Agile History

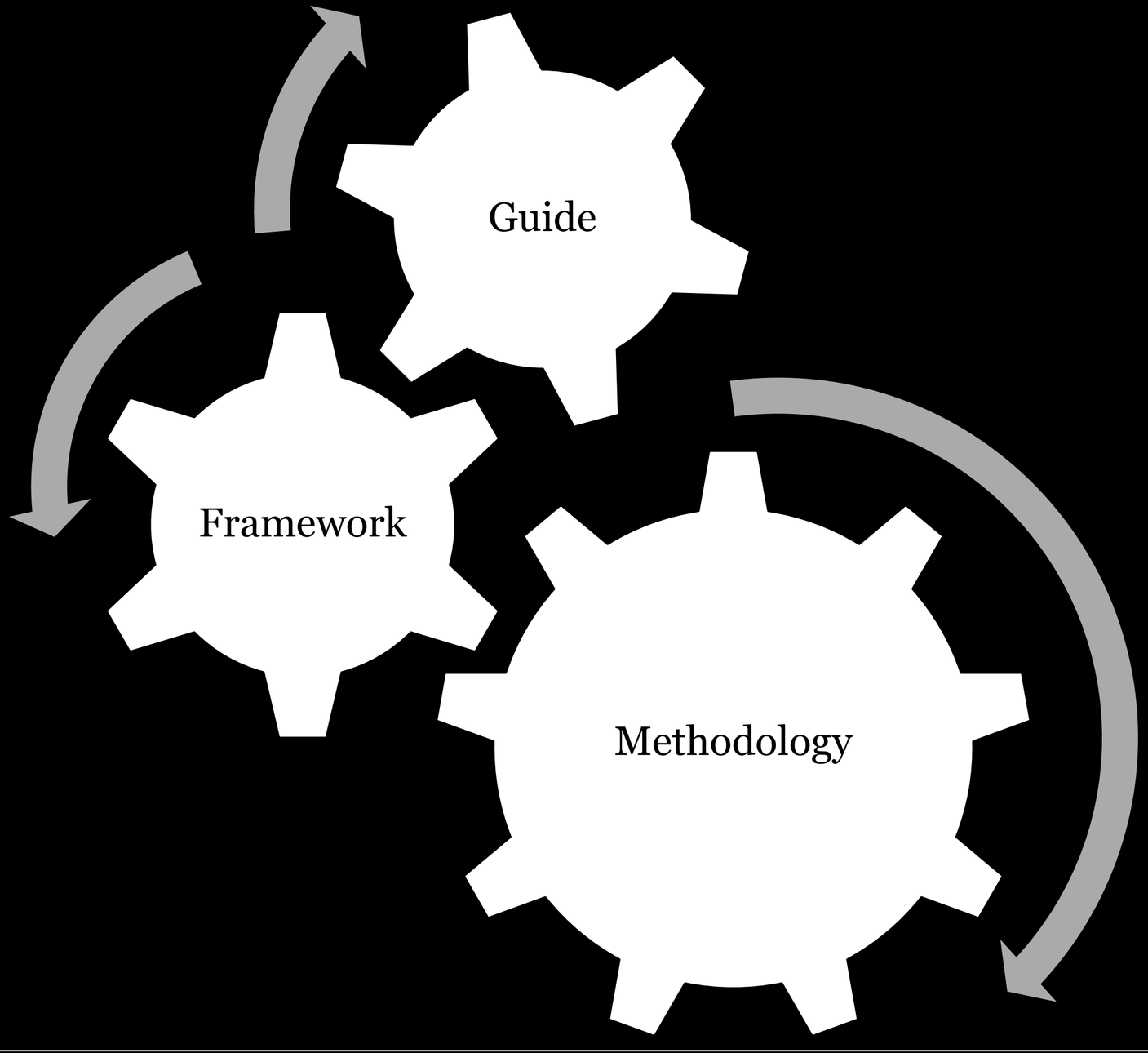
Systems Analysis & Design

Learning Objectives

By the end of this session, you will have acquired the following information:

- The difference between methodology, framework, and guide
- The Waterfall approach
- The Iterative and Incremental approach
- The Agile Manifesto
- Agile principles
- The Cynefin Framework

Three Terms


Guide
Framework
Methodology

Waterfall Approach

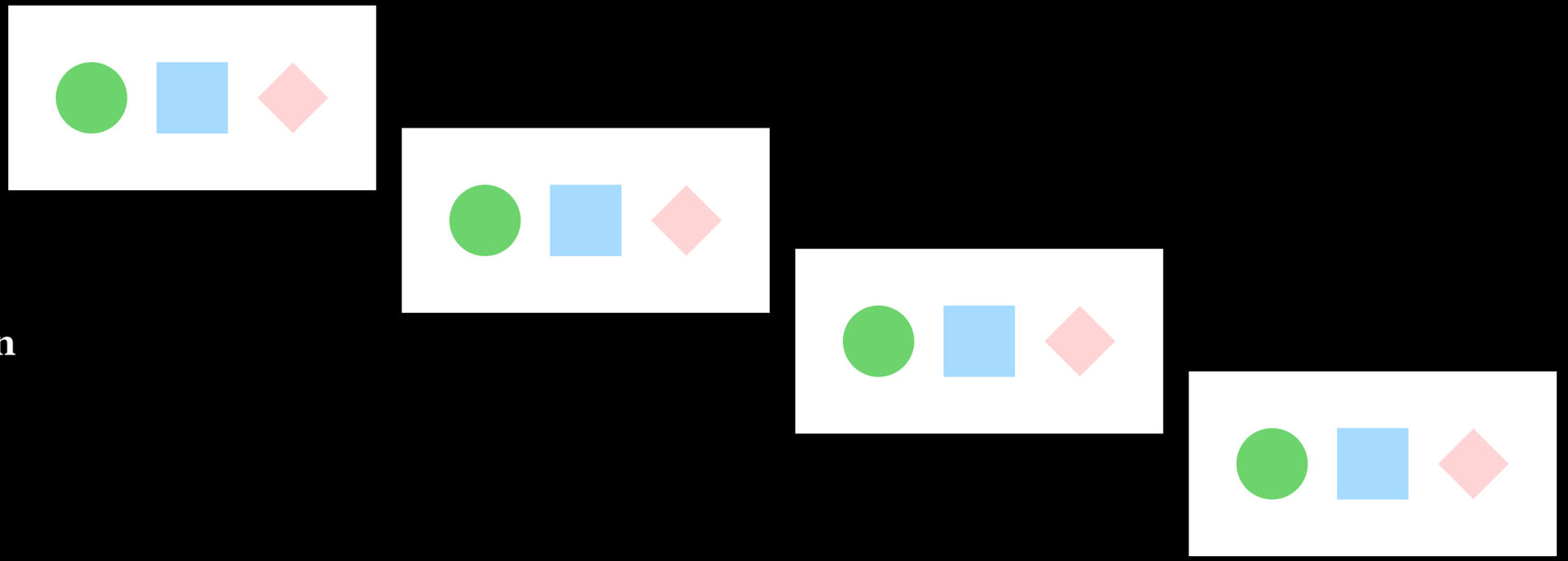
Time

Analysis

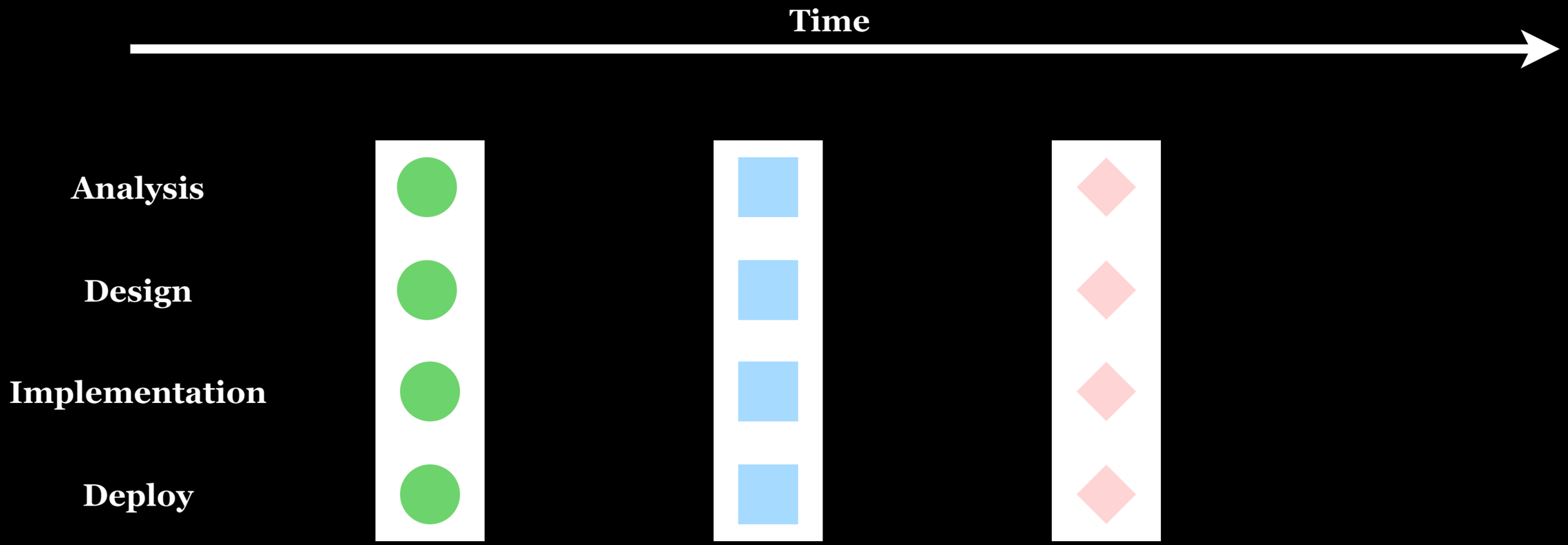
Design

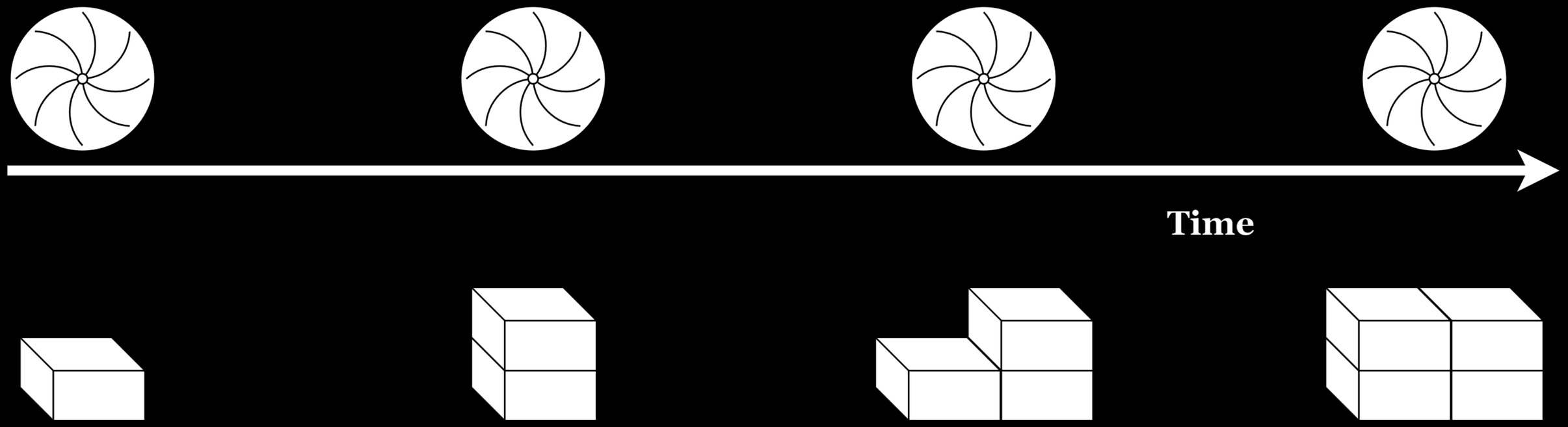
Implementation

Deploy



Iterative & Incremental Approach


Time
Analysis
Design
Implementation
Deploy


Time

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

Customer Satisfaction

1. Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.

Embrace Change

2. Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.

Frequent Delivery

3. Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.

Collaboration

4. Business people and developers must work together daily throughout the project.

Motivated Individuals

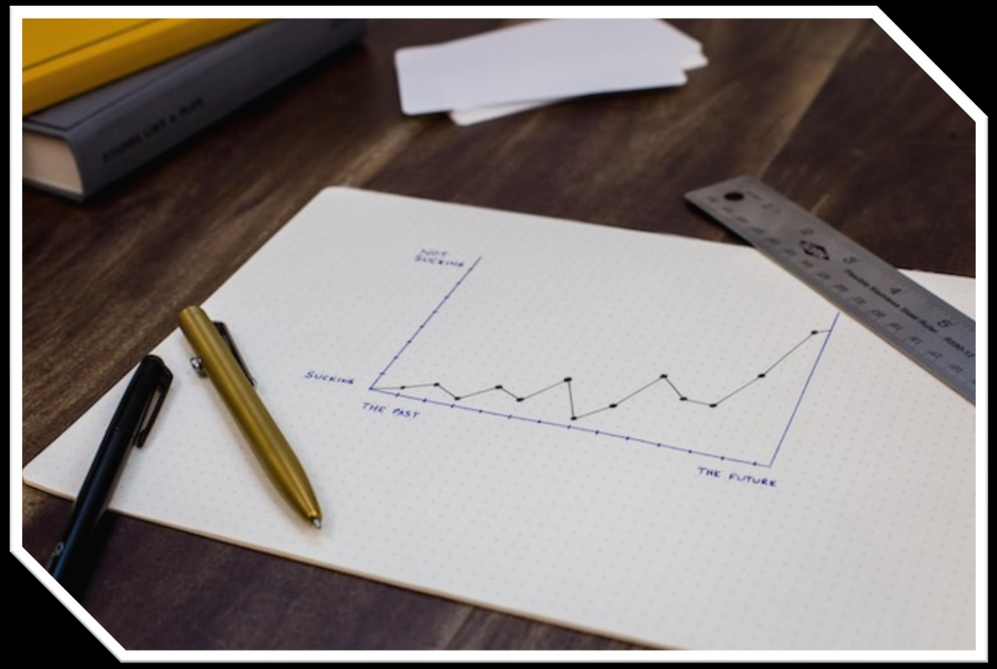
5. Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.

Effective Communication

6. The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.

Measure of Progress

7. Working software is the primary measure of progress.



Sustainable Development

8. Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.

Technical Excellence

9. Continuous attention to technical excellence and good design enhances agility.

Simplicity

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

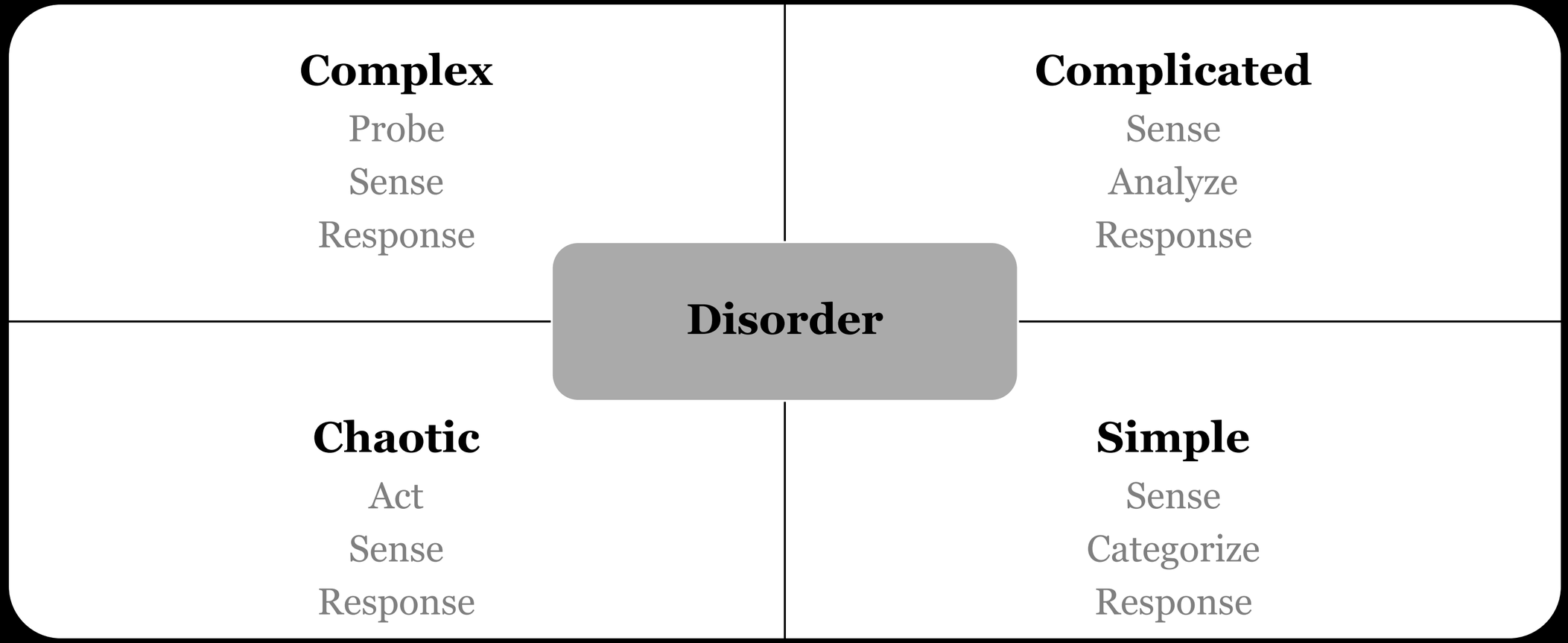
Self-Organizing Teams

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

Continuous Improvement

12. At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.

The Cynefin Framework



Further Resources

- Essential Scrum: A Practical Guide to the Most Popular Agile Process (pages 51-54)
- Martin Fowler Post: Waterfall Process