# Agile Estimating

吳行中

# Outlines

- The purpose of planning
- Why planning fails
- Agile planning
- How to estimate?

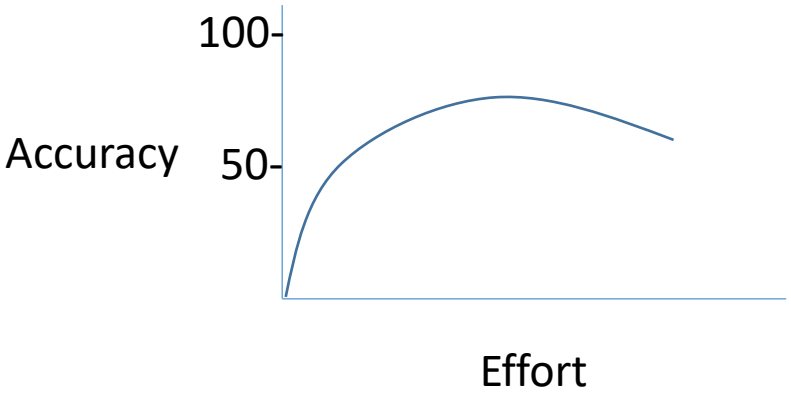
# The Purpose of Planning

- Plans guide our investment decisions.

(Cohn, 2006)

# Two Extremes

- Planning is difficult, and plans are often wrong.
  - No planning
  - Over-planning



# A Good Planning (1)

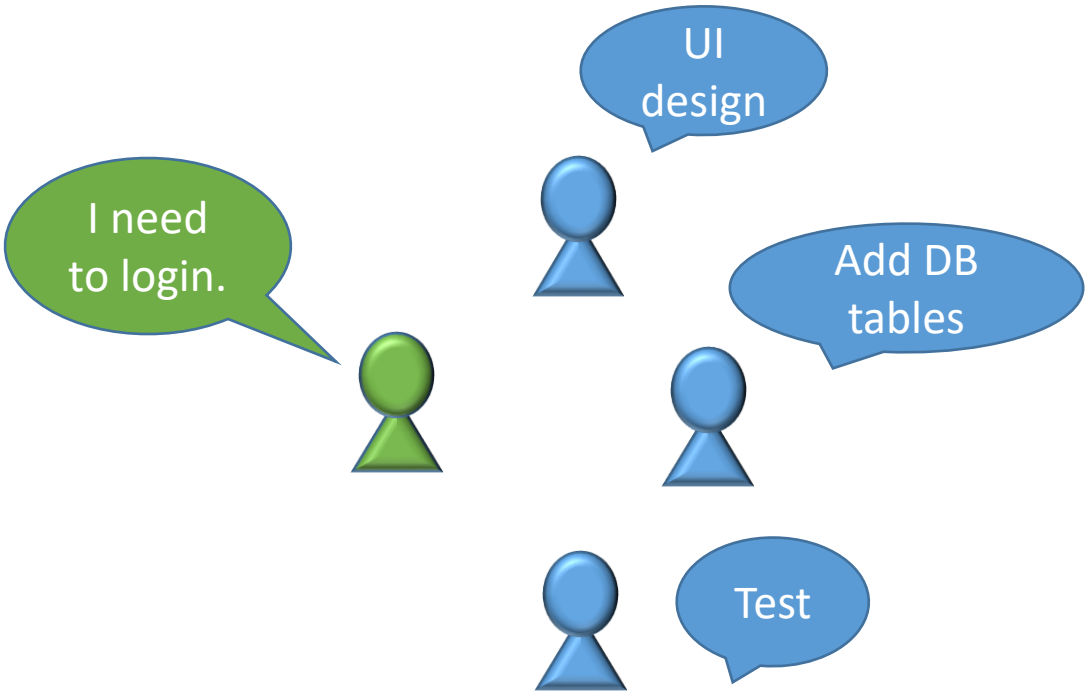
- Reducing risk
  - Expose potential dark corners
- Reducing uncertainty
  - The most critical risk facing most projects is the risk of developing wrong product.
- Supporting better decision making
  - Make sure we are working on the most valuable projects possible.

# A Good Planning (2)

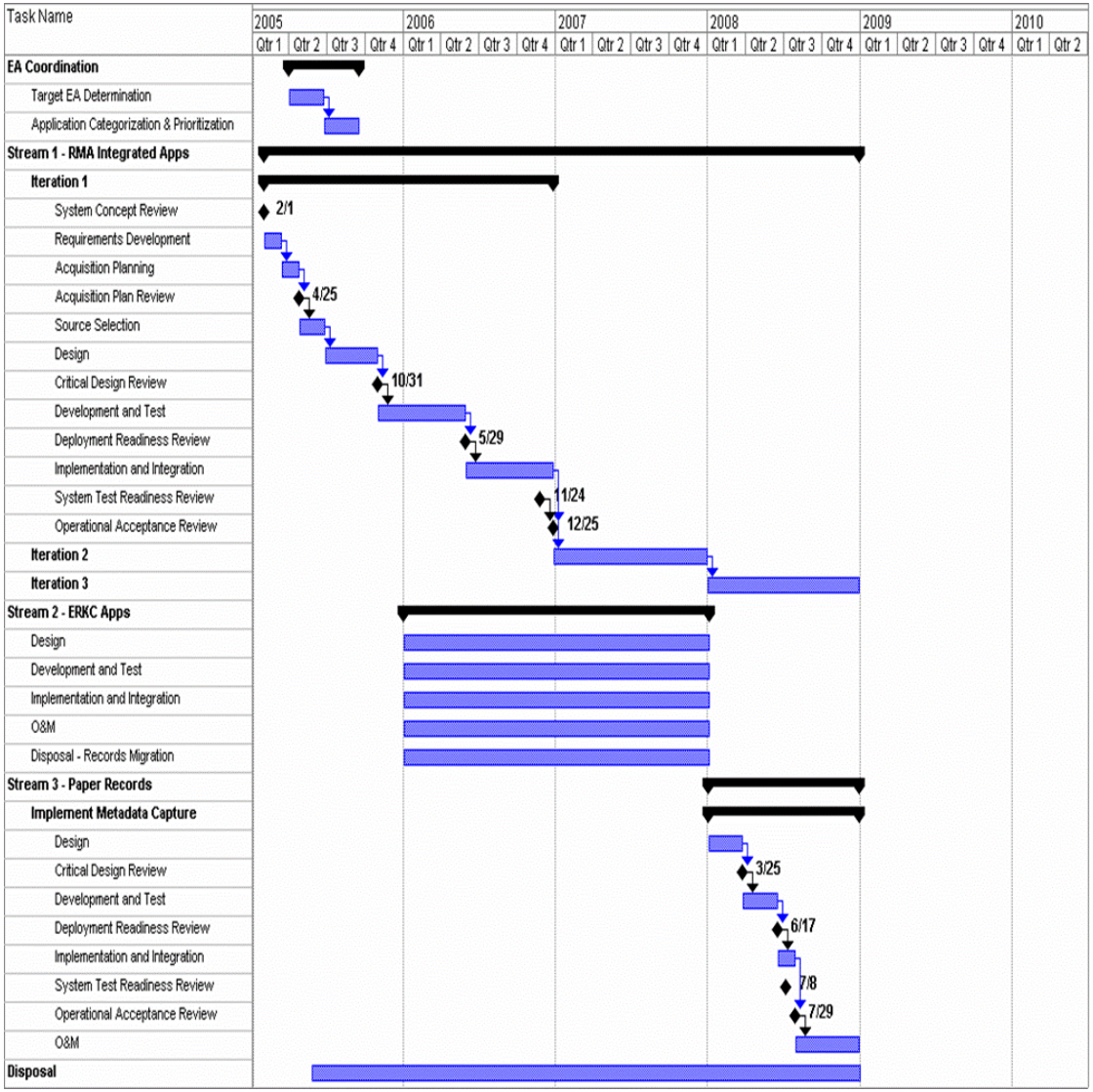
- Establishing trust
  - reliable estimates -> sustainable pace -> high quality code -> reliable estimates
- Conveying information
  - Dose not guarantee an exact set of features
  - Establish a set of baseline expectations

# Why Planning Fails

# Features or Activities



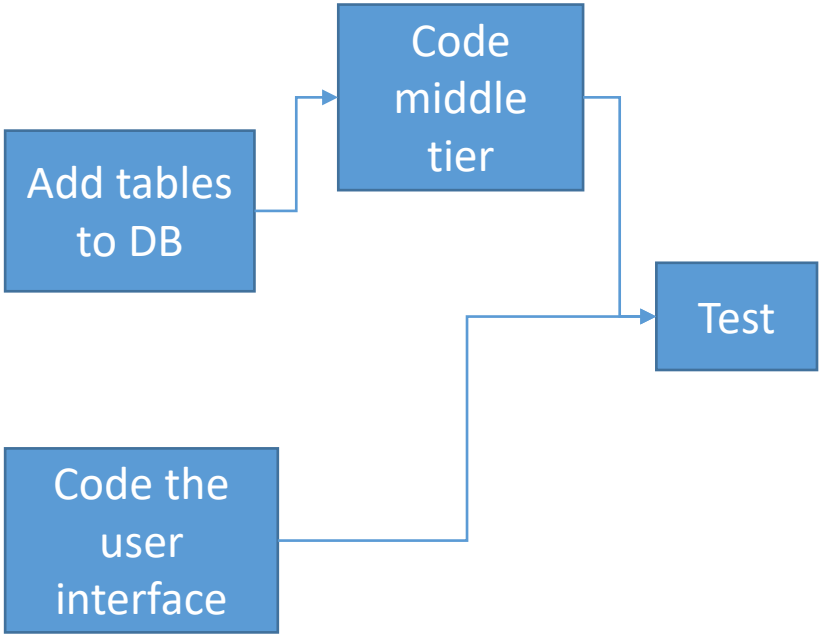
# WBS


Task Name
2005
2006
2007
2008
2009
2010
Qtr 1 Qtr 2 Qtr 3 Qtr 4 Qtr 1 Qtr 2 Qtr 3 Qtr 4 Qtr 1 Qtr 2 Qtr 3 Qtr 4 Qtr 1 Qtr 2 Qtr 3 Qtr 4 Qtr 1 Qtr 2 Qtr 3 Qtr 4 Qtr 1 Qtr 2
EA Coordination
Target EA Determination
Application Categorization & Prioritization
Stream 1 - RMA Integrated Apps
Iteration 1
System Concept Review
Requirements Development
Acquisition Planning
Acquisition Plan Review
Source Selection
Design
Critical Design Review
Development and Test
Deployment Readiness Review
Implementation and Integration
System Test Readiness Review
Operational Acceptance Review
Iteration 2
Iteration 3
Stream 2 - ERKC Apps
Design
Development and Test
Implementation and Integration
O&M
Disposal - Records Migration
Stream 3 - Paper Records
Implement Metadata Capture
Design
Critical Design Review
Development and Test
Deployment Readiness Review
Implementation and Integration
System Test Readiness Review
Operational Acceptance Review
O&M
Disposal
2/1
4/25
10/31
5/29
11/24
12/25
3/25
6/17
7/8
7/29

# Planning by Activities

- Activities don’t finish early
  - Parkinson’s Law
    - Work expands so as to fill the time available for its completion
    - If she finish early, her boss may
      - Accuse her of having given a padded estimate
      - Expect her to finish more activities early

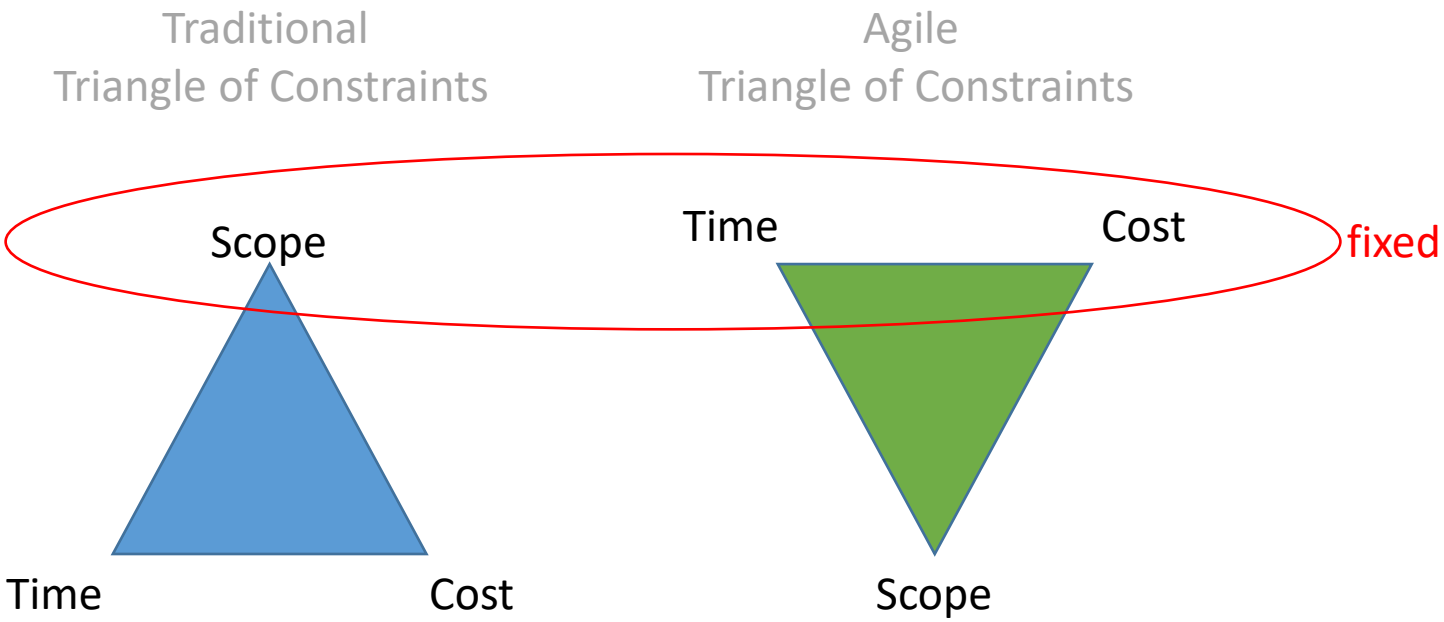
# Lateness is passed down



# Features are not developed by (customers') priority

- With the end of the project approaching, the team scrambles to meet the schedule by dropping features.
  - Some features drooped are of greater value than those that are delivered.

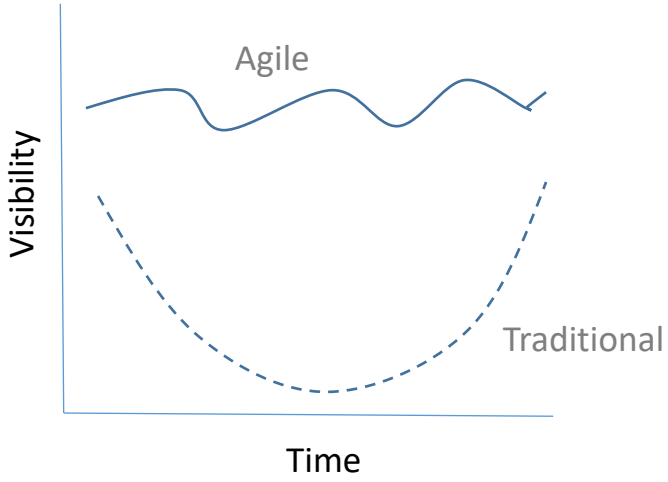
# Constraints

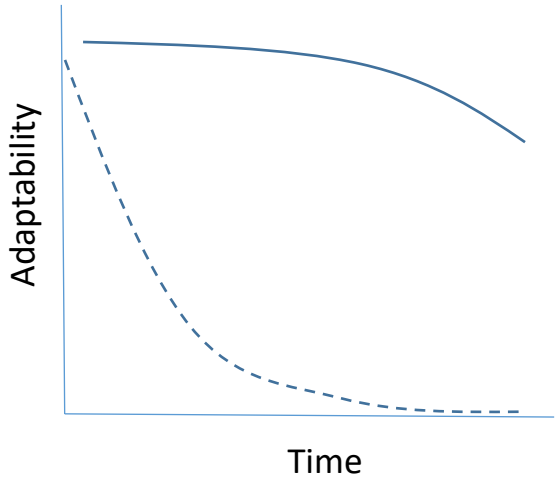


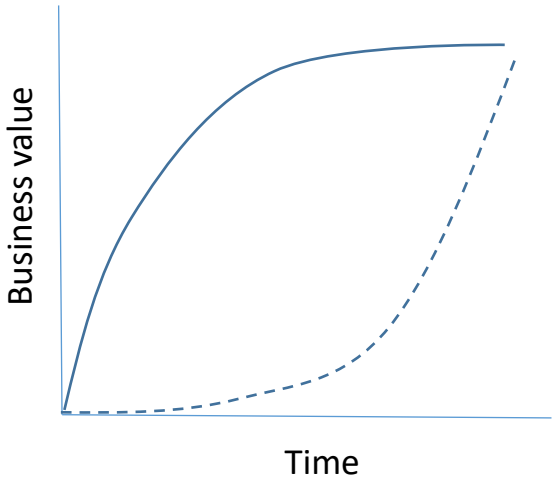

(Griffiths, 2015, p. 26)

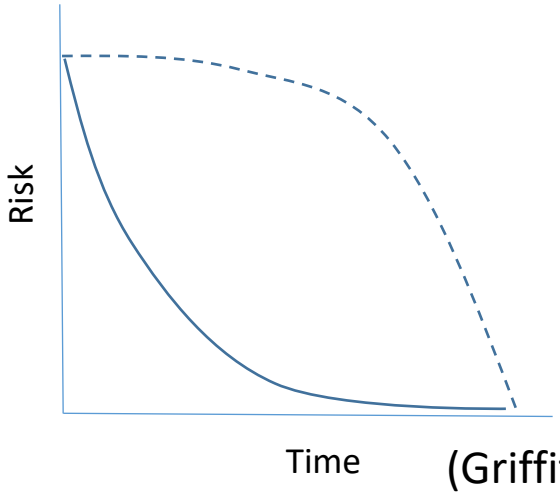
# First thing first

# Agile Value Proposition




(Griffiths, 2015, p. 150)

# Target

Traditional

Agile

# An Agile Approach

- Work as one team
  - Cross-functional team
  - Self-organizing
- Inspect and adapt
  - Work in short iteration
  - Deliver something each iteration
  - Focus on business priorities

# Work as one team

- Aimed at a common goal
- “Throw it over the wall” mentality
  - Analysts -> designers -> coders -> testers
- “We’re-all-in-this-together” mindset
  - Although an agile team should work together as one whole team, there are number of specific roles.

# Scrum

# 人是制度的產物

- 基本歸因錯誤 (Fundamental Attribution Error)
  - 我犯錯是因為環境
  - 別人犯錯是因為個性
- Scrum 不會要大家指責別人或挑出錯誤
  - 而是促使大家集中共識與完成工作來獎勵良好行為。

(Sutherland, 2015, p. 95)

# Cross-functional Team

- 你不會只從自己的角度來思考

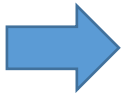
# Cross-functional Team

- 具備所有技能
- 在同一個房間共事
- 分享所有情報

(Sutherland, p. 83)

# Self-organizing

- 自主
  - 成就感
  - 激發人類底層的積極性



Pride-in-work

Pride-in-accomplishment

Pride-in-contribution

- Put real choice on the table
  - Commitment comes from having choice. (Block, 2011, p.44)

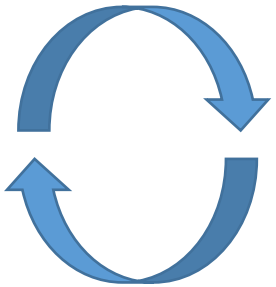
# 達到卓越之境

- 在適切的時候出現在適切的地點
- 一開始把球踢過去的前鋒，想必深信他的隊友會來到應該來到的地點。

(Sutherland, p. 101)

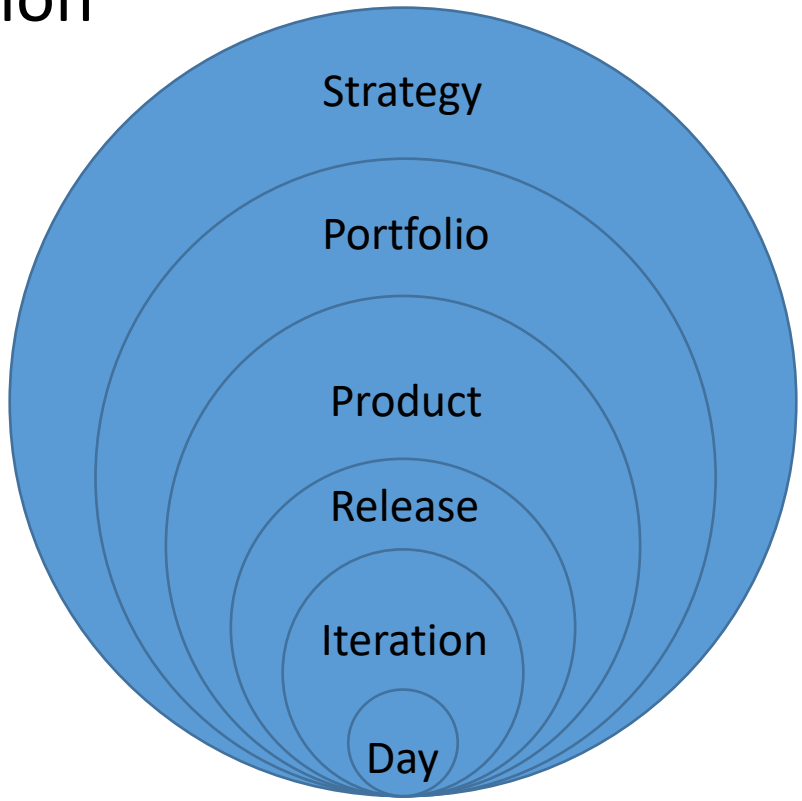
# Inspect and Adapt

- An agile team works in short iteration
  - 2-4 weeks long
- Deliver something each iteration
- Focus on business priorities
  - User story
- Feedback
  - Relatively stable
  - Maximize the return on the project investment



# Multiple Levels for Planning

- Planning onion



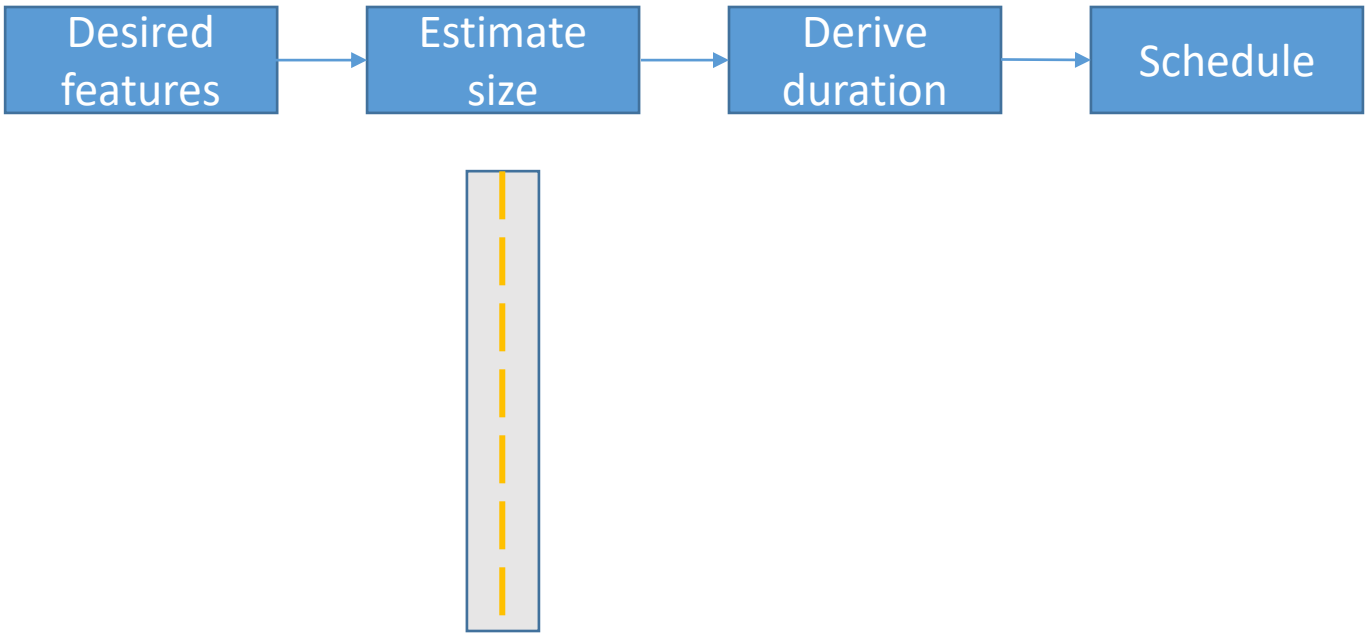

Agile teams plan at least at the release, iteration and day levels.

(Cohn, 2006, p. 28)

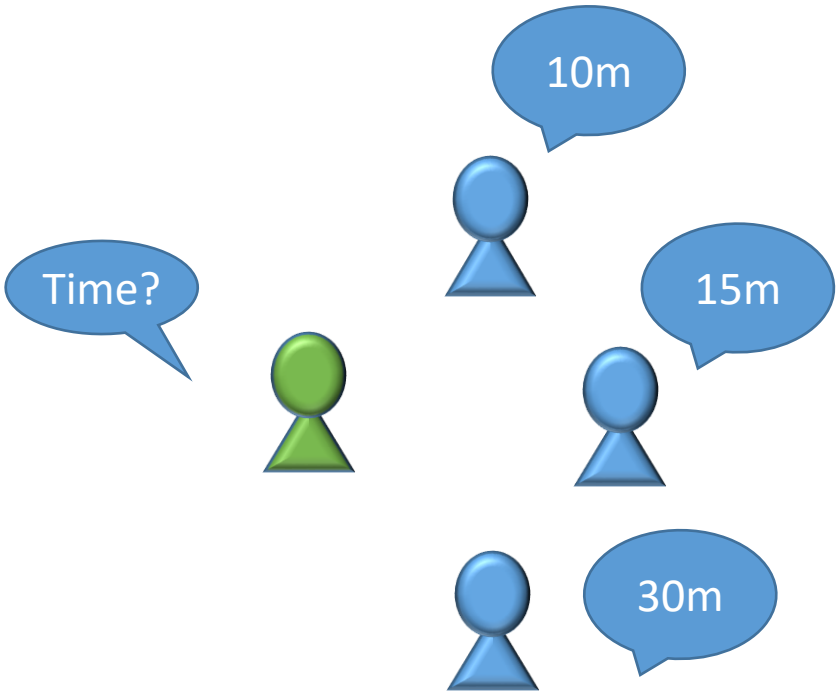
# How to estimate?

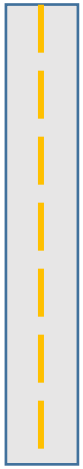
- Educated Guess (有根據的猜)
  - Expert opinion
  - Analog
  - Split

# Size or Duration?



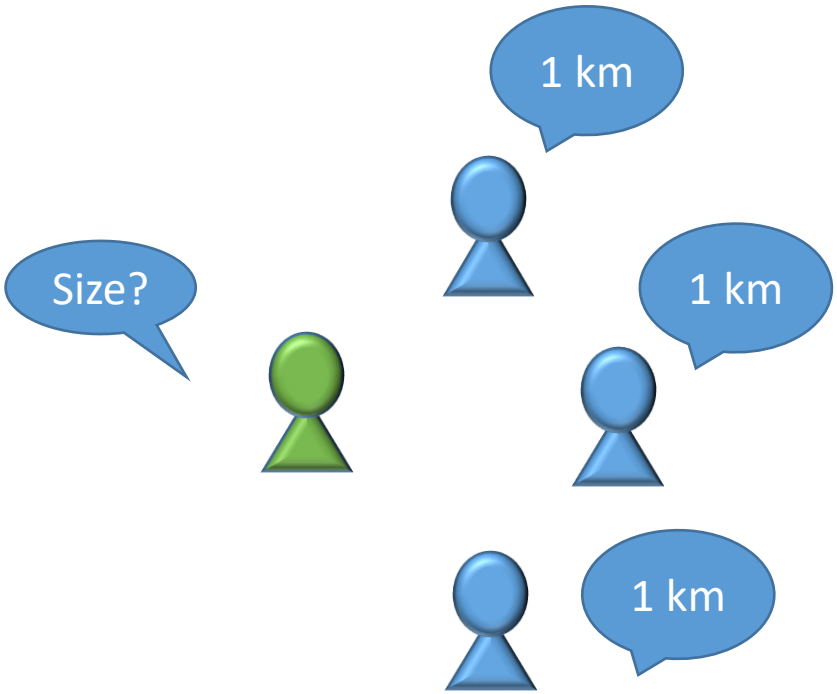
# Ideal Days

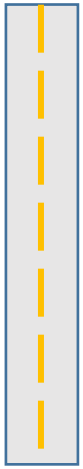


We are probably both right.

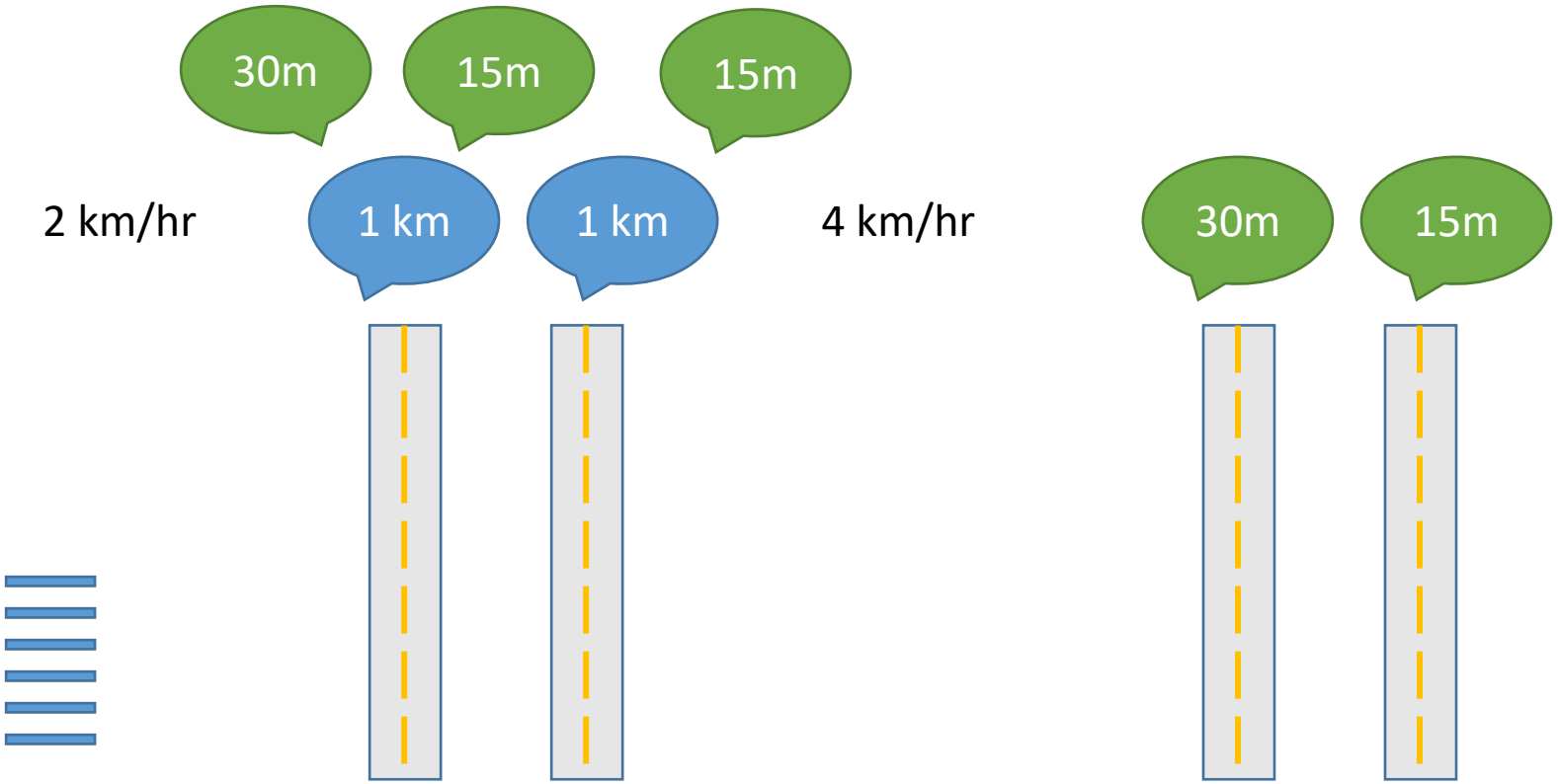
# Story points



Their discussion of trail size are meaningful.

# Story points estimates do not decay



# Drive cross-functional Behavior

- Story points
  - We can estimate story point by **analog** only
    - 2 developer of different skill or experience can agree on the size while disagreeing about how long it will take to do.
  - High-level discussion about every thing
- Ideal days
  - My ideal days are not your ideal days
  - How long “their part”
  - A little deeper discussion

(Chon, p.69-75)

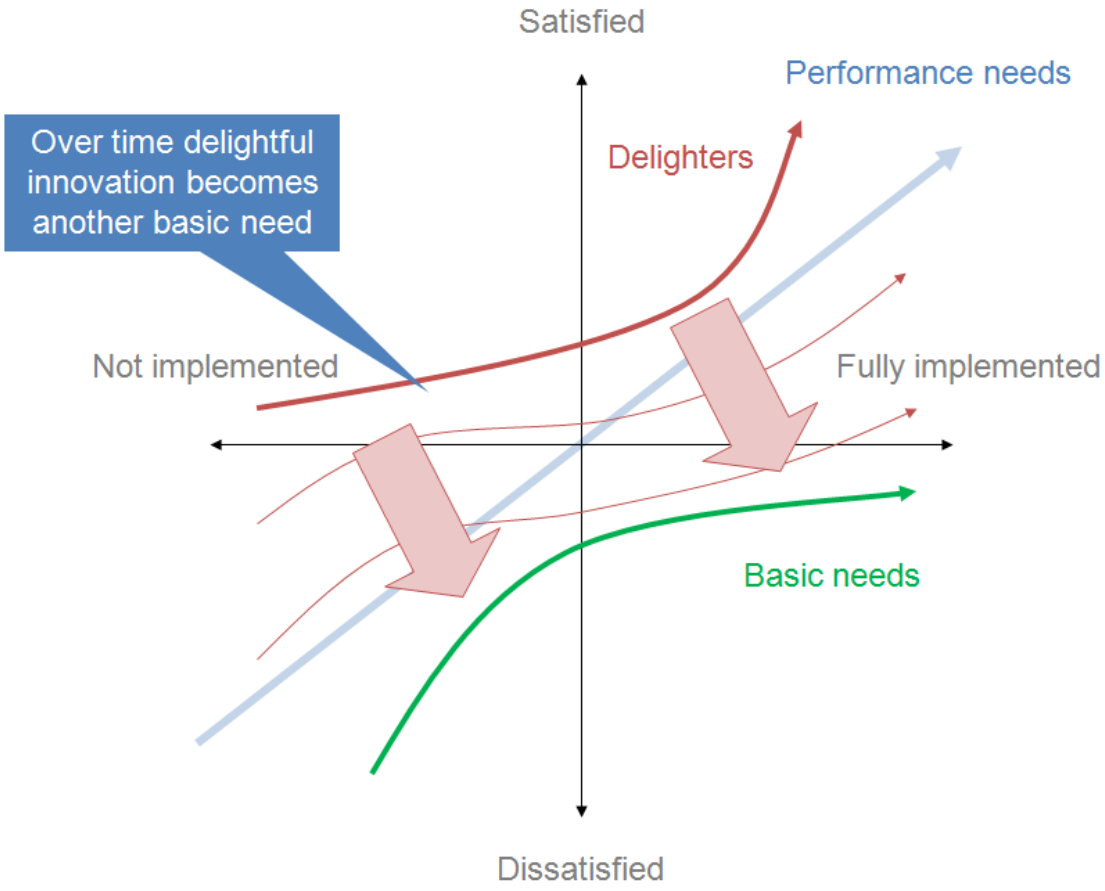
# Financial Prioritization

| Theme | Story Points | Cost | NPV | ROI | Discounted Payback Period |
|---|---|---|---|---|---|
| Overnight service | 150 | 101250 | 46341 | 45% | 7 quarters |
| Custom reporting | 90 | 60750 | 34533 | 15% | 6 quarters |
| Partner integration | 60 | 40500 | 30013 | 49% | 3 quarters |

Making a decision is not cut and dried.

(Cohn, p. 108)

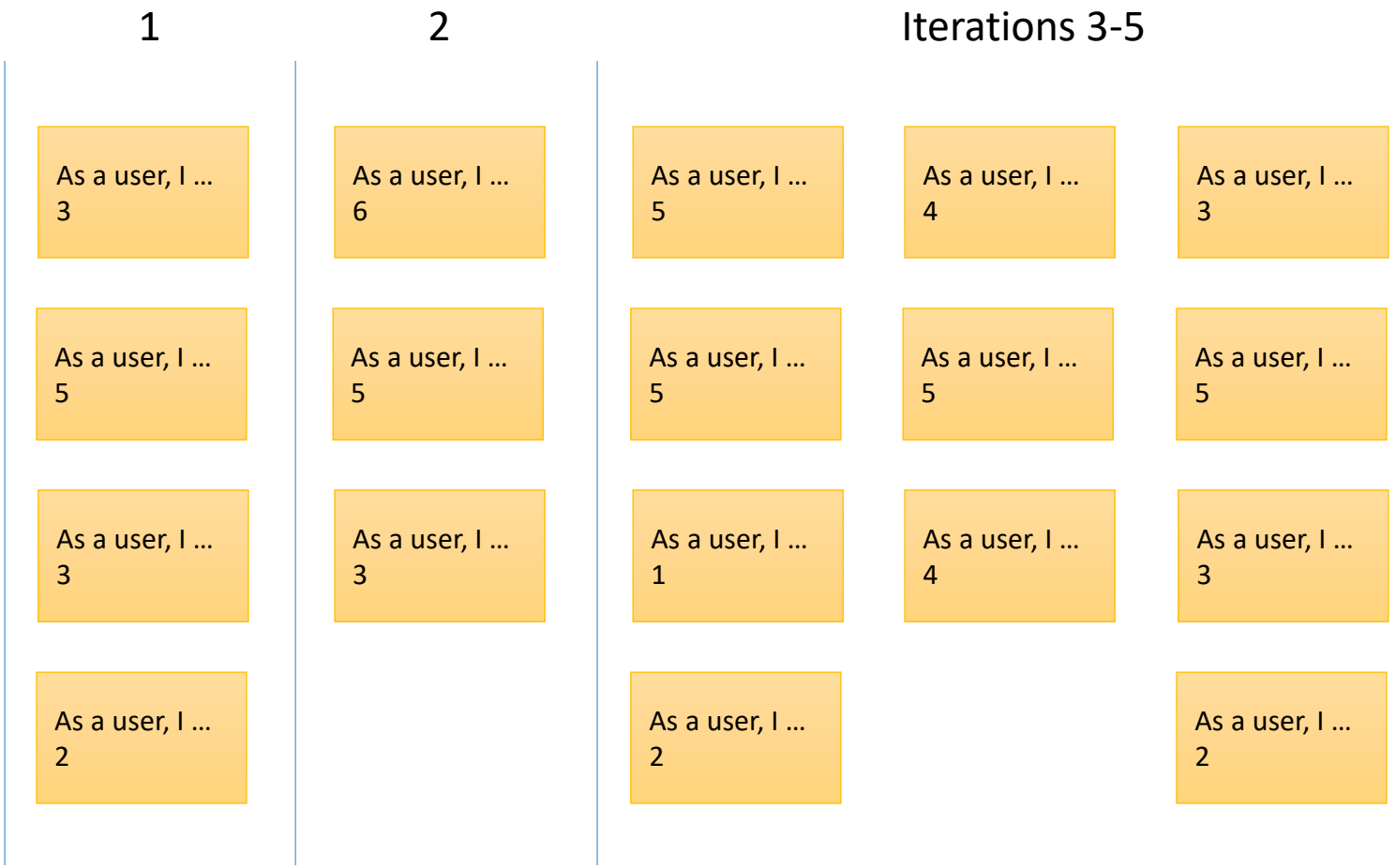
# Kano Model


Satisfied
Performance needs
Over time delightful innovation becomes another basic need
Delighters
Not implemented
Fully implemented
Basic needs
Dissatisfied

# Factors in Prioritization

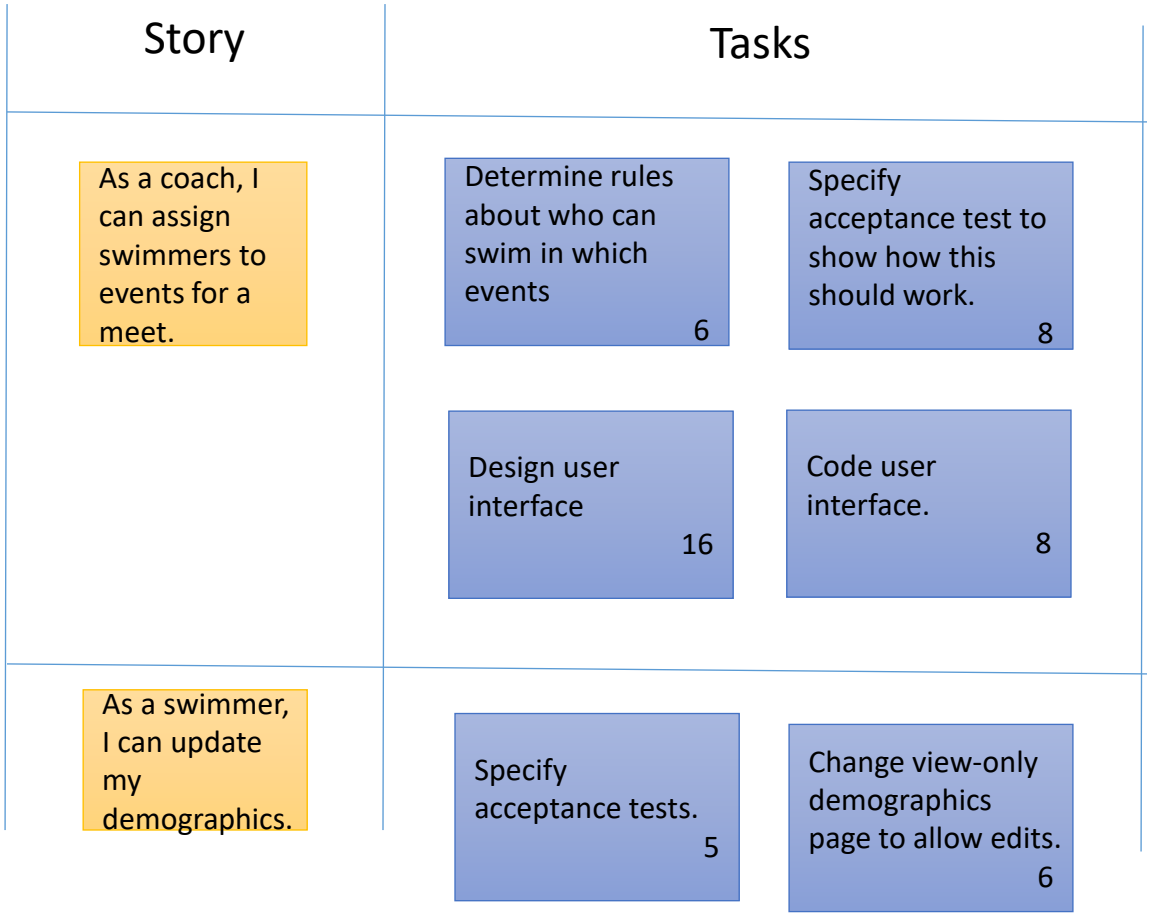
- Financial value
- Cost of developing (and perhaps supporting) new features
- New knowledge
  - What -> end uncertainty
  - How -> means uncertainty
- Risk
  - Schedule
  - Cost
  - Functional

# Arranging iteration in columns



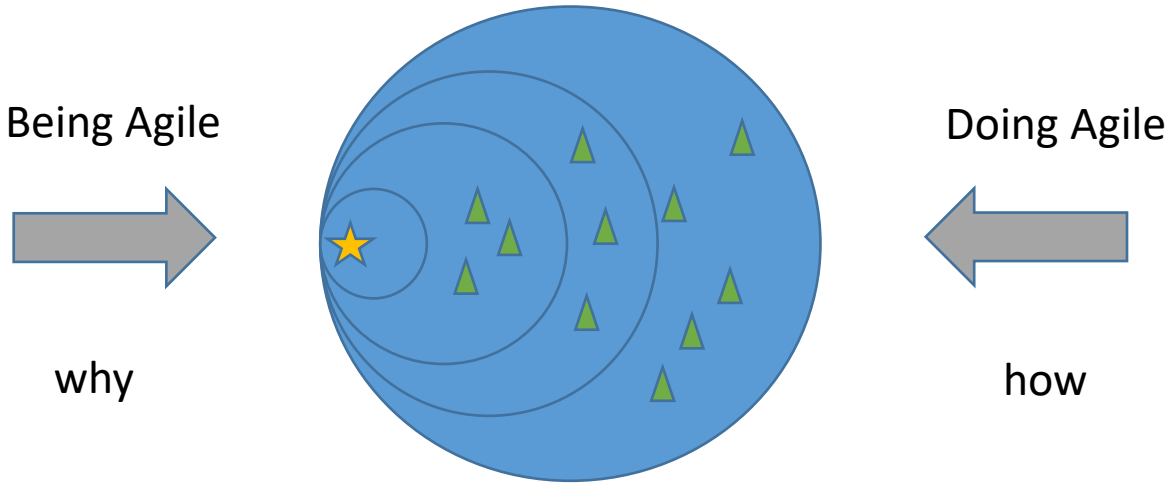

(Cohn, p. 138)

# Tasks




(Cohn, p. 147)

# Being Agile versus Doing Agile




(Griffiths, 2015, p. 24)

# Recommends

# Mindset

- Seek first to understand, then to be understood
  - Feature, paying customers, real users
- Think win-win
- Be proactive
- Begin with the end in mind

# 善用 retrospective

- Sharpen the saw
- Self-organizing
  - 創造自己的流程
- Team-building
- Respect
  - 80%的完美

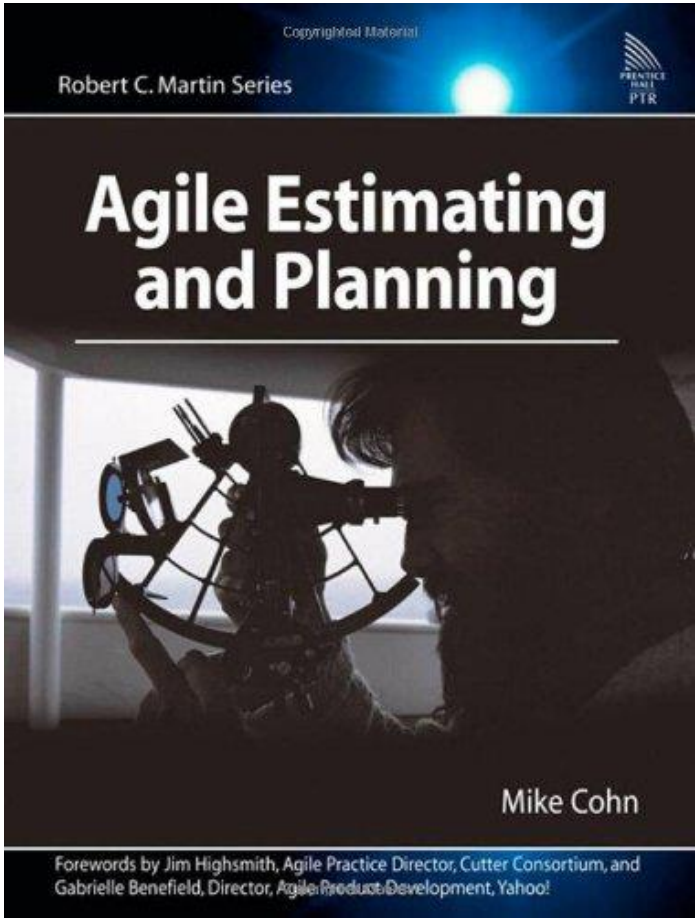
# 給自己時間、給別人時間

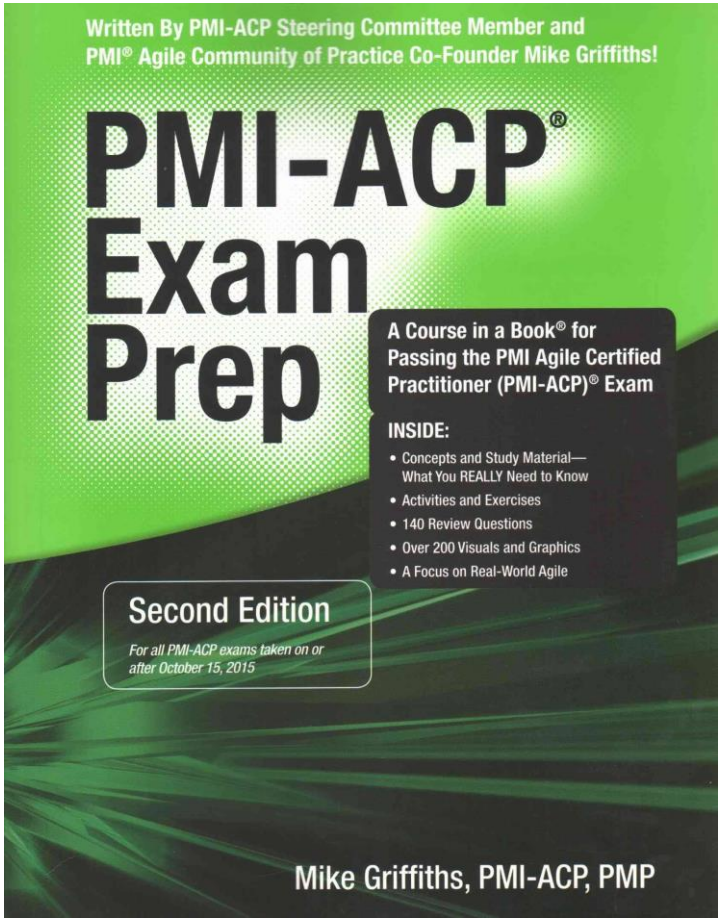
- 尊重歷史
  - 認識與目標的差距
  - 認識自己的特殊情境
- 不要急
  - 不要覺得是任何人的錯
  - 不一定要一步到位
- 建樓梯

# 化繁為簡

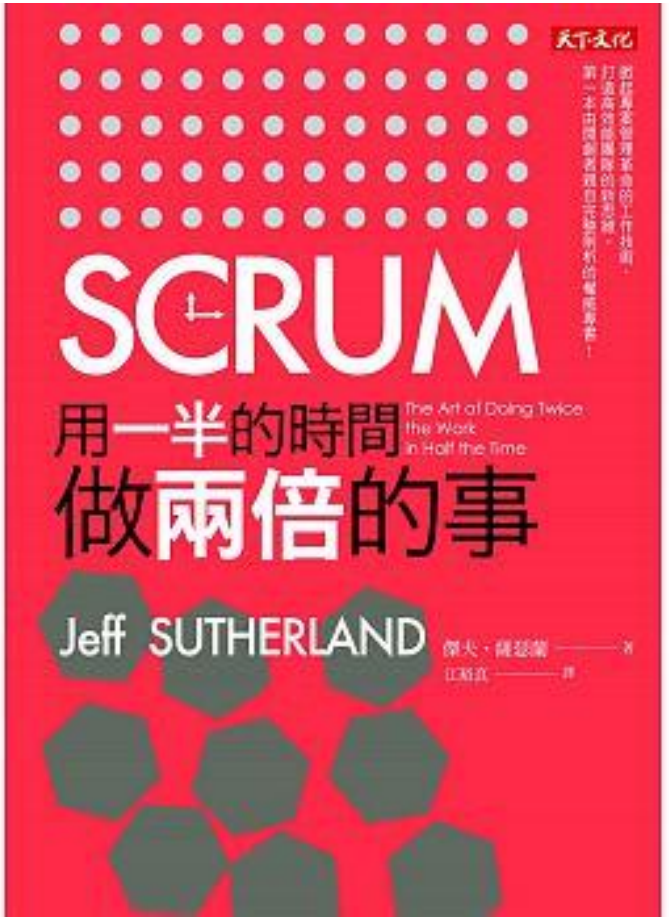
- 回想規則背後的理由
- 平衡考量
- 用agile, 不被agile所用
- 但可以從 Scrum Guide 出發

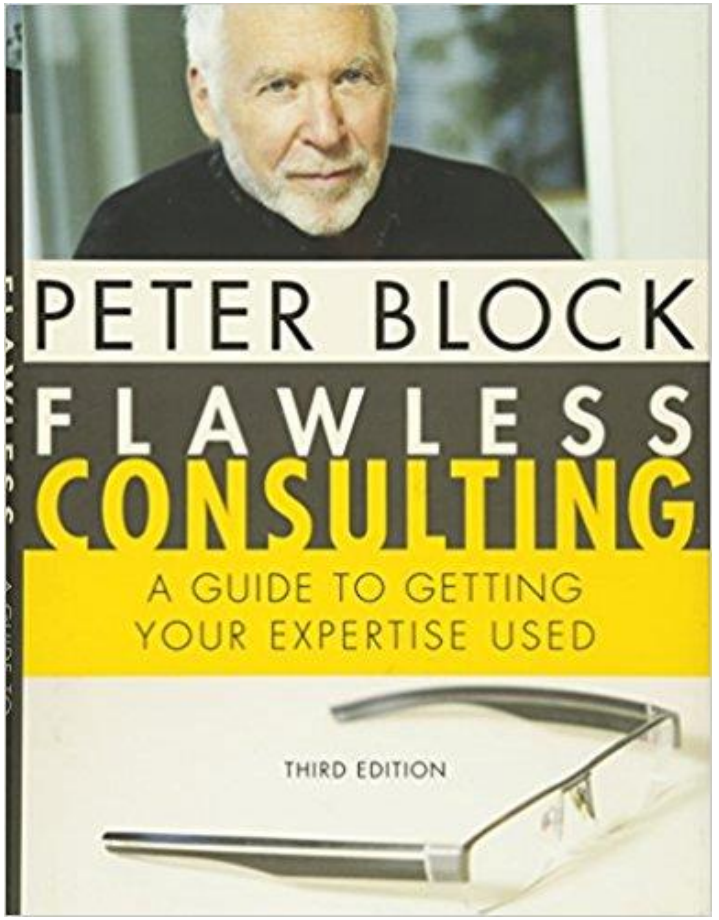
# Reference


Copyrighted Material
Robert C. Martin Series
PRENTICE HALL PTR
Agile Estimating and Planning
Mike Cohn
Forewords by Jim Highsmith, Agile Practice Director, Cutter Consortium, and Gabrielle Benefield, Director, Agile Product Development, Yahoo!


Written By PMI-ACP Steering Committee Member and PMI® Agile Community of Practice Co-Founder Mike Griffiths!
PMI-ACP® Exam Prep
A Course in a Book® for Passing the PMI Agile Certified Practitioner (PMI-ACP)® Exam
INSIDE:
• Concepts and Study Material—What You REALLY Need to Know
• Activities and Exercises
• 140 Review Questions
• Over 200 Visuals and Graphics
• A Focus on Real-World Agile
Second Edition
For all PMI-ACP exams taken on or after October 15, 2015
Mike Griffiths, PMI-ACP, PMP

# Reference


SCRUM
用一半的時間做兩倍的事
The Art of Doing Twice the Work in Half the Time
Jeff SUTHERLAND


PETER BLOCK
FLAWLESS CONSULTING
A GUIDE TO GETTING YOUR EXPERTISE USED
THIRD EDITION

# Failure Modes

- We make mistakes
- We prefer to fail conservatively
- We prefer to invent rather than research
  - Especially engineers
- We are creatures of habits
- We are inconsistent

# Success Modes

- We are good at looking around
- We are able to learn
- We are malleable
- We take pride in our work
  - We are able outside of our job description to repair or report an issue, because it is the right thing to do for the project.

# Success Strategies

- Balance discipline with tolerance
- Start with something concrete and tangible
- Copy and alter
- Watch and listen
- Support both concentration and communication
  - Flow

# Success Strategies

- Match work assignments with person
- Retain the best talent
- Use rewards that preserve joy
    - Pride-in-work
    - Pride-in-accomplishment
    - Pride-in-contribution
- Combine rewards
- Get feedback
    - A little bit of feedback can replace a lot of analytical work.

(Griffiths, 2015, pp. 328-330)