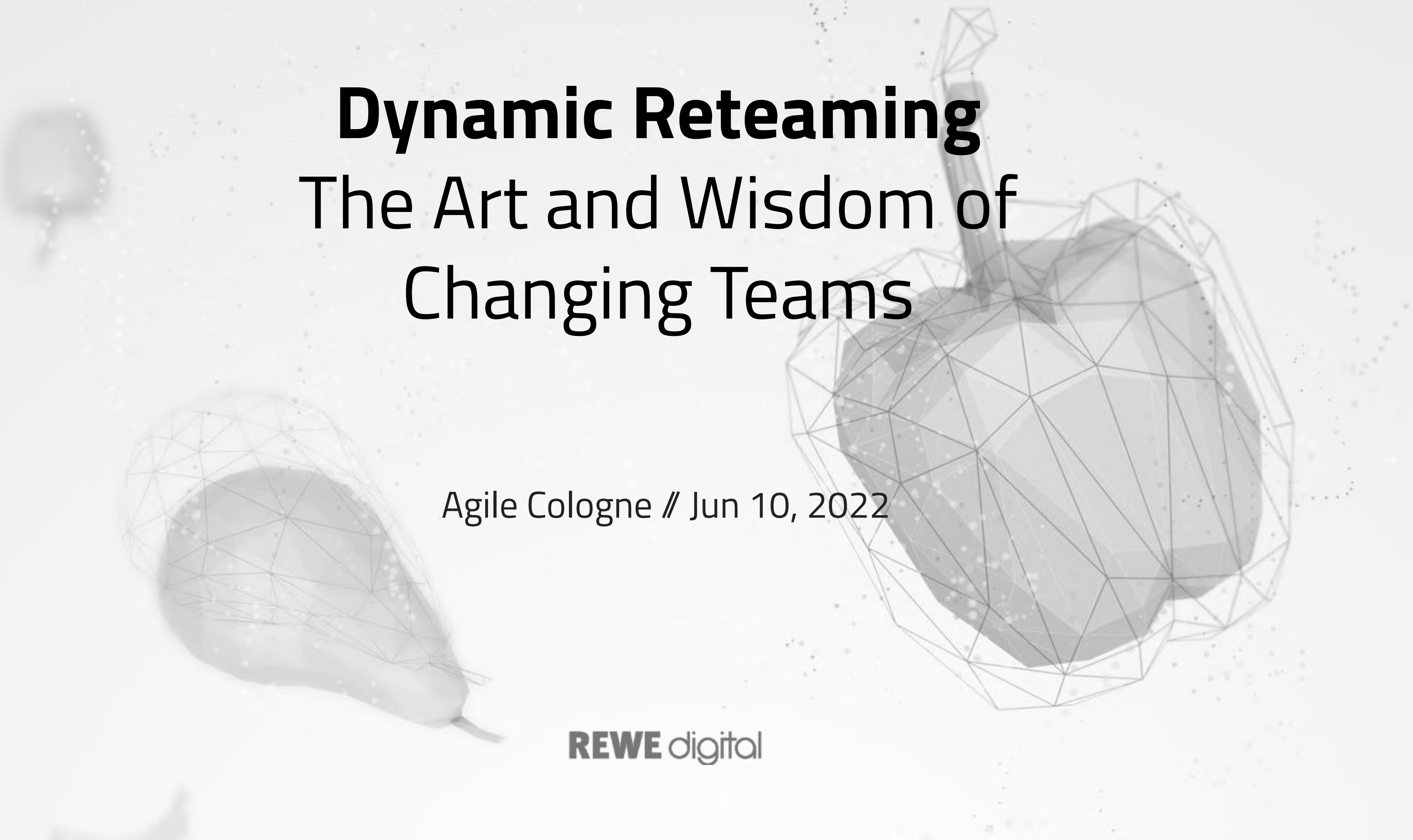
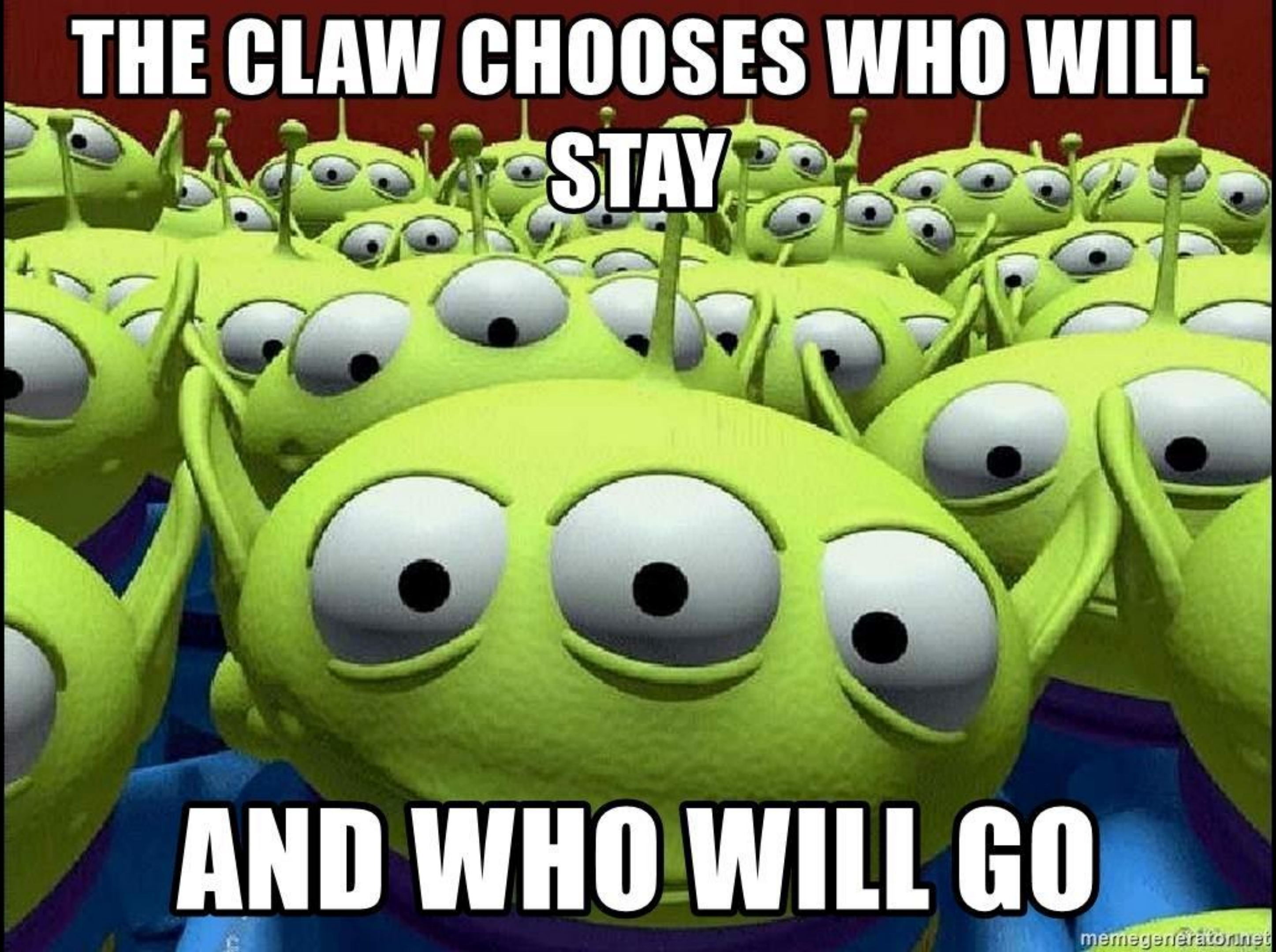


Dynamic Reteaming

The Art and Wisdom of Changing Teams

Agile Cologne // Jun 10, 2022

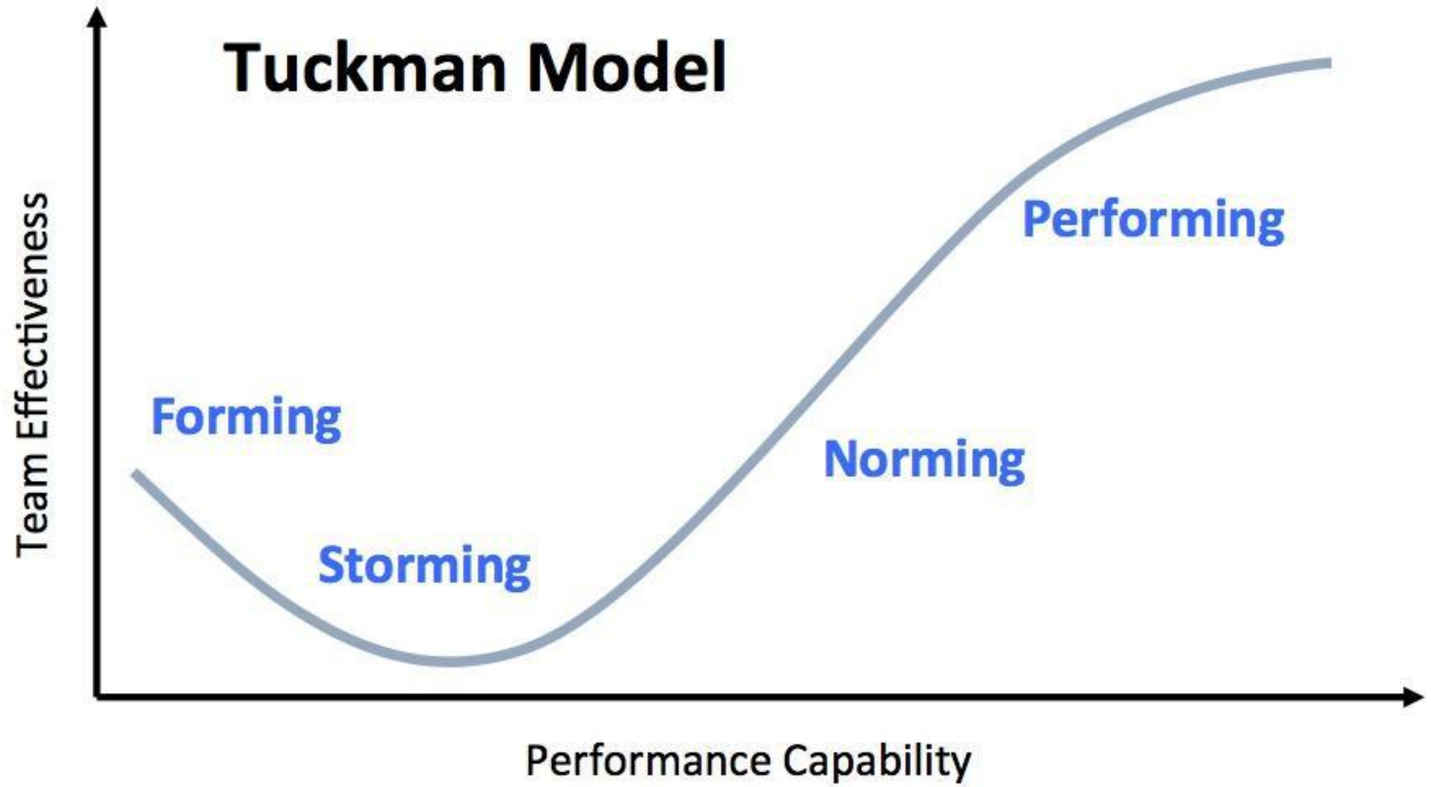
REWE digital


THE CLAW CHOOSES WHO WILL STAY
AND WHO WILL GO
memegenerator.net

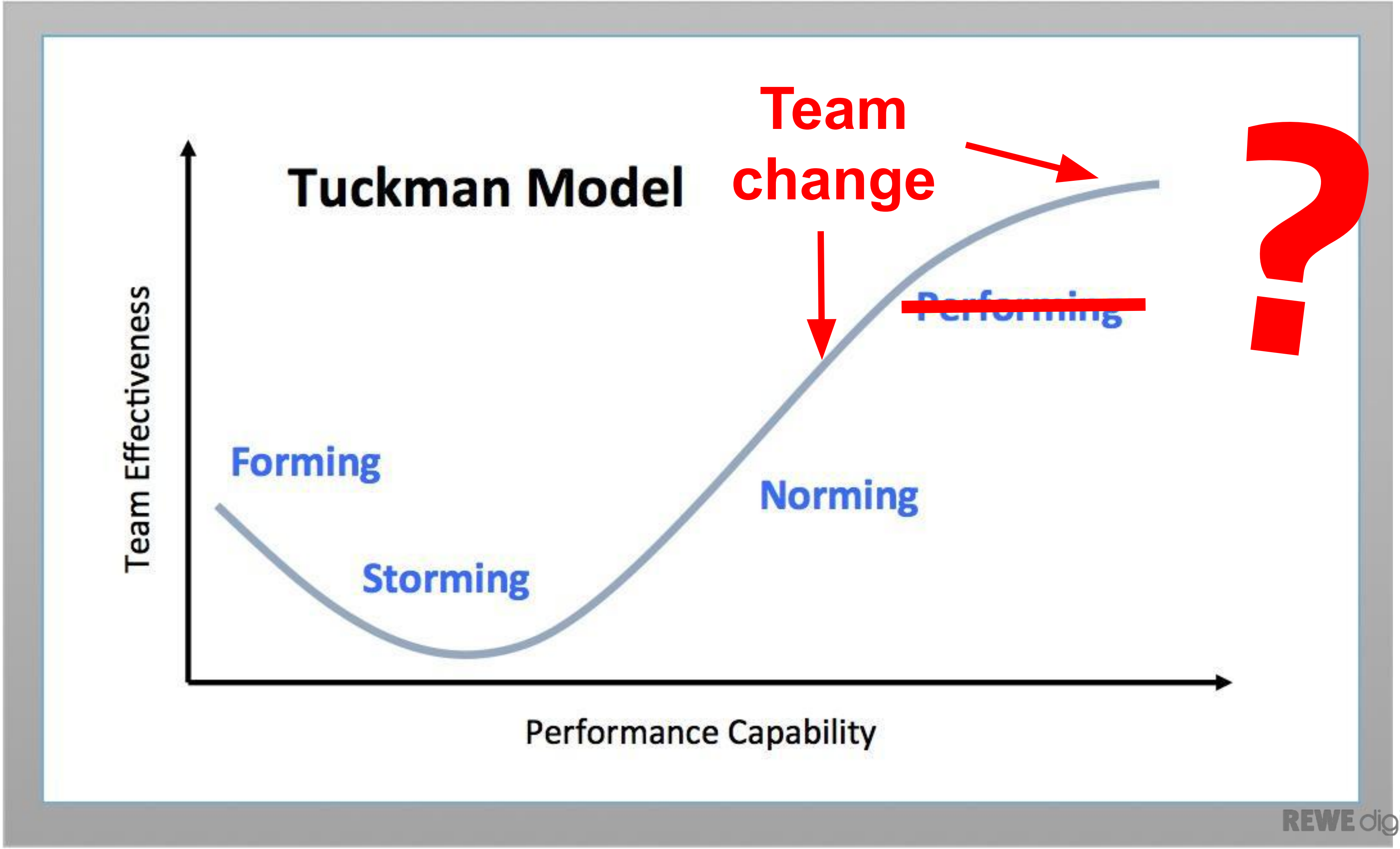
Stable teams

Stable teams are awesome!

- Ownership
- Mastery
- Autonomy
- Purpose
- ...



Tuckman Model


Tuckman Model
Team Effectiveness
Performance Capability
Forming
Storming
Norming
~~Performing~~

Team change

?

Too much of a good thing…

- Focus <> Tunnel vision
- Local optimization <> Thinking Inside-the-box
- Stability <> Status quo bias
- “Do not disturb” <> Shrinking comfort zone
- “Us” <> “Them”

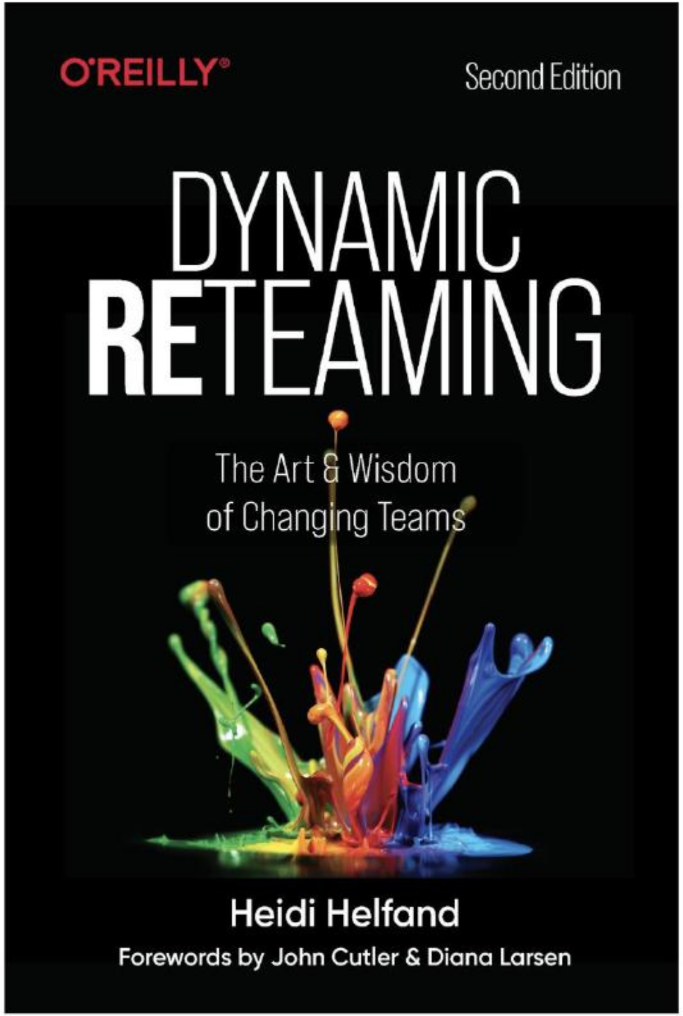
~~Stable~~ Static teams

Static teams MIGHT…

?

- focus only on performing
- miss the bigger picture
- neglect innovation and sustainability
- question psychological safety
- participate less in organizational matters
- simply implode
- lead to an **overall less flexible** system

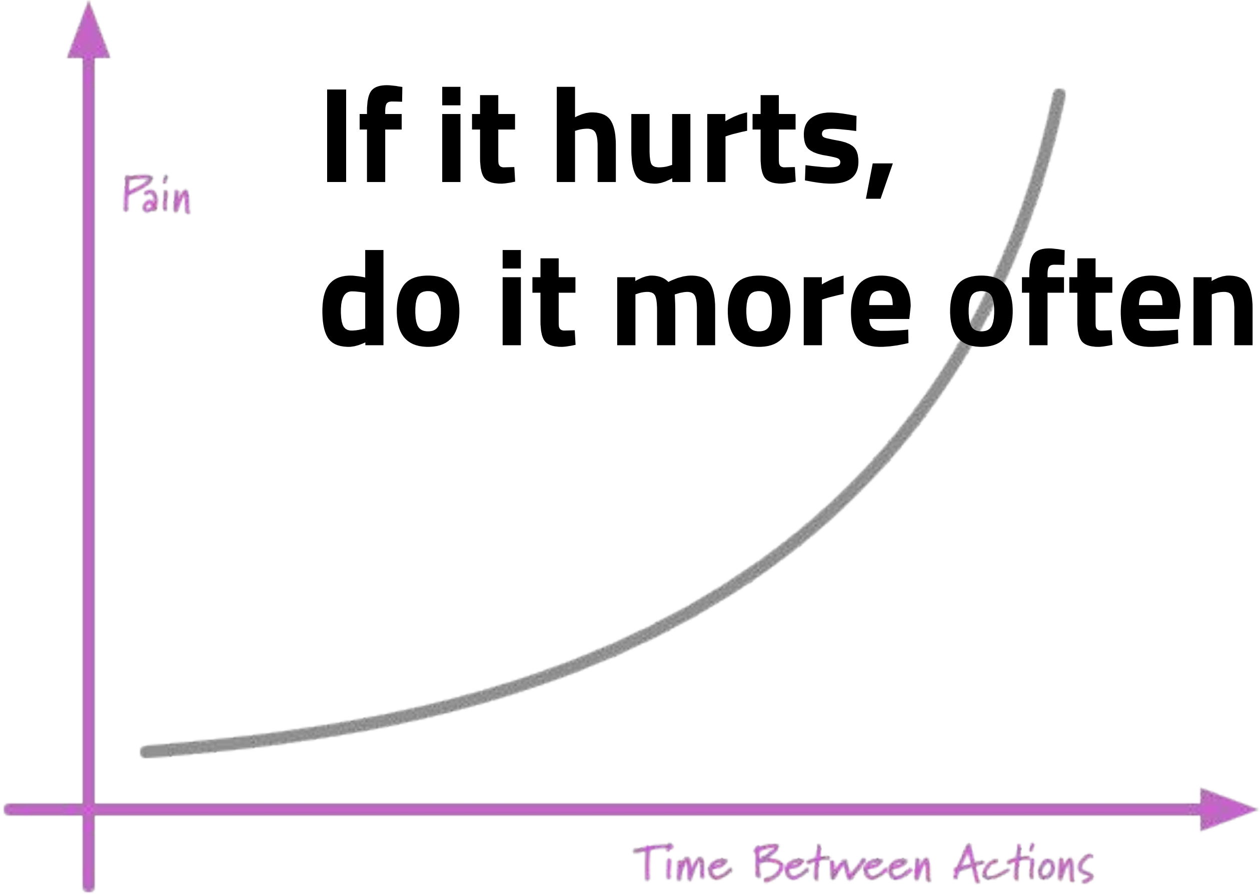
Dynamic Reteaming



Teams are changing all the time, so we might want to get good at it!

REWE digital

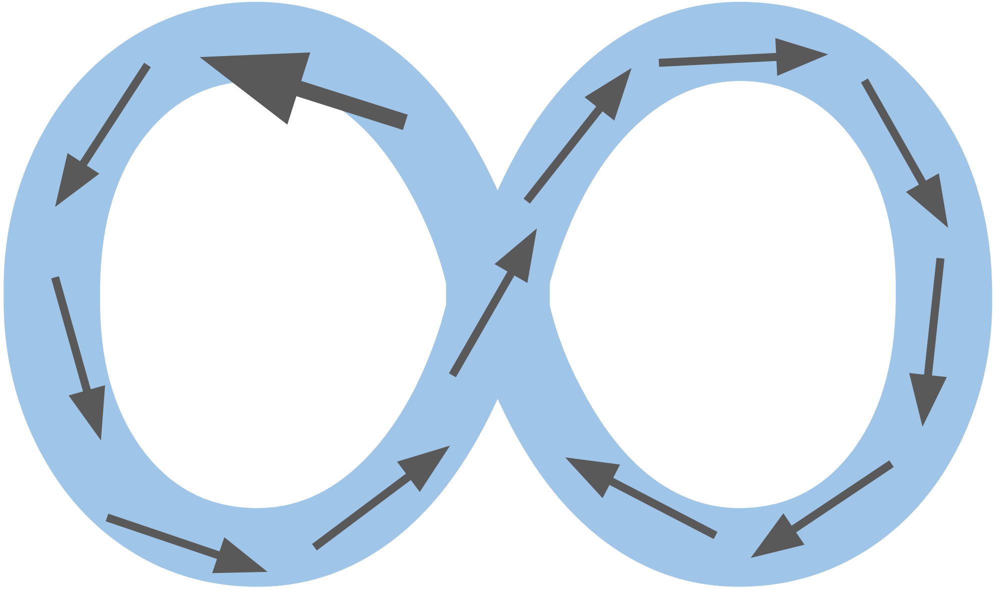
If it hurts,
do it more often...



https://www.martinfowler.com/bliki/FrequencyReducesDifficulty.html

Birth

Maturity



Adolescence

Creative Destruction

Reteaming Patterns

- One-by-one
- Grow-and-split
- Merging
- Isolation
- Switching

Case studies and learnings

Example: Redgate (pretty close)

- Adapt to changing strategy every year
- Focus on switching pattern
- Self-selection event with team charters
- People name preferences
- Reteaming occurs over a month
- A third of people move teams

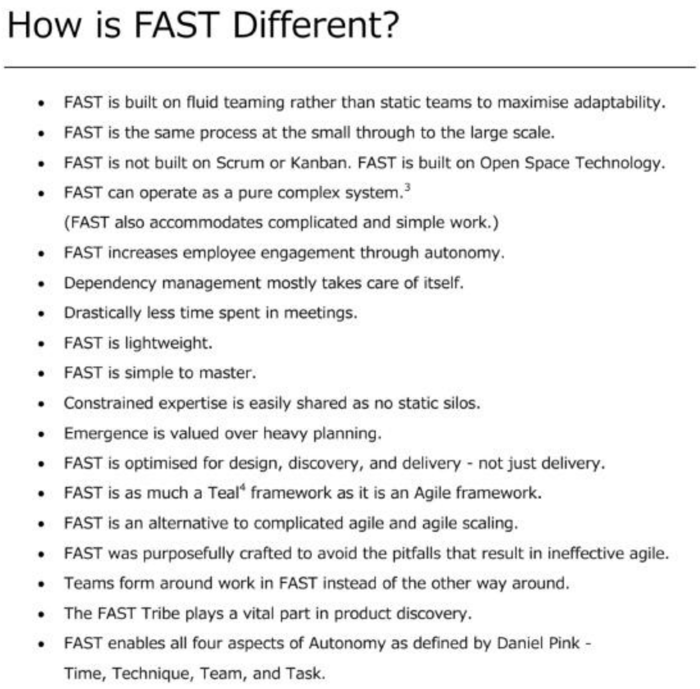
https://leadingagileteams.com/2021/11/23/podcast-appearance-dynamic-reteaming-at-redgate-on-mastering-agility/

Example: Tesla (far out)

- Everything is optimized for speed (“Accelerate the world’s transition to sustainable energy”)
- Backlog is generated by machine learning
- Open-space-like reteaming per shift (“where can I be most useful?”)
- Collaboration similar to mob programming

https://www.youtube.com/watch?v=EwtRJDRcXsY

FAST Agile



How is FAST Different?

- FAST is built on fluid teaming rather than static teams to maximise adaptability.
- FAST is the same process at the small through to the large scale.
- FAST is not built on Scrum or Kanban. FAST is built on Open Space Technology.
- FAST can operate as a pure complex system.[3]
  (FAST also accommodates complicated and simple work.)
- FAST increases employee engagement through autonomy.
- Dependency management mostly takes care of itself.
- Drastically less time spent in meetings.
- FAST is lightweight.
- FAST is simple to master.
- Constrained expertise is easily shared as no static silos.
- Emergence is valued over heavy planning.
- FAST is optimised for design, discovery, and delivery - not just delivery.
- FAST is as much a Teal[4] framework as it is an Agile framework.
- FAST is an alternative to complicated agile and agile scaling.
- FAST was purposefully crafted to avoid the pitfalls that result in ineffective agile.
- Teams form around work in FAST instead of the other way around.
- The FAST Tribe plays a vital part in product discovery.
- FAST enables all four aspects of Autonomy as defined by Daniel Pink - Time, Technique, Team, and Task.

https://www.fastagile.io/home

- **Empower teams!**
  - Don’t revert to top-down team design
- **Be transparent!**
  - About goals and environment
- **Keep moving!**
  - Regard switching as default
  - Staying is fine, but make it a conscious decision

Thanks! What do you think?