

# Agile Software Development

What is it and do we want it?

Dr Jennifer Ferreira

[jennifer@ecs.vuw.ac.nz](mailto:jennifer@ecs.vuw.ac.nz)

8 November 2019

# Software development process

- Identify distinct phases: Requirements, Design, Implementation, Verification
- Organise the phases into a logical sequence of activities
- Control the variables of cost, time, scope, and quality



# Software Development Process

## Iterative Development

Business value is delivered incrementally in time-boxed cross-discipline iterations.

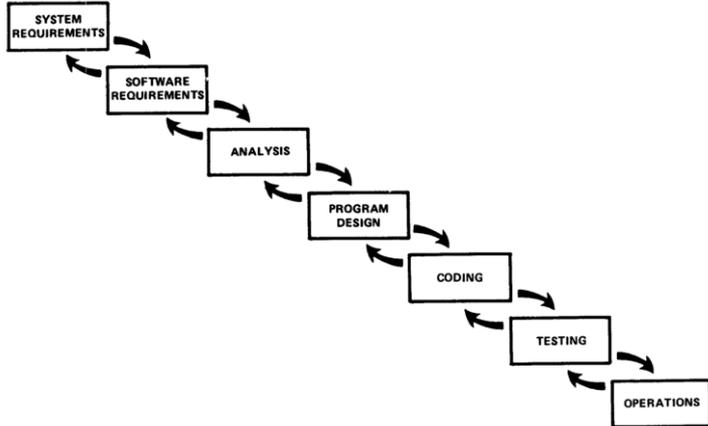
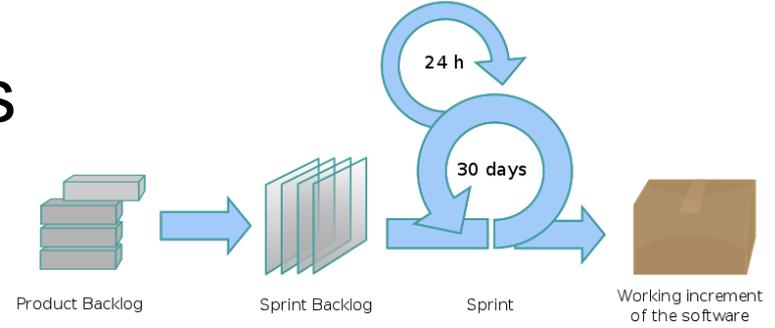
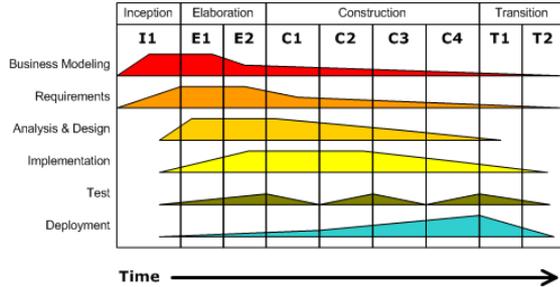


Figure 3. Hopefully, the iterative interaction between the various phases is confined to successive steps.

## Extreme Programming (XP)

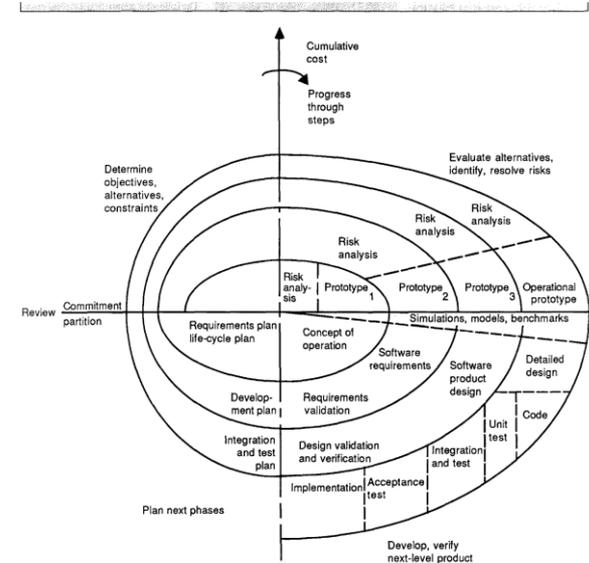
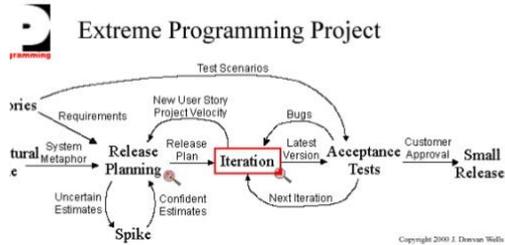
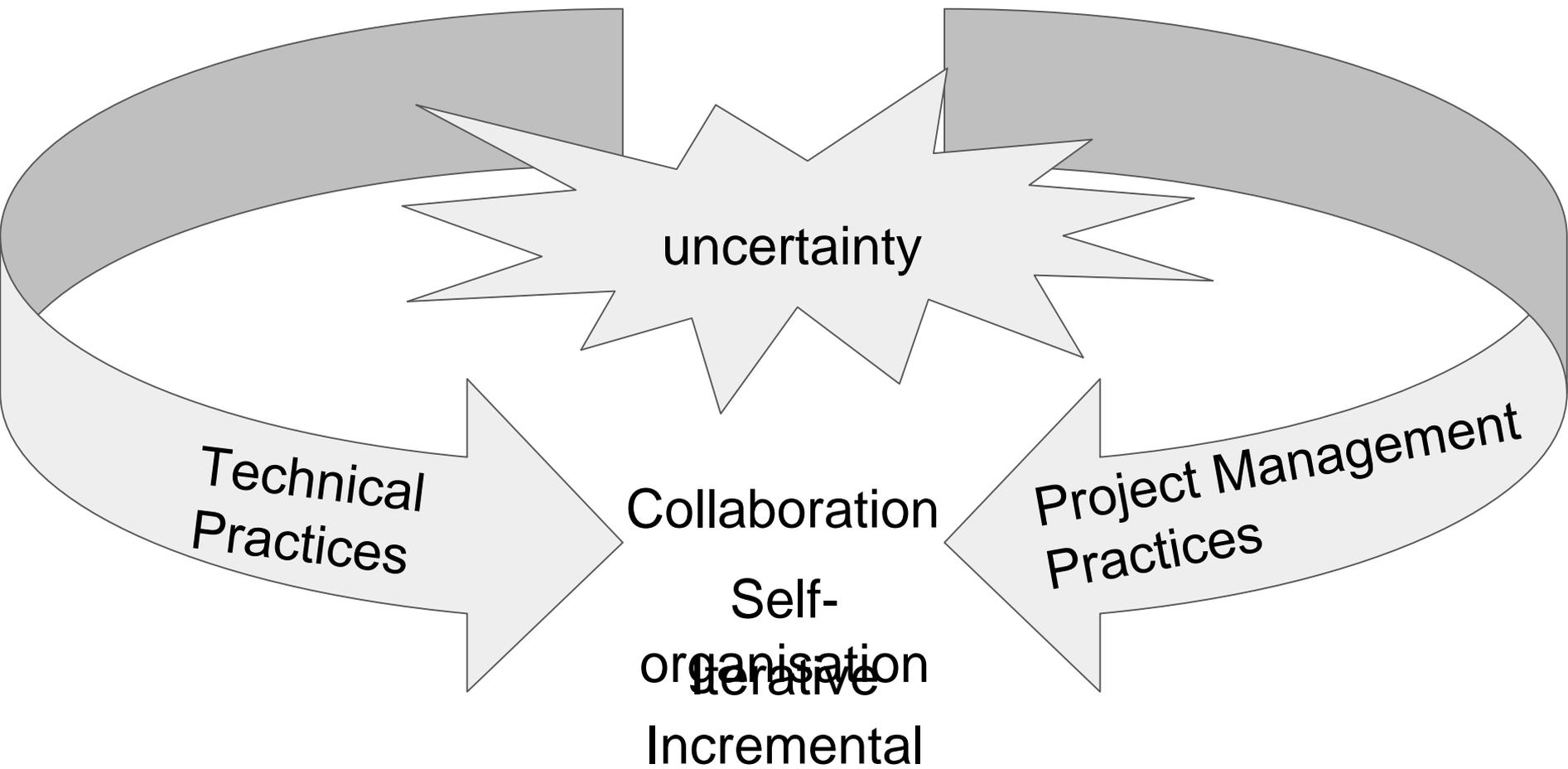


Figure 2. Spiral model of the software process.

# Agile Software Development

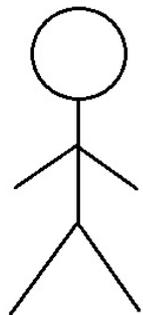


uncertainty

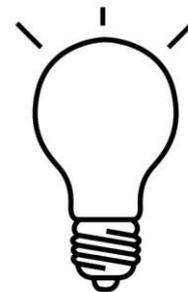
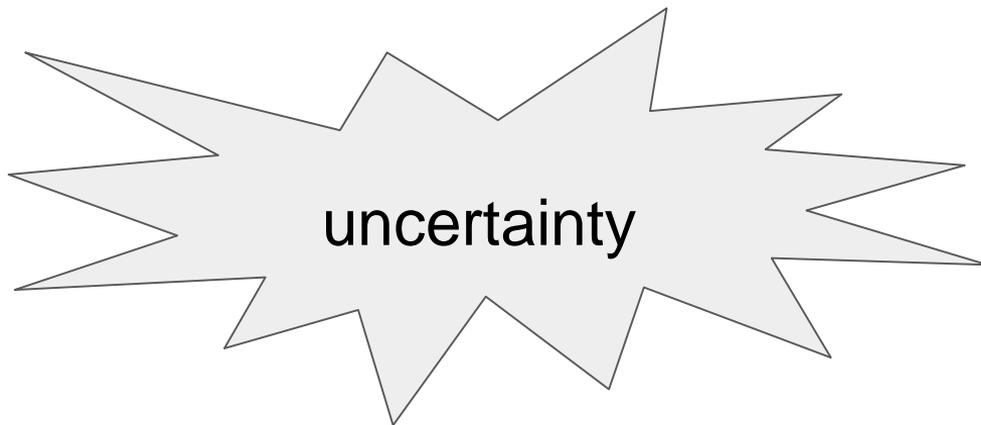
Technical Practices

Project Management Practices

Collaboration  
Self-  
organisation  
Iterative  
Incremental



Customer



Coding

# Agile Software Development

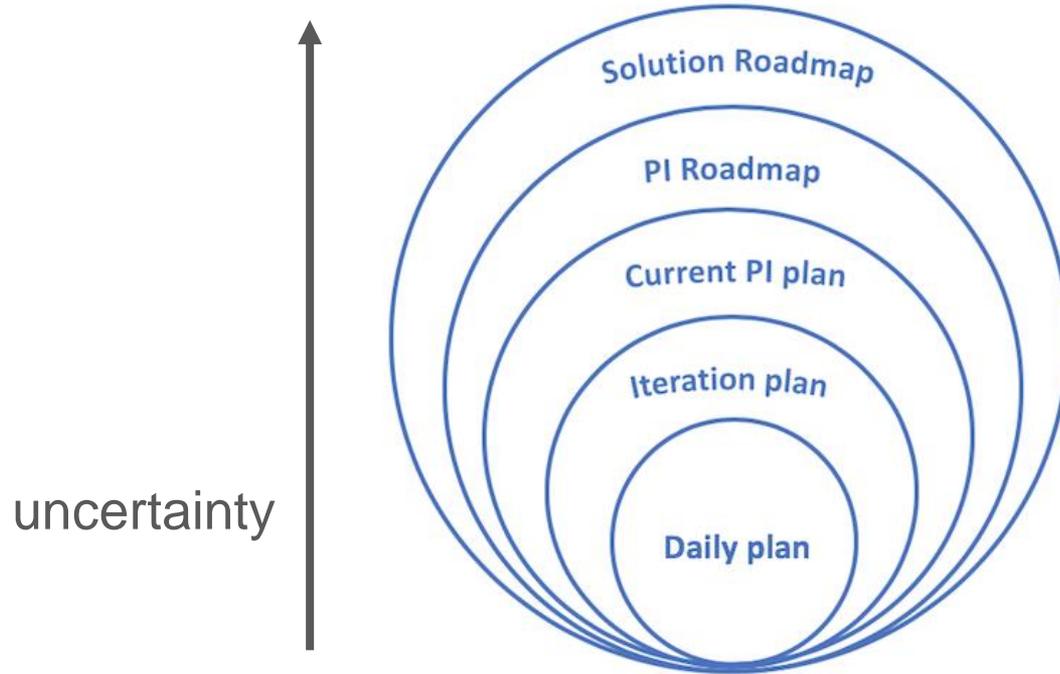
1. Requirements
2. Teams
3. Specific challenges for scientific software development
4. Starting points with Agile Development

# Agile Requirements

1. Planning horizons
2. User stories
3. Backlog

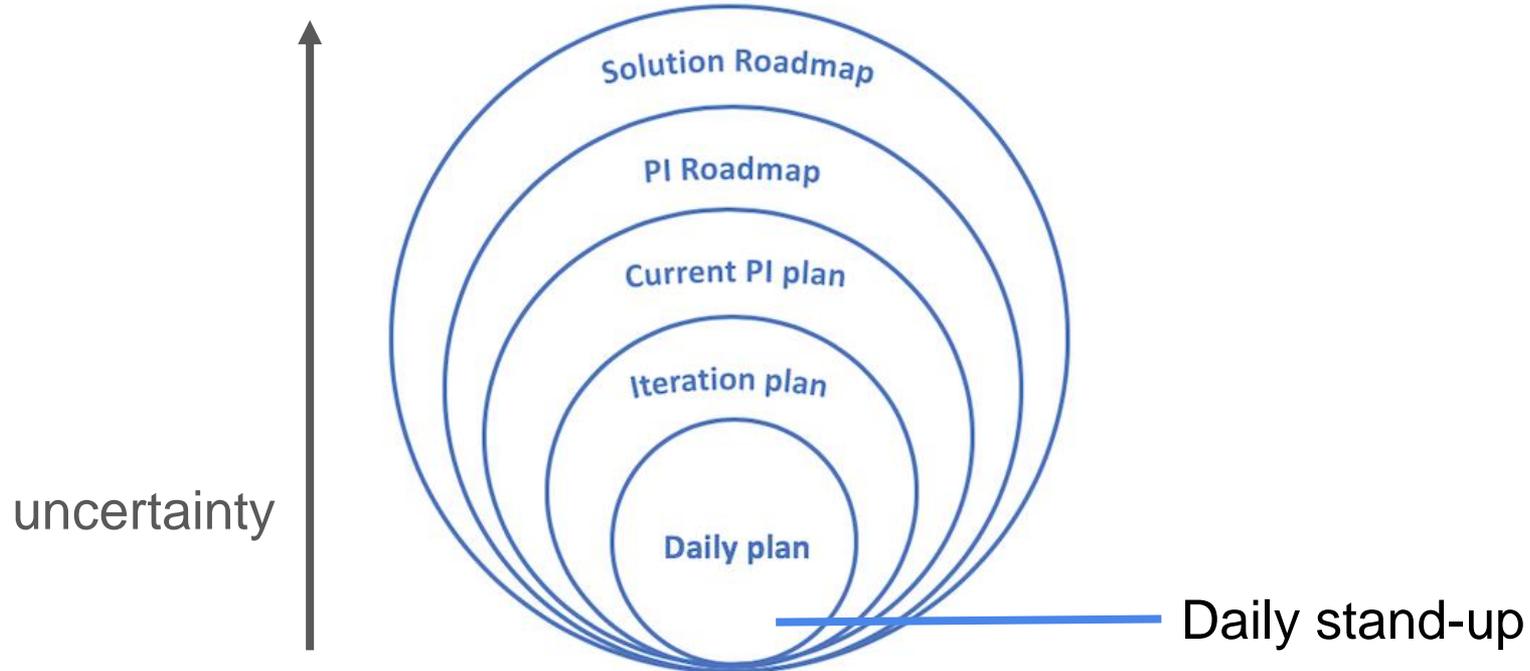
# Agile Requirements: Time horizons

Solution Roadmap



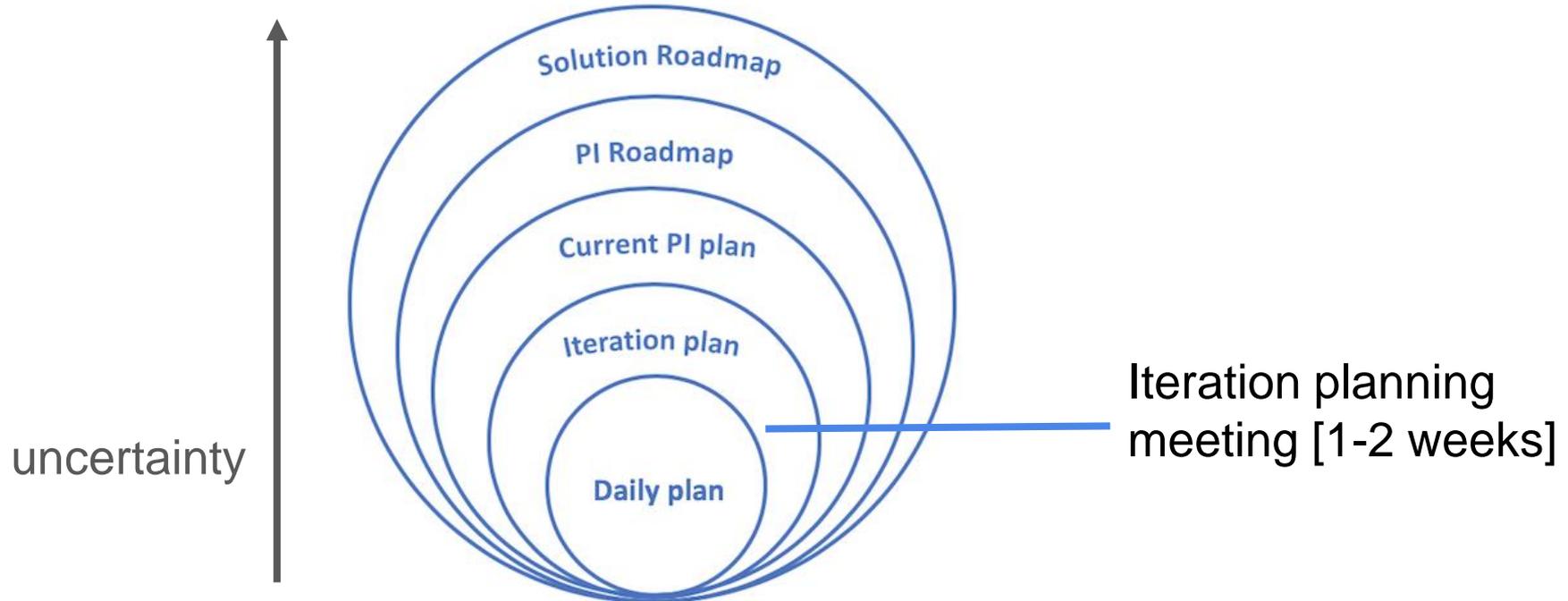
# Agile Requirements: Time horizons

Solution Roadmap



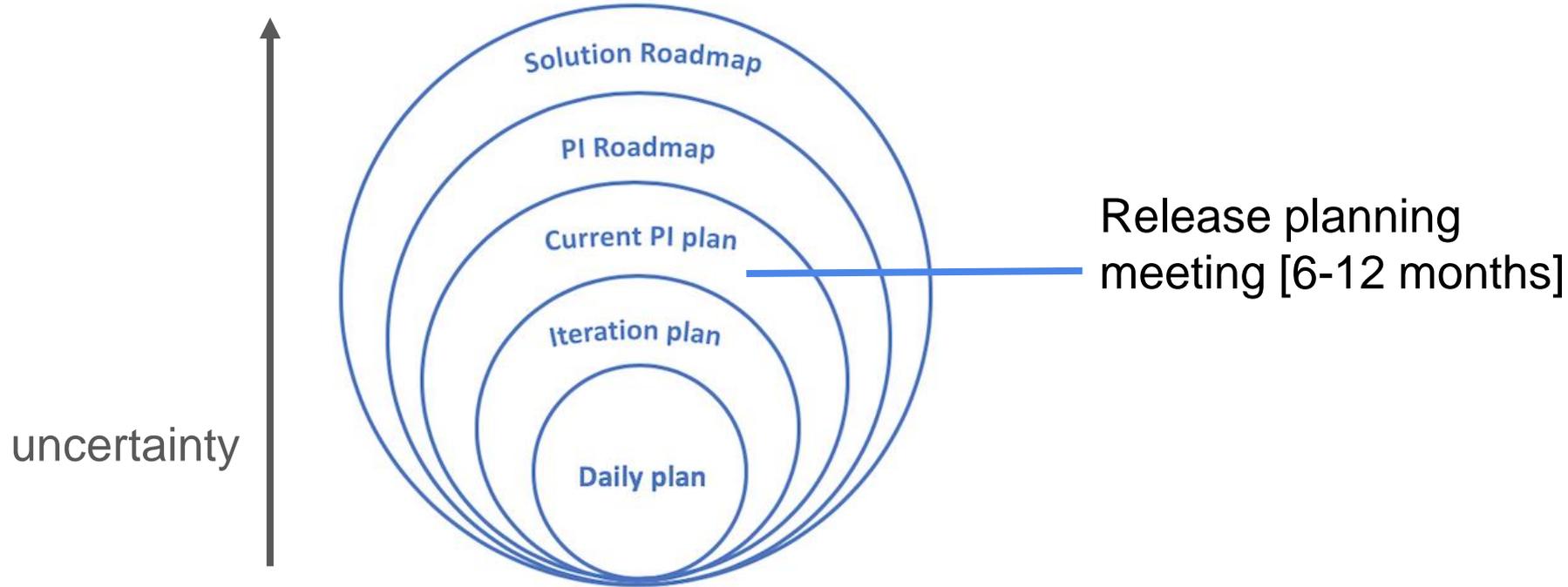
# Agile Requirements: Time horizons

Solution Roadmap



# Agile Requirements: Time horizons

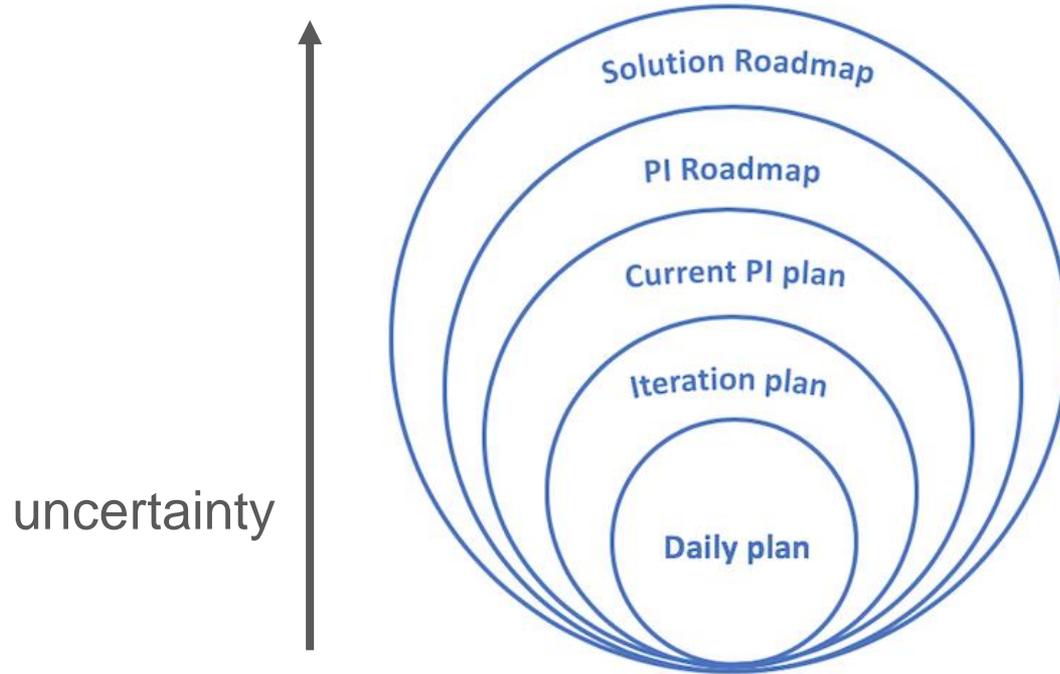
Solution Roadmap



Release planning meeting [6-12 months]

# Agile Requirements: Time horizons

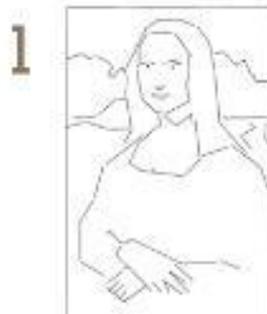
Solution Roadmap



## incremental



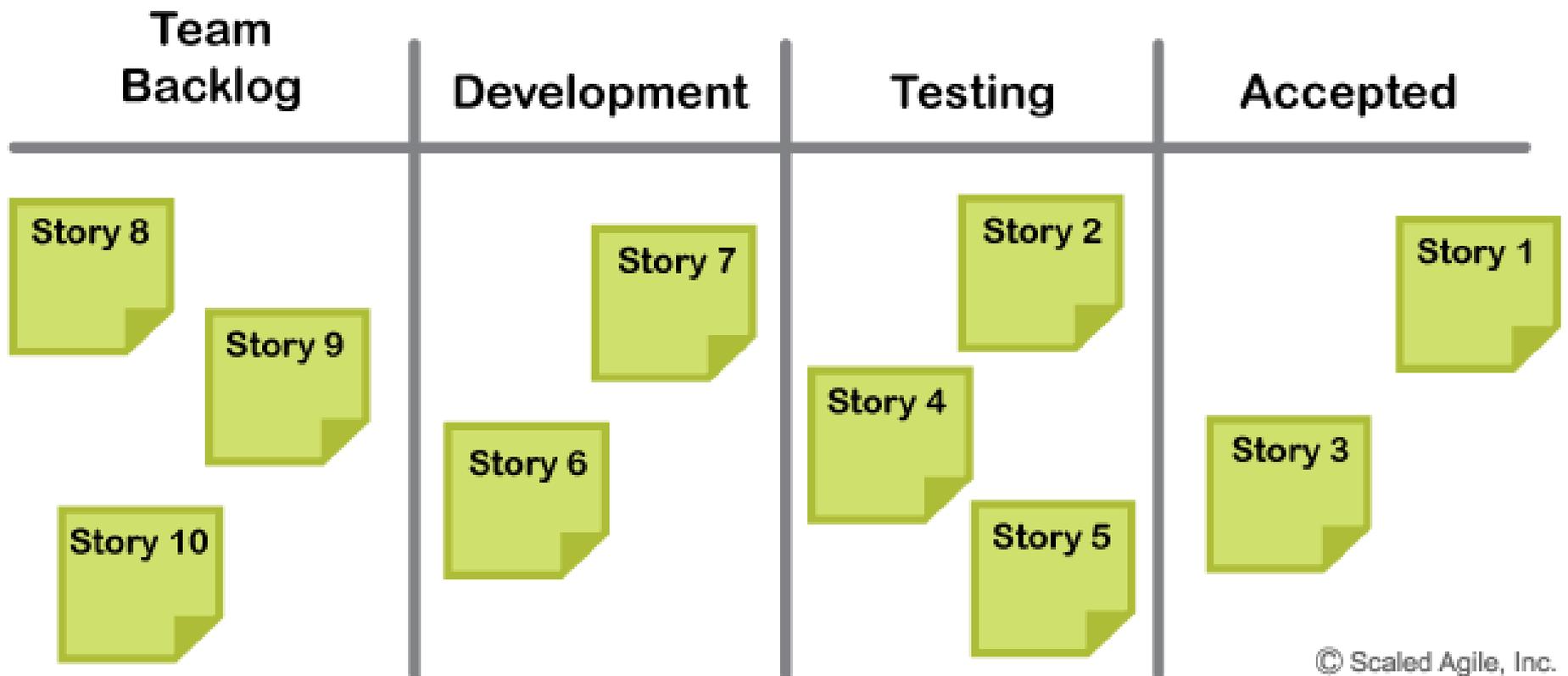
## iterative



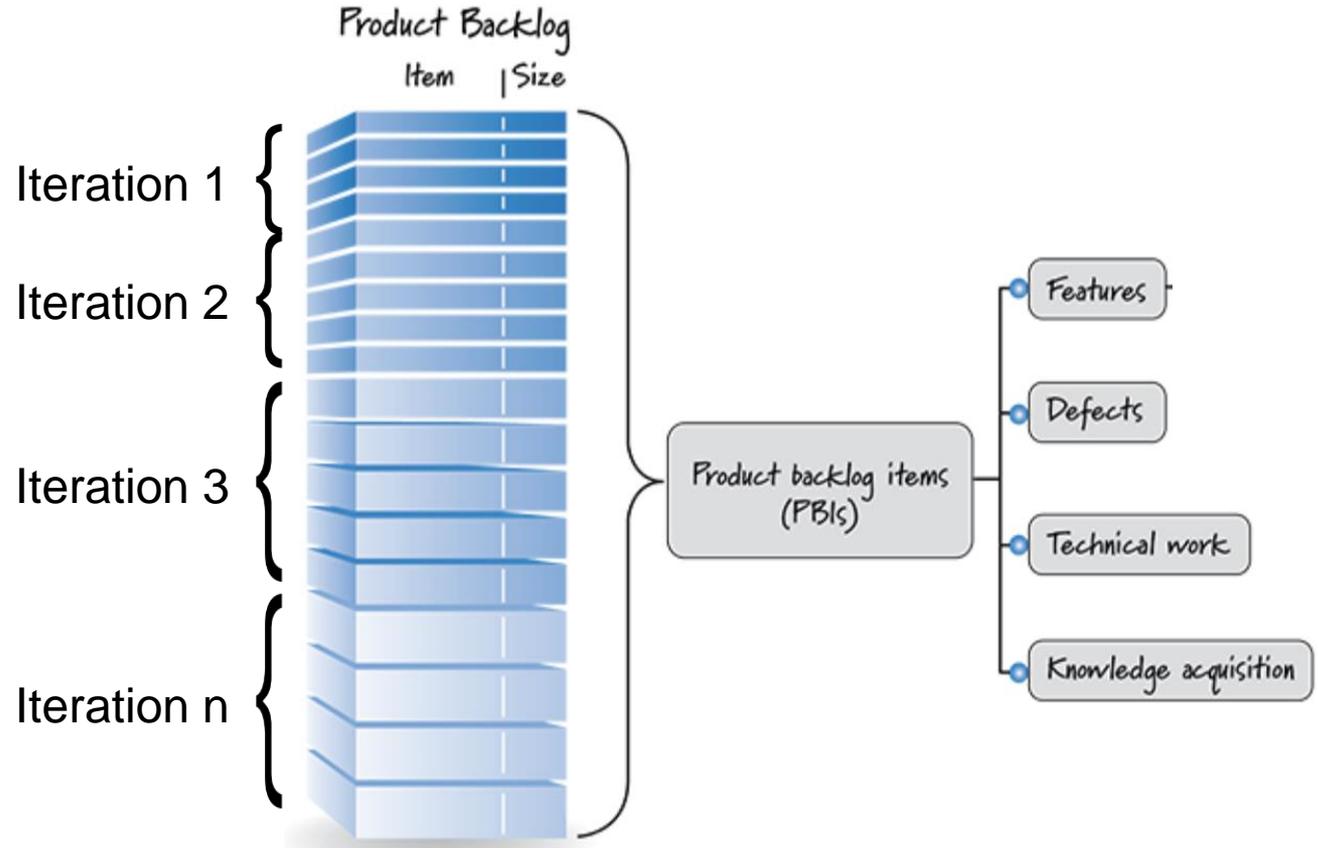
# Agile Requirements: User stories

*As a < type of user >, I want < some goal > so that < some reason >.*

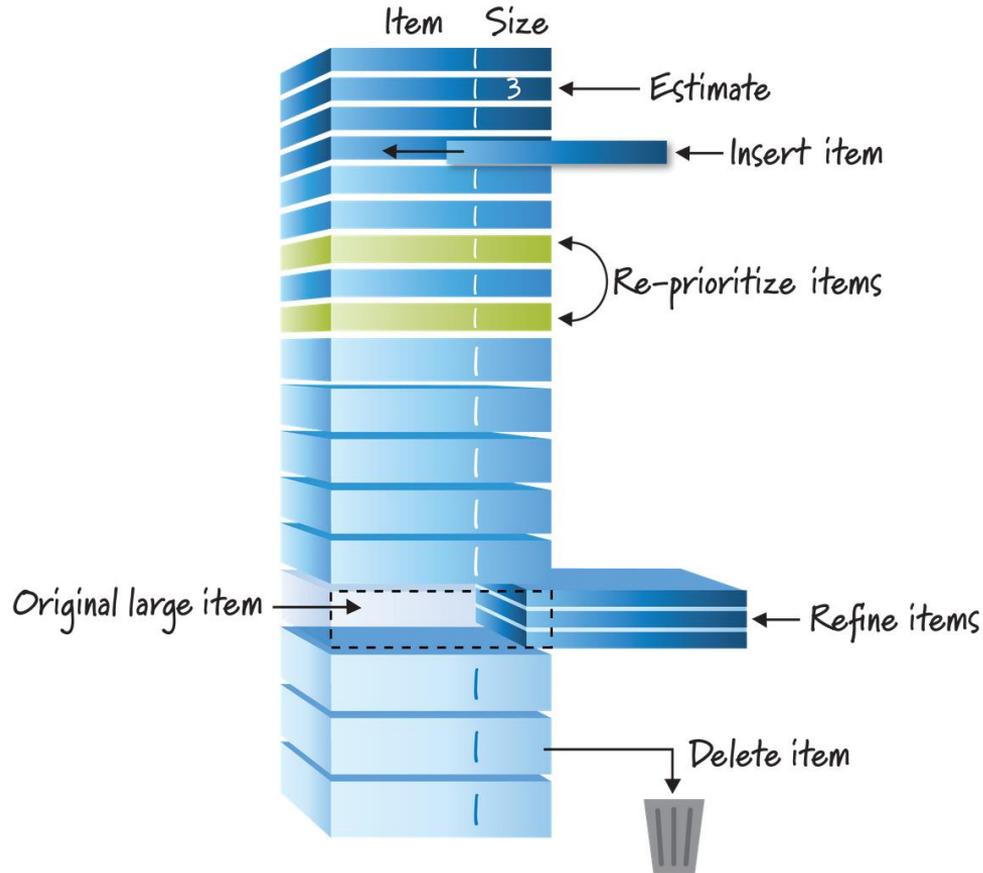
|               |
|---------------|
|               |
|               |
| As a .....    |
|               |
| I need .....  |
|               |
| So that ..... |
|               |



# Agile Requirements: Backlog



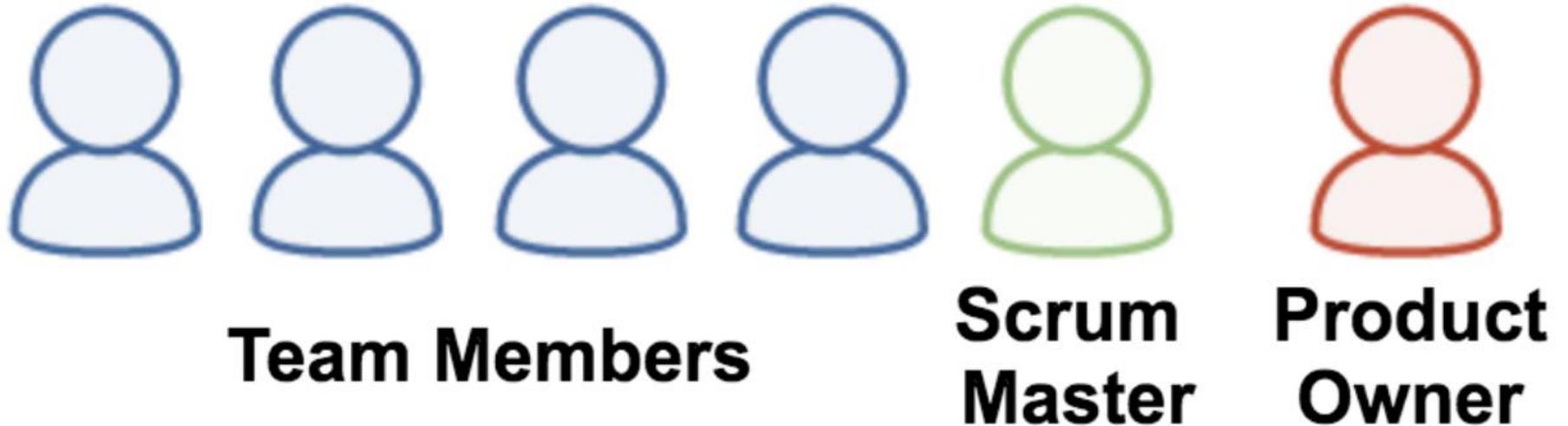
# Agile Requirements: Backlog



# Agile Teams

Self-organising

Multidisciplinary



# Agile Teams

Self-organising

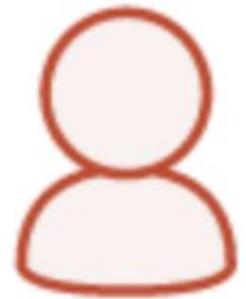
Multidisciplinary



**Team Members**



**Scrum  
Master**



**Product  
Owner**

# 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.



**Scrum  
Master**

# Agile Teams

Self-organising

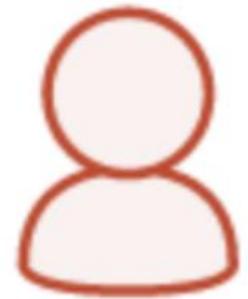
Multidisciplinary



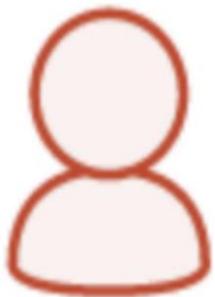
**Team Members**



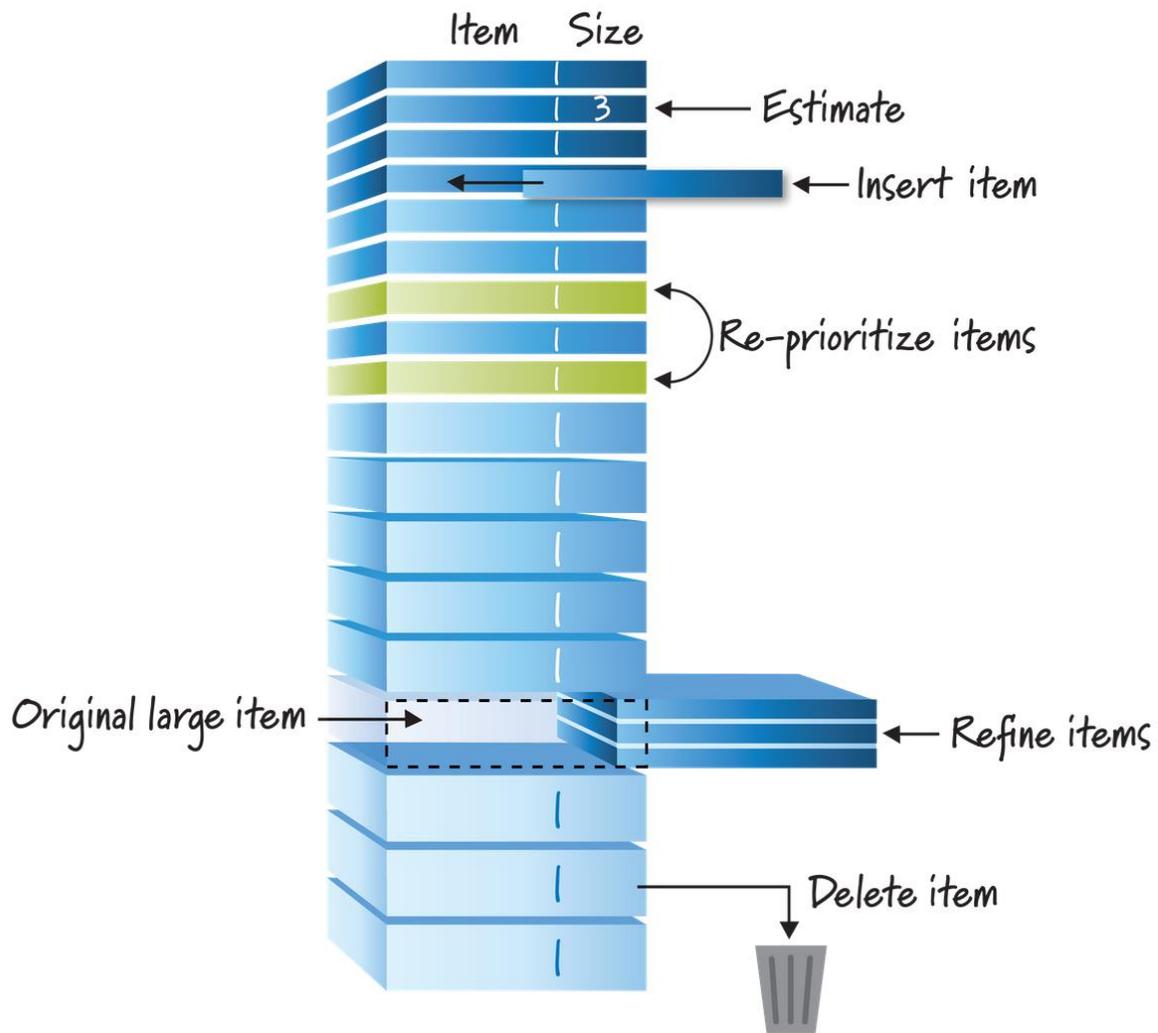
**Scrum  
Master**



**Product  
Owner**



**Product  
Owner**



# Agile Teams

Self-organising

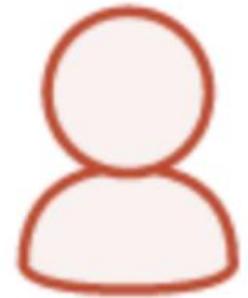
Multidisciplinary



**Team Members**

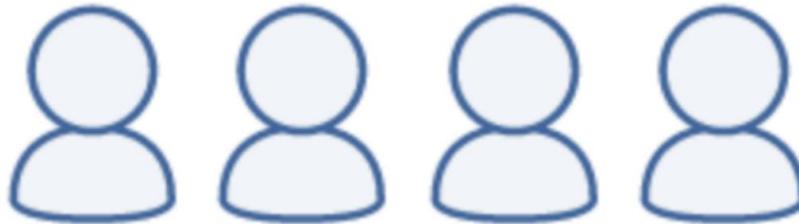


**Scrum  
Master**



**Product  
Owner**

|               |
|---------------|
|               |
| As a .....    |
| I need .....  |
| So that ..... |



## Team Members

- Pair programming
- Unit testing
- Continuous integration
- Refactoring

```
1 import java.util.Scanner;
2 import java.io.PrintWriter; - Library-Import Instruction Set (3 lines)
3 import java.io.File;
4 import java.io.FileOutputStream;
5 public class Main{
6     static int firstInteger = 0;
7     static int secondInteger = 0;
8     static int multiplyResult = 0;
9     public static void main(String[] args){
10         Scanner sc = new Scanner(System.in); func(10); - File-Writer Function-Invocation Instruction
11         System.out.print("first input: ");
12         firstInteger = sc.nextInt(); func(12); - File-Writer Function-Invocation Instruction
13         System.out.print("second input: ");
14         secondInteger = sc.nextInt(); func(14); - File-Writer Function-Invocation Instruction
15         multiplyResult = firstInteger * secondInteger; func(15); - File-Writer Function-Invocation Instruction
16         System.out.println("result: " + multiplyResult); func(16); File-Writer Function-Invocation Instruction
17     }
18     public static void func(int lineNumber) { - File-Writer Function-Declaration Instructions (13 lines)
19         try{
20             PrintWriter writer = new PrintWriter(
21                 new FileOutputStream(new File("states.txt"), true));
22             writer.println("firstInteger:"+firstInteger);
23             writer.println("secondInteger:"+secondInteger);
24             writer.println("multiplyResult:"+multiplyResult);
25             writer.println("lineNumber:"+lineNumber);
26             writer.close();
27         }catch(Exception e){
28             e.printStackTrace();
29         }
30     }
31 }
```

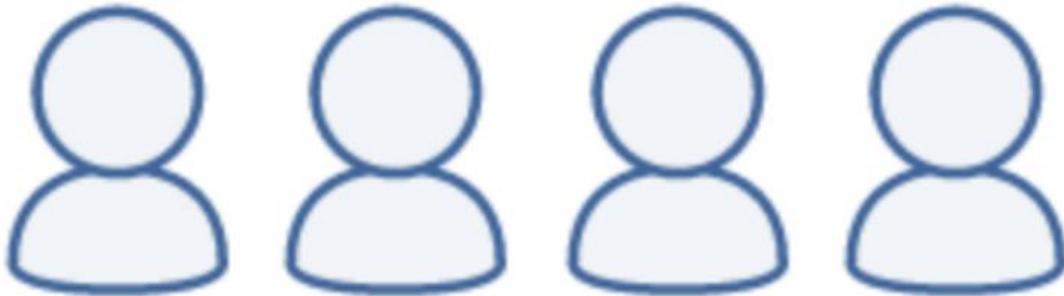
# Agile Teams

Self-organising

Multidisciplinary



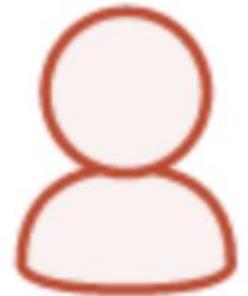
Customer



**Team Members**



**Scrum Master**



**Product Owner**

# Collaboration in Agile Software Development

## Sharing knowledge and ownership

- Co-located
- Pair programming
- Face to face meetings
- Working in one room

# Collaboration in Agile Software Development

## Sharing knowledge and ownership

- **Co-located**
  - Pair programming
  - Face to face meetings
  - Working in one room
- **Distributed**
  - Frequent visits
  - Video chats
  - Scrum-of-scrums
  - Code reviews
  - Rotating team members
  - Multiple communication modes
  - Mirroring

# Agile Meets Scientific Software Development

- time-boxed sprints
- short daily meetings
- release planning
- self-organising team
- user stories
- continuous integration
- refactoring
- pair programming ?
- unit testing ?

# Starting Points with Agile Development

- Assign one person responsibility for the “product” (Product Owner role).

# Starting Points with Agile Development

- Assign one person responsibility for the “product” (Product Owner role).
- Develop a collaboration schedule for synchronising between the Agile Team and source of requirements (Customer role).

# Starting Points with Agile Development

- Assign one person responsibility for the “product” (Product Owner role).
- Develop a collaboration schedule for synchronising between the Agile Team and source of requirements (Customer role).
- Make infrastructural decisions for the whole team
  - Communication tools
  - Source code management
  - Testing frameworks/IDEs/Software versions

# Starting Points with Agile Development

- Assign one person responsibility for the “product” (Product Owner role).
- Develop a collaboration schedule for synchronising between the Agile Team and source of requirements (Customer role).
- Make infrastructural decisions for the whole team
  - Communication tools
  - Source code management
  - Testing frameworks/IDEs/Software versions
- Start small

# Starting Points with Agile Development

- Assign one person responsibility for the “product” (Product Owner role).
- Develop a collaboration schedule for synchronising between the Agile Team and source of requirements (Customer role).
- Make infrastructural decisions for the whole team
  - Communication tools
  - Source code management
  - Testing frameworks/IDEs/Software versions
- Start small

**inspect & adapt**

# Agile Software Development

What is it and do we want it?

Dr Jennifer Ferreira

[jennifer@ecs.vuw.ac.nz](mailto:jennifer@ecs.vuw.ac.nz)

8 November 2019

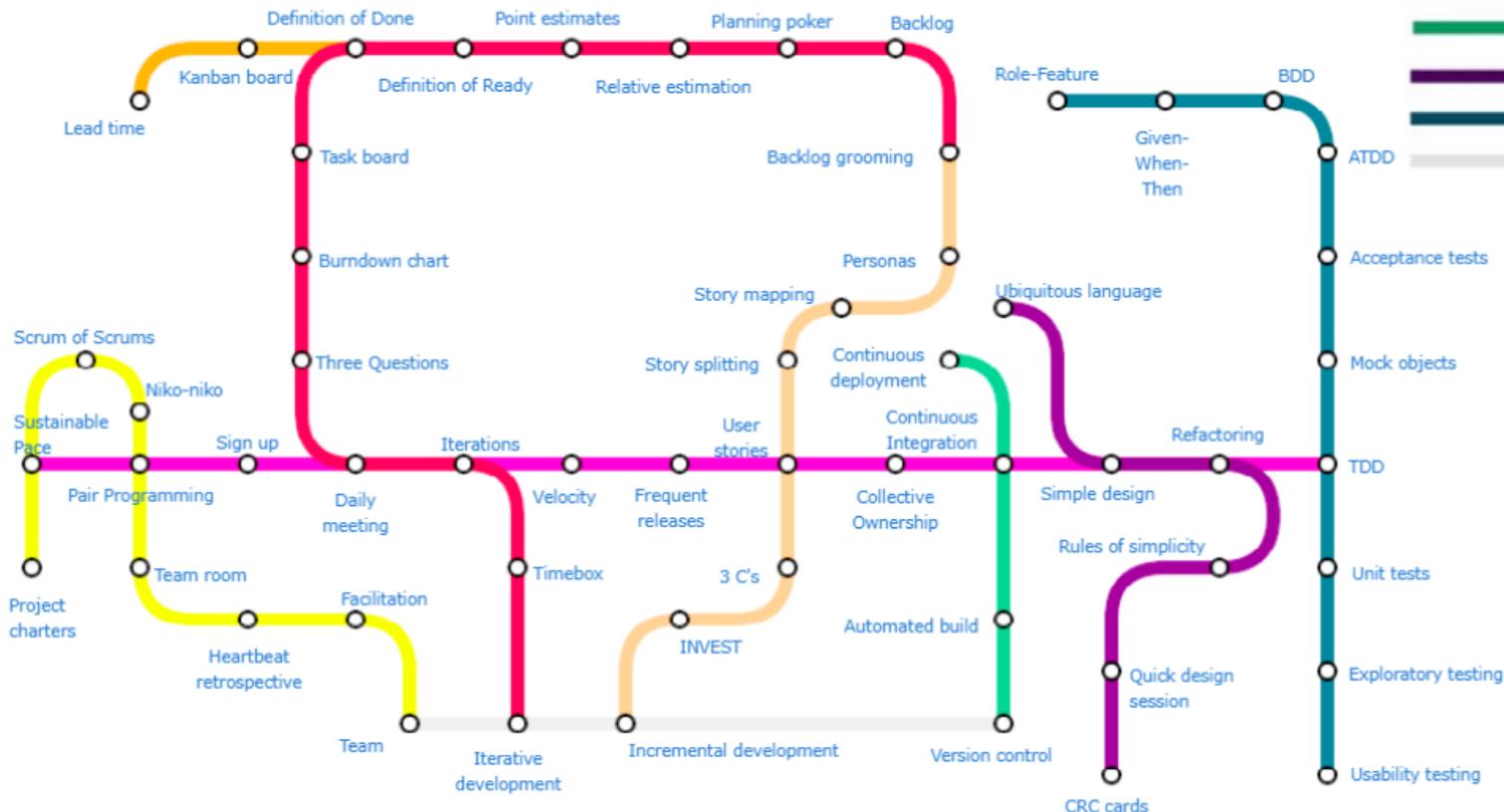
Agile umbrella

Pervasiveness of Agile today vs early 2000's:

<https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8409911>

# Subway Map to Agile Practices

- █ Extreme Programming
- █ Teams
- █ Lean
- █ Scrum
- █ Product management
- █ Devops
- █ Design
- █ Testing
- █ Fundamentals



# Challenges of Scientific SE

<https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=4548403>

<https://journals.plos.org/plosbiology/article?id=10.1371/journal.pbio.1001745>

# Comparing Agile to Scientific SE

Kanban: <https://www.inflectra.com/methodologies/kanban.aspx>

Kanban vs. Scrum: <https://www.atlassian.com/agile/kanban/kanban-vs-scrum>

# Technical Practices

# Stages

Crawl, walk, run, fly