

第七章

軟體專案成本管理

執行單位：國立臺灣科技大學
軟體工程學程中心

大綱

- 軟體專案成本管理簡介
- 軟體專案成本管理架構
- 軟體專案成本管理規劃
- 軟體專案成本預估

學習目標

■ 透過本章節希望學生能具備軟體專案成本管理與軟體專案成本管理相關的知識，例如：

- 軟體專案成本管理的架構、程序及其規劃
- 各階段應執行的工作項目
- 軟體專案成本預估的方法、模式與技術
- 透過軟體專案成本之管理可達成的目標

軟體專案成本管理簡介



何謂軟體專案的成本管理

- 規劃一個軟體專案及其子項目需要多少時間、工作量、成本與資源來完成。
- 基礎的成本管理活動：
 - 成本資料蒐集、成本會計建立、成本資料分析。
- 進階的成本管理活動：
 - 成本的配置、成本的降低。

為何要管理軟體專案的成本

- 控制軟體專案的成本以免預算編列過高或不足。
- 作為人員僱用及時程安排之基礎。
- 作為成本效益分析的基本資料，以提高軟體專案的利潤。
- 提昇與同業競爭者競標的優勢。
- 確保軟體專案的開發成功。

何時要做軟體專案的成本管理

■ 軟體開發專案的各個階段都需要進行成本管理：

● 早期

- 預估需投入的金錢、人員與設備等。

● 中期

- 監控與調配各項資源的運用狀況等。

● 晚期

- 軟體專案包裝與發布等。

如何管理軟體專案成本

- 專案成本資料的蒐集
- 專案成本之分析
- 預估成本模式的選擇
- 反復的改善及調整
- 學習經驗的累積
- 制度之建立

軟體專案成本管理之相關人員

- 高階管理者-統籌規劃競爭策略。
- 專案經理-人員僱用、時程安排、資源規劃等依據。
- 專案開發人員-確保軟體開發成本及品質合乎顧客的期望。

專案成本之績效管理

循成本控制基準以管理專案績效

- 釐定工作範圍 (要做什麼?)
- 將尚未發生的活動賦予「時間」座標 (何時做?)
- 委派資源 (誰去做? 用多少人、機、料?)
- 編列預算 (要多少錢?)
- 建立績效管理基線 (應做多少? 應該花多少?)
- 現況呈報 (做了多少? 花了多少?)
- 績效評估 (超前還是落後? 有沒有超支?)
- 早期預警 (問題出在那裏?)
- 精確預測 (何時做完? 還要多少錢?)

軟體專案成本管理架構



軟體專案成本管理程序

■ 各軟體成本管理程序間彼此互動並交互影響。

● 資源規劃

- 執行一專案所需人員、軟硬體工具等資源。

● 成本預估

- 估算完成某活動項目所需花費的金額。

● 預算編列

- 分配預估之預算至單一工作項目中。

● 成本控制

- 控制所有工作項目成本的變動。

資源規劃

■ 步驟1：輸入所需資訊

- 工作分解架構(WBS)，各子工作項目中應包括工作責任、細部任務、所需之工作時間等。
- 定義範疇，包括專案的內涵與目標，以作為資源規劃之依據。
- 可用資源內容，歷史資料及組織政策。

■ 步驟2：配合的方法、工具及技術與可供選擇之可行性方案。

■ 步驟3：產出資源規劃。

成本預估

■ 步驟1：輸入所需資訊

- 工作分解架構(WBS)、各資源的單位成本及其使用期間、資源規劃與歷史資料。
- 財務資料及各項預估之會計科目類別。

■ 步驟2：配合的方法、工具及技術與可供選擇之可行性方案。

■ 步驟3：產出成本預估資料。

- 各項工作項目之成本預估値及可容許範圍、相關說明文件與成本管理計畫。

預算編列

- 步驟1：輸入所需資訊。
 - 工作分解架構(WBS)、各工作要項之時程表、產出之成本預估。
- 步驟2：配合的方法、工具及技術以做好資源撫平。
- 步驟3：產出成本控制基準。
 - 據以評量及監督專案各階段成本之績效。

成本控制

- 審視影響成本基準的因素、實際成本與成本基準差異的原因並進行妥善的管理，以降低失敗風險。
- 步驟1：輸入所需資訊。
 - 成本基準、產出之成本預估、績效報告、變更成本基準之請求。
- 步驟2：配合的方法、工具及技術以做好實際發生成本之監督控制。
- 步驟3：產出成本控制資料。
 - 變更通知、更新預算計畫及修正作業、預測完工成本與詳細紀錄成本控制的相關資訊。

專案成本之監控步驟

- 編製專案計畫
- 執行計畫
- 監控並記錄各輸出資訊及實際發生情形
- 比對實際與規劃之間的差異
- 執行修正動作

專案成本之監控資訊

- 成本節餘、時程超前：
 - 專案執行有效率，狀況良好。
- 成本節餘、時程落後：
 - 專案執行缺乏效率。
- 成本超支、時程超前：
 - 專案為增加人力的趕工狀態。
- 成本超支、時程落後：
 - 專案執行遭遇重大問題。

因應各專案成本管理資訊之方案

- 成本節餘、時程超前：
 - 繼續保持該專案的執行成效。
- 成本節餘、時程落後：
 - 專案執行缺乏效率，故應立即加派人手以儘速趕工。
- 成本超支、時程超前：
 - 專案為增加人力的趕工狀態，最好修正增加預估之工作量或縮短專案的時程編排。
- 成本超支、時程落後：
 - 面對專案執行種種的變異，應重新修正人力配置及原預估的時程計畫。

軟體專案成本管理規劃



軟體專案成本管理之進行步驟

- ❖ 步驟1：子系統分解
- ❖ 步驟2：軟體規模估計
- ❖ 步驟3：專案初步預估
- ❖ 步驟4：成本影響因子設定
- ❖ 步驟5：子系統工作量之彙整
- ❖ 步驟6：應監控的成本資訊
- ❖ 步驟7：衡量成本績效
- ❖ 步驟8：分析專案現況
- ❖ 步驟9：調適修正

步驟1：子系統分解

- 建立完整的工作分解結構(Work Breakdown Structure, WBS)—清楚地定義工作的範疇。工作分解結構必須是以產品為導向之樹狀結構，包括硬體、軟體與服務，其分解深度則視管理所需，通常至少要分解至第三層。
- 將軟體開發專案分解成多個可獨立工作的子系統後或功能單元，這些子系統將分散在多個點同時進行，並由不同的軟體開發團隊負責實作。

步驟2：軟體規模估計

- 軟體規模可採用程式碼行數(Source Lines of Code, SLOC)或功能點數(Function Point, FP)衡量。
- 依軟體需求定義，計算未調整功能點數利用「未調整功能點與程式碼長度轉換率」，將未調整功能點數轉換為程式碼行數(SLOC)。

步驟3：專案初步預估

- 找出影響規模的因子，針對軟體規模大小透過指數關係來影響工作量(ex：前例可循、開發彈性、系統架構/風險解決、團隊內聚力、開發流程成熟度)。
- 將軟體規模**Size**、規模指數**b**及常數值**a**，代入基本等式以計算初步之工作量預估：

$$MM = a \times SIZE^b$$

步驟4：成本影響因子設定

- 針對不同的軟體開發環境之成本考量，一般以線性關係來影響工作量(例如：平臺困難度、人員能力與經驗、軟體工具與設備等)。
- 在早期設計階段，考量有些因素所知有限，如開發的軟體大小、目標平台的性質、參與專案的人員特質…等，可能無法適當評估，故在此一階段考量的環境影響因子對開發工作量的影響程度，應不同於在後期軟體開始建構階段的影響因子。
- 環境影響因子，分析其影響工作量的程度大小等級。

步驟5：子系統資訊之彙整

- 一大型軟體系統之開發會分解成多個可獨立作業的子系統或功能單元。
- 各子系統估計結果包含不同層次的估計資訊，包括軟體專案估計、子系統估計、軟體生命週期階段估計及工作項目估計等。

步驟6：應監控的成本資訊

- 以步驟5所彙整完成之子系統資訊，作為專案控制的基準，用以監控管制各分項工作之排程及資源分配、執行、檢核、計算實獲值管理績效指標及回報。

步驟7：衡量成本績效

- 比對各子系統實際耗用與所訂定之成本控制基準之成本差異及時程差異。
- 客觀地進行評估-
 - 實際完成之數量
 - 實際達成的階段
 - 尚須完成工作之期程

步驟8：分析專案現況

- 評量管制帳戶內細項工作安排之邏輯順序是否適當。
- 評量管制帳戶內細項工作之內容、資源配置、時間安排是否適當。

步驟9：調適修正

- 瞭解變異產生的原因及可能之因應對策後，就必須下決策是否應變更及如何變更績效評量基準。
- 調整策略可能包括：
 - 調整工作：重新規劃未完成的工作，使用所剩餘的時間及資源，達成原訂專案目標（時程、效能、成本）。
 - 調整專案：當所餘預算不足之時，重新規劃全案並提出追加預算額度。
 - 修改合約或變更需求。

軟體專案成本之分類

- 包括硬體、軟體、訓練、薪資等成本。
- 依成本會計的分類方式可分：
 - 人事成本、設備成本、費用及其他分攤費用等。
- 依軟體生命週期中區分的軟體成本
 - 開發階段成本：可劃分為需求分析、設計、編碼、測試各階段所佔開發成本的百分比。
 - 維護階段成本：可劃分為預防性維護、更正性維護、改善性維護各階段所佔維護成本的百分比。

軟體成本預估



成本管理之度量目標

■ 軟體開發成本管理主要是針對軟體開發過程中，規劃的專案活動與所花費的人力、成本及時間等投入開發軟體的資源預估之契合。

● 工作量(Effort)：

- 在軟體專案開發生命週期中，所需人力資源，一般以人月(Person-Month, PM)為單位。

● 成本(Cost)：

- 針對人力所需工作量之成本預估。
- 依照成本會計的分類方式。

● 開發時程(Duration)：

- 在時程未定的狀況下，可透過所預估的工作量而得。
- 在時程固定的狀態下，預估出人力在各階段的分配。

計算軟體專案規模大小之成本

- 影響軟體開發成本的因素有很多，軟體大小是影響軟體成本的最重要因素。最普遍被使用的度量指標為：
 - 程式碼行數(Lines of codes, LOC)：
 - 以開發者的觀點，並取決於程式語言且應用最廣。
 - 功能點數(Function Points, FP)：
 - 從使用觀點出發的度量指標，是度量開發軟體系統規格功能性大小的方法。

軟體成本的影響因子

- 一般軟體之成本/工作量影響因子乃是軟體開發專案中，顯著對軟體成本/工作量產生影響的屬性集合：
 - 產品屬性：
關於軟體產品發展的特性需求。
 - 平臺屬性：
電腦硬體平台所造成的軟體限制。
 - 人員屬性：
關於此專案人員的相關工作經驗與能力所造成的影響。
 - 專案屬性：
有關軟體開發專案的特殊性質。

軟體成本預估步驟 (1/2)

■ 已建立起過去軟體開發專案的歷史資料庫時：

- 從所建的資料庫中找出對開發成本有顯著影響的軟體成本因子。
- 依據本身組織過去歷史資料，決定採用哪些方法以建構軟體成本預估模式。
- 比對不同建構方法下的預估結果並加以分析與實際值之間的誤差來不斷地修正模式。

軟體成本預估步驟 (2/2)

■ 未蒐集過去軟體開發專案的歷史資料時：

- 選擇使用已建構完成的模式中，較適用於欲開發之軟體性質的成本預估模式。
- 依據本身開發環境及軟體屬性的顯著影響軟體成本因子來修正軟體成本因子參數值及其權重。
- 比對使用幾個不同預估模式下的預估結果並加以分析與實際值之間的誤差來不斷地修正。

軟體成本預估方法

■ 根據軟體開發生命週期階段

- 依循開發生命週期各階段來進行預估，可以避免遺漏重要開發流程的成本項目估算。
- 各開發流程階段，如需求分析、設計、編碼、測試等所佔開發成本的百分比不同。

■ 根據各主要功能性之系統模組作為成本架構分類

- 依照軟體系統模組需求的功能性所預期耗費的成本來進行預估，較接近使用者的觀點。

預估成本之流程

■ 由上而下 (Top-Down) 的應用

- 依過去經驗或歷史資料各開發階段所佔的比率來分配總預估值到其下的各子階段中。
- 依據個別子系統的複雜度及實作上的困難度來分配整個系統層面下各子系統所需工作量的估算值之比率。

■ 由下而上 (Bottom-Up) 的應用

- 由開發生命週期各階段下，所有子階段中的主要細部工作項目所需的成本相加總而得。
- 加總估算出的各模組或子系統所需花費的工作量，以得到整個系統開發工作量的預估值。

軟體成本預估模式

■ 參數式預估模式(Parametric Models)：

- 依據過去開發完成軟體專案的歷史資料，找出影響開發工作量的因素，再以統計迴歸(Statistical Regression)的方法推導出軟體工作量的估算方程式，例如：COMOMO、SLIM、Estimacs。

■ 類比式預估模式(Analogy Models)：

- 將資訊系統專案特徵化(Characterization) 如：系統複雜度(Complexity)、專案類別或開發平台等，來與歷史資料作相似度的比對，進而挑選出最相似的歷史專案，而得到估算結果，例如：COSTAR。

軟體成本與工作量

- 軟體專案開發的生命週期有許多不同的階段。
- 各個階段中的活動都有其所需的資源，一般而言可將所需的資源區分為兩部分：
 - 1、開發環境
 - 2、人員
- 所謂的軟體成本是指在軟體專案開發生命週期中，各個階段所需人力的花費。
- 由於各公司的薪資結構或國家貨幣不同，因此將過去所預估使用軟體成本(Software Cost)改成軟體工作量(Software Effort)。

模式的使用 (1/3)

■ 專家經驗模式 (Expert Experience-Based)

- 是目前國內軟體產業界最常使用的軟體成本預估方法，由專案經理本身的經驗為基礎，輔以過去開發過的軟體專案資料來得到主觀的專案預估值。
- 例如：TRW Wolverton模式、德菲法、計畫評核術等。

模式的使用 (2/3)

統計迴歸分析模式 (Regression Analysis)

- 以統計迴歸分析所建構的軟體工作量預估模式，模式中工作量影響因素與預估工作量之間的關係為線性或非線性。

一般而言線性模式可表示為：

$$E = C_0 + \sum_{i=1}^n C_i X_i$$

非線性模式的表示：

$$E = (a + bS^c)m(x)$$

模式的使用 (3/3)

■ 人工智慧模式(Artificial Intelligence, AI)

● 類神經網路模式(Artificial Neural Networks, ANN) :

- 一種包含硬體與軟體平行計算系統，使用大量的相連人工神經元來模仿生物神經網路的能力，以達到類似人腦所能處理複雜問題的能力。

● 案例推論(Case-Based Reasoning, CBR) :

- 應用相似組織中有相似類型本質的問題，透過使用那些相關性高及可被利用的資料去做預測。
- 案例庫可有效率的儲存歷史專案的資料。
- 當案例庫中完全缺乏過去可用的相似問題之專案資料，會容易導致預測的偏差。

COCOMO

以三種模式來表示不同的詳細程度

- 基本模式(Basic Model)：只考慮軟體規模。
- 中級模式(Intermediate Model)：考慮軟體規模及15個成本調整因子。
- 詳細模式(Detailed Model)：生命週期開發階段之各成本比率不同。

每種模式又依軟體雜程度分為三種，分別為

- 有機型(Organic Mode)：小型、簡單、低複雜度、低風險的專案。
- 中間型(Semi-Detached Mode)：中等規模、中等複雜度的專案。
- 內嵌型(Embedded Mode)：高複雜度、限制條件嚴格的專案。

COCOMO II

- 依循3階段的開發進行模式來預估：
- 早期雛形階段：應用組合模式 (Applications composition)
 - 依據應用點與簡單的公式來進行工作量的預估。
- 早期設計階段：早期設計模式 (Early design)
 - 依據功能點來進行預估後，再轉換為程式碼行數。
- 後期結構階段：後期結構模式 (Post architecture)
 - 依據程式碼行數來進行預估。

工具的使用

- 應用專案管理工具及技術，增加軟體專案成本管理的便利性與效率，電腦化工具如：
Project、Excel。
- 計畫成本與實際成本之追蹤比較、預測成本變更的影響，且便於蒐集專案之歷史資料。
- 透過工具做好各種有關專案成本資訊之整合、WBS與時程安排、要徑分析與專案排程、資源分派、資源撫平、專案成本比較基準之應用、產生管理報表等。

結論

- 軟體成本管理是如何讓軟體專案於規劃的時間與經費下完成，並在特定時間內及變動的環境下滿足客戶的功能需求。
- 可以基於資源分配優先順序的不同，並透過逐步調整資源的投放日期和數量，以實現資源均衡。
- 找出可作為專案成本度量和計算的顯著因子，並對軟體的特性和工作環境有一定程度的了解，下對專案成本進行計劃和控制。
- 依照成本管理規劃來監督費用實施情況；確保所有有關變更都準確地記錄在成本管理項目中；將確認的變更通知專案中的相關人員。

自我評量

- 什麼是軟體專案成本管理？
- 有哪些工作項目？
- 有哪些已廣泛被應用的模式、方法與技術？
- 軟體專案成本管理之程序為何？
- 執行軟體專案成本管理之預期效益為何？

參考資料

- 黃世禎老師, “軟體指標度量與分析方法”
- 林信惠.黃明祥.王文良, “軟體專案管理”, 智勝文化, 2002 .02.01
- Hareton Leung Zhang Fan, "Software Cost Estimation," Department of Computing , The Hong Kong Polytechnic University, (2001).
- Sarah Jane Delany, Dublin Pádraig Cunningham, "The Application of Case-Based Reasoning to Early Software Project Cost Estimation and Risk Assessment," Department of Computer Science DIT Kevin Street, Department of Computer Science Trinity College Dublin, (2000).
- Kevin Strike, "Software Cost Estimation with Incomplete Data," IEEE Transactions On Software Engineering, Vol. 27, No. 10, pp. 890 (2001).
- Barry W. Boehm, "Software Cost Estimation with COCOMO II," Prentice Hall PTR, (2000).
- Symons, C. R., "Function point analysis: Difficulties and improvements," IEEE Transactions On Software Engineering, Vol. 14, pp.2-11 (1988).
- A Winsor Brown USC-CSE, University of Southern California Center for Software Engineering , "Emerging Extensions “, CIF 15 Oct. 24, 2000
- Nancy Merlo-Schett, Department of Information Technology University of Zurich, "COCOMO “, seminar cost estimation W 2002/2003