HUMAXVINA

SOFTWARE DEVELOPMENT LIFE CYCLE

Embedded Fresher Training – June 12th, 2022

Contents

01 Introduction to Project & SDLC

02 SDLC Model

03 Agile Software Development

04 Agile Practices

Introduction to Project & SDLC

What is Project?

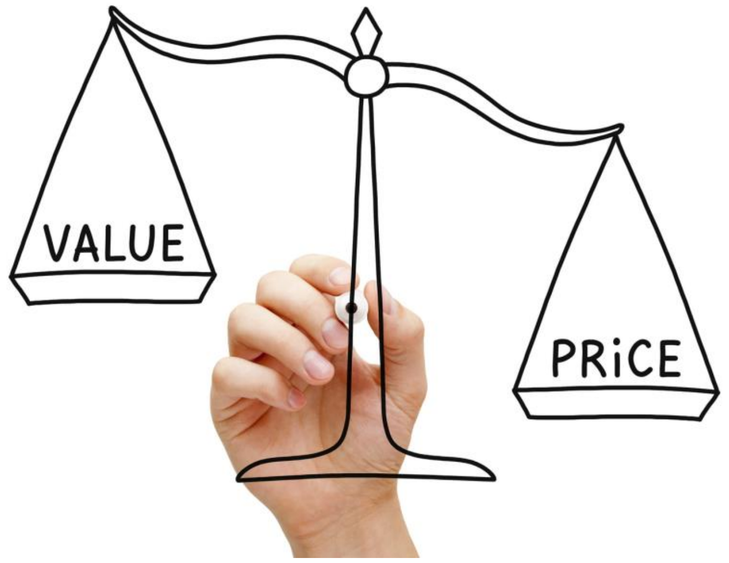
Definition:
- Is a **temporary endeavor** with a beginning and an end.
- Creates a **unique product, service, or result**.

Example of projects:
- Developing a new product or service
- Build a new house
- Implement a core network

Why Organizations Initiate A Project?

Value

- The worth, importance, or usefulness of something.
- Different stakeholders perceive value in different ways. Customers can define value as the ability to use specific features or functions of a product.

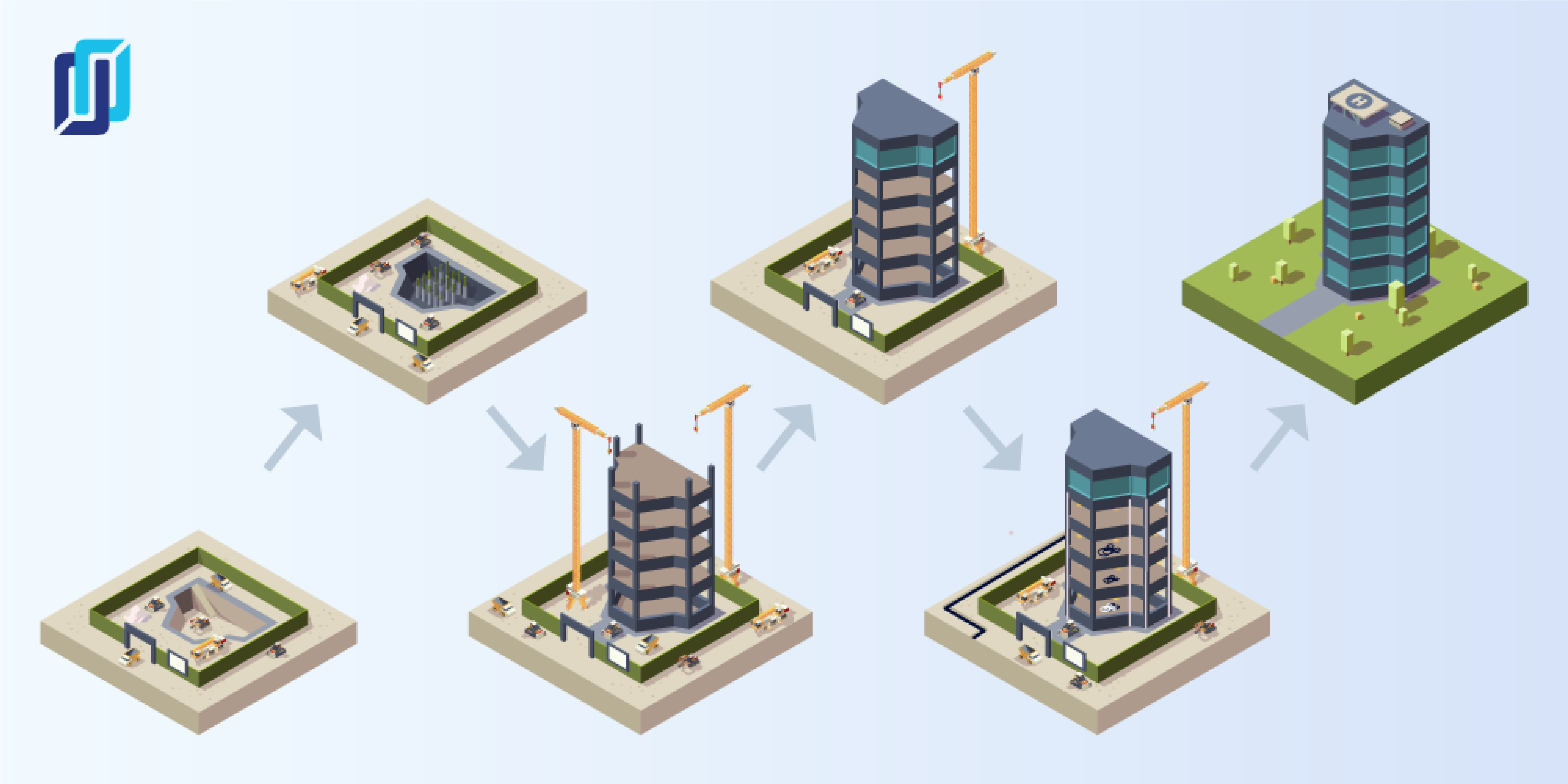


- Business value: Organizations can focus on business value as determined with financial metrics, such as the benefits less the cost of achieving those benefits.
- Societal value: can include the contribution to groups of people, communities, or the environment.

Project Phase

What is it?

- A collection of logically related project activities that culminates in the completion of **one or more deliverables**.
- The transition from one phase to another within a project's lifecycle generally involves some form of technical transfer or handoff.



Construction Project: *Feasibility-> Planning -> Design -> Production -> Turnover -> Maintenance*

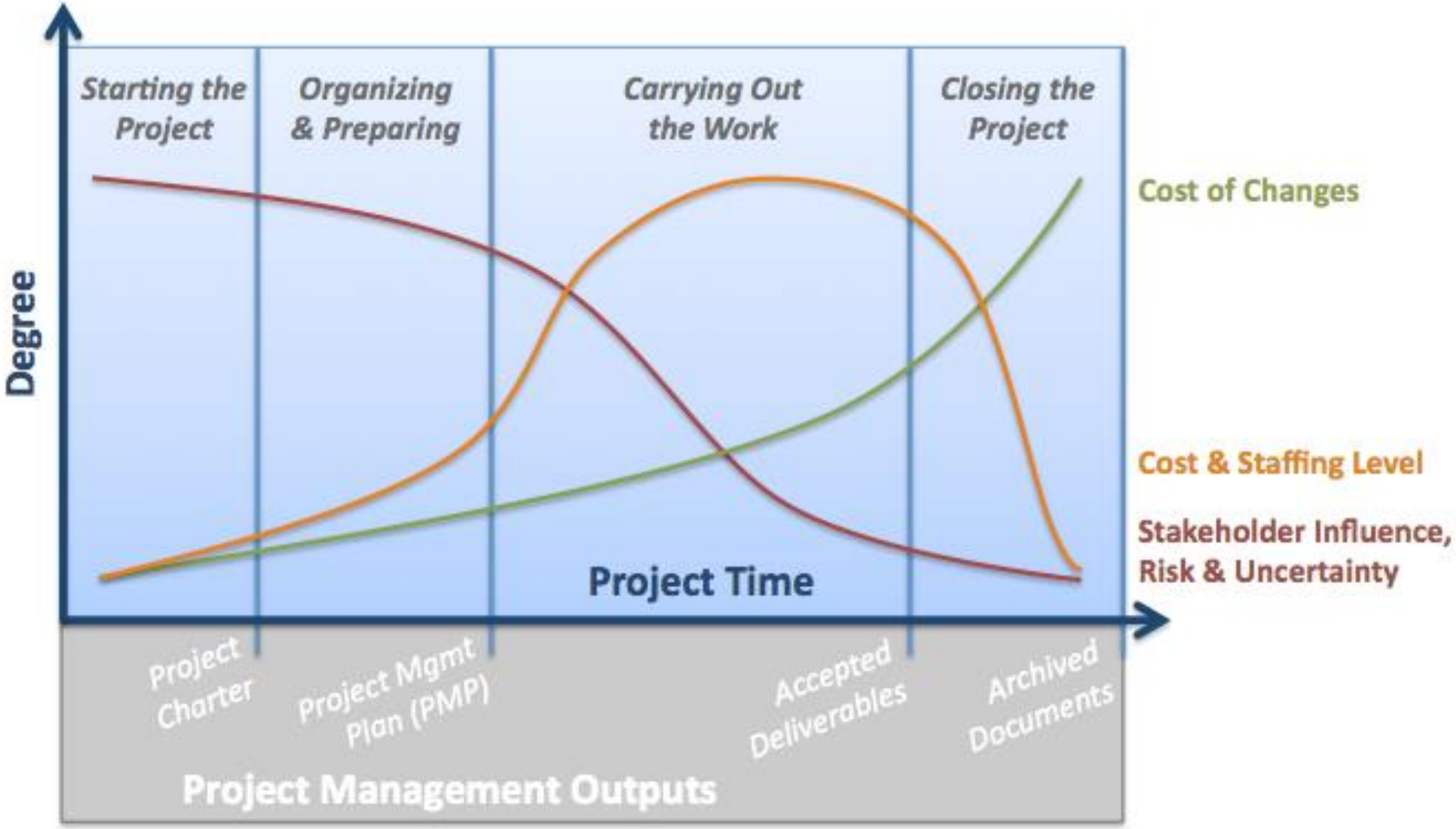
Project Life Cycle

What is it?

- The **series of phases** that a project passes through from its initiation to its closure.
- There are many different types of project life cycles, depending on the industry, or on the **organization's preferences**.

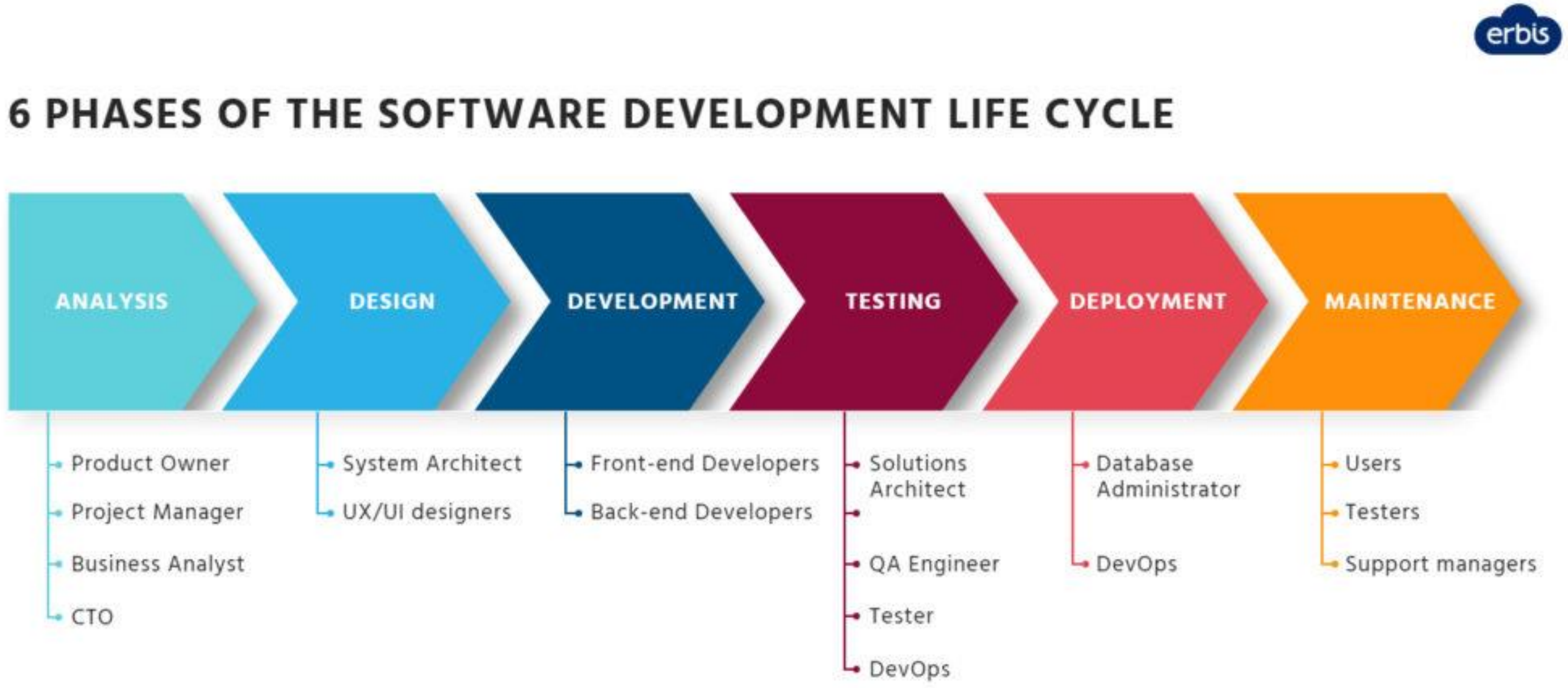
PROJECT LIFE CYCLE PHASES
Phase-1
Phase-2
Phase-3
Phase-4
Phase-5
Phase-6

Characteristics of Project Life Cycle



Software Development Life Cycle

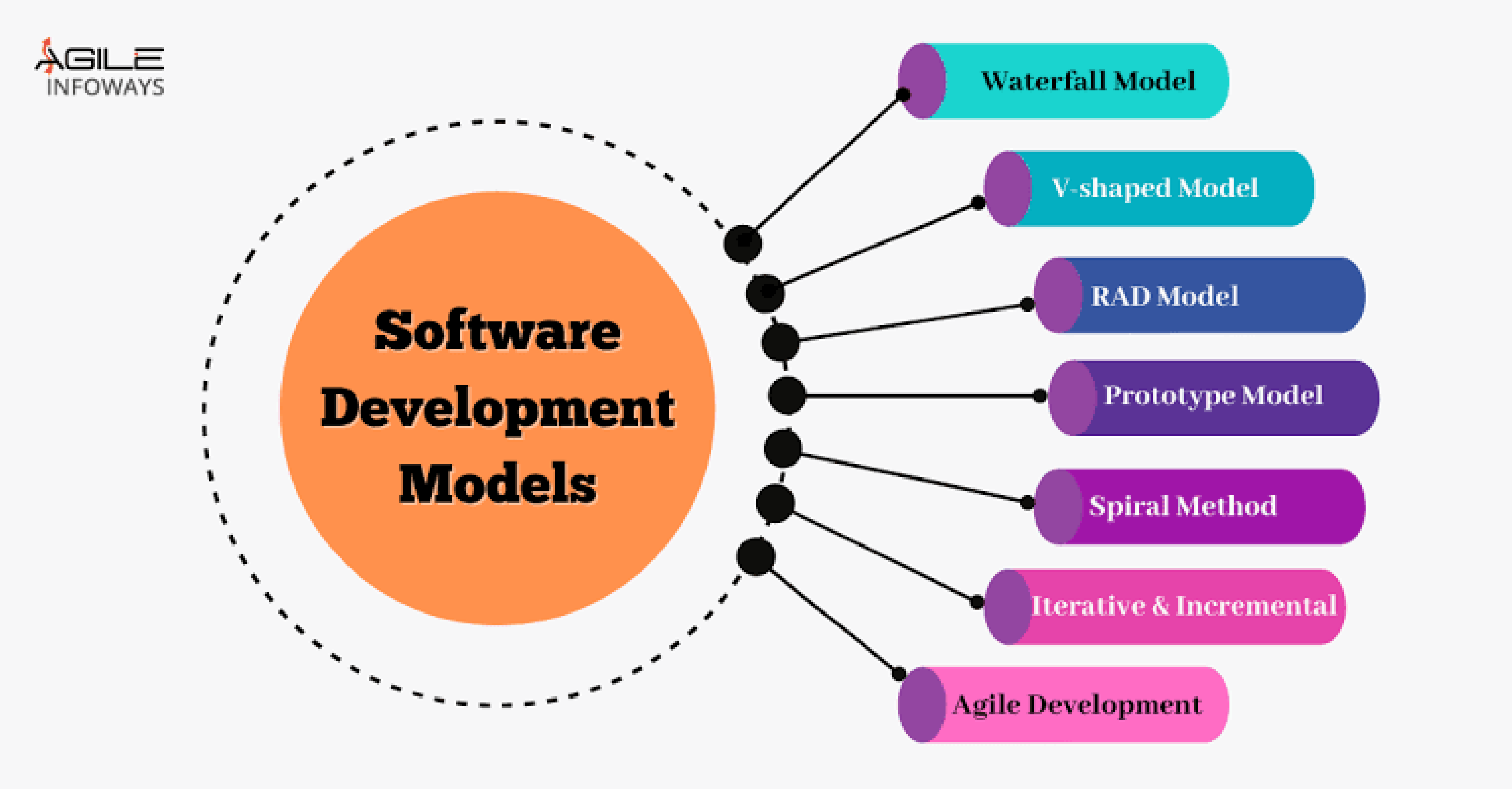
The systems (software) development life cycle (SDLC) is a process for planning, creating, testing, and deploying an information system.



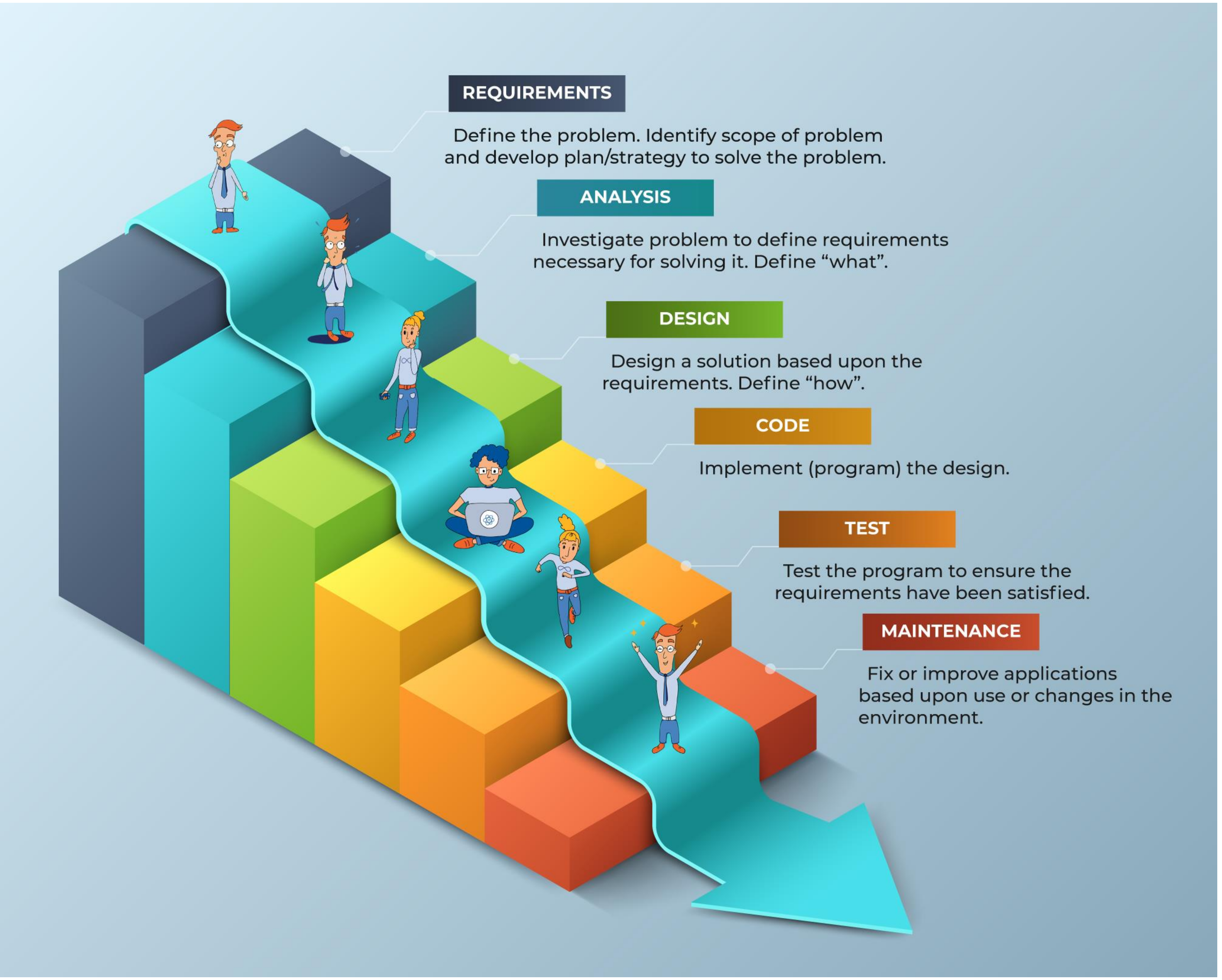
SDLC can be considered as a subset of PLC. While PLC focuses on the project management phases, SDLC focuses on the software engineering phases for building an IT solution.

SDLC Model

Software Development Models



Waterfall Model



REQUIREMENTS

Define the problem. Identify scope of problem and develop plan/strategy to solve the problem.

ANALYSIS

Investigate problem to define requirements necessary for solving it. Define “what”.

DESIGN

Design a solution based upon the requirements. Define “how”.

CODE

Implement (program) the design.

TEST

Test the program to ensure the requirements have been satisfied.

MAINTENANCE

Fix or improve applications based upon use or changes in the environment.

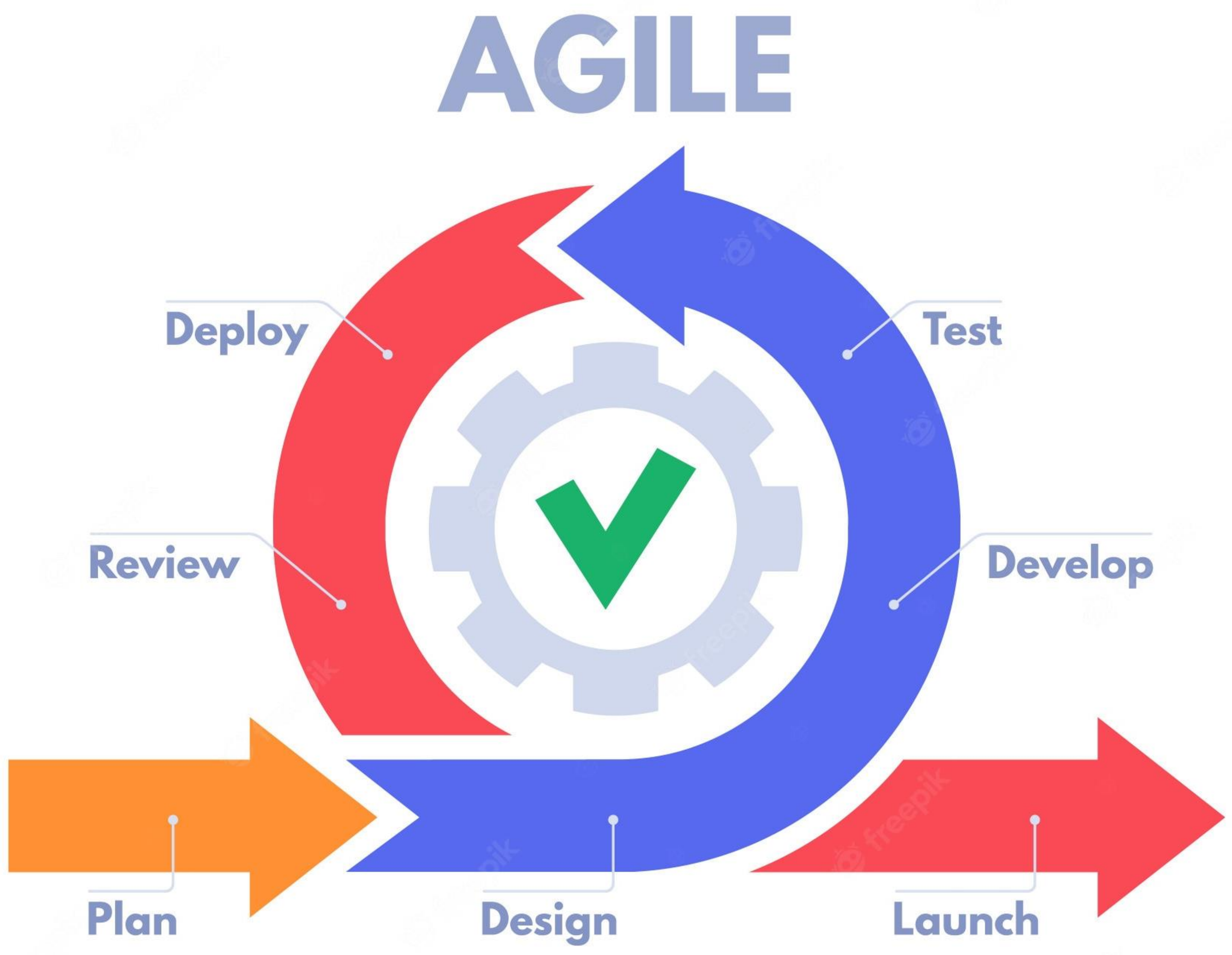
- The waterfall model is a breakdown of project activities into linear sequential phases, where each phase depends on the deliverables of the previous one and corresponds to a specialization of tasks.

- In software development, it tends to be among the less iterative and flexible approaches, as progress flows in largely one direction ("downwards" like a waterfall).

Agile Software Development

- Popularized in the 2001 Manifesto, Agile is an iterative approach to project management and software development that helps teams deliver value to their customers faster and with fewer headaches.

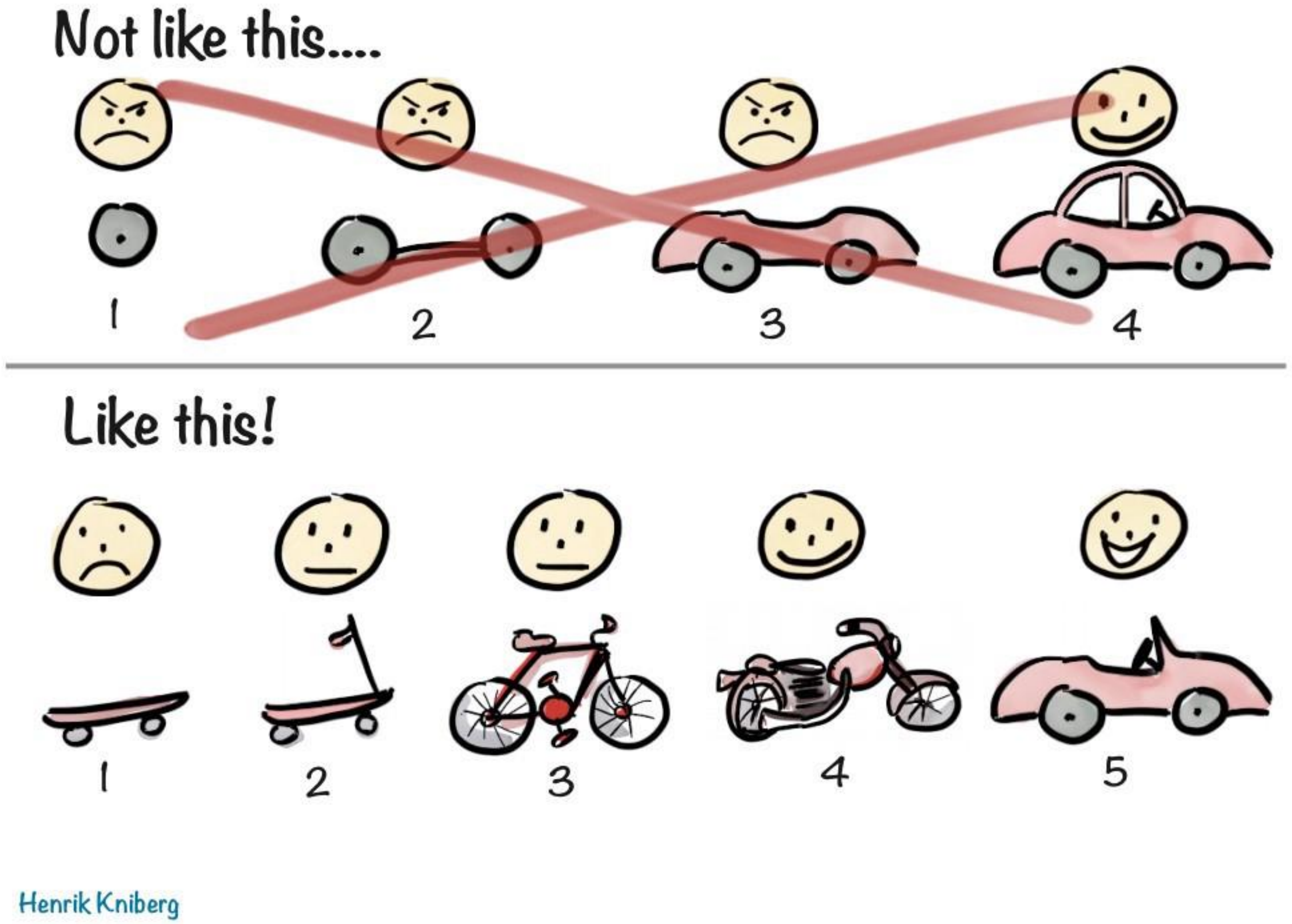
- Instead of betting everything on a "big bang" launch, an agile team delivers work in small, but consumable, increments.



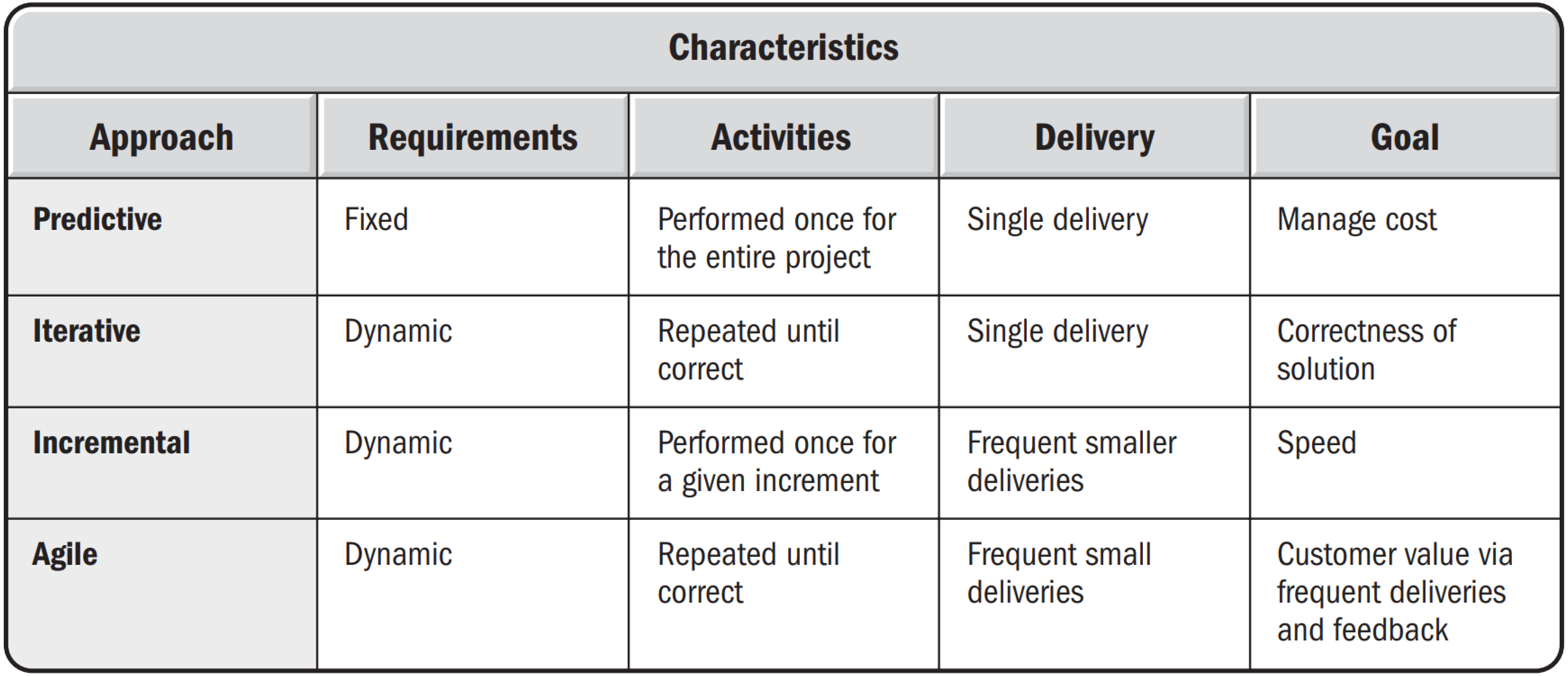
Waterfall vs Agile

Waterfall
(Predictive)

Agile
(Adaptive)



Characteristics of Different Life Cycles



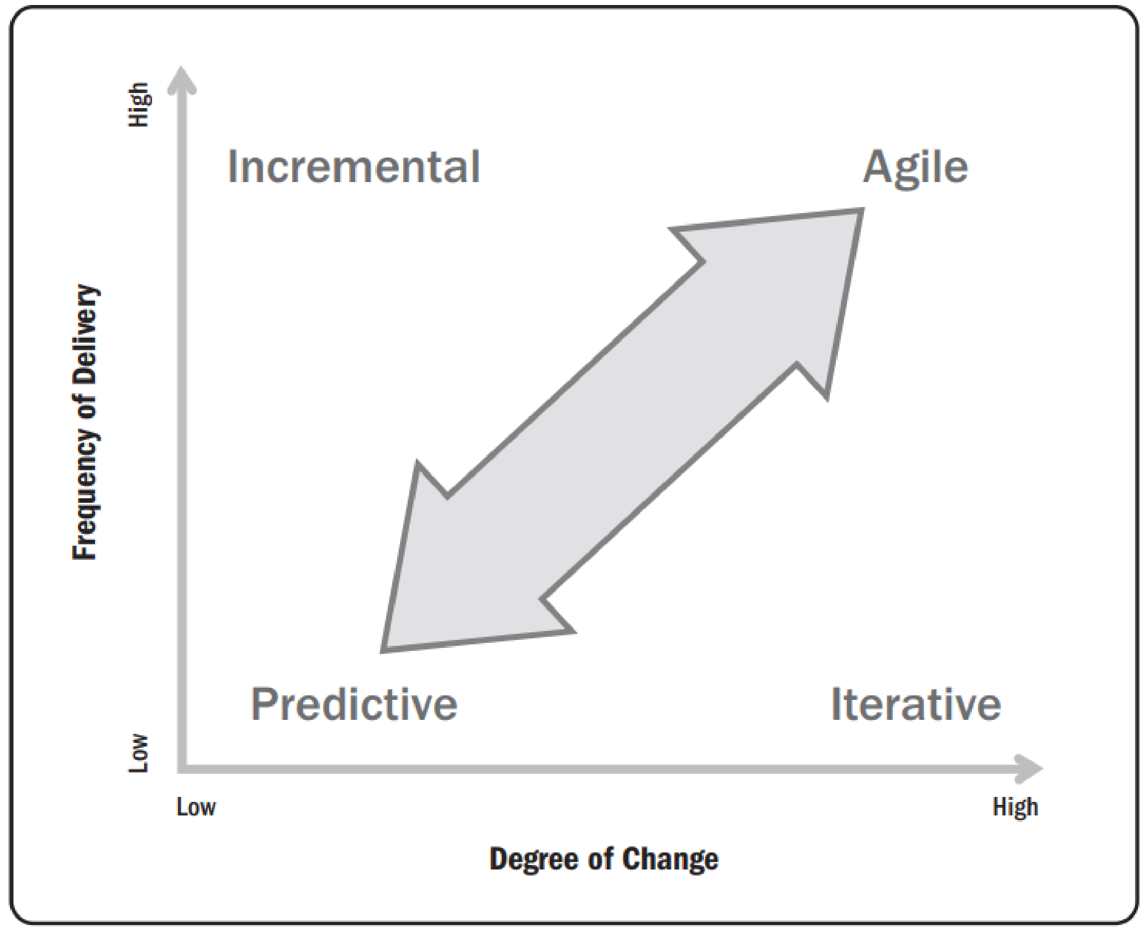
Characteristics				
Approach	**Requirements**	**Activities**	**Delivery**	**Goal**
Predictive	Fixed	Performed once for the entire project	Single delivery	Manage cost
Iterative	Dynamic	Repeated until correct	Single delivery	Correctness of solution
Incremental	Dynamic	Performed once for a given increment	Frequent smaller deliveries	Speed
Agile	Dynamic	Repeated until correct	Frequent small deliveries	Customer value via frequent deliveries and feedback

Hybrid Life Cycle As Fit-for-purpose

The goal of SDLC is to produce business value in the best possible way given the current environment. It does not matter if that way is agile or predictive.

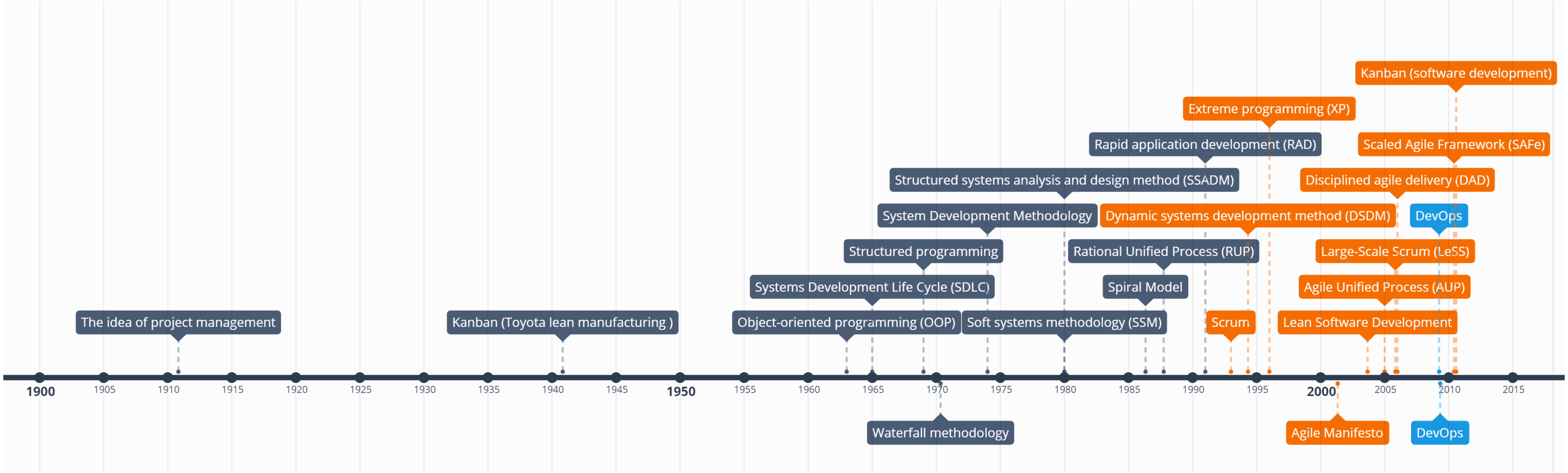
The question to ask is: “**How can we be the most successful?**”

Predictive (Waterfall)	Adaptive (Agile)
Process-based	Principle-based
Definable work	High-uncertainty work
Plan-driven	Value-driven
Prescriptive	Descriptive



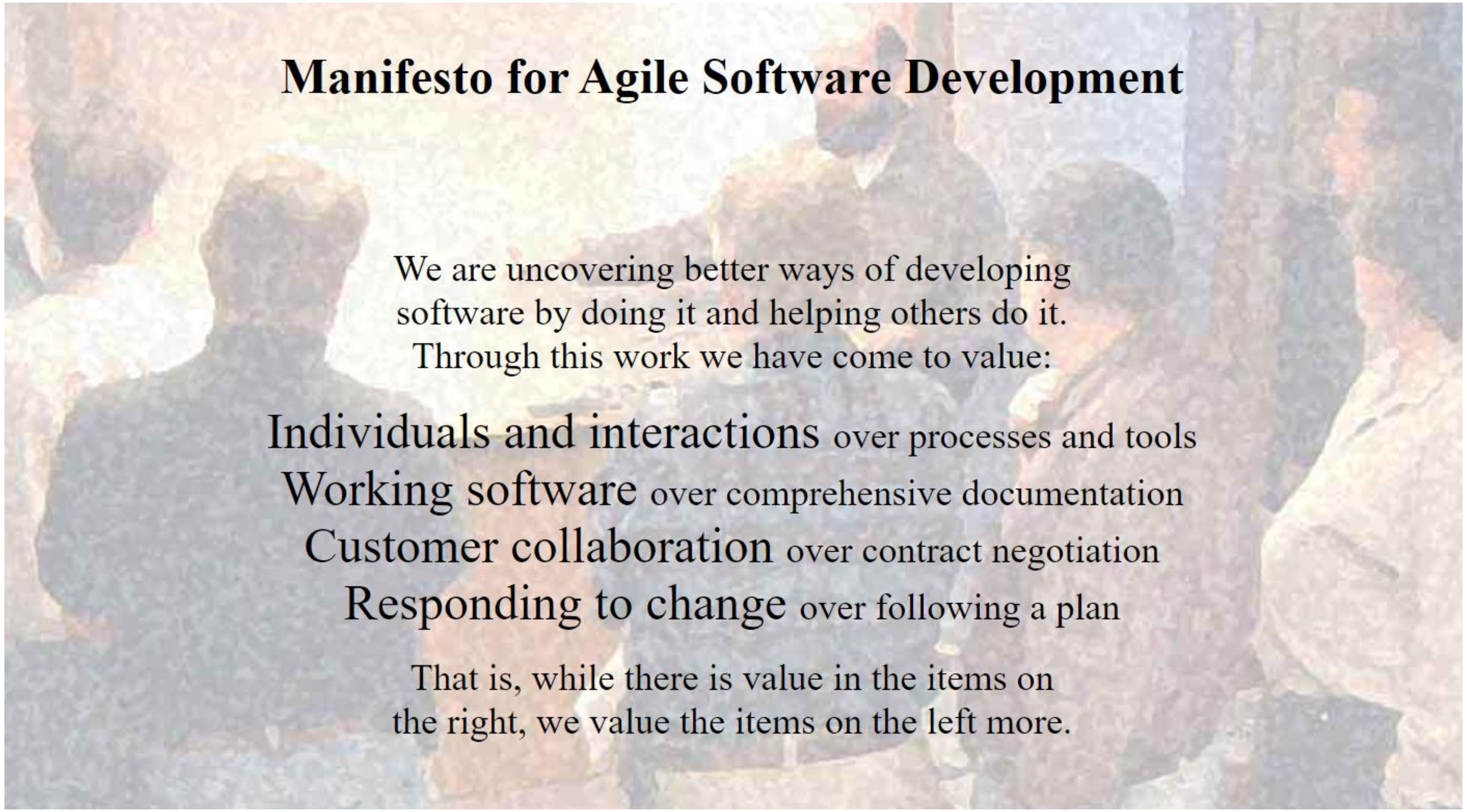
Agile Software Development

Software Development Process History


Kanban (software development)
Extreme programming (XP)
Rapid application development (RAD)
Scaled Agile Framework (SAFe)
Structured systems analysis and design method (SSADM)
Disciplined agile delivery (DAD)
System Development Methodology
Dynamic systems development method (DSDM)
DevOps
Structured programming
Rational Unified Process (RUP)
Large-Scale Scrum (LeSS)
Systems Development Life Cycle (SDLC)
Spiral Model
Agile Unified Process (AUP)
The idea of project management
Kanban (Toyota lean manufacturing)
Object-oriented programming (OOP)
Soft systems methodology (SSM)
Scrum
Lean Software Development
1900
1905
1910
1915
1920
1925
1930
1935
1940
1945
1950
1955
1960
1965
1970
1975
1980
1985
1990
1995
2000
2005
2010
2015
Waterfall methodology
Agile Manifesto
DevOps

Confidential

Agile Manifesto



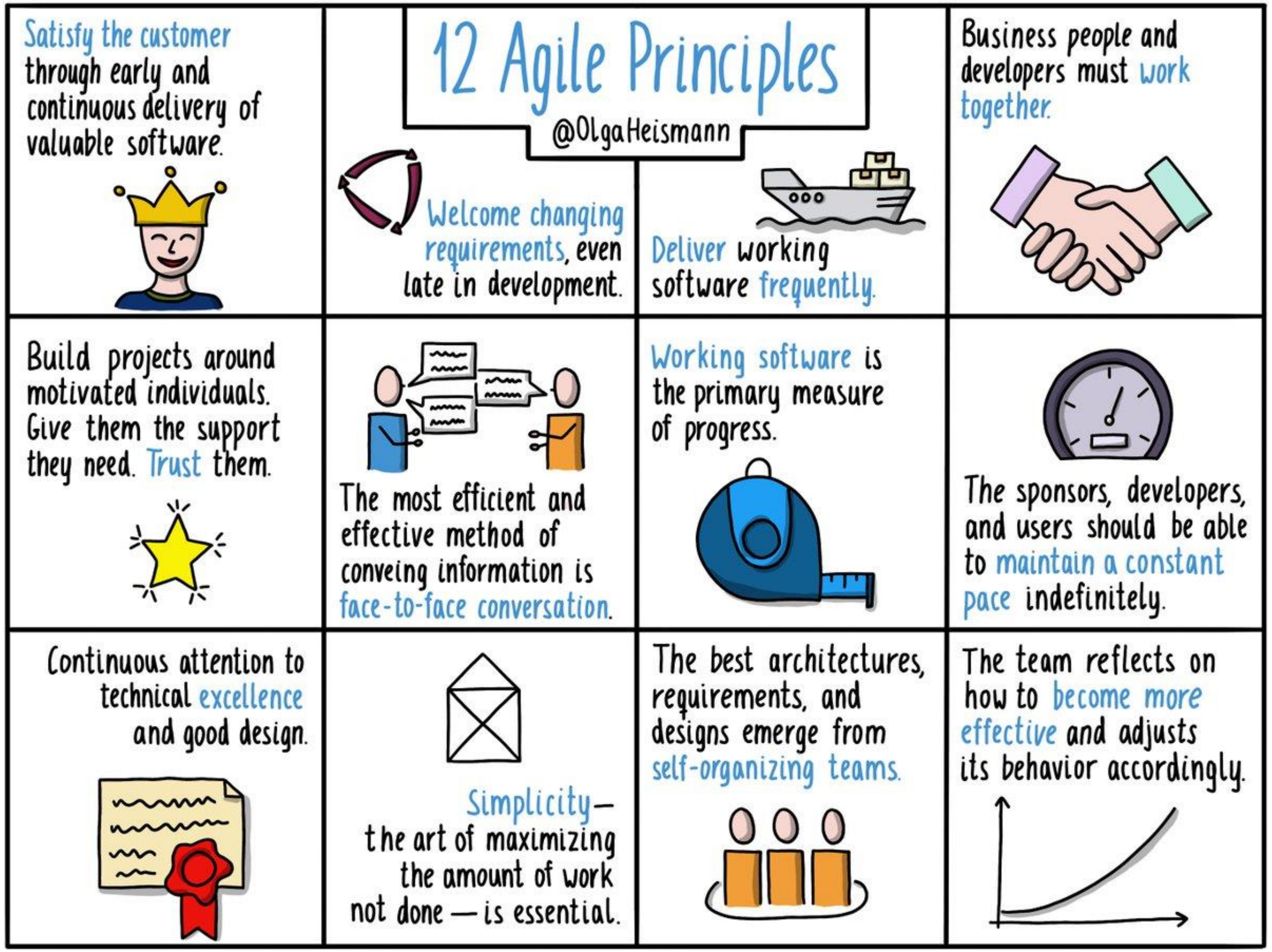
Manifesto for Agile Software Development

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 Principles



12 Agile Principles

@OlgaHeismann

Satisfy the customer through early and continuous delivery of valuable software.

Welcome changing requirements, even late in development.

Deliver working software frequently.

Business people and developers must work together.

Build projects around motivated individuals. Give them the support they need. Trust them.

The most efficient and effective method of conveing information is face-to-face conversation.

Working software is the primary measure of progress.

The sponsors, developers, and users should be able to maintain a constant pace indefinitely.

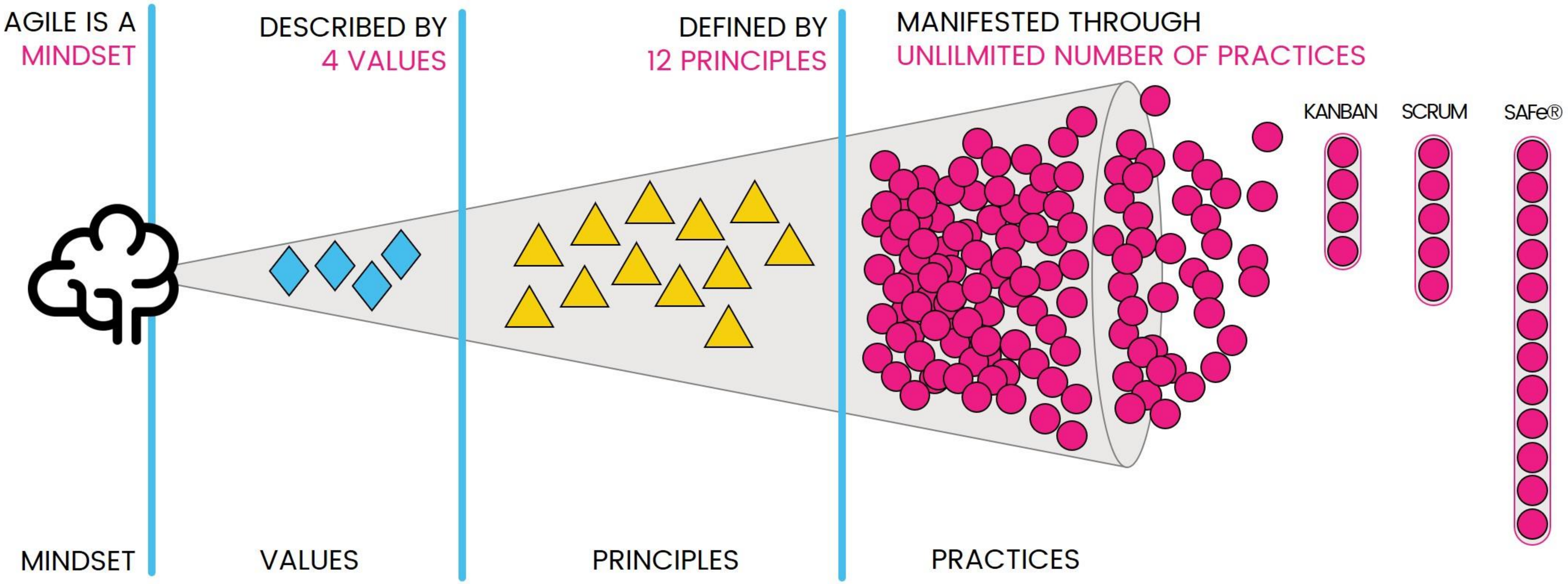
Continuous attention to technical excellence and good design.

Simplicity— the art of maximizing the amount of work not done — is essential.

The best architectures, requirements, and designs emerge from self-organizing teams.

The team reflects on how to become more effective and adjusts its behavior accordingly.

Agile is a Mindset



AGILE IS A MINDSET

DESCRIBED BY 4 VALUES

DEFINED BY 12 PRINCIPLES

MANIFESTED THROUGH UNLILMITED NUMBER OF PRACTICES

Adapted from Ahmed Sidky's Agile Mindset

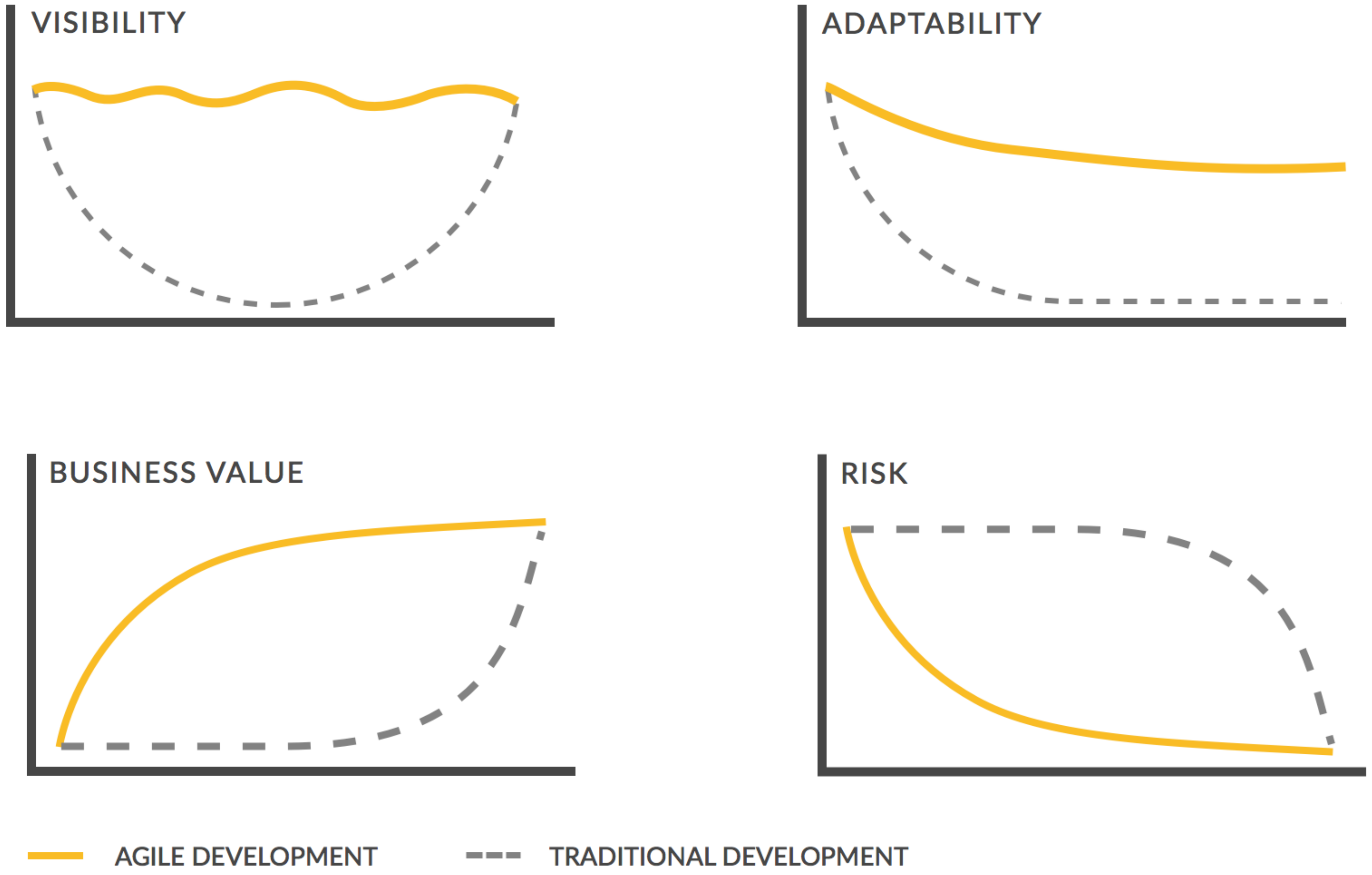
Value-Driven Delivery

- Deliver Value Early:

- Deliver the highest-value portions of the project **as soon as possible**
- Increase stakeholder satisfaction

- Minimize:

- Waste
- Non-value-adding activities



Common Misunderstandings

- ***Agile has no Plan?***

- No, it has Plan. But the plan accepts and adapts with change.

- ***Agile has no Document?***

- No, it has Document. But it will do the document at the last responsibility moment.

- ***Agile has no Process?***

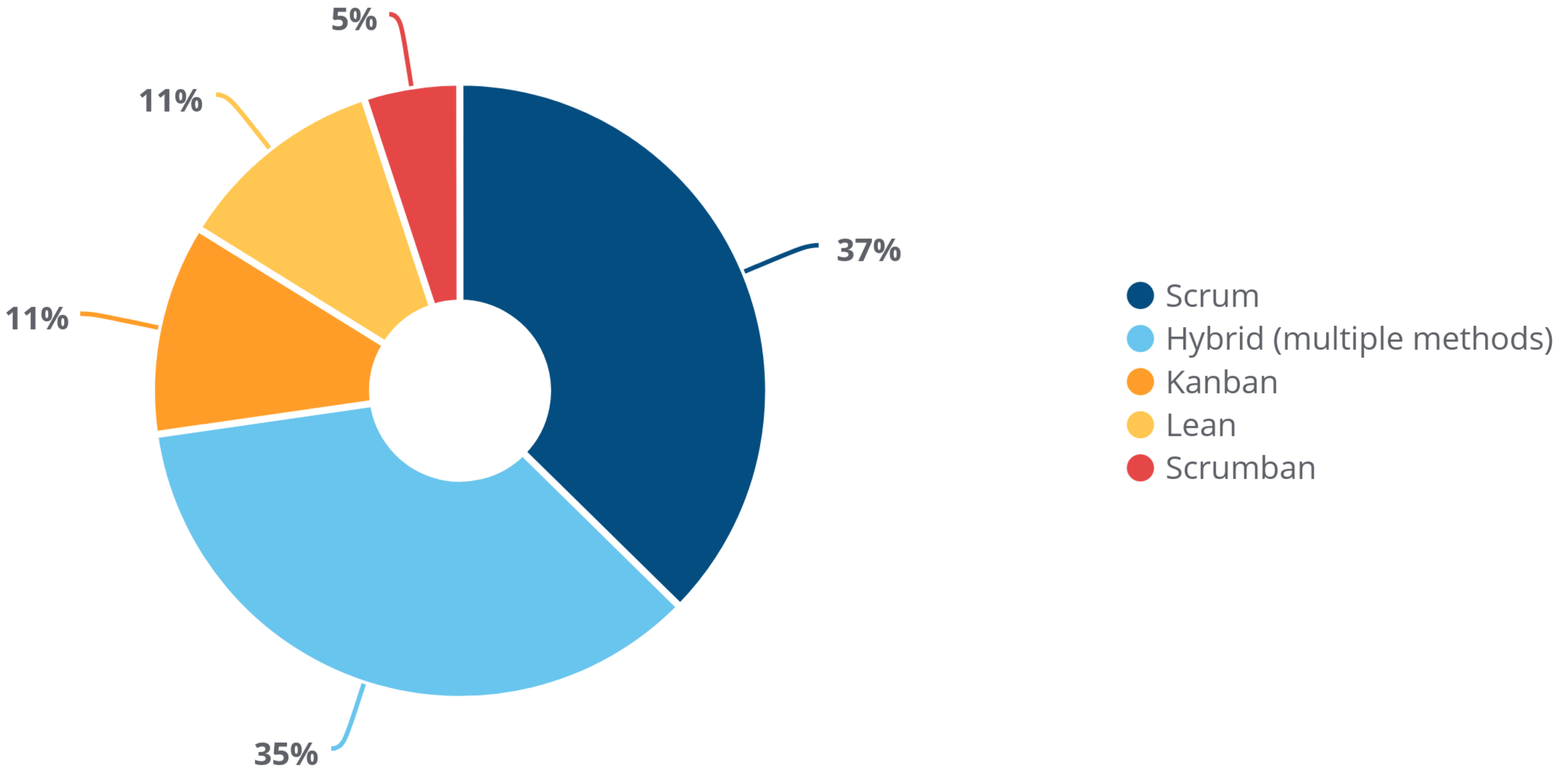
- No, it has Process. But it accepts that process will be defined by team.

- ***Agile have no Contract?***

- No, it has Contract. But the contract allows change and accepts agile method from both sides.

Agile Practices

Agile Methods Used



Q: What Agile method do you use?
N = 303
Source: Capterra 2019 Agile Project Management Software User Survey

Capterra

Confidential

Kanban – A Visual Management Method

In 1940, Toyota Production System under the leadership of Taichii Ohno came up with better engineering process for “Just In Time” delivery process called “KANBAN”.

Kanban is a Japanese term meaning **visual signal** or **card**. Toyota engineers use the same for actual manufacturing work.

Kanban Board

To do
In progress
Done

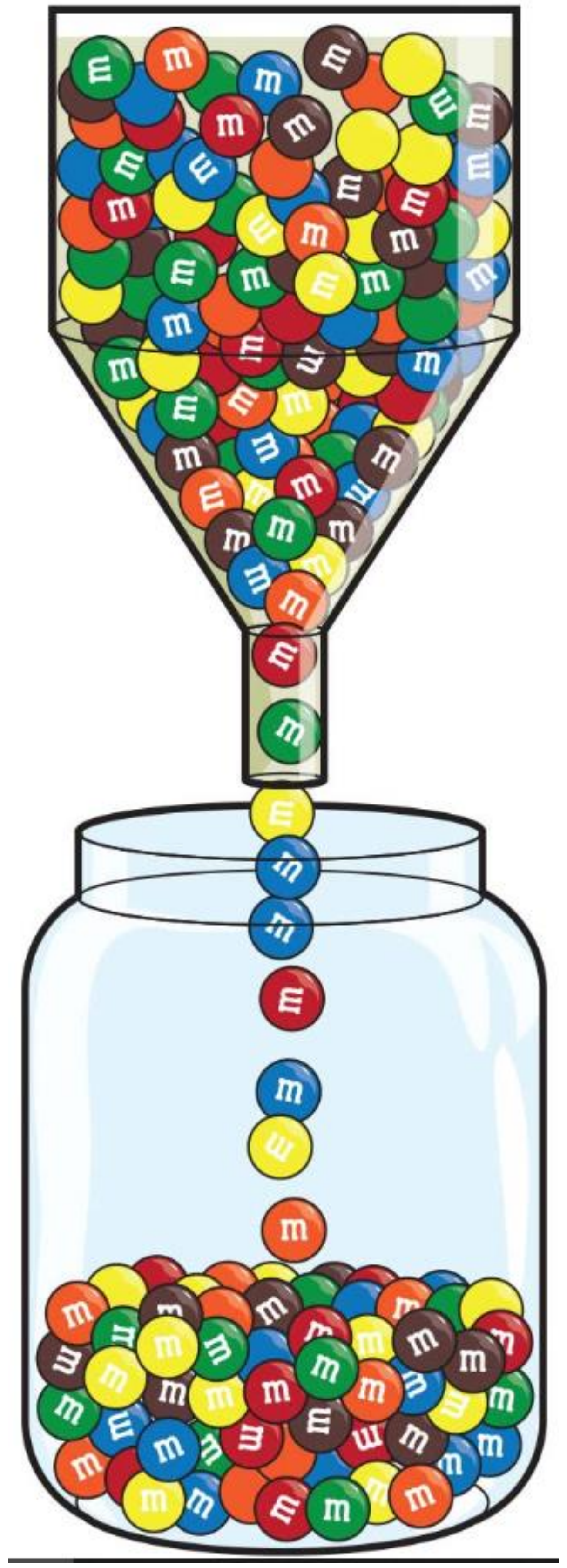
Visualize work:

- Transparency
- Explicit Policies
- Information Radiator

Theory of Constraint & WIP Limit

Work in Progress (WIP):
- Work has been started but not yet been completed
- Bottlenecks in process, deliver no return of investment
- Risk in form of potential rework, since there may still be changes to items until those items have been accepted

WIP Limit:
Prevent team to take too many different work all at once
- Use Kanban board and restrict the amount of work in the system
• Help to identify and remove bottlenecks
• Reduce the risk of tied-up capital, rework and waste



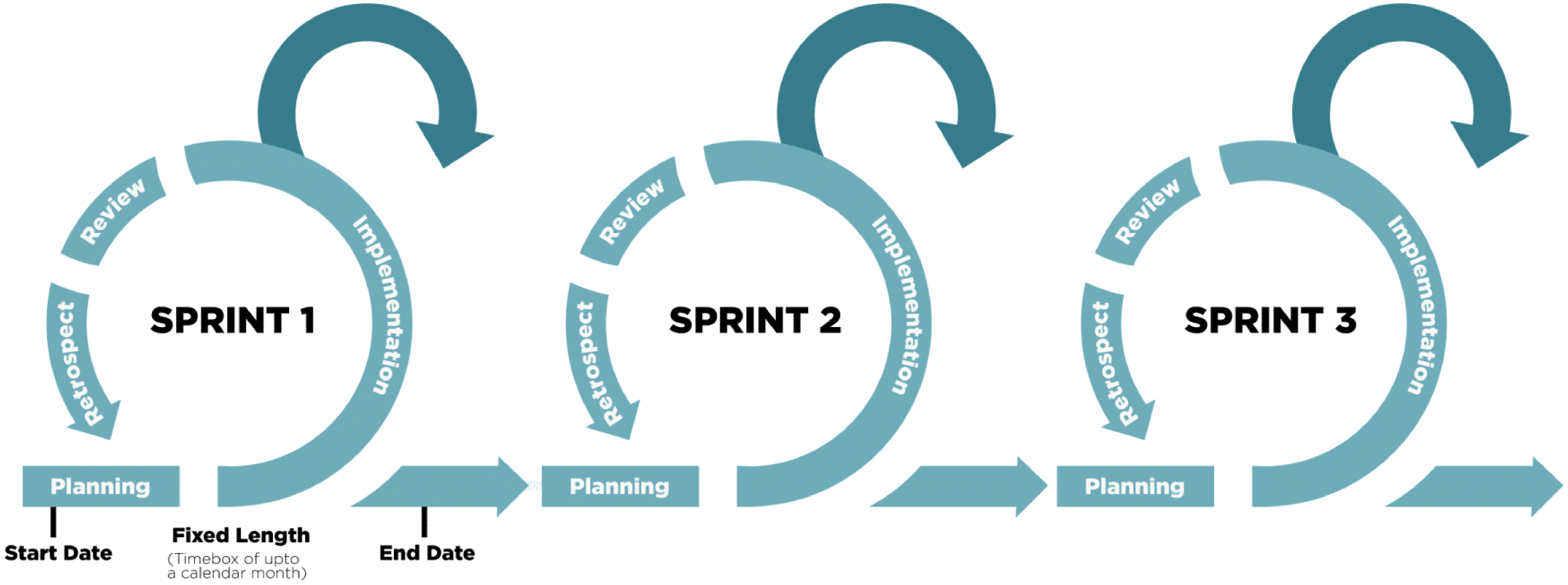
Scrum Framework

- 1986 – The name Scrum appears in a paper by management experts Hirotaka Takeuchi and Ikujiro Nonaka, called “The New New Product Development Game”, relating to rugby stressing team collaboration for project success

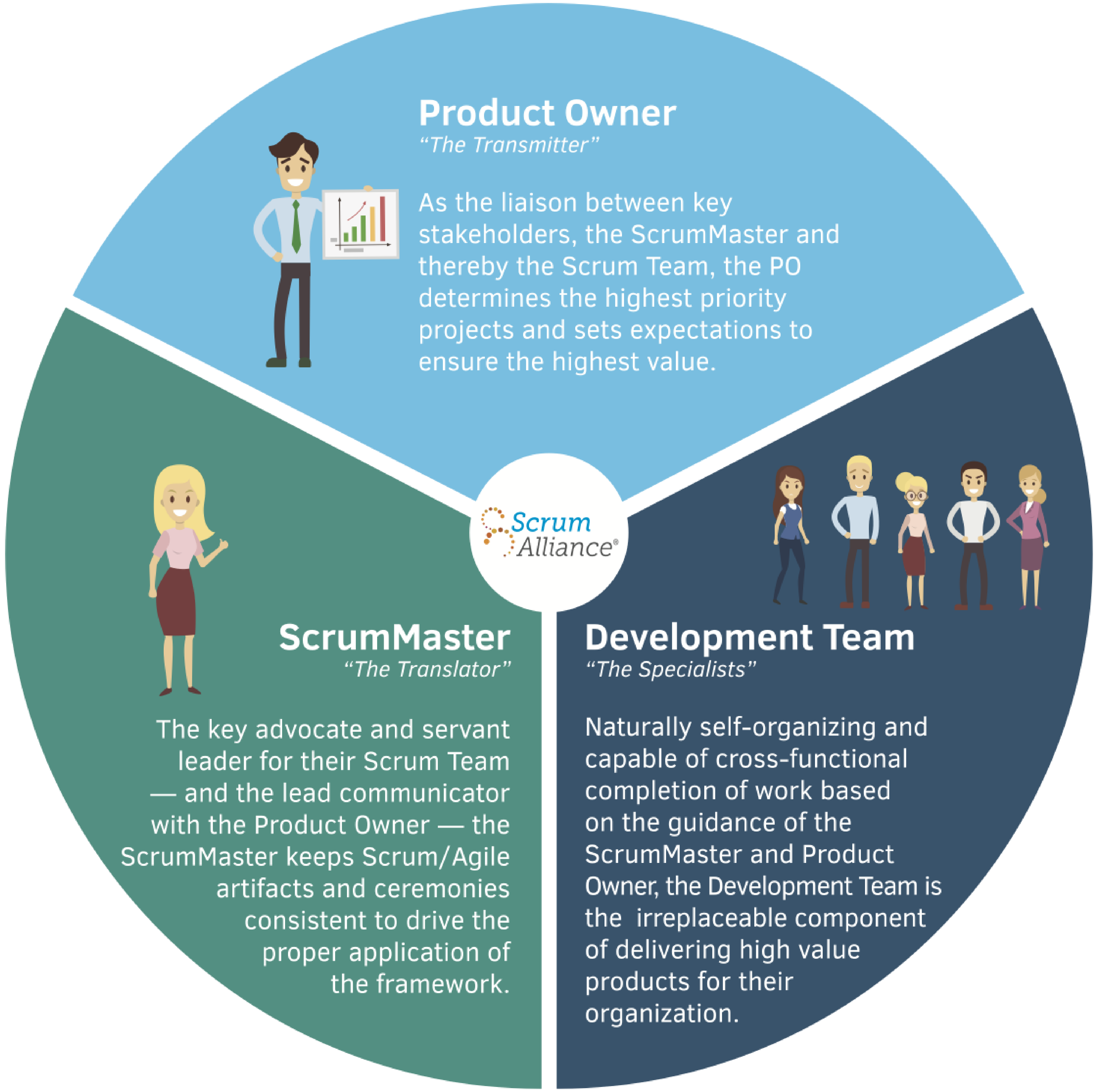
- 1995 – Jeff Sutherland and Ken Schwaber come up with process, which they presented to the OOPSLA conference in Austin, Texas

Scrum Sprints

- Timeboxed
- No change are made in middle of a sprint that could endanger the goal
- Output with a product increment
- Like a mini project



Scrum Roles



Product Owner

“The Transmitter”

As the liaison between key stakeholders, the ScrumMaster and thereby the Scrum Team, the PO determines the highest priority projects and sets expectations to ensure the highest value.

ScrumMaster

“The Translator”

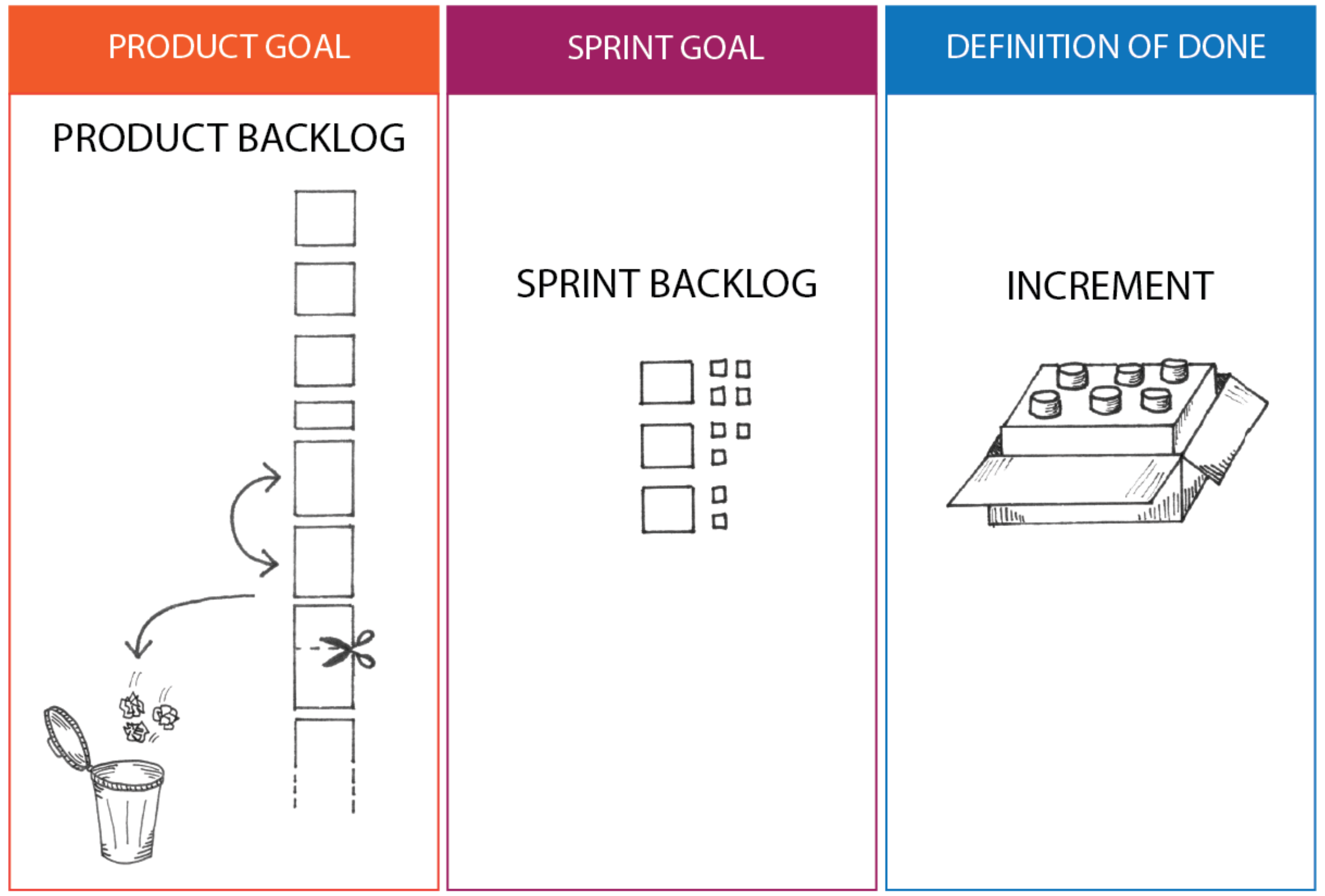
The key advocate and servant leader for their Scrum Team — and the lead communicator with the Product Owner — the ScrumMaster keeps Scrum/Agile artifacts and ceremonies consistent to drive the proper application of the framework.

Development Team

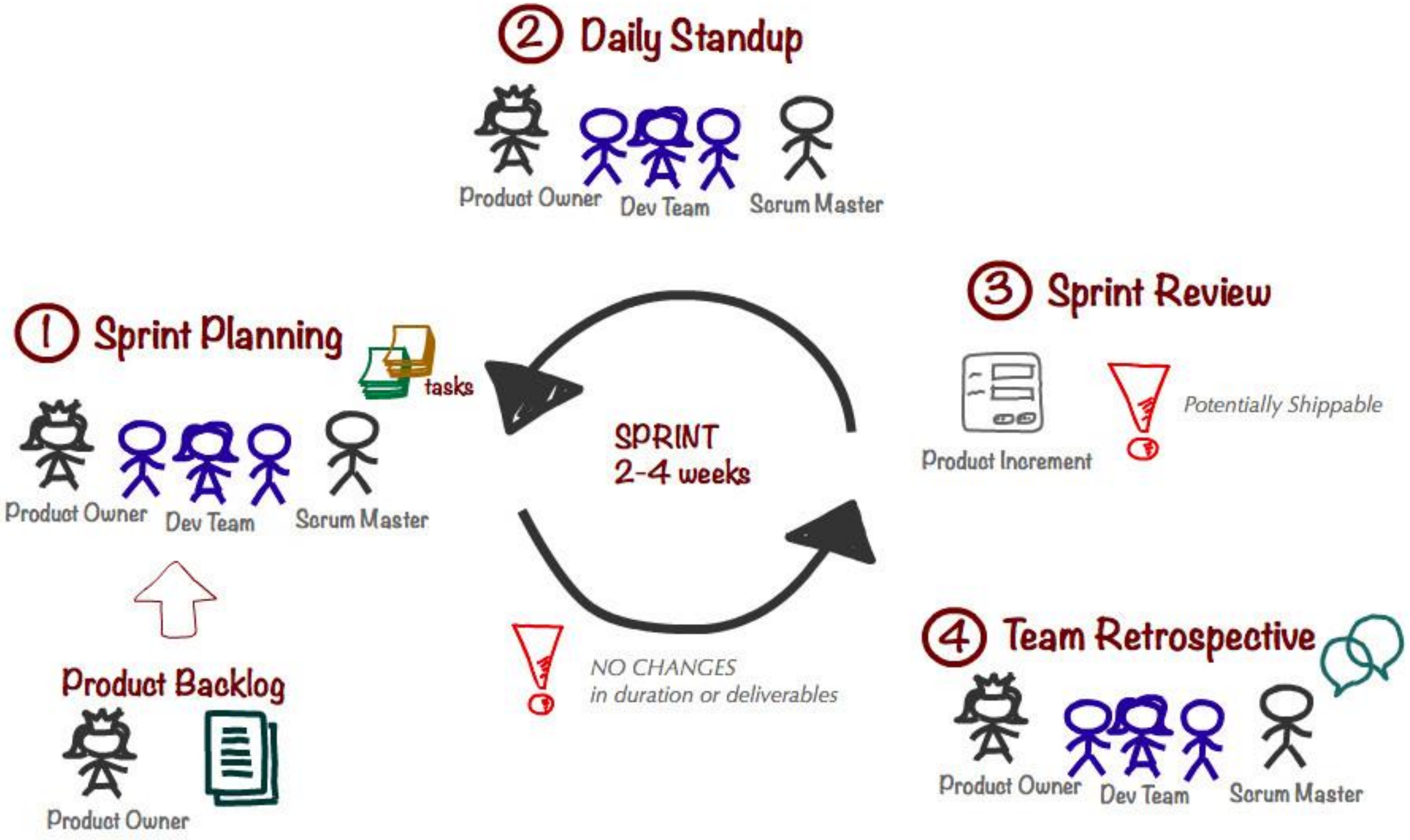
“The Specialists”

Naturally self-organizing and capable of cross-functional completion of work based on the guidance of the ScrumMaster and Product Owner, the Development Team is the irreplaceable component of delivering high value products for their organization.

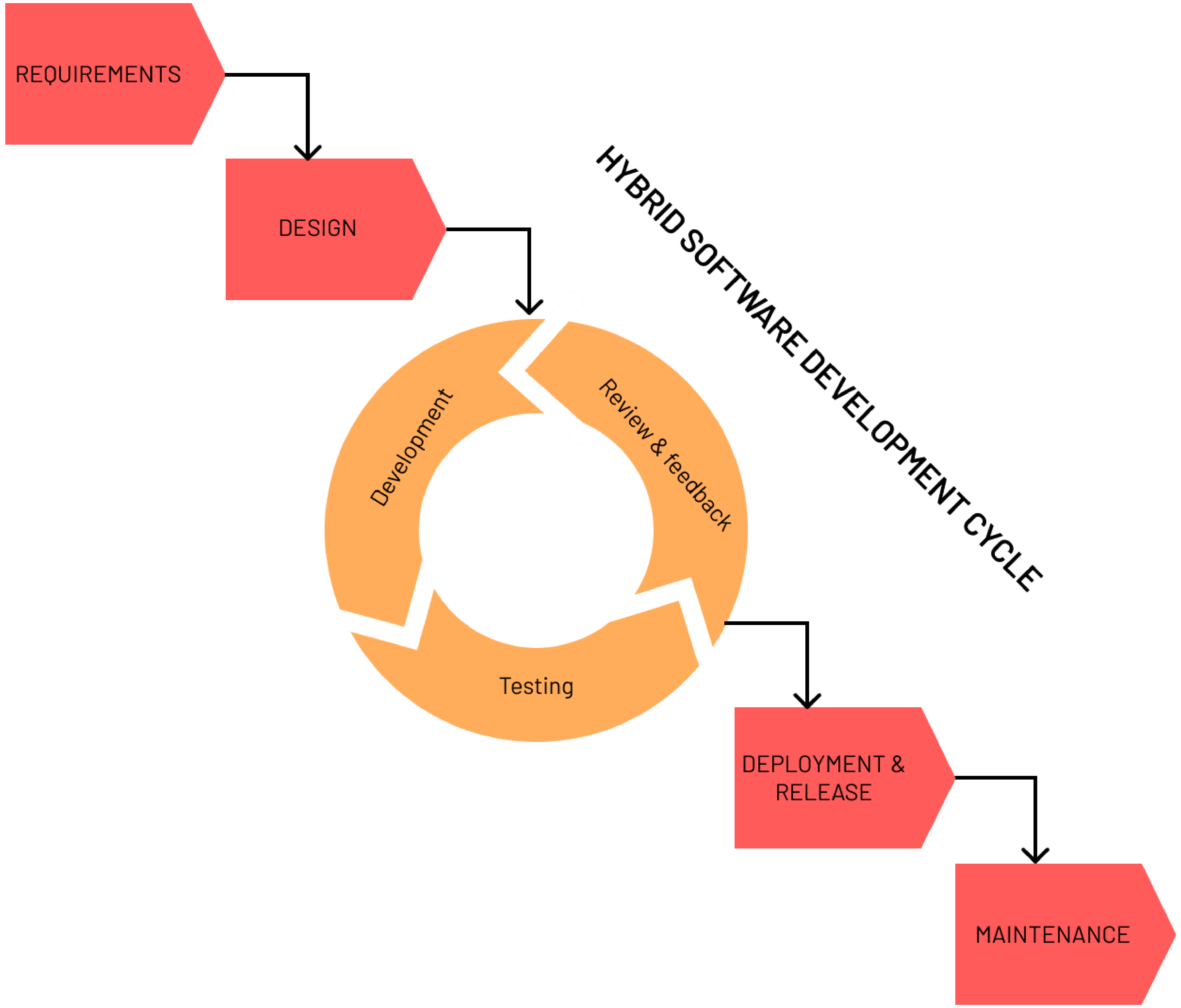
Scrum Artifacts & Ceremonies



PRODUCT GOAL	SPRINT GOAL	DEFINITION OF DONE
PRODUCT BACKLOG	SPRINT BACKLOG	INCREMENT



Real-Life Example



THANK YOU!