Scrum 框架 /skrʌm/

Scrum的創始人之一Ken Schwaber受《哈佛商業評論》論文影響，從1990年代開始實踐新方法，在1993年拯救了一個瀕臨失敗的專案。

創新的過程控制

Ken在1994年建立的「控制混亂」網站中，就指出不確定性管理的必要性，他認為自己與另一位創始人Jeff Sutherland對Scrum的研究總結背後蘊含著第一性原理：「複雜性過程需要試驗性的過程控制」。在那個年代，研發型的組織一定都是走在創新之路上的，現在很多研發組織實際走向了富士康模式，但是卻無法管理好。這背後的原因是技術與管理的進步使一些東西變得確定了，然後卻有更多新的不確定因素湧現出來，「控制混亂」前路漫長。

1995年，Jeff Sutherland應邀將《哈佛商業評論》的文章轉發給正在創立極限編程的Kent Beck。Jeff曾是越戰中美軍的戰鬥飛行員，對變化無常的戰場複雜形勢有著切身的體會，要想生存就必須要建立響應變化的能力，沒人能夠走直線的。除了那篇論文，他們還借鑒了精益思想、Timebox概念、SmallTalk社區的面向對象設計思想……等理論基礎，並於1995年正式在OOPSLA發布了Scrum研發過程論文，期間也受到Mike Beedle的幫助，正式提出了Scrum研發框架。根據Jeff的說法，Scrum在創立之初就受到高德拉特的約束理論和精益思想的影響，關注於「Muri, Mura, Muda」。

Muri：(日語：不合理)避免給予人員和組織不必要的壓力，現在也稱為「可持續發展」。

Mura：(日語：不均匀、不一致或瑕疵)消除跳變和不一致，進行整流，有助於達到超生產力(Hyperproductive)。最初幾個Sprint主要是消除流動中的干擾，從而使後續的Sprint更加順暢。

Muda：(日語：無駄-消除浪費)激進地消除活動中的7大浪費，持續改善。

直到2001年，在美國Snowbird和大家一起形成敏捷宣言，連同《Agile Software Development with Scrum》一書（中譯本《Scrum敏捷專案管理》譯者李國彪）和現代IDE的出現，Scrum逐漸流傳開來。

早年在探索Scrum實踐中，Jeff說類似極限編程的實踐也用了很多，但是後來為了從管理角度將敏捷更好地推廣，於是從Scrum中去掉了技術實踐的部分，泛化為一種工作方法，Scrum更多是一種管理實踐。Ken注意到對於軟體開發而言，技術能力實踐對Scrum落地必不可少，於是借鑑XP極限編程提出了Scrum Developer訓練內容。

Scrum框架的意圖

為什麼要採用Scrum？因為商業組織往往面臨這些痛點：

1. 產品做出來沒人用，忙得像狗卻沒有成就感

2. 計畫趕不上變化，需求變更太快太多，無法聚焦完成一件事

3. 業務與研發總是在拉扯，用戶總是在投訴

4. 時間緊、任務重，每件事情都是高優先狀態，總是在救火之中，做不完又焦慮

5. 所有壓力和風險都在後期集中爆發，越忙越添亂

6. 技術和架構跟不上時代，知識缺口越來越大，逐漸與社會主流脫節

7. 好的習慣和流程堅持不下來，形同虛設

8. 招不到優秀人才來幫忙，還不斷流失，團隊一盤散沙

9. 個人工作與生活失衡，從加班到辭職

Scrum框架的定義

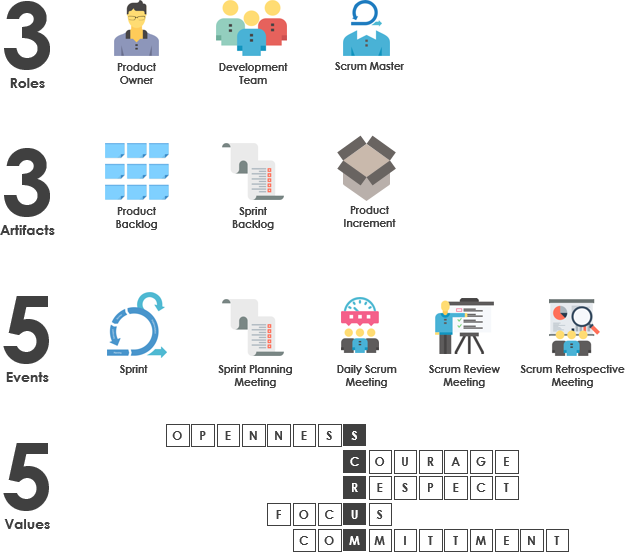
Scrum基於試驗性過程控制理論(Empirical Process)，或稱之為經驗主義，是自然科學研究方法的基礎。經驗主義主張知識源自實際經驗，以及觀察當前已知情況下做出決定所獲得。(注意區別於教條主義和理性主義，以及完全基於過去經驗進行判斷的過去經驗主義或固定思維，或者忽視理論指導的局部經驗主義)。「透明、檢視和適應」是試驗性過程控制的三大支柱，支撐起每一個試驗性過程控制的實施。Scrum採用一種迭代、增量式的方法來優化對未來的預測和管理風險，建立組織響應變化的敏捷能力，從而達致更好的效果。Scrum借鑑了精益思想、時間盒、模組化設計……等，並完整地體現了敏捷宣言和敏捷原則。

**Scrum**是一種通過**「檢視－調整」**來開發和維護複雜產品的框架，是遵循敏捷宣言和原則的一種流派，整合了**3個角色、3個工件、5個事件、5個價值觀，簡稱「3355」**。在這個框架中，整個開發過程由若干個短的迭代週期組成，稱為Sprint。

每個Sprint的建議長度是1～4週。使用產品Backlog來管理產品的需求，它是一個按照價值排序的需求列表。每個迭代中，Scrum團隊從產品Backlog中挑選最高優先級的需求進行工作。挑選出的需求在Sprint計畫活動上經過討論、分析和估算得到相應的迭代目標和交付計畫，我們稱它為Sprint Backlog。

迭代中每天會有一個站立式的Daily Scrum。在每個迭代結束時，Scrum團隊將邀請業務和利益關係人確認開發團隊的產出。隨後，團隊進行回顧，不斷改進工作方式。Scrum不僅適用於軟體開發專案，也可用於任何複雜的或是創新性的項目和探索，以及組織變革設計。

※迭代（iteration）是重複回饋過程的活動，其目的通常是為了接近並且到達所需的目標或結果。每一次對過程的重複被稱為一次「迭代」，而每一次迭代得到的結果會被用來作為下一次迭代的初始值。

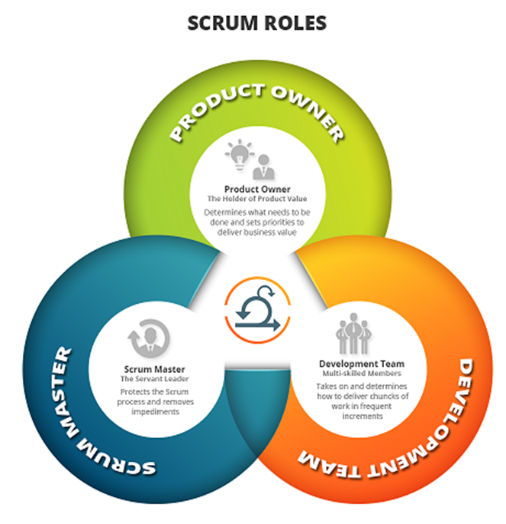


※Scrum框架定義了三種角色 (Roles)

1. Development Team：負責需求的軟體建置開發、部署

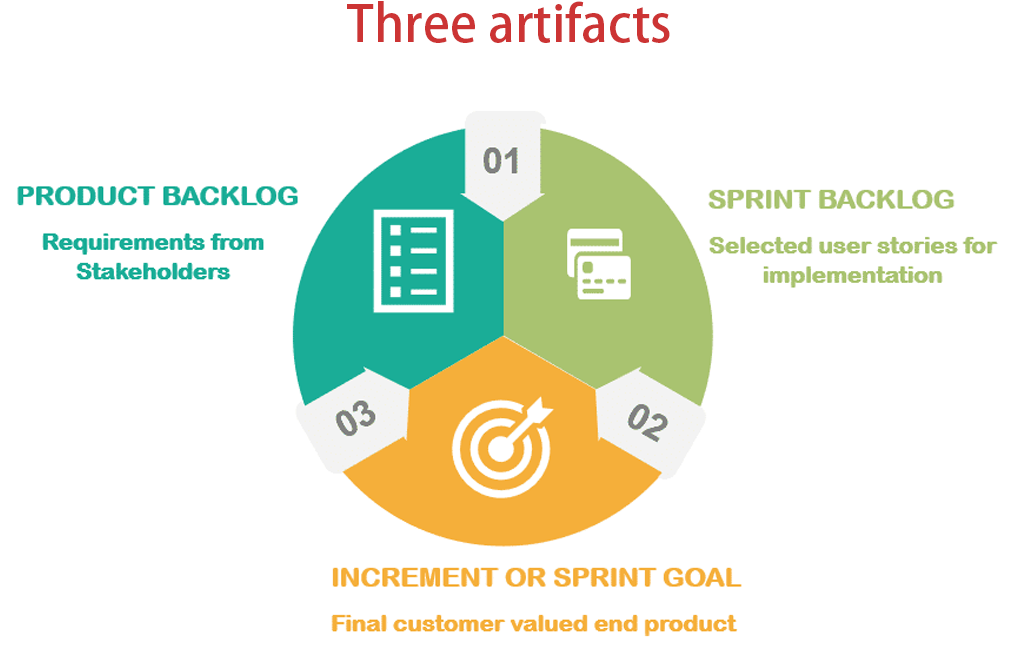
(2) Product Owner：負責決定軟體開發的功能

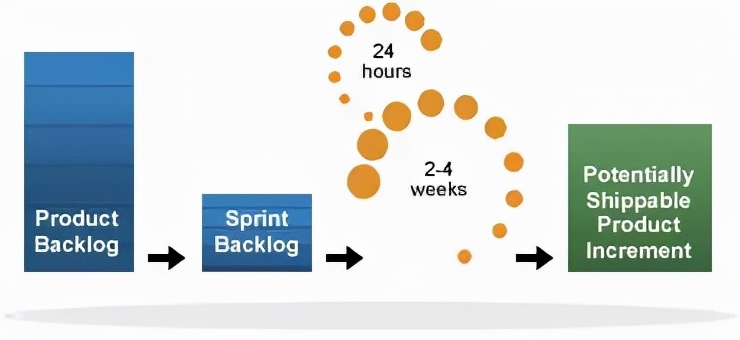
(3) Scrum Master：負責提倡以及確保 Scrum 在團隊中順利進行

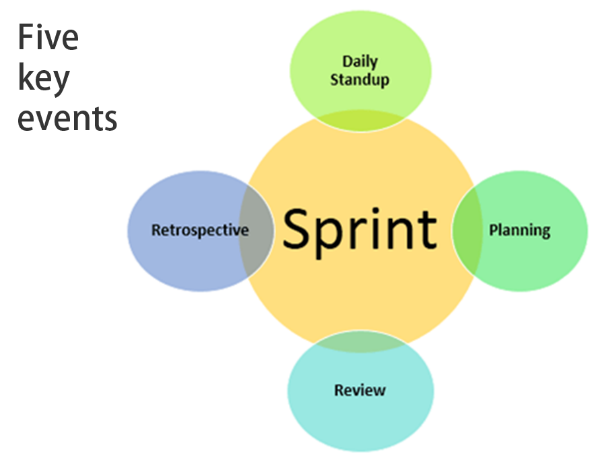


3個工件 (Artifacts) /ˈɑːr.t̬ə.fækt/

1. Product Backlog 產品待辦列表：就是產品的「What」。PO通過「講故事」的方式，讓團隊理解產品的目標，通過提問和講解，使整個團隊對用戶故事有充分和統一的理解。
2. Sprint Backlog 疊代待辦列表：是在每個Sprint的開始，按照優先級選出的將在該Sprint完成的目標故事。
3. Product Incremeny







※scrum框架定義了五個事件 (Event)

(1) Sprint：(Sprint本身是一個事件，包括下方4個事件)

(2) Sprint Planning Meeting計劃會議

(3) Daily Scrum Meeting每日站會

(4) Sprint Review Meeting 評審會議

(5) Sprint Retrospective Meeting 回顧會議

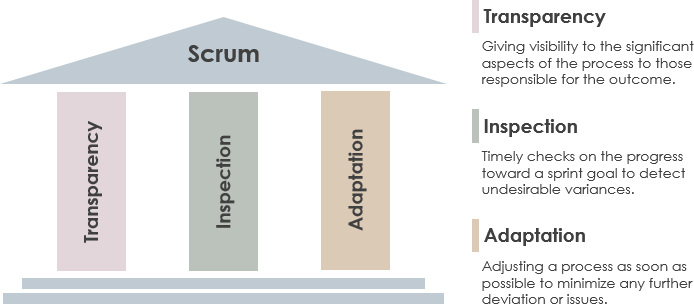
5個價值觀 (Values)

為了讓Scrum團隊能夠高效運作

1. Commitment 承諾：需要對目標承諾，
2. Focus 專注：有專注精神，
3. Courage 勇氣：接受挑戰的勇氣，
4. Openness 開誠：開誠布公的心態，
5. Respect 尊重：互相幫助，互相尊重；

當這些價值觀滲透到團隊之中，前面的335才能發揮出最大的價值，才能使團隊成為真正的「Agile Team」





3個支柱 (Pillars)

1. Transparent 透明
2. Inspection 檢查
3. Adaptation 適應

Agile的12原則

1、我們最重要的目標，是通過持續不斷地及早交付有價值的軟件使客戶滿意。

Scrum：產品需求列表中的優先級及每次迭代後持續提交工作軟件

2、欣然面對需求變化，即使開發後期也一樣。為了客戶的競爭優勢，敏捷過程掌控變化。

Scrum：產品需求列表可以隨時更新

3、經常地交付可工作的軟件，相隔幾星期或一兩個月，傾向於採取較短的周期。

Scrum：1-4週的的迭代周期

4、業務人員和開發人員必須相互合作，項目中的每一天都不例外。

Scrum：跨職能團隊及每日15分鐘的站立會議

5、激發個體的鬥志，以他們為核心搭建項目。提供所需的環境和支援，輔以信任，從而達成目標。

Scrum：團隊自我管理原則

6、不論團隊內外，傳遞信息效果最好和效率最高的方式是面對面的交談。

Scrum：5~9人的小團隊，kanban的使用以及開放環境的面對面溝​​通

7、可工作的軟件是進度的首要度量標準。

Scrum：對工作的軟件的關注，演示會要求提供可以被驗收和演示的內容

8、敏捷過程倡導可持續開發。責任人、開發人員和用戶要能夠共同維持其步調穩定延續。

Scrum：時間盒概念，迭代速率的使用

9、堅持不懈地追求技術卓越和良好設計，敏捷能力由此增強。

Scrum：要求對產品需求列表中每個用戶故事都定義一個完成標準

10、以簡潔為本，它是極力減少不必要工作量的藝術。

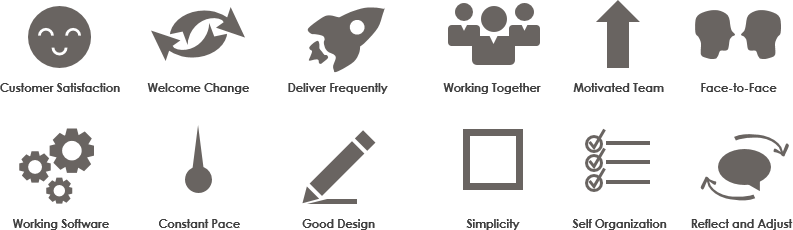
Scrum：盡量製作正好夠的東西已經在恰當時間做決策

11、最好的架構、需求和設計出自組織團隊。

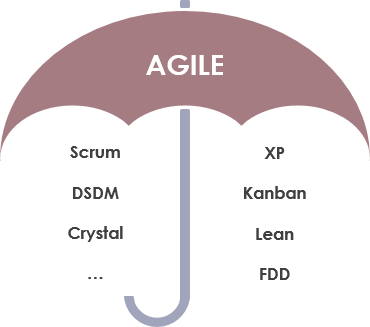
Scrum：如何做完全放在團隊手中

12、團隊定期地反思如何能提高成效，並依此調整自身的舉止表現。

Scrum：回顧會的作用



前 7 種流行的敏捷開發方法

1. [Scrum](https://zh.wikipedia.org/wiki/Scrum)
2. [動態系統開發法](https://zh.wikipedia.org/w/index.php?title=%E5%8B%95%E6%85%8B%E7%B3%BB%E7%B5%B1%E9%96%8B%E7%99%BC%E6%B3%95&action=edit&redlink=1)（DSDM）
3. [水晶方法](https://www.cnblogs.com/minjieagile/p/12931608.html) （Crystal）
4. [極限編程](https://zh.wikipedia.org/wiki/%E6%9E%81%E9%99%90%E7%BC%96%E7%A8%8B)（XP）
5. [看板管理（kanban）](https://zh.wikipedia.org/wiki/%E7%9C%8B%E6%9D%BF_(%E8%BD%AF%E4%BB%B6%E5%BC%80%E5%8F%91))
6. [精實軟體開發](https://zh.wikipedia.org/wiki/%E7%B2%BE%E7%9B%8A%E8%BD%AF%E4%BB%B6%E5%BC%80%E5%8F%91)（Lean）
7. [功能驅動開發](https://zh.wikipedia.org/w/index.php?title=%E5%8A%9F%E8%83%BD%E9%A9%85%E5%8B%95%E9%96%8B%E7%99%BC&action=edit&redlink=1)（FDD）
8. [自適應軟體開發](https://academic-accelerator.com/encyclopedia/zh/adaptive-software-development)（ASD）
9. [敏捷建模](https://www.ddiworld.com.tw/solution/test/talent_review/td) （DDI）
10. [敏捷統一過程](https://zh.wikipedia.org/w/index.php?title=%E6%95%8F%E6%8D%B7%E7%B5%B1%E4%B8%80%E9%81%8E%E7%A8%8B&action=edit&redlink=1)（AUP）
11. [紀律敏捷交付](https://academic-accelerator.com/encyclopedia/zh/disciplined-agile-delivery) （DAD）
12. [快速應用程式開發](https://zh.wikipedia.org/wiki/%E5%BF%AB%E9%80%9F%E6%87%89%E7%94%A8%E7%A8%8B%E5%BA%8F%E9%96%8B%E7%99%BC)（RAD）

From：https://kknews.cc/tech/e8qgp9z.html

From：https://blog.csdn.net/qq\_68292738/article/details/129757274

From：https://ithelp.ithome.com.tw/articles/10212375

From：<https://dotblogs.com.tw/Unified_Modeling_Language/2021/09/20/164837>

其他參考：https://medium.com/doflowy/%E4%BB%80%E9%BA%BC%E6%98%AFscrum-%E4%B8%8D%E6%98%AF%E5%B7%A5%E7%A8%8B%E5%B8%AB%E4%B9%9F%E8%83%BD%E6%87%82%E7%9A%84scrum%E5%85%A5%E9%96%80%E6%95%99%E5%AD%B8-1cc6683575f8