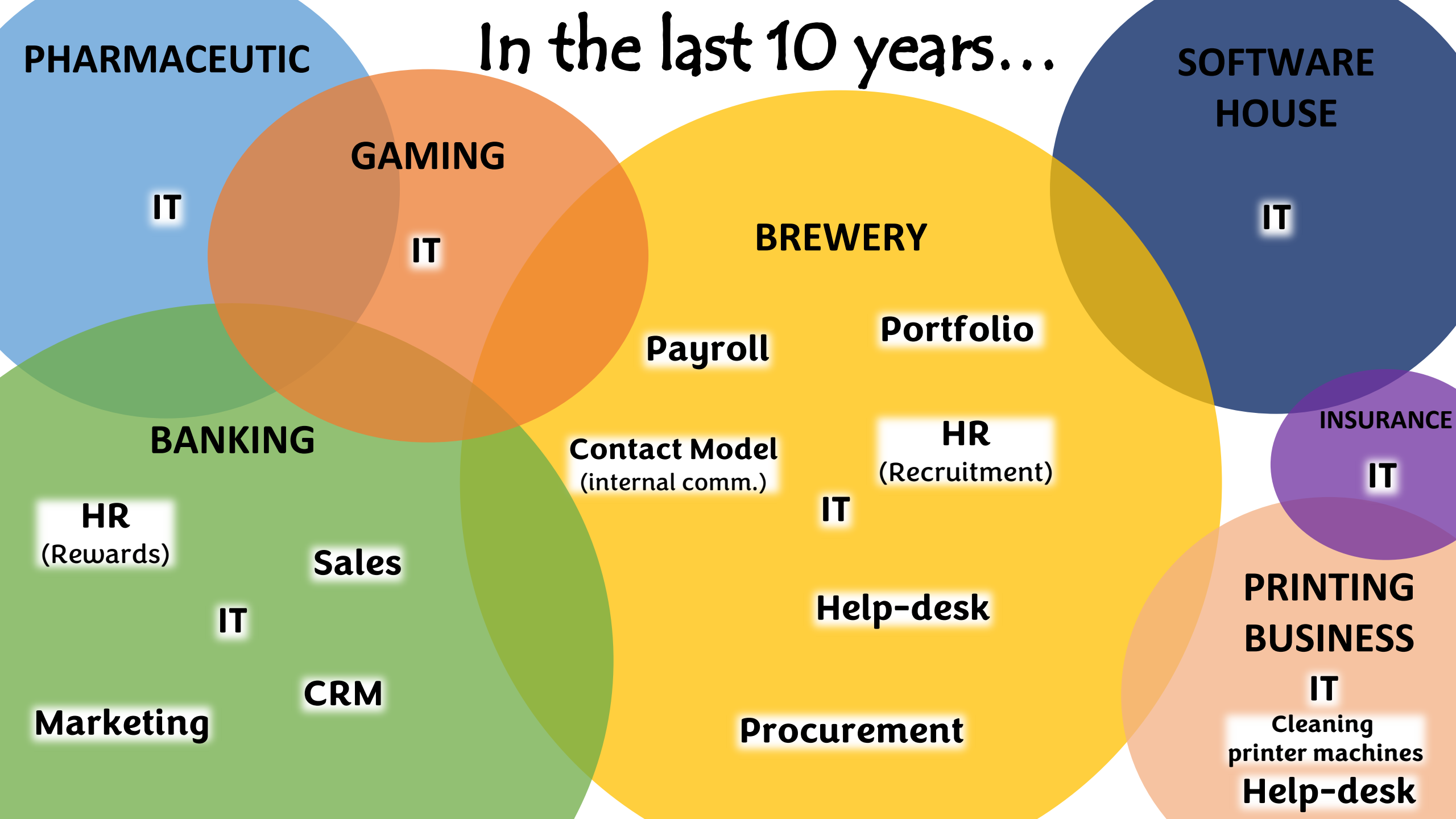
10 Vicious Circles at Work

Detecting and Breaking Them Up

Artur Margonari

In the last 10 years…



VICIOUS CIRCLE

"A sequence of reciprocal cause and effect in which two or more elements intensify and aggravate each other, leading inexorably to a worsening of the situation."

– Oxford Reference

Abbreviations / acronyms

Unknown abbreviations and acronyms appear
(meetings, emails, documents)

Nobody asks so "it can flow"
(or afraid of looking dumb)

More abbreviations and acronyms appear

It gets harder to ask now because I work here for years
(I'll look even dumber!)

Abbreviations / acronyms

Unknown abbreviations and acronyms appear (meetings, emails, documents)

Nobody asks so "it can flow" (or afraid of looking dumb)

More abbreviations and acronyms appear

It gets harder to ask now because I work here for years (I'll look even dumber!)

Abbreviations and acronyms

Consider:

- **“What does it mean?”** (even if you know it)
- Awareness: **Psychological Safety**
- **Not using** it yourself (be clear, not clever)
- Creating a **glossary**

2 Scrum Master superhero

VARIATIONS:

- Team member
- Boss

The Scrum Master superhero solves an impediment by him/herself

The team doesn't learn how to solve it

The same (or similar) impediment happens again

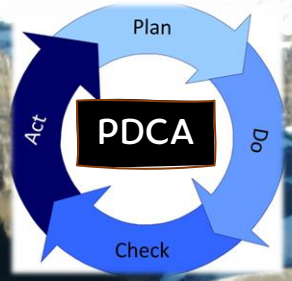
Everyone looks at the Scrum Master superhero to solve it

Superheros

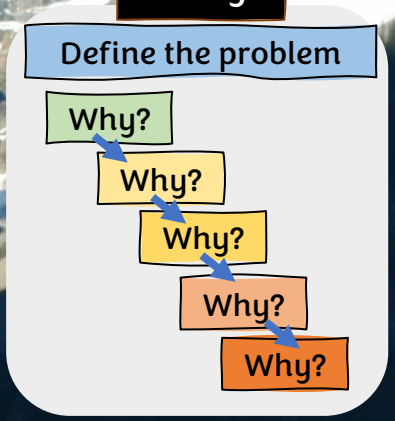
(SM, team member, boss)

Consider:

- Finding the **root causes**
- Creating a **Skills Matrix**
- Creating a **Wiki**
- Try **not** to be the **superhero**



5 Whys



Ishikawa / Fishbone

A3 problem solving

Skills Matrix

		Today	Future	Today	Future	Today	Future	Today	Future	Today	Future	Today	Future	AVG	Action plan
EXPERTISE	Analysis	3	4	2	4	3	4	3	4	4	4	4	4	2.6	Coding Dojo: same as Lambda but for Orch
	Lambda	4	4	1	2	1	2	1	3	2	2	2	2	1.5	
	NodeJS	4	4	3	4	2	3	3	4	2	2	2	2	2.3	
	React	4	4	4	5	3	4	3	4	1	1	2	2	2.5	
	Database	3	3	2	2	2	3	2	3	4	4	5	5	2.5	
	Branching/merging	4	4	4	4	4	4	4	4	1	3	5	5	3.0	
	PR request	4	4	3	4	3	3	4	4	2	3	5	5	2.9	
	Testing	1	1	1	1	2	2	3	3	4	4	5	5	2.8	
	Testing automation	1	1	2	4	1	1	1	1	2	2	4	5	1.7	Schedule a session with small group to exp
	Deployment	1	1	2	2	1	1	1	1	2	3	5	5	1.7	
	Demo to stakeholders	1	1	1	1	1	1	1	1	5	5	4	4	2.2	
	Workflow automation	2	3	2	3	1	1	1	1	2	2	3	4	1.9	
	Facilitation	1	1	1	3	1	1	1	1	3	3	3	3	1.9	
	Agile	1	1	3	4	2	2	2	2	3	3	4	4	2.6	
PRODUCT AREA	Finance	5	5	2	4	3	4	1	2	5	5	3	4	2.6	
	Time tracking	3	4	2	4	3	4	3	4	5	5	3	4	2.9	
	Action Spreadsheet	2	3	2	4	2	3	2	3	5	5	3	4	2.6	
	Review System	1	1	1	3	1	2	1	2	5	5	3	4	2.4	
	Resource mgmt	1	1	1	3	1	1	1	2	5	5	3	4	2.1	
	Project Mgmt	3	3	4	5	2	2	4	4	5	5	3	4	3.0	
	Skills mgmt	3	3	1	3	1	1	1	2	5	5	3	4	2.4	
	Leave mgmt	2	3	1	3	1	1	4	4	5	5	3	4	2.6	
	Scheduling	1	2	1	3	4	4	1	2	3	5	3	4	2.2	

Taskboard x meetings

Task board is messy and outdated

People don't keep it updated

The board is not used in events

Events are less valuable and board is not updated

Taskboard x meetings

Consider:

- Finding the **root causes**
- Keeping it **simple / easy** to use
- Agreeing on **constraints**
- **Experimenting**
- It's **their board**, not yours!

4 High team turnover

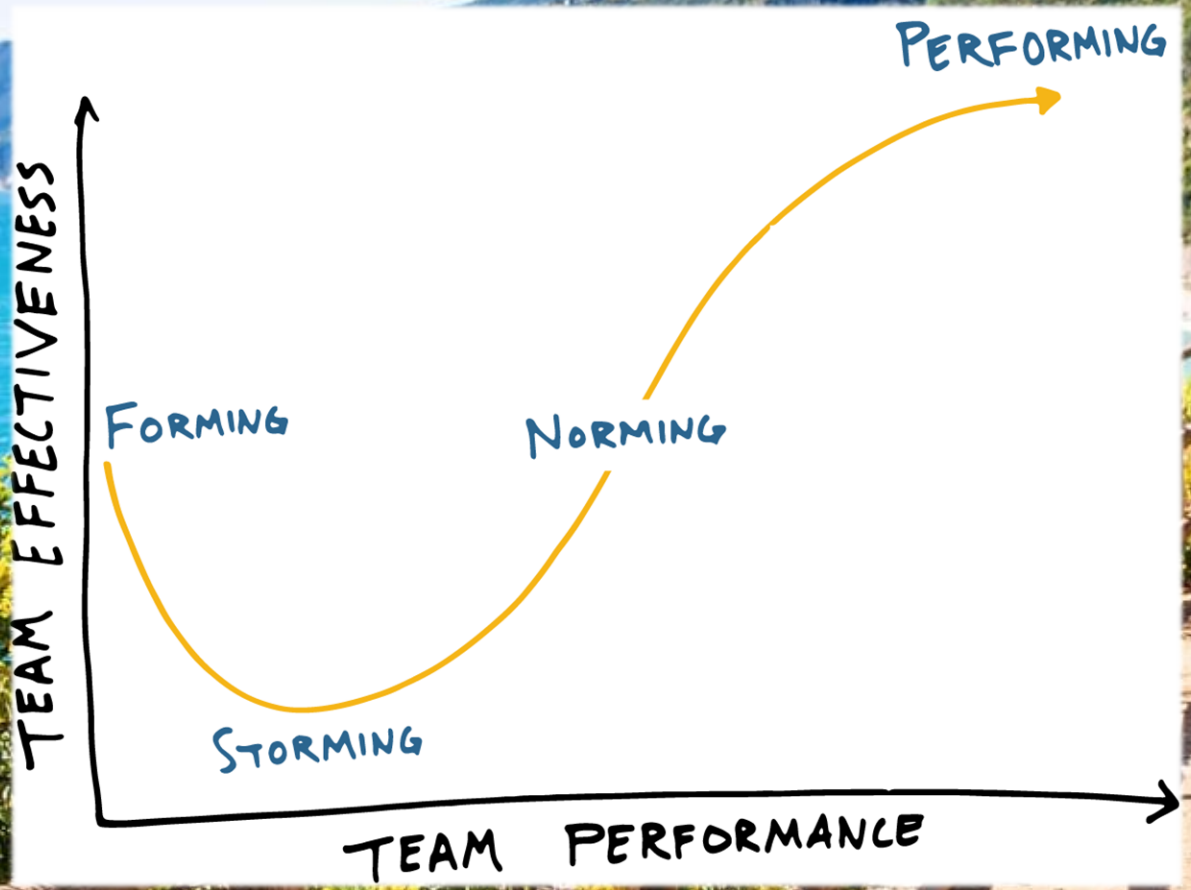
"Let's hire 2 more analysts so we can go faster"

Onboarding is needed

Productivity slows down

"They are not moving fast enough"

High team turnover



Consider:

- Finding the **root causes** (yep!)
- Working on **team engagement/NPS**
- Visualizing the **impact of high turn over**
- Creating and using **skills matrix**
- Forgetting about the **“1 month handover”**

Technical debt

Technical debt is created

Increase of complexity
(spaghetti code)

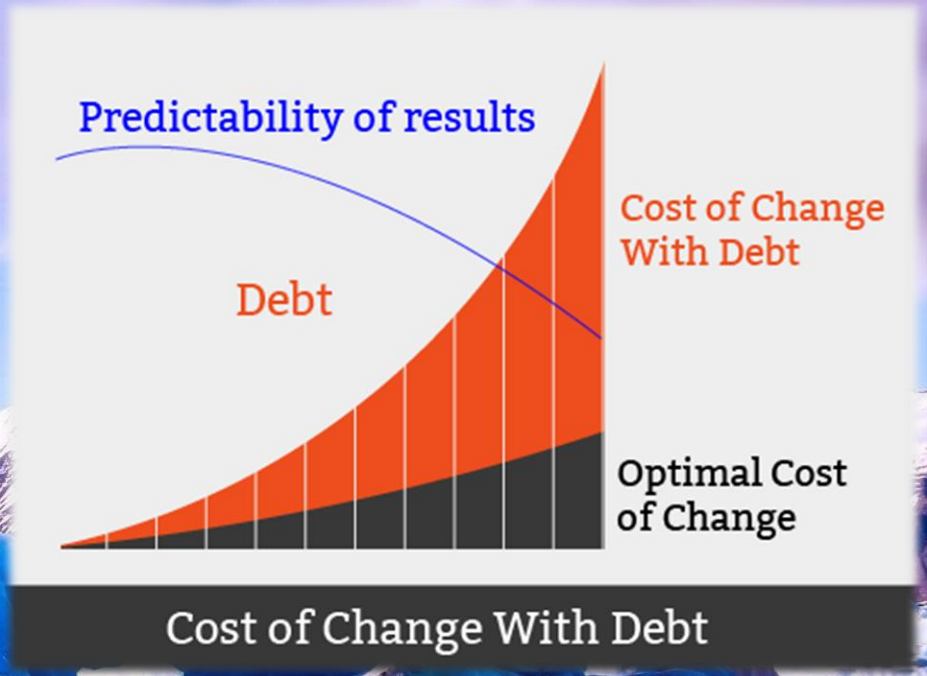
Harder to refactor,
predict/forecast

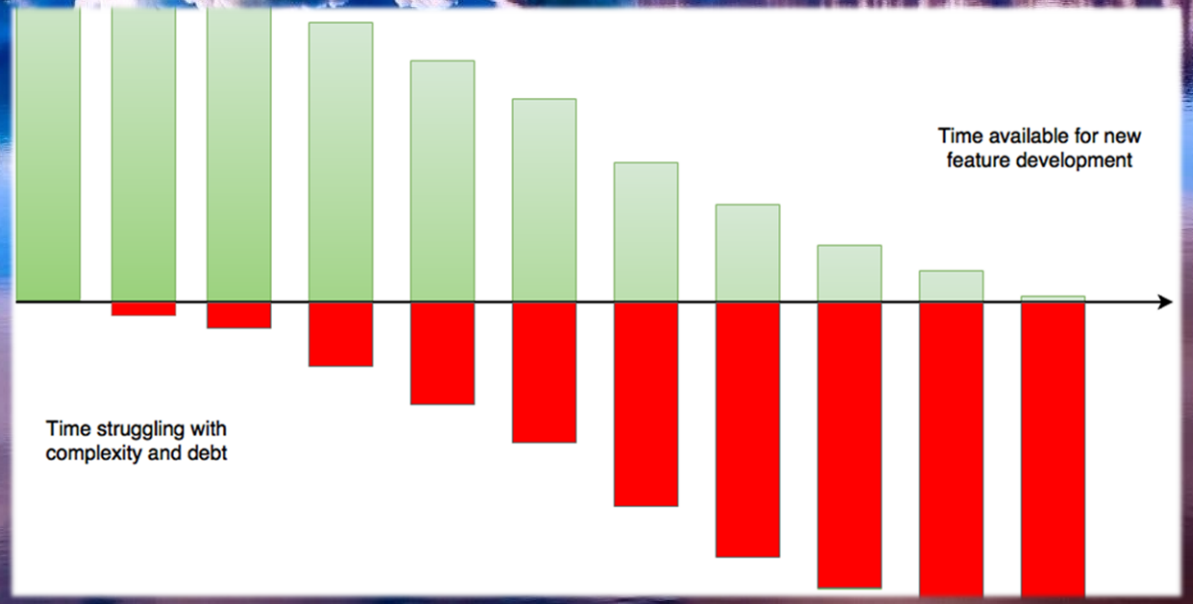
"Let's move faster!"
(speed vs quality)

Technical debt

Consider:

- Finding the **root causes** :)
- Visualizing the **impact of technical debt**
- Dedicating X% of the Sprint for **refactoring**
- Creating **coding conventions**
- Automating **tests** and **processes**
- **Scouts rule**





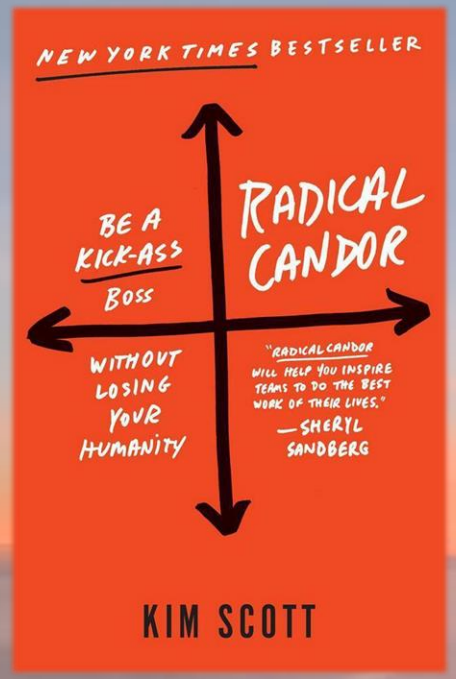
6 Conflict-resolution debt

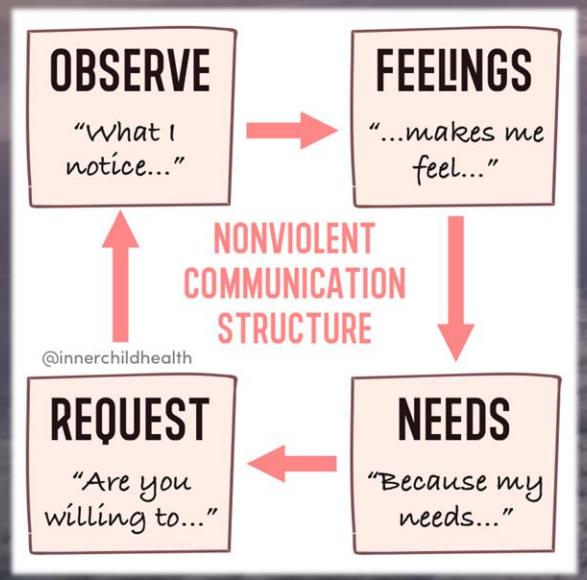
A conflict happens in the team

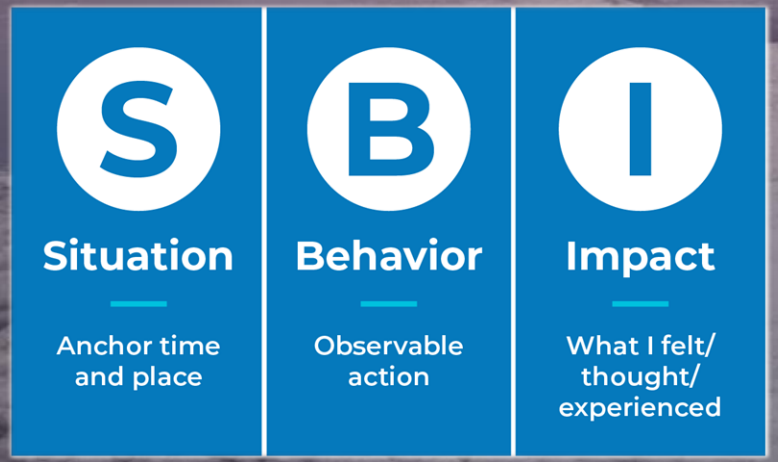
Too busy/don't know how to solve conflicts

The unsolved conflicts generate more conflicts

Less energy to spent on solving them







Conflict-resolution debt

Consider: (I pay you a beer if you guess the 1st step)

- Finding the **root causes**! :D
- Introducing:
  - **Radical Candor** (Kim Scott)
  - **The Responsibility Process** (Chris Avery)
  - **Non-Violent Communication** (Marshal B.)
  - **Impact feedback** (fact + how you felt)
- **Starting small**
- Creating/updating **team agreements**

7 No WIP limit

Work In Progress

Work is pushed top-down onto the team "It's high priority"

Team tries to get 50+ things done at the same time

Context switching, no focus and waste

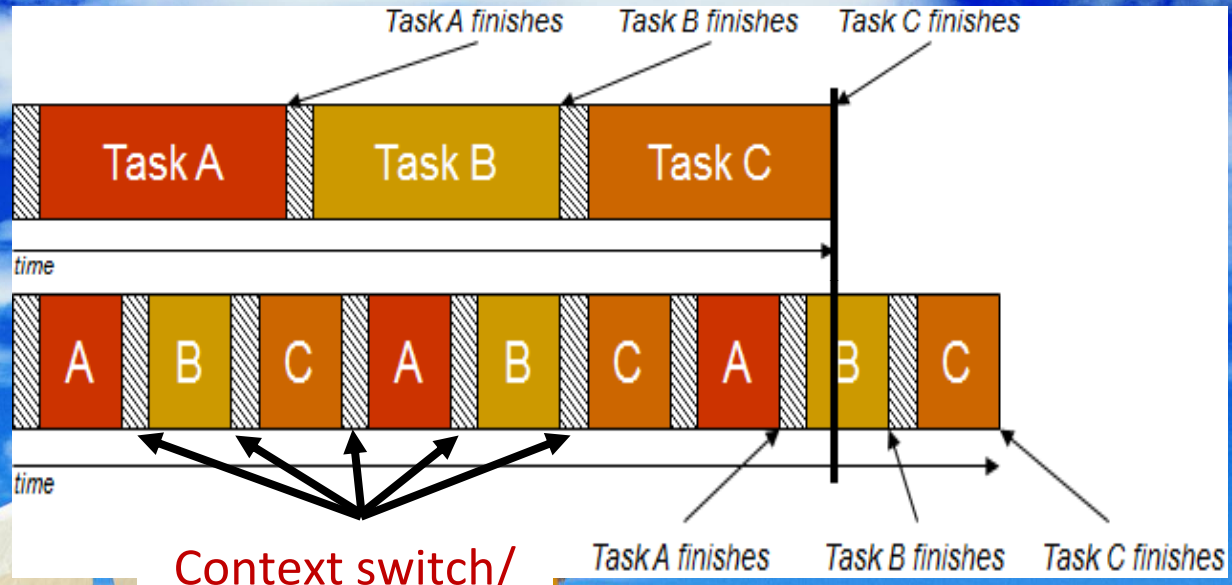
"They are not going fast enough…"

No WIP limit

Consider:

- Finding **root causes** (surprise!)
- Visualizing the **impact of high WIP**
- Aligning with **PO / leadership / stakeholders**
- New motto: **“Finish more, Start less”**

  Or “Stop starting, Start finishing”



Context switch/ waste

8 No time for improvement

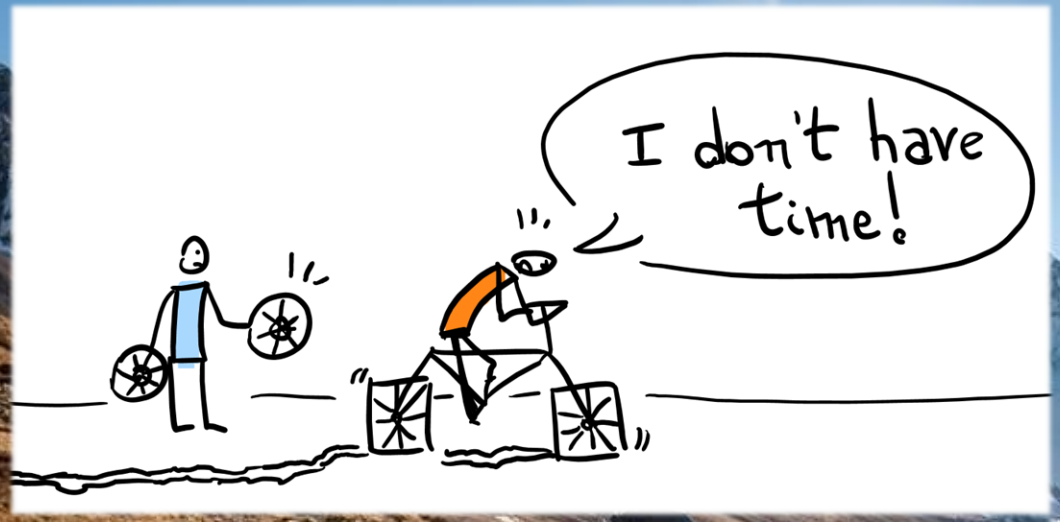
"A 1-day training? Too long.
Can you do it in 1 hour?"

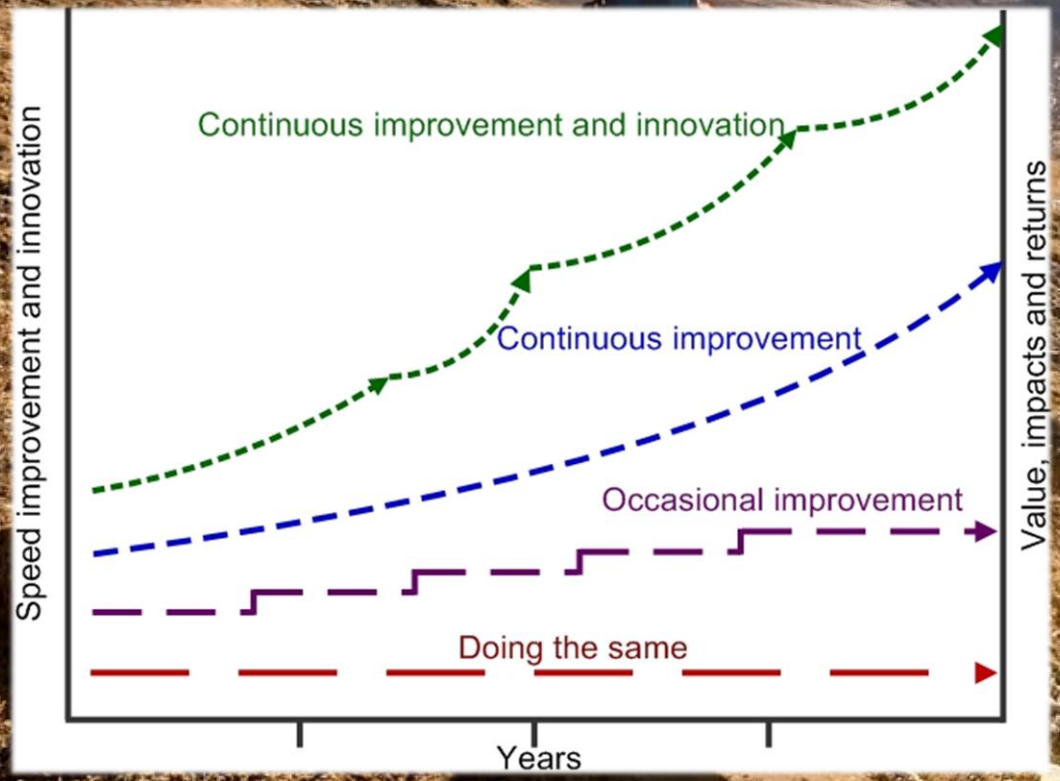
Lots of unnecessary conflicts
due to misalignment

Many other meetings take
place to fix these conflicts

"We have too many meetings.
We are too busy!"

No time for improvement





Consider:

- Finding the **root causes** (surprise!)
- Visualizing: **with** and **without** improvement
- Getting **leadership support** (dedicated time)
- Trying **small + success cases** (numbers!)
- Don't let the risk of bankruptcy to be the reason to improve/innovate.

9 Toxic behaviours

☢️ Company doesn't deal with toxic behaviours

"But he/she gets things done, you know?"

🏃 Great employees start leaving...

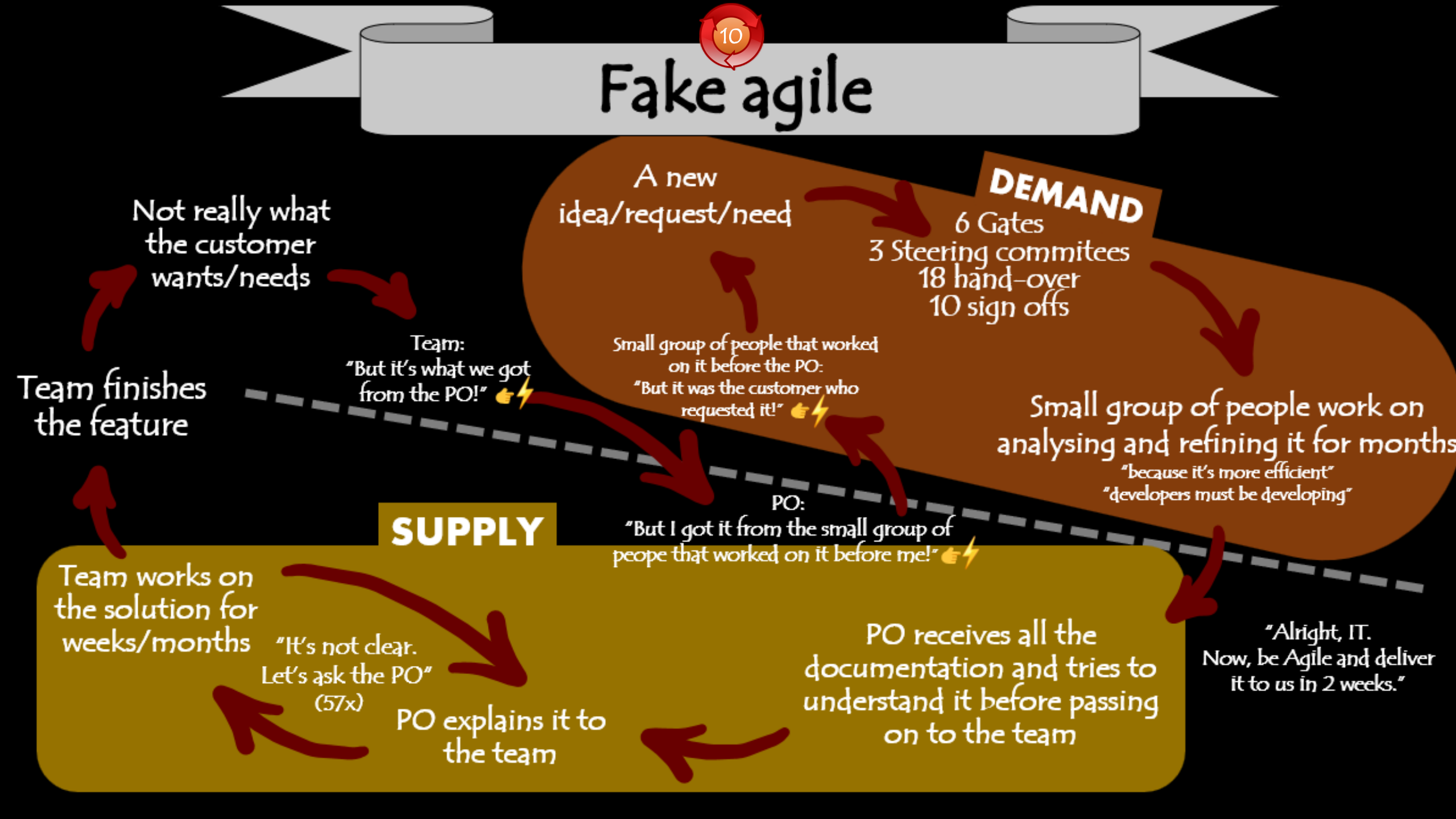
🧑‍💼 Employees w/ toxic behaviours reach leadership roles

🤝 These new leaders hire people with "similar mindset"

Toxic behaviours

Consider:

- **Separating the behavior** from the person
- Roooooooooot causes! **Listen to understand**
- Possible reasons:
  - Personal issues
  - WHY of Agile not clear
  - HR not involved (bonus/career path)
- Adjusting **Team Agreements** (Big NO GO)
- Having **hard conversations**
- Moving a person? **Make reasons** very **clear**!


10
Fake agile
DEMAND
A new idea/request/need
6 Gates
3 Steering commitees
18 hand-over
10 sign offs
Small group of people work on analysing and refining it for months
"because it's more efficient"
"developers must be developing"
"Alright, IT. Now, be Agile and deliver it to us in 2 weeks."
PO receives all the documentation and tries to understand it before passing on to the team
PO explains it to the team
"It's not clear. Let's ask the PO" (57x)
Team works on the solution for weeks/months
SUPPLY
Team finishes the feature
Not really what the customer wants/needs
Team: "But it's what we got from the PO!"
PO: "But I got it from the small group of peope that worked on it before me!"
Small group of people that worked on it before the PO: "But it was the customer who requested it!"

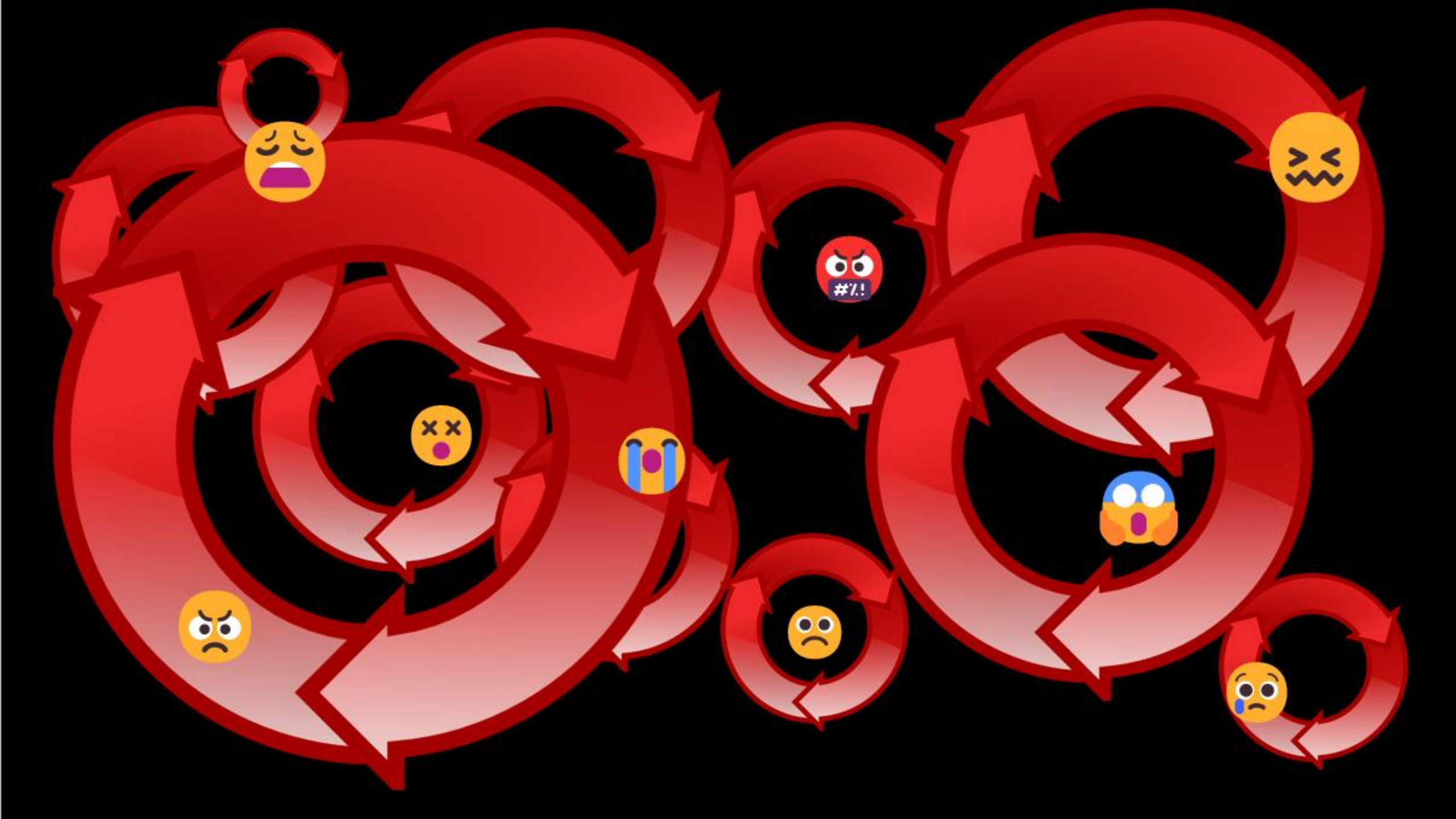
Fake Agile

Consider:

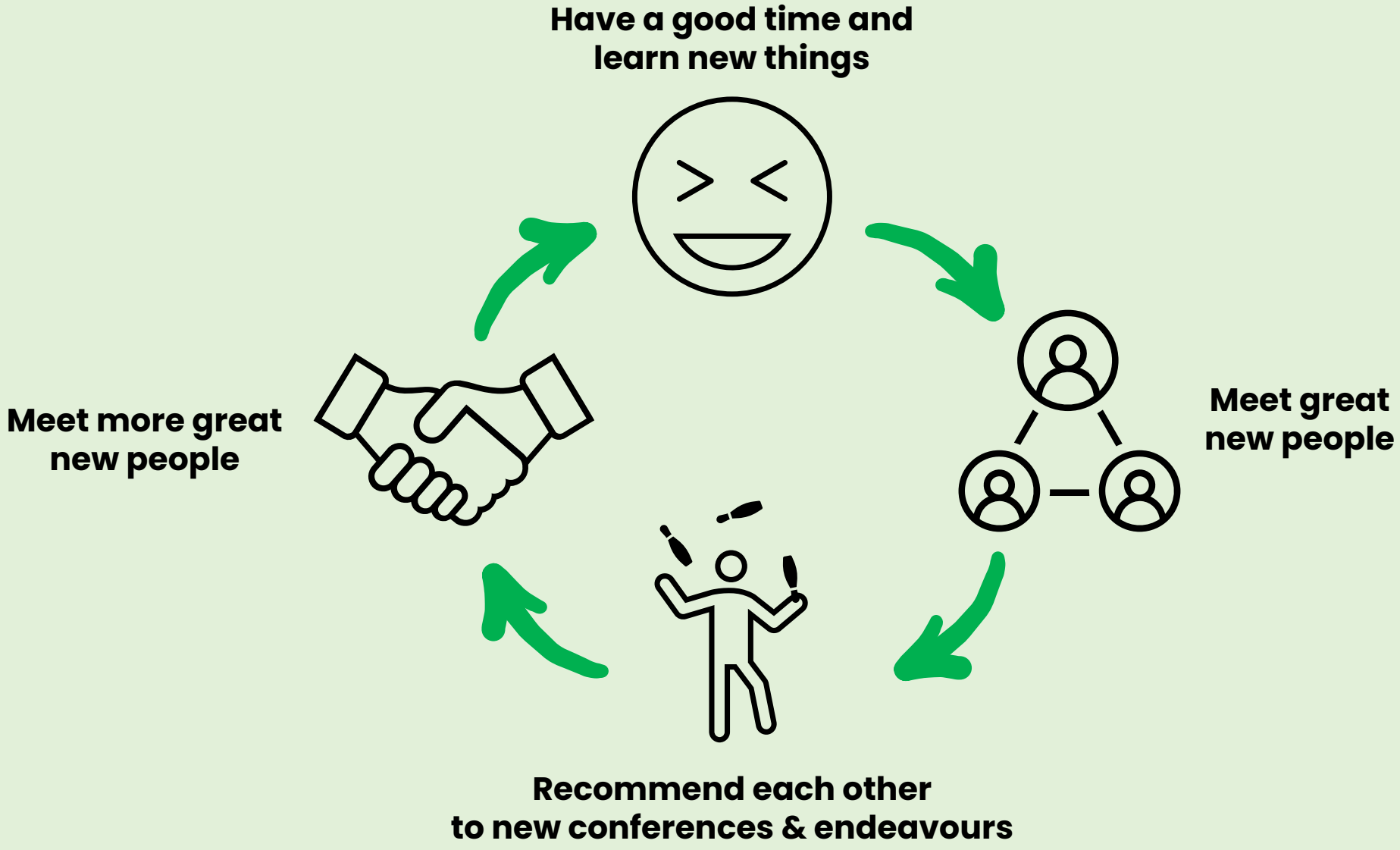
- Find the **root causes**
- Involving **business/leadership/HR**
- Translating it into “**pocket-pain**”
- **Value Stream Mapping**
- **User Story Mapping**
- Creating **success cases**

VICIOUS CIRCLES

Abbreviations/acronyms
Superhero (x3)
Taskboard x meetings
High team turnover
Technical debt
Conflict-resolution debt
No WIP limit
No time for improvement
Toxic behaviours
Fake Agile



Now, enjoy a positive self-reinforcing loop! :)



Thanks for having me!

Questions?

Let's connect on in !

ROTI + selfie ☺