

Scrum 3.0 —

Same same but different

Boris Glöger

borisglöger consulting

6 levels of influence -

agile management in

large organizations

Scrum 3.0

Org. 4.0

6

Scrum / Kanban

5

Product Development

4

Know How / Skills

3

INFRASTRUCTURE

2

ARCHITECTURE

1

— progress not perfection —

- ▷ Working agile is still in development
- ▷ 6 Levels
- ▷ my own history is connected to this development.
- ▷ We discovered faster ways to deliver! ▷

Organisation 4.0

agile management framework

Product Development

Skills and Knowhow

Infrastruktur

Architecture



Organisation 4.0

agile management framework

Process

Mindset

social invention

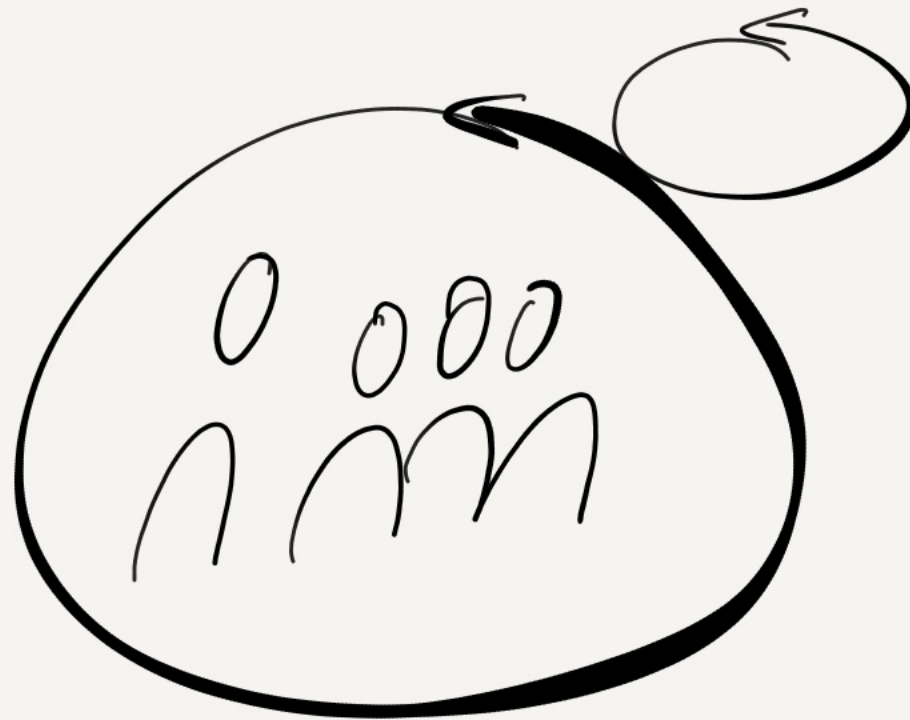
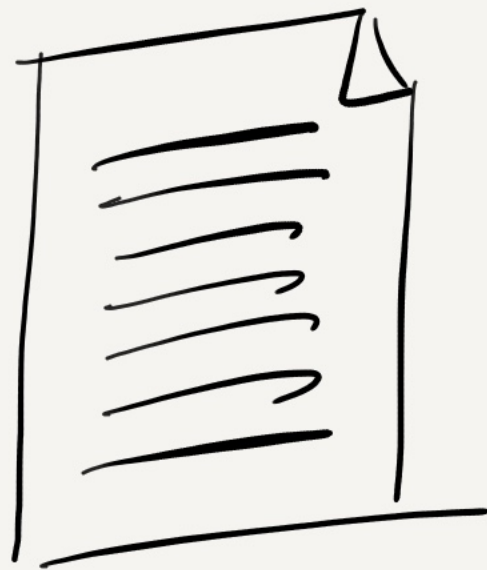
Product Development

Skills and knowhow

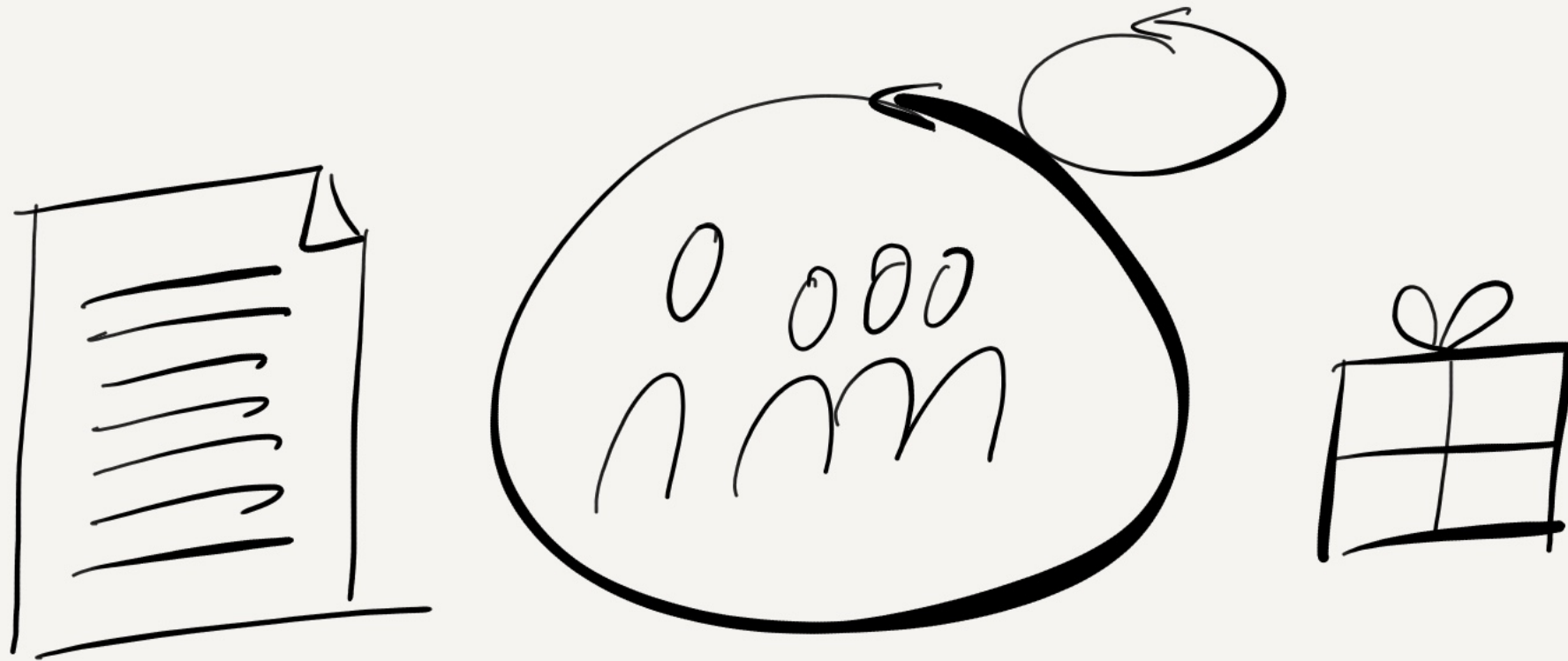
Infrastructure

Architecture

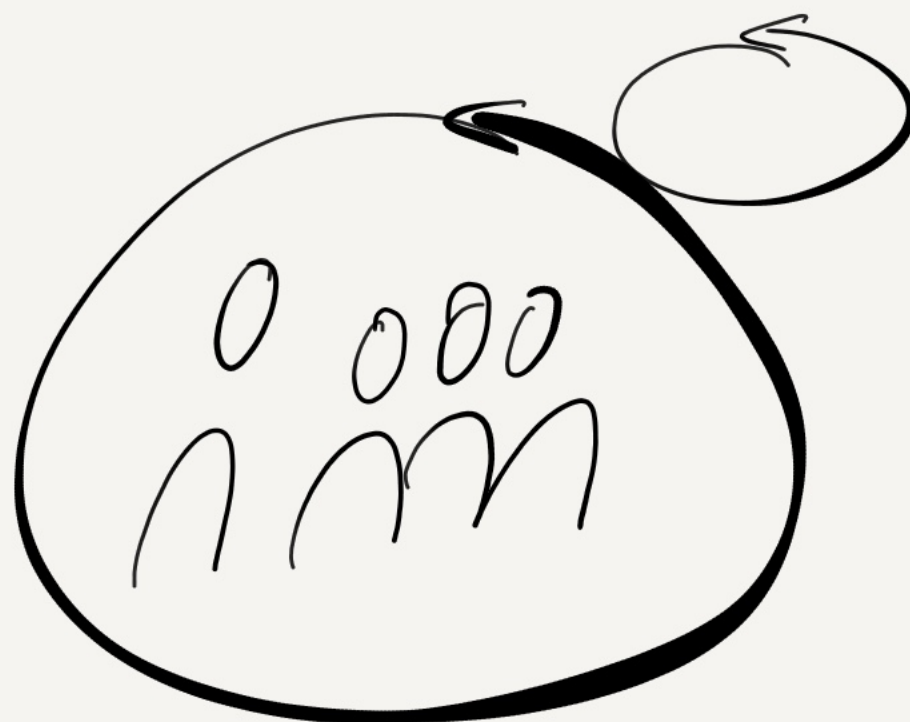
It started with this picture ...



traditional Scrum Thinking does not!



It was designed in the 90th.



Scaling ?

community of practices
disciplined agile

product owner teams
committed backlog

Less

SOS

Premises

SATC

Epics

guilds

tribes

Nexus

Story Mapping

Kanban + scaled Kanban

User Stories

agile can be designed to be simple

- ▷ ask the teams [multi-disciplinary]
- ▷ deliver in short iteration
- ▷ reflect
- ▷ repeat

community of practices
disciplined agile

product owner teams
committed backlog

Less

SOS

Premises

SATC

Epics

guilds

tribes

Nexus

Story Mapping

Kanban + scaled Kanban

User Stories

Which companies are the most
successful? (role model)

▷ Apple	▷ Uber	▷ Bootz.org
▷ Google	▷ Airbnb	▷ Liip
▷ Amazon	⋮	▷ Mayflower
▷ Tesla	⋮	▷

all are build around agile principles! ⚡

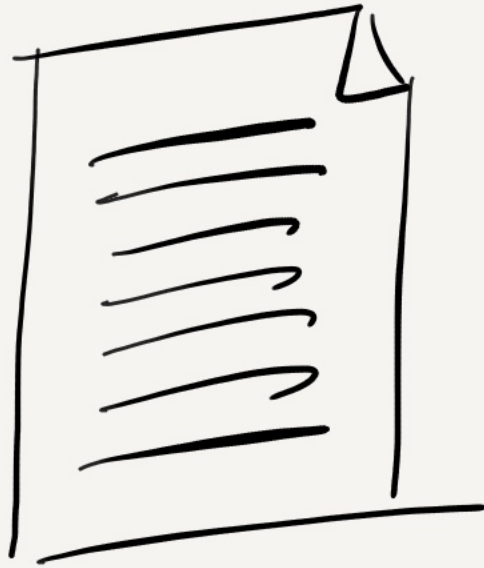
What must we learn from them?

What must we learn from them?

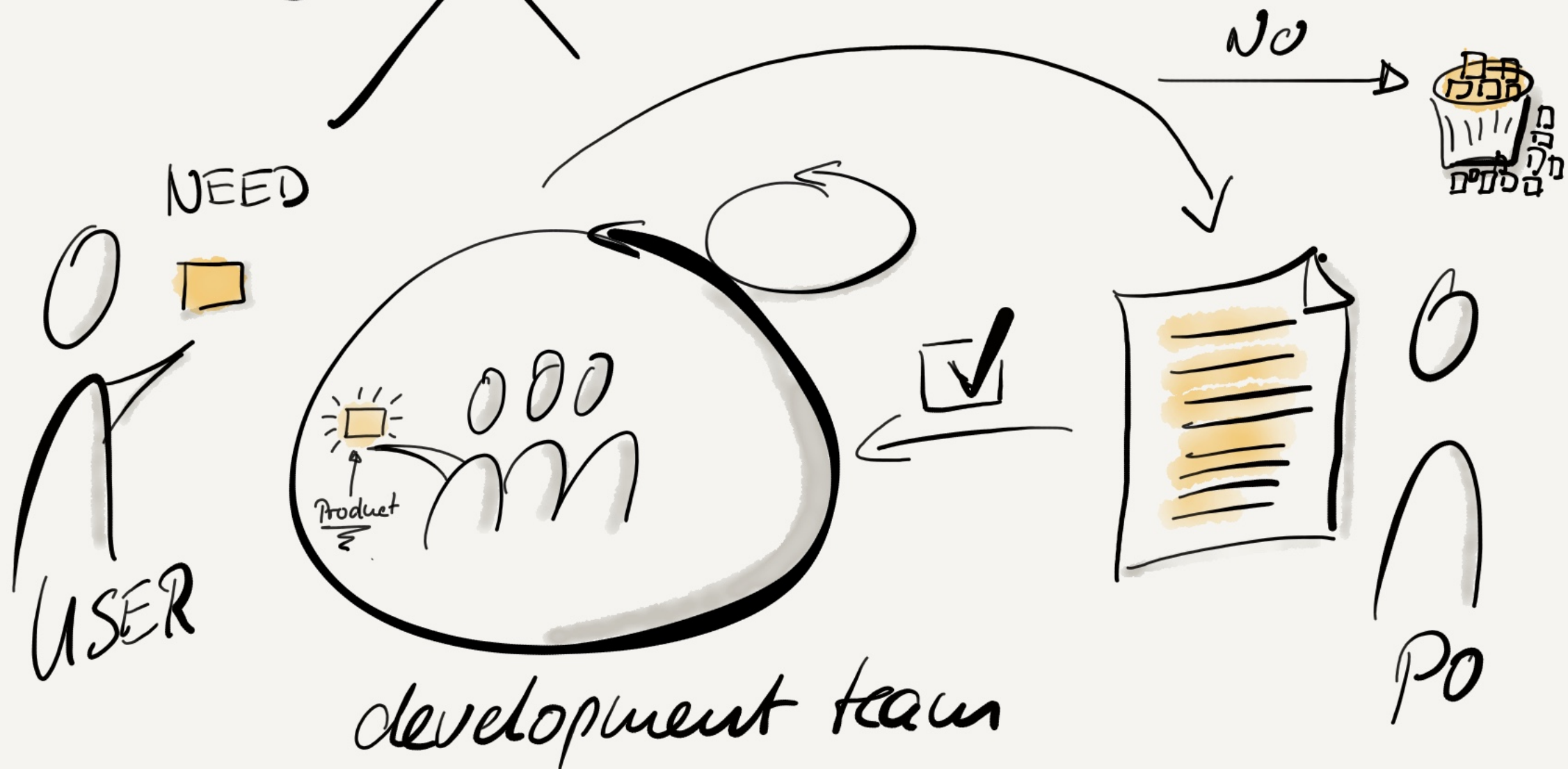
What must we learn from them?

We must solve problem the users have !

~~Customer~~ → User-centric

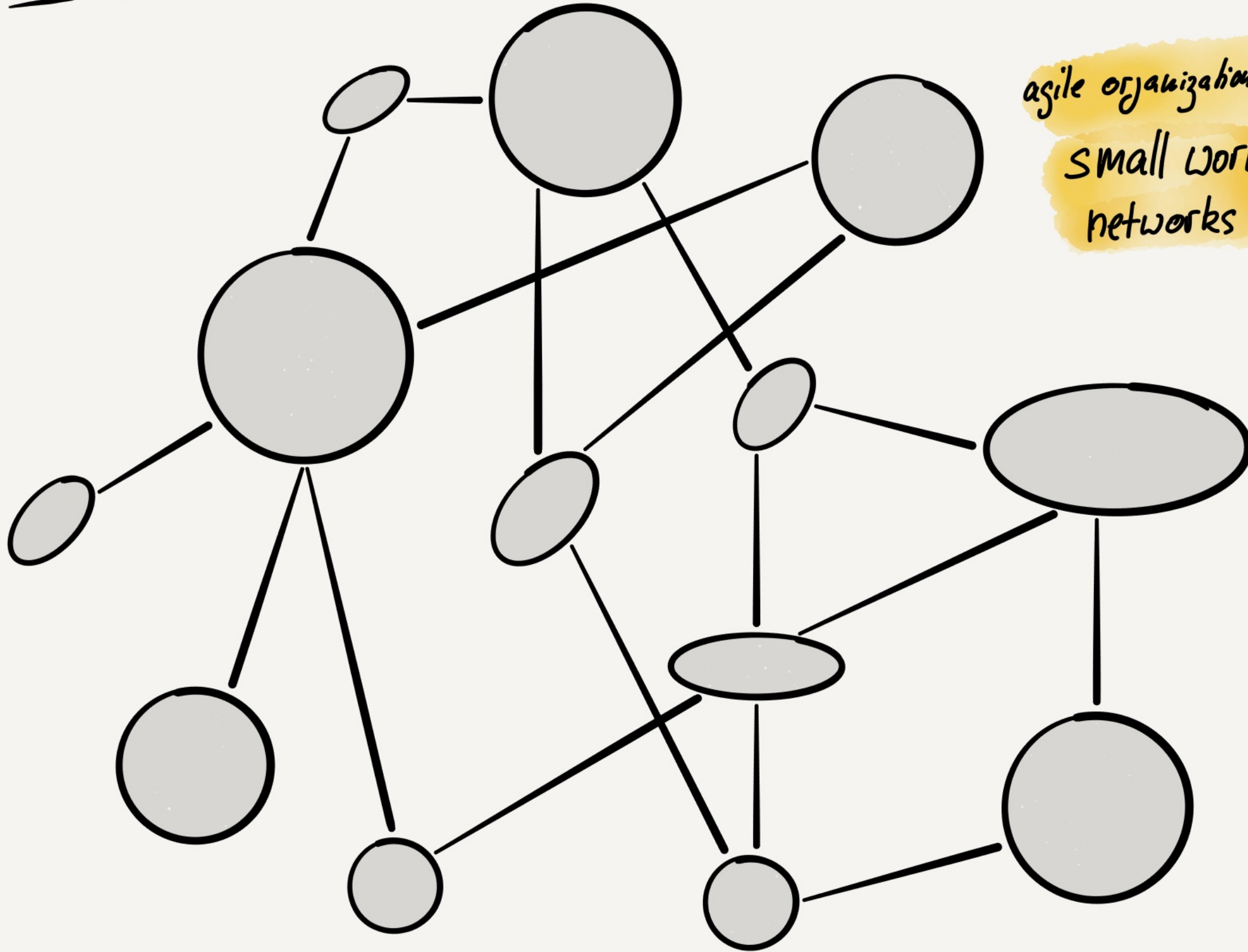


~~Customer~~ → User-centric



1. Architecture

agile organizations architecture
small world
networks

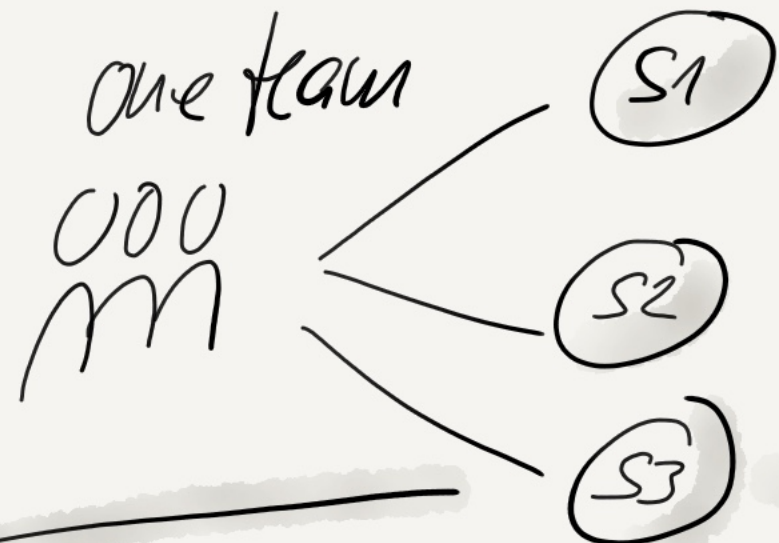


Paradigm - Shift

- ▷ de-coupling
- ▷ polyglot
- ▷ no reuse
- ▷ small
- ▷ Build fast
- ▷ no core-system

one way of
doing this =

Micro - services



② Infrastructure

or no silos using technology?

Communication

- ▷ team-centric
- ▷ access to users
- ▷ rooms
- ▷ internet
- ▷ chat-room

development infrastructure

goal: continuous
deployment

- ▷ more hardware
- ▷ new tools
- ▷ automation

⇒ [dev ops]

3. know how + Skills

- ▷ agile architecture
- ▷ automation
- ▷ pairing, swarming
- ▷ scrum / kanban
- ▷ dev ops
- ▷ TDD

▷ 21 Languages 😊

▷ design patterns

▷ Design Thinking

▷ SOFT-SKILLS

▷ business talk

▷ - - - - -

agile development = social invention

4. Product Development

"Don't ask users what they want!"

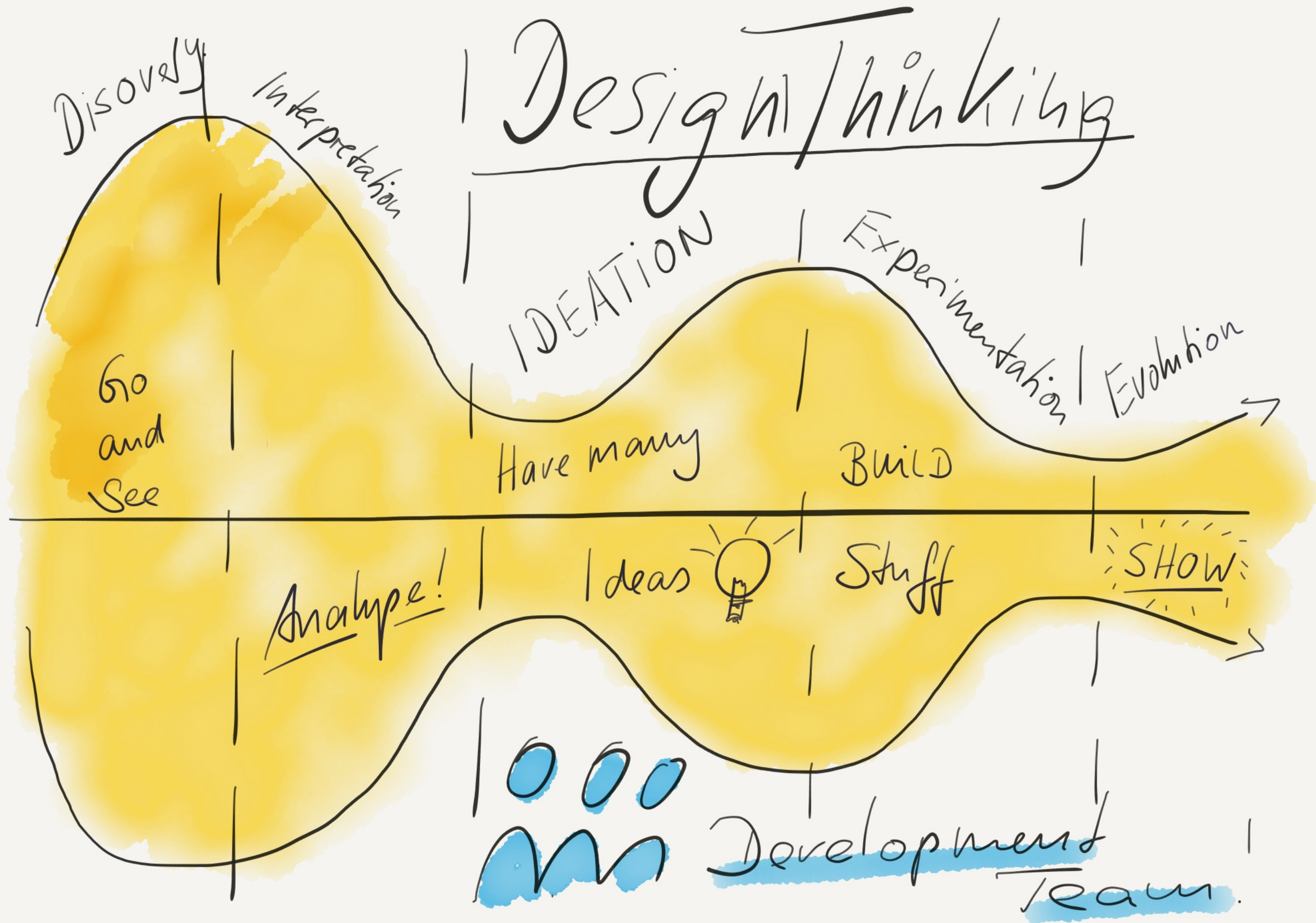
but

Observe them and identify their needs!

Design and Build a solution!

Present!

Repeat!



5. Scrum + TOC = Focus

▷ Backlog size = 1 (or $3 \times$ Velocity to start with)

▷ No estimation but forecasting

▷ dev team writes most User Stories

▷ Product Owner = Vision + Constraints

▷ PO knows technology

▷ Mob-programming

▷ Retrospectives - 1 Question

▷ No meetings! = Working Sessions

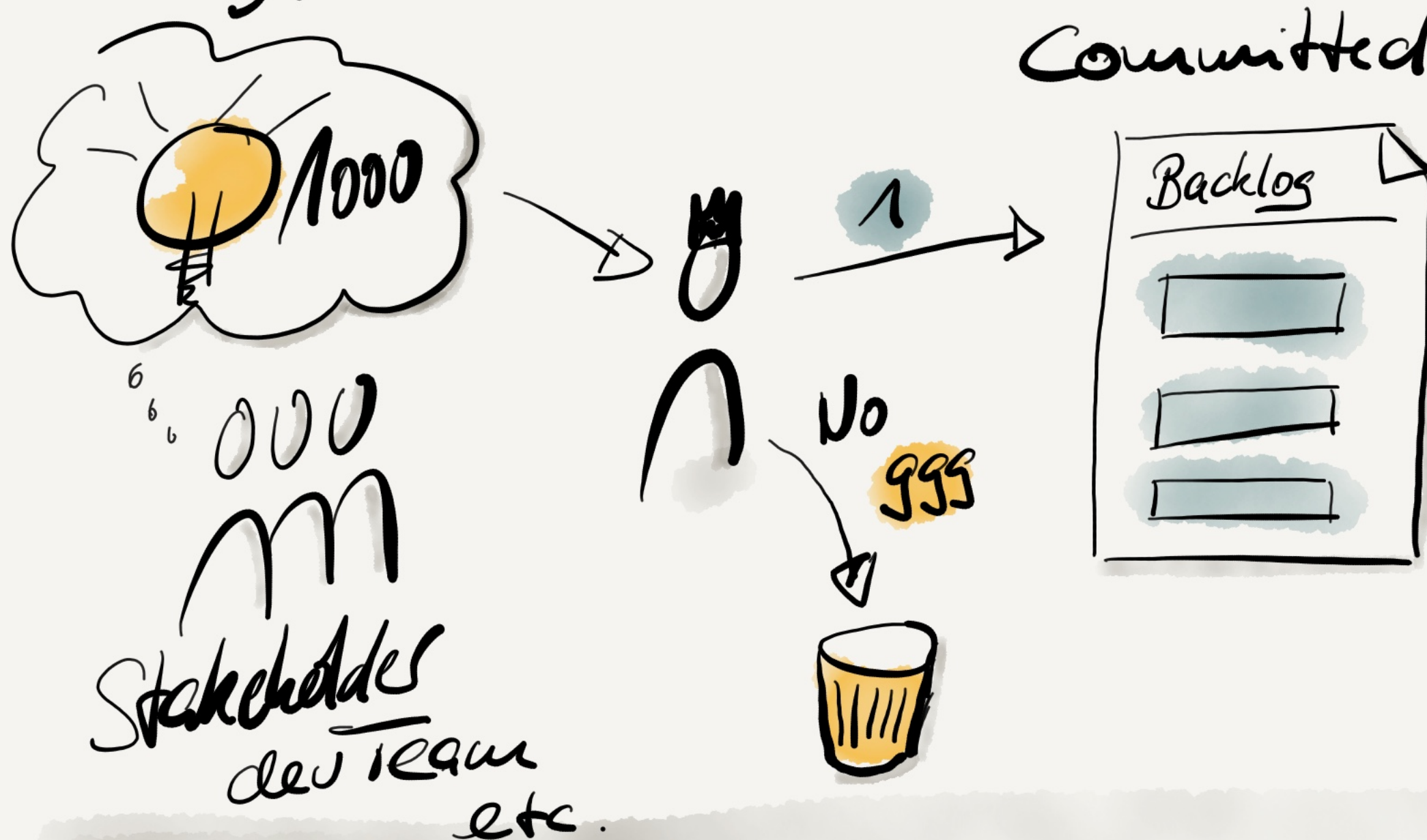
Tip:

1. Work on defects
2. integration (issues)
3. infrastructure
4. new features

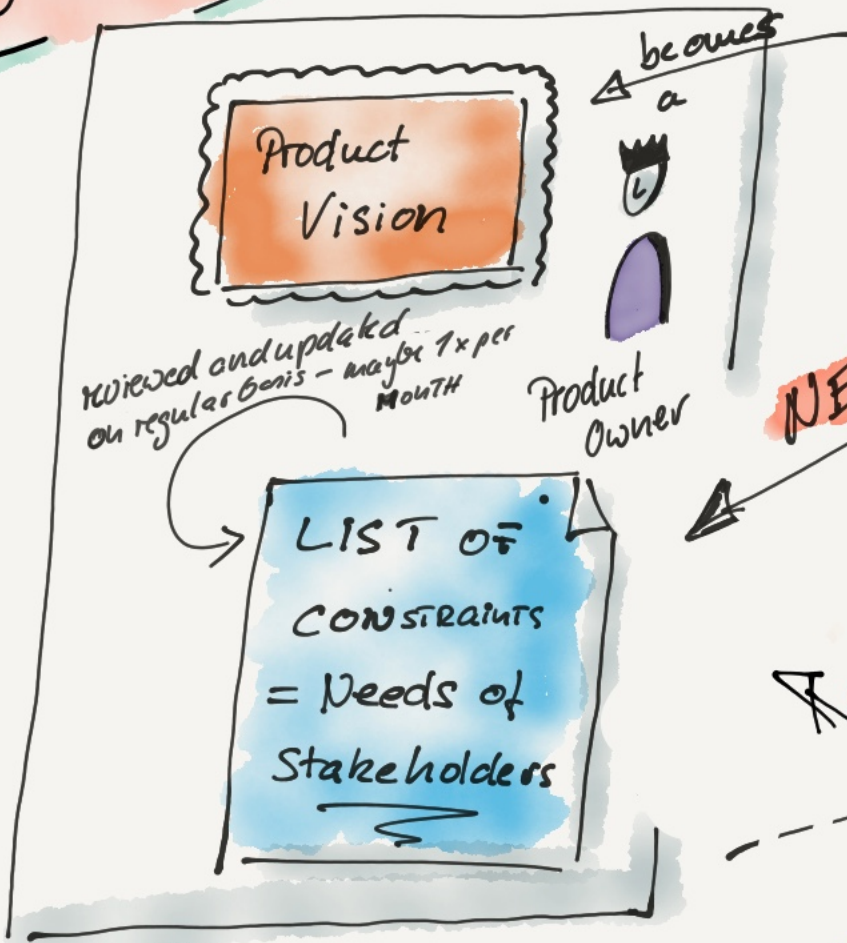
Tell a team: "Nobody will test your product, it will go live as it is!"

Fokus — starts with the Product Owner

Principle No. 1. → keep a small Product Backlog

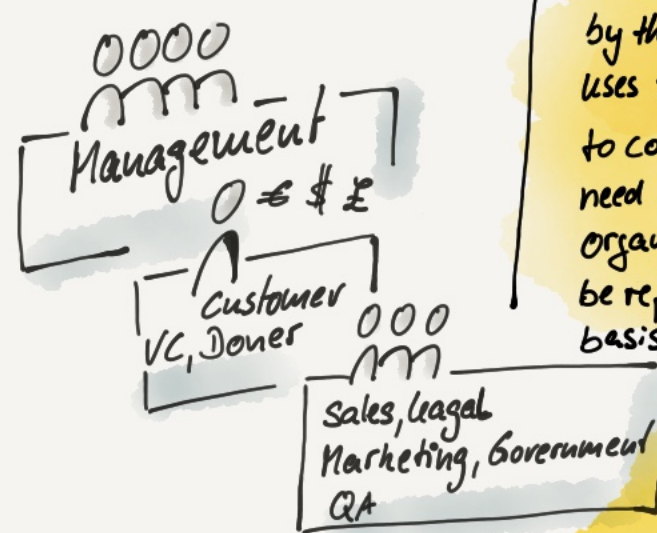


Strategic [Long-term]



Product Idea

Stakeholder



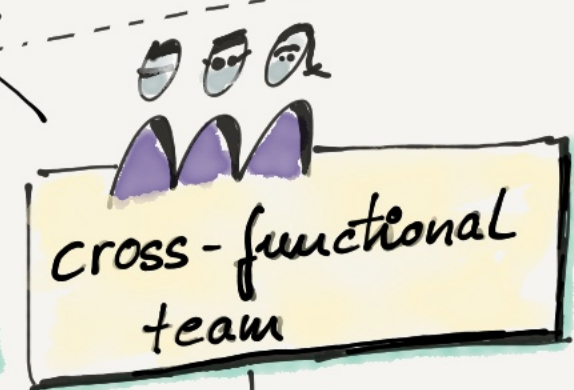
Stakeholder have needs!
These needs will be captured by the Product Owner. He uses the List of Constraints to communicate all these need across the whole organisation. This List will be reflect upon on a regular basis.

Stakeholder is everybody who belongs to the context of the project or organisation.

- ⇒ Customer (external, internal)
- ⇒ other departments
- ⇒ development team
- ⇒ User
- ⇒ Management

NEEDS

NEEDS



WORKS with and for



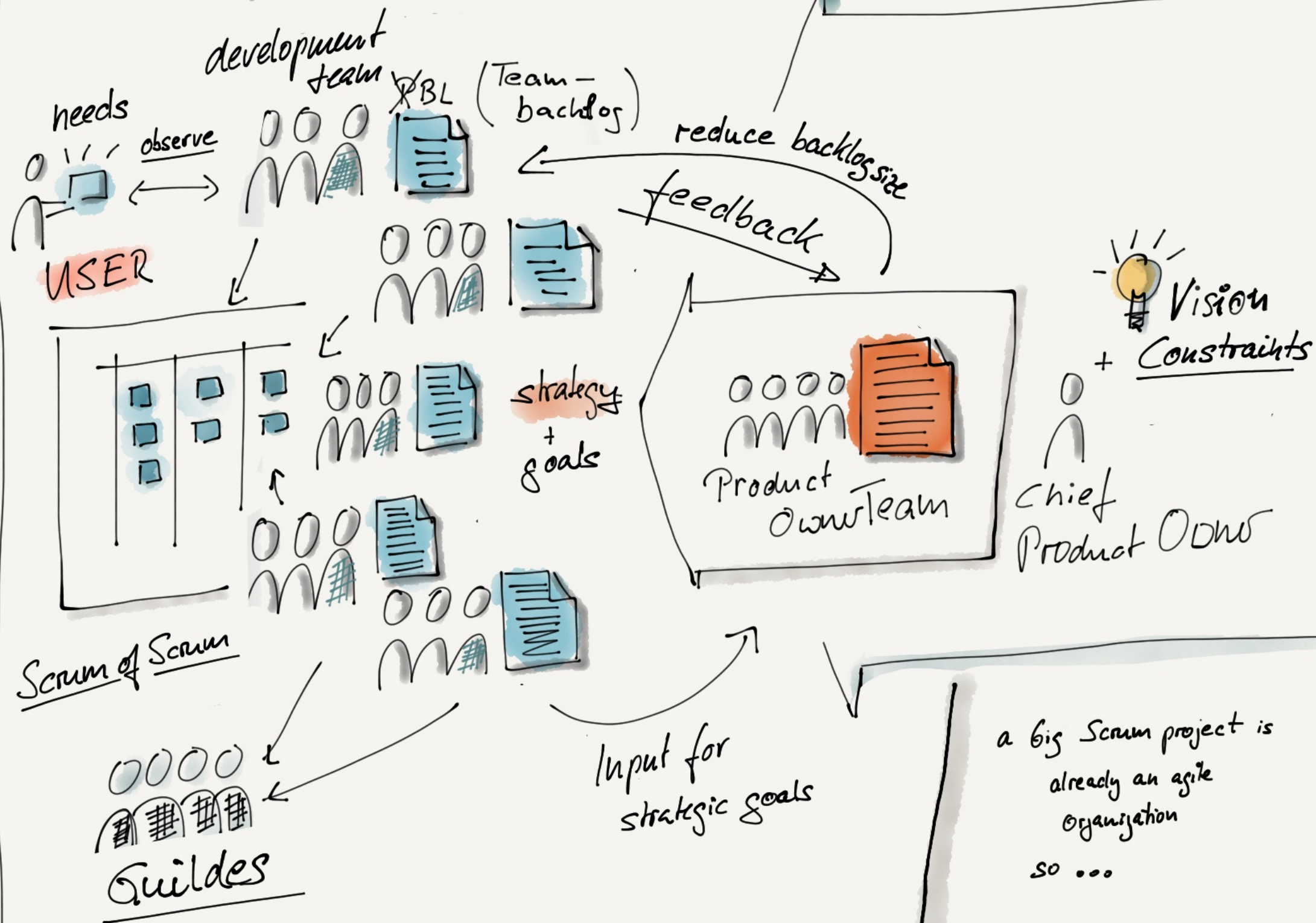
tactical short term



Writing down
f.e. User Stories



large scale scrum - one possible way!



6. Organization 4.0

#1 basic blue print = Open Space Technology -

- ▷ Marketplace
- ▷ Who shows up are the right people
- ▷ Host creates rules
- ▷ Decision to the teams
- ▷ Information is with the team

What does this mean for e.g. Scrum-Teams?

Implications:

- ⇒ Meetings / Events are voluntary!
- ⇒ Teams decides who are on the teams!
- ⇒ Projects can only be done if people show up!
- ⇒ People can move freely to projects if they are invited.
- ⇒ ...

6 levels of influence -

agile management in

large organizations

Scrum 3.0

Org. 4.0

6

Scrum / Kanban

5

Product Development

4

Know How / Skills

3

INFRASTRUCTURE

2

ARCHITECTURE

1