

# AGILE SOFTWARE DEVELOPMENT

# Introduction to Agile

# Outline

- What is Agility?
- Agile Manifesto
- Agile Principles
- Traditional vs Agile Approach
- Agile Team Characteristics
- Agile Success Factors
- Benefits and Challenges
- Agile Flavors

## What is “Agility”?

**Agility** is the ability to change the body's position efficiently, and requires the integration of isolated movement skills using a combination of balance, coordination, speed, reflexes, strength, and endurance

Source: Wikipedia



In business, agility means the **capability of rapidly and efficiently adapting to changes**. Recently agility has been applied e.g. in the context of agile software development and agile enterprise.

*Source: Wikipedia*



# What is Agile Software Development?

Agile software development refers to a **group of software development methodologies that are based on similar principles**.

Agile methodologies generally promote:

- A project management process that encourages **frequent inspection** and **adaptation**;
- a leadership philosophy that encourages **team work, self-organization** and **accountability**;
- a set of engineering best practices that allow for **rapid delivery of high-quality software**;
- and a business approach that **aligns development with customer needs** and **company goals**.

The ability to create and respond to change in order to succeed in an uncertain and turbulent environment.

*Source: AgileAlliance*

# History of Agile

## History & Influences

### Early 1900s

- Walter Shewhart: Plan-Do-Study-Act, SPC

### Mid 1900s

- Edward Deming: SPC, TQM
- Toyota: Toyota Production System (TPS)
- Peter Drucker: Knowledge Worker

### Late 1900s

- Womack and Jones: Lean Thinking
- Eli Goldratt: Theory of Constraints
- Tom Gilb: Evo
- The Toyota Way

## Evolution

### Early 1990s

- Crystal Methods
- Lean Software Development
- Dynamic Software Development Method (DSDM)

### Mid 1990s

- Feature Driven Development (FDD)
- eXtreme Programming (XP)
- Adaptive Software Development

### 2001: Manifesto for Agile Software Development

- <http://www.agilemanifesto.org>

### 2005: Declaration of Interdependence

- <http://www.pmdoi.org/>

# 17 Software developers created Agile

Kent Beck	James Grenning	Robert C. Martin
Mike Beedle	Jim Highsmith	Steve Mellor
Arie van Bennekum	Andrew Hunt	Ken Schwaber
Alistair Cockburn	Ron Jeffries	Jeff Sutherland
Ward Cunningham	Jon Kern	Dave Thomas
Martin Fowler	Brian Marick	



## Metaphor

***“Driving is not about getting the car going in the right direction. Driving is about constantly paying attention, making a little correction this way, a little correction that way.”***



*This is the paradigm for Agile. Stay aware. Adapt. Change.*

# 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."*

# 12 principles of Agile

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



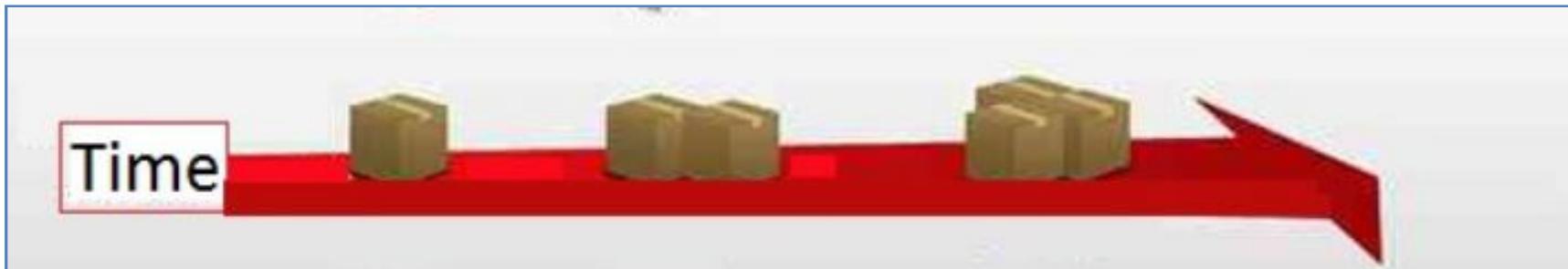
## 12 principles of Agile contd...

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



## 12 principles of Agile contd...

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



## 12 principles of Agile contd...

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



## 12 principles of Agile contd...

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



## 12 principles of Agile contd...

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

### **Effective Communication**



## 12 principles of Agile contd...

7. **Working software** is the primary measure of progress.



(working software)



## 12 principles of Agile contd...

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



## 12 principles of Agile contd...

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



## 12 principles of Agile contd...

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



## 12 principles of Agile contd...

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



## 12 principles of Agile contd...

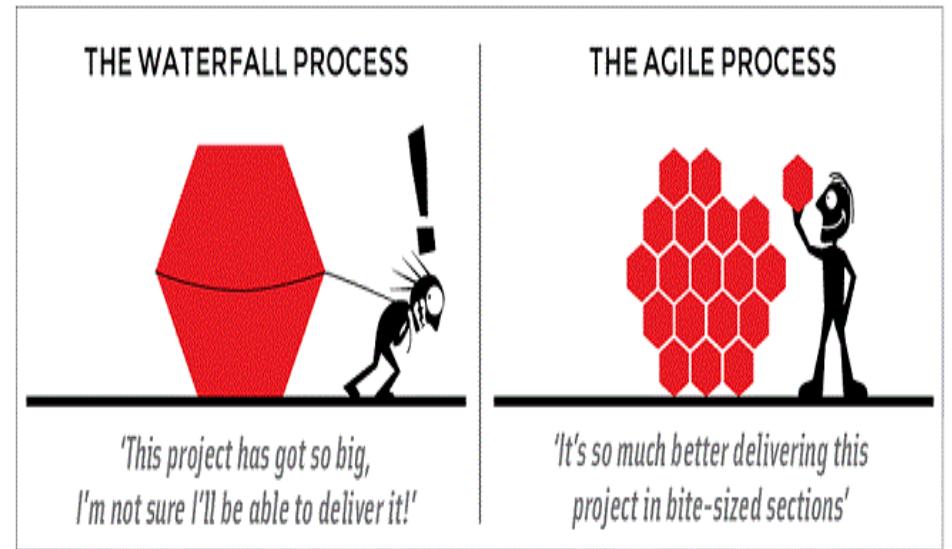
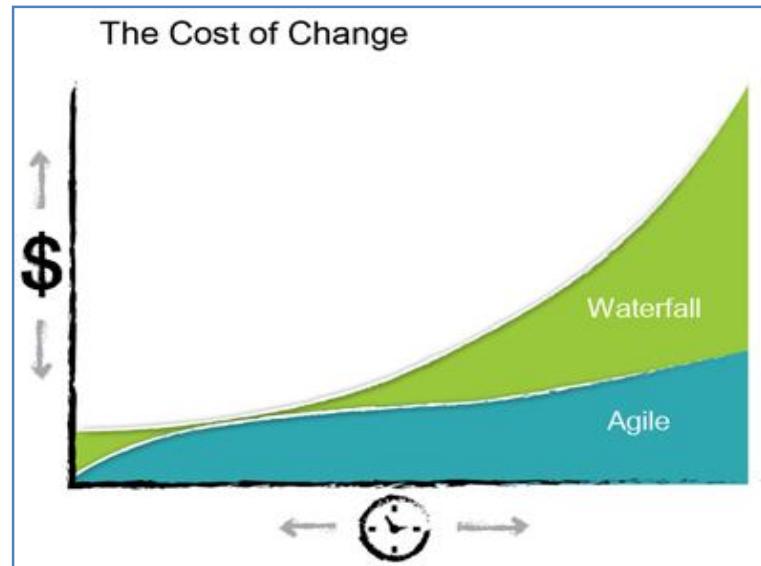
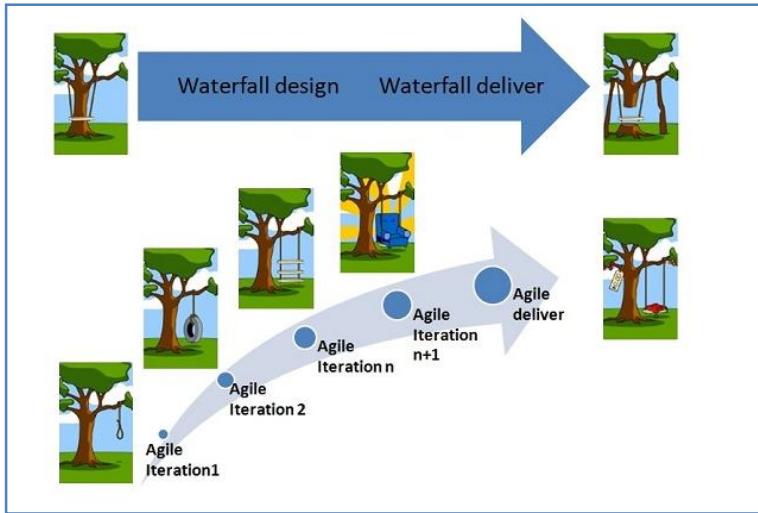
12. At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.

### Inspect/Adapt



# Coin Game

# Waterfall vs Agile



# Agile Team Characteristics



# Key Success Factors for Agile Software Development

- Management Support
- Agile Thinking
- Cross-Functional Teams
- Empowered Team Members
- Single Voice of the Business
- Shared Accountability
- Servant Leadership
- Collaboration
- Continuous Flow of Value
- Attention to Technical Excellence
- Rapid Risk Reduction
- Early Feedback and Adaptation
- Openness and Transparency
- Trust

# Benefits of Agile

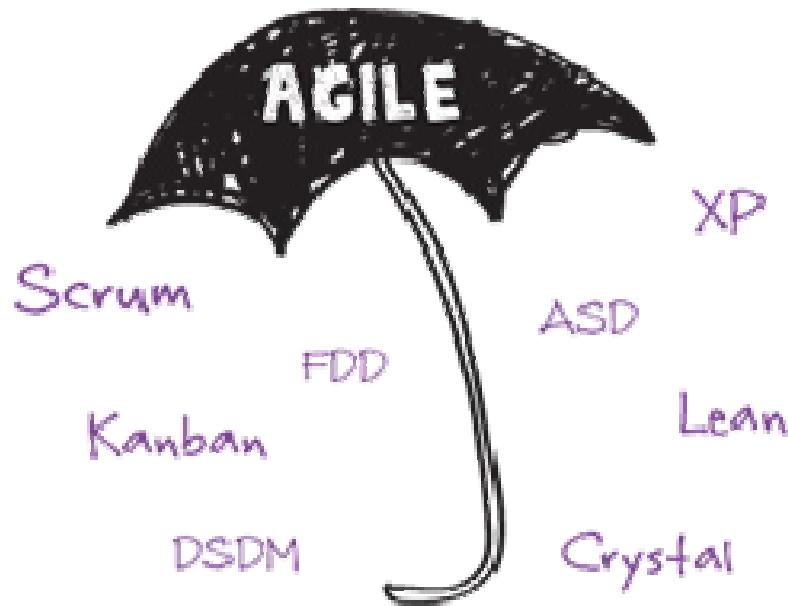
- Faster Time to Market
- Frequent Delivery
- Transparency
- Flexibility
- Increase Quality
- Increase Productivity
- Customer Satisfaction
- Predictability
- Decrease Risk

# Agile Challenges

- Customer Availability and Co-Operation
- Customer with clear vision
- People working together
- Team Composition
- Project Cost and Effort Estimation
- Doing Agile over Being Agile

# Agile Flavors

- Scrum
- XP (eXtreme Programming)
- Lean Software Development
- Kanban
- DSDM (Dynamic Systems Development Method)
- Crystal
- FDD (Feature Driven Development)
- Others



# Thank You!