Growth
from
Knowledge

# More than meets the eye

## The advanced techniques to become fully-rounded agilists and truly conquer projects!

G. Luisi
Agile Coach
www.gfk.com

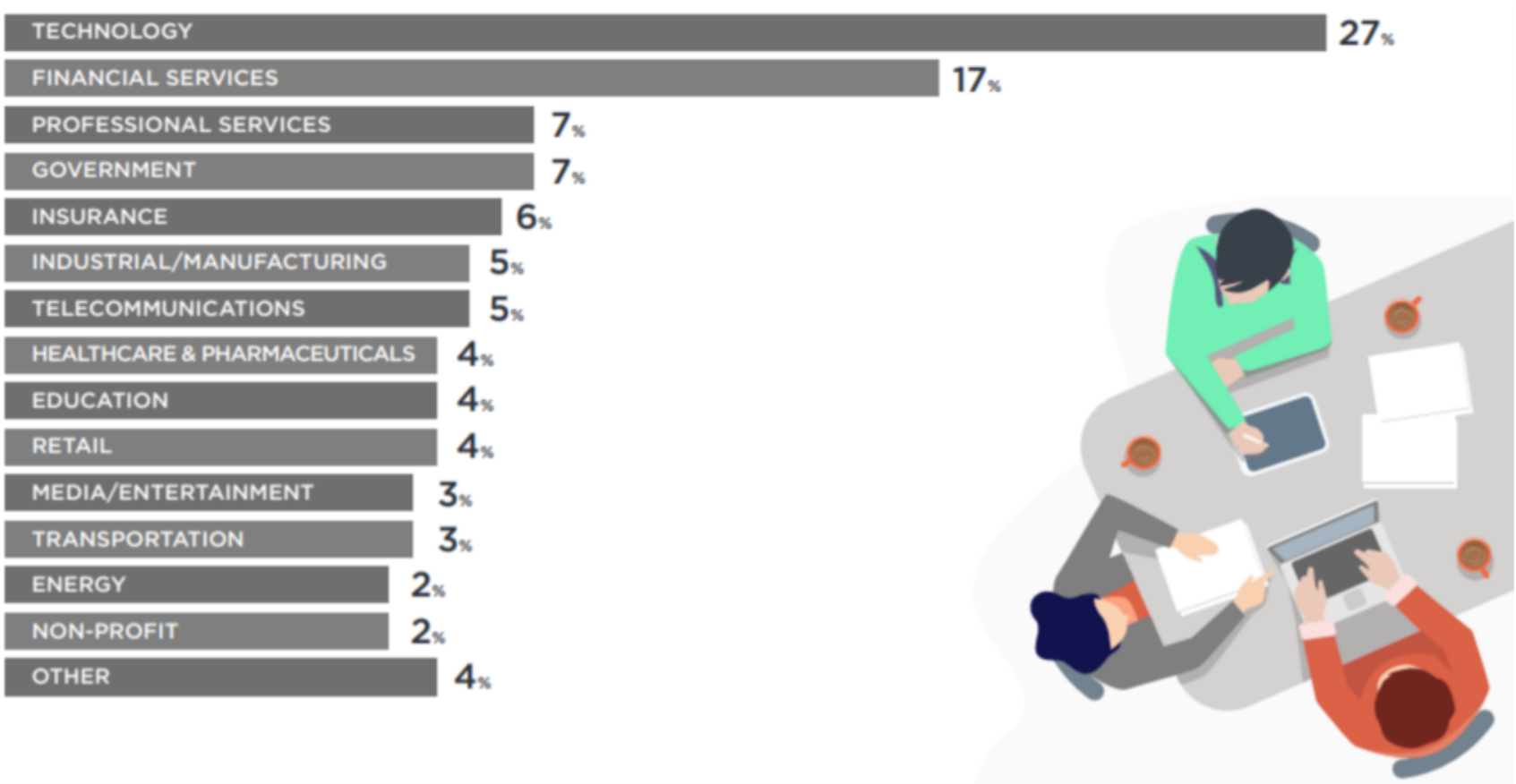
# It's hard to establish how widespread agile methods are, but these are some indicative statistics:

- 80% of major USA federal IT projects described themselves as "Agile" already in 2017
- only 27% of the respondents to the 2020 State of Agile Survey comes from the technology sector

https://www2.deloitte.com/us/en/insights/industry/public-sector/agile-in-government-by-the-numbers.html/#endnote-sup-5
14th State of Agile Report, 2020, https://stateofagile.com/

# In fact, Agile is being adopted far and wide

## You WILL encounter it beyond technology

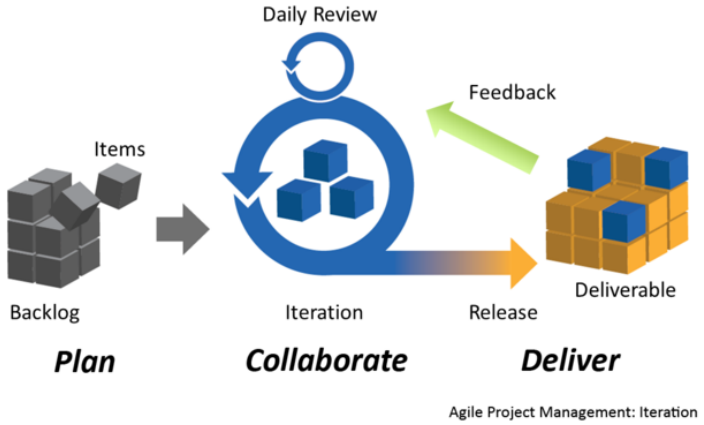



14th State of Agile Report, 2020, https://stateofagile.com/

# The outcome is that likely you will work in an agile environment

## You will have to learn and practice:

## Agile Methods



Agile Project Management: Iteration


SAFe® PROVIDED BY SCALED AGILE

Fluid Scaling Technology

Disciplined Agile

NEXUS Scrum.org

LeSS

Enterprise Scrum Business Agility for a better FUTURE

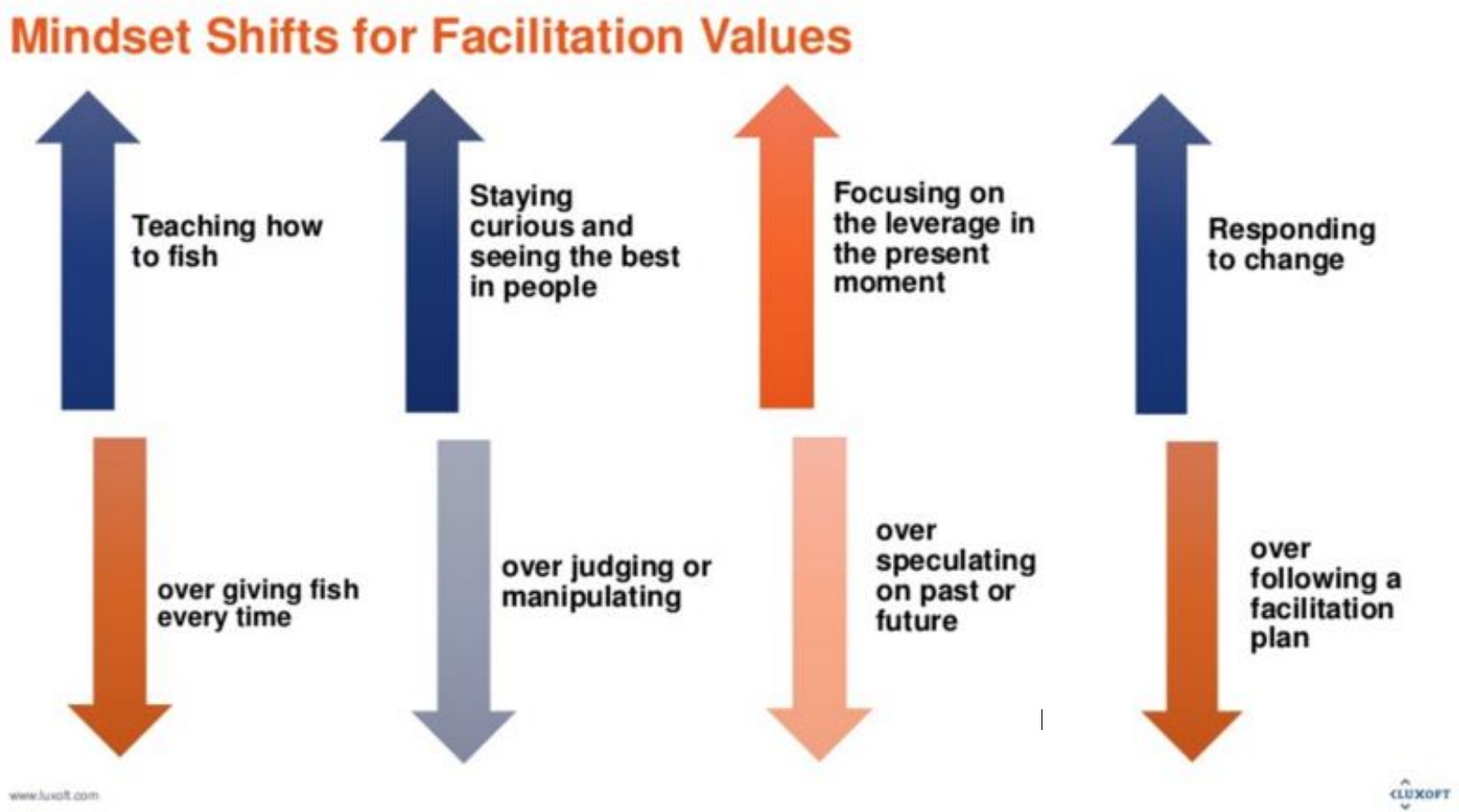
Spotify®

SCRUM @SCALE

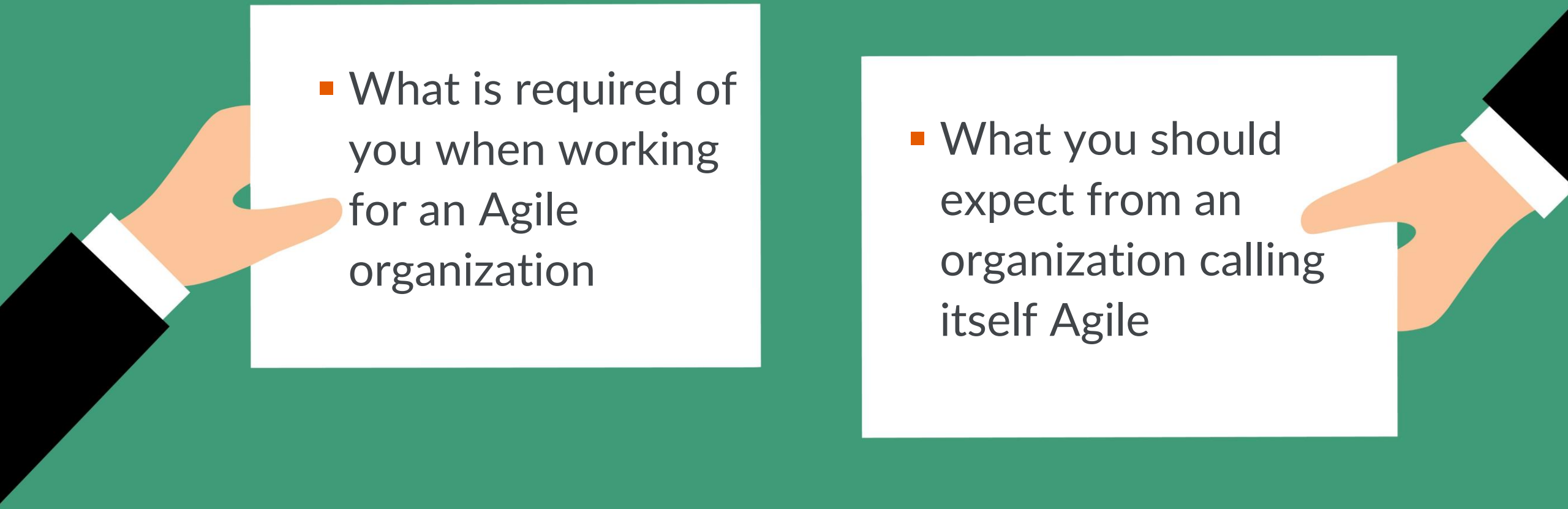
# The outcome is that likely you will work in an agile environment

## You will have to learn and practice:

## The Agile Mindset



# The outcome is that likely you will work in an agile environment



- What is required of you when working for an Agile organization

- What you should expect from an organization calling itself Agile

# Is this still a proper measure of success?

M.B. Haga, MFA

# Within the Agile mindset, the **team** rules

We value

**INDIVIDUALS AND INTERACTIONS**

OVER

**PROCESSES AND TOOLS**

The Agile Manifesto

Within a Scrum Team, there are **no** sub-teams or **hierarchies**.

***Commitment, Focus, Openness, Respect, and Courage***

The Scrum Team commits to achieving its goals and to **supporting each other**.

The Scrum Guide

In other words,

*Surrender the "**ME**" for the "**WE**".*




Jahn Henne


# Crossroads: Success or Failure with a yes/no outcome



www.ccPixs.com

GfK Growth from Knowledge

# We are bound to fail: let's embrace it!

According to a 2017 report from the Project Management Institute (PMI), **14 percent of IT projects fail**.

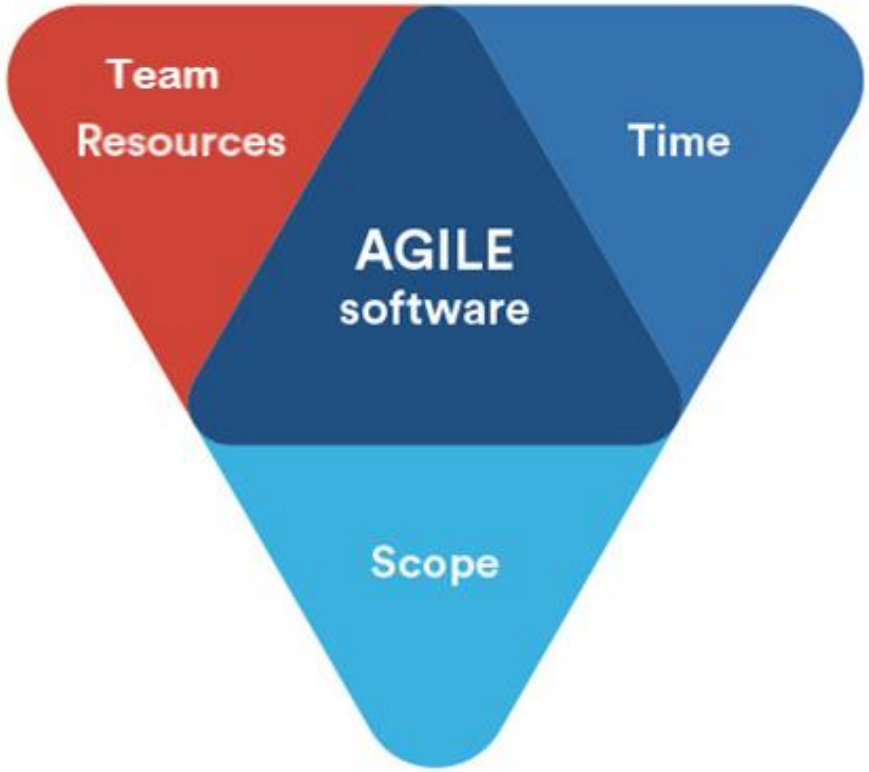
However, that number only represents the total failures. Of the **projects** that didn't **fail** outright,

- **31 percent** didn't meet their goals
- **43 percent** exceeded their initial budgets,
- and **49 percent** were late.

# Commitment to deliver has three dimensions

## Agile recognize you cannot fix all three of them!

**All factors affect COST**




Flexibility is key!
Teams and projects cannot commit to a fixed delivery


Is there anything to gain from failure?
Any reason why we should accept it?
LEARN

# An Agile organization is a learning organization

## And failures are learning opportunities

- Agile recognize **our environment is non static**, and that **learning is the goal** of the Agile Mindset.

- Learning organizations facilitate the learning of its members, who continuously transform themselves. These organizations, which promote a **continuous learning culture**, are capable to survive and thrive in the midst of **rapid change and high complexity**, which is considered **the new default state**.

Learning is linked to the promise from teams and team members of relentless improvement

# Moving beyond the retrospective

## Introducing two tools for relentless improvement

**Retrospectives** are the core of the relentless improvement process, but they are usually not enough to identify (anti)patterns within the work of the team, or to promote evolving teamwork improvements when they take time to implement. To do better we can use:

- Team Metrics
- The Team Improvement Backlog

# Team Metrics

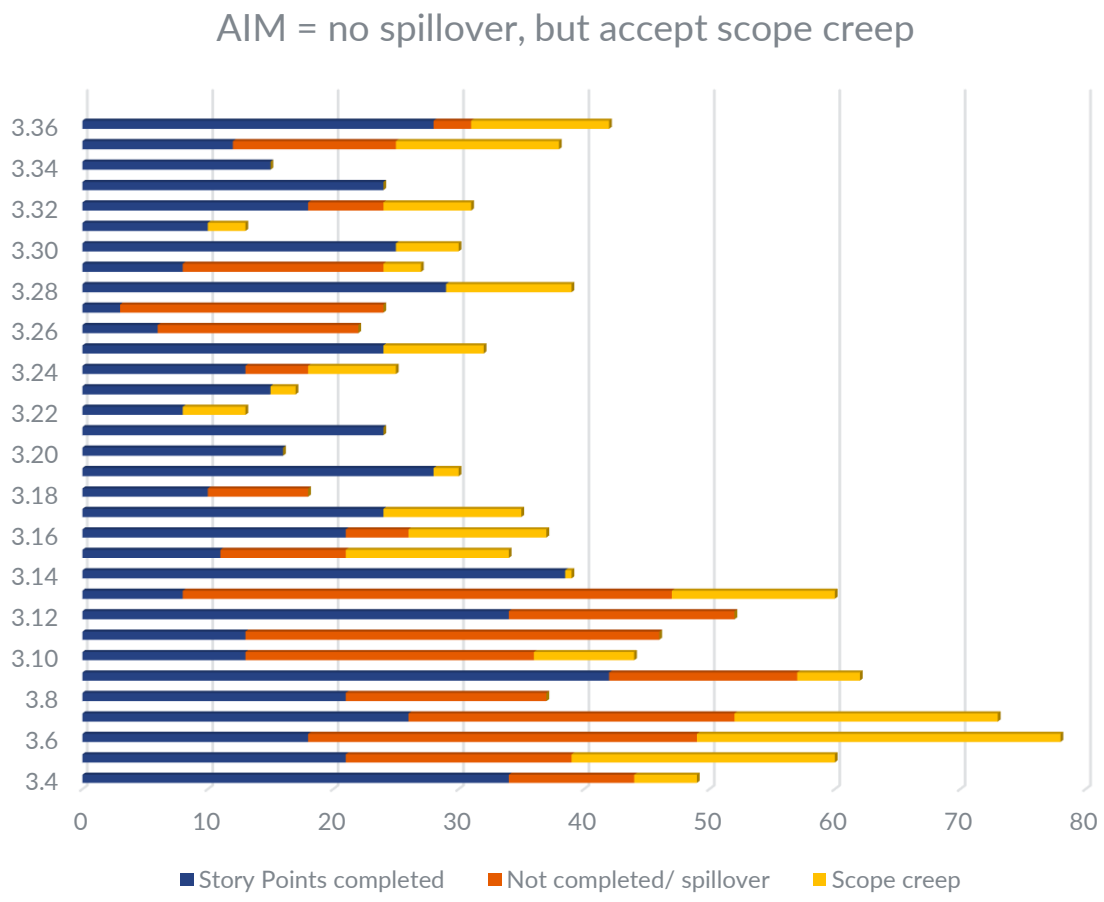
- A set of metrics to **understand what is happening** with the team
- These metrics are **for the team only**. They should not be used to gauge progress on the project

# Metrics you want to track

- **TEAM STABILITY** – if the team is unstable, the environment is volatile. Raise the issue that without stability, the team becomes unpredictable
- **RATIO OF COMMITTED VS. DELIVERED STORIES** – does the team over or under commits regularly?
- **SCOPE CREEP** – do we routinely start work which was not initially planned? What are the consequences?

# A real life example

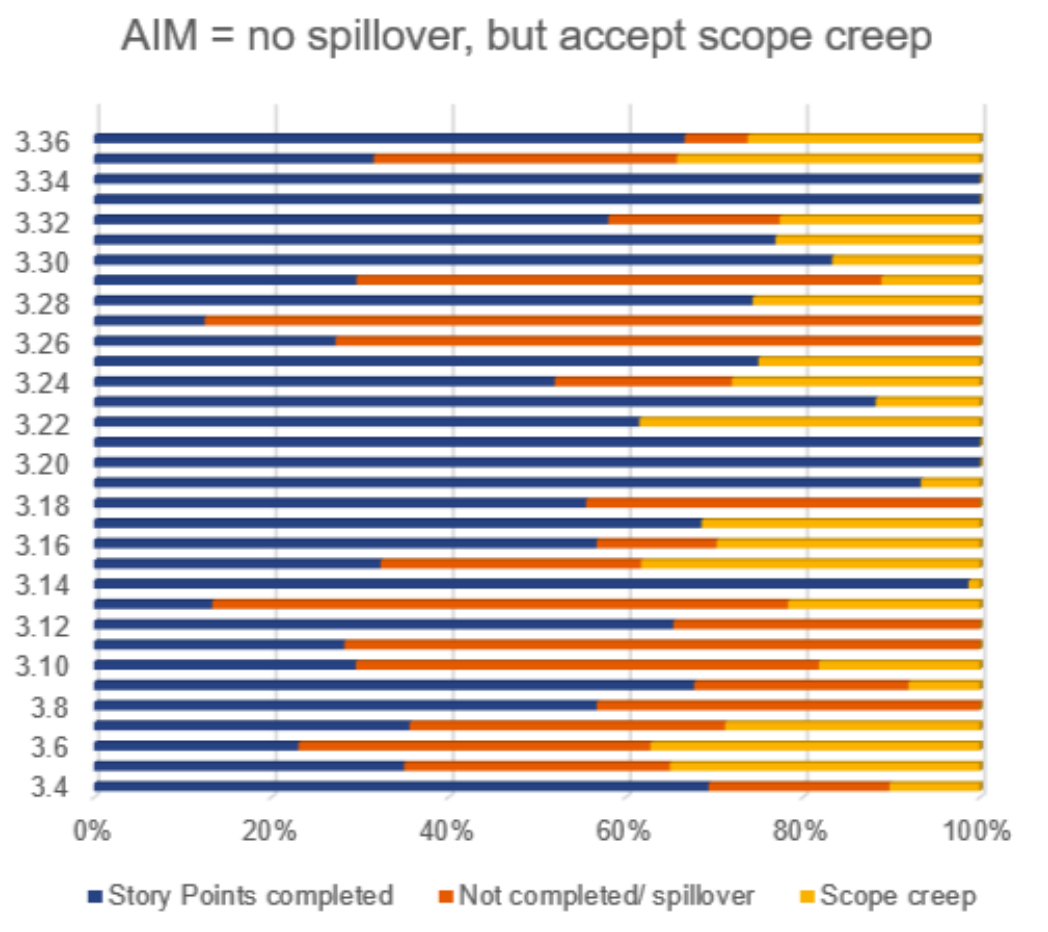
## Tracking metrics for Team SirPrice




- Sprints had both low rate of completion AND scope creep.
- We tracked the progression of the issue, having introduced the idea to accept scope creep
- What does this chart does not tell us?

# A real life example

## Tracking metrics for Team SirPrice




- We know nothing of team stability or team availability.
- We don't know of known bottlenecks requiring removal
- We don't know if we get a constant stream of bugs which need to be handled right away

# More **Metrics** you want to track

- **RATIO OF DEVELOPMENT TASKS vs. BUGS vs. SPIKE, ETC.**– are we spending a lot of time fixing bugs? The team may be trying to work too fast, discounting quality.
- **BOTTLENECKS** – is there a particular status where tickets/stories get stuck?
- **CYCLE TIME**– how long does it take for a ticket/story to reach the done state from when it was ready to be developed? If the cycle time is high, it may mean that our tickets/story are too big or complex.

# Metrics provide value when we put them **in correlation** with each other

- Team Metrics help us **paint a picture** of the work of the team
- The provide tangible **evidence of anti patterns** and can help find solutions (by changing ways of working, escalating, etc.)
- They should **generate a conversation**, NOT substitute it
- Teams should **not get fixated on metrics**: they will never be perfect, and that's ok

With some data at hand, it is time for the team to discuss how to improve and to record new commitments in the Team Improvement Backlog.

# The Team Improvement Backlog

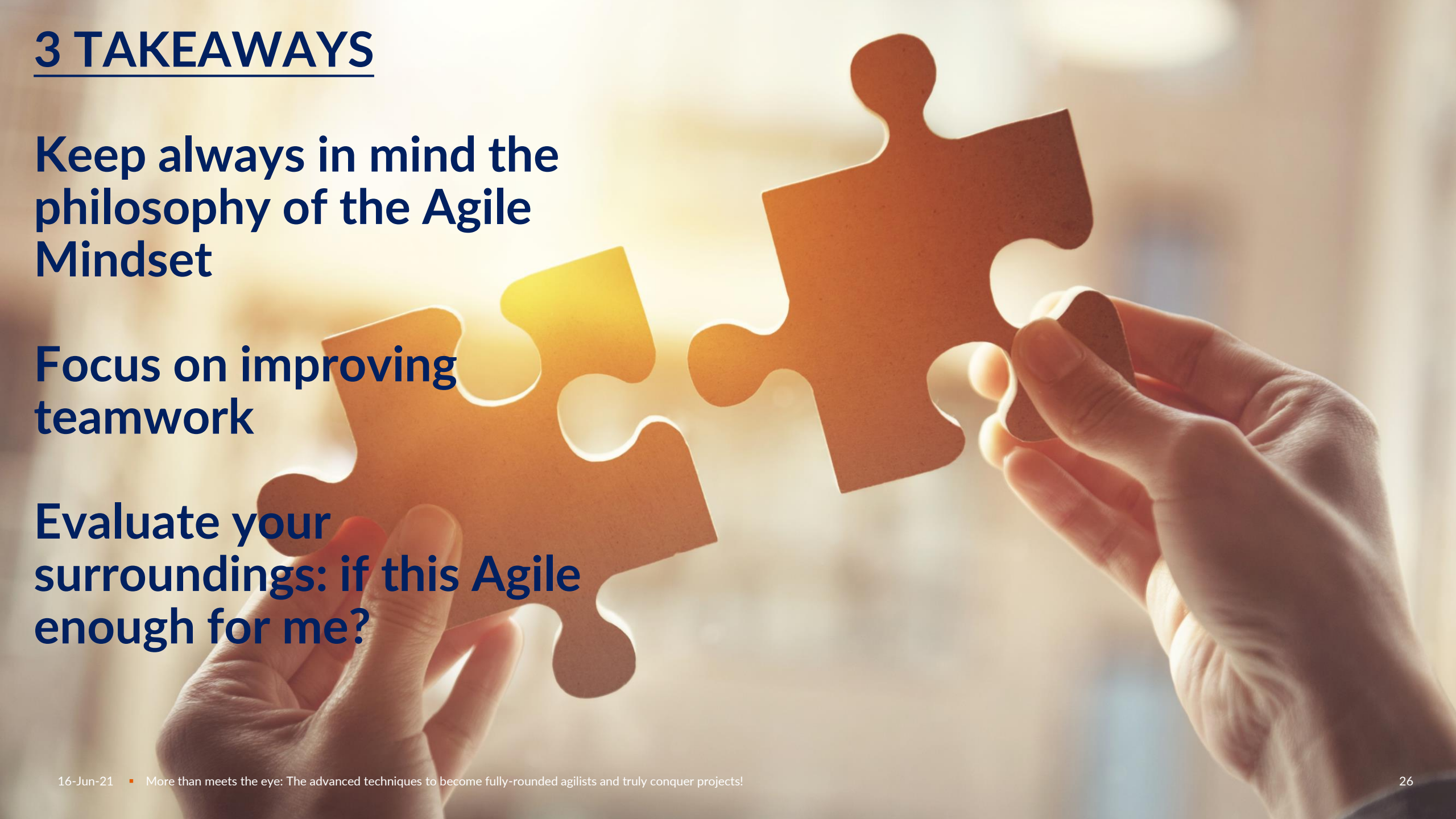
- An **evolving repository of intentions** and actions to **improve teamwork**
- Like a backlog, keep track of what you want to implement, what you are trying to and what is reliably implemented

# Team Improvement Backlog: a real-life example

## Part of the Team SirPrice improvement backlog

| Backlog | Progress | Currently reliably implemented |
|---|---|---|
| **Be more diligent about data at the beginning of the sprint**. Let's try to be ahead of the game and check data before tackling the story. | **Switching Epics:** The tickets from the same epic should not all done by one person. Someone takes the first, then someone continues, so we avoid silos. This will promote better notebooks | Do **weekly knowledge share** within the team |
| Plan for **two workstreams at the Refinement and Planning** (make sure that planned streams have suitable story points) | The team wants to **commit to breaking down tickets more** | Attempt to **reduce** to zero **sprint spill-over** (incomplete stories). We accept to have scope increase |
| **First create smaller tickets**, then group them around a story, maintaining the hierarchy: Epic---> Story--->smaller story/subtask/task | Commit to keep in mind this resolution: **minimal notebooks and have other functions in other files** whenever possible, to become more end to end and engineering focused | PAIRING - Tickets should be **peer reviewed by two people. Peer reviewing of tickets must be done by the pair who has not worked on that story (where applicable)** |
| Adopt best practices for design and development work as defined by the Machine Learning Engineers (MLFlow but also coding standards for notebooks and more) | | PAIRING **- Keep Pair Programming to 2 people** |
| **Focus on analytical conclusions as much as providing insights - Translate the results into actions more**. Every ticket should have a valid conclusion. We need to add to tickets in Jira a clear summary or comments with the next steps, expressed as a recommendation for what to do next. | | PAIRING **- Increase pairing** where possible, and make it a habit, even when tickets sound trivial |
| TEAM REPO - even if the repo is not runnable, let's commit to collect plots and some functions which we could search for after. A description could be good too. | | PAIRING -Split the Data Scientists in **two pairs and regularly rotate** them |
| | | PAIRING - Each pair works on one ticket at the time |

# 3 TAKEAWAYS



**Keep always in mind the philosophy of the Agile Mindset**

**Focus on improving teamwork**

**Evaluate your surroundings: if this Agile enough for me?**

Growth
from
Knowledge

# Thank you for having me! Again!!!

And just in case you are interested, this is our career page:
www.gfk.com/careers