



理論與實踐輕量級指南

第 2.0 版

Eric Lee 譯

Pete Deemer - GoodAgile - [www.goodagile.com](http://www.goodagile.com)

Gabrielle Benefield - Evolve - [www.evolvebeyond.com](http://www.evolvebeyond.com)

Craig Larman - [www.craigarman.com](http://www.craigarman.com)

Bas Vodde - Odd-e - [www.odd-e.com](http://www.odd-e.com)

讀者提示：在網絡上可以找到很多簡明的 Scrum 描述，這篇簡介旨在提供更多在實踐方面的細節。這篇簡介成為 Scrum 學習的終點。建議考慮要引入 Scrum 的團隊用 Ken Schwaber 所著的《Scrum 敏捷項目管理實戰》，或者 Mike Cohn 所著的《Scrum 敏捷軟件開發》來裝備自己，並且充分利用眾多優秀的 Scrum 培訓和教練等資源，具體細節可以在 [scrumalliance.org](http://scrumalliance.org) 找到。

我們感謝 Ken Schwaber, Jeff Sutherland 博士以及 Mike Cohn 的無私貢獻。

本簡章的最新英文版本在：[http://www.infoq.com/minibooks/Scrum\\_Primer](http://www.infoq.com/minibooks/Scrum_Primer)

翻譯版本在：<http://www.scrumprimer.org/>

© 2012 Pete Deemer, Gabrielle Benefield, Craig Larman, Bas Vodde

## 超越傳統開發方式

傳統的開發方式由單一職能團體組成，反饋環延遲甚至薄弱，使用預言性質的計劃和從分析到測試流程，這在變化無常的當今世界中成效甚微。由於直到開發後期才会有真正可工作的軟件，這種方式會延遲反饋、學習，以及潛在投資回報。從而導致透明度不足、改進能力缺失、靈活度減少、商業和技術風險增加。

取而代之的是跨職能團隊與迭代開發，這種方式也存在了幾十年，但並不如傳統模式那樣被廣泛使用。

Scrum 把已被證實的產品開發概念打包到簡單的框架中，包括：真正的團隊、跨職能團隊、自我管理團隊、短迭代全週期反饋環、降低變動成本。這些概念提升了敏捷性與反饋、使提早實現投資回報成為可能並降低風險。

## 概括

Scrum 是跨職能團隊以迭代、增量的方式開發產品或項目的一種開發框架。它把開發組織成被稱為 Sprint 的工作週期。這些迭代每個都不超過 4 週（最常見的是兩週），並且無間歇地相繼進行。

Sprint 是受時間箱限制的，無論工作完成與否它們都會在特定日期結束，並且從不延長。通常由 Scrum 團隊來選定一個 Sprint 的時長，並且對於他們所有的 Sprint 都使用這一時長，直到這個團隊能力提高，可以使用較短週期。在每個 Sprint 的初始，跨職能團隊（大約 7 名成員）從排好優先級的列表中選擇事項（客戶需求）。團隊對於在 Sprint 結尾他們相信自己可以交付哪些目標集合達成一致意見，這些交付應該是有形的並且能被真正“完成”的。在 Sprint 過程中不可以增加新事項，Scrum 在下一 Sprint 時才接受變化，當前這麼短的一個 Sprint 週期裡只注重於短小、清晰、相對固定的目標。團隊每天都進行簡短會面來檢驗工作進程，並調整後續步驟以確保完成剩餘工作。在 Sprint 結尾，團隊與利益關係人一起回顧這個 Sprint，並演示所構建的產品。團隊成員從中獲取可以結合到下一 Sprint 中的反饋。Scrum 強調在 Sprint 結尾產生真正“完成”了的“可工作產品”。在軟件領域是指已經集成的、完全測試過的、已經為最終用戶生成文檔的、潛在可交付的系統。其中的主要角色、工件和事件如圖表 1 所總結。

# SCRUM

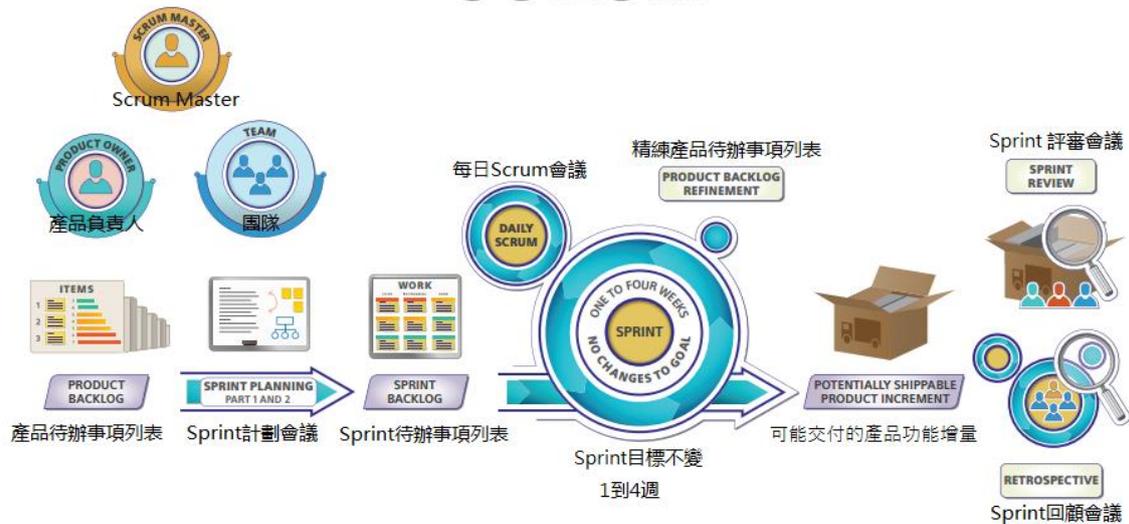


圖 1 Scrum 概括

Scrum 中的一大主題就是“審視與調整”。因為開發工作不可避免地包含學習、創新和意外事件，Scrum 強調進行小步驟開發，同時檢驗最終產品和當前實踐的功效，然後調整產品目標與過程實踐。周而復始。

## Scrum 中的角色

在 Scrum 中有三個角色：產品負責人，團隊和 ScrumMaster。他們在一起被稱為 Scrum 團隊。

**產品負責人**負責最大化投資回報 (ROI, return on investment)，通過確定產品特性，把它們翻譯成一個有優先級的列表，為下一個 Sprint 決定在這個列表中哪些應當優先級最高，並且不斷地重新調整優先級和梳理這個列表。對於商業產品而言，產品負責人對產品的贏虧負責。對於內部應用的情況，從非（可以產生收入的）商業產品的角度來講產品負責人為 ROI 負責，但從選擇每個 Sprint 具有最高價值事項的角度來講他們仍為最大化投資回報負責。在實踐中，“價值”的含意很模糊，優先級的制定可能會受希望讓關鍵客戶滿意、保持與戰略目標一致、解決風險、改進以及其他因素的影響。在有些情況下，產品負責人與客戶是同一個人。這在內部應用時很常見。有時，客戶可能是帶著不同需求的幾百萬人，在這種情況下，產品負責人與很多產品開發組織中的產品經理或者產品市場經理類似。然而，產品負責人與傳統的產品經理有些不同，因為產品負責人積極地並經常地與團隊互動，重點放在與所有的利益相關人一起工作並且評審每個 Sprint 的結果，而不是把開發相關的決定交由項目經理來代理。值得

注意的是，在 Scrum 中，有一個人，且只有一個人做產品負責人並擁有其最終的權力。他對工作所產生的價值負責。然而這個人卻不一定必須獨自工作。

**團隊（也叫開發團隊）** 建造產品負責人所指定的產品，例如應用程序或者網站。Scrum 中的團隊是“跨職能”的，它包含了每個 Sprint 為了交付潛在可交付的產品所需的所有專業能力，並且它是“自組織”（自管理）的，被給予很高程度的自治和責任。由團隊來決定在某個 Sprint 中要完成多少（從產品負責人給出的集合中的）事項，以及怎樣才能最好地達到這個目的。

團隊中的每個成員都只是“團隊成員”。請注意在採用 Scrum 的團體中沒有任何固定的專業頭銜。不會有業務分析員，沒有數據管理員，沒有架構師，沒有團隊組長，沒有交互/用戶體驗設計師，也沒有程序員，他們在每個 Sprint 中以任何恰當的方式一起工作來達到他們為自己設置的目標。

因為只有“團隊成員”，這種團隊不只是跨職能的而且還會表現出“多重學習”的特點。每個人當然都有特長，但仍繼續學習其他專長。每個人都會有主要的、次要的甚至是第三位的技能，並準備好“到工作所需要的地方去”。每個人都可能承擔自己並不熟悉領域的任務來幫助完成一個事項。例如，一個主要技能是交互設計的人可能以自動化測試為次要技能。以技術文檔寫作作為主要技能的人可能同時幫助做分析和編程。

Scrum 中的團隊由七個人左右（加上或減去兩個人）組成，對於一個軟件產品來講團隊可能包含具有分析、開發、測試、接口設計、數據庫設計、架構、文檔等等技能的人。團隊開發產品，並且向產品負責人提供如何把產品做得更出色的想法。在 Scrum 中，當所有成員在 Sprint 中都百分之百地投身於同一個產品的工作時，產量和效率最高。團隊避免了在多個產品或項目之間的多任務，從而避免了分散注意力和工作內容切換的高代價消耗。穩定的團隊會具有更高的生產率，因此要避免改變團隊成員。有很多人組成的產品開發團體要被組織成多個團隊，每一個都關注於產品的不同特性，緊密合作，共同努力。由於一個團隊往往要做所有完成以用戶為中心的特性所需的工作（計劃、分析、編程和測試），這樣的團隊也被稱為“特性團隊”。

**ScrumMaster** 幫助產品開發團體學習並應用 Scrum 來達成商業價值。ScrumMaster 會做任何力所能及的事情來幫助團隊、產品負責人和組織取得成功。ScrumMaster 不是團隊成員的經理，也不是項目經理、團隊帶頭人或者團隊的代表。相反，ScrumMaster 為團隊服務。他幫助移除阻礙，保護團隊免受外部干擾，並且幫助團隊採用現代的開發實踐。他教育、培養並指導產品負責人、團隊以及組織中其他的人來提高他們使用 Scrum 的技巧。ScrumMaster 是教練和導師。ScrumMaster 確保每個人（包括產品負責人和管理者）理解 Scrum 中的原則和實踐，並且他們幫助帶領組織完成敏捷開發要取得成功所需的那些通常都很困難的改變。

因為 Scrum 使得影響團隊和產品負責人有效性的很多阻礙和風險都變得可見了，由專注的 ScrumMaster 全身心地幫助解決這些問題就非常重要，否則團隊或者產品負責人會發現很難取得成功。ScrumMaster 應該是專門全職的，然而，小一點的團隊可能由其中一個團隊成員扮演這個角色（這樣做時他們只能肩負較輕的日常工作量）。出色的 ScrumMaster 可能來自於任何背景或專業：工程、設計、測試、產品管理、項目管理或者質量管理。

ScrumMaster 和產品負責人不可以是同一個人，因為他們的關注點太過不同，合併這兩個角色通常導致困惑和衝突。合併這兩個角色的一個常見的不幸結果是產品負責人進行微觀管理，這與 Scrum 所要求的自管理團隊相衝突。與傳統的管理者不同，ScrumMaster 不會告訴別人要做什麼或者分配任務，他們推進流程，為團隊的自組織和管理提供支持。如果 ScrumMaster 以前在管理團隊的崗位上工作，那麼要在 Scrum 上取得成功，他需要很大程度上改變與團隊互動的心態和思考方法。

注意：在 Scrum 中完全沒有項目經理這個角色。這是因為不需要。項目經理的傳統職責已經被分開並分配在 Scrum 的三個角色中了，主要是給了團隊和產品負責人，而不是 ScrumMaster。在實踐 Scrum 時加入一個項目經理意味著對 Scrum 基本的理解錯誤，典型的結果是責任衝突、職權不明和局部優化。可能一個（從前的）項目經理會擔當 ScrumMaster 的角色，但這能否成功很大程度上取決於個人，以及這個人對這兩個角色基本區別的理解程度，不光是日常職責，還有取得成功所需的思維方式。一個好的、能夠透徹地了解 ScrumMaster 這個角色，並開始培養取得成功所需的核心技能的方式是參加由 Scrum 聯盟所提供的認證 ScrumMaster 培訓。

除了這三種角色，還有其他的利益相關人也會為產品的成功做出貢獻，包括管理者，客戶和最終使用者。有些利益相關人，如職能性經理（例如工程經理）可能會發現他們的角色仍有價值但在引入 Scrum 時發生了改變。例如：

- 他們要以尊重 Scrum 的規則和精神的方式來為團隊提供支持
- 他們幫助移除團隊和產品負責人發現的阻礙
- 他們要確保他們的專業和經驗可以被利用

在 Scrum 中，這些個人把花在扮演“保姆”角色上的時間（分配任務、得到狀態報告以及其他各種形式的微觀管理）替換為扮演團隊的“導師”和“僕人”（指導、培養、幫助移除障礙，幫助解決問題，提供有創造性的輸入以及指導團隊成員的技能發展等）。在這種轉換中，管理者們可能需要改變他們的管理風格。例如，使用啟發式的問題來幫助團隊發現問題的解決方案，而不是簡單的決定一個方案然後分配給團隊去做。

## 產品待辦事項列表 (Product Backlog)

當一個團體計劃向 Scrum 過渡時，在首個 Sprint 可以開始之前，他們需要有一個（按 1、2、3 順序）排好優先級的、以客戶為中心的特性列表，即**產品待辦事項列表**。

產品待辦事項列表存在（並演化）於產品的整個生命週期，它是產品的路線圖（如圖 2 和 3）。任何時候，產品待辦事項列表都是“團隊依照優先排列順序完成工作”的唯一、最終的概括。一個產品只有一個產品待辦事項列表，這就意味著產品負責人必須縱觀全局做出優先級排列的決策，以體現利益相關人（包括團隊）的意願。

		New Estimates at Sprint ... Sprint的新估算							
優先級 Priority	事項 Item	細節 (wiki鏈接) Details (wiki URL)	估算 Initial Size Estimate	1	2	3	4	5	6
1	As a buyer, I want to place a book in a shopping cart (see UI sketches on wiki page) 作為買家，我想把書放入購物車（看wiki頁面用戶界面草圖）	...	5						
2	As a buyer, I want to remove a book in a shopping cart 作為買家，我想從購物車中刪除書	...	2						
3	Improve transaction processing performance (see target performance metrics on wiki) 提高交易處理性能（看wiki頁面的標性能指標）	...	13						
4	Investigate solutions for speeding up credit card validation (see target performance metrics on wiki) 探討加速信用卡驗證的解決方法（看wiki頁面的標性能指標）	...	20						
5	Upgrade all servers to Apache 2.2.3 將所有服務器升級到Apache 2.2.3	...	13						
6	Diagnose and fix the order processing script errors (bugzilla ID 14823) 分析並修復訂單處理腳本錯誤（錯誤號：14834）	...	3						
7	As a shopper, I want to create and save a wish list 作為購物者，我想創建並保存願望表	...	40						
8	As a shopper, I want to add or delete items on my wish list 作為購物者，我想增加或刪除願望表中的條項	...	20						

圖 2 產品待辦事項列表



圖 3 可視化管理：產品待辦事項列表事項貼在牆上

產品待辦事項列表包括不同種類的事項：全新的客戶特性（“使所有用戶可以將書籍放入購物車”），也有主要的技術改進目標（如“用 Java 代替 C++ 重寫系統”），還有改進目標（如“加速測試”）、調研工作（“探討加速信用卡有效驗證的解決方法”），還可能會有已知的缺陷（“分析並修復訂單處理腳本錯誤”），如果問題不多的話（如果系統有很多缺陷，通常就會有單獨的缺陷跟踪系統）。

產品待辦事項列表事項可以用任何明確且可持續的方式來表述。與常見的誤解正相反，產品待辦事項列表並不是由“用戶故事”組成的，而是只包含事項。這些事項可以通過用戶故事、用例或任何大家認為有用的需求方式來體現。但是無論何種方式，多數事項應當以交付客戶價值為重點。

一個好的產品待辦事項列表要 DEEP……

**粗細適宜的 (Detailed appropriately)**。優先級列表頂端的事項比低級別的事項更為精確和詳細，因為前者要比後者先被開發。比如，待辦事項列表頂端的百分之十可能包含非常小且分析得很詳細的事項，而其他的百分之九十則不是那麼具體。

**估算過的 (Estimated)**。當前發布版本的事項需要有估算，而且隨著大家了解得更多和新信息的融入應當在每個 Sprint 中重新考慮這些估算。團隊提供給產品負責人產品待辦事項列表中每個事項的工作量估算和技術風險估算。產品負責人和其他商業利益相關人提供產品需求的

價值信息，其中可能會包括獲得的收益、減少的成本、商業風險、對不同利益相關人的重要性等等。

**湧現式的 (Emergent)**。為了響應學習和變化，要定期梳理產品待辦事項列表。每個 Sprint，可能要加入、刪除、修改、分解或者調整事項的優先級別。因此，產品負責人會不斷地更新產品待辦事項列表，以反映客戶需求的變化、新想法或見解、競爭而導致的變化、出現的技術障礙等等。

**排好優先級的 (Prioritized)**。在產品待辦事項列表頂端的事項具有最高優先級，或者是從 1 開始順序排列。一般來說，最高優先級別的事項應當最物超所值：高收益（商業價值）低花銷（成本）。提高某事項優先級的另一誘因是在高風險來襲之前及早解決掉它。

傳統的開發方式通常不強調依照最物超所值的標準交付，但這是 Scrum 的主題之一，因此產品負責人就必須掌握如何評定“商業價值”。這可能需要 ScrumMaster 協助產品負責人來學習。“商業價值”到底意味什麼呢？一些產品團體對於每個產品待辦事項列表事項使用簡單、相對的價值點估算，這些就綜合成了對包括獲得收益、成本減少、利益相關人偏好、市場區分等等因素的“推測”。一些團體更是由一個或幾個客戶擔負費用來資助某特定事項的開發，然後用該事項的確切（短期）收益來代表價值。對於另一些團體，這種針對特定事項的價值估算太不集中或太細碎，他們使用範圍更大的、基於商業成果的方式（“在九月一日之前增加百分之十的訂購”），這樣價值只有在當多個對產生成果起作用的事項一起交付的時候才會實現。這種情況下，產品負責人需要確定最小可行產品（Minimum Viable Product）的下一個增量。

關於工作量估算，通常的做法是按照相對規模（影響工作量的因素、複雜程度及不確定程度）使用“故事點”或簡單的“點”為單元來估算。

這些都只是一些建議，Scrum 並沒有規定在產品待辦事項列表中對事項表述或排列優先級的方法，也沒有規定估算的方法或單位。

在 Scrum 中有一個常見的做法是追蹤每個 Sprint 完成的工作量，比如每個 Sprint 平均完成 26 點。如果保持平均水平並且沒有其他變化的條件下，根據這個信息他們可以預測出完成所有特性的發布日期，或到特定日期可以完成多少特性。這個平均值就稱為“速率”。速率與產品待辦事項列表事項規模估算用相同的單位來表達。

產品待辦事項列表中的事項在規模或工作量上可以有非常大的不同。較大事項會在產品待辦事

項列表梳理會或 Sprint 計劃會議上被分解為若干個較小事項，較小的事項也可能會被合併。接下來幾個 Sprint 的產品待辦事項列表事項應當是很小且足夠精細的，那麼團隊可以做到充分理解，使在 Sprint 計劃會議上做出的預期更有實際意義，這就是“拿來就能做”的規模。

耗費很多時間和資金的重大工程改進應當被包含到產品待辦事項列表中，因為它們很有可能是潛在的商業投資，最終由以商業為導向的產品負責人來決定。注意，在 Scrum 中，團隊對於從產品待辦事項列表中提取多少事項到當前 Sprint 中有自主的權力，因此他們可以獨立自主地進行較小的工程改進工作，這些工作會被納入到正常商業運轉成本中，而且這些也是對開發人員妥善完成工作的要求。這就是說在每個 Sprint 中，團隊的大部分時間通常應當放在產品負責人目標上，而不是內部工程任務。

對 Scrum 的一個誤解是說它避免撰寫詳細規格說明書，實際上，產品負責人和團隊決定他們需要多少細節而且每個待辦事項列表事項的要求都不同，這些都依據團隊的見解和其他一些因素來決定。要用最少的空間描述最重要的東西，換句話說就是不要描述事項的每個細節，只要清晰明了可以理解，並隨著團隊、產品負責人和利益相關人之間的不斷對話而擴充。低優先級的產品待辦事項列表事項，它們不會在近期內被納入工作，因此它們通常是“粗粒的”（大的、需求並不詳細的）。高優先級、細粒度的產品待辦事項列表事項將會在近期內被實現，它們的描述會更為詳細。

## 完成的定義

每個 Sprint 的輸出的正式名字為“潛在可交付產品增量”。在開始第一個 Sprint 之前，產品負責人、團隊和 ScrumMaster 必須審視對於把一個產品待辦事項列表中的事項做到潛在可交付所需要的所有事情。所有為了交付產品所需的活動都應被包含在“潛在可交付”的定義中，並且要在這個 Sprint 中完成。

遺憾的是，當團隊開始使用 Scrum 時他們通常做不到在每個 Sprint 都能交付出“潛在可交付增量”這個目標。這通常是因為團隊缺少自動化或者不夠跨職能（例如，技術文檔撰寫者還沒有被包含在跨職能團隊中）。隨著時間的過去，團隊必須要提高從而能夠在每個 Sprint 交付“潛在可交付產品增量”。但是為了開始，他們需要建立一個他們當前能力的基線。這會被記錄在“完成的定義”中。

在第一個 Sprint 開始前，產品負責人和團隊要對於“完成的定義”達成共識，“完成的定義”是創建“潛在可交付產品增量”所需要的活動的子集（對於一個好的團隊來講兩者是一樣的）。團隊會根據“完成的定義”來計劃他們在 Sprint 中的工作。

一個好的產品負責人總是會希望“完成的定義”與“潛在可交付”越接近越好，因為這樣會增加開發中的透明度並降低延遲和風險。如果“完成的定義”不等同於“潛在可交付”，那麼就會有工作被延遲到發布之前，這會導致風險和延遲。因此被延遲的工作有時被稱為未完成的工作。

Scrum 團隊應當持續地改進，這一點會表現在對“完成的定義”的擴展上。

## Sprint 計劃

**概要：**這是一個為 Sprint 做準備的會議，通常分成兩部分（第一部分關於“做什麼”，第二部分關於“怎麼做”）。

**參與者：**第一部分：產品負責人、團隊、ScrumMaster。第二部分：團隊、ScrumMaster、產品負責人（不強制參加，但是要能被找到來回答問題）。

**長度：**每個部分的時間箱的小時數與 Sprint 的周數相等。

在每個 Sprint 的開始，會召開“Sprint 計劃會議”。它分為兩個不同的子會議，其中第一個被稱為“Sprint 計劃會議第一部分”。

在 Sprint 計劃會議第一部分中，產品負責人和團隊審視產品待辦事項列表中產品負責人有興趣在這個 Sprint 中實現的那些高優先級的事項。通常，這些事項應當已經在前一個 Sprint 中（的產品待辦事項列表梳理會議裡）被很好地分析過了，因此在這個會上只有較少的需要在最後時刻澄清的問題。在這個會議中，產品負責人和團隊討論產品待辦事項列表中這些高優先級事項的目標和上下關聯，讓團隊洞悉產品負責人的思想。第一部分關注於理解產品負責人想要什麼以及為什麼需要它。

在第一部分的最後（總是很忙碌的）產品負責人可以離開，不過他必須在會議的第二部分過程中可以被聯繫到（例如，可以接電話）。

在第一部分中，團隊和產品負責人還可以設計出“Sprint 目標”。這是對 Sprint 目的的一個概括性的描述，在理想的情況下這些目的有一致的主題。Sprint 目標也為團隊實際可能的交付提供了範圍上的靈活性，因為儘管他們可能不得不拿掉一些事項（因為 Sprint 是受時間箱控制的），他們卻仍可以承諾交付與 Sprint 目標精神一致的、有形的並“完成”的一些東西。

Sprint 中所承接的這些事項應當有多大？每個事項應當被拆得足夠小，它在估計上只需要花比整個 Sprint 少得多的時間。一個常用的指導原則是一個事項從估計上來講要小得足夠由整個團隊在四分之一個 Sprint 或者更少的時間內完成。

Sprint 計劃會議第二部分關注於如何實現團隊決定要開始做的那些事項。團隊預期他們在 Sprint 結尾能夠完成多少事項，從產品待辦事項列表的頂部開始（換句話說，就是從對於產品負責人來講最高優先級的事項開始），並依次查看事項。以下是 Scrum 中的關鍵實踐：由團隊決定將完成多少工作，而不是由產品負責人分配給他們。因為團隊是基於他們自己的分析和計劃，這使得預期更可靠。雖然產品負責人對於團隊接受多少工作沒有控制權，但他明白產品待辦事項列表中的事項是從頂部開始拿取的，換句話說，就是先拿那些被他評為重要的事項。團隊可以為列表中很後面的事項進行遊說，這通常發生在團隊和產品負責人發現低優先級的某些東西很容易並可以恰當地與高優先級的事項合併時。

Sprint 計劃會議通常會持續幾個小時，但對於兩週的 Sprint 來講不應超過四個小時，團隊對於要完成的工作做出正式的預期，要做到這一點需要仔細的思考。第一部分和第二部分的時間箱長度是相等的。對於一個兩週的 Sprint 來講每個部分最多是兩個小時。

Scrum 並沒有定義具體如何做 Sprint 計劃會議第二部分。有些團隊用他們以前 Sprint 的速率來指導可預期多少工作。有些團隊會用先計算可容納工作量的更精細的方式。

當使用可容納工作量的方式時，團隊的 Sprint 計劃會議第二部分從估算每個成員有多少時間花在 Sprint 相關的工作上開始。換言之，就是他們的平均工作日減去花在參加會議、收發郵件、午餐休息等等時間。對於大多數人來講這樣的結果是平均每天 4–6 小時在 Sprint 相關的工作上，其他的時間花在了郵件、午餐、社交網站、開會和喝咖啡上。這就是團隊接下來 Sprint 可容納的工作量。算出了這個容量以後，團隊要算出他們在這些時間裡能完成產品待辦事項列表中的多少事項，以及團隊將如何完成它們。這通常從在白板上討論設計開始。一旦大家理解了整體的設計，團隊會把產品待辦事項列表中的事項分解成較細粒度的工作。在開始處理產品待辦事項列表中的事項以前，團隊可能會先關注於為前一個 Sprint 的回顧會議中所創建的改進目標生成一些任務。然後，團隊選擇產品待辦事項列表中的第一個事項，即對產品負責人來講最高優先級的事項，然後依次處理，直到他們“容量填滿”為止。他們為每個事項建立一個工作列表，有時由產品待辦事項列表中的事項分解出的任務組成，或者當產品待辦事項列表中的事項很小，只要幾個小時就能實現時，就簡單的由產品待辦事項列表事項組成。這個在 Sprint 過程中要被完成的工作的列表叫做“Sprint 待辦事項列表”（參見圖 4 和圖 5）。

		New Estimates of Effort Remaining at end of Day... 每日結束時所剩餘工作量的最新估計							
產品待辦列表事項 Product Backlog Item	Sprint中的任務 Sprint Task	志願者 Volunteer	初始工作量估 Initial Estimate of Effort	1	2	3	4	5	6
As a buyer, I want to place a book in a shopping cart 做為買家，我希望把書放到購物車中	modify database 修改數據庫		5						
	create webpage (UI) 創建頁面(UI)		8						
	create webpage (Javascript logic) 創建頁面 (Javascript邏輯)		13						
	write automated acceptance tests 寫自動化驗收測試		13						
	update buyer help webpage 更新買家幫助網頁		3						
	...								
Improve transaction processing performance 改進事務處理效率	merge DCP code and complete layer-level tests 合併DCP代碼並完成分層測試		5						
	complete machine order for pRank 完成pRank的機器順序		8						
	change DCP and reader to use pRank http API 把DCP和讀入器改為用pRank http API		13						

圖 4：一種建立Sprint 待辦事項列表的例子

在 Sprint 計劃會議的結尾，團隊為他們相信自己在 Sprint 結束時所能交付的工作設置一個現實的目標。過去，這被稱為 Sprint 的承諾，團隊承諾盡他們的最大努力來達到他們的目標。遺憾的是，這有時會被誤讀為血書一樣的諾言，而不是團隊認真地“決意做到”。為了避免這個混淆，現在 Sprint 的目標被稱做“預期”，用來與產品負責人溝通。

Scrum 鼓勵多面手，而不只是“在其位，謀其政”，例如“測試人員”只做測試。換言之，團隊成員“工作需要什麼就做什麼”並且盡全力貢獻。如果有很多測試工作，那麼所有的團隊成員都可能來幫忙。毫無疑問有些人更擅長測試技術（等等），這並不暗示所有人都是通才，而是指團隊成員們共同工作並且互相學習新技術。這樣一來，在 Sprint 計劃會議中生成並估計任務時，不必而且也不應該認為所有的任務都由那些能把任務“做得最好”的人來主動承擔。而應該是，當需要拿一個新的任務時一次只主動承擔一個任務，有目的地兼顧選擇那些需要學習的任務（可能通過與這方面的專家結對的方式）。這是為什麼不要事先在 Sprint 計劃會議中分配任務的原因之一，而應該在 Sprint 的過程中“按需”完成。

舉個例子來說，很少會出現某一特定任務可能必須要由張三來做，因為這對於其他人來講需要花太長時間來學習或者根本不可能學會的情況。比方說張三是唯一一個具有某種畫圖藝術細胞的人，而其他團隊成員到死都畫不出個簡筆小人兒來。如果這種現象並不少見，並且沒有隨團隊的學習而越變越少的話，那一定是有什麼地方不對了，可能有必要問一問把所有這些必須由張三來做的畫圖任務都放在這麼短的一個 Sprint 中是否可行。

很多團隊的 Sprint 待辦事項列表採用一個牆面大小的任務板的形式（通常叫做 Scrum 板），在 Sprint 過程中，任務（寫在報事貼上）在被標為“待辦”、“在製品”和“完成”的列之間遷移。參見圖 5。



圖 5：可視化管理 —— 貼在牆上的 Sprint 待辦事項列表任務

產品負責人可決定希望團隊進行一個新的事項工作，但是直到下個 Sprint 之前他將不能做出這個改動。如果有外部環境情況的出現嚴重地改變了優先級，並使得團隊繼續工作下去變成浪費時間，那麼產品負責人或者團隊可以終止這個 Sprint。團隊停下來，開始新的 Sprint 的 Sprint 計劃會議。這麼做所帶來的破壞通常是有好處的，它抑制了產品負責人和團隊訴諸於這種戲劇性變化的動機。

保護團隊避免在 Sprint 過程中改變目標會帶來一些強有力的正面影響。首先，團隊明白他們的目標絕對不會改變，這加強了團隊的關注度，從而保證了工作的完成。其次，它使得產品負責人慎重思考他在產品待辦事項列表中排在高優先級並讓團隊在 Sprint 中接受的那些事項。

通過遵守這些 Scrum 的規則，產品負責人得到兩樣東西。首先，他有信心團隊對於盡全力完成他們所選擇的一個現實明確的工作集是有承諾的。經過一段時間後，團隊對於在更為實際的預期下選擇和交付會變得相當熟練。其次，產品負責人要在下一個 Sprint 開始之前做出他想要在產品待辦事項列表中做的改變。在這段時間，增加、刪除、修改和重排優先級都是可能和

可以接受的。產品負責人卻不能改變已經在當前 Sprint 的開發過程中被選擇了的事項，他只隔一個或少於一個 Sprint 的時間就可以實現做出任何修改的願望。改變方向，改變需求、或者只是改變主意，這些圍繞著改變的症狀不見了。這可能是為什麼產品負責人也和其他人一樣熱衷於 Scrum 的原因。

## 每日 Scrum 會議

**概述：**在團隊成員間更新信息和進行協調。

**參與者：**團隊必須參加；產品負責人不是必須的；ScrumMaster 通常會在場，但要保證團隊自己主持會議。

**時長：**最長 15 分鐘。

一旦 Sprint 開始，團隊就要投身於另一個 Scrum 實踐，那就是每日 Scrum 會議。在每個工作日指定時間都會舉行這個很短的（15 分鐘，或者更短）會議。團隊中的每個人都要參加。為了讓它保持簡潔，建議每個人都保持站立姿勢。這是團隊同步他們的工作並互相報告遇到的阻礙的機會。在每日 Scrum 會議中，每個團隊成員一個接一個地向團隊的其他成員報告三件事：

- 1) 自上次會議以來完成了哪些工作？
- 2) 在下次會議前有哪些工作會被做完？
- 3) 遇到了什麼阻礙？

注意每日 Scrum 會議並不是向管理者報告狀態的會議。這是一個自組織團隊互相分享目前工作情況的時刻，從而幫助他們進行工作協調。有人要把這些阻礙記錄下來，ScrumMaster 負責幫助團隊成員解決它們。在每日 Scrum 會議中只有很少或不會有深入的討論，其形式只是做回答那三個問題的報告。如果的確需要討論，那麼在每日 Scrum 會議後馬上就會有一個或多個並行的跟進會議，但在 Scrum 中沒要求任何人必須參加這些會議。部分或者全部的團隊成員需要針對於他們在每日例會中所收到的信息進行調整時間跟進會議很常見，換言之，這是又一個審視並調整的周期。對於新接觸 Scrum 的團隊，通常情況下建議管理者或者其他會被認為有權威的人不要參加每日例會。這樣做的風險是會讓團隊感到“被監視”。它使得團隊在壓力之下每天都報告出重大進展（這是不切實際的期望），並使團隊不願報告問

題。並且它會破壞團隊的自管理，引入微觀管理。對於一個利益相關人，更有益的做法是在會議後再找到團隊，並幫助解決那些讓團隊進展慢下來的阻礙。

## 在 Sprint 過程中跟踪進度

Scrum 中的團隊是自管理的，想要做到這一點，團隊必須要知道自己做得怎麼樣。每天，團隊成員都會在 Sprint 待辦事項列表（圖 6）上更新他們對還需要多少工作量來完成他們當前工作的估計。一個也很常見的做法是有人把這些剩餘工作量加起來，然後在“Sprint 燃盡圖”上畫點（圖 7 和圖 8）。這個圖顯示每天新的對團隊完成工作所剩餘工作量的估計。理想中，這應該是一條向下的斜線圖，其軌跡指向 Sprint 最後一天“沒有剩餘工作量”的那一點。所以它被稱為“燃盡”圖。有時它看上去不錯，但通常不會這樣。這是產品開發的現實。重要的是它顯示了團隊向著他們目標的進展，這不是按過去已經花了多少時間（這對於“進展”來講是個無關的信息），而是按將來還剩餘多少工作量來表示的，這也就是團隊與他們目標間的距離。如果這條燃盡線在接近 Sprint 結尾時沒有向下接近完成點，那麼團隊需要做出調整，比如減小工作範圍，或者找到在保持可持續步調下更高效的工作方式。

儘管 Sprint 燃盡圖可以用電子表格來創建和顯示，很多團隊卻發現用貼在他們工作空間牆上的紙來展示，用筆來更新效果更好。這種“低技術含量/高感知度”的方案既快捷又簡單，並且比電腦上的圖表可見性更好。

		New Estimates of Effort Remaining at end of Day...							
		每日結束時所剩餘工作量的最新估計							
產品待辦列表事項 Product Backlog Item	Sprint 中的任務 Sprint Task	志願者 Volunteer	初始工作量估 Initial Estimate of Effort	1	2	3	4	5	6
As a buyer, I want to place a book in a shopping cart 做為買家，我希望把書放到購物車中	modify database 修改數據庫	Sanjay	5	4	3	0	0	0	
	create webpage (UI) 創建頁面(UI)	Jing	3	3	3	2	0	0	
	create webpage (Javascript logic) 創建頁面 (Javascript邏輯)	Tracy & Sam	2	2	2	2	1	0	
	write automated acceptance tests 寫自動化驗收測試	Sarah	5	5	5	5	5	5	
	update buyer help webpage 更新買家幫助網頁	Sanjay & Jing	3	3	3	3	3	3	
...									
Improve transaction processing performance 改進事務處理效率	merge DCP code and complete layer-level tests 合併DCP代碼並完成分層測試		5	5	5	5	5	5	
	complete machine order for pRank 完成pRank的機器順序		3	8	8	8	8	8	
	change DCP and reader to use pRank http API 把DCP和讀入器改為用pRank http API		5	5	5	5	5	5	
<b>Total</b>			50	49	48	44	43	34	

圖 6：每日在 Sprint 待辦事項列表中更新剩餘的工作量

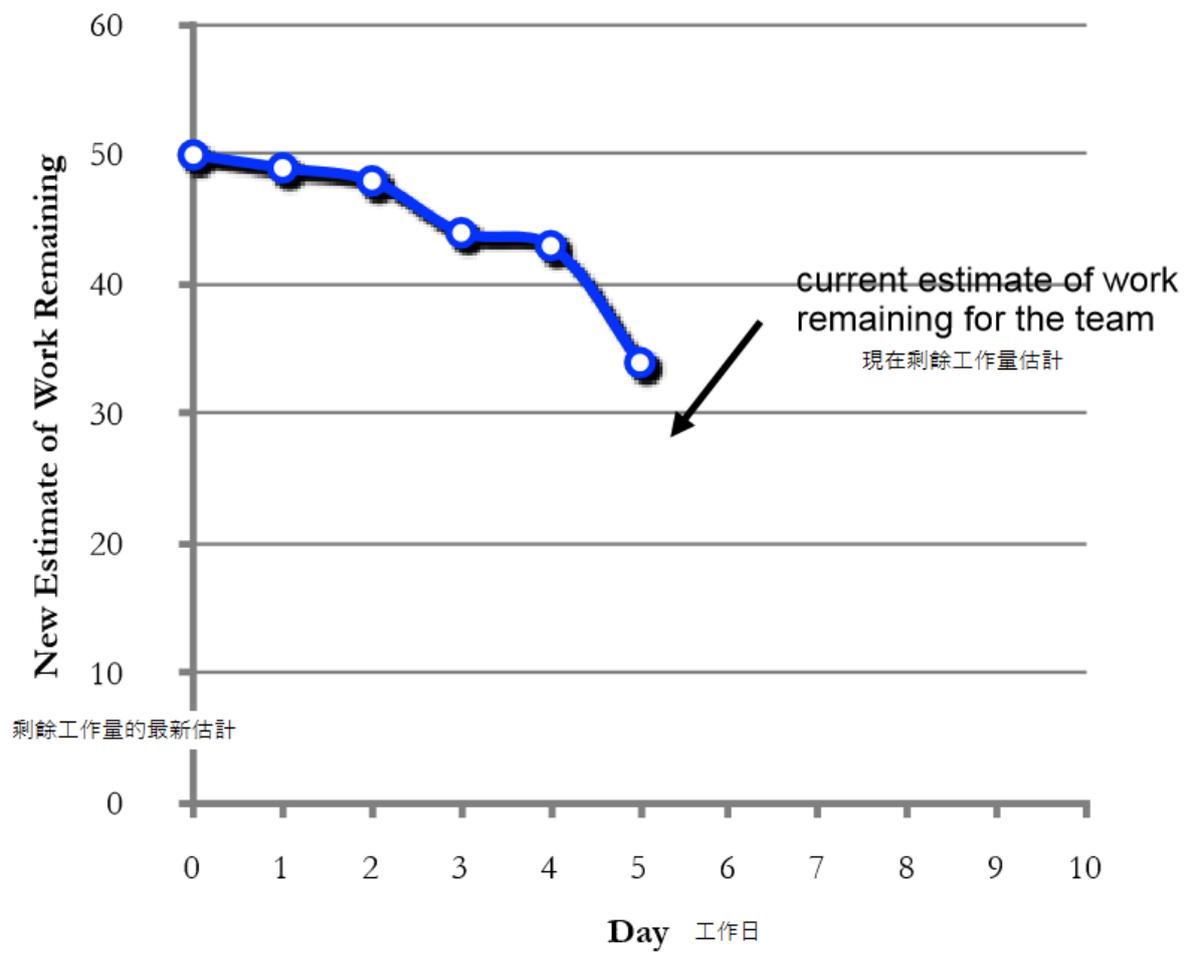


圖 7 : Sprint 燃盡圖

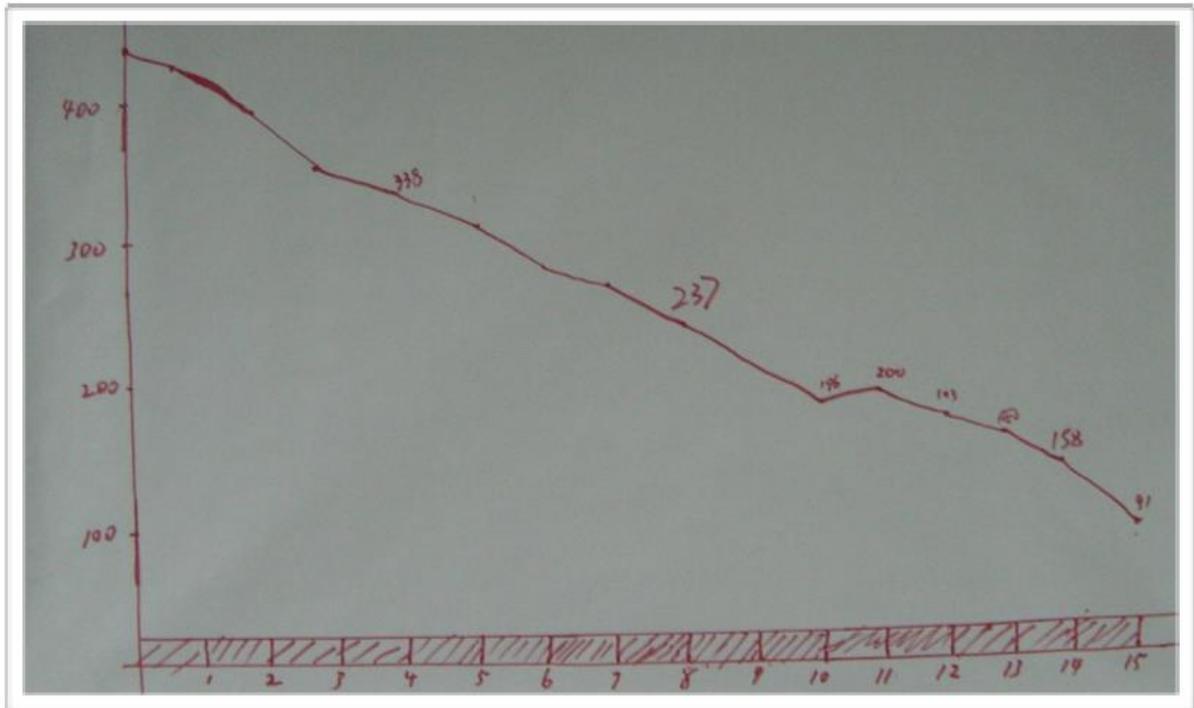


圖 8：可視化管理：手繪 Sprint 燃盡圖

### 產品待辦事項列表梳理

**概述：**為了將來的 Sprint 拆分大事項、分析事項、重新估計並重排優先級。

**參與者：**團隊；如果產品負責人是能夠幫助梳理細節的專家，那麼他會全程參與整個活動，否則他可以只參與其中一部分來設定方向或者重排優先級；其他理解需求並能給予團隊幫助的人；ScrumMaster 將在會議的開始部分來指導團隊如何有效地開這個會，否則的話他不必參加。

**時長：**通常來講不多於團隊一個 Sprint 工作容量的 10%，但有時對於“有大量分析工作”的事項來講要更長一些。例如，對於兩週的 Sprint，可能要有一天的時間花在梳理上。

Scrum 中一個很少有人知道，但很有價值的指導原則是，每個 Sprint 中整個團隊要把一定百分比的時間專門用於梳理產品待辦事項列表上，做為對將來 Sprint 的支持。這包含了詳細的需求分析、把大的事項拆小、估計新的事項以及重新估計已有事項。Scrum 中沒有說這個工作該如何來完成，但是一個經常用到的方法是在接近 Sprint 中部或者結尾時開一個專注的研討會，這樣團隊和產品負責人以及其他利益相關人就可以不受打斷地專門做這些事情。

這種梳理活動並不是為了當前 Sprint 已經選擇的那些事項。它是為了將來的事項，最可能是為了下一兩個 Sprint。有了這個實踐方法，Sprint 計劃會議變得相對簡單些，因為產品負責人和 Scrum 團隊將從一組清晰的，被很好分析過並認真估計過的事項入手。沒有開過這種梳理工作研討會（或者沒做好）的特徵是 Sprint 計劃會議中出現重大的疑問或者發現，或者令人困惑並感覺不完整。這樣一來計劃工作通常要延續到 Sprint 中去，這顯然不是大家想要的。

## **Sprint 評審會議**

**概述：**對於功能性的產品增量進行審視並調整。

**參與者：**團隊、產品負責人、ScrumMaster。產品負責人可以邀請其他恰當的利益相關人參加。

**時長：**時間箱為 Sprint 中的每一周對應一個小時。

在 Sprint 結束後，就到了 Sprint 評審會議，人們會評審這個 Sprint。出席會議的人有產品負責人、團隊成員以及 ScrumMaster，再加上客戶、利益相關人、專家、領導層和任何有興趣的人。對於兩週的 Sprint 來講最長為兩個小時。任何參與者都可以問問題和提意見。

評審會議常常會被錯誤地打上“演示”的標籤，但這並沒有抓住這個會議真正的用意。Scrum 中的一個關鍵思想是審視並調整。觀察並了解發生的情況，然後根據反饋進行演進，不斷進行。Sprint 評審會議是對於產品的審視並調整活動。它讓產品負責人了解產品和團隊的當前狀況（也就是對於這個 Sprint 的評審），也讓團隊了解產品負責人和市場的當前狀況。因此，評審會議中的一個關鍵元素是團隊與產品負責人之間進行深入的對話來了解情況和得到建議等等。評審會議當然要包含使用團隊所創造的真的，運行起來的軟件，但如果評審的關注點只是在看產品而沒有進行交談，這麼做就失衡了。

Sprint 評審會議中“運行起來的軟件”這部分並不是由團隊來做“報告”，不會用到幻燈片。它是指親手來檢驗正在運行的真正的軟件，例如在一個用於開發的沙盒環境中。在評審會議議室裡會有一台或多台電腦，人們可以在上面檢驗和使用運行起來的軟件。最好有一個積極交互的過程，由真實用戶和產品負責人動手與軟件交互，而不是由團隊做出一個被動的展示過程。

盡量做到在不超過 30 分鐘的時間之內做完 Sprint 評審會議的準備，否則就意味著有什麼地方做得不對。

## Sprint 回顧會議

**概述：**針對流程和環境進行審視並調整。

**參與者：**團隊、ScrumMaster、產品負責人（非必須）。團隊可能會邀請其他利益相關人，否則這些人不准參加。

**時長：**時間箱為對應 Sprint 中的每一周為 45 分鐘。

Sprint 評審會議包含了對於產品的審視並調整。而在評審會議之後的“Sprint 回顧會議”則包含了針對流程和環境的審視並調整。這是團隊討論什麼方式能工作，什麼方式不工作的機會，並對要嘗試的改變達成一致意見。有時 ScrumMaster 可以扮演回顧會議的效率協調者，但可能還是找一個中立的外人來協調這個會議更好。一個好的做法是，ScrumMaster 們互相幫助協調彼此的回顧會議，這會起到在團隊間“交叉受粉”的效果。

組織 Sprint 回顧會議有很多技巧，《敏捷回顧》（《Agile Retrospectives》）一書中提供了很有幫助的一系列技巧。

很多團隊的回顧會議只關注於問題，這很不好。這使得人們以為回顧會議是某種讓人情緒低落的或者負面的事件。正相反，要保證每次回顧會議也關注於正面的事物或者優勢。有幾本關於“欣賞式探詢”的書在這方面給出了更多的點子。

回顧會議如果總是用同樣的分析技巧可能就會變得無聊起來，這樣的話就要隨著時間的推移引入不同的技巧。

## 開始下一個 Sprint

Sprint 評審之後，產品負責人可以根據任何新的見解來更新產品待辦事項列表，如添加新事項、刪除過時事項或修改現有事項。產品負責人負責保證這些變化反映在產品待辦事項列表中。圖 9 是個更新了的產品待辦事項列表的例子。

		New Estimates at Sprint ...						
		Sprint的新估算						
優先級 Priority	事項 Item <small>細節 (wiki鏈接)</small>	Details (wiki URL)	初始估算 Initial Estimate	1	2	3	4	5
1	As a buyer, I want to place a book in a shopping cart (see UI sketches on wiki page) <small>作為買家，我想把書放入購物車 (見wiki頁面用戶界面草圖)</small>	...	5	0	0	0		
2	As a buyer, I want to remove a book in a shopping cart <small>作為買家，我想從購物車中刪除書</small>	...	2	0	0	0		
3	Improve transaction processing performance (see target performance metrics on wiki) <small>提高交易處理性能 (見wiki頁面的標性能指標)</small>	...	13	13	0	0		
4	Investigate solutions for speeding up credit card validation (see target performance metrics on wiki) <small>探討加速信用卡驗證的解決方法 (見wiki頁面的標性能指標)</small>	...	20	20	20	0		
5	Upgrade all servers to Apache 2.2.3 <small>將所有服務器升級到Apache 2.2.3</small>	...	13	13	13	13		
6	Diagnose and fix the order processing script errors (bugzilla ID 14823) <small>分析並修復訂單處理腳本錯誤 (錯誤號: 14834)</small>	...	3	3	3	3		
7	As a shopper, I want to create and save a wish list <small>作為購物者，我想創建並保存願望表</small>	...	40	40	40	40		
8	As a shopper, I want to add or delete items on my wish list <small>作為購物者，我想增加或刪除願望表中的條項</small>	...	20	20	20	20		
...			537	580	570	500		

圖 9 更新了的產品待辦事項列表

Sprint 之間沒有間歇期，團隊通常在某個下午開 Sprint 回顧會議，第二天早上就直接進入下一個 Sprint 計劃會議（不包括週末休息）。

敏捷開發的一個原則是“可持續的步伐”，團隊只有保持合理水平的常規工作時間才可以一直繼續這一周期。生產力隨著團隊實踐的演變和剷除影響團隊生產力的障礙而增長，而不是通過加班加點或降低質量來取得。

Sprint 持續進行直到產品負責人決定產品可以發布。Scrum 的完美願景是在每個 Sprint 末尾生產出潛在可交付的產品，這就意味著不再需要任何如測試或文檔等周邊的工作。這意味著每個 Sprint 中所有的工作都完成了，可以在 Sprint 評審之後進行交付或發布。然而，許多組織都存在開發實踐、工具和基礎架構薄弱的問題，並不可能實現這個完美的願景，因此就需要“發布 Sprint”來處理剩餘的工作。當需要“發布 Sprint”時，我們當這是不得以而為之，這時組織的任務就是要改進他們的實踐直到不再需要這個“雞肋”的做法。

## 發布管理

有一個時而會被提到的問題，即在迭代模式中如何進行長期發布計劃。有兩種情況需要考慮：

- 1) 第一次發布的新產品，
- 2) 已有產品後來的發布。

在新產品或剛剛採用 Scrum 的現有產品情況下，在第一個 Sprint 之前需要進行初始的產品待辦事項列表梳理，產品負責人和團隊一起塑造一個恰當的 Scrum 產品待辦事項列表。這可能需要幾天甚至一周的時間，並且需要召開研討會（有時稱為初始產品待辦事項列表創建或發布計劃），和一些詳細的需求分析，還有被列為第一個發布的所有事項的估算。

在 Scrum 中出人意料的是，對於已經建立了產品待辦事項列表的在建造產品，下一發布應當不需要任何特殊或大量的發布計劃。為什麼呢？因為產品負責人和團隊應該每個 Sprint 都進行產品待辦事項列表的梳理工作（利用每個 Sprint 百分之五到十的時間），持續不斷地為未來做準備。這種“持續的產品開發”模式使得我們不再需要在傳統順序生命週期開發中見到的那種不停打斷的準備－執行－結果的階段性工作方式。

在初始產品待辦事項列表梳理研討會和每個 Sprint 中持續對待辦事項列表梳理的過程中，團隊和產品負責人將會進行發布計劃，並隨著知識的增加而梳理估算、優先級排列和內容。

有些發布是日期驅動的，比如“我們要在 11 月 10 日的展銷會上發布我們產品的 2.0 版本”。在這種情況下，團隊將會在有限的時間內盡量完成盡可能多的 Sprint（而且構建盡可能多的特性）。還有一些產品需要構建特定特性之後才可以被稱為完成，產品直到滿足了這些需求之後才可以發行（不管要多長時間）。既然 Scrum 強調每個 Sprint 生產出潛在可交付代碼，那麼產品負責人可以選擇開始做過渡性發布，以便客戶盡快收穫成品的好處。

因為他們不可能預測所有的東西，那麼重點就在於創建和梳理一份計劃以給發布提供一個大體的方向，並表明如何作出折衷決策（比如調整範圍還是調整時間表）。把這個當作引導你向最終目的地的路線圖。但是究竟選擇哪條路線和在行進過程中的決定都可以在途中確定。

“到達目的地比選擇何種途徑更重要”。

多數的產品負責人會選擇一個發布方式。比如，他們會決定一個發布日期，並和團隊一起估算在產品待辦事項列表中可以在該日期內完成的事項。被加入到當前發布的事項有時稱為“發布事項”。在需要“固定價格／固定日期／固定交付內容”承諾的情況下（比如，合同開發），這

些參數中的一個或幾個就必須有內置緩衝，以顧及不確定性和變化。在這方面，Scrum 與其他方法沒有不同。

## 對應用程序或產品的關注

於應用程序或產品（無論要投入市場，還是在組織內部應用），Scrum 讓團體遠離舊式的以項目為中心的模式，而轉向持續進行應用程序或產品開發的模式。從開始、中間到結尾的項目不復存在。因此，也不需要傳統的項目經理。而只是一個固定的產品負責人和一個長期、自管理的團隊，他們在一系列“無盡的”、固定時長的 Sprint 中協作，直到產品或應用程序停止開發。所有必需的“項目”管理工作由團隊和產品負責人（他可以是內部業務客戶或是來自產品管理的人員）共同處理，而不是由 IT 經理或項目管理辦公室人員來處理。

Scrum 也可以應用在一次性初創的（而不是創建或演進長生命期的應用程序）真正項目中。但是在這種情況下，團隊和產品負責人進行項目管理工作。

如果一個或幾個現有應用程序中沒有足夠的新工作以保證每個應用程序都有一個專注、長期的團隊進行工作怎麼辦？在這種情況下，穩定、長期的團隊可以在一個 Sprint 中負責一個應用程序的事項工作，然後下一 Sprint 負責另一個應用程序的事項。這樣做使得 Sprint 的時間通常都非常短（比如一周）。

有時候可能都沒有足夠的新工作以使用上面的解決方法，那麼團隊可以在同一 Sprint 中負責幾個不同應用程序事項的工作。但是，要注意這個方法可能會轉變成低效的跨越多個應用程序的多任務。在 Scrum 中的一個基本生產力主題就是在一個 Sprint 中，團隊集中精力於一個產品或一個應用程序的工作。

## 常見的挑戰

Scrum 不僅僅是一套具體的實踐，確切地說更為重要的是這個框架提供了透明度並給出了“審視並調整”的機制。Scrum 通過讓影響產品負責人和團隊效率的機能失調和障礙變得顯而易見，使它們得到及時處理。比如，產品負責人可能不清楚市場情況、特性或如何估算它們的相對商業價值。亦或者團隊可能對於工作量估算或開發工作還不夠熟練等。

Scrum 框架會迅速揭露這些弱點。Scrum 並不能解決開發的問題，只是讓問題都赤裸裸地顯現出來，並為人們嘗試用短週期和小改進試驗來解決問題提供了框架。

假設團隊因為任務分析和估算技能不佳，而不能交付他們預期在首個 Sprint 完成的工作。對於團隊而言，這是次失敗。但是實際上，這是個必需經歷的第一步，以使得團隊對於以後的預

期更加實際和周到。Scrum 的這種模式使得機能失調顯現出來，讓團隊能夠及時處理。正是這種基本的機制帶給了使用 Scrum 的團隊所能經歷到的最大好處。

一個常見的錯誤做法是當使用某個 Scrum 實踐遇到挑戰時改變 Scrum。比如，團隊交付有困難，他們可能決定延長 Sprint，那麼時間永遠都是充裕的，並且在這過程中確保自己永遠都不用學習如何更好地估算和管理時間。這樣，沒有經驗豐富的 ScrumMaster 的教練和支持，這個組織的 Scrum 就會蛻變成只是其自身弱點和功能不調的鏡像，並且破壞了 Scrum 所能帶來的真正利益：也就是使得好壞都顯露無遺，給組織提供自我提升的機會。

另一個常見的錯誤做法就是假定某個實踐是不准許或禁止的，只是因為 Scrum 沒有明確地提到。

比如，Scrum 沒要求產品負責人為產品製定長期策略，也沒要求工程師對於復雜的技術問題向經驗豐富的前輩討教。Scrum 將這些都留給相關的個人來做正確的決定，在多數情況下，以上兩個實踐（和其他一些）都是特別推薦的。

還有一個需要注意的是管理者強制團隊使用 Scrum。Scrum 提供給團隊空間和工具來自我管理，這種從管理層強壓下來的命令並不是取得成功的方法。一個比較好的做法就是讓團隊先從同事或管理者處了解到 Scrum，進行全面的專業培訓，然後由團隊決定在一定的時期內忠實遵循 Scrum 實踐，在該時期結束時團隊將評價其工作經歷，再決定是否繼續使用 Scrum。

雖然首個 Sprint 通常對於團隊來說是極富挑戰性的，但值得慶幸的是在 Sprint 結尾時 Scrum 帶來的好處就會顯現出來。所以很多新的 Scrum 團隊都驚叫：“Scrum 太難了，但是比起我們以前的做法它簡直是太好了！”

## 附錄 A：推薦閱讀

關於 Scrum 已經有很多發表了的材料。在這個參考部分，我們想推荐一些線上材料和幾本書以供閱讀。

線上材料：

The Lean Primer - An introduction to Lean Thinking, an important influence to Scrum.

<http://www.leanprimer.com>

The Distributed Scrum Primer - Additional tips for teams who aren't co-located.

<http://www.goodagile.com/distributedscrumprimer/>

The ScrumMaster Checklist - A list of question that good ScrumMasters use.

<http://www.scrummasterchecklist.org/>

Feature Team Primer - Scaling Scrum with Feature Teams,

<http://www.featureteams.org>

The Agile Atlas - Core Scrum. ScrumAlliance description of Scrum.

<http://agileatlas.org/atlas/scrum>

Scrum Guide - Scrum.org description of Scrum.

<http://www.scrum.org/Scrum-Guides>

Agile Contracts Primer - How to make Scrum-friendly contracts.

<http://www.agilecontracts.org/>

書籍：

《Leading Teams》 作者：Richard Hackman

《精益和敏捷開發大型應用指南》 作者：Craig Larman, Bas Vodde

《精益和敏捷開發大型應用實戰》 作者：Craig Larman, Bas Vodde

《Scrum 敏捷項目管理》 作者：Ken Schwaber

《Scrum 敏捷軟件開發》 作者：Mike Cohn

## 附錄 B：術語

### 燃盡

在一個 Sprint 中、一個發布中或者一個產品中，剩餘工作量隨時間變化的趨勢。原始數據來自 Sprint 待辦事項列表和產品待辦事項列表，剩餘工作量記錄在縱坐標上，時間（例如 Sprint 中的天數，Sprint 個數等）記錄在橫坐標上。

### 每日 Scrum 會議

每個團隊每天召開的短會，在這過程中團隊成員檢驗他們的工作，對工作和進度進行同步並把阻礙報告給 ScrumMaster 來移除。在每日 Scrum 會議之後可能會開跟進會議來調整後面的工作以優化這個 Sprint。

### 開發團隊

團隊的另一個名稱。

### 完成

所有參與者都認為為完成，並遵循了組織的標準、規範和指導原則。當在 Sprint 評審會議中把某些東西描述為“完成”時，它必須符合這個達成一致的定義。

### 剩餘工作量估計（Sprint 待辦事項列表事項）

一個團隊成員仍需要工作在某任務上的小時數的估計。當工作在這項 Sprint 待辦事項列表中的任務時，它的這項估計在每天結束時要進行更新。估計的值為所有預計剩餘的工作量，這和多少人工作無關。

### 增量

產品功能，由團隊在每個 Sprint 中開發，是潛在可交付的，或者對於產品負責人的利益相關人來講是有用的。

可能交付的產品功能增量

整個產品或系統的一個完整的部分，如果產品負責人或利益相關人選擇實現它，那麼它將是有用的。

Sprint

一次迭代，或者一個相似工作的重複週期，為產品或系統產出增量。不會超出一個月，通常大於一週。全部工作過程中的 Sprint 時長是固定的，工作在同一系統或產品上的團隊使用同樣長度的周期。

產品待辦事項列表 (Product Backlog)

一個排好優先次序的需求列表，包含了把它們變成完成的產品功能所需的時間估計。在產品待辦事項列表中優先次序越高的事項的估計就越精確。隨著業務條件和技術的變化，這個列表的內容會不斷湧現和改變。

產品待辦事項列表事項 (Product Backlog Item)

功能性需求、非功能性需求和問題，按照對於務業的重要性和依賴關係排優先級，並且對它有估計。估計的精確程度取決於產品待辦事項列表事項的優先級和粒度，在下個 Sprint 可能會被選中的最高優先級的事項粒度很小並且很精確。

產品負責人

負責管理產品待辦事項列表的人，也是負責最大化產品價值的人。產品負責人負責代表所有與項目及其所產生產品有利害關係人的利益。

Scrum

這不是個縮寫，是在英式橄欖球中死球後再度開球的機制。

ScrumMaster

負責 Scrum 流程、對它的正確應用以及把它的好處最大化的人。

### Sprint 待辦事項列表

團隊在 Sprint 中工作的列表。它通常被分解為一組更詳細的任務。這個列表在 Sprint 計劃會議中出現，可能會由團隊在 Sprint 中更新，按需刪除條目或增加新任務。Sprint 待辦事項列表中的每個任務在 Sprint 中都會被跟踪記錄，並顯示出其剩餘工作量估計。

### Sprint 待辦事項列表中的任務

團隊或團隊中的某成員把承諾的產品待辦事項列表條目變成系統功能定義的一個任務。

### Sprint 計劃會議

一個時間箱為 4 小時的會議（對於兩週的 Sprint），開啟每個 Sprint 的序幕。會議被分為兩個 2 小時的片斷，每個也受時間箱控制。在第一部分，產品負責人把產品待辦事項列表中的高優先級部分展示給團隊。團隊和產品負責人一起合作來幫助團隊決定他們能把多少產品待辦事項列表在下個 Sprint 中變成實際的功能。在第二部分，團隊通過做設計和把工作分解來計劃他們將如何做到這些，這樣他們就知道如何來達成 Sprint 目標。

### Sprint 回顧會議

一個由 ScrumMaster 協調的會議，整個團隊討論剛完成的 Sprint 並找出哪些改變可能會使得下一個 Sprint 更有趣或更有效。

### Sprint 評審會議

在每個 Sprint 的最後的一個時間箱為兩小時的會議（對於兩週的 Sprint），團隊與產品負責人還喔 利益相關人合作，一起檢驗這個 Sprint 的產出。這通常開始於對已經完成的產品待辦事項列表事項進行評審，然後是對於機遇、約束和風險的討論，再然後討論下一步最好做什麼（可能會導致產品待辦事項列表的改變）。只有完成的產品功能才可以演示。

### 利益相關人

與項目的產出有利益關係的人，可能因為他們是出資人、使用者或者是會被它影響到的人。

## 團隊

一組跨職能工作的人，負責自我管理在每個 Sprint 開發產品增量。

## 時間箱

不允許事件或會議超過的一個時間段。例如，每日 Scrum 會議的時間箱為 15 分鐘，無論如何都會在 15 分鐘後終止。對於會議，實際長度可以短一些。對於 Sprint，長度是不變化的。