

# How to manage agile development?

Rose Pruyne | Jack Reed

# What will we cover?

Introductions

Overview and principles

User story exercise

Retrospective exercise

Getting started

Q&A

# About me:

Jack Reed

Geospatial Web Engineer, Stanford University

Passionate about making geospatial data accessible

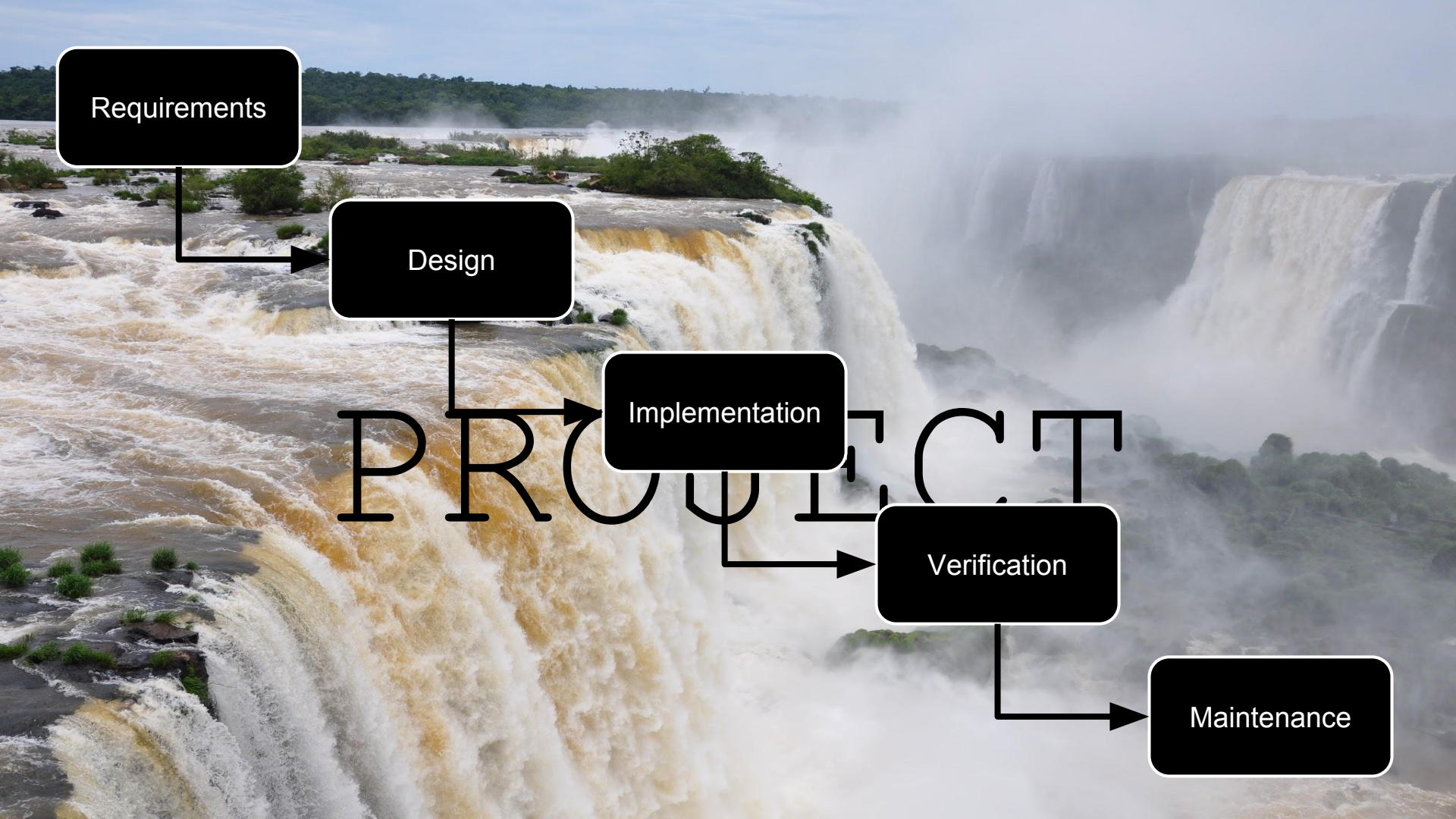
Not an Agile Coach or a ScrumMaster

# More than a buzzword?

*The word “agile” has been subverted to the point where it is effectively meaningless, and what passes for an agile community seems to be largely an arena for consultants and vendors to hawk services and products.*

Dave Turner - [Agile Is Dead \(Long Live Agility\)](#)

# Waterfall vs. Agile



Requirements

Design

Implementation

Verification

Maintenance

PROJECT

# Agile - A set of principles

1. Customer satisfaction by rapid delivery of useful software
2. Welcome changing requirements, even late in development
3. Working software is delivered frequently (weeks rather than months)
4. Close, daily cooperation between business people and developers
5. Projects are built around motivated individuals, who should be trusted
6. Face-to-face conversation is the best form of communication (co-location)
7. Working software is the principal measure of progress
8. Sustainable development, able to maintain a constant pace
9. Continuous attention to technical excellence and good design
10. Simplicity—the art of maximizing the amount of work not done—is essential
11. Self-organizing teams
12. Regular adaptation to changing circumstances

# 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.

<http://agilemanifesto.org>



Extreme Programming (XP)

Scrum-ban

Crystal Methods (Crystal Clear)

Feature Driven Development (FDD)

Scrum

Disciplined Agile Delivery

Agile Modeling

# Agile

Lean software development

Adaptive Software Development (ASD)

Agile Unified Process (AUP)

Dynamic Systems Development Method (DSDM)

Kanban (development)

***Scrum is a framework,  
agility is a concept***

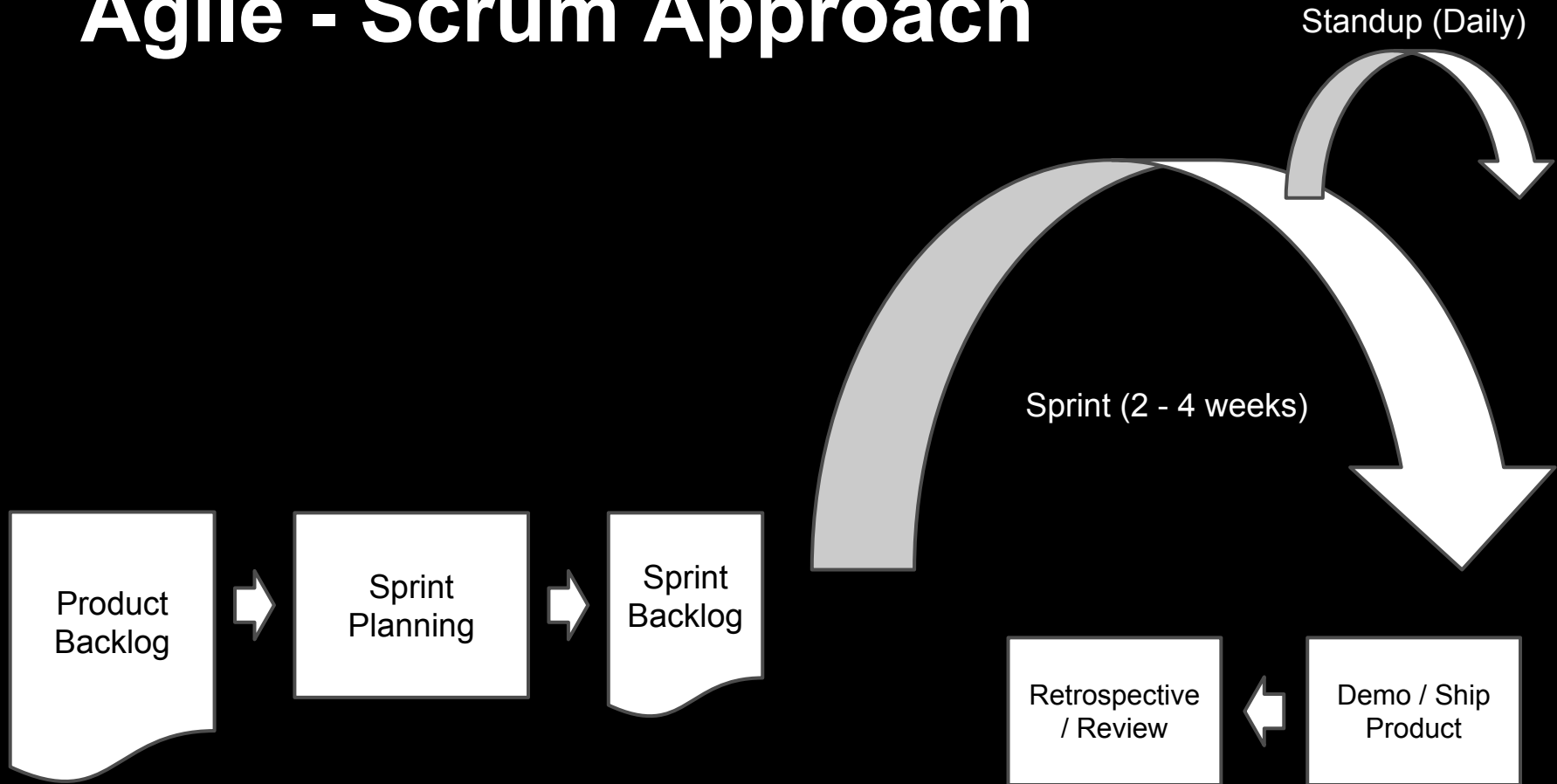
<http://scrummethodology.com/scrum-is-a-framework-agility-is-a-concept/>

# Agile - Scrum Approach

## Scrum Roles

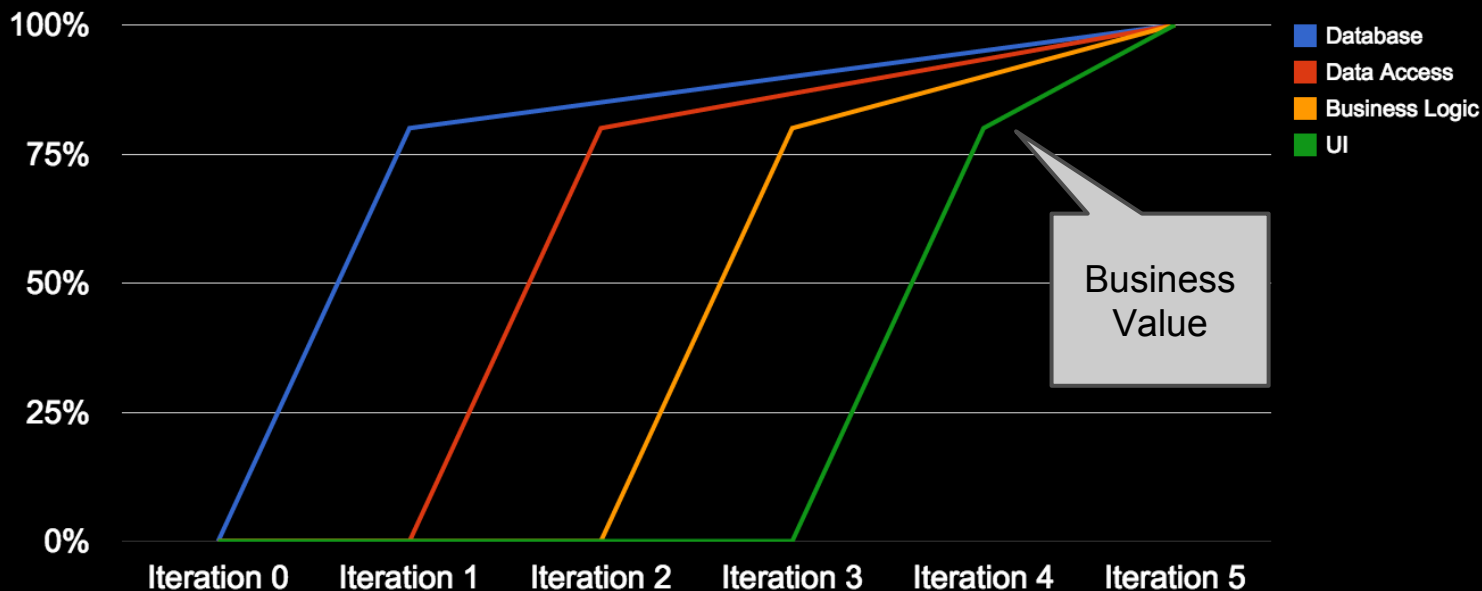
- Product Owner -- holds the vision for the product
- ScrumMaster -- helps the team best use Scrum to build the product
- Development team -- builds the product

# Agile - Scrum Approach

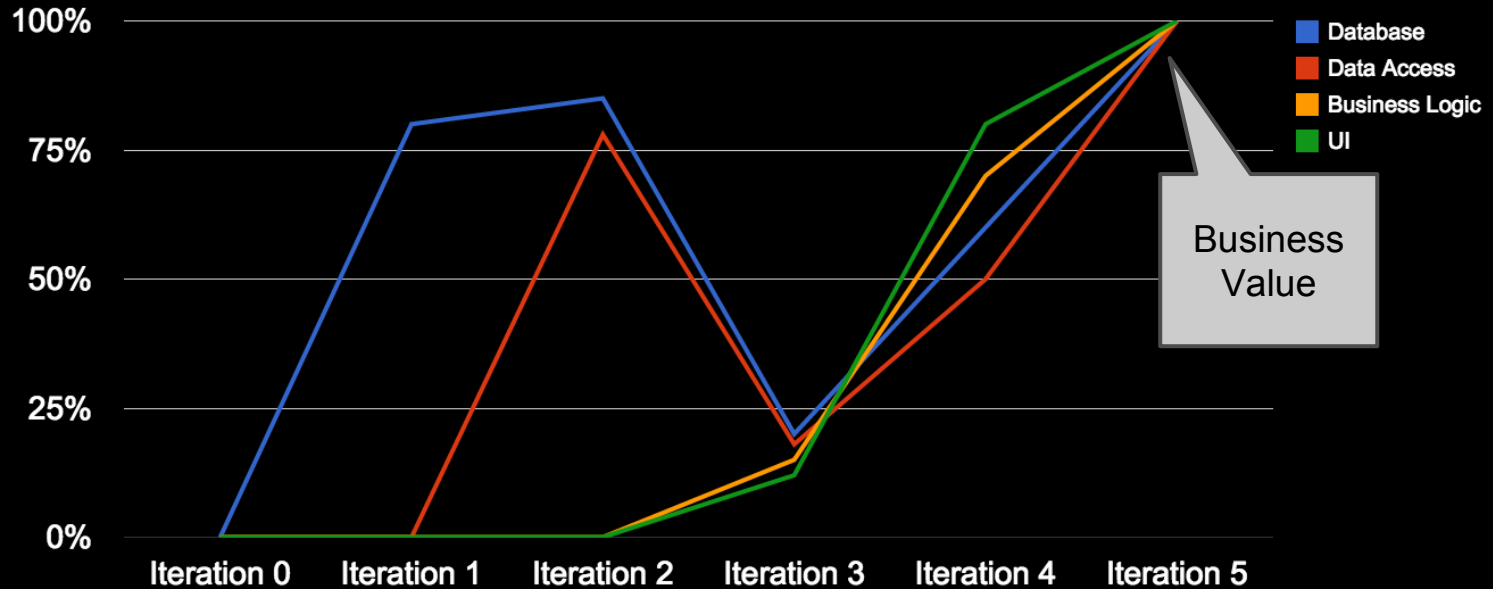


**Builds business value  
sooner**

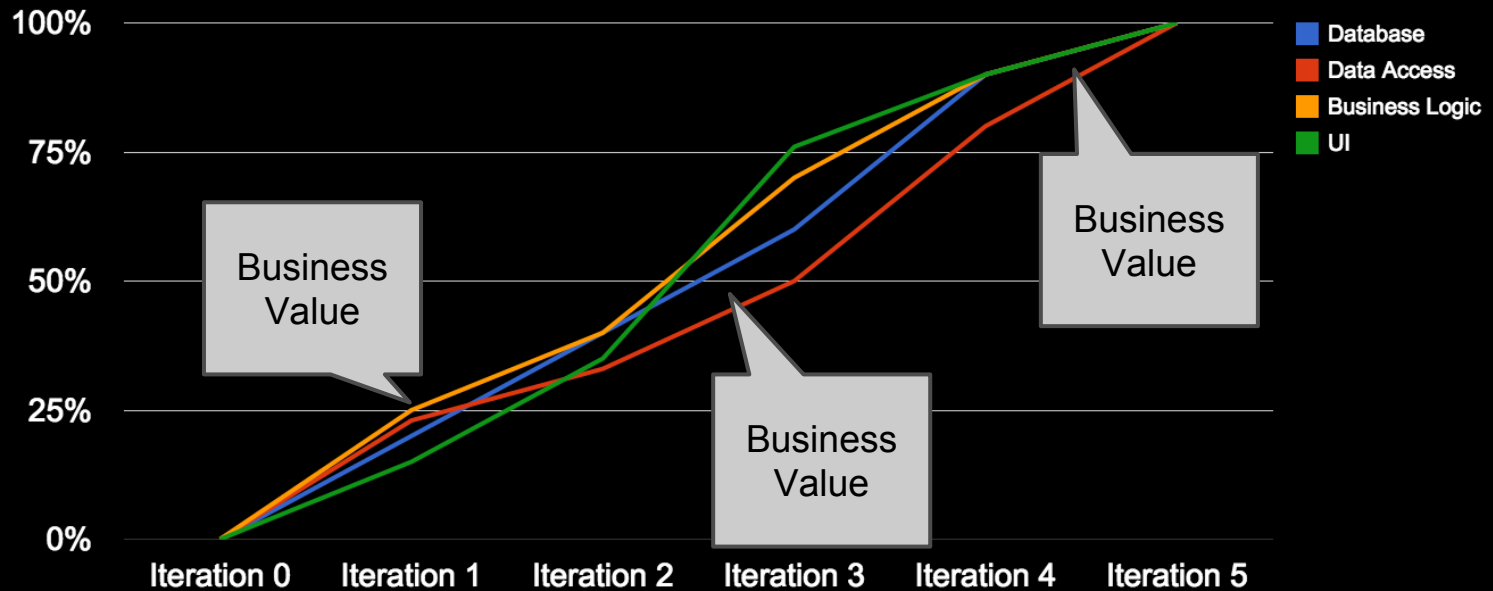
# My Hydra Project - Traditional Approach



# My Hydra Project - Traditional Approach (Reality)

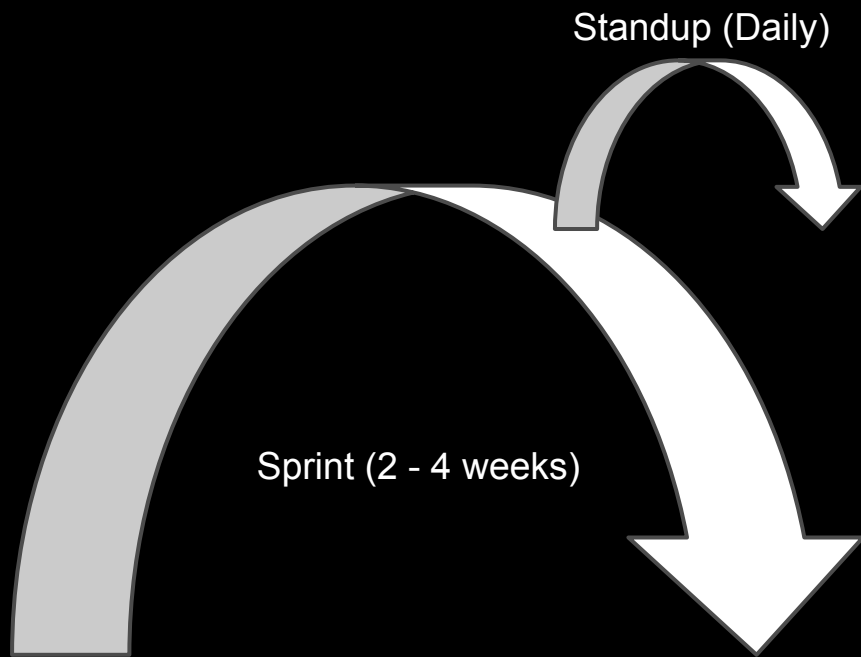


# My Hydra Project - Agile Approach





**You know things sooner**



**Better quality product**

**What about the real  
world?**

# Back to basics

1. Customer satisfaction by rapid delivery of useful software
2. Welcome changing requirements, even late in development
3. Working software is delivered frequently (weeks rather than months)
4. Close, daily cooperation between business people and developers
5. Projects are built around motivated individuals, who should be trusted
6. Face-to-face conversation is the best form of communication (co-location)
7. Working software is the principal measure of progress
8. Sustainable development, able to maintain a constant pace
9. Continuous attention to technical excellence and good design
10. Simplicity—the art of maximizing the amount of work not done—is essential
11. Self-organizing teams
12. Regular adaptation to changing circumstances