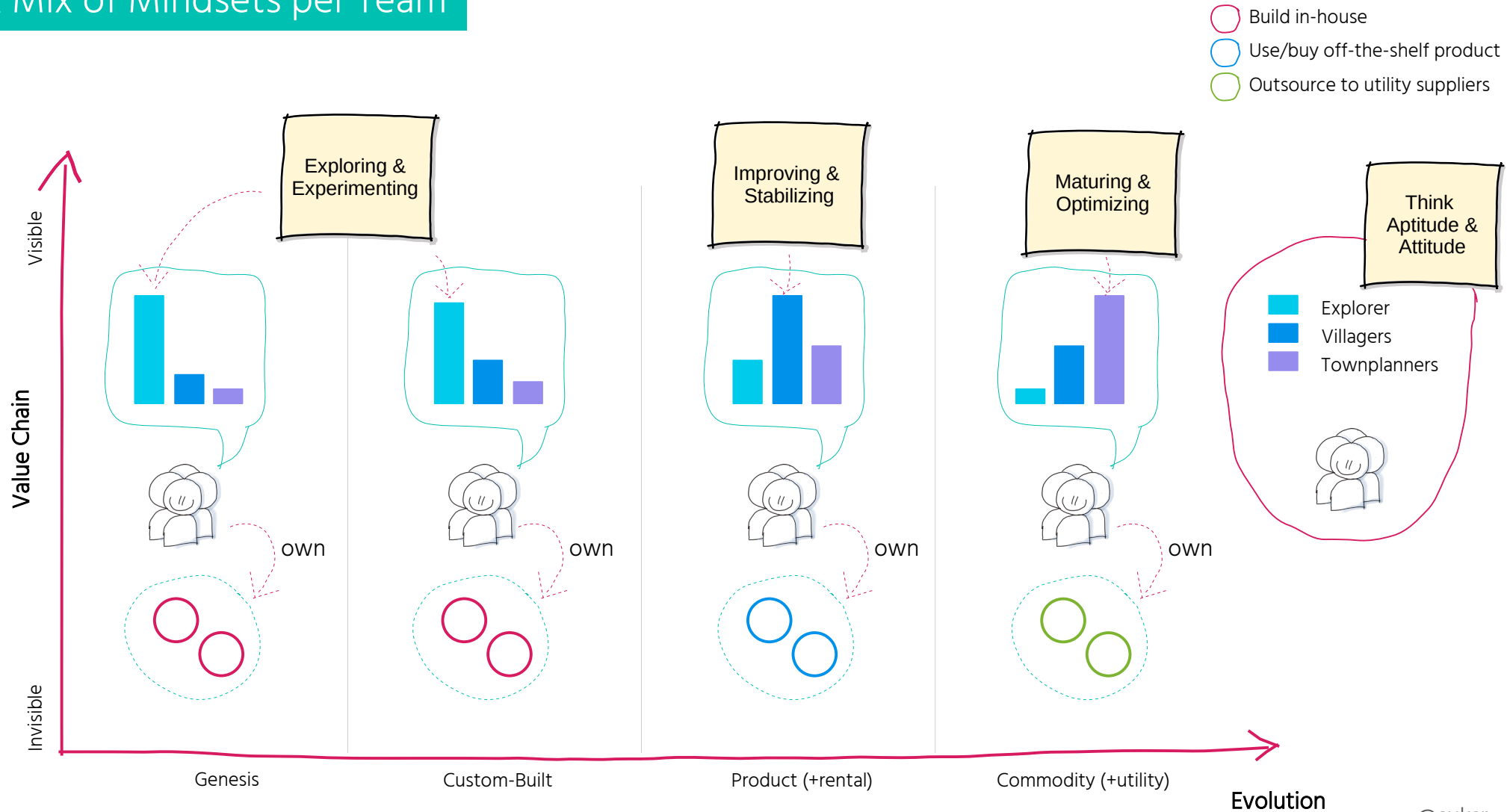
# A Mix of Mindsets per Team



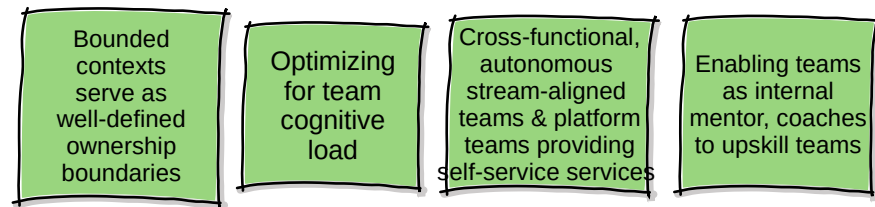
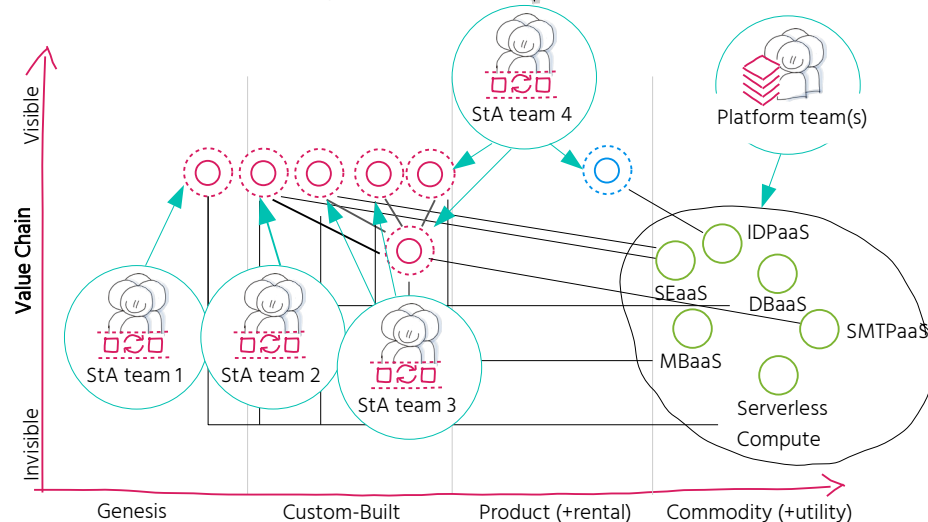
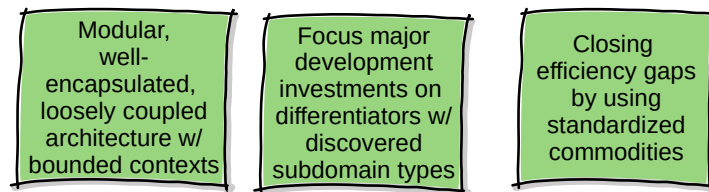
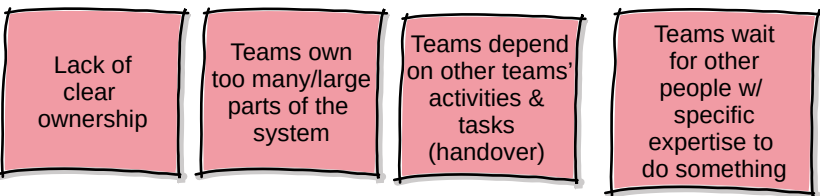
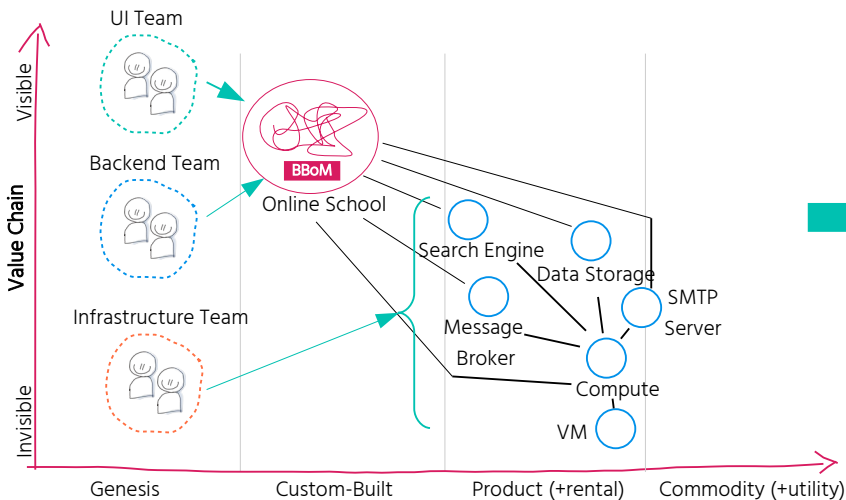
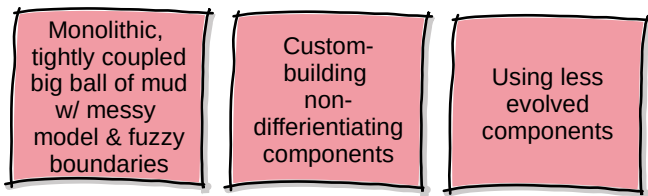
# A Mix of Mindsets per Team



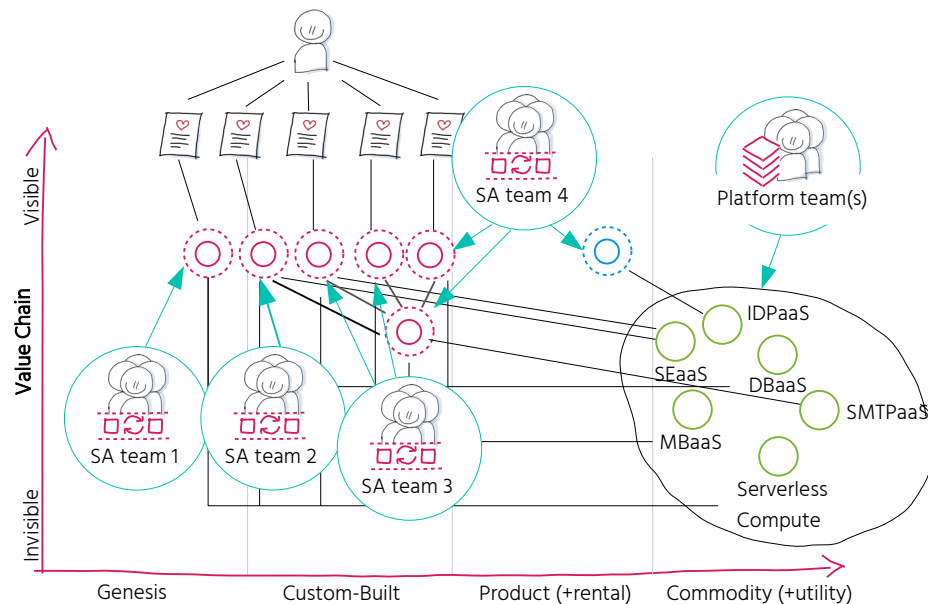
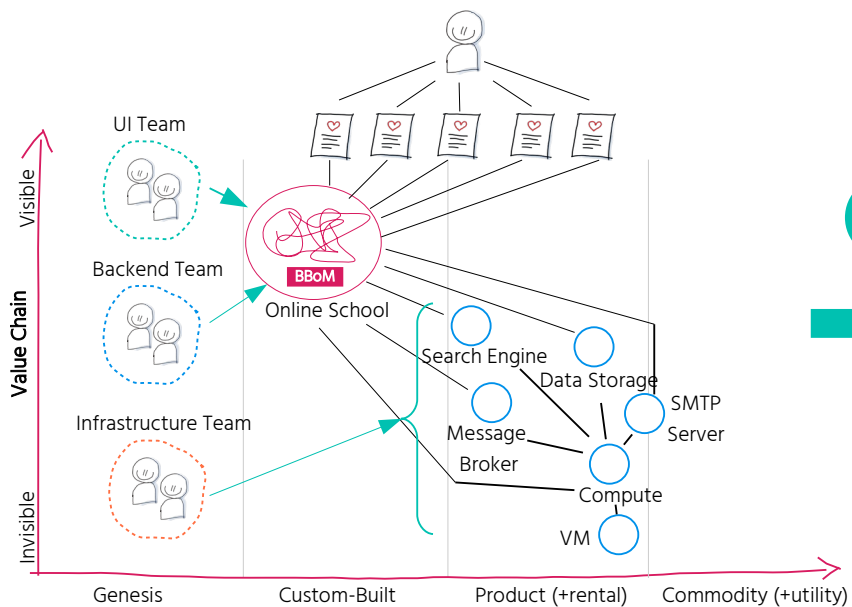
# A Mix of Mindsets per Team



# Unlocking Blockers to Flow

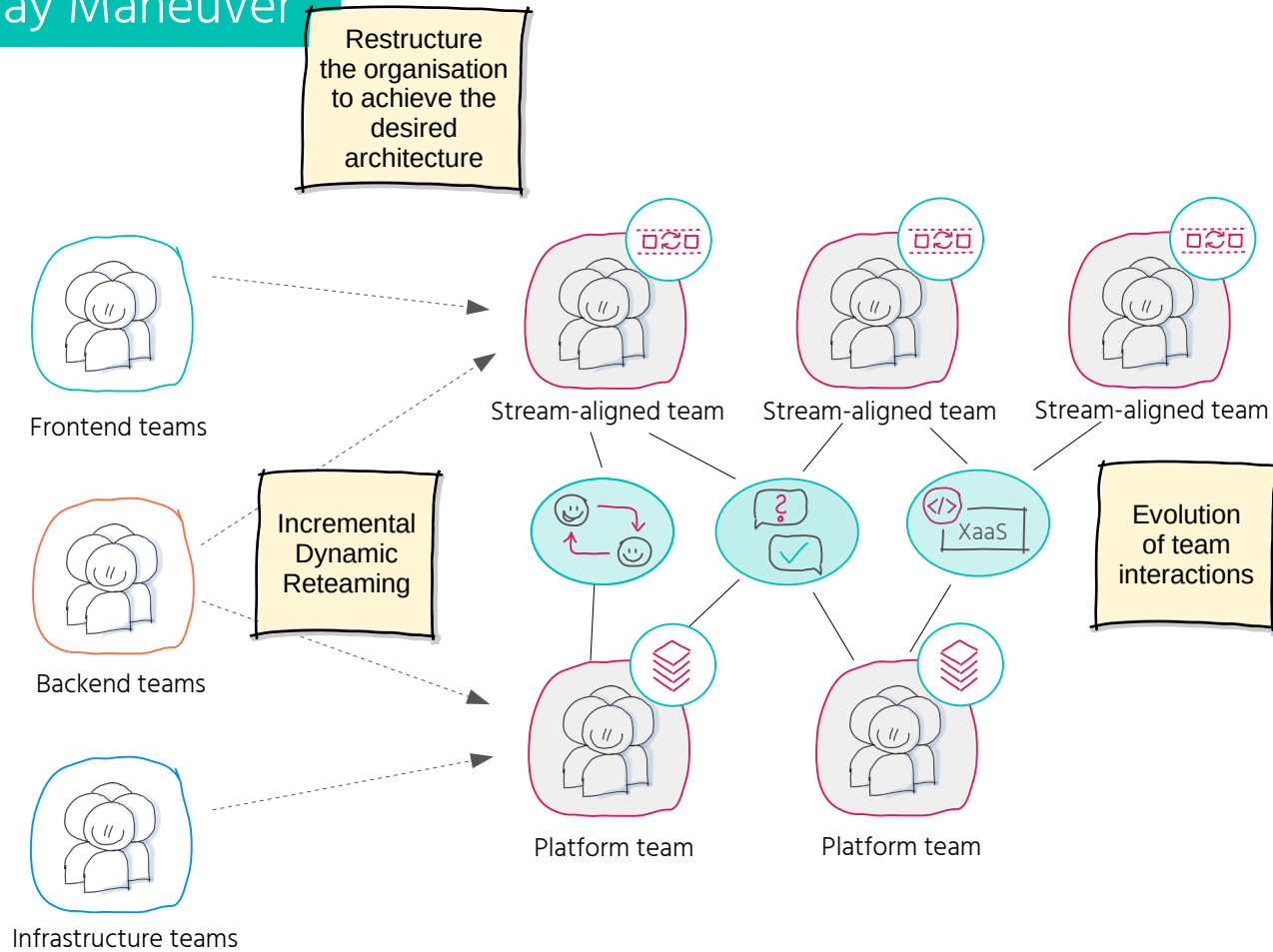


# How to transition?





# Reverse Conway Maneuver



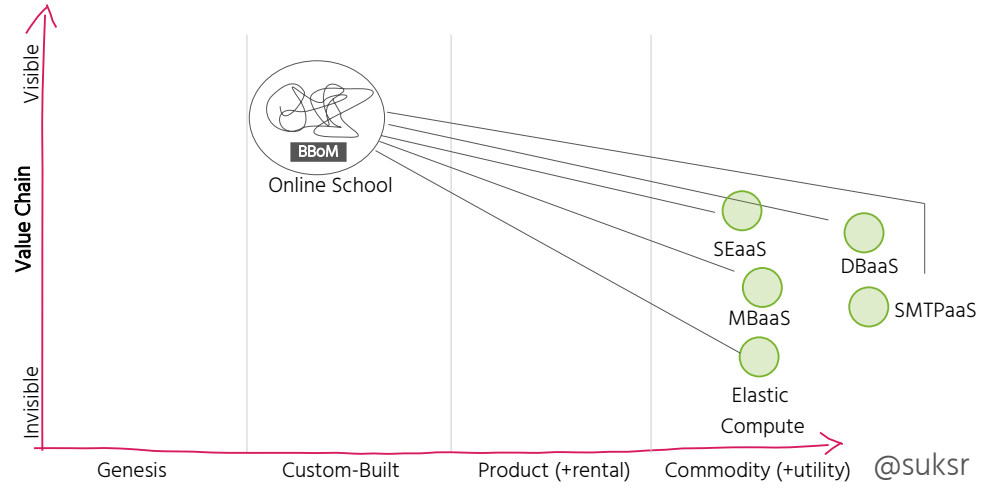
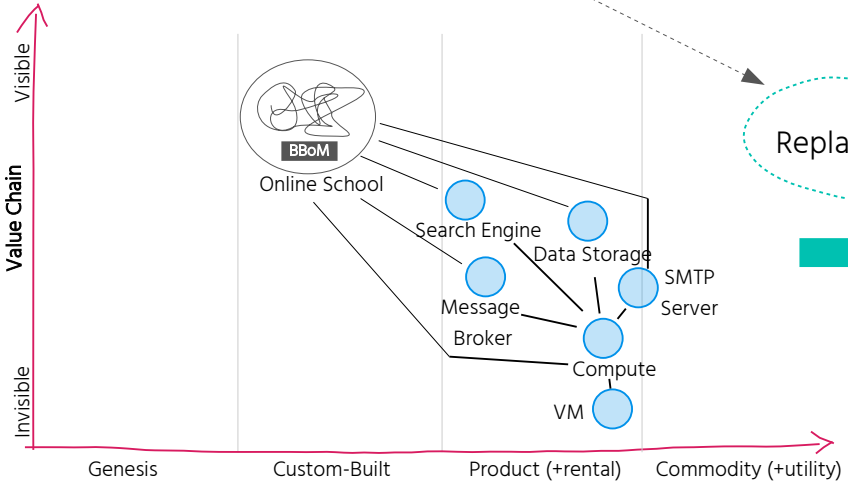
# Evolution of Team Topologies



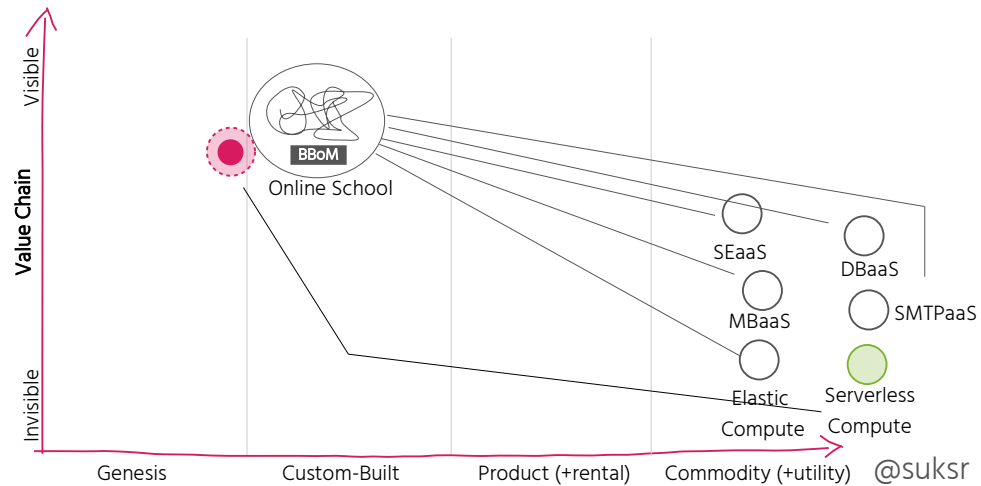
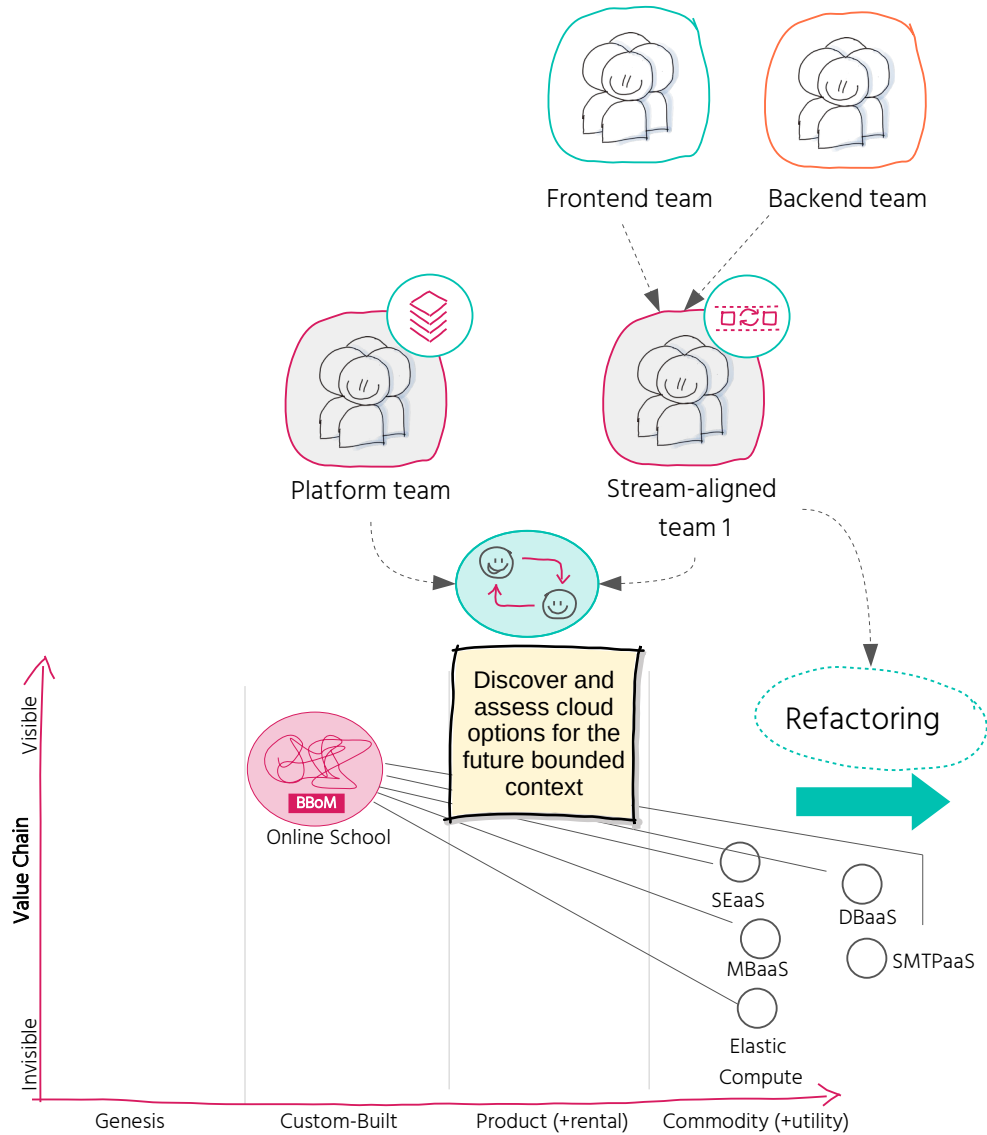
Frontend team    Backend team    Infrastructure team



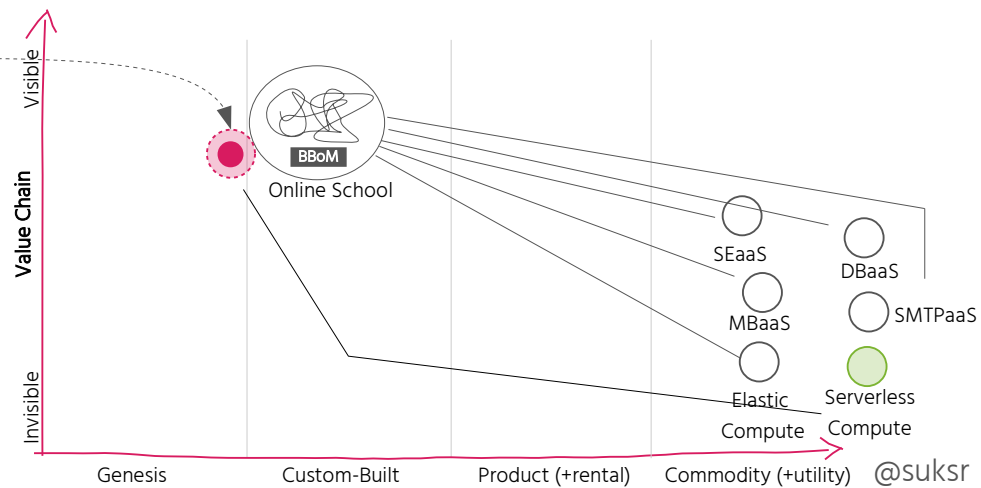
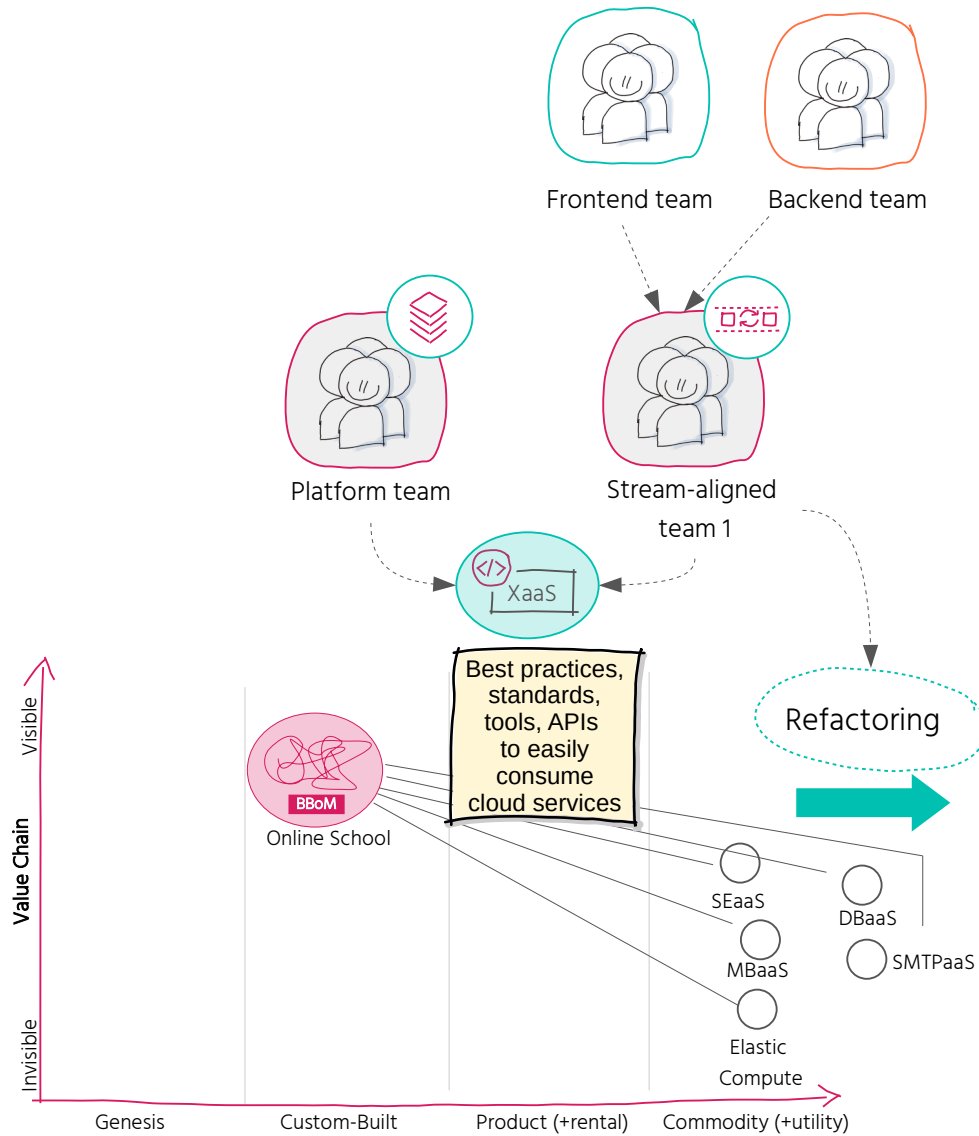
Platform team



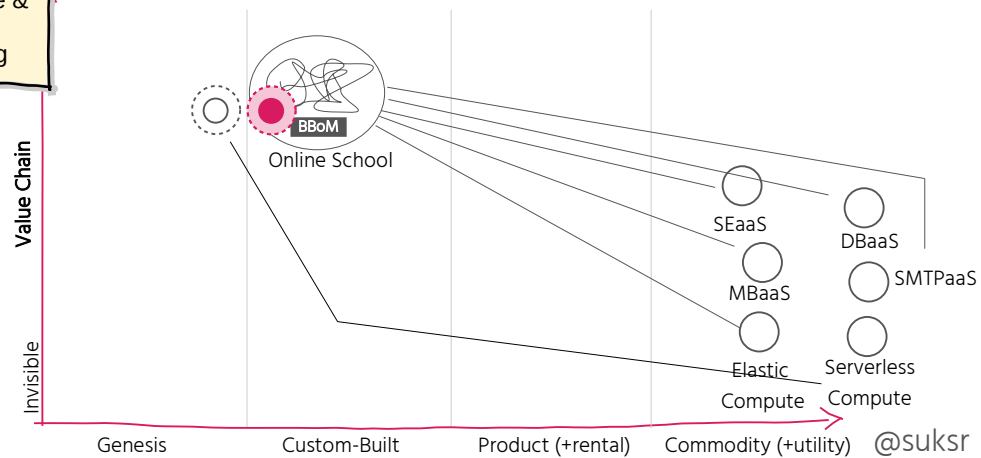
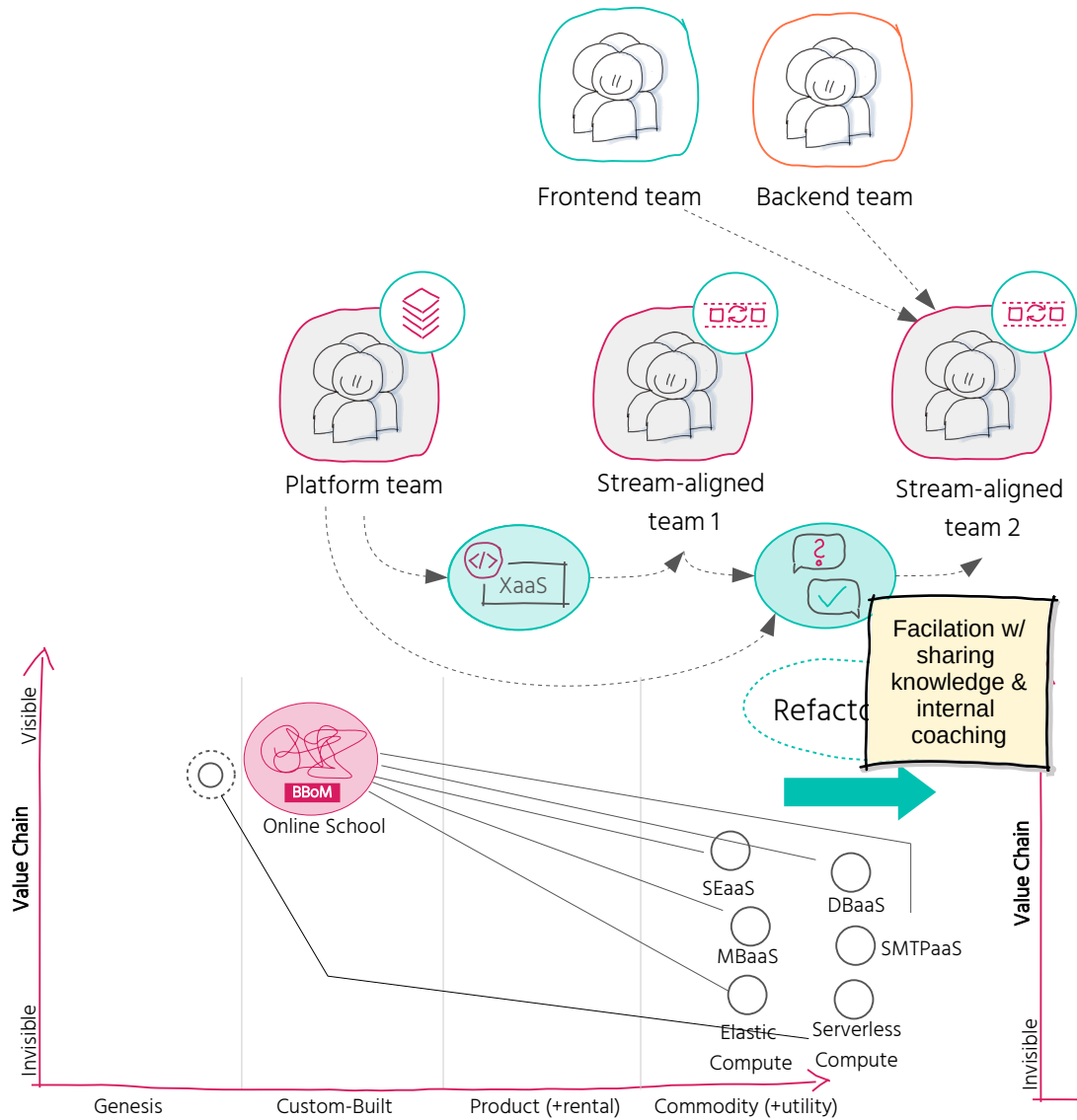
# Evolution of Team Topologies



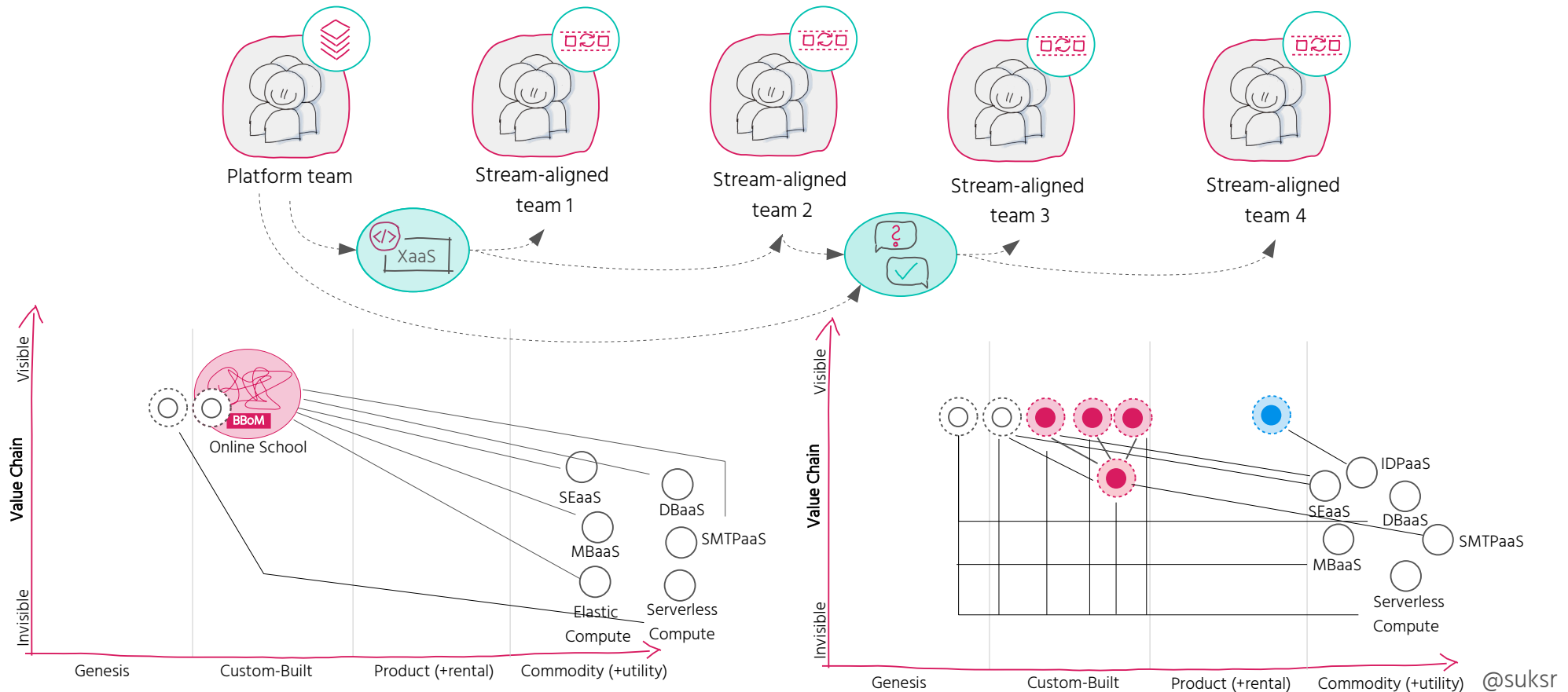
# Evolution of Team Topologies



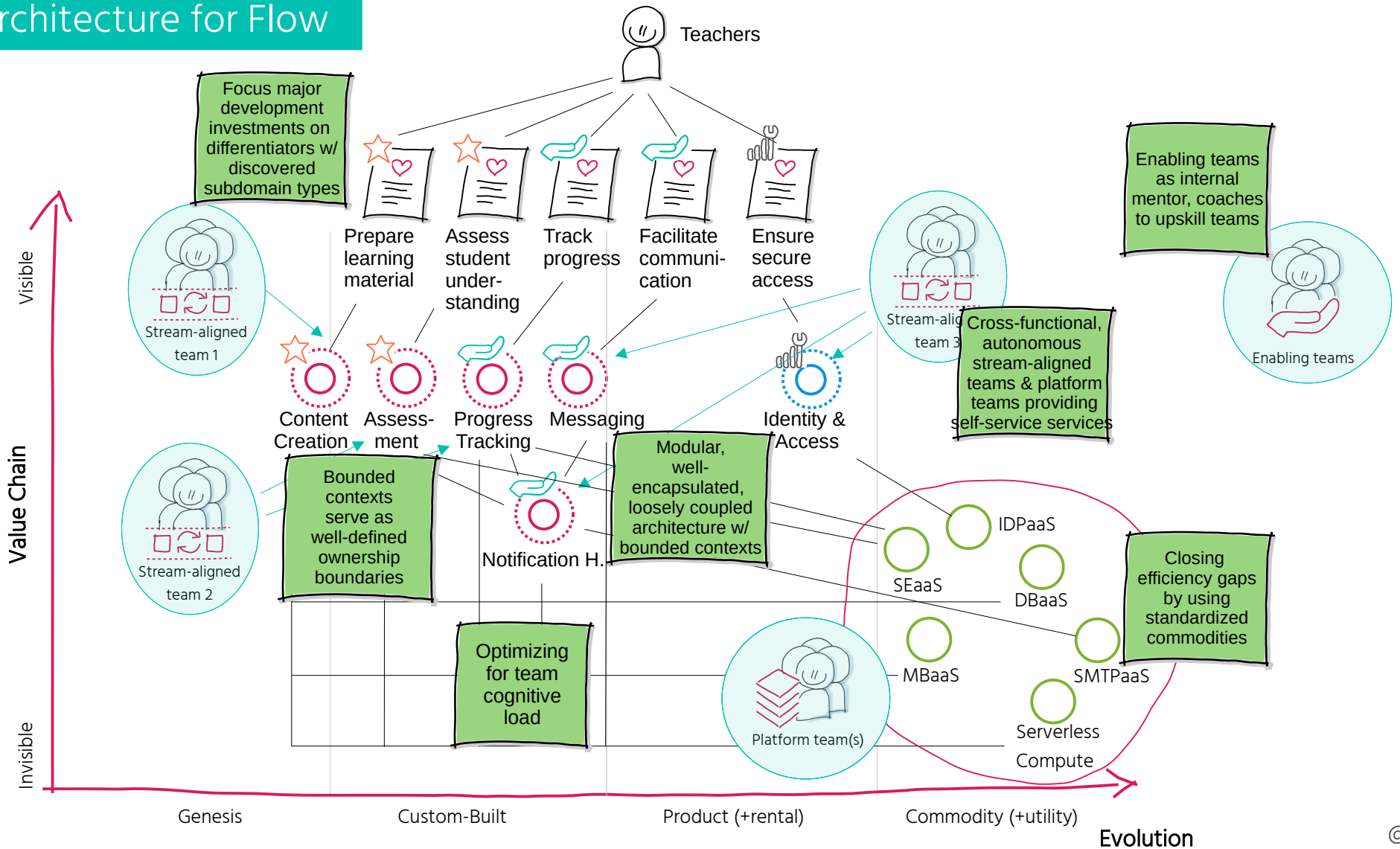
# Evolution of Team Topologies



# Evolution of Team Topologies



# Architecture for Flow



# Summary

WARDLEY MAPS  
TOPOGRAPHICAL  
INTELLIGENCE IN  
BUSINESS

Domain-Driven  
**DESIGN**  
Tackling Complexity in the Heart of Software

TEAM  
TOPOLOGIES

Generate value aligned to user needs

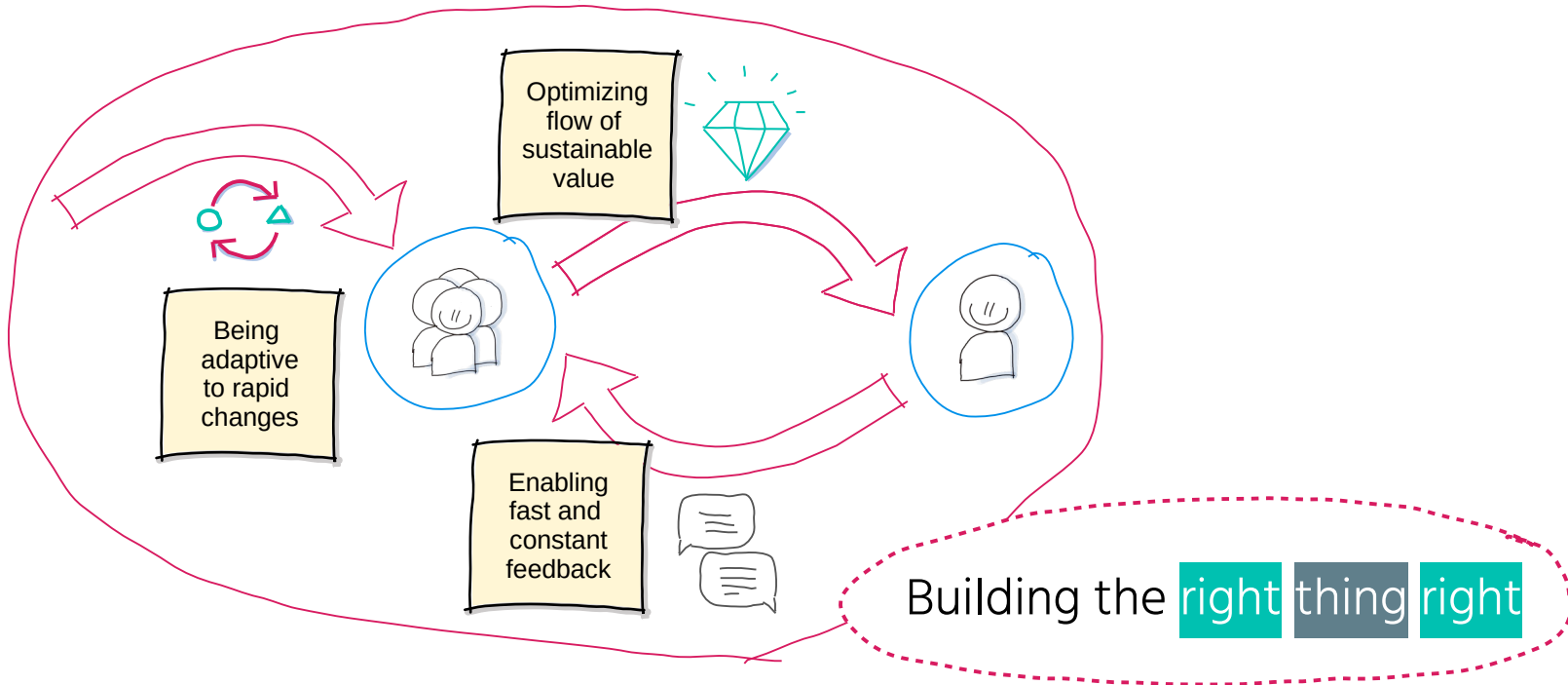
Unlock blockers to flow

Focus major development investments on differentiators

Close potential efficiency gaps

Align teams & evolve team interaction to system & strategy

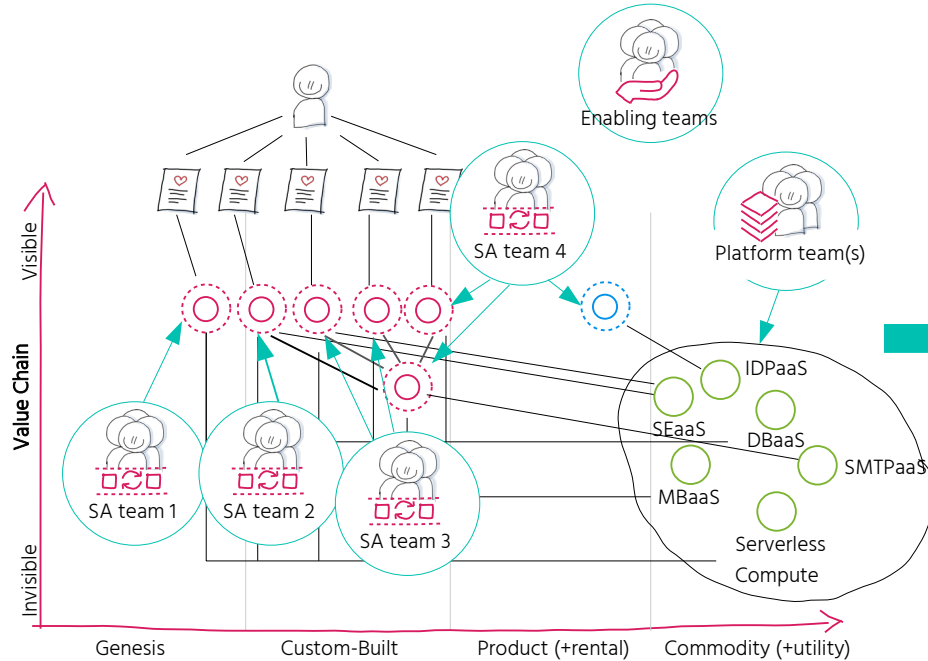
Establish learning organisation



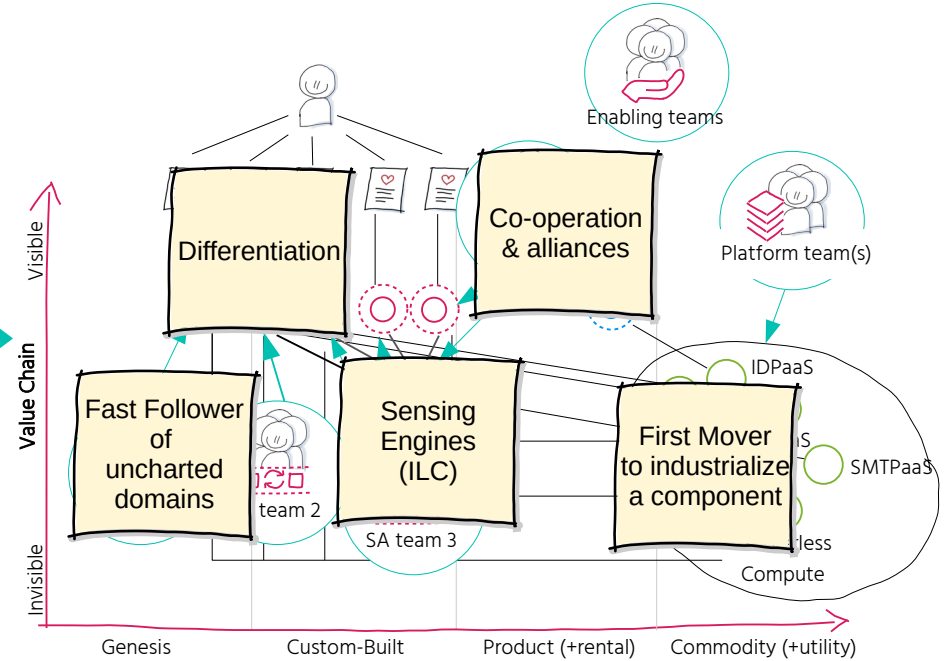
Building the right thing right



# Looking Ahead



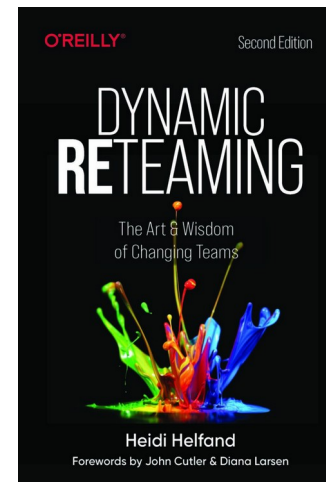
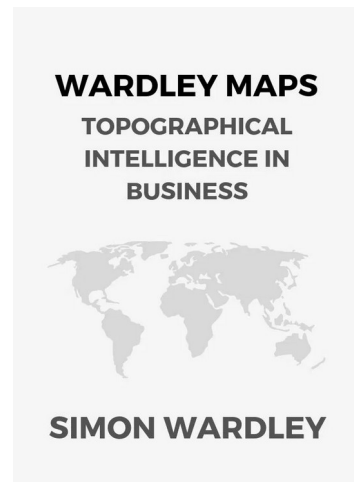
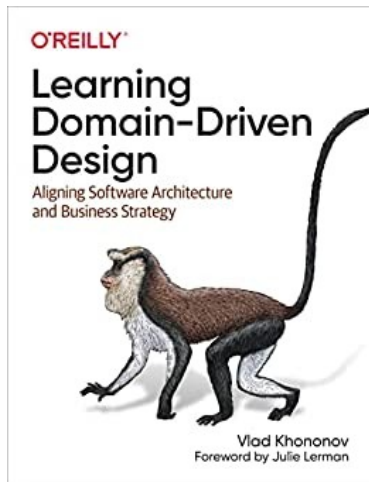
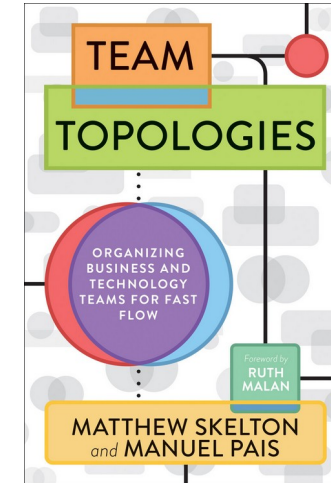
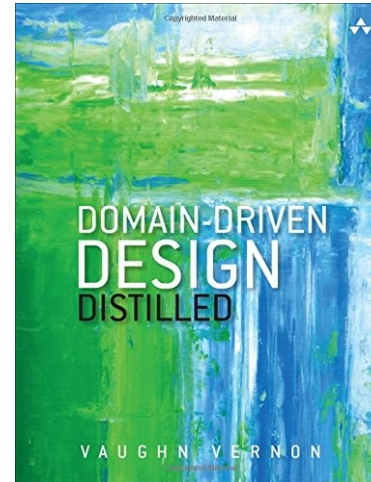
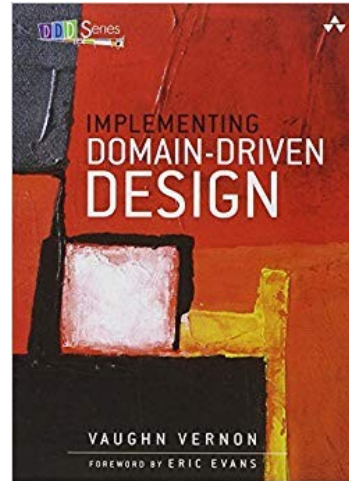
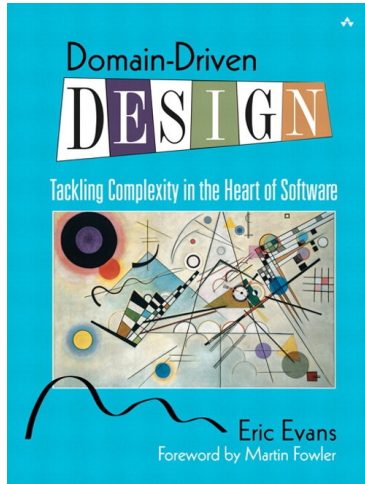
Responsive to change introduced by others



Leading future change introduced by us

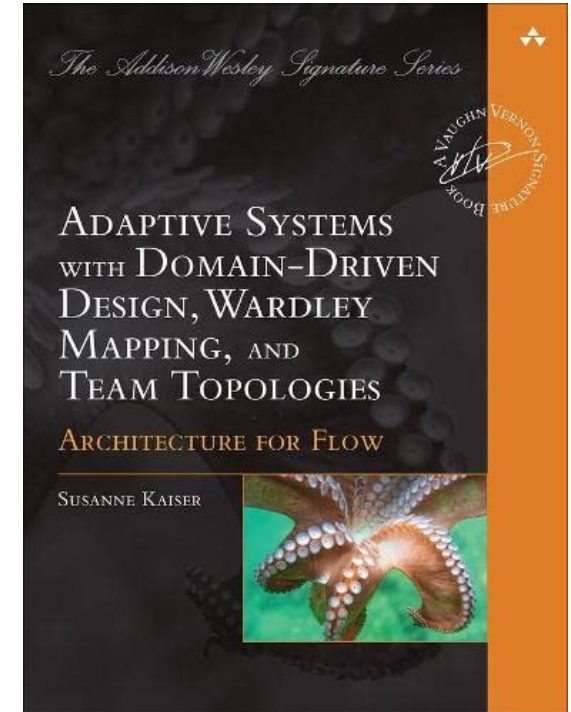
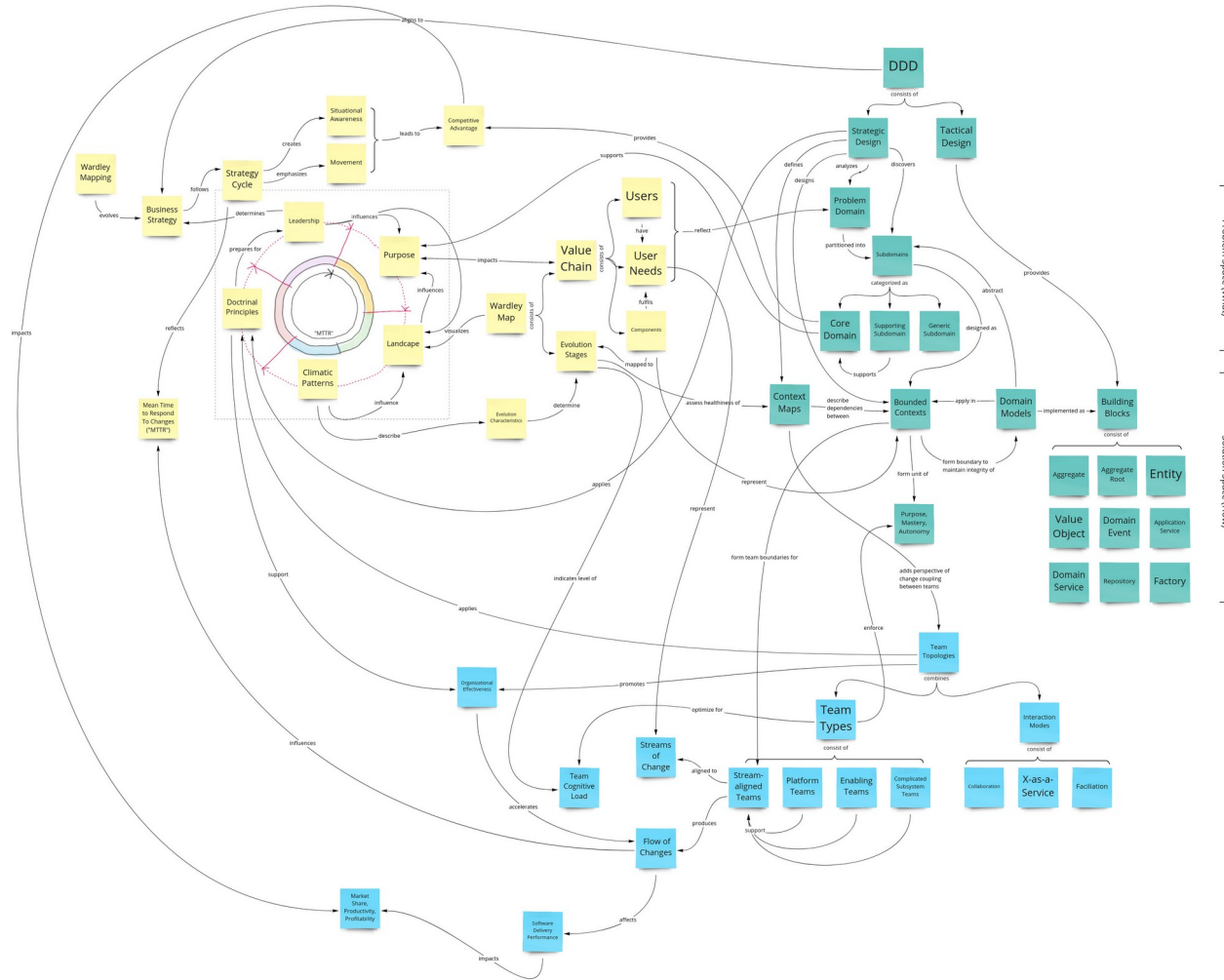


# Some References



- <https://medium.com/wardleymaps>
- <https://learnwardleymapping.com/>
- <https://github.com/wardley-maps-community/awesome-wardley-maps>
- <https://github.com/ddd-crew>
- <https://www.dddheuristics.com>

If you are interested in more details ...



# THANK YOU

Susanne Kaiser

Independent Tech Consultant

Team Topologies Valued Practitioner

@suksr

@suksr@mastodon.social

@suksr.bsky.social

<https://linkedin.com/in/susannekaiser1>

<https://susannekaiser.net>

Link to  
slide deck



<https://susannekaiser.net/talks/>