

Agile Tutorial

Agenda



- Agile Introduction
 - Scrum
 - XP
- LEAN Introduction
 - Kanban
- Recap



Agile Introduction

What is Agile

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.

Kent Beck
Mike Beedle
Arie van Bennekum
Alistair Cockburn
Ward Cunningham
Martin Fowler

James Grenning
Jim Highsmith
Andrew Hunt
Ron Jeffries
Jon Kern
Brian Marick

Robert C. Martin
Steve Mellor
Ken Schwaber
Jeff Sutherland
Dave Thomas



Nice To Meet You...

FEB 2001: 17 THINKERS WRITE
THE AGILE MANIFESTO



Individuals and interactions over processes and tools



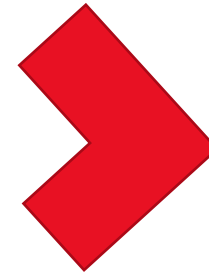
Working software over comprehensive documentation



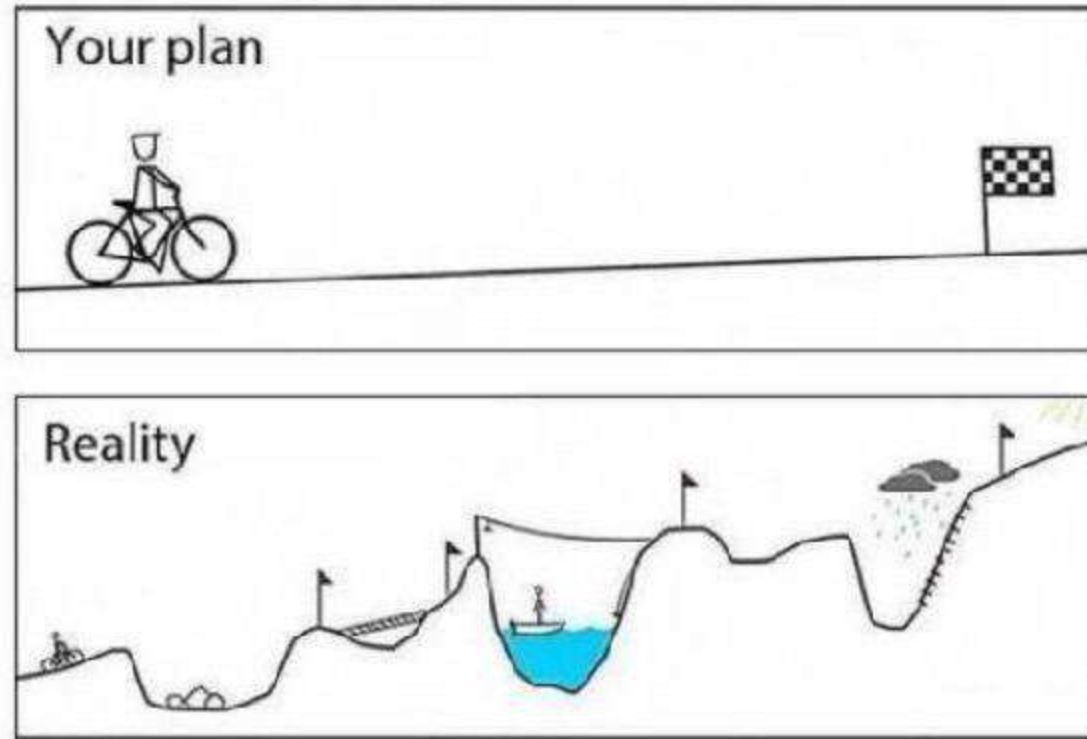
VS



Customer collaboration over contract negotiation



Responding to change over following a plan



Agile Principles

1. Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.
2. Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.
3. Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.
4. Business people and developers must work together daily throughout the project.
5. Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.
6. The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.

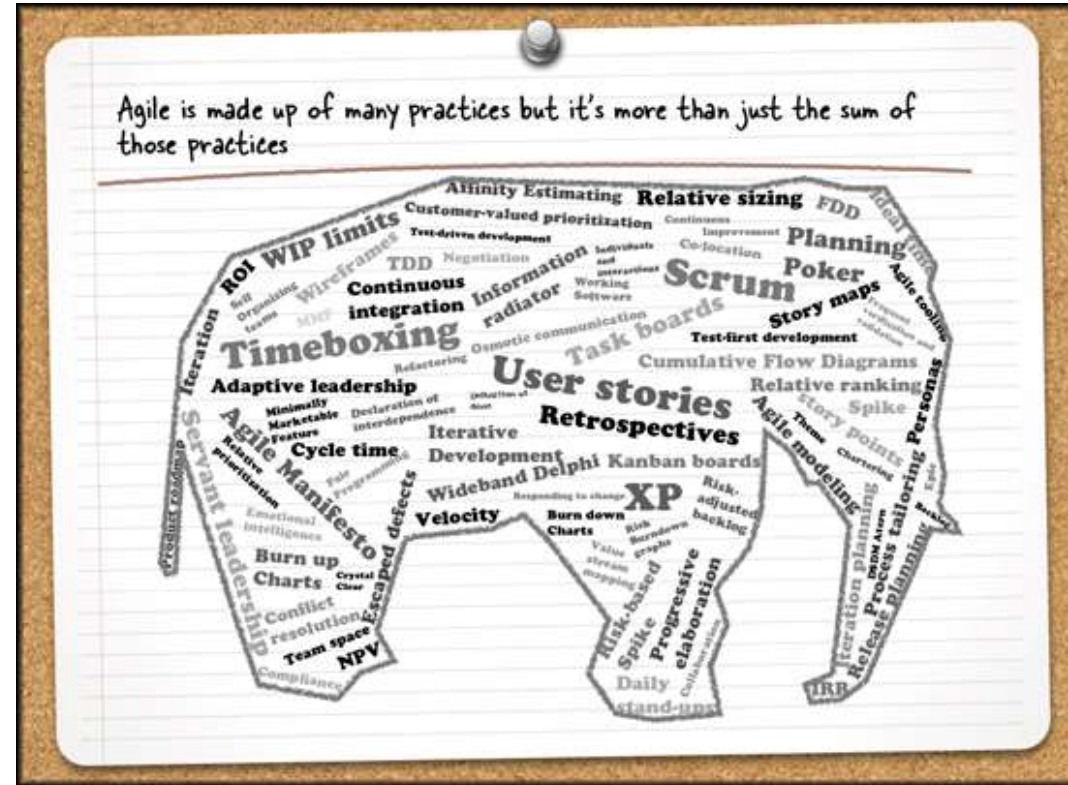
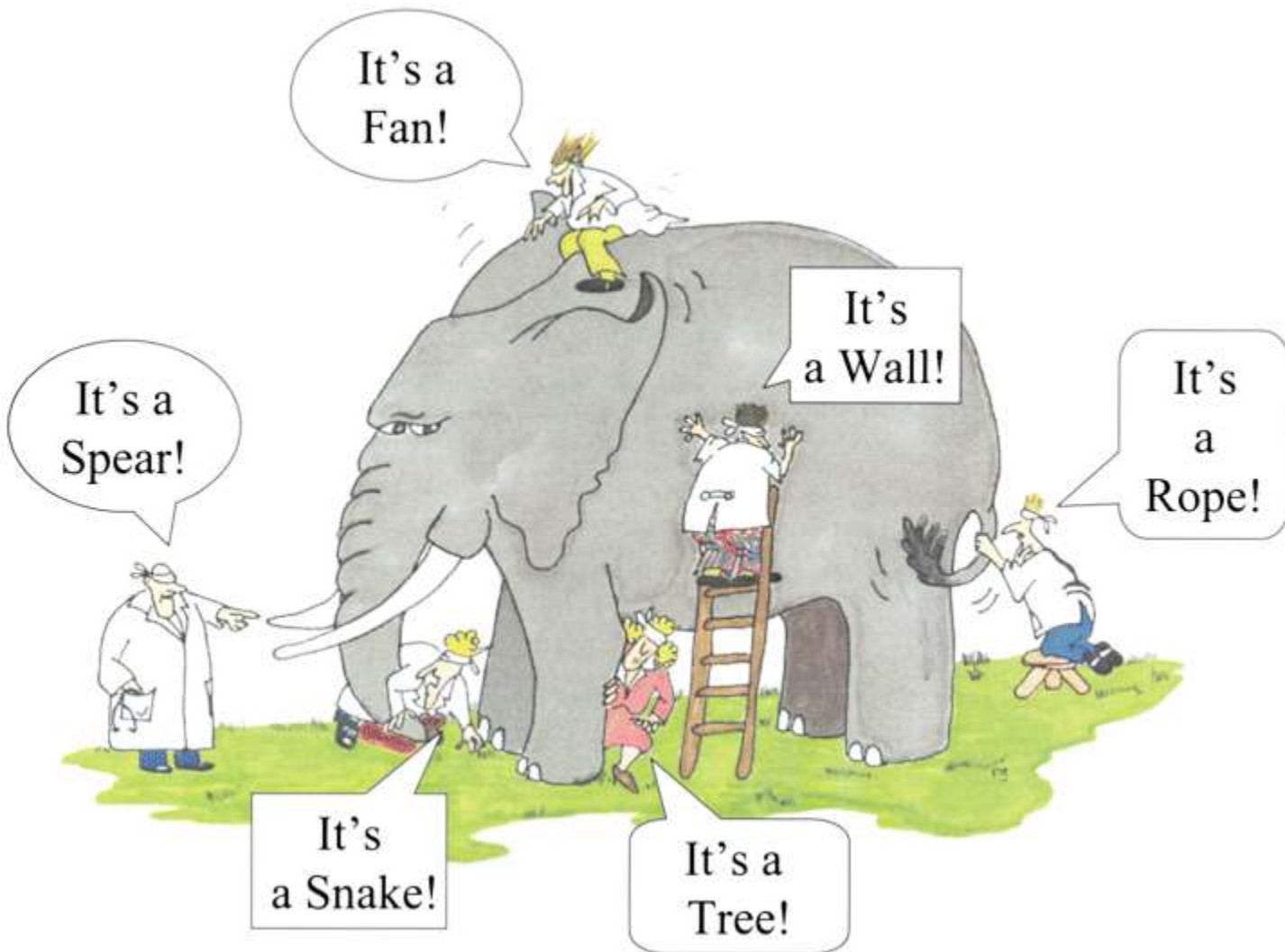


Agile Principles(Continues)

7. Working software is the primary measure of progress.
8. Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.
9. Continuous attention to technical excellence and good design enhances agility.
10. Simplicity — the art of maximizing the amount of work not done — is essential.
11. The best architectures, requirements, and designs emerge from self-organizing teams.
12. At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.



Agile Elephant

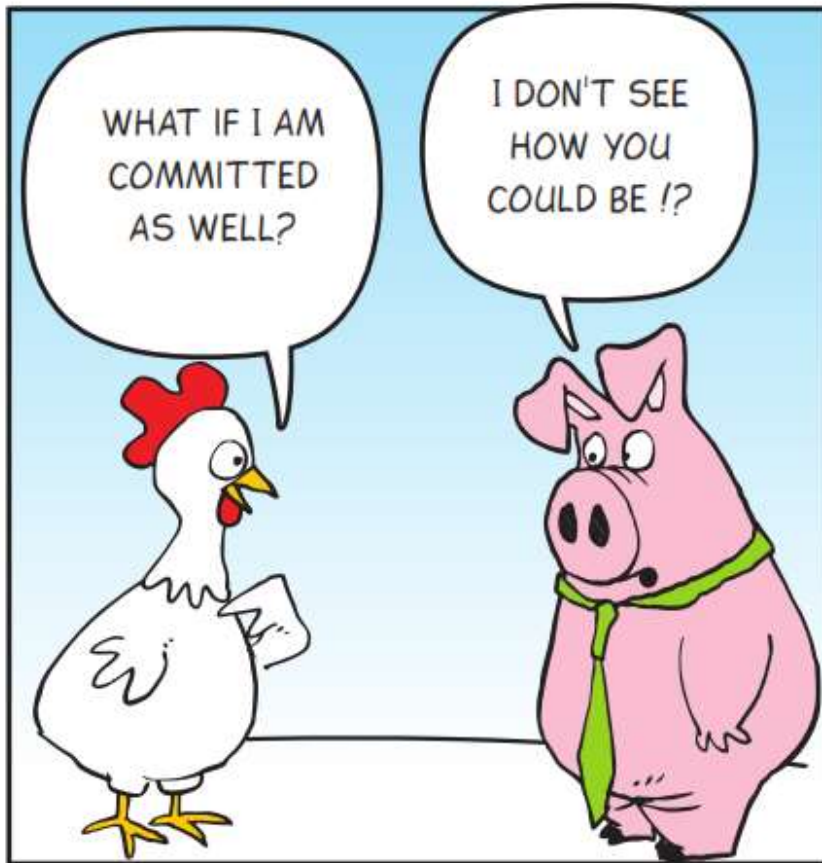


Chicken and Pig

Agile Safari™

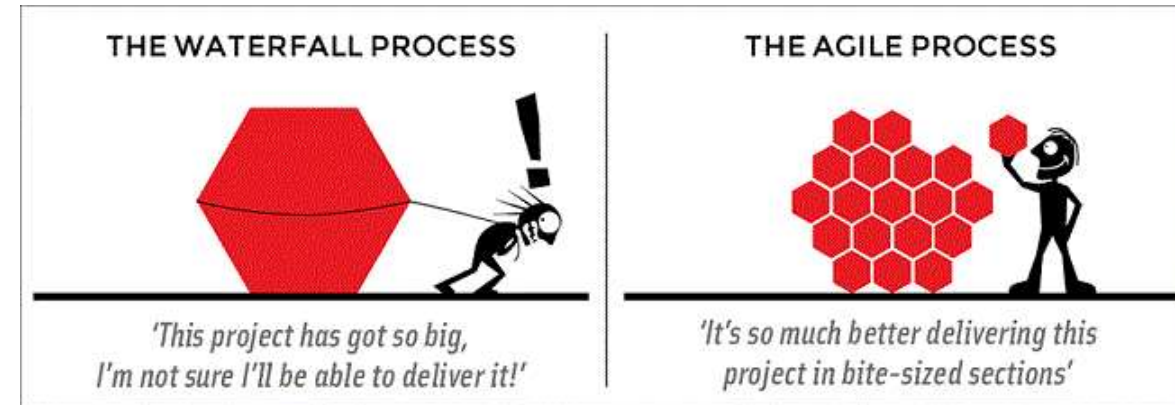
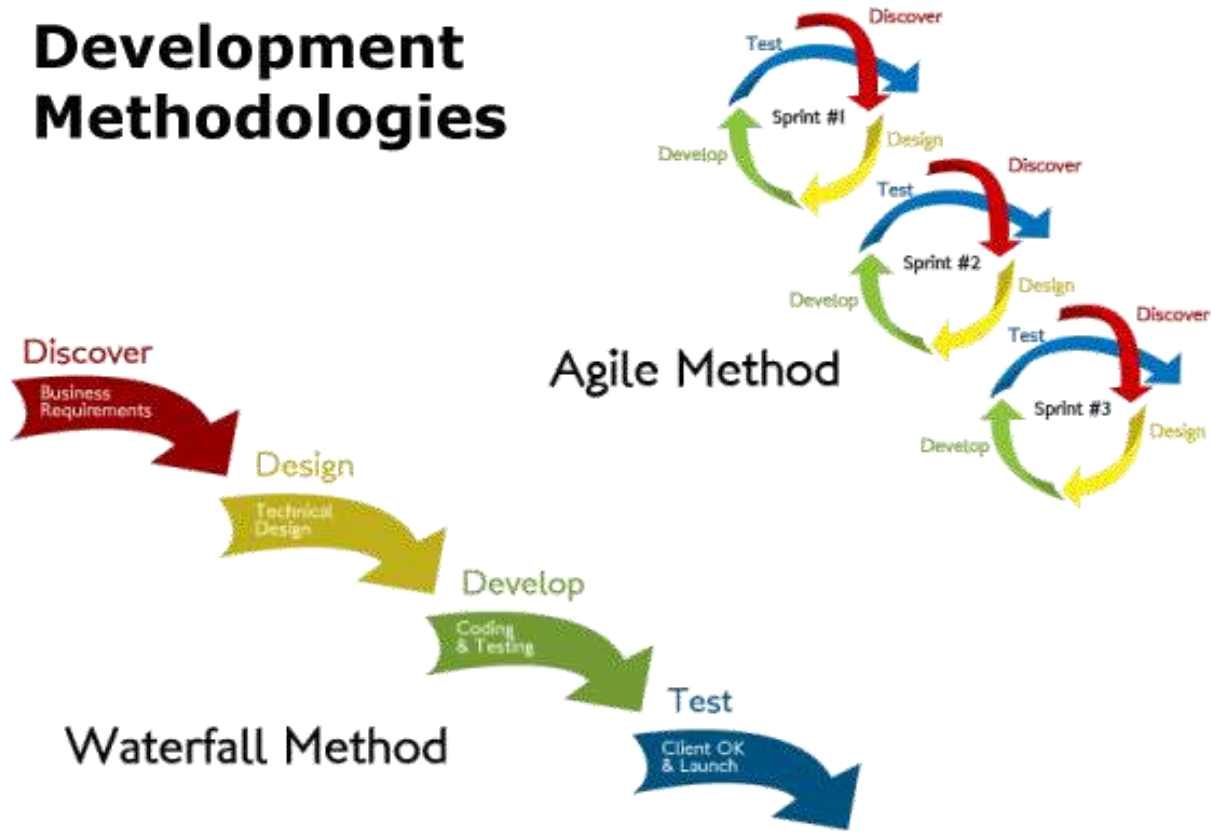
PIG & CHICKEN PART 2

© 2014 Jake Calabrese agilesafari.com

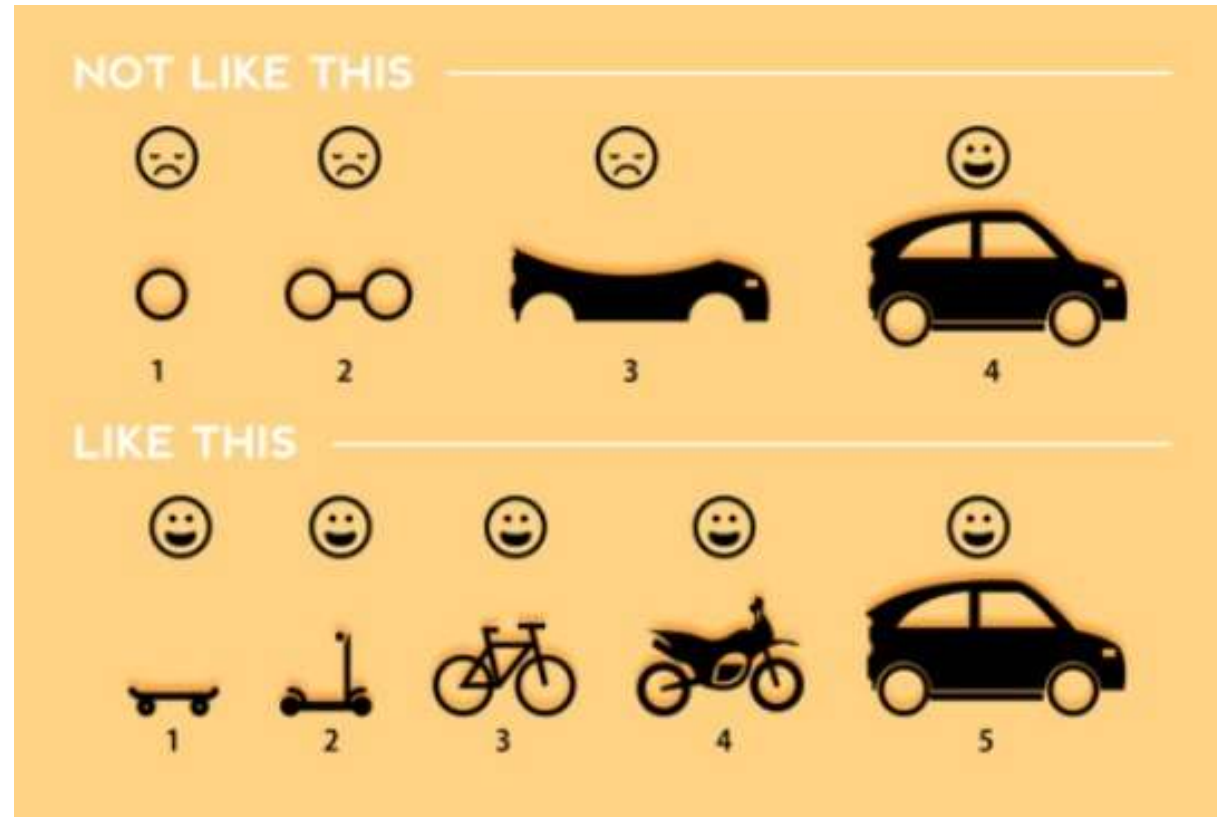


Waterfall vs Agile

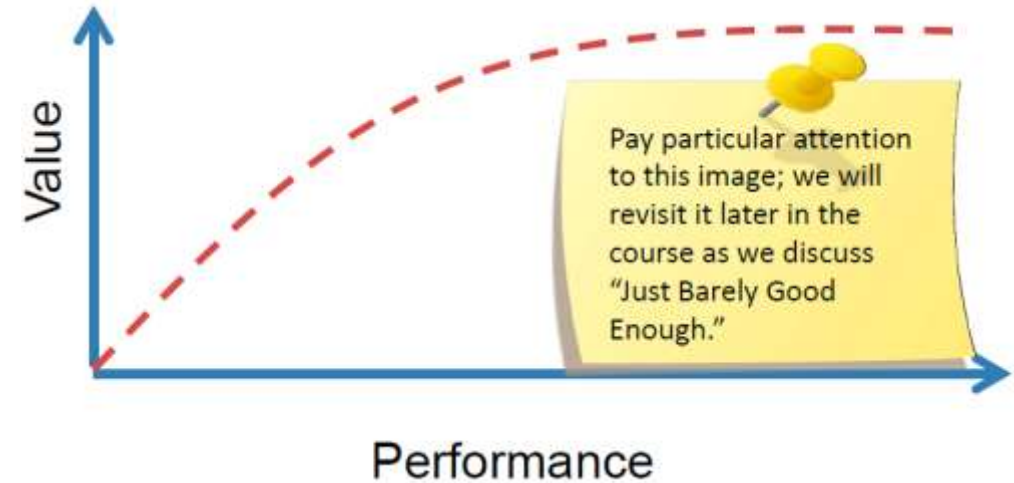
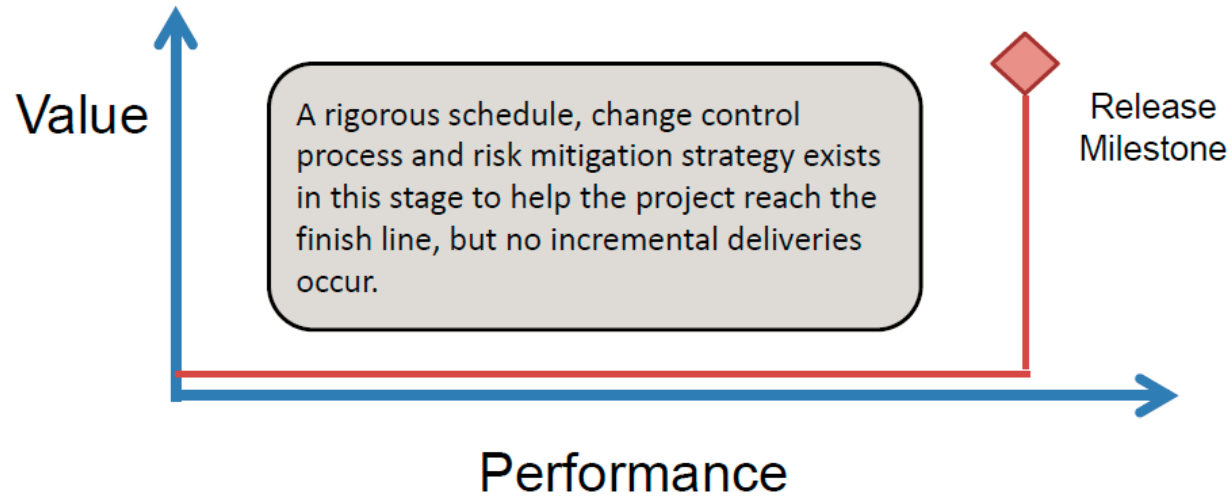
Development Methodologies



Value Driven



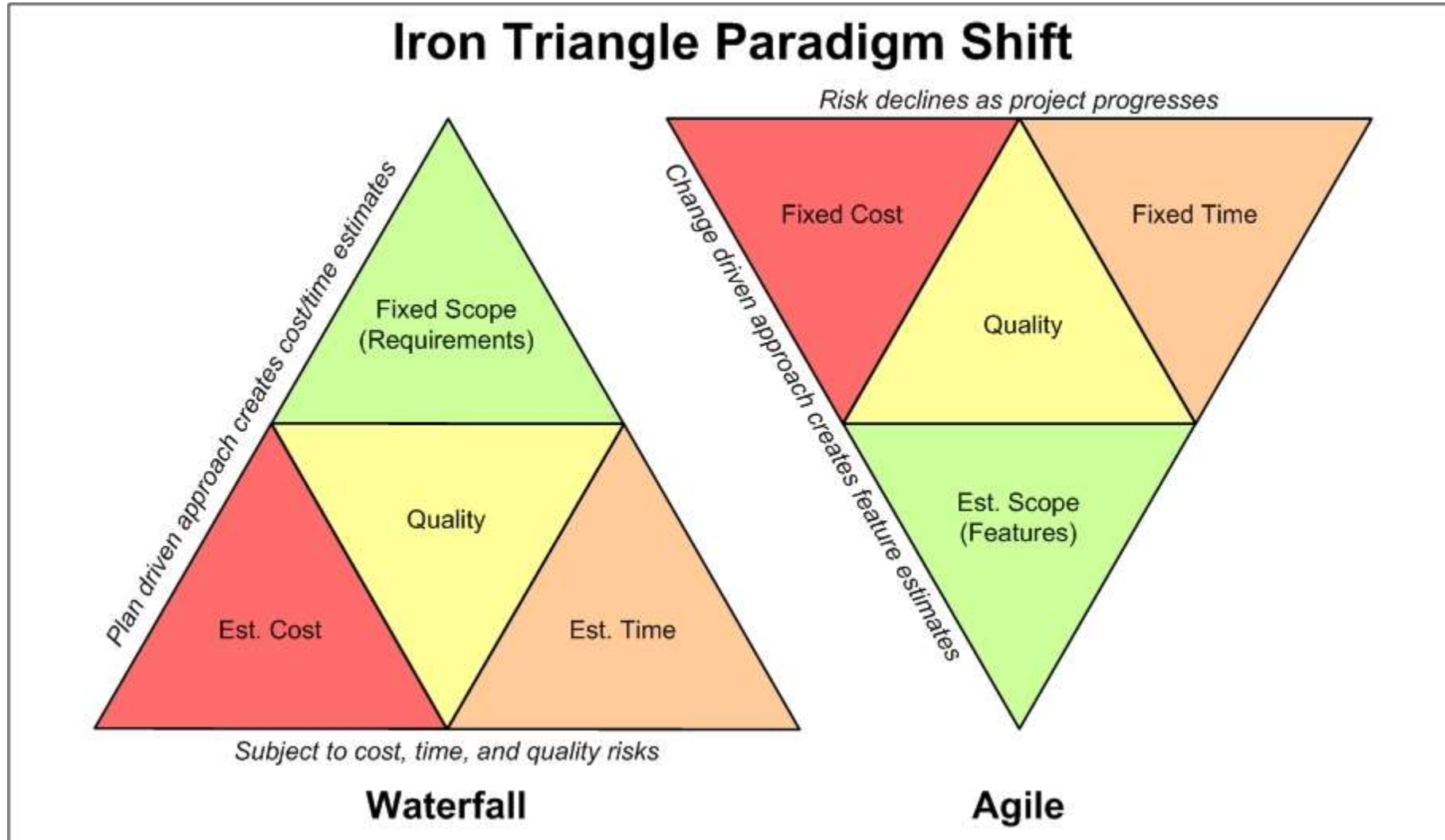
Plan Driven vs Value Driven



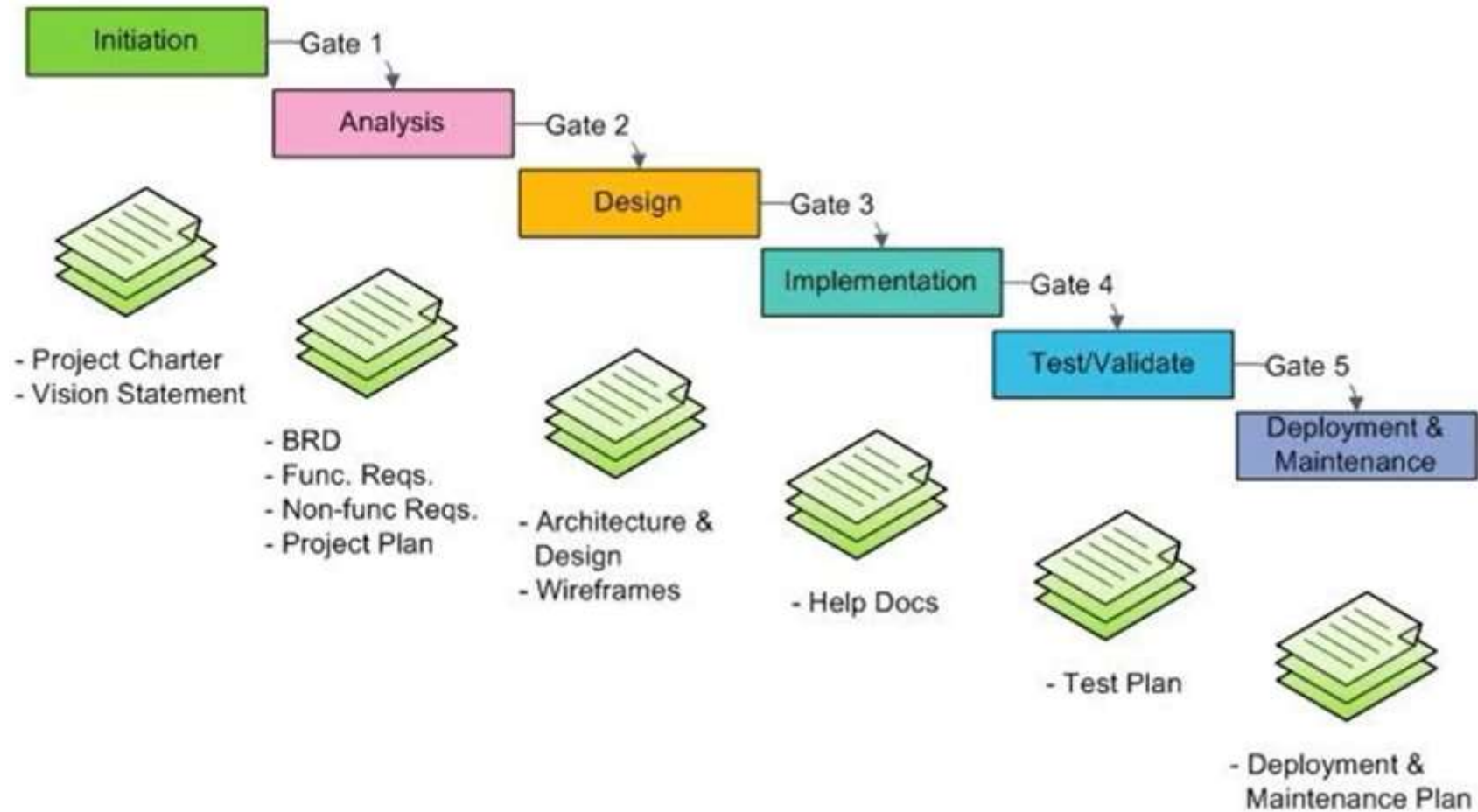
Customer Feeling



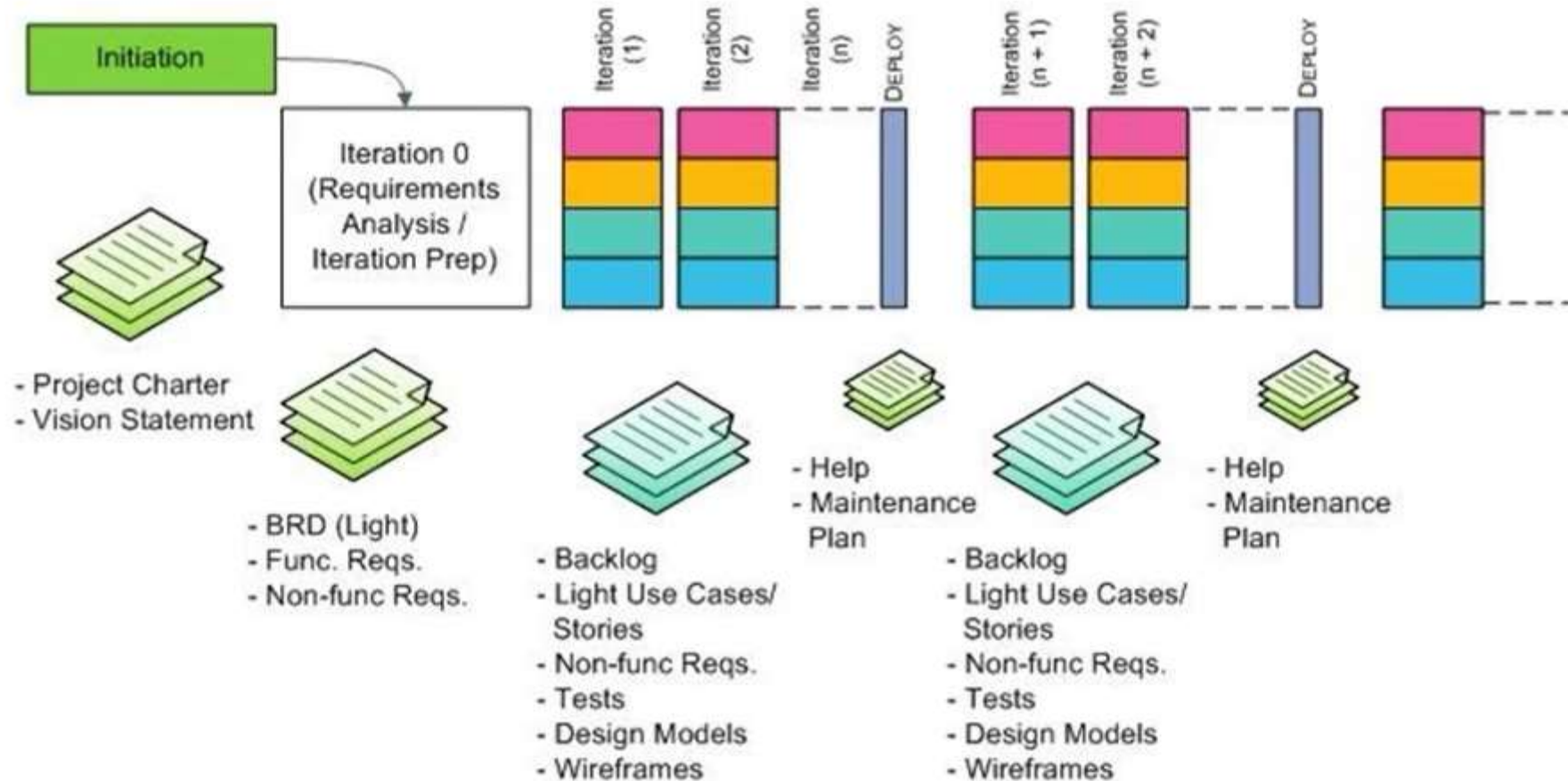
Plan Driven vs Value Driven



Traditional Project Documents



Possible Documents in Agile



User Story Card

User Story Template Cards

Story Card Template

Story Name

As a Agile Team Member,
I want to have a template for
writing our User Stories,
so that all of our cards can be
consistent.

Size: 0

Acceptance Criteria:

Packet contains the following:

- 100 Story Card Templates
- Instructions & Ideas for using the cards

COMMENTS:

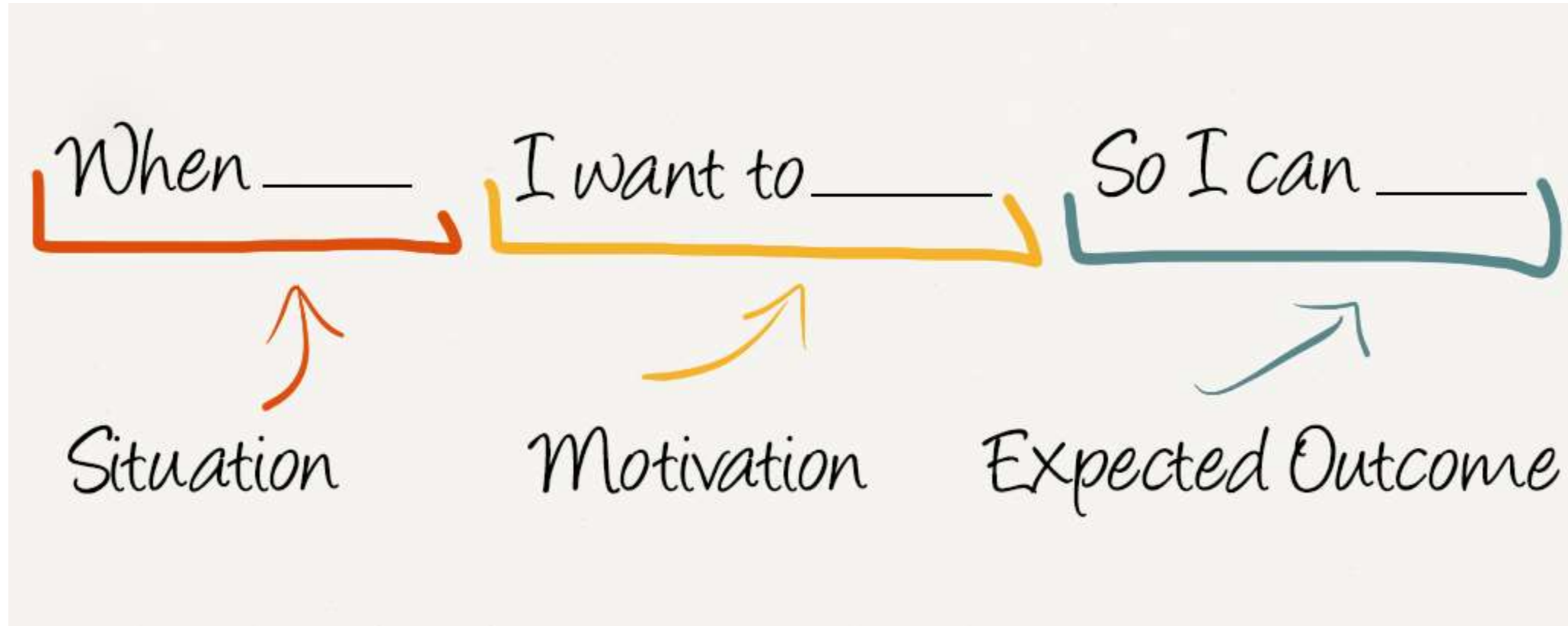


Developing Teams That Develop Software™

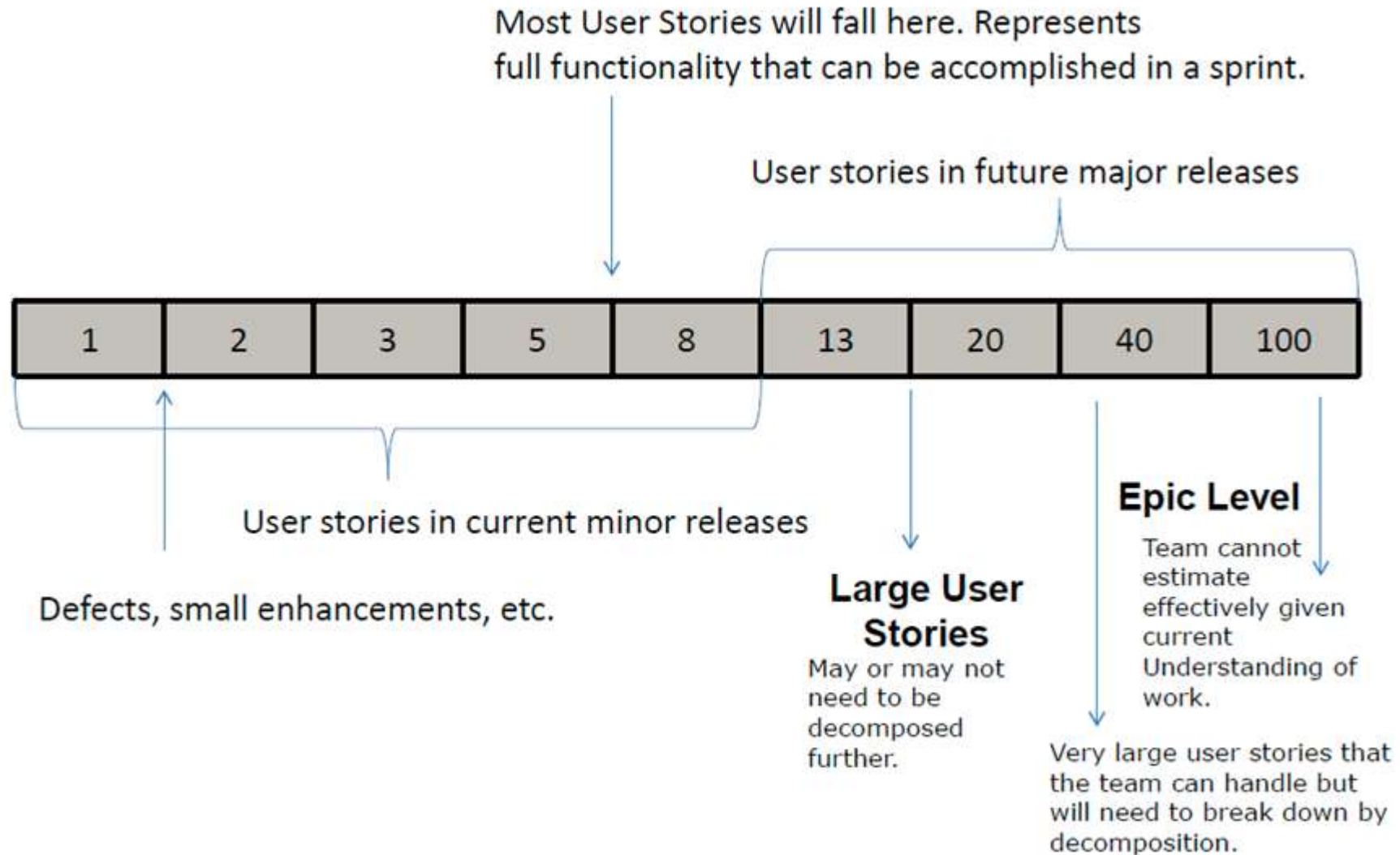
www.davisbase.com @davisbase #agile




User Story



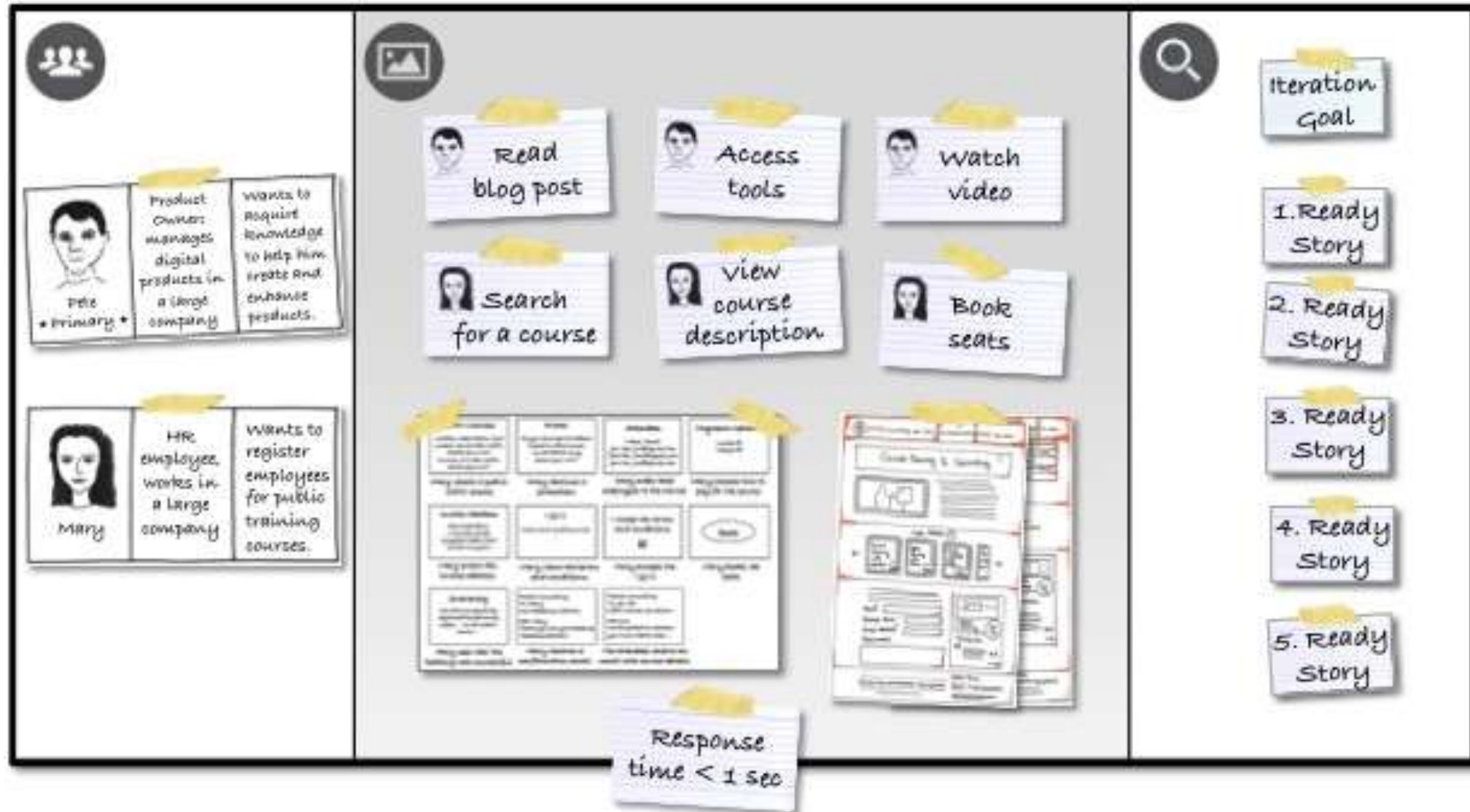
Agile Estimation - User Story Point



Persona

 <p>Peter</p>	<p>Works as product manager for a mid-sized company.</p> <p>Is 35 years old, holds a marketing degree.</p> <p>Has got experience working as a product owner on software products with agile teams.</p> <p>Has had some Scrum training.</p>	<p>Has managed mature products successfully. Now faces the challenge of creating a brand-new product.</p> <p>Wants to leverage his agile knowledge but needs advice on creating innovative product using agile techniques.</p>
--	--	--

User Story Card and Persona





Scrum

Scrum in Brief

- Scrum is an **iterative** and **incremental** agile software development framework for managing product development.
- It defines "a **flexible, holistic** product development strategy where a development team **works as a unit** to reach a common goal", challenges assumptions of the "traditional, sequential approach to product development, and enables teams to **self-organize** by encouraging physical co-location or close online collaboration of all team members, as well as **daily face-to-face communication** among all team members and disciplines involved.



Scrum Values

- **Commitment**

- Team members individually commit to achieving their team goals, each and every Sprint.

- **Courage**

- Team members know they have the courage to work through conflict and challenges together so that they can do the right thing.

- **Focus**

- Team members focus exclusively on their team goals and the Sprint Backlog; there should be no work done other than through their backlog.

- **Openness**

- Team members and their stakeholders agree to be transparent about their work and any challenges they face.

- **Respect**

- Team members respect each other to be technically capable and to work with good intent.



Terms

- **Backlog**

- A backlog is a prioritized list of work for the development team that is derived from the roadmap and its requirements.

- **Sprint**

- A series of **fixed-length** iterations.

- **Sprint Planning**

- A team planning meeting that determines what to complete in the coming sprint.

- **Daily Scrum**

- Also known as a daily scrum, a 15-minute mini-meeting for the software team to sync.

- **Sprint Review/ Sprint Demo**

- A sharing meeting where the team shows what they've shipped in that sprint.

- **Sprint Retrospective**

- A review of what did and didn't go well with actions to make the next sprint better.



Roles in Scrum - Essential Roles

- **Product Owner**

- Build and manage the product backlog
- Closely partner with the business and the team to ensure everyone understands the work items in the product backlog
- Give the team clear guidance on which features to deliver next
- Decide when to ship the product with the predisposition towards more frequent delivery

- **Scrum Team**

- Scrum teams are the champions for sustainable development practices. The most effective scrum teams are tight-knit, co-located,

- **Scrum Master**

- Scrum masters are the champion for scrum within their team. They coach the team, the product owner, and the business on the scrum process and look for ways to fine-tune their practice of it.



Roles in Scrum - Additional Roles

- Technical Expert(s)
- Domain Expert(s)
- Users
- Other Stakeholders



Chickens and Pigs in Scrum

- Customer Unit

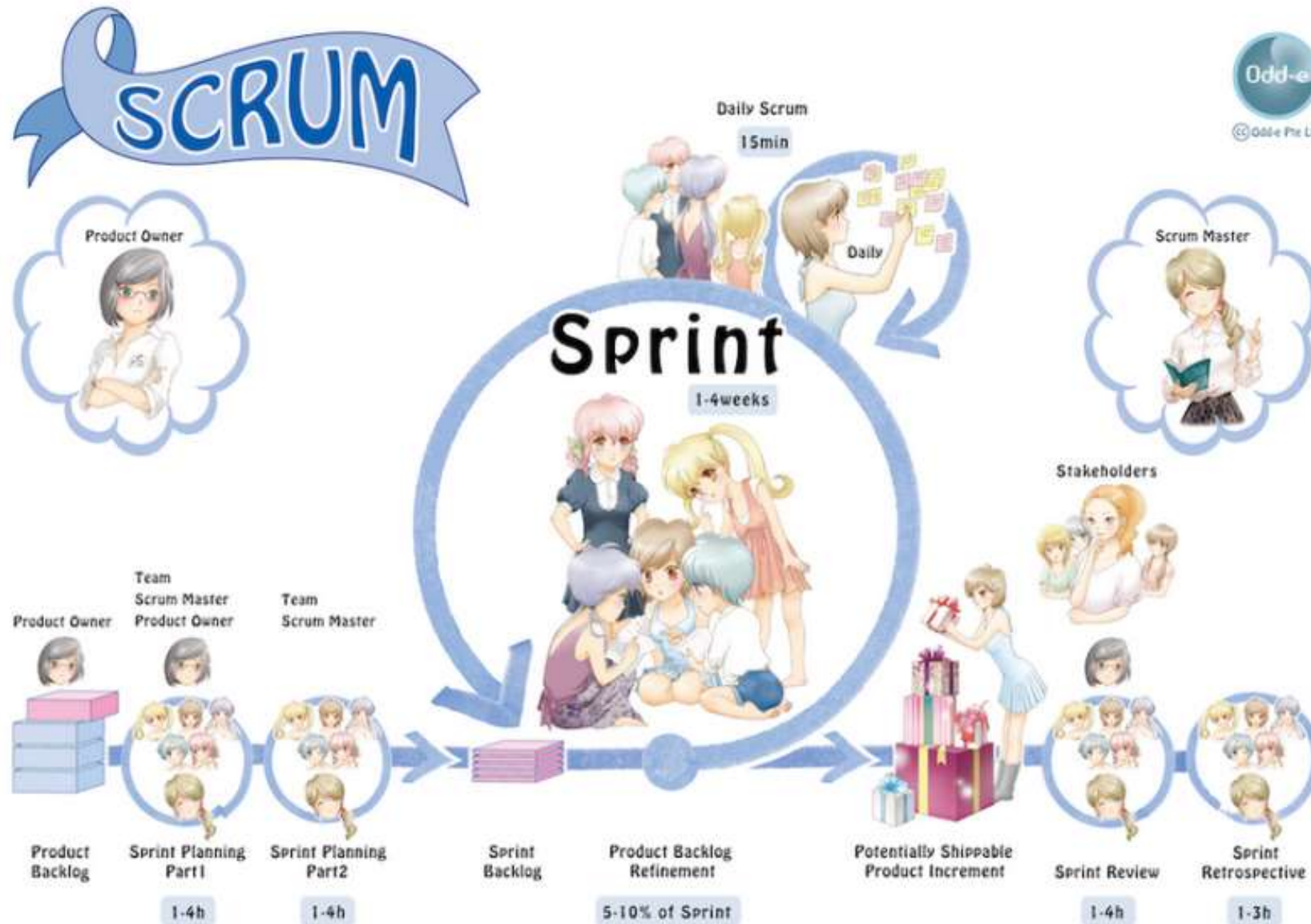
- People who are involved but not dedicated to the project are known as Chickens - they may attend Scrum meetings as observers
 - Customer
 - Product Manager/ Product Owner
 - Marketing
 - Executives
 - Client Services

- Development Unit

- Members of Scrum Team are known as Pigs because they are committed to delivering the Sprint Goal
 - Developer
 - Product Analyst
 - QA
 - IT
 - Project Manager
 - Graphic Designer
 - Technical Writer



Scrum Flow in 1 Picture



Famous Tools and Techniques

- Sprint Planning

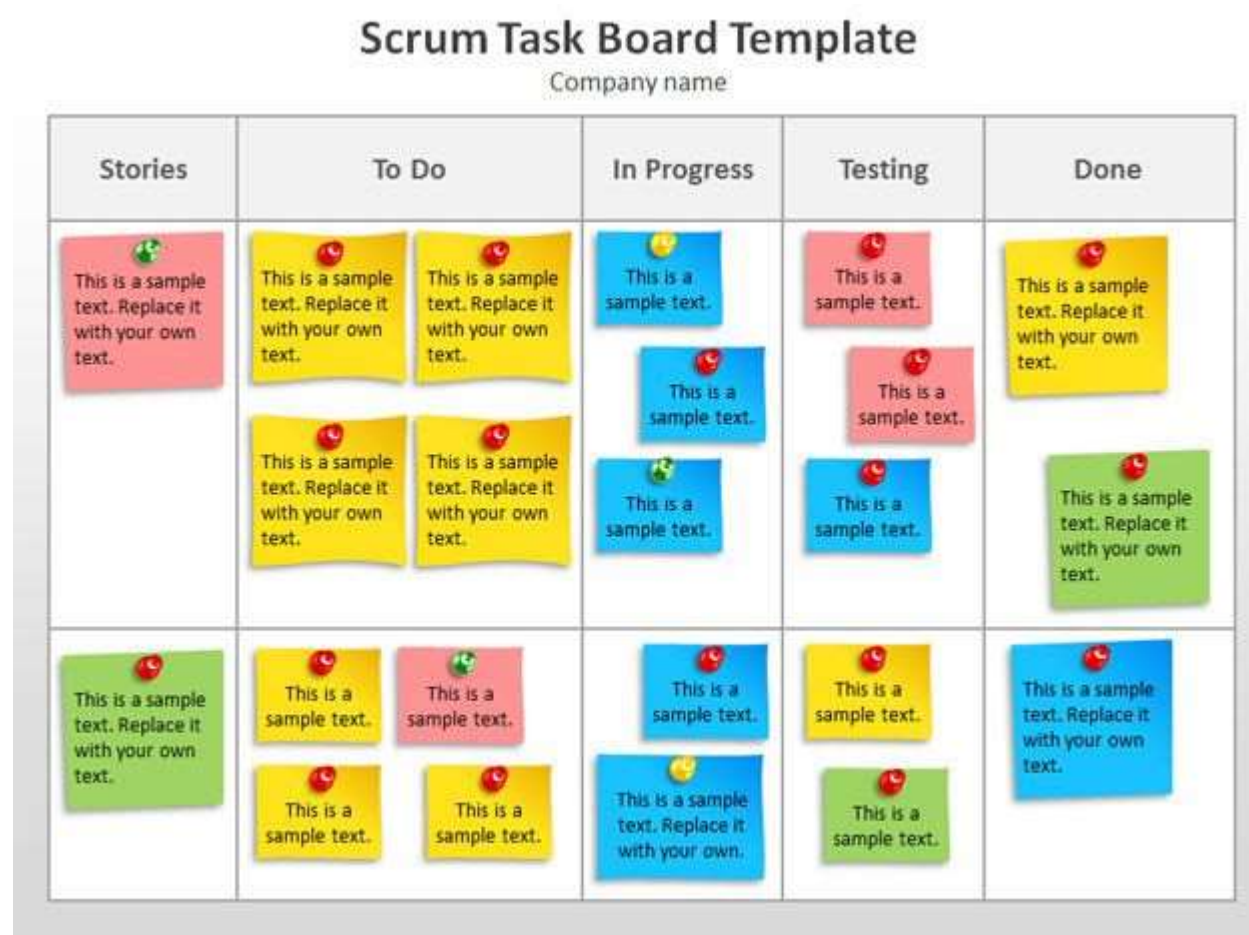
At the beginning of a Sprint, the Scrum Team holds a Sprint Planning event to:

- Communicate the scope of work that is intended to be done during that Sprint
- Select Product Backlog Items that can be completed in one Sprint
- Prepare the Sprint Backlog that details the work needed to finish the selected Product Backlog Items
- Time-boxed to a four-hour limit for a two-week Sprint (pro rata for other Sprint durations)
- Once the Development Team prepares the Sprint Backlog, they commit (usually by voting) to deliver tasks within the Sprint.



Famous Tools and Techniques

- Scrum Task Board



Famous Tools and Techniques

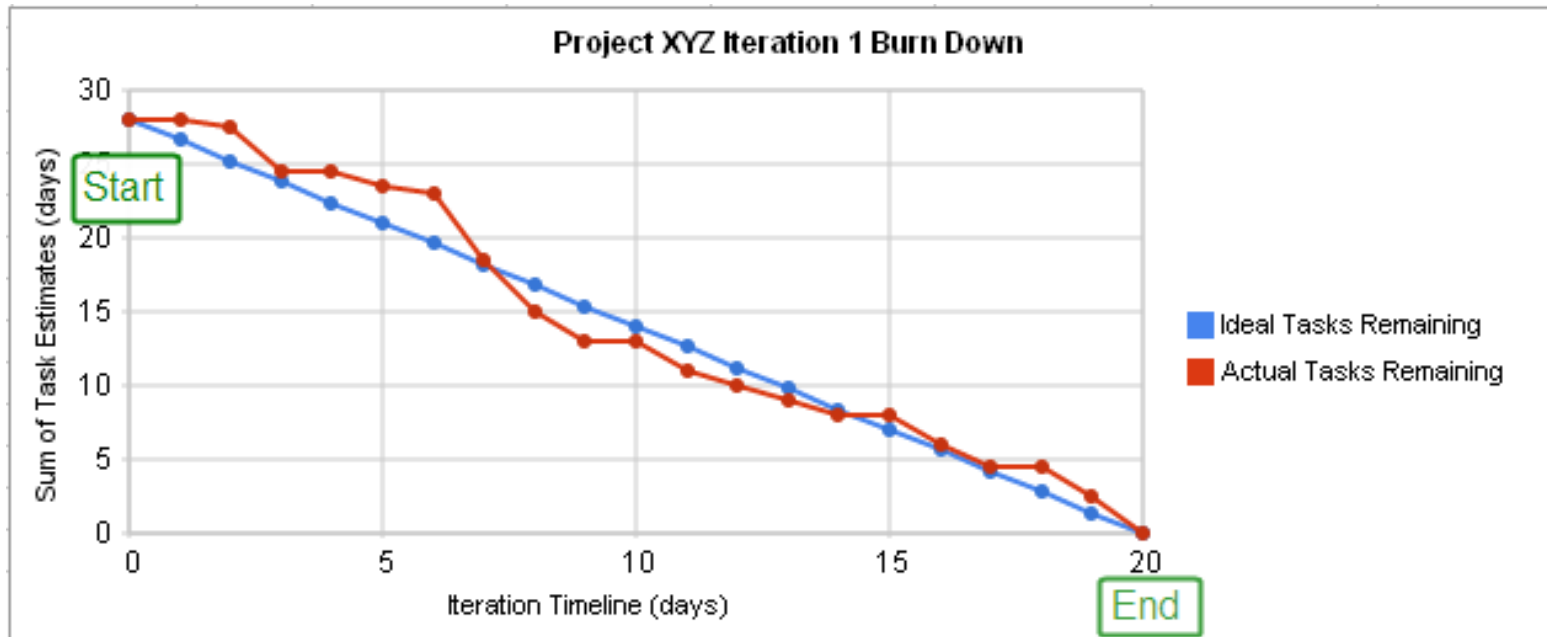
- Daily Scrum

- What work did you complete yesterday?
- What have you planned for today?
- Are you facing any problems or issues?



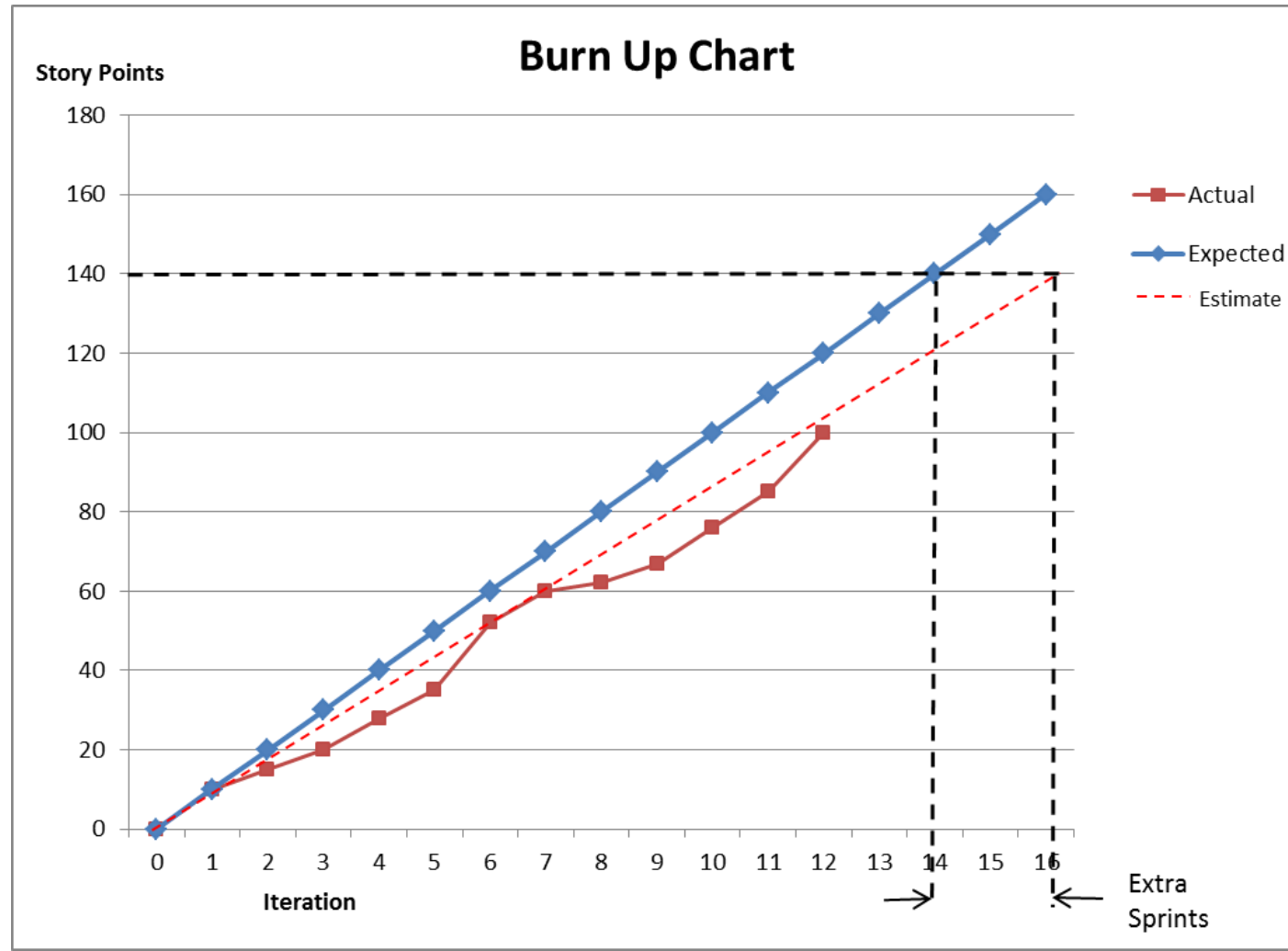
Famous Tools and Techniques

- Burn-Down Chart



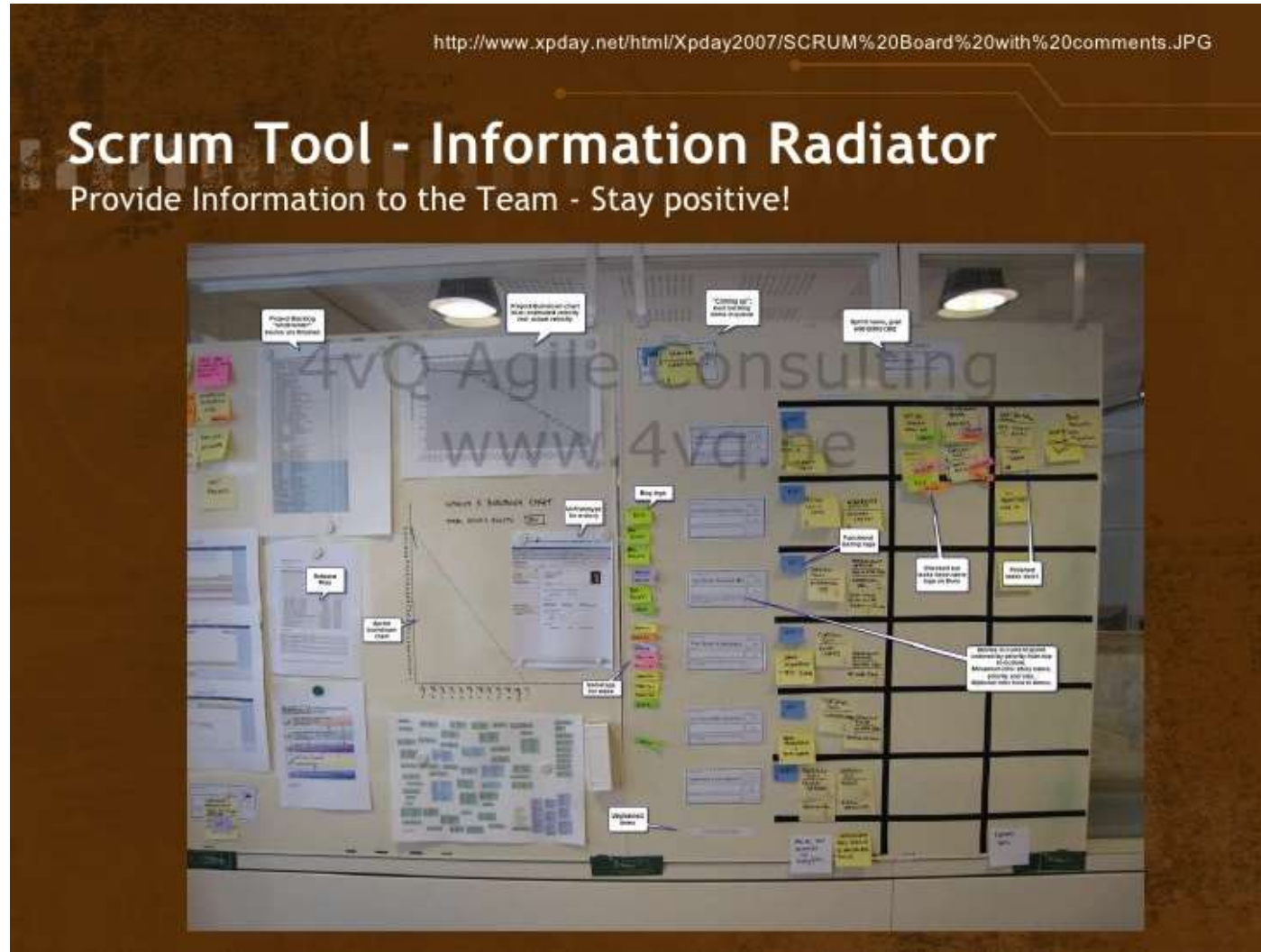
Famous Tools and Techniques

- Burn-Up Chart



Famous Tools and Techniques

- Scrum Information Radiator



Famous Tools and Techniques

- **Sprint Review**

- Reviews the work that was completed and the planned work that was not completed
- Presents the completed work to the stakeholders (a.k.a. the *demo*)

- **Guidelines for Sprint Reviews:**

- Incomplete work cannot be demonstrated
- The recommended duration is two hours for a two-week Sprint (pro-rata for other Sprint durations)



Famous Tools and Techniques

- **Sprint Retrospective**

- It is a meeting facilitated by the ScrumMaster at which the team discusses the just-concluded sprint and determines what could be changed that might make the next sprint more productive. The sprint review looks at what the team is building, whereas the retrospective looks at how they are building it.
- 3 main questions/points for discussion:
 - What went well during the sprint cycle?
 - What went wrong during the sprint cycle?
 - What could we do differently to improve





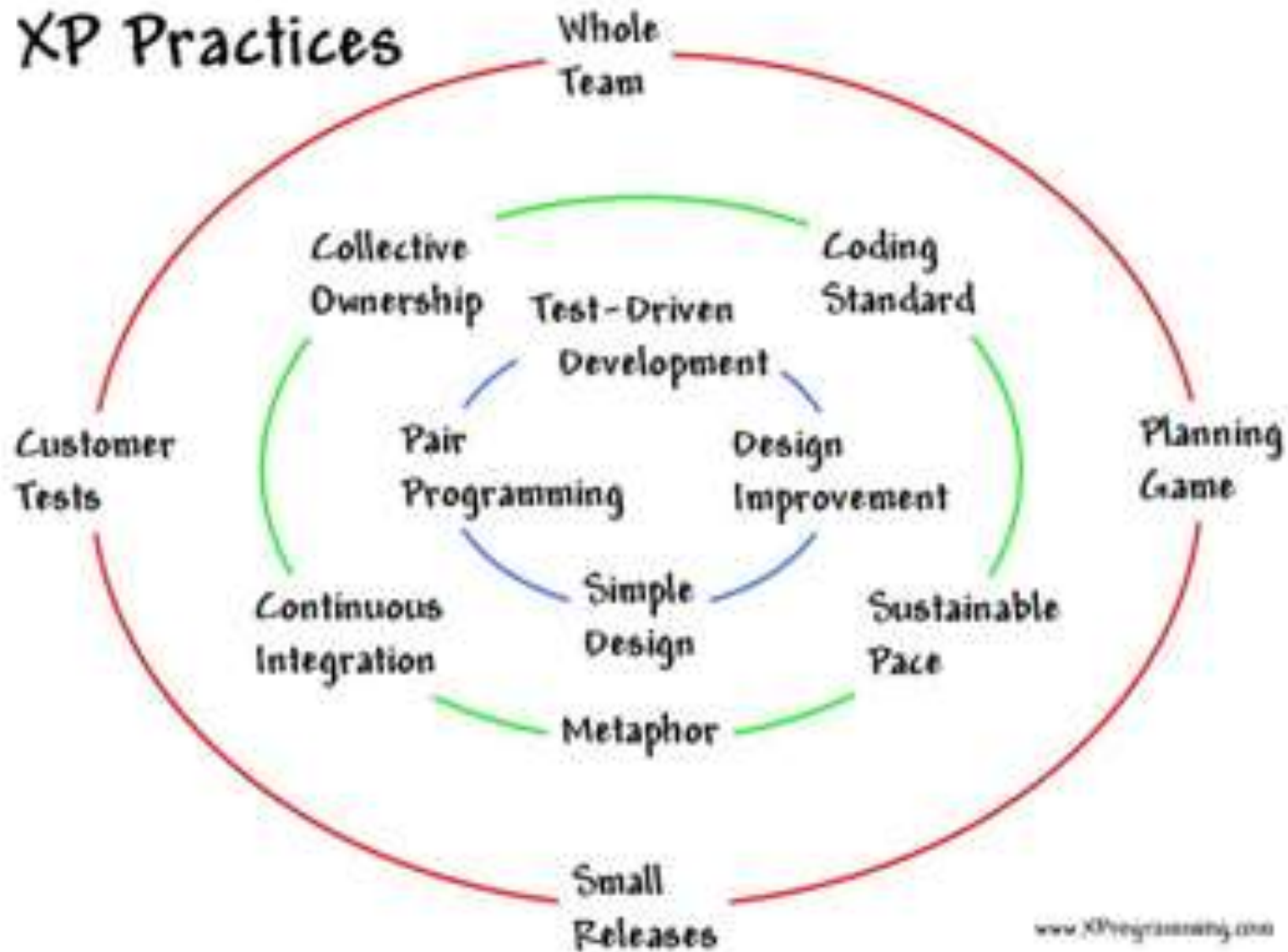
XP

XP in Brief

- Extreme Programming (XP) is a **pragmatic** approach to **program development** that emphasizes **business results first** and takes an **incremental, get-something-started** approach to building the product, using **continual testing** and **revision**.
- Comparing with Scrum, Scrum **does not** prescribes engineering practices; But XP **does**.



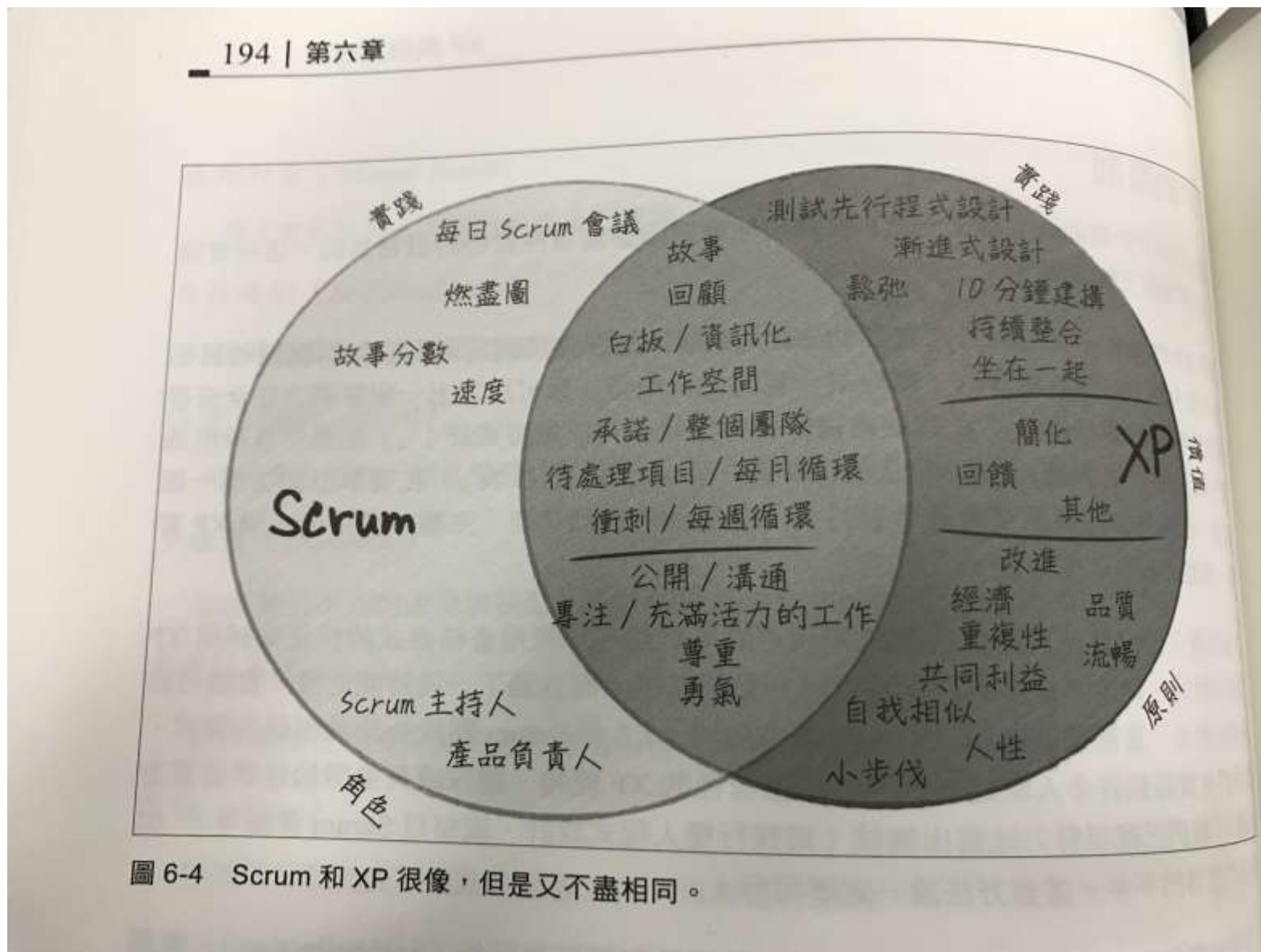
Practices All in 1



XP Continual Testing And Revision



Scrum and XP





LEAN

What is LEAN

- Toyota realised it needed knowledge in order to develop a car. It used experiments to acquire this knowledge — and this was the beginning of the lean systems Toyota uses today

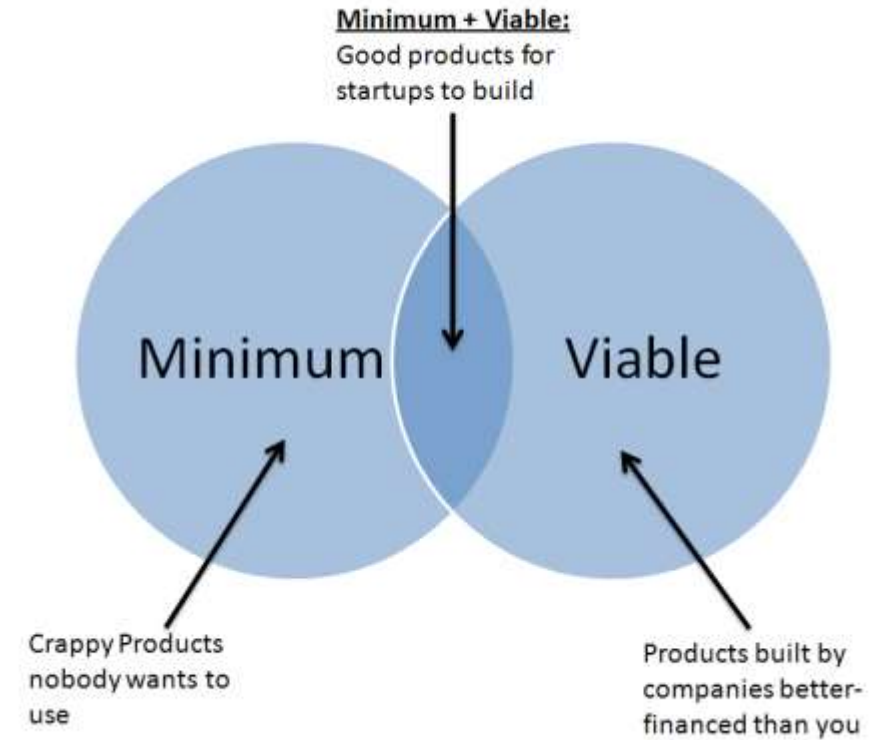
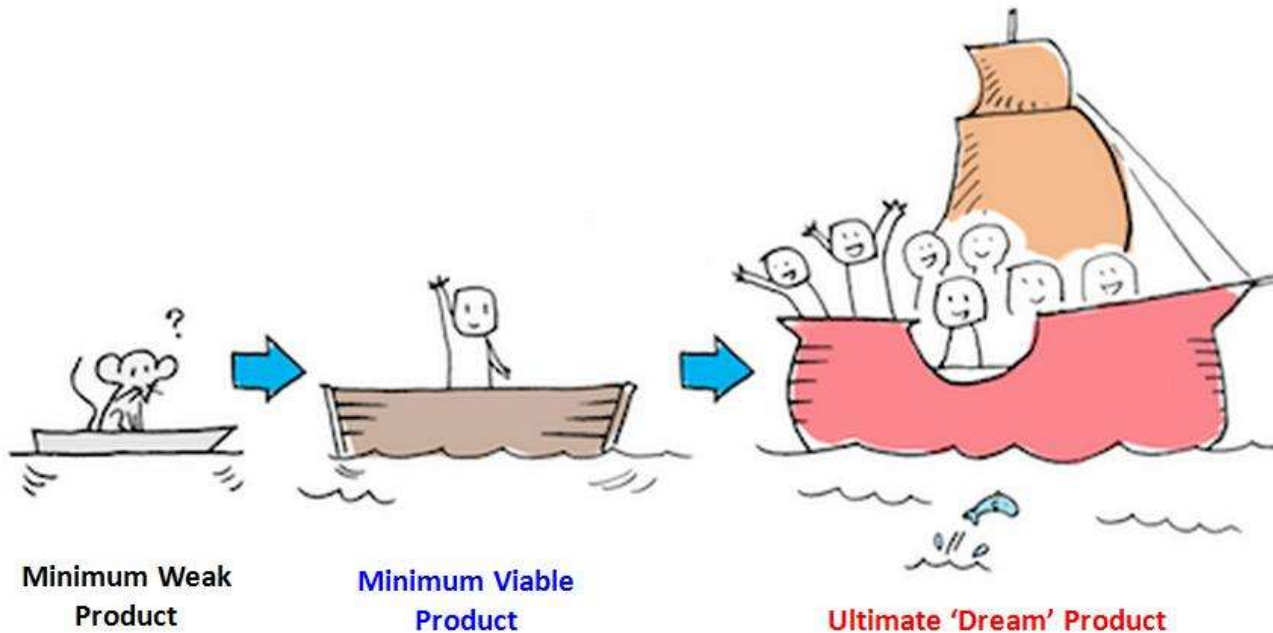


7 Key Principles of Lean Development

- Eliminate Waste
- Build Quality In
- Create Knowledge
- Defer Commitment
- Deliver Fast
- Respect People
- Optimize The Whole



Minimum Value Product(MVP)



Learning From Practices



赛亚人(正常状态)



超级赛亚人1
(头发变黄, 竖起)



超级赛亚人2
(头发竖起更多, 前额只留一根发)



超级赛亚人3
(没有了眉毛, 头发长)



超级赛亚人4
(有尾巴, 身上有红毛)



超级赛亚人5
DRAGON BALL
LZ.52PK.COM



Kanban

What is Kanban

- Kanban is a popular framework used by software teams practicing **LEAN** software development.



Pull System

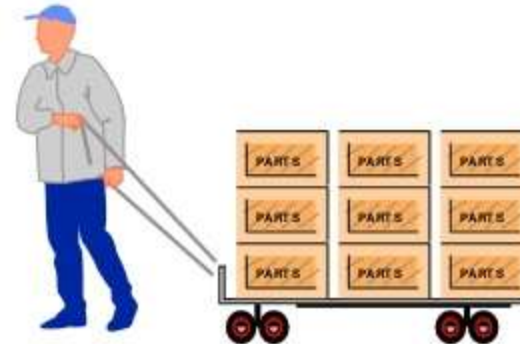
Push vs. Pull

**Make all we can
just in case.**



- Production Approximation
- Anticipated Usage's
- Large Lots
- High Inventories
- Waste
- Management by Firefighting
- Poor Communication

**Make what's needed
when we need it**



- Production Precision
- Actual Consumption
- Small Lots
- Low Inventories
- Waste Reduction
- Management by Sight
- Better Communication

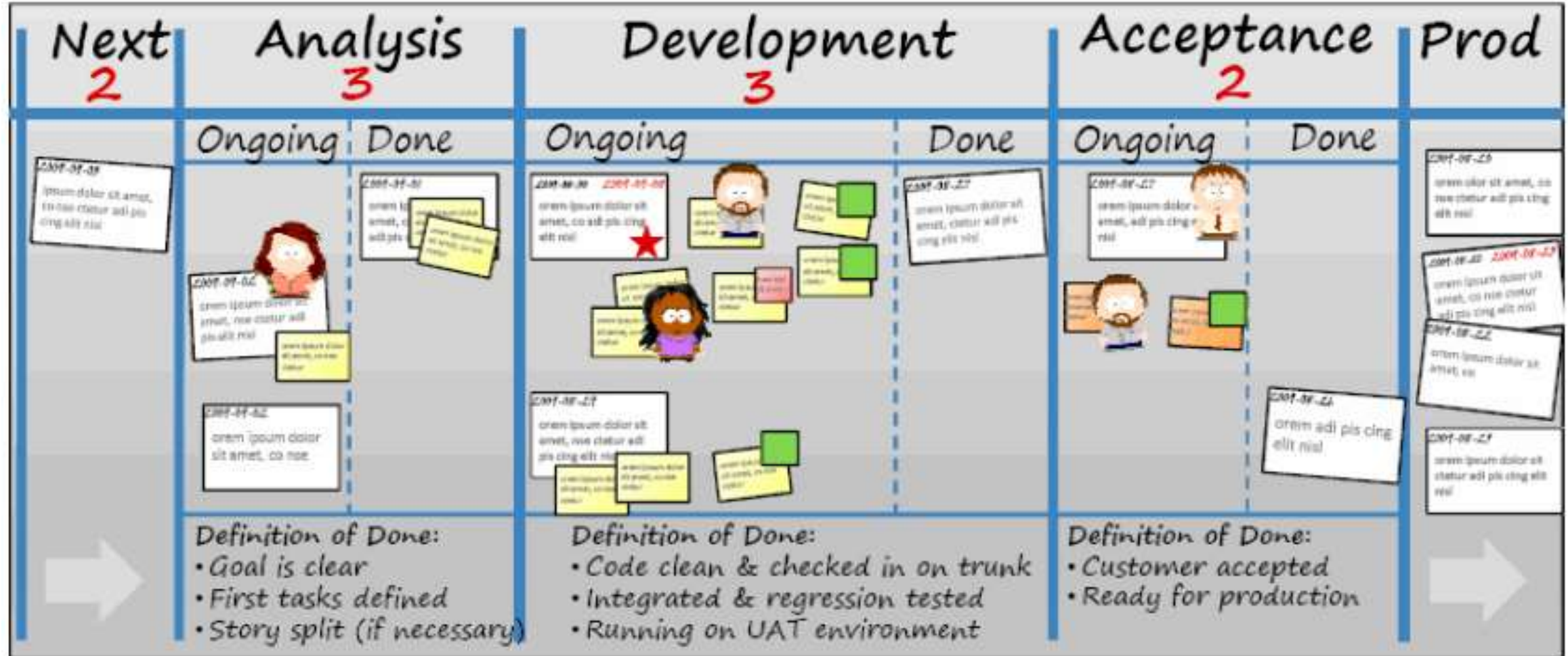


WIP(work-in-progress) Limit

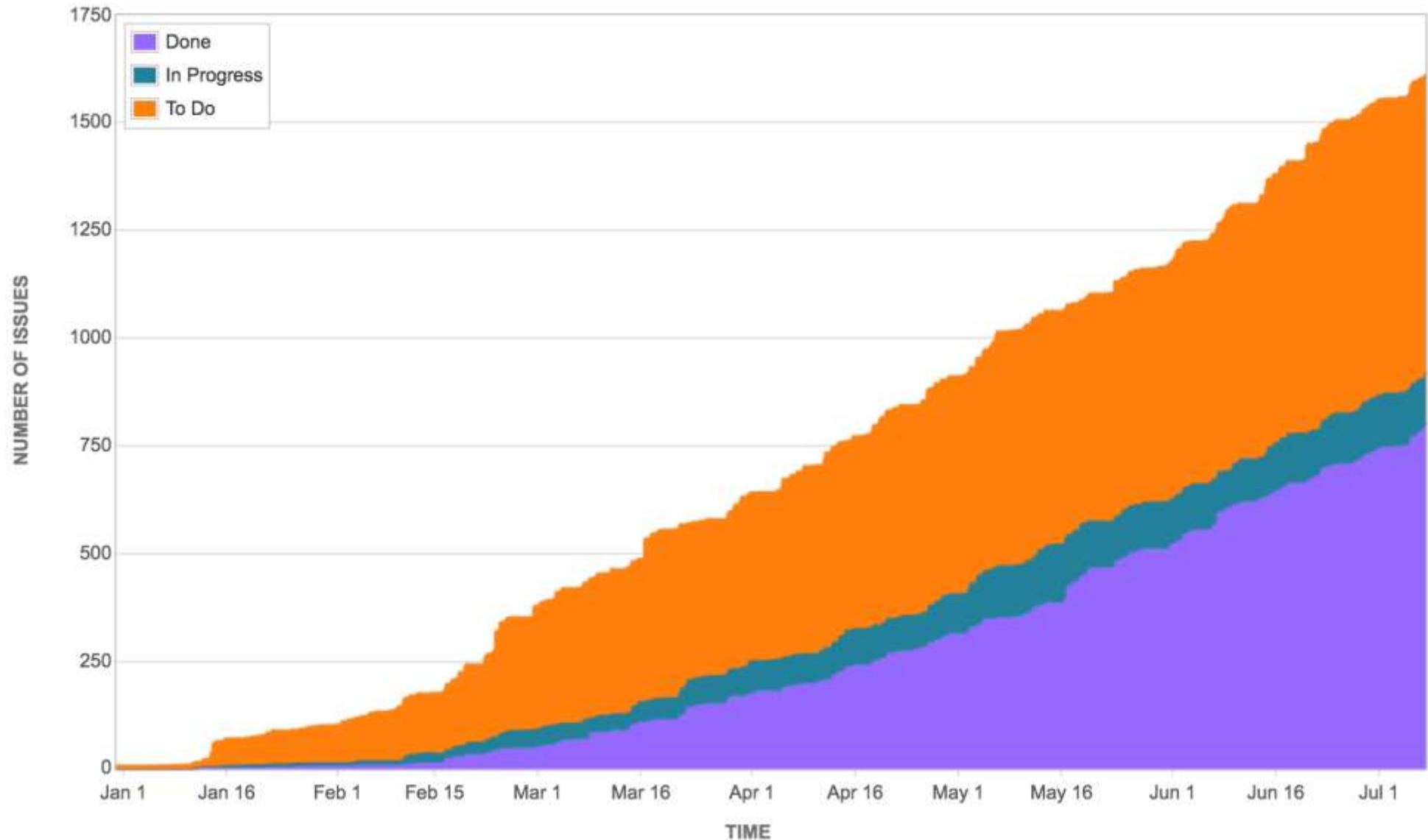
- WIP(work-in-progress) Limit
 - WIP limits in the context of the Kanban board, stand for the maximum amount of work items a given phase or the whole workflow can hold.
- Benefit for WIP



Kanban Flow



Cumulative Flow Diagram





Recap Again

This is one of Agile

IA evolved and expanded into IxD and her sisters
Darwinism at work

The Evolution of Mario



*Dates based on US release dates.

Geekstir.com

#LeanUX | @jboogie



Waterfall, Agile and LEAN

What's the difference?

WATERFALL

Problem: **Known**

Solution: **Known**

Progress: **Next Stage**

AGILE

Problem: **Known**

Solution: **Unknown**

Progress: **Working product**

LEAN START-UP

Problem: **Unknown**

Solution: **Unknown**

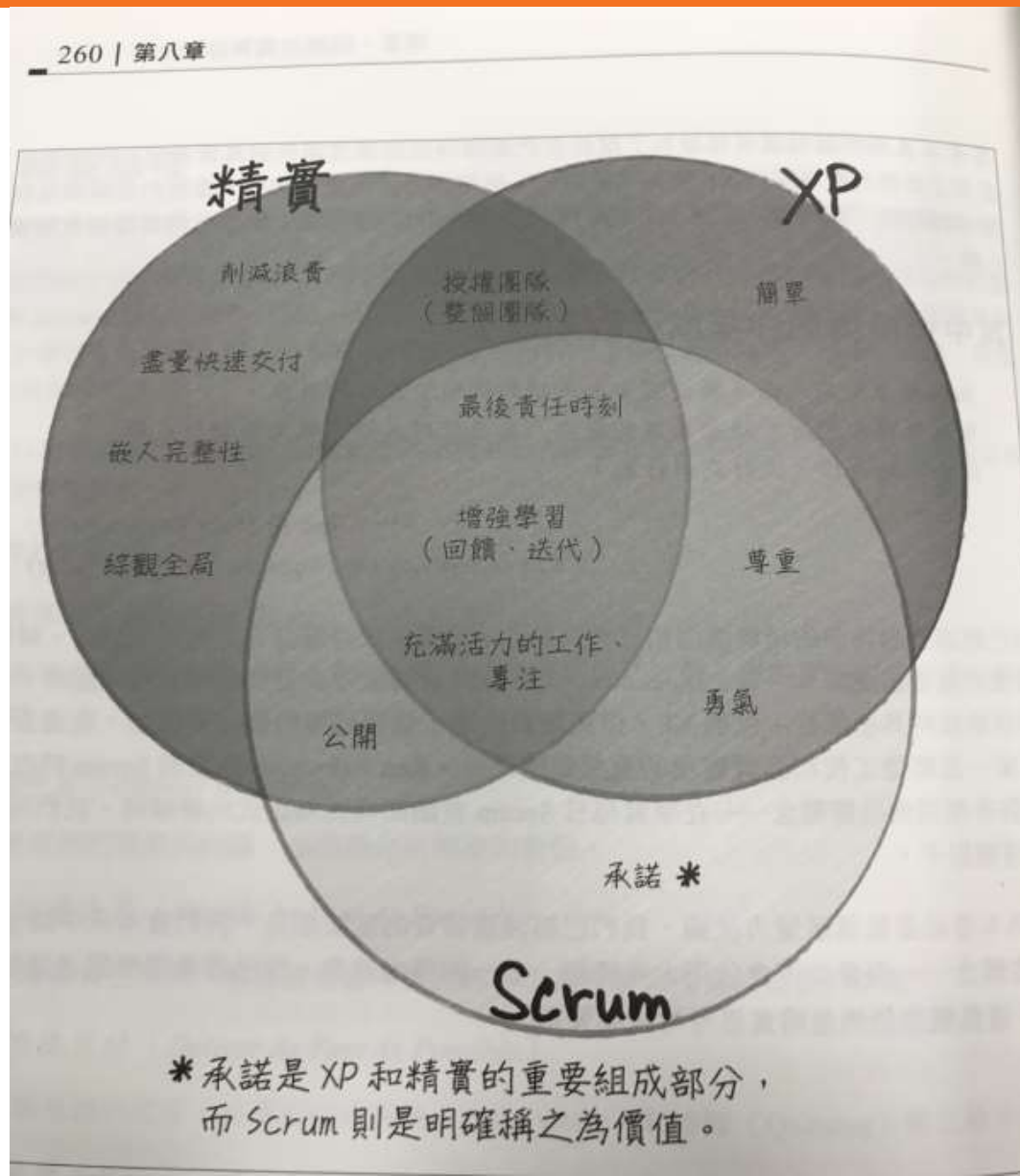
Progress: **Validated Learning**



Better Together



Scrum, XP and Kanban



To Be Brief...



Bad Smell in Agile Teams



TYPES OF AGILE TEAMS TO AVOID



SHUHARI

守 破 離

shu

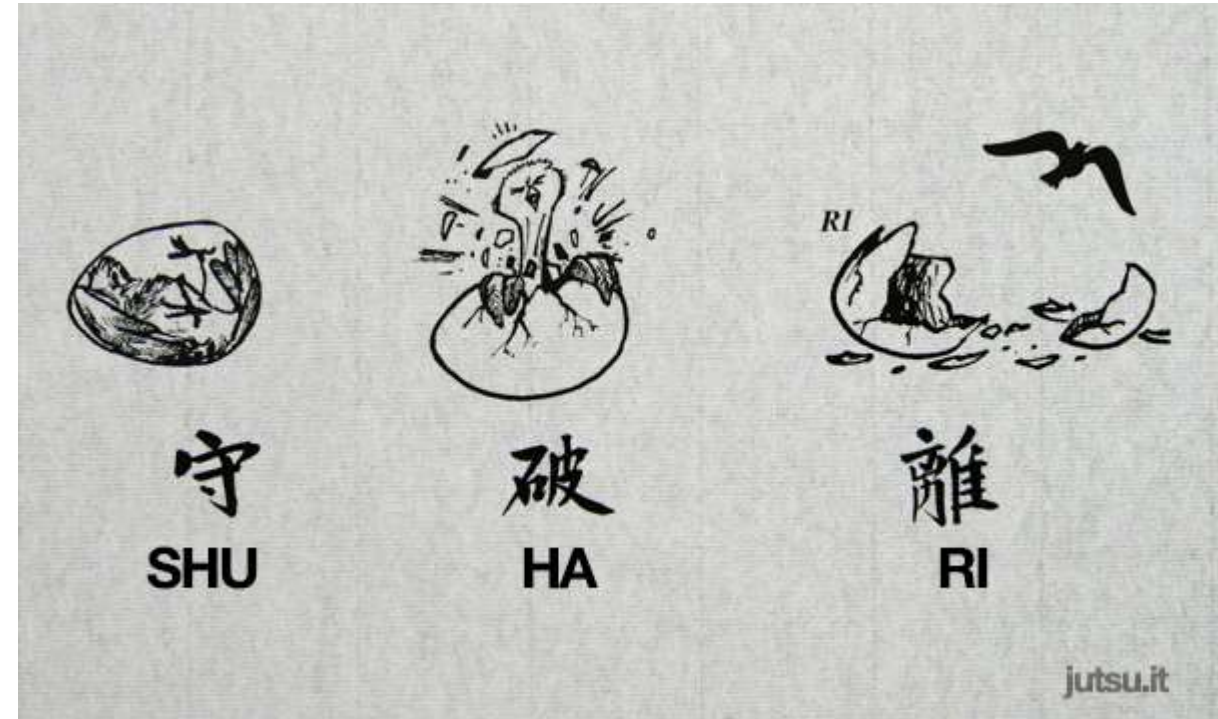
obey
fundamentals
repetition

ha

detatch
self recognition
innovation

ri

separate
creative technique
transendence



Self Organizing Agile Team



- Competency
- Collaboration
- Motivation
- Trust and respect
- Continuity



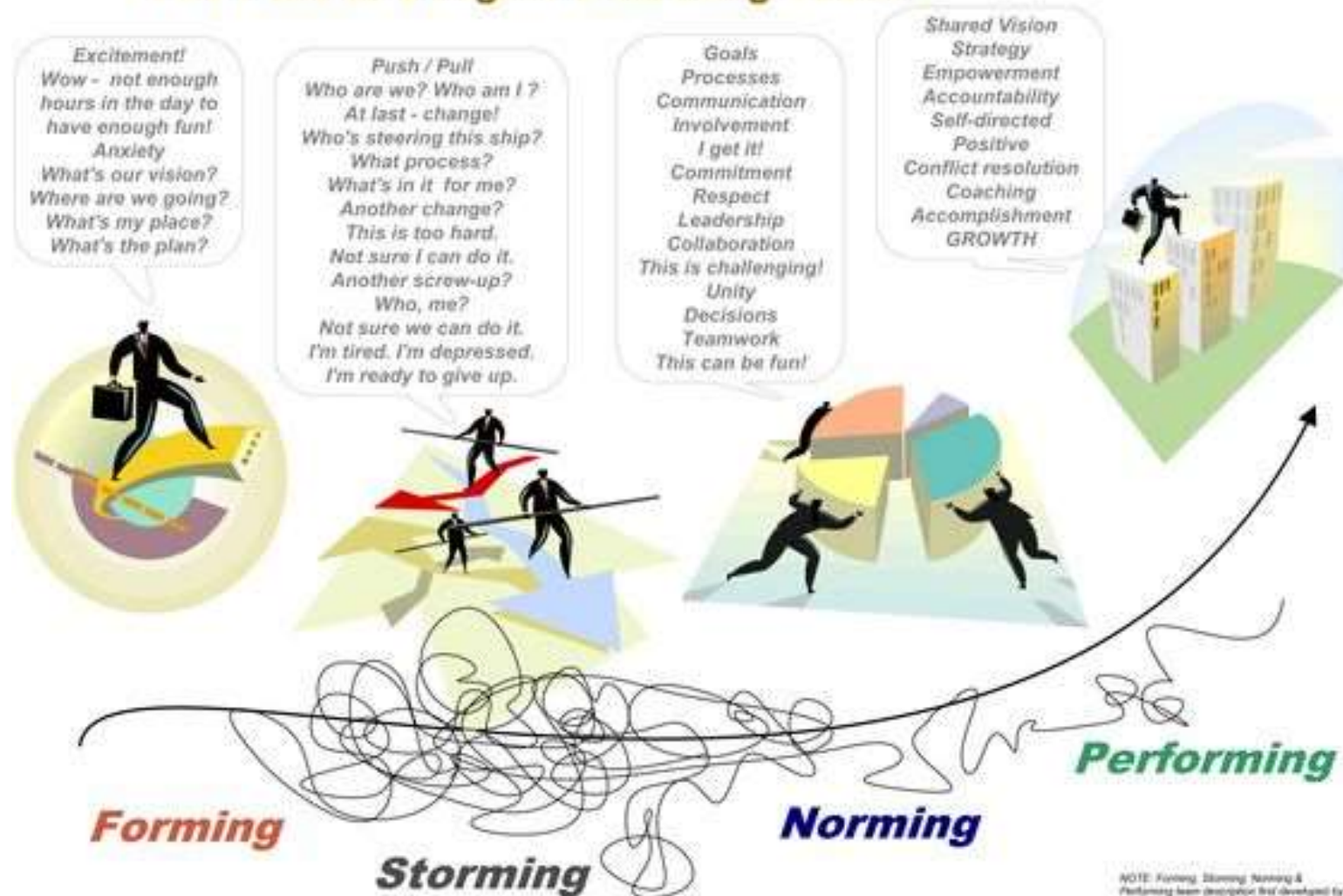
Not Just Easy and Simple...

Are you too busy to improve?



It works progressively...

The Path to a High-Performing Team



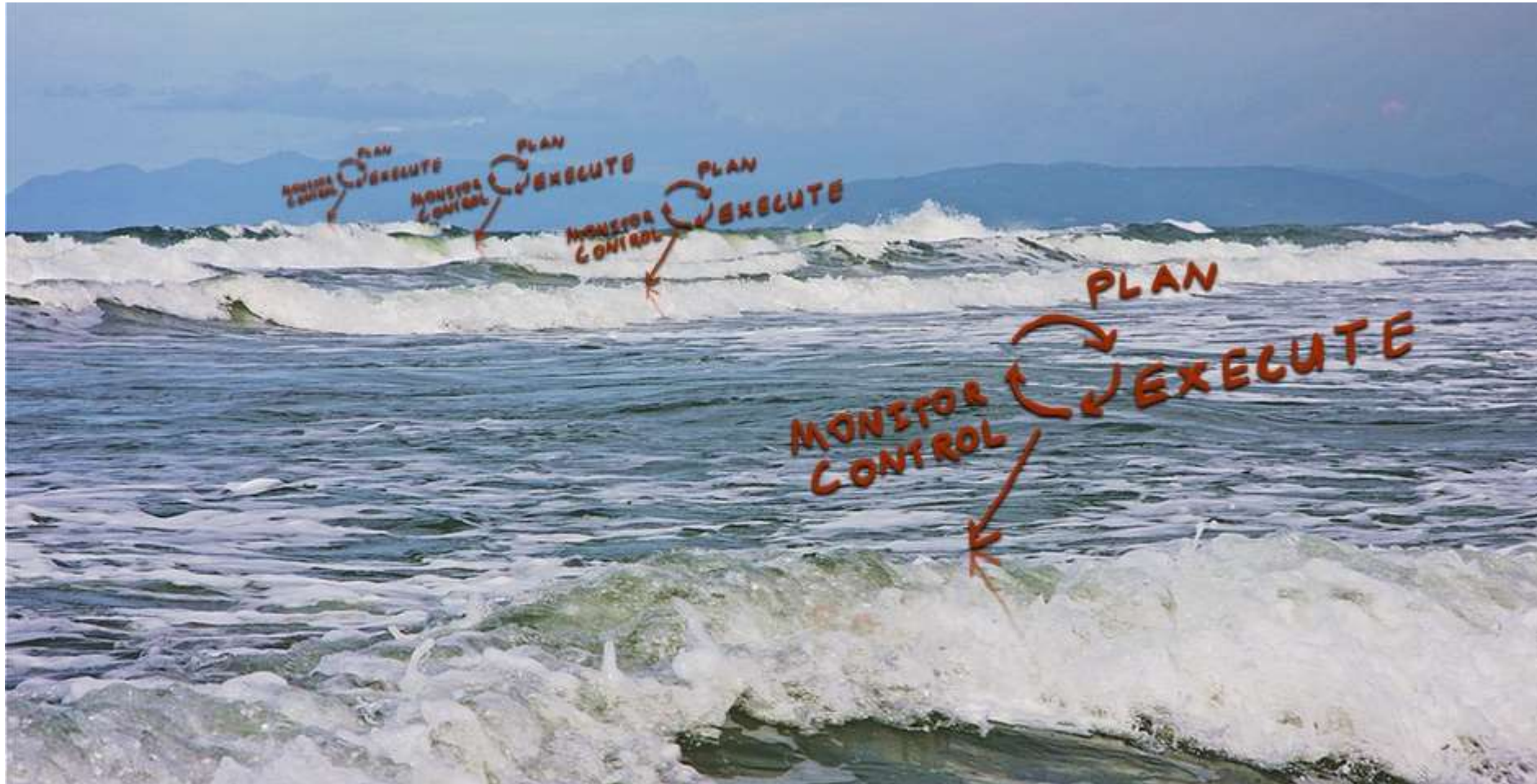
NOTE: Forming, Storming, Norming & Performing team descriptions first developed by Bruce Tuckman



Agile Is a Tour

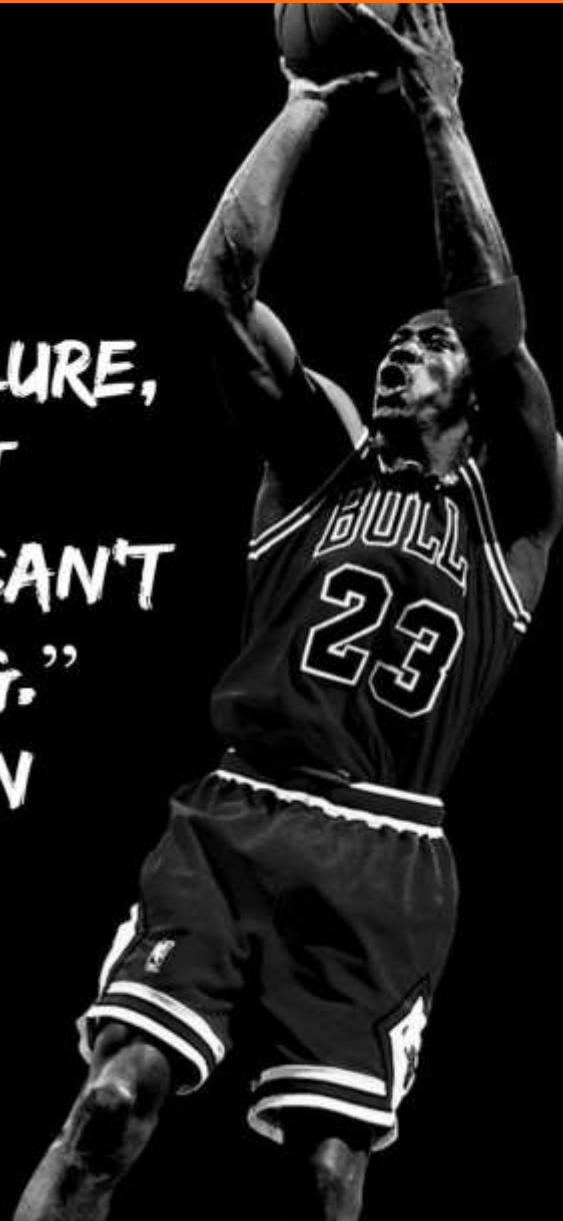


You Need This



Just Do It...

**"I CAN ACCEPT FAILURE,
EVERYONE FAILS AT
SOMETHING. BUT I CAN'T
ACCEPT NOT TRYING."
— MICHAEL JORDAN**



And Keep Kaizen

改

KAI=Change

善

ZEN=Good

改善

KAIZEN
(Continual
Improvement)



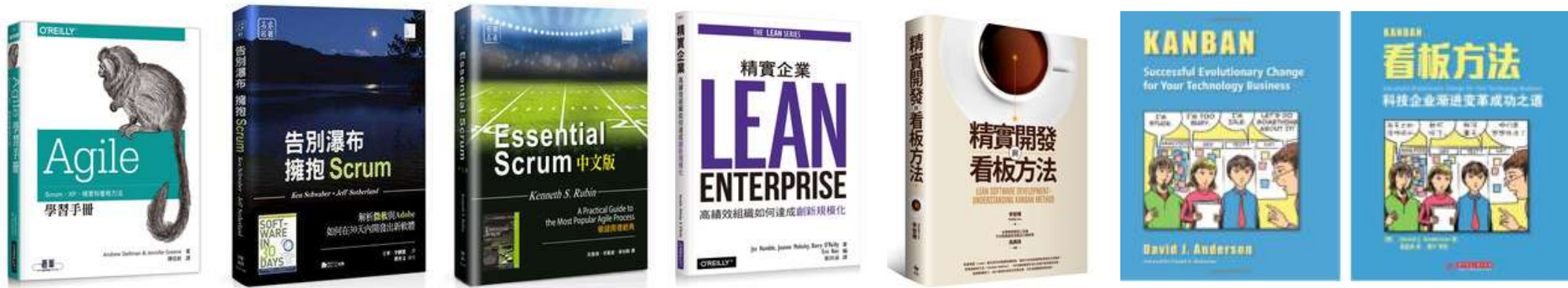
Certifications

- [PMI - Agile Certified Practitioner \(PMI-ACP\)](#)
- [Scrum Alliance – CSM, CSD, CSPO and CSP](#)
- [Scrum.org – PSM, PSPD, PSPO and SPS](#)
- [LeanKanban University - Practitioner \(TKP\) and Manager \(KMP\)](#)



Books

- Agile 學習手冊 | Scrum、XP、精實和看板方法 (Learning Agile: Understanding Scrum, XP, Lean, and Kanban)
- 告別瀑布，擁抱 Scrum：解析微軟與 Adobe 如何在 30 天內開發出新軟體 (Software in 30 Days: How Agile Managers Beat the Odds, Delight Their Customers, And Leave Competitors In the Dust)
- Essential Scrum 中文版：敏捷開發經典 (Essential Scrum: A Practical Guide to the Most Popular Agile Process)
- 精實開發與看板方法
- 精實企業 | 高績效組織如何達成創新規模化 (Lean Enterprise: How High Performance Organizations Innovate at Scale)
- 看板方法:科技企业渐进变革成功之道



Reference

- <http://danielcoding.net/scrum-part-1-agile-manifesto/>
- <https://www.codeproject.com/articles/1064114/agile-software-development-basics>
- <http://www.softwaretestingstudio.com/agile-values-model-process/>
- http://www.slideshare.net/jgothelf/lean-ux-getting-out-of-the-deliverables-business/12-Customer_collaboration_over_contract_negotiation
- [Scrum懶人包 – 10分鐘讀懂Scrum與敏捷軟體開發入門 \(含中文英文名詞對照 \)](#)
- [Part 1 – Agile UX: Understanding the Agile World from a UX Perspective](#)
- [Scrum \(software development\)](#)
- [你真的搞懂了什麼叫敏捷式 \(Agile \) 開發嗎?](#)
- <https://www.atlassian.com/agile/kanban>
- <https://kanbanize.com/blog/kanban-101-applying-wip-limits/>

