

Why agile?

Principles and Values to Change our
Perspective of the World

SOFIA WOLOSCHIN

ICA-ACC, CSM, PMI-ACP, PMP

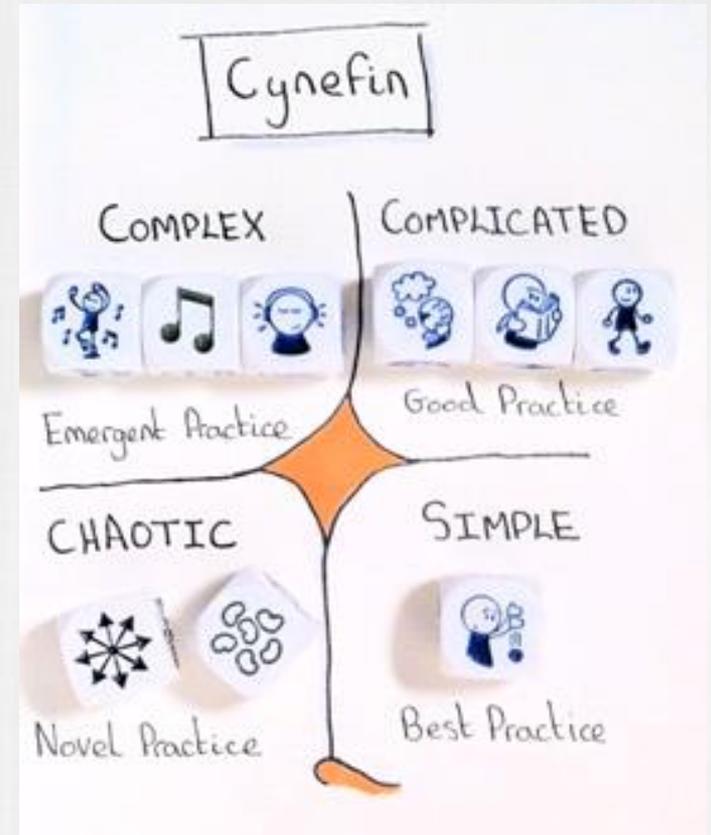
Experiences?

Have you been in an Agile team?

Complex Environments

Cynefin framework

- Inherent uncertainty of complex systems
- Uncertain markets – disruption - innovation
- Complex business logic
- Complex technical conditions
- Cultural impact, Relationships, interdependency: people
- Understand environment, infinite variables



Complexity

Where does the project sit in the scale?

Evolutionary approach

Anticipation

- Early planning
- “Complete” upfront design
- Post –development test
- Hand overs
- Early definition on all requirements

Adaption

- Real time Planning
- Emergent Design
- Integrate testing
- Collaboration
- Sign offs
- Just in time /enough requirements

What about?

Risk

Knowledge

Predictability

Collaboration

Quality

Change ...

Systems Thinking



People are already doing their best, the problems are with the system.

Only Management can change the system.

W. Edwards Deming

Systems Thinking

Learning System

Humility.
Incompetence.
Learning.
Co-evolution.

Create Community

Peer to peer "centricity".
Networks of practice.
Collective responsibility.

Autonomy

Self-organisation.
Empowered teams.
Peer pressure.
Team performance.

Accept Complexity

Cannot eliminate.
Cannot "manage".
"Live with the flow".
Learn your way out of it.

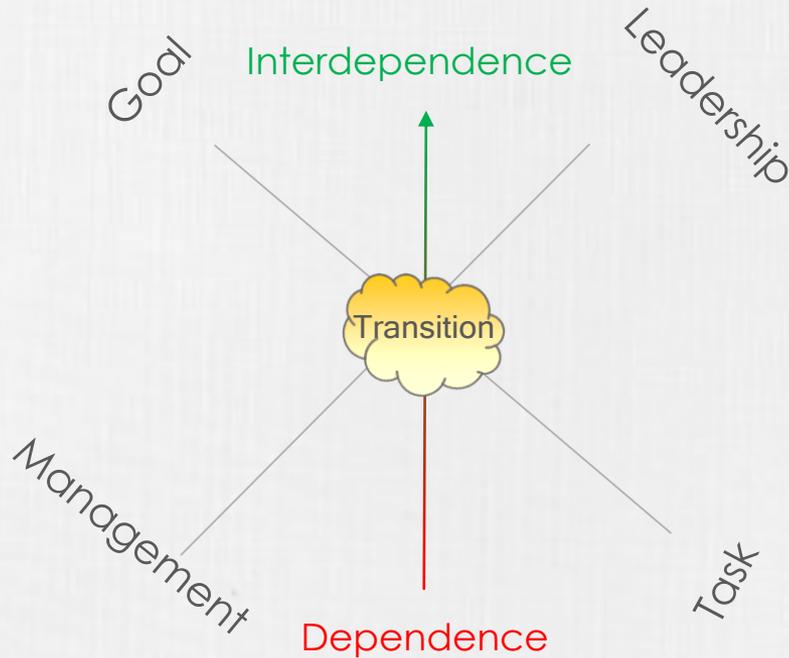
Social Intelligence

Shared knowledge.
Collectively wiser.
Shared decision making

New Leadership Mindset

LEAD. FACILITATE. FOCUS on VALUES. CLIENT CENTRICITY. STOP THE NONSENSE.

The new leadership mindset



LEADER

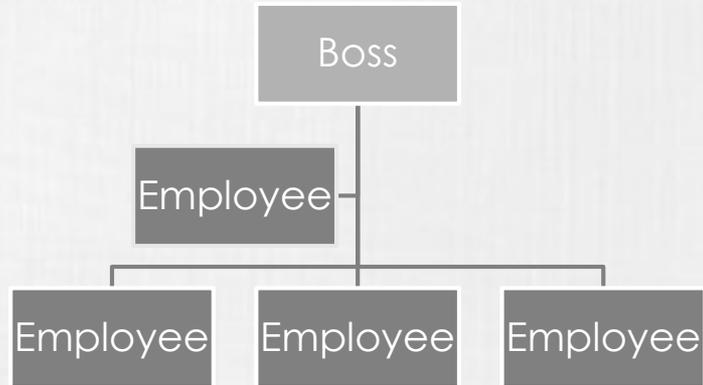
- Focus on *Outcome*
- Self managing team
- Knowledge in team / system
- Leader as Servant

MANAGER

- Focus on managing to Task
- Command and control
- Knowledge retained (boss)
- Parent / child relationship

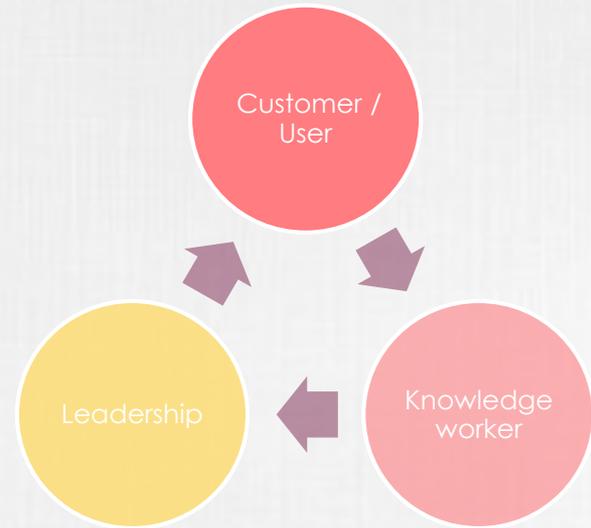
The new all mindset

Pre-Agile mindset



Hierarchies

Agile mindset



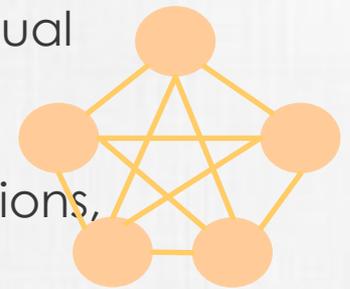
Networks

Interdependence

- Outcomes based management
- Team self organises
- Collective responsibility to achieve definition of done
- Team starts together and ends together
- Focus on cycle time, getting a piece of work done
- Skillsets needed to connect & succeed: T-shape skillsets
- Knowledge retained by the organisation
- Emergent properties part of the whole, not within individual part
- Success based on how to manage the system's interactions, not their action

Secondary skills

Specialty



Interdependence

Interdependence

How does it apply to you as a team member?

Feedback cycles



Feedback

which of these give you opportunities to introduce quality?

Defined vs empirical process

- Complex, interdependent systems cannot be “managed” by a defined process
- Empiricism enables a complex system to learn and adapt
- A group with autonomy to self organise is much faster at changing and adapting than in a bureaucracy
- Bureaucratic organisations impose process and rules to complexity, hence most of the projects fail.

Empiricism

Does it ring a bell...?

Agile as a value system

- Values based system - focused on what provides value, empowerment and continuous improvement
- Agile principles can be applied in other types of work to create an evolving and effective culture
- To establish this form of culture the focus is on the collective responsibility towards outcome, through the creation of cross-skilled teams.

Feedback

How does Quality equate to Value?

Agile Manifesto

We are uncovering better ways of developing software by doing it and helping others do it. Through this work we have come to value:

Individuals and interactions *over* processes and tools
Working software *over* comprehensive documentation
Customer collaboration *over* contract negotiation
Responding to change *over* following a plan

**That is, while there is value in the items on the right,
we value the items on the left more.”**

Agile Alliance, Manifesto of Agile Software Development, 2001

Agile Principles

1. Our highest priority is to satisfy the customer through **early and continuous delivery of valuable software**.
2. Welcome changing requirements, even late in development. Agile processes **harness change** for the customer's competitive advantage.
3. Deliver working software **frequently**, from a couple of weeks to a couple of months, with a preference to the shorter timescale.
4. **Business** people **and developers** must work together daily throughout the project.
5. Build projects around **motivated** individuals. Give them the **environment and support** they need, and **trust** them to get the job done.
6. The most efficient and effective method of conveying information to and within a development team is **face-to-face conversation**.
7. **Working software** is the primary measure of progress.
8. Agile processes promote **sustainable** development. The sponsors, developers, and users should be able to maintain a **constant pace indefinitely**.
9. Continuous attention to **technical excellence** and **good design** enhances agility.
10. **Simplicity**--the art of maximizing the amount of **work not done**--is essential.
11. The best architectures, requirements, and designs emerge from **self-organizing** teams.
12. At regular intervals, the team **reflects** on how to become more effective, then **tunes and adjusts** its **behaviour** accordingly

Doing Agile vs Being Agile

A couple of war stories ...

It is not the icing...

Software quality is fundamental to creating valuable, sustainable software that works, is of technical excellence and good design.

You build quality from the moment you write the first line of code. It is built in, you cannot add it later ...

