

Coach Takeshi's

# Agile 101

Lifecycle  
agile-od.com



# What is Agile

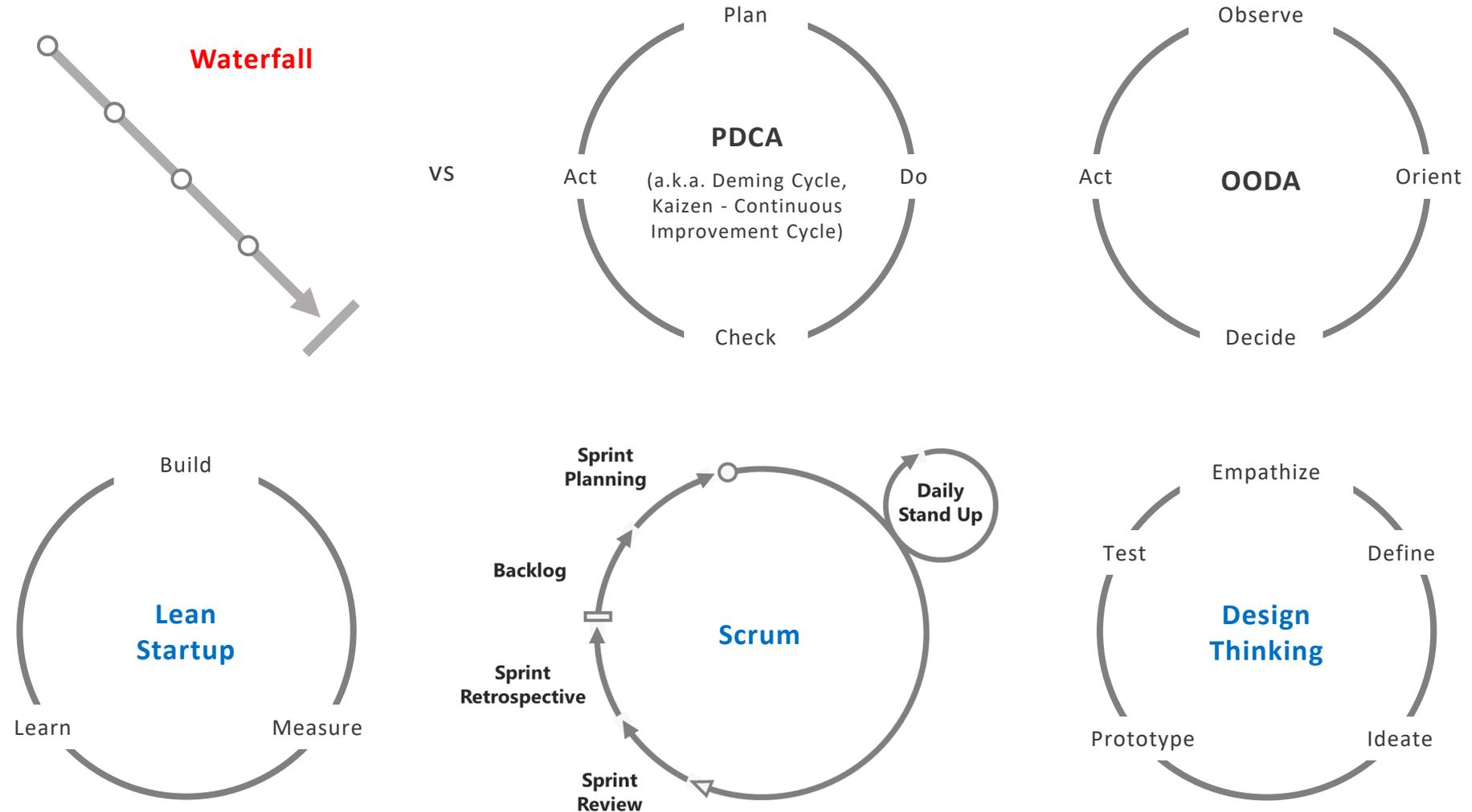
# Glossary: Agile, Lean, Scrum, Sprint, Design Thinking

- **Agile** and **Lean** are technically different things, but it's okay to consider them similar – **experimental & iterative approaches** for highly uncertain challenges.
- **Lean Manufacturing (Lean TQM)** and **Lean Startup** are technically different things, but it's okay to consider both as **Lean**.
- **Scrum** is one of the most popular Agile implementation frameworks.  
**Sprint** is a Scrum terminology. It's what the iteration cycle in Scrum is called.
- Loop **Design Thinking** in the Agile bucket too, as again it's in the same spirit.
- To summarize, consider: **Lean ≈ Agile > Scrum > Sprint** and  
**Design Thinking ∈ Agile**
- *There are further derivatives such as **Design Sprints** and **USM** (User Story Mapping), which are again purpose driven tools that have their own valid use (I use them both too). **XP** (Extreme Programming), **Kanban** (not to be confused with Lean Manufacturing's Kanban) etc. are software development framework alternatives to Scrum.*
- ***Scaled Scrum** or **Scrum of Scrums** is looking like a battle of methodologies with **LeSS**, **SAFe**, **Nexus**, **Prince2** etc. each claiming superiority.*

# Is Agile a fad?

- **Agile** did not invent trial and error.
- In fact, **PDCA** (Plan-Do-Check-Act, W. Edwards Deming, 1950), **OODA** (Observe-Orient-Decide-Act, John Boyd, 1976) and many more thoughts on trial and error going all the way back to Socrates (circa 400BC) and Confucius (circa 500BC), all teach the same.
- They are all models of **experimentation & iteration**.
- By default, we humans don't like trial and error – it's a biological trait stemming from our survival instincts of uncertainty avoidance and resistance to change. Meanwhile, risk taking is rewarding, and the probability of success from risk taking increases through trial and error – that's a fact from pre-historic times. That's why throughout history many strings of experimental and iterative models have emerged as each is a new attempt to get better at trial and error.
- Agile is just one such latest attempt to get **experimentation & iteration** right. It doesn't matter what it's called – the spirit is same.

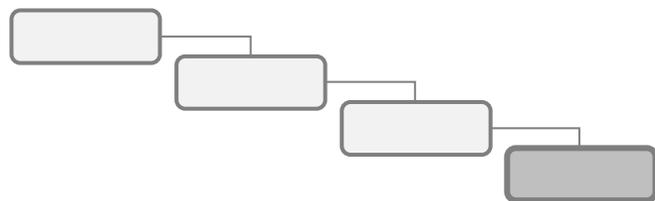
# The many flavors of **experimental & iterative** approaches



## Why Agile?

# (1) Agile is an antidote to waterfall project management

## Waterfall Project Management



*One waterfall*

*No testing until completed*



VS

## Agile Product Development



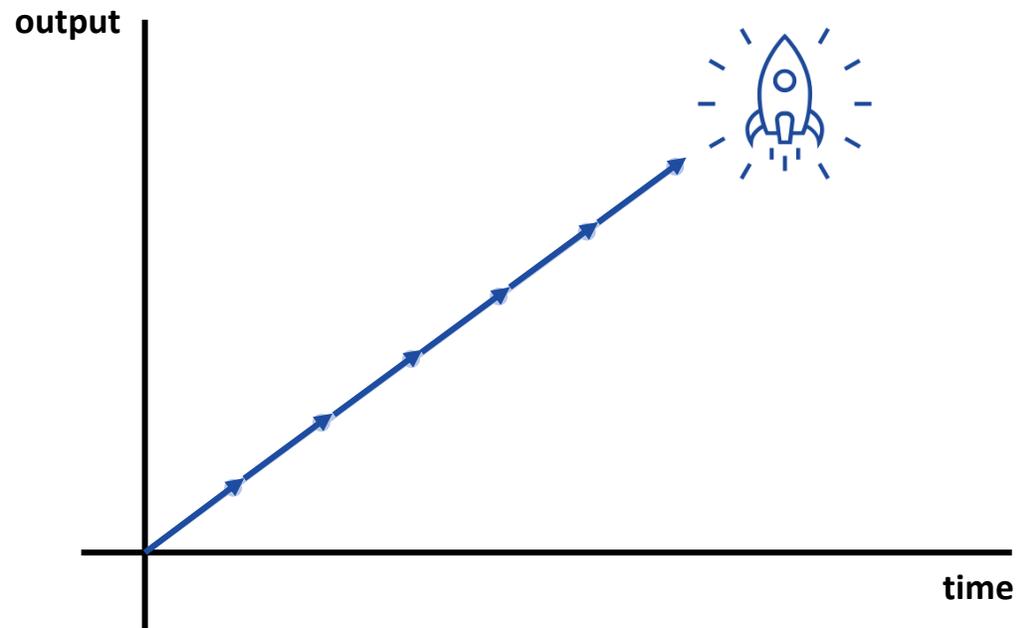
*Many Sprint iterations*

*Many Small test with MVPs*

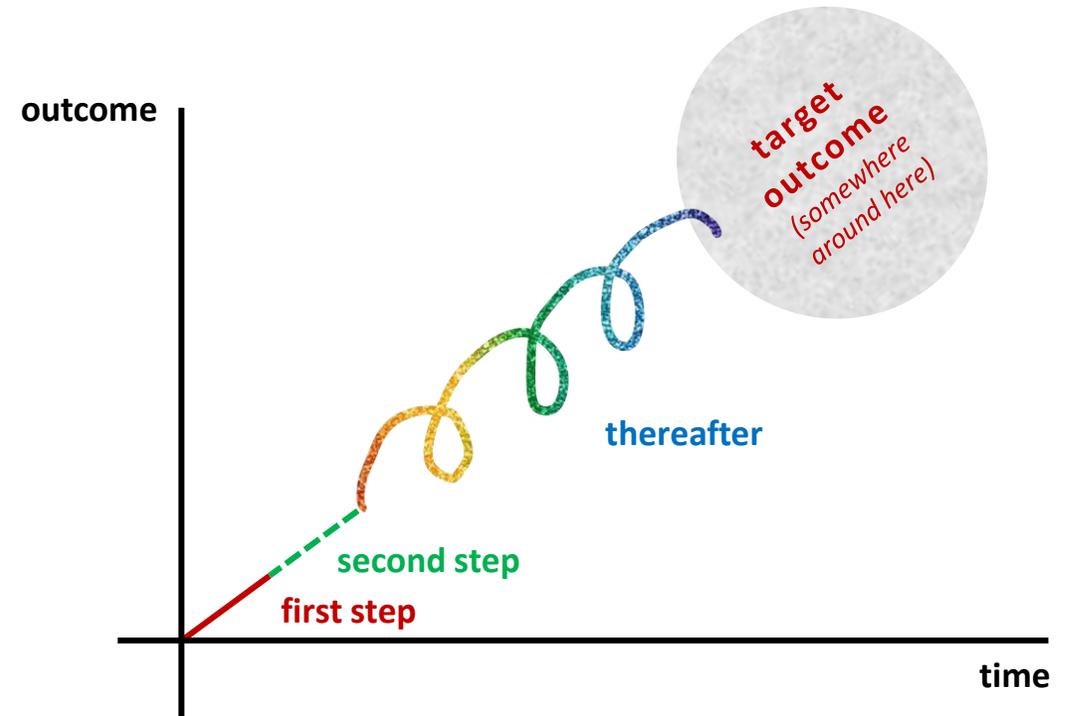


# The **ambivalent** nature of Agile product development

**Waterfall**  
Project Management

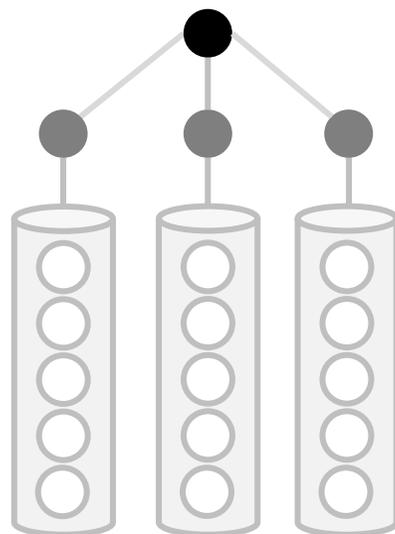


**Agile**  
Product Development



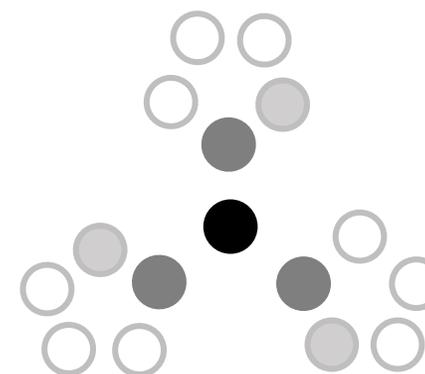
## (2) Agile is an antidote to the vertical organization

### Traditional Organization



VS

### Agile Organization



- *Centralized hierarchical structure*
  - *Leaders make strategic decisions*
  - *Middle managers make tactical decisions and gives instructions*
  - *Rank & file execute and deliver*
- *Functional silos*

- *Decentralized network structure*
  - *Self-organization*
  - *Team makes its own decisions*
  - *Leaders as facilitators, managers as coaches*
- *Cross-functional teams*

## Soft reason, why Agile?

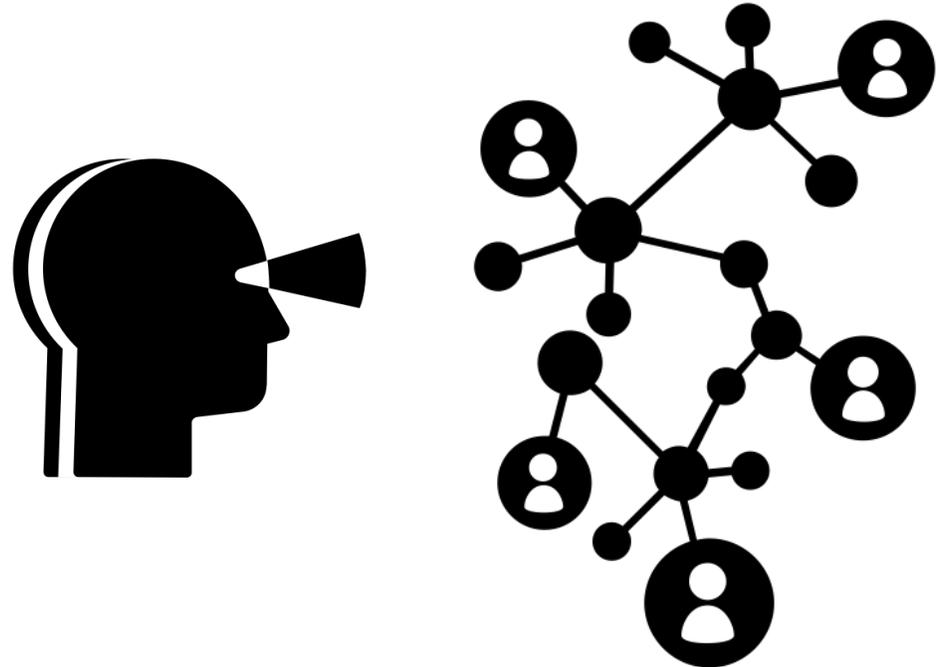
*Because we need a new way to **lead** in the age of **networked knowledge workers**.*

Social and digital is accelerating the service economy transformation in our world – the essence of Globalization.

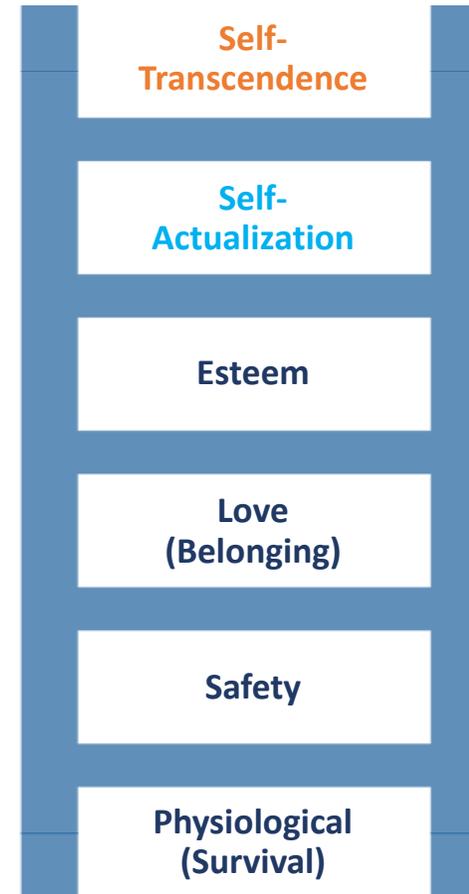
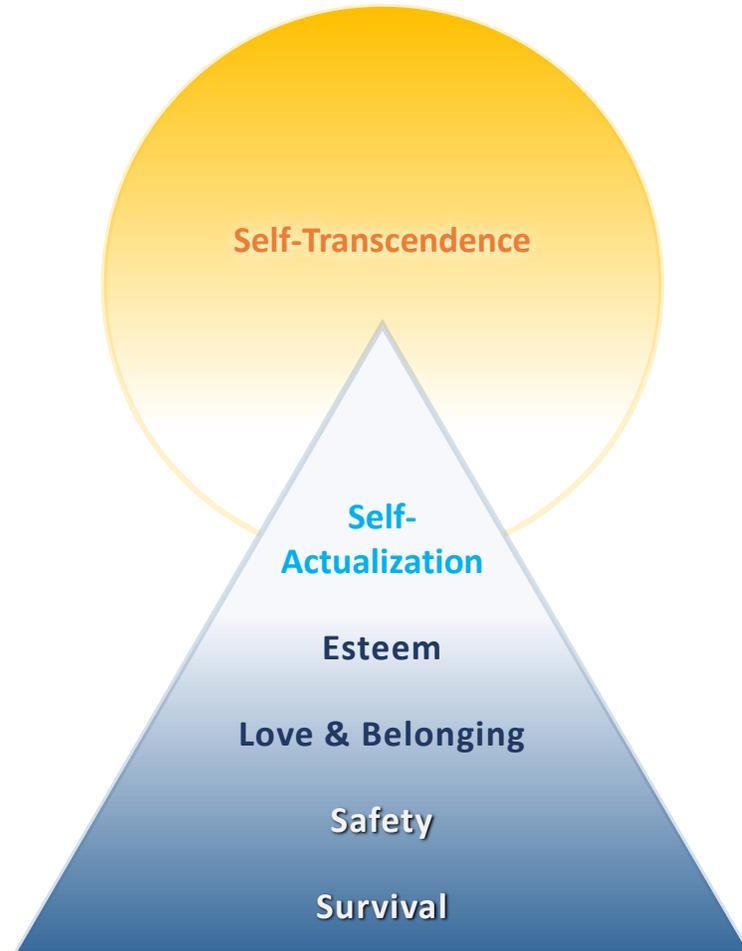
In the service economy, the competitive advantage is knowledge (including skills and expertise) and connectivity (access to resources – technology is the enabler). At the individual worker level, professional development is naturally gravitating to these two attributes.

Innovate or die from obsolesce – this is the reality that organizational leaders face today. And the world is now way to complex for a single master mind leader to know what to do. Survival of the organization depends on how well leaders can tap into the collective intelligence and drive of today's professional – the networked knowledge worker.

The new organization needs to follow the amorphous nature of the knowledge workers' networks. It's time to evolve from the traditional vertical, hierarchical organization, to a more cross-functional, networked organization.



# Soft reason, why Agile?



- **Personal mastery** is strong intrinsic motivation for **self-actualization**. When our autonomy is respected, we take strong pride in our work.
- When we group, it's natural for us to go over and beyond for the collective good; i.e. **self-transcendence**. Our strong sense of **belonging**, powers self-organization.
- **Trust**, is at the heart of agile. When we believe in our ability to work autonomously and self-organize, we tap into the highest forms of human motivation.

Visual conceptual adaptation of Maslow, A. H. (1969). Theory Z. The Journal of Transpersonal Psychology, 1(2), 31-47.

# Hard reason, why Agile?

Assumptions	Investment	Probability of Success	Return on Success	Weighted Average ROI
Investment in one go	\$100	20%	x20	\$400
Investment in 5 iterations Each iteration +20% "Kaizen" (continuous improvement)	\$20	20%	x20	\$80
	\$20	40%	x20	\$160
	\$20	60%	x20	\$240
	\$20	80%	x20	\$320
	\$20	100%	x20	\$400
Outcome	\$100		ROI 3x	\$1,200

High risk,  
high return  
opportunity

Product  
Market Fit  
NAILED

Now you can MILK  
THE COW

*It doesn't stop here*

Agile **DE-RISKS**  
Agile **MAKES MONEY**

# Agile **Scrum**

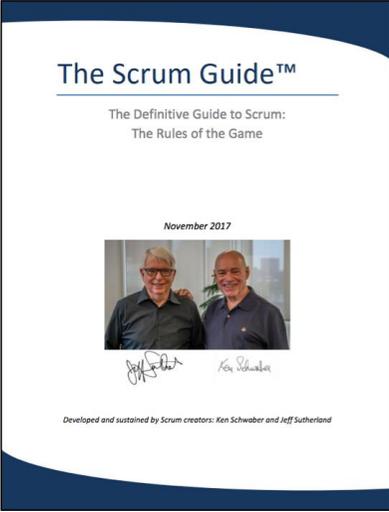
# Scrum: the most popular implementation framework in Agile

- Started as a software development framework, but wide application today; e.g. applied to business, operations.
- A **time-constraint approach** where work is delivered in short (typically 2 week “Sprints”) iterations.
- A team based approach: Scrum Teams are comprised of 3 to 9 **Development Team** (“DT”) members, one **Scrum Master** (“SM”), and one **Product Owner** (“PO”).
- The SM is the “how” guy. The PO is the “what” guy. **Both SM and PO are not bosses (not managers)**. Scrum Teams **self-organize**.

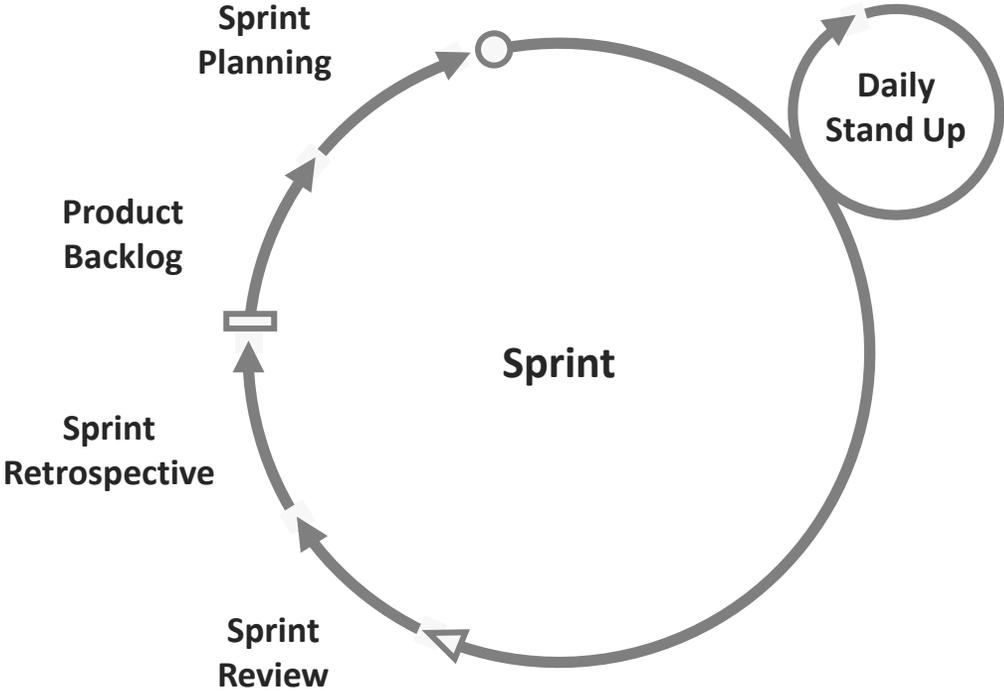


Images courtesy of sp-studio.de. Character concepts by Ron Eringa.

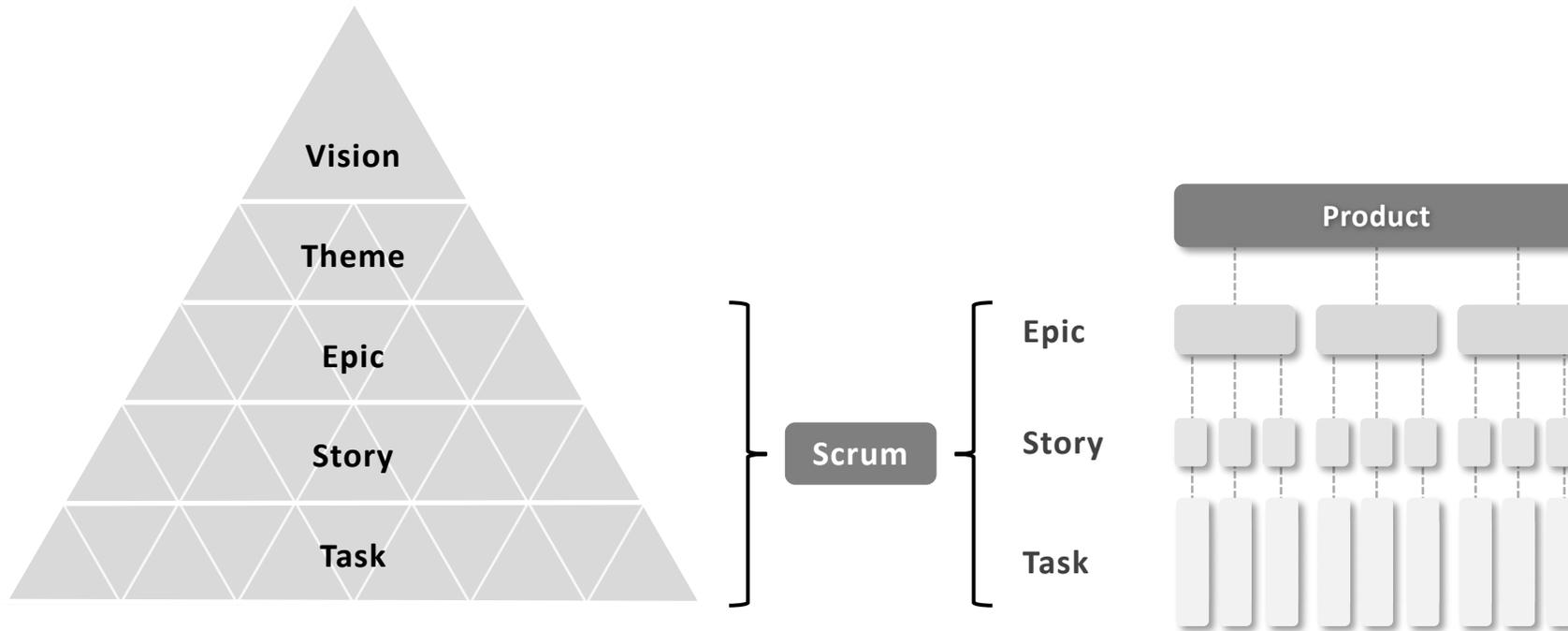
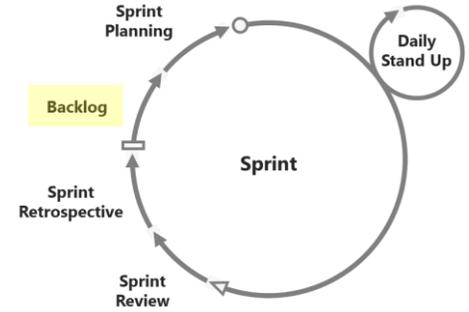
# Scrum is a lightweight framework for developing complex products



<https://www.scrumguides.org/docs/scrumguide/v2017/2017-Scrum-Guide-US.pdf>



# Scrum Product Backlog & Sprint Backlog



# Scrum – Sprint Planning

- **Have Product Backlog refined and prioritized**

Think of everything that would be needed in the product. User Stories are helpful ways of expressing the value of Product Backlog items.

- **Estimate work**

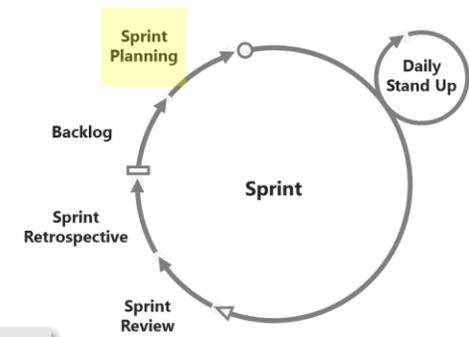
Estimate how much effort it will take each Product Backlog item to get done. Use Planning Poker etc.

- **Pull Product Backlog items into the Sprint Backlog**

What are the items that have the biggest business impact, that are most important to the customer, that can make the most money, and are the easiest to do?

- **Decide what To Do in the coming Sprint**

Only put in what can be done within the Sprint. Definition of Done for each Product Backlog item is a prerequisite.

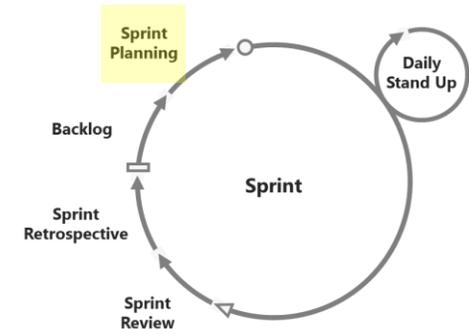
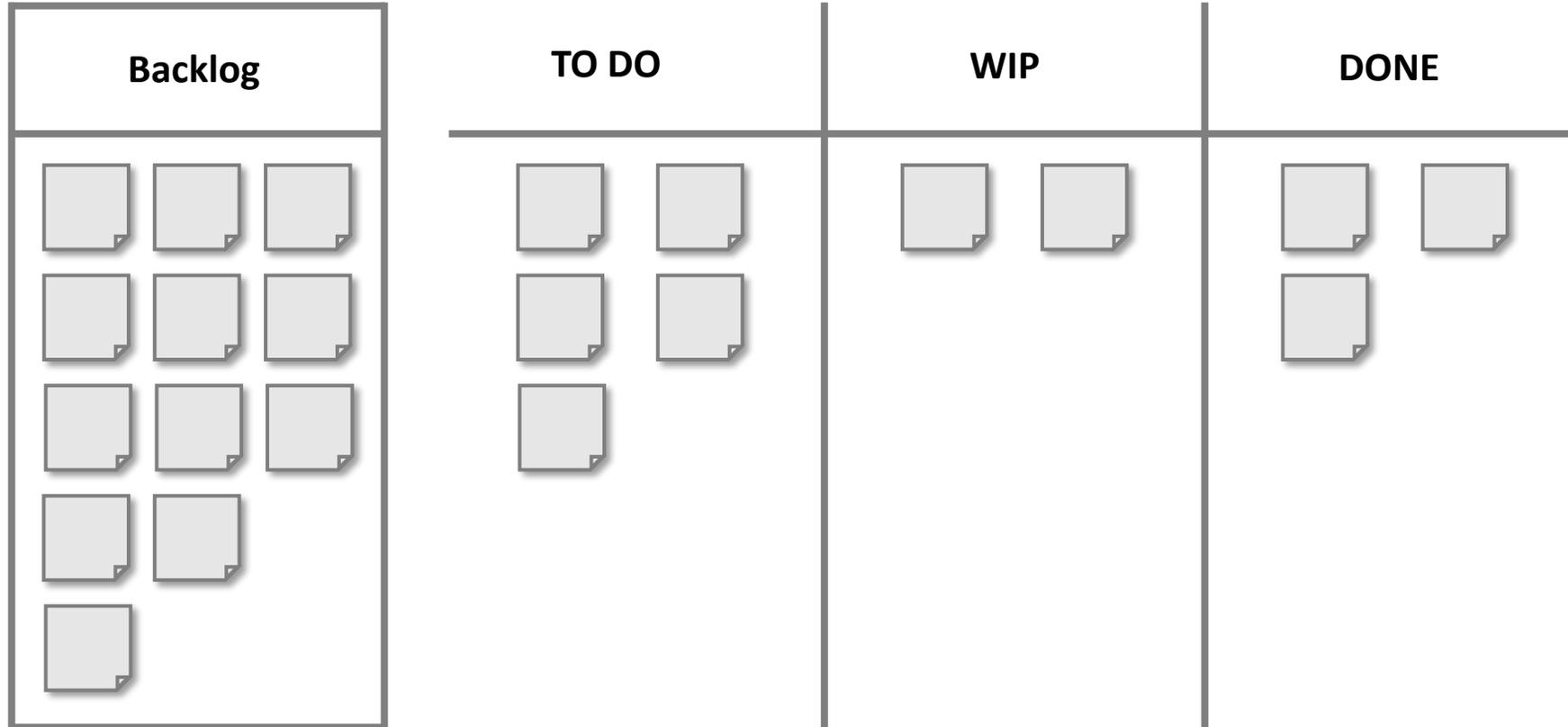


**User Story:**  
**As a [user],**  
**I want [what],**  
**so that [value].**

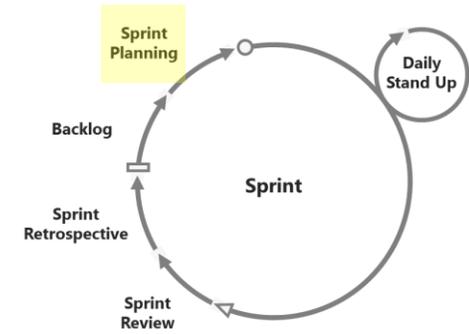
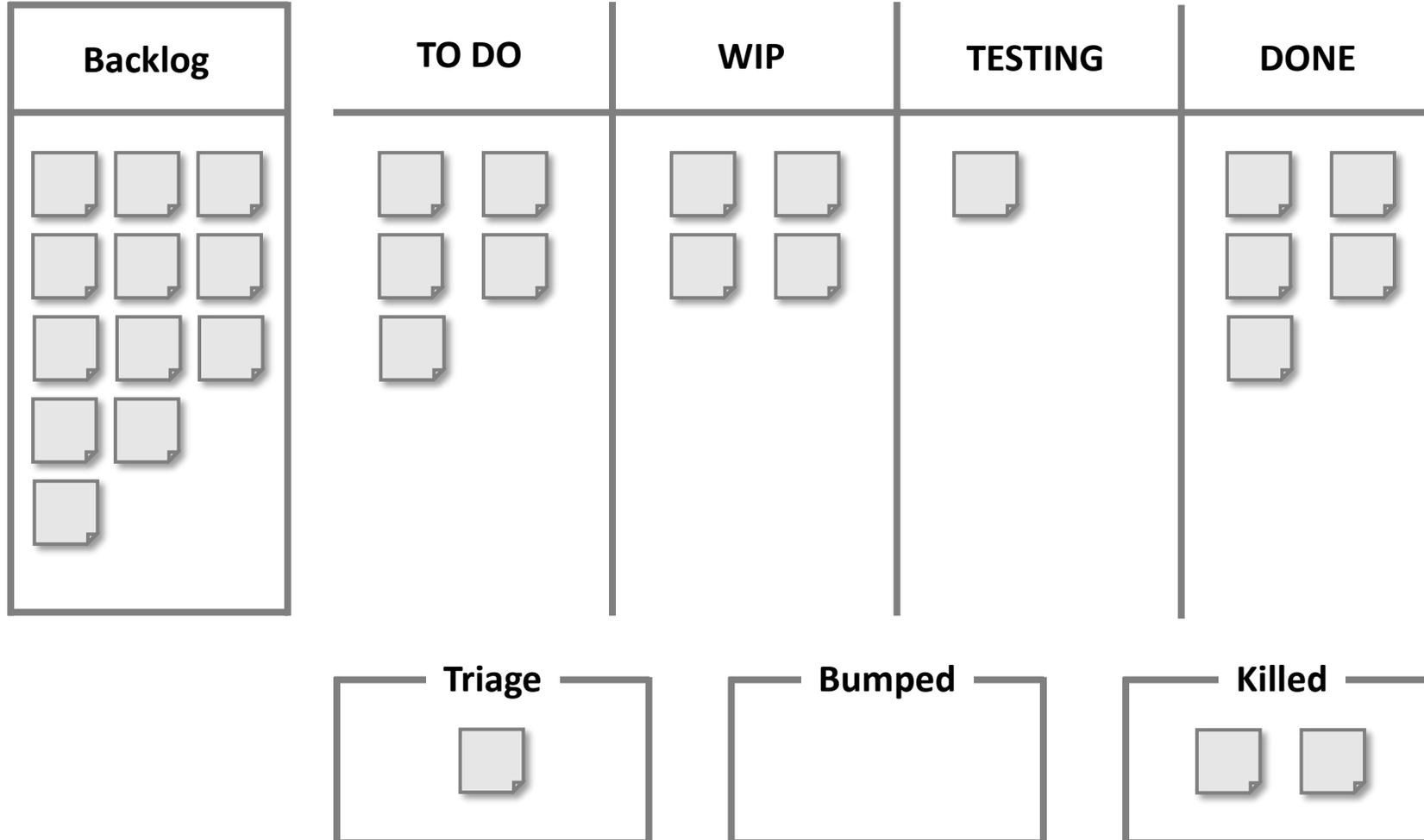


*Planning Poker cards with Fibonacci numbers*

# Scrum Board (Kanban Board)



# Scrum Board (Kanban Board, *variation*)

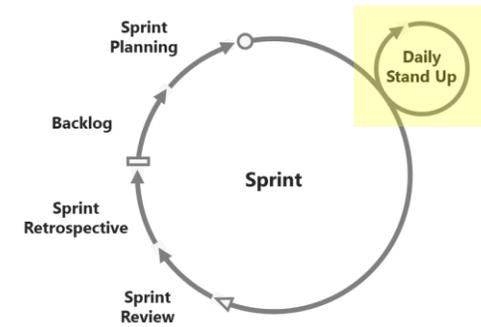


# Scrum – Daily Scrum (a.k.a. daily stand-up)

- **Everyday, same time, 15 minutes**
- **Everyone shares 3 things**

- ✓ What work was done yesterday?
- ✓ What work is planned for today?
- ✓ Any impediments in the way?

*Alternatively, the team can also go through in sequence of the Done and To Do items on the Kanban Board. This can be a better format if multiple team members are involved in the work for each Scrum Board item.*



- **The Scrum Master is not the boss (nor the Product Owner)**

The Scrum Master doesn't give instructions to the Development Team what to do. As a matter of fact, the Daily Scrum is required only for the Development Team. The Scrum Master does not have to facilitate.

- **Not a status update meeting**

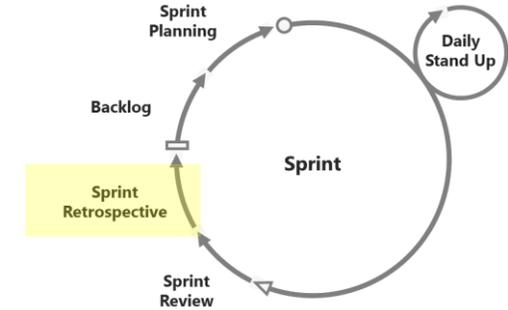
Status updates just give snap shots. What's important is to check-in if all is flowing. If someone is stuck, fellow team members come in to help, and the Scrum Master facilitates removal of the impediment where necessary. Even if everything is going well, there's still room for improvement. Let the team speak up.

## Scrum – Sprint Review



- The end of Sprint “**what**” meeting
- Scrum Team invites **stakeholders**, partners, **customers** to attend
- PO facilitates
- Scrum Team will showcase what was built and released, often as a demo
- PO is responsible for asking real feedback, not just pat on backs
- Feedback is to be incorporated back into product design, and consequently Backlog rebuilding

# Scrum – Sprint Retrospective



- The end of Sprint “**how**” meeting
- Only Scrum Team attends
- SM facilitates
- The purpose of the Sprint Retrospective is to:
  - ✓ *Inspect how the last Sprint went with regards to people, relationships, process, and tools;*
  - ✓ *Identify and order the major items that went well and potential improvements; and,*
  - ✓ *Create a plan for implementing improvements to the way the Scrum Team does its work.*

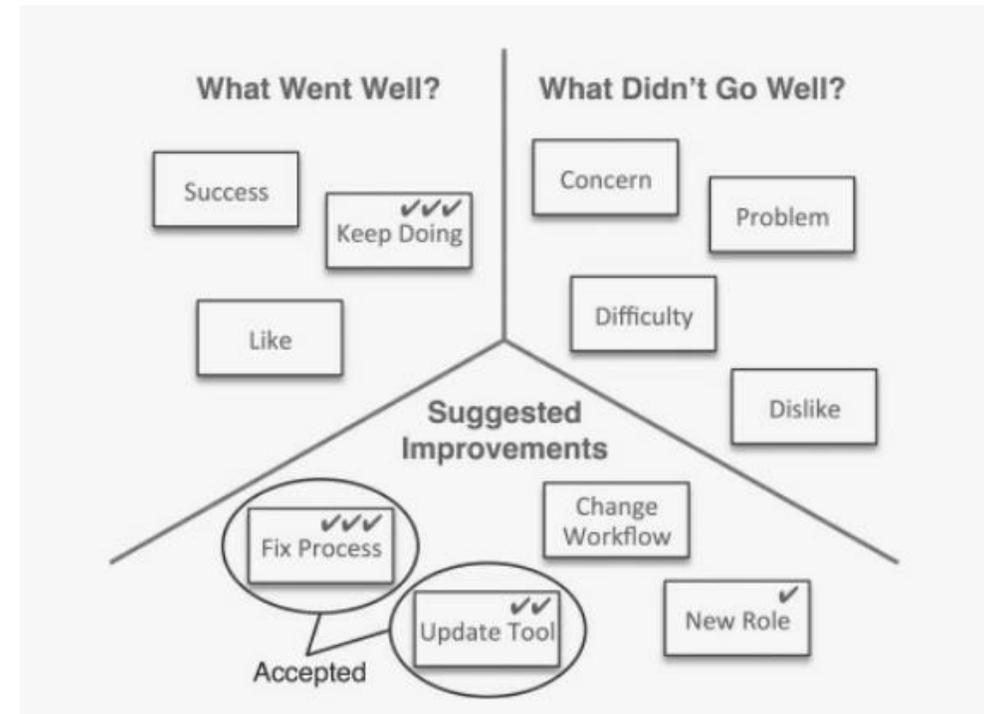


Diagram by Brinker, Scott, Hacking Marketing: Agile Practices to Make Marketing Smarter, Faster, and More Innovative, Wiley, 2016.

# A lot can go wrong with Scrum



- *The Spiritless Scrum: Sprints as Mini-Waterfalls*
- *Territorial Scrum*
- *Scrum Master in Command*
- *Sprint Till You Drop*

<https://agile-od.com/lean-agile/sprint-till-you-drop-and-three-other-tales-of-scrum-gone-wrong>

# Advice: If you're going to do Scrum, do it properly

A lot can go wrong with Scrum

Reference reading (find them on <https://agile-od.com/insight>)

**Single Sprint  
Scrum Pilot**

Test the waters with a *single*  
Sprint *full* Scrum experiment

Images courtesy of sp-studio.de.  
Character concepts by Ron Fingas.

Lifecycle  
agile-od.com

The graphic features the title 'Single Sprint Scrum Pilot' in blue and black text. Below the title is the subtitle 'Test the waters with a single Sprint full Scrum experiment'. In the center, there are six cartoon characters from the South Park series. At the bottom right, there is a logo for 'Lifecycle agile-od.com' with a small gear icon.

Cornerstone Article

A pretty good summary of

# Lean Agile Scrum

The graphic shows a black and white photograph of a library with many bookshelves. Overlaid on the image is the text 'Cornerstone Article' in a white box, followed by 'A pretty good summary of' and the large, bold title 'Lean Agile Scrum'.

## Common Agile **Traps**

# Is Agile a methodology?



methodology



approach, framework,  
modality, mindset, style,  
attitude, a way, spirit,  
**culture**

*Agile is process driven, but not fixed process.*

# The Spirit of Agile



Relentless pursuit of customer value



Failure is welcome



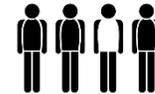
Accept uncertainty



No one gets blamed for trying and failing



Build, Measure, **Learn**



Leaders as facilitators  
Managers as coaches



Incremental iteration



Cross-functional teams



**Trust** in autonomy



Self-organized teams

# Agile and agility

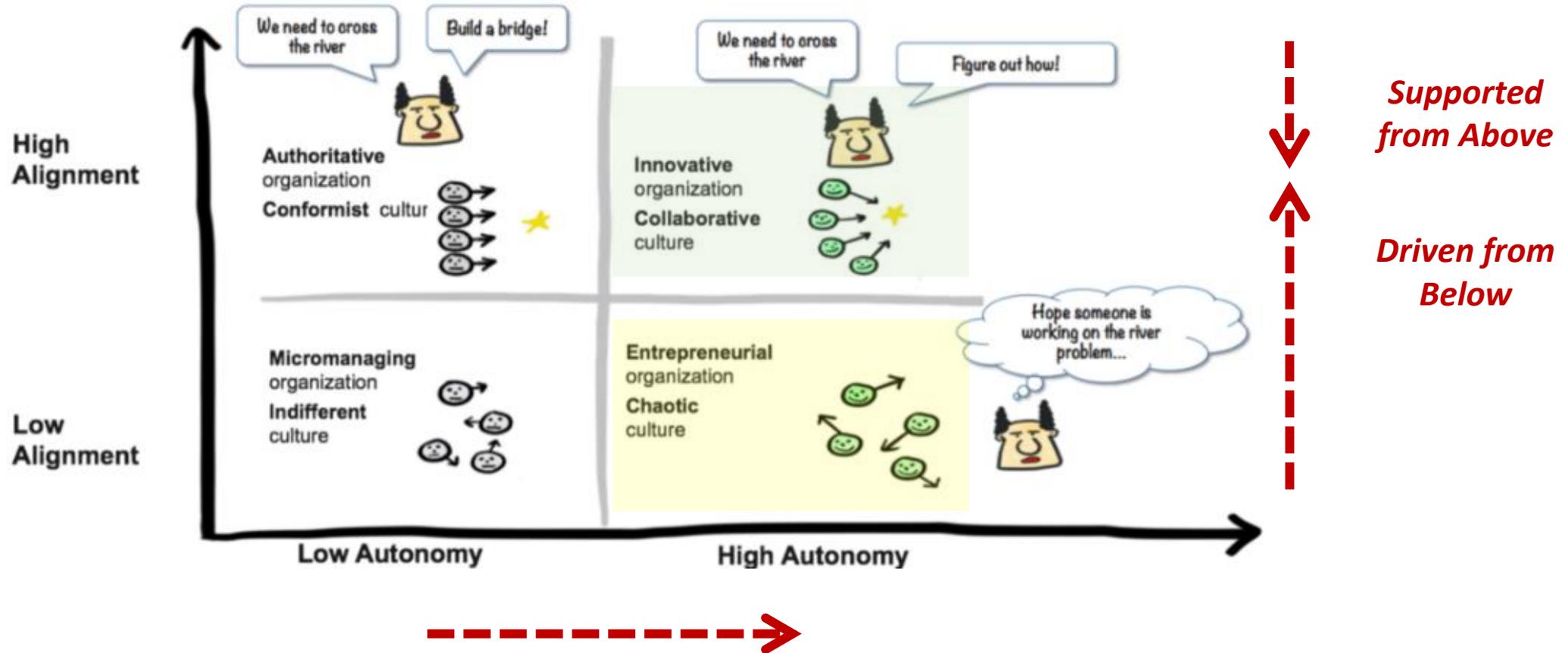
		Agile	agility
Learning	<b>Adapt and iterate</b>	Yes	Yes
Learning	<b>Experiment</b>	Yes	Yes
Learning	<b>Fail fast, learn fast</b>	Yes	Yes
Learning	<b>Perform under uncertainty and ambiguity</b>	Yes	Yes
Org	<b>Stakeholder and customer collaboration</b>	Yes	Encouraged
Org	<b>Cross-functional teams</b>	Yes	Encouraged
Org	<b>Empowered, autonomous, self-organized teams</b> (vs teams with line managers)	Yes	Encouraged
Org	<b>Leaders and managers as coaches and facilitators</b> (vs leaders and managers as decision makers and instruction providers)	Yes	Encouraged
Process	<b>Product development mindset</b> (vs project management)	Yes	Encouraged
Process	<b>Customer/user centricity</b>	Yes	Yes
Process	<b>Process framework driven</b>	Yes	Encouraged
Process	<b>Time-boxed or capacity controled</b>	Yes	Encouraged
Process	<b>Adaptive planning</b> (vs predictive, fixed plans)	Yes	Yes
Process	<b>Continuous improvement</b>	Yes	Yes
Process	<b>Empiricism</b> (evidence, fact, data driven management)	Yes	Yes
Process	<b>Design driven</b>	Encouraged	Encouraged

# Agile and agility

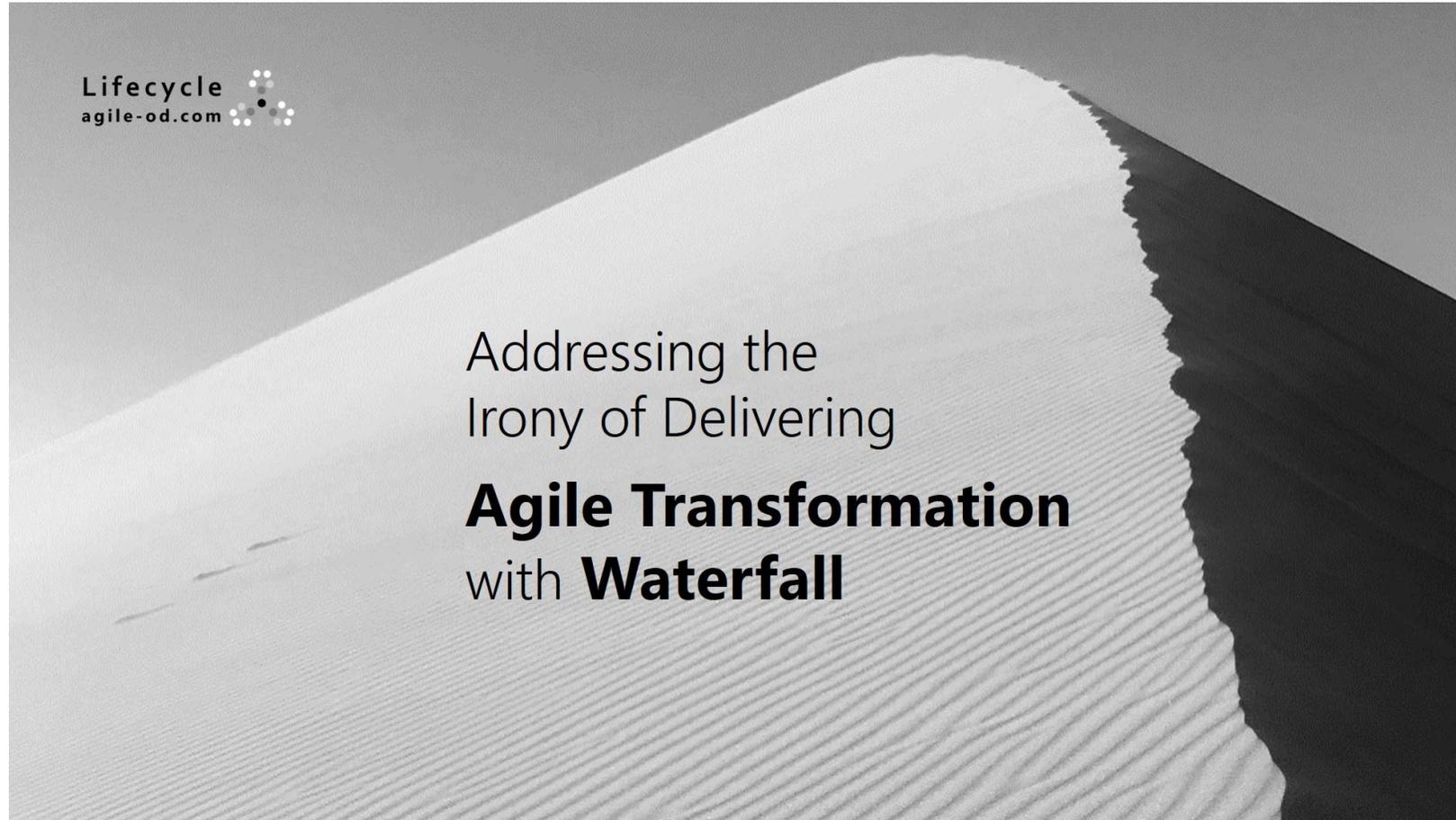
		Agile	agility
Learning	Adapt and iterate	Yes	Yes
Learning	Experiment	Yes	Yes
Learning	Fail fast, learn fast	Yes	Yes
Learning	Perform under uncertainty and ambiguity	Yes	Yes
Process	Customer/user centricity	Yes	Yes
Process	Adaptive planning (vs predictive, fixed plans)	Yes	Yes
Process	Continuous improvement	Yes	Yes
Process	Empiricism (evidence, fact, data driven management)	Yes	Yes
Org	Stakeholder and customer collaboration	Yes	Encouraged
Org	Cross-functional teams	Yes	Encouraged
Org	Empowered, autonomous, self-organized teams (vs teams with line managers)	Yes	Encouraged
Org	Leaders and managers as coaches and facilitators (vs leaders and managers as decision makers and instruction providers)	Yes	Encouraged
Process	Product development mindset (vs project management)	Yes	Encouraged
Process	Process framework driven	Yes	Encouraged
Process	Time-boxed or capacity controled	Yes	Encouraged
Process	Design driven	Encouraged	Encouraged



# Agile Transformation Challenge: Organizational Alignment



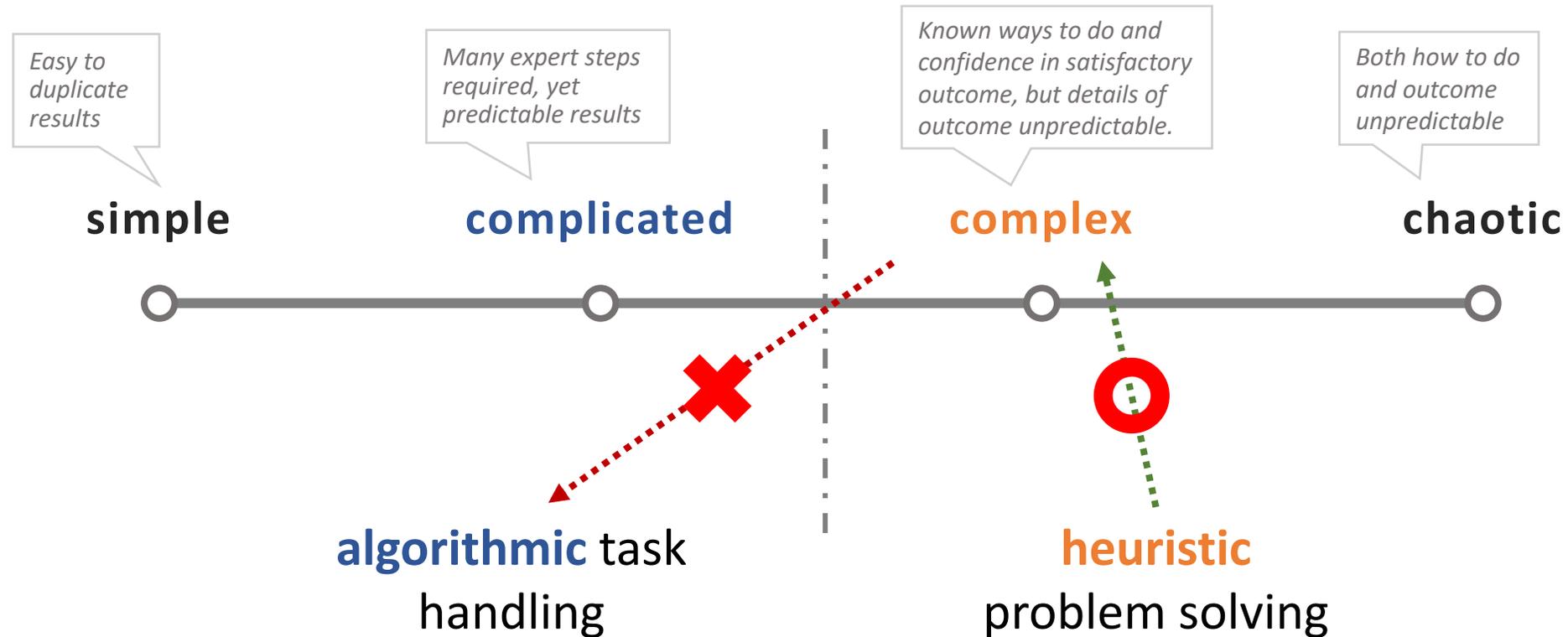
# Waterfall Agile



<https://agile-od.com/lean-agile/waterfall-agile>

## Algorithmic task handling vs **Heuristic** (trial & error) problem solving

We often confuse **Complicated**, with **Complex**



The **crux of the problem** is, we tend to attempt solving **complex** problems, **algorithmically**.

- **Lean**  $\approx$  Agile
- **Design Thinking**  $\in$  Agile

# Lean: Lean Manufacturing & Lean Startup

	Lean Manufacturing	Lean Startup
<b>Objective</b>	How to build better	Discovering what to build
<b>Solution Application</b>	Things previously perceived as deterministic	Uncertainty
<b>Typical user</b>	Large enterprises, mass production units	Startups, entrepreneurial units
<b>Modality</b>	Reduce waste, continuous improvement	Build, measure, learn with MVP, iterate

Common Lean values as generally known today

- **Reduction of waste:** Focus on value creating activity and elimination of anything that doesn't contribute to value
- **Iteration:** Build-measure-learn and continuous improvement
- **Localized activity:** Genchi, Gemba (where the customer is or at the production floor) and allowing product strategy to be determined through local experimentation, and the spirit of "get out of the building"

Popular Lean Manufacturing Terminology

- **Kanban** (かんばん)
- **Muda, Muri, Mura** (むだ、むり、むら): *waste*
- **Genchi-Genbutsu** (現地現物)
- **Gemba** (現場)
- **Kaizen** (改善): *continuous improvement*
- **Kakushin** (革新)

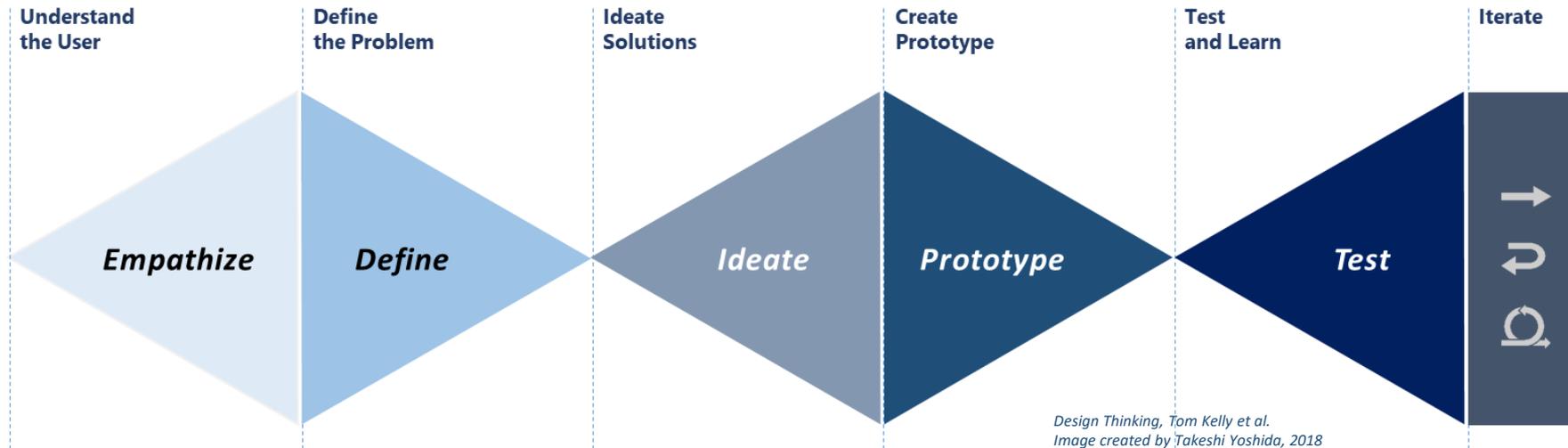
Popular Lean Startup Terminology

- **Build, measure, learn**
- **Persevere, tweak or pivot**

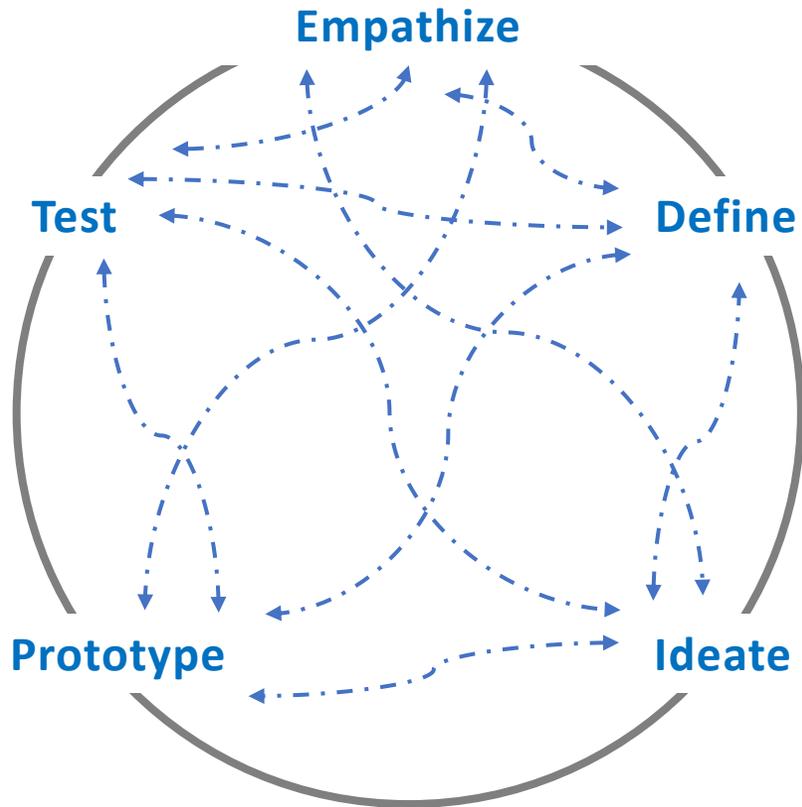
[Read the summary here:](#)



# Design Thinking



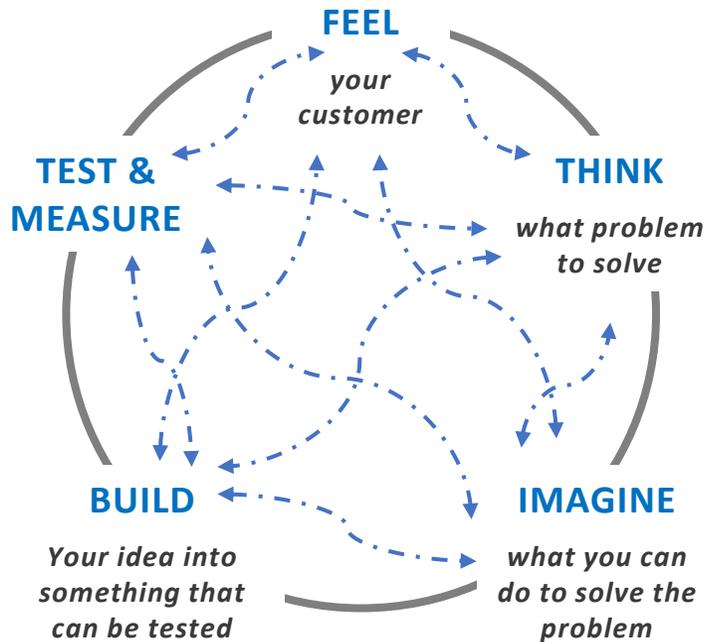
*Design Thinking is process driven*



**Design Thinking**

*Process driven doesn't mean it has to be sequential  
(This is how our brains work)*

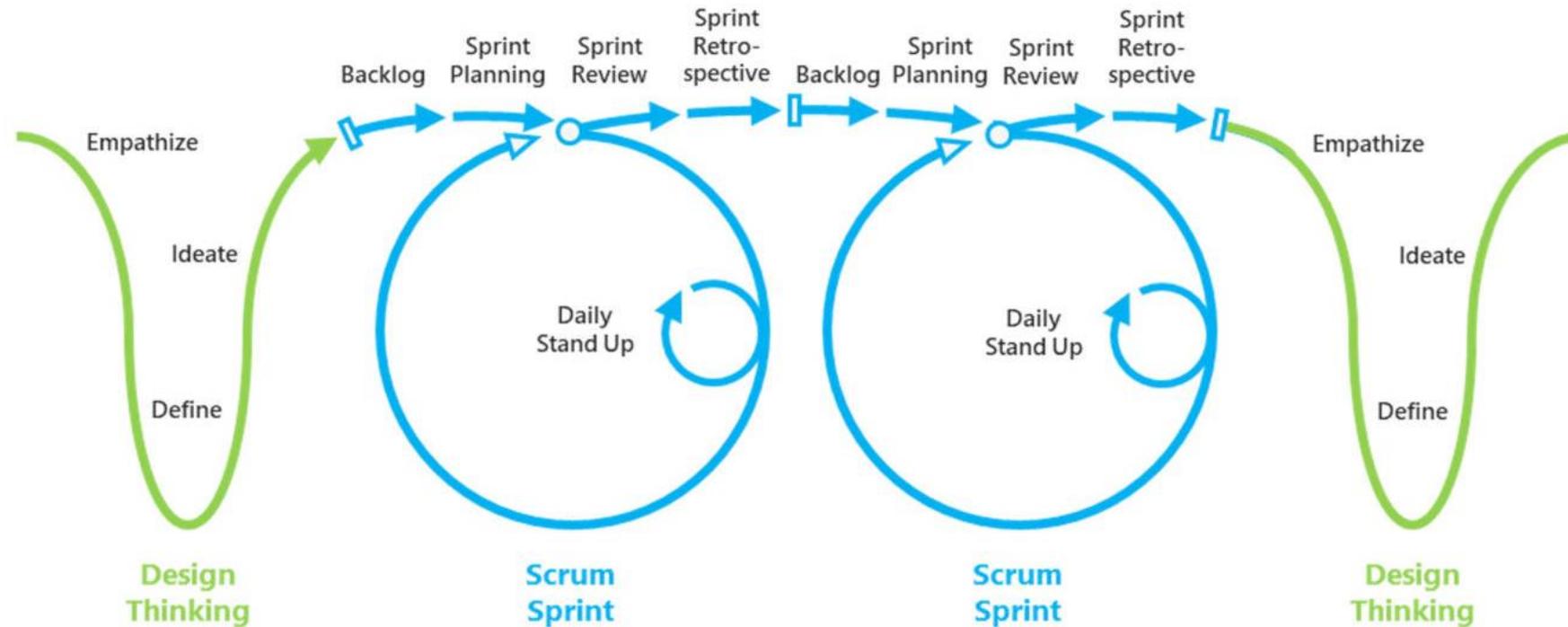
**Creativity in steps**



# Hybrid Agile Example

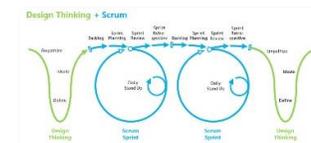
Combining **Design Thinking**'s strength on **ideation**, and **Scrum**'s powerful **iterative building**

## Design Thinking + Scrum



[Read the article here:](#)

**Agile is a high order synthesis of *Flexibility* and *Discipline***



## About Coach Takeshi

# Coach Takeshi

## Takeshi Yoshida

Chief Coach and Founder, Lifecycle Pte. Ltd. (Singapore)

1994~2001: **Morgan Stanley**

2001~2002: **INSEAD MBA**

2002~2004: **Bank of America**

2004~2009: **Deutsche Bank**

2009~2011: **Barclays**

2011~current: **Lifecycle**

- *Serial startup entrepreneur*
- *Coach, trainer, facilitator*
- *INSEAD corporate executive program lecturer, trainer, facilitator*



### Behavioral Coach

- Positive Psychology
- Conversational Intelligence
- Non-Violent Communication
- Radical Candor
- Psychological Safety
- Appreciative Inquiry
- Choice Theory
- Theory X & Y
- Multipliers and Diminishers
- Habit formation

### Process Coach

- Design Thinking
- Agile Scrum
- Lean Startup
- Lean TQM
- OODA
- Liberating Structures
- Lego® Serious Play®
- User Story Mapping
- Design Sprint

### Organization Development Professional

- Systems Theory
- Evidence Based Management (EBM)
- Organizational Ambidexterity
- Theory E & O
- Objectives & Key Results (OKR)
- Innovation Accounting

Full bio: <https://agile-od.com/takeshi>



### Professional Certifications



International Association of Positive Psychology Coaches (IAPPC)  
Certified Positive Psychology Coach (CPPC) Level II  
(Credential: <https://agile-od.com/cppc>)



International Coach Federation (ICF)  
Associate Certified Coach (ACC)  
(Credential: <https://coachfederation.org>)



Scrum.org  
Professional Scrum Master II (PSM II)  
Professional Scrum Product Owner (PSPO)  
(Certifications: <https://www.scrum.org/user/498256>)

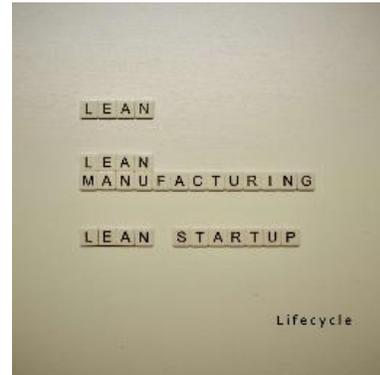


Association of Master Trainers in the LEGO® SERIOUS PLAY® Method  
Certified LEGO® SERIOUS PLAY® Facilitator  
(Certification: <https://agile-od.com/lsp>)

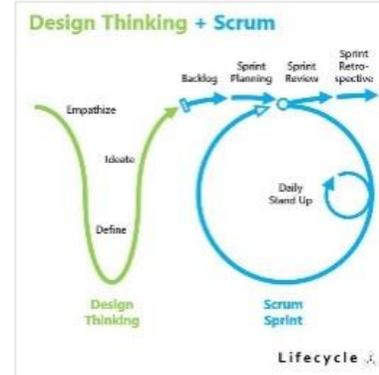
# More readings from Coach Takeshi



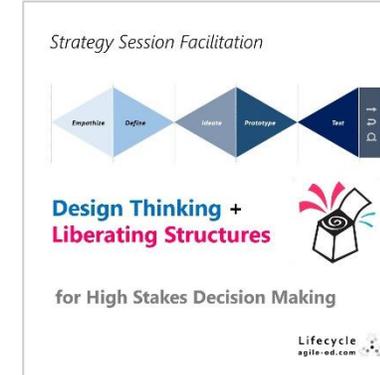
[A Pretty Good Summary of Lean, Agile, Scrum](#)



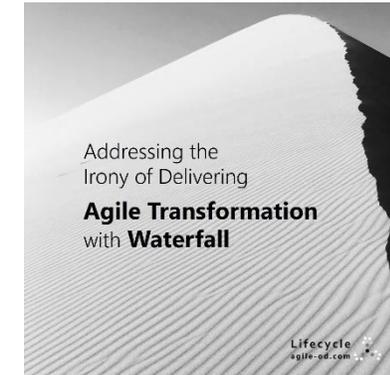
[Lean, Lean Manufacturing, Lean Startup: Explained](#)



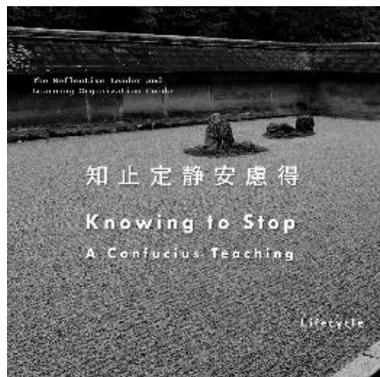
[Try Design Thinking + Scrum: A Powerful Hybrid Agile Approach](#)



[Strategy Session Facilitation with Design Thinking + Liberating Structures](#)



[Waterfall Agile: Addressing the Irony of Delivering Agile Transformation with Waterfall](#)



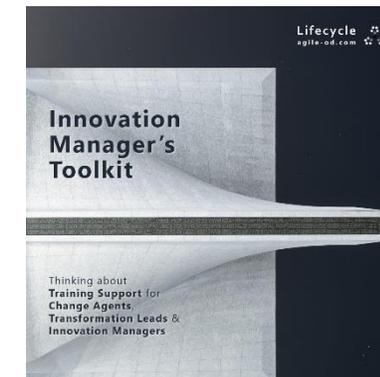
[Knowing to Stop, a Confucius Teaching](#)



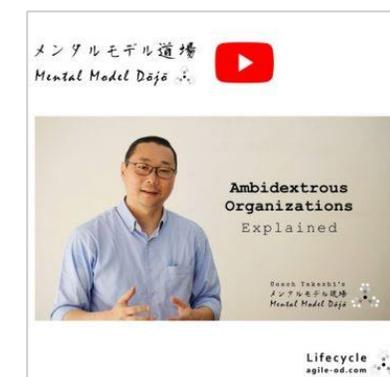
[Radical Candor, My Go To Feedback Routine](#)



[How to Get Scrum Right on First Attempt: Single Sprint Scrum Pilot](#)

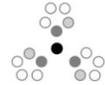


[Innovation Manager's Toolkit](#)



[Ambidextrous Organizations Explained](#)

**Lifecycle**  
agile-od.com



Lifecycle Pte. Ltd.  
9 Jalan Membina 23-02  
Singapore 169483

**Takeshi Yoshida**

Founder & Chief Coach

takeshi@agile-od.com  
<https://agile-od.com>  
+65-9234-9727