



品質為基礎的持續改善程序以降低軟體專案風險

A Quality-based Continuous Improvement Procedure for Reducing Software Project Risk

賴森堂 Sen-Tarng Lai¹

摘要

軟體專案的成功比率一直都很低，失敗率卻高達 40% 以上，專案失敗的三個關鍵因素分別為：時程延誤、成本超出預算、以及品質不能滿足使用者需求。計畫變更是軟體專案開發過程無法避免的挑戰，變更內容往往反映在已規劃的經費、時程與資源等相關項目上，能夠克服各式各樣變更衝擊，才能降低專案的失敗風險。軟體專案管理是由多項管理作業所彙集而成，其中成本、時程與品質管理是形成專案失敗的主要關鍵，軟體開發過程中，受到計畫變的影響，經常成為專案失敗的關鍵。為了降低計畫變更的高風險，本文探究專案管理的相關作業，並剖析與蒐集成本、時程與需求的關鍵品質因子，進而提出一套專案管理品質量測(PMQM)模式，標示專案管理作業的品質問題與缺失，且建立一套以品質為基礎的持續改善程序(QCIP)，適時擬定改善方式與矯正措施，以高品質的管理作業降低軟體專案風險。

關鍵詞：專案管理、品質因子、品質量測模式、專案風險、PMQM

Abstract

Software project failure ratio always is very high. Three critical reasons of project failure are cost over budgets, time schedule serious delay and quality cannot meet user requirement. In addition, plan change is the software development project has to face challenge. Able to overcome the impact of various plans change, software development risk can be reduced. Any one plan change case to the software project will result in significant impact. The plan change causes the completed file must be revised, the budget must be re-estimated, schedule must be adjusted, and quality must be reassured. And plans change often became the failure key of software project. In order to reduce software project development risk. In this paper, survey the related operations of project management, analyzing and collecting the quality factor of critical project management operation, and propose a Project Management Quality Measurement (PMQM) model. Based on PMQM model, build a Quality-based Continuous Improvement Procedure (QCIP). Using the QCIP to accept and handle plan change, the project management quality can be controlled and software development risk can be reduced.

Keywords: project management, measurement model, quality factor, development risk, PMQM

¹實踐大學資訊科技與管理學系助理教授(聯絡地址：104 台北市大直街 70 號，聯絡電話：02-25381111 轉 1820，E-mail: stlai@mail.usc.edu.tw。)

壹、緒論

軟體專案管理計畫(Software Project Management Plan; SPMP)主要著重於經費、時程、品質與資源等項目的規劃(Schach, 2008),而這些規劃項目則取決於完整的需求分析與精確的成本估算,能夠精確估算出軟體開發成本,才能具體且有效的擬定出軟體專案的經費、時程、品質與資源等管理計畫(Boehm, 1981)、(Pressman, 2010)。軟體開發過程中,各個階段都必須達成其特定任務且完成既定的產品文件,每件撰寫完成的規劃項目與開發文件,都要通過嚴格的檢視與審查步驟,才能發行且成為後續開發作業遵循及引用的依據,其中各個階段的開發文件還必須填入撰寫人員、完成日期、審查意見等相關資訊,再依建構管理(Configuration Management; CM)機制將完成之文件交付管制(Pressman, 2010)。不過,受到錯誤更正、使用單位提出的需求變更、技術與環境的演進及計畫相關資源的調整等事件的影響,迫使軟體專案計畫必須不斷的進行調整與修訂,隨之衍生出軟體開發經費持續追加、時程一再往後延、而品質不易掌握等不利的現象,其中又以使用單位經常無預期的提出需求變更案最為關鍵。因為一旦計畫變更案通過審核,已完成的規劃項目與開發文件必須進行調整或修訂,不僅在投入的經費、開發的時程與產品的品質造成實質面的衝擊,也對員工士氣、人員的流動及客戶的信賴度造成心態面的影響,造成高度開發風險更是軟體專案失敗的主因。因此,為了有效降低軟體專案的開發風險,就必須深入剖析專案管理的關鍵品質,且規劃一套持續改善程序管制專案管理品質,才能減輕計畫變動的衝擊,具體降低軟體專案開發風險。

依據分析軟體開發專案的知名顧問公司 Standish Group 對 9236 個軟體專案的剖析報告中,指出只有 29% 的軟體專案是成功的,18% 於執行過程中被取消,而剩餘的 53% 是屬於延後完成、超出預算或是功能未能滿足需求等狀況(Hayes, 2004)。探究軟體專案失敗的主要原因就是計畫變更過於頻繁,軟體開發過程中,因應各種突發狀況所提出的計畫變更案,迫使已完成的產品文件與規劃項目必須不斷調整與修訂,因此變更作業品質將主導變更後的開發作業,缺乏高品質的變更作業,將導致規劃項目無法及時且有效的反映在預算規劃、開發時程及需求規格等重要項目的修訂上,使得每次變更就拉開規劃項目與軟體開發作業之間的距離,漸漸導致專案計畫完全無法有效控管預算執行、開發時程、產品品質及資源配置等軟體專案的運作,最後間接導致專案的失敗(Boehm, 1991)、(Boehm, 1989)、(Fairley, 1994)、(Weller, 1994)。為此,唯有以持續改善控制專案計畫的品質,方能有效降低軟體專案風險。

因應各種狀況提出的計畫變更案通過審核後,首先受到衝擊的就是已完成的開發文件與規劃項目,特別是成本、時程及需求規格等項目,這些文件與項目必須依據計畫變更的內容進行調整與修訂,越靠近交付移轉階段所提出之計畫變更案,對於整個開發作業的衝擊就越大,因為這個時段大部份的開發文件與規劃項目都已完成,至於無形效益的損失也很可觀,使用單位預期落空、開發團隊士氣低落、人員的流失及工作品質滑落等現象,為專案計畫帶來極高的風險。為了降低專案計畫的失敗風險,本文探究影響專案管理品質的

關鍵計畫，並剖析與蒐集專案管理關鍵計畫的主要品質因子，進而提出一套專案管理品質量測(Project Management Quality Measurement; *PMQM*)模式，及時標示出專案管理的品質問題與缺失，且以品質為基礎的持續改善程序(Quality-based Continuous Improvement Procedure; *QCIP*)，有效監控計畫變更後的專案管理，以高品質的專案管理降低軟體專案開發風險。本文第二節將說明畫變更的關鍵因素以及計畫變更對軟體專案的衝擊。第三節將提出一套計畫變更管理作業程序，且深入探討影響管理作業的品質因子。蒐集與量化變更作業品質因子是評估與改善作業品質的重要依據，第四節將針對變更作業品質因子蒐集與量化方式進行具體的描繪。第五節將提出一套變更作業品質量測模式，及時標示出變更作業的品質問題與缺失，適時矯正缺失以改善變更作業品質。第六節將彙整變更作業品質量測模式在降低失敗風險之優勢，且針對本主題作個結論。

貳、軟體專案管理與失敗因素之探討

頻繁計畫變更造成的衝擊是導致專案失敗的關鍵，本節將說明計畫變更提案的關鍵因素以及計畫變更對軟體專案的衝擊。

一、軟體專案管理

專案管理品質是軟體計畫成敗的主要關鍵，專案管理結合了多項的管理子計畫，其中以成本管理、時程管理及需求管理等三項子計畫最具指標性，從 Standish Group 所提出的研究報告中，可以發現成本超出預算、時程嚴重落後及產品不能符合使用者需求等狀況是軟體專案無法成功的三個主因。因此，為了具體降低軟體專案的失敗風險，勢必要有效管控且強化成本管理、時程管理及需求品質管理等子計畫品質。此外軟體開發過程中，使用單位對於軟體需求忽左忽右，無法描繪清楚且明確的方向，導致專案開發過程中，針對已通過審核的需求規格提出新增、刪除或修改等要求。在計畫變更案中，需求變更占的比率最高，不僅導致成本增加、時程延後及品質難以掌控，甚至經常成為軟體專案失敗的關鍵(賴森堂, 2012)。計畫變更也是專案管理必須面臨的一項重要考驗，能夠克服各式各樣計畫變更衝擊，才能有效降低軟體專案的失敗風險。以下針對三項專案管理子計畫為應付計畫變更的衝擊，應具備的管理與維護能力說明如下：

- 成本管理：當需求規格通過審查後，成本管理必須依據已制定的一套成本管理政策、程序及相關技術，於專案執行過程中，負責成本預估、成本報告與成本控制等相關作業，其主要的目的就是有效的掌控專案計畫成本。成本管理必須有能力完成下面三項任務：
 - 成本管理產出的各項文件都須具備正確、完整且一致的內容
 - 充分蒐集專案成本的歷史數據，評斷估算方法的優劣，採取適當的成本估算
 - 為應付計畫變更的衝擊，成本管理應建立一套成本重估或調整的機制
- 時程管理：當需求規格通過審查且專案成本完成初步估算後，時程管理可以配合工

作分解結構 (Work Breakdown Structure; WBS) 所細分的子專案、任務及子工作項 (Tausworthe, 1980)，配置且安排各項作業的工作時程，其主要的目的就是有效的掌控專案的工作時程。時程管理必須有能力完成下面三項任務：

- 時程管理產出的各項文件都須具備正確、完整且一致的內容
 - 時程管理應配合 WBS, CPM 及 Pert 等工具或技術，妥善規劃各項工作時程
 - 為應付計畫變更的衝擊，時程管理應建立一套工作時程重估或調整的機制
- 需求品質管理：開發過程中，開發人員的疏失是難以排除的，各階段完成前的檢視或審查活動若不能找出產品隱含的問題與缺失，將導致不完整、不一致或不正確的錯誤被注入各個階段的產品文件中，這些錯誤隨著開發作業遞延至後續階段且持續擴大(Pressman, 2010)，直到問題被發現時，已經不是局部修改就可以改善的，而是必須透過計畫變更提案。此外使用單位對於軟體需求無法描繪清楚且提供明確的方向，導致專案開發過程中，針對已通過審核的需求規格提出新增、刪除或修改等變更要求，各種計畫變更都需投入額外的人力、時間及成本，才能使產品品質符合使用者的需求。因此，品質管理必須有能力完成下面三項任務：
 - 需求品質管理產出的各項文件都須具備正確、完整且一致的內容
 - 需求必須透過完善的品質活動來掌控品質，品質管理應建制一套品質制度
 - 為應付計畫變更的衝擊，需求品質管理應配合開發作業的建構管理機制

二、計畫變更對軟體專案的影響

軟體專案失敗的比率高達 70%，卻無法有效且具體的改善，其主要根源便是計畫變更的次數過於頻繁，任何一次計畫變更都會增加軟體專案的開發風險，而多次的計畫變更勢必造成成本不斷追加、時程嚴重落後、產品品質難以有效掌握等狀況，最後成為高風險專案(賴森堂，2004)、(賴森堂，2005)、(賴森堂，2012)、(Hayes, 2004)。軟體開發的前置階段，必須針對專案計畫的開發成本進行仔細估算，SPMP 藉由完整的需求分析以及精確的成本估算，才能具體的規劃出軟體專案需投入的經費、時程與人力資源，同時擬訂出多項的規劃文件。需求分析完成後，各階段依需求規格書以及階段性任務陸續完成開發文件且通過審查。不過，在軟體開發過程中，因應各種狀況經常無預期提出的計畫變更案，一旦變更案通過變更審查委員會(CCB)的審核，後續的變更作業勢必對整個計畫造成影響。完成的開發文件、擬訂的規劃項目以及估算出的開發成本與計畫時程都要進行修訂與調整，除了須投入額外的人力與時間外，正在執行的工作與活動也會受到阻礙，因而導致成本追加、時程後延、品質不易掌握的軟體開發風險，如果專案計畫與開發製作無法適時的提出矯正或改善，勢必引導軟體專案走入失敗的困境。以下從成本、時程與需求品質等三項關鍵因素，說明計畫變更的衝擊(參閱圖 1 所示)：

- 成本的衝擊：已經估算且通過審議的開發成本，必須重新估算，以擬定出額外投入的人力成本，協助進行開發文件與規劃項目的修改與調整，因此開發成本勢必要追

加。

- 時程的衝擊：已經規劃妥善的工作時程，必須重新調整，以擬定出額外的時間，協助進行開發文件與規劃項目的修改與調整，因此勢必造成開發時程延後。
- 需求品質的衝擊：已完成審查的開發文件與規劃項目進行修改與調整，經常受到人力不足與時程緊迫的因素，出現不正確、不完整、不一致的修改缺失，這些缺失對於軟體後續開發的品質將造成非常不利影響。

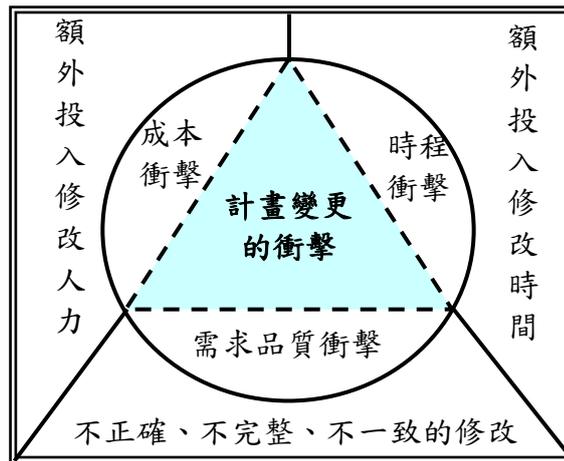


圖 1. 計畫變更案衝擊成本、時程與需求等關鍵項目的示意圖

參、專案管理項目與品質因子

預算、時間與規格是影響軟體專案成敗的主因。本節將探討專案成本、時間與需求管理的作業細節，且深入剖析影響三項管理作業的品質因子。

一、關鍵專案管理

專案管理結合了多項的管理子計畫，其中以成本管理、時程管理及需求管理等三項子計畫最具指標性，對於不同管理子計畫的品質因子應採取不同的蒐集方式，以下列出三項管理子計畫的關鍵品質因子：

(1) 成本管理的關鍵品質可以從成本管理計畫相關文件品質、成本估算品質及成本計畫更動品質等三方面進行探討：

- 成本管理文件品質：成本管理必須擬訂策略、程序且完成計畫書與相關報告，這些文件是其他管理項目必須參考且引用的資訊，因此成本管理文件必須具備以下品質因子：
 - 成本文件應採取統一的撰寫格式、

- 成本文件應具備正確性、完整性及一致性等基本品質
 - 成本管理估算品質：軟體專案成本估算作業應具備以下品質因子：
 - 應採取一套成本估算的標準作業流程(SOP)、
 - 應蒐集且保存相關專案的成本估算方式、
 - 應使用具有互補特質的軟體成本估算模式
 - 成本計畫更動品質：能夠配合計畫變更的成本管理應具備以下品質因子：
 - 應該具有高度的調整彈性、
 - 應建立一套與其他管理子計畫之間的交互影響關係表、
 - 對於成本變更前後的計畫內容，應完整記錄變更原因、變更項目、變更時間與負責人員等資料。
- (2) 時程管理的關鍵品質可以從時程管理計畫相關文件品質、時程調整品質及時程計畫更動品質等三方面進行探討：
- 時程管理文件品質：時程管理必須依據 WBS、CPM、Pert 等工具或技術，完成計畫書與相關報告，這些文件是其他管理項目必須參考且引用的資訊，因此時程管理文件必須具備以下品質因子：
 - 時程文件應採取統一的撰寫格式、
 - 時程文件應具備正確性、完整性及一致性等基本品質
 - 時程規劃品質：軟體專案時程規劃作業應具備以下品質因子：
 - 應採取一套時程規劃的標準作業流程、
 - WBS 應具備可定義的、可管理的、可預估的、獨立的、可整合的、可量測的及可安置的等多項特質、
 - 應多加利用現有的技術與軟體輔助工具協助時程規劃作業
 - 時程計畫更動品質：能夠配合計畫變更的時程管理應具備以下品質因子：
 - 應該具有高度的調整彈性、
 - 應建立一套與其他管理子計畫之間的交互影響關係表、
 - 對於時程變更前後的計畫內容，應完整記錄變更原因、變更項目、變更時間與負責人員等資料。

(3) 需求品質管理的關鍵品質可以從軟體需求相關文件品質、需求項目品質及需求項目更動品質等三方面進行探討：

- 軟體需求文件品質：軟體需求必須依據需求擷取過程中，取得的重要需求資訊進行分析，完成需求規格書與品質計畫書，這些文件是其他管理項目必須參考且引用的資訊，因此軟體需求與品質文件必須具備以下品質因子：
 - － 需求文件應採取統一的撰寫格式、
 - － 需求文件應具備正確性、完整性及一致性等基本品質
- 需求項目確認品質：軟體專案時程規劃作業應具備以下品質因子：
 - － 應採取一套需求項目確認品質的標準作業流程、
 - － 需求項目應具備可定義的、可管理的、可驗證的、獨立的、可整合的、及可安置的等多項特質。
- 需求計畫更動品質：能夠配合計畫變更的品質管理應具備以下品質因子：
 - － 應該具有高度的調整彈性、
 - － 應建立一套與其他管理子計畫之間的交互影響關係表、
 - － 對於需求變更前後的計畫內容，應完整記錄變更原因、變更項目、變更時間與負責人員等資料。

二、專案管理品質矯正方式

次數過於頻繁的計畫變更案經常帶來高度的開發風險，若不能擬定一套持續矯正方式與步驟，最後將造成軟體專案淪為失敗計畫。為了降低計畫變更帶來的高風險，本文規劃一套以品質為基礎的持續改善程序，且以專案管理品質量測模式為依據，配合法則式推論，協助標示關鍵管理品質問題與缺失，且適時提出矯正建議與改善措施，有效管控專案管理的品質，降低計畫變更所造成的軟體開發風險。量測指標是一種相對的標準，當指標、量測值或量度小於 0.6 可以視為「品質過低」或是「不可接受範圍」，不過也可以隨著環境、領域或需求等情況適度調整，以下為七項法則式推論與建議措施：專案管理品質因子的蒐集方式

專案管理品質因子隱含在相關管理作業的子工作項中，如何有效且具體的蒐集各個子工作項的品質因子，是專案管理品質量化評估的重要依據。以下從六個方面說明品質因子的蒐集方式：

- 管理文件基本品質因子：可以蒐集各項管理文件繳交前的審查活動記錄，以及管理文件的撰寫指引、程序或審查的查核表單等。
- 成本估算品質因子：可以透過成本管理計畫書所擬定的成本估算模式進行剖析，同時蒐集成本估算過程中，是否已充分參考或引用成功專案的歷史數據。
- 時程規劃品質因子：可以透過時程管理計畫書所規劃的工作時程內容進行剖析，對於時程規劃過程中，可蒐集是否配合 WBS, CPM, Pert 及甘特圖等工具或技術進行工作時程的制定。
- 需求確認品質因子：可以檢視需求確認活動的記錄、需求確認查核表的內容是否涵蓋重要的需求項目特質、是否採取標準的需求確認程序。
- 計畫更動品質因子：可以檢查文件變更申請單、審查活動、文件版本記錄等相關資料，協助符合度的量化蒐集。
- 未涵蓋的項目：可以依 3.1 節所描繪的工作項目品質因子，進行符合度的量化蒐集。

三、專案管理品質因子的正規化

個別或低階的品質因子無法具體表現其影響力，必須將個別或低階的品質因子適度的結合才展現變更作業品質因子的影響力，因此高階且具體的品質項目量測值都是由一些低階的品質因子結合而成(Boehm, 1981)、(Conte etc. 1986)、(Lai, 2000)、(Lai and Yang, 1998)，其中每一個受衝擊項目的評分就相當於一低階品質因子量度值，這些低階量度值依其影響度及發生率，又可設定成不同的參數或權位值，接著再配合特定的結合公式就可計算出高階品質項目的量測值。利用各種方式所蒐集到的品質因子量度值，都可以依同樣的方式計算出特定衝擊項目的量測值，因為個別的量度只能評量某項衝擊的單項特質，為了量測整體變更作業的品質指標，必須將個別項目的衝擊度做適當的結合。量度結合的方式可以分為線性結合(Linear Combination) (Boehm, 1981)、(Fenton, 1991)、(Lai, 2000)、(Lai and Yang, 1998) 與非線性結合(Nonlinear Combination) (Boehm, 1981)、(Conte etc. 1986)、(Fenton, 1991)，其中線性結合的特質包括：思維型態較容易被接受、公式化的步驟較單純、具備較高的調整彈性及適合資料蒐集量較少之應用等。非線性結合的特質包括：能夠較精確的表現量度結合的效果、公式化的步驟較複雜且需要工具的支援、需要資深的統計技術人員的協助及適合資料蒐集量較少之應用等，線性結合與非線性結合主要特質比較內容請參閱表 1。考量實用性、修改彈性與簡單性，本文以線性結合方式來建立計畫變更影響因子的結合步驟。不同層面的影響因子有不同的量度表達方式，因此在進行線性結合之前，必須將相關的品質因子量度值正規化(Normalization)，正規化的作業必須考量品質因子的發生機率、影響程度及蒐集到的量化值，優質的品質因子量度值正規化後趨近於 1，劣質的品質因子量度值正規化後趨近於 0，再配合事先設定的權位值及結合公式可以將低階且具高

度相關性的量化品質因子結合成基層及高階特性項目的量測值。

表 1. 線性結合與非線性結合主要特質比較表

特質 \ 量度結合模式	線性結合	非線性結合
精確度	中等	高
公式化	步驟單純	步驟複雜
調整彈性	高	低
配合人員	資深軟體開發人員	資深統計技術人員
資料蒐集量	少	多

肆、專案管理品質量測模式與矯正方式

本節將提出一套變更管理作業品質量測模式，在專案管理作業中，適時標示品質問題與缺失，以具體降低品質缺失衍生出軟體專案失敗危機。

一、專案管理品質量測模式

專案管理品質量測指標可以由成本管理、時程管理與需求品質管理等三項關鍵品質量測值結合而成，而各項關鍵品質量測值則由一些子工作項的基層量度值結合而成，透過線性結合公式，可以將高度相關性的基層品質因子結合成基層量度，這些基層量度可以進一步結合成高階品質量測值，最後再將管理品質量測值加以結合，成為專案管理品質量化指標，以下分四個步驟說明量化指標的產生過程：

- (1) 成本品質測值(Cost Management Quality Measurement; CMQM) 先由影響成本管理文件、成本重估及成本計畫更動等子工作項品質的量化因子結合成三項基層量度，再將成本管理文件、成本重估及成本計畫更動等三項基層量度結合成成本管理品質量測值，結合公式如公式(1)所示：

CMQM: Cost Management Quality Measurement

CPDM: Cost Management Document Metric

W_{cm1} : Weight of CMDM

CREM: Cost R-Estimation Metric

W_{cm2} : Weight of CREM

CPCM: Cost Plan Change Metric

W_{cm3} : Weight of CPCM

$$CMQM = W_{cm1} * CMDM + W_{cm2} * CREM + W_{cm3} * CPCM$$

$$W_{cm1} + W_{cm2} + W_{cm3} = 1 \quad (1)$$

- (2) 時程管理品質測值(Time Management Quality Measurement; TMQM) 先由影響時程管理文件、時程調整及時程計畫更動等子工作項品質的量化因子結合成三項基層量度，再將時程管理文件、時程調整及時程計畫更動等三項基層量度結合成時程管理品質量測值，結合公式如公式(2)所示：

TMQM: Time Management Quality Measurement

TMDM: Time Management Document Metric W_{tm1} : Weight of TMDM

TAM: Time Adjustment Metric W_{tm2} : Weight of TAM

TPCM: Time Plan Change Metric W_{tm3} : Weight of TPCM

$$TMQM = W_{tm1} * TMDM + W_{tm2} * TAM + W_{tm3} * TPCM$$

$$W_{tm1} + W_{tm2} + W_{tm3} = 1 \quad (2)$$

- (3) 需求管理品質測值(Requirement Management Quality Measurement; RMQM) 先由影響需求規格文件、需求確認及需求規格更動等子工作項品質的量化因子結合成三項基層量度，再將需求規格文件、需求確認及需求規格更動等三項基層量度結合成需求管理品質量測值，結合公式如公式(3)所示：

RMQM: Requirement Management Quality Measurement

RSDM: Requirement Spec. Document Metric W_{rm1} : Weight of RSDM

RAM: Requirement Assurance Metric W_{rm2} : Weight of RAM

RSCM: Requirement Spec. Change Metric W_{rm3} : Weight of RSCM

$$RMQM = W_{rm1} * RSDM + W_{rm2} * RAM + W_{rm3} * RSCM$$

$$W_{rm1} + W_{rm2} + W_{rm3} = 1 \quad (3)$$

- (4) 最後透過公式(4)，將成本管理、時程管理與需求管理等三項關鍵作業品質量測值結合成專案管理品質量化指標(Quality Measurement Indicator for Project Management; QMIPM)，結合公式如公式(4)所示：

QMIPM: Quality Measurement Indicator for Project Management

CMQM: Cost Management Quality Measurement W_{cmqm} : Weight of CMQM

TMQM: Time Management Quality Measurement W_{tmqm} : Weight of TMQM

RMQM: Requirement Management Quality Measurement W_{rmqm} : Weight of RMQM

$$QMIPM = W_{cmqm} * CMQM + W_{tmqm} * TMQM + W_{rmqm} * RMQM$$

$$W_{cmqm} + W_{tmqm} + W_{rmqm} = 1 \quad (4)$$

本品質量測模式一共匯集了 9 組基層專案管理子工作項之品質因子，且透過線性結合模式產生 9 項基層變更子工作項量度及三項高階管理作業量測值，經過三個層次的量度結合，最後產生專案管理品質量測指標(QMIPM)，稱此量測模式為專案管理品質量測(PMQM)模式，其架構如圖 2 所示。

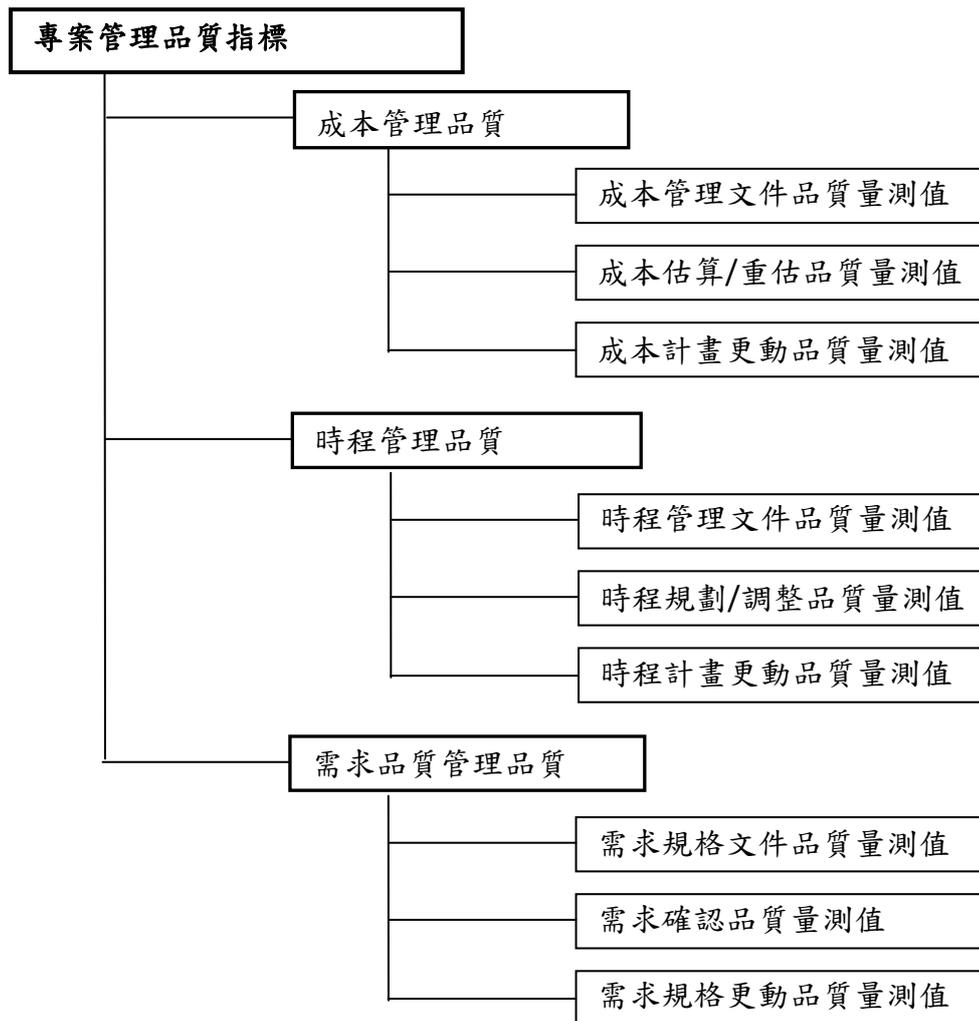


圖 2. 專案管理品質量測(PMQM)模式架構圖

二、專案管理品質矯正方式

次數過於頻繁的計畫變更案經常帶來高度的開發風險，若不能擬定一套持續矯正方式與步驟，最後將造成軟體專案淪為失敗計畫。為了降低計畫變更帶來的高風險，本文規劃一套以品質為基礎的持續改善程序，且以專案管理品質量測模式為依據，配合法則式推論，協助標示關鍵管理品質問題與缺失，且適時提出矯正建議與改善措施，有效管控專案管理

的品質，降低計畫變更所造成的軟體開發風險。量測指標是一種相對的標準，當指標、量測值或量度小於 0.6 可以視為「品質過低」或是「不可接受範圍」，不過也可以隨著環境、領域或需求等情況適度調整，以下為七項法則式推論與建議措施：

<Rule 1> IF 「專案管理品質量測指標」屬於「品質過低」的現象

THEN 針對「成本管理」、「時程管理」、與「需求品質管理」等量測值進行下一步分析。

<Rule 2> IF 「成本管理量測值」屬於「品質過低」的量測值

THEN 可以進一步分析三個成本管理工作項「品質量度」是否在可接受範圍。

<Rule 2-1> IF 成本管理工作項「品質量度」低於可接受範圍

THEN 建議措施 1：若屬於成本管理文件品質缺失，則找出對映的文件細項工作且進行矯正。

建議措施 2：若屬於成本估算品質缺失，則找出對映的估算細項工作且進行矯正。

建議措施 3：若屬於成本更動品質缺失，則找出對映的更動細項工作且進行矯正。

<Rule 3> IF 「時程管理量測值」屬於「品質過低」的量測值

THEN 可以進一步分析三個時程管理工作項「品質量度」是否在可接受範圍。

<Rule 3-1> IF 時程管理工作項「品質量度」低於可接受範圍

THEN 建議措施 1：若屬於時程管理文件品質缺失，則找出對映的文件細項工作且進行矯正。

建議措施 2：若屬於時程規劃品質缺失，則找出對映的規劃細項工作且進行矯正。

建議措施 3：若屬於時程更動品質缺失，則找出對映的更動細項工作且進行矯正。

<Rule 4> IF 「需求品質管理量測值」屬於「品質過低」的量測值

THEN 進一步分析三個需求品質管理工作項「品質量度」是否在可接受範圍。

<Rule 4-1> IF 需求品質管理工作項「品質量度」低於可接受範圍

THEN 建議措施 1：若屬於需求規格文件品質缺失，則找出對映的文件撰寫

細項工作且進行矯正。

建議措施 2：若屬於需求確認品質缺失，則找出對映的需求確認細項工作且進行矯正。

建議措施 3：若屬於需求規格更動品質缺失，則找出對映的需求規格更動細項工作且進行矯正。

伍、以品質為基礎的持續改善程序

專案管理的指標性管理計畫包括成本管理、時程管理及需求品質管理等三項關鍵計畫，因此當管理計畫完成擬定，或計畫變更通過且完成計畫變更相關作業後，必須針對成本管理、時程管理及需求品質管理等三項關鍵子計畫進行檢視以確實其品質。若品質未能符合規定的標準，就必須找出品質的問題或缺失，且回到前面的作業，對後影響的文件或規劃項目重新進行修訂與調整。此項品質檢視活動必須重複且持續，直到三項關鍵子計畫都通過品質檢視活動，此外一旦有計畫變更案通過審核，此項改善程序就必須被啟動，為此，本文稱此項程序為品質為基礎的持續改善程序(QCIP)，其流程如圖 3 所示。

品質為基礎的持續改善程序分成三個作業階段：

(1) 計畫擬定/變更階段：

- 新的軟體專案依使用單位的需求，擬定各項管理計畫，透過正式的審查或檢視活動確認管理計畫的內容，且針對確認活動找出的缺失進行修訂。
- 既存的軟體專案，則依計畫變更要求規劃各項管理計畫修訂作業，明確標示受影響的項目與修訂內容，且針對受影響的管理項目進行調整與修訂。

(2) 管理品質確認階段涵蓋兩個關鍵步驟：

- 蒐集關鍵管理計畫的品質因子，且依不同性質的品質因子進行量化。
- 依專案管理品質量測(PMQM)模式，產生專案管理品質量測指標(QMIPM)。

(3) 品質改善階段是改善程序的主要步驟：

- 透過法則式的專案管理品質矯正方式，標示出專案管理品質的問題與缺失。
- 確認品質的問題與缺失。
- 依品質缺失進行矯正與改善作業。

品質為基礎的持續改善程序將隨著專案管理的運作，持續被執行，直到軟體專案能夠達成預期的目標。

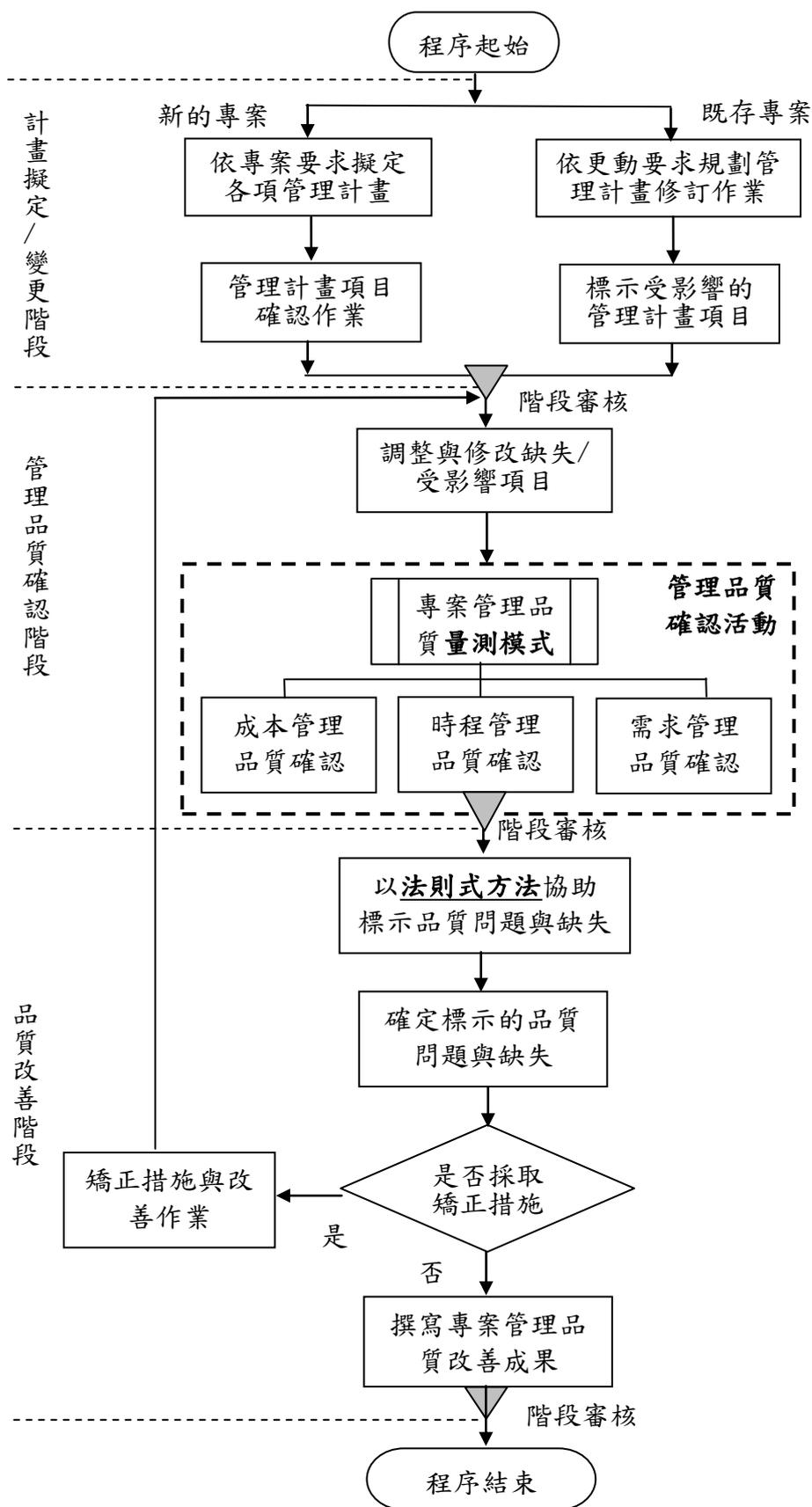


圖 3. 以品質為基礎的持續改善程序(QCIP)流程圖

陸、結論

軟體系統開發過程中，受到使用單位的變更需求、資源的變動、資訊技術與環境的演進以及開發人員的缺失等多樣化因素的影響，經常造成時程延後、預算追加、殘留許多錯誤或不能符合使用單位的要求等狀況，最後導致軟體開發計畫淪為失敗的專案。其中最關鍵的因素就是計畫變更過於頻繁，軟體專案開發過程中，計畫變更造成已規劃項目及完成開發文件必須配合修訂與調整，衍生軟體專案預算不斷追加、時程往後延遲、需求品質難以掌握等現象，最後更成為軟體專案失敗的關鍵。為了降低軟體專案淪為失敗計畫的風險，本文針對專案管理的品質缺失帶來的開發風險進行深入探討，且規劃一套品質持續改善程序，並剖析與蒐集影響專案管理作業成效的品質因子，進而提出一套專案管理品質量測(PMQM)模式，以 PMQM 模式協助找出的專案管理品質問題與缺失，配合法則式推論與建議措施，有效降低專案管理品質造成的軟體開發風險。本文以 PMQM 模式協助專案管理的持續改善作業，其優勢如下：

- (1) 以量測模式產生專案管理品質量測指標。
- (2) 以量測指標及量化機制協助標示專案管理作業的品質問題與缺失。
- (3) 配合法則式推論，協助專案管理作業的持續改善。

本文提出之 PMQM 模式具備三項特質：

- 簡化公式—以線性結合公式取代複雜的非線性結合公式。
- 易於標示問題與缺失—對照公式，可迅速標示出專案管理的品質問題與缺失。
- 提高調整彈性—可以隨著軟體專案應用領域的不同，彈性調整公式與項目。

誌謝

本論文接受實踐大學 101 學年度「校內補助專題研究計畫案」(計畫編號:101-05-04010)之補助。

參考文獻

- 賴森堂(2004)，「提高軟體專案成功率之品質風險量測模式」，第二屆資訊科技與人文管理教育論壇研討會論文集，中央警察大學。
- 賴森堂(2005)，「輔助軟體專案管理的多層次風險量測模式」，第一屆台灣軟體工程研討會論文集，中央大學。
- 賴森堂(2012)，「強化變更作業關鍵品質以降低軟體開發風險」，臺北科技大學學報第四十五之一期，32-48 頁，2012 年 8 月。
- Boehm, B. W. (1991), "Software risk management: Principles and practices," IEEE Software,, 8(1), pp. 32-41.

- Boehm, B. W. (1989), "Tutorial: Software Risk Management," IEEE CS Press, Los Alamitos, Calif.
- Boehm, B.W. (1981), "Software Engineering Economics," Prentice-Hall, New Jersey.
- Conte, S. D., H. E. Dunsmore, and V. Y. Shen (1986), "Software Engineering Metrics and Models," Benjamin/Cummings, Menlo Park.
- Fairley, R. (1994), "Risk management for Software Projects," IEEE Software, vol. 11, no. 3 , pp. 57-67.
- Fairley, R. (1985), "Software Engineering Concepts," McGraw-Hill, Inc.
- Fenton, N.E. (1991) , "Software Metrics - A Rigorous Approach," Chapman & Hall.
- Hayes F., (2004) "Chaos is Back," Computerworld, Normember.
- Galin, Daniel (2004), "Software Quality Assurance – From theory to implementation, " Pearson Education Limited, England.
- Lai, S. T. (2000), "A Quality Measurement Model for Software Maintenance," Proceeding of the Second World Congress on Software Quality (2WCSQ), Japan, pp. 331-336.
- Lai, S. T. and C. C. Yang (1998), "A Software Metric Combination Model for Software Reuse," Proceedings of 1998 Asia-Pacific Software Engineering Conference, Taiwan, Taipei, pp. 70-77.
- Low G. C. and D. R. Jeffrey (1990), "Function Points in Estimation and Evaluation of the Software Process", IEEE Trans. On Software Engineering, vol. 16, no. 1, pp.64-71.
- Pressman, R. S. (2010), "Software Engineering: A Practitioner's Approach," McGraw-Hill, New York.
- Schach, S. R. (2008), "Object-Oriented Software Engineering," McGraw-Hill Companies.
- Tausworthe, Robert C.(1980), "The Work Breakdown Structure in Software Project Management, the journal System and software, no 1., pp. 181-186, 1980.
- Weller, E.F. (1994), "Using Metrics to Manage Software Projects," IEEE Computer, 27(9), pp.27-24.