

第十三章

軟體專案度量與評估

執行單位：國立臺灣科技大學
軟體工程學程中心



大綱

- 度量值的類型與特質
- 軟體產品度量值
- 軟體流程度量值
- 軟體產品評估
- 軟體流程評估

學習目標

透過本章節希望學生能具備軟體度量的概念與軟體流程與產品評估相關的知識，例如：

- 瞭解度量值的類型與需具備哪些良好的特性
- 瞭解軟體專案管理相關的產品度量值
- 瞭解軟體專案管理相關的流程度量值
- 瞭解軟體專案管理相關的軟體產品評估
- 瞭解軟體專案管理相關的軟體流程評估

度量值的類型與特質



基礎度量值和衍生度量值 (1/3)

- 度量值 (Measures) 可分為基礎度量值 (Base measures) 與衍生度量值 (Derived measures) 兩類。
- 基礎度量值來自於直接度量即可獲得，衍生度量值則需要結合多個基礎度量值所衍生出來，而且無法透過直接度量獲得。
- 衍生度量值推導自基礎度量值，通常比基礎度量值所推論出的資訊更為豐富。

基礎度量值和衍生度量值 (2/3)

- 一般使用的基礎度量值，包括下列例子：
 - 產品大小的估計值或實際度量值（例如程式碼行數, LOC）
 - 人力與成本的估計值或實際度量值（例如：團隊人數、專案開始至結束的時間）
 - 品質度量（例如：缺失數）

基礎度量值和衍生度量值 (3/3)

- 一般使用的衍生度量值，包括下列例子：
 - 工程師生產力 (Programmer productivity)：工程師撰寫的程式碼行數 / 工程師所花費的工時
 - 模組的缺失密度 (Module defect density)：該模組內發現的缺失數 / 模組的大小
 - 需求穩定性 (Requirements stability)：專案開始時的需求數 / 專案結束時所完成交付的總需求數
 - 可靠度 (Reliability)：平均失效時間 (Mean Time Between Failure, MTBF)

好的軟體度量指標應具備的特性 (1/2)

■ 簡單和易用

- 容易學習，在計算過程中不必花費大量的時間與成本

■ 一致和客觀

- 一致性且結果不能模擬兩可

■ 具說服力

- 使用者對該度量指標的正確性、有效性與實用性等，在直覺的判斷上是沒有問題的，並具說服力

好的軟體度量指標應具備的特性 (2/2)

- 信度 (Reliability)
 - 準確度 (Accuracy)
 - 精密度 (Precision)
- 效度 (Validity) (或一致性)：度量的穩定性，亦即在不同得的情況下都能維持度量的信度 (準確度與精密度)
- 實用性 (Practically)：度量的經濟性 (成本效益分析)、便利性 (度量值容易獲得或可用軟體系統或工具取得)、易瞭解性 (度量值是有意義的，並且在直覺上是很容易瞭解與具說服力)、解釋能力 (可以依據度量值來採取後續動作) 等
- 適用範圍的考量 (一般性與特定性)

客觀和主觀的度量值

- 確保基本度量值與衍生度量值資料的收集與計算，不會因人而異
- 大部分工程或人文社會領域使用度量指標值的判斷與計算都會有一些人為主觀的判斷，但可經由標準作業程序的制訂來減少人為主觀判斷所造成的差異
- 主觀的指標是不可避免，只必須清楚該物度量指標適用的範圍與其精確性的程度

軟體產品度量值



軟體功能性大小度量指標

- 軟體功能性大小度量方法是以使用者觀點將軟體系統切割成數個元件以便評估該軟體系統的大小，優點是獨立於程式語言或程式師習慣，並且能夠在軟體開發早期便能求得，更可以搭配其他模式或度量值來進行標準化的分析與應用
- 目前有四個軟體功能性大小度量方法的國際標準，分別為ISO /IEC 19761（COSMIC完全功能點）、ISO /IEC 20968（MKII功能點分析）、ISO/IEC 24570（NESMA）與ISO /IEC 20926（原功能點分析但不含14個調整因子的計算）

軟體產品品質度量指標

- 可維護性(Maintainability)
- 效率(Efficiency)
- 易用性(Usability)
- 可靠性(Reliability)
- 客戶滿意度(User Satisfaction)

可維護性 (1/2)

■ 修復時間 (Time to Restore)

- 從軟體產品執行失敗至立即恢復，並找到失敗原因進而修改完成所必需花費時間的度量值
 - 軟體產品功能修復所花費的時間
 - 功能被修復的時間記錄-軟體產品失敗的時間記錄
 - 單位功能點或程式碼行數所需花費修復軟體產品的工作量之指標
 - 修復軟體產品所花費的工作量/軟體的功能點數或原始程式碼行數

可維護性 (2/2)

■ 語法複雜性 (Cyclomatic Complexity)

- 計算軟體元件中獨立路徑的個數，用來評估系統的資料流或控制流複雜度的度量值，倘若軟體元件擁有高的複雜度，其該元件則可能會有較高的缺失率，所以需要增加額外的檢查或測試。
 - 決策點的數目 (nodes)
 - 線性獨立路徑的數目 (edges)

效率 (1/2)

■ 產出能力 (Throughput)

● 於特定情況下執行所需功能時，軟體產品提供可接受的預估產出能力或實際產出能力的度量值。

■ 單位功能點或程式碼行數處理交易次數之指標

■ 交易處理次數/軟體的功能點數或原始程式碼行數

■ 單位功能點或程式碼行數完成交易處理次數之指標

■ 交易處理完成次數/軟體的功能點數或原始程式碼行數

■ 單位功能點或程式碼行數成功傳遞資料包裹數目之指標

■ 傳遞成功的資料包裹數目/軟體的功能點數或原始程式碼行數

效率 (2/2)

■ 時機 (Timing)

- 於特定情況下執行所需功能（例：需求請求的處理）時，軟體產品提供可接受的反應處理時間，該度量值是在功能執行的起啟時間和功能執行完成時間之間的度量值。
 - 執行功能的反應處理時間
 - 功能執行完成的時間-功能執行的起啟時間
 - 單位功能點或程式碼行數所需花費執行處理時間之指標
 - 反應處理時間/軟體的功能點數或原始程式碼行數

易用性

■ 操作性錯誤

- 於特定使用情況下，軟體產品可被理解、學習、使用的能力的度量值。

- 操作錯誤的次數/軟體的功能點數或原始程式碼行數

可靠性

■ 失敗(Failure)

● 於特定使用情況下，軟體發生失敗的次數、時間的度量值。

■ 軟體發生失敗的間隔

■ 此次軟體發生失敗的日期時間-上一次軟體發生失敗的日期時間

■ 軟體發生失敗次數/軟體的功能點數或原始程式碼行數

客戶滿意度量指標

■ 客戶支援(Customer Support)

- 技術支援服務的執行效能 (Help Desk calls density)
- 技術支援服務的有效性 (success of the Help Desk services)
- 技術支援服務的生產力 (Help Desk productivity)
- 技術支援服務的效率 (Help Desk effectiveness)
- 修護生產力 (Corrective Maintenance Productivity)
- 修護效率 (Corrective Maintenance Effectiveness)

客戶支援 (1/3)

■ 技術支援服務的執行效能

● 單位功能點或程式碼之技術支援服務的次數的指標

- 求助技術支援服務的總次數/軟體的功能點數或原始程式碼數

■ 技術支援服務的有效性

● 成功求助技術支援服務的比率

- 求助技術支援服務所完成的次數/求助技術支援服務的總次數

客戶支援 (2/3)

■ 技術支援服務的生產力

- 單位功能點或程式碼所需花費求助技術支援服務時間的指標
 - 求助技術支援服務所花費時間/軟體的功能點數或原始程式碼數

■ 技術支援服務的效率

- 回應單位技術支援服務的呼叫所需花費時間的指標
 - 求助技術支援服務所花費時間/求助技術支援服務的總次數

客戶支援 (3/3)

■ 修護生產力

- 單位功能點或程式碼所需花費修復錯誤時間的指標
 - 修復錯誤所花費時間/軟體的功能點數或原始程式碼數

■ 修護效率

- 平均修復一個錯誤所需花費時間的指標
 - 修復錯誤所花費時間/發現錯誤總數

ISO/IEC 9126 (1/2)

- ISO/IEC 9126定義了一個通用的軟體品質模式，讓軟體品質評估的組織能夠客觀且量化的規劃品質需求及評估軟體產品品質
- 評估軟體產品品質時，此品質模式不只用來規劃品質需求，並提供了內、外部及使用品質度量指標來協助整個軟體生命週期各階段之產品的評估，以確保最終軟體產品的品質

ISO/IEC 9126 (2/2)

- ISO/IEC 9126標準共有四個單元：
 - 9126-1 品質模式 (Quality Model)：
定義軟體產品品質模式及品質特徵與子特徵。
 - 9126-2 外部度量指標 (External Metrics)：
定義每個子特徵的外部度量指標
 - 9126-3 內部度量指標 (Internal Metrics)：
定義每個子特徵的內部度量指標
 - 9126-4 使用品質度量指標 (Quality in Use Metrics)：定義使用品質度量指標

軟體流程度量值



軟體流程度量值

- 時程與進度
- 資源與成本
- 錯誤與缺失

時程與進度指標

■ 里程碑日期(Milestone Dates)

- 指專案進行中，部分應完成的子活動，包含開始和結束日期。
 - 完成里程碑日期比例
 - 延遲里程碑日期比例
 - 專案時程變動率

■ 工作進度(Work Unit Progress)

- 專案進行中，各工作進度，主要以工作為單元，說明工作進行狀況。

時程與進度指標

■ 生產力(Productivity)

- 既定限定資源、人力、成本(Input)，所產出產能(Output)
 - 產出/直接人員
 - 產出/全體開發人員
 - 產出/(全體開發人員+支援人員)

■ 交付率(Delivery Rate)

- 在時程下所輸出比例，軟體從需求分析至交到客戶手上使用，交付程度。
- 功能點數(FP)/專案進行時間(Elapsed Time)
 - 開發交付率(Development Time-to-delivery Rate)
 - 增強交付率(Enhancement Time-to-delivery Rate)

資源與成本指標

■ 成本(Cost)

- 投入資源的花費，包含實質上的成本，隱藏的成本。
 - 員工成本(Labor Cost)
 - 專案成本(Project Cost)
 - 轉換成本(Conversion Cost)
 - 裝置成本(Installation Cost)
 - 訓練成本

■ 工作量(Effort)

- 員工工作程度，指工作完成所需投入的量值。
- 人月(Personnel Month)或人小時(Man Hours)
 - 專案工作量變動率

資源與成本指標

■ 人力資源(Staff Resource)

● 人員各項為公司製造生產力

- 人員能力 (Ability of Employee)
- 人員績效 (Achievement of Employee)
- 人員負擔 (Application Maintenance Load per person)
- 所需人力 (Need of Employee)
- 人力訓練百分比 (Training Ratio)
- 人員工作變動量

錯誤與缺失指標

■ 錯誤與缺失(Error and Defect)

- 產品異常(anomaly)，在生命週期早期發現任何遺漏(omissions)不完美(imperfections)，或軟體在趨近於完成階段所發生的測試或操作錯誤(faults)
- 導致功能單位產生意外狀況或在軟體中表現錯誤(fault)狀態
 - 缺失移除率
 - 缺失未移除率
 - 操作性錯誤率
 - 缺失率
 - 平均錯誤發生時間(Mean-time-to-Failure,MTTF)
 - 平均錯誤修復時間

錯誤與缺失指標

■ 測試(Testing)

- 系統或元件在特定環境下執行的活動，結果或觀察值被記錄評估。
- 測試案例(test case)和測試程序(procedure)的集合
 - 測試次數
 - 需求涵蓋比
 - 通過測試比率
 - 未通過測試比率
 - 測試涵蓋率

軟體產品評估

東

南

大

學



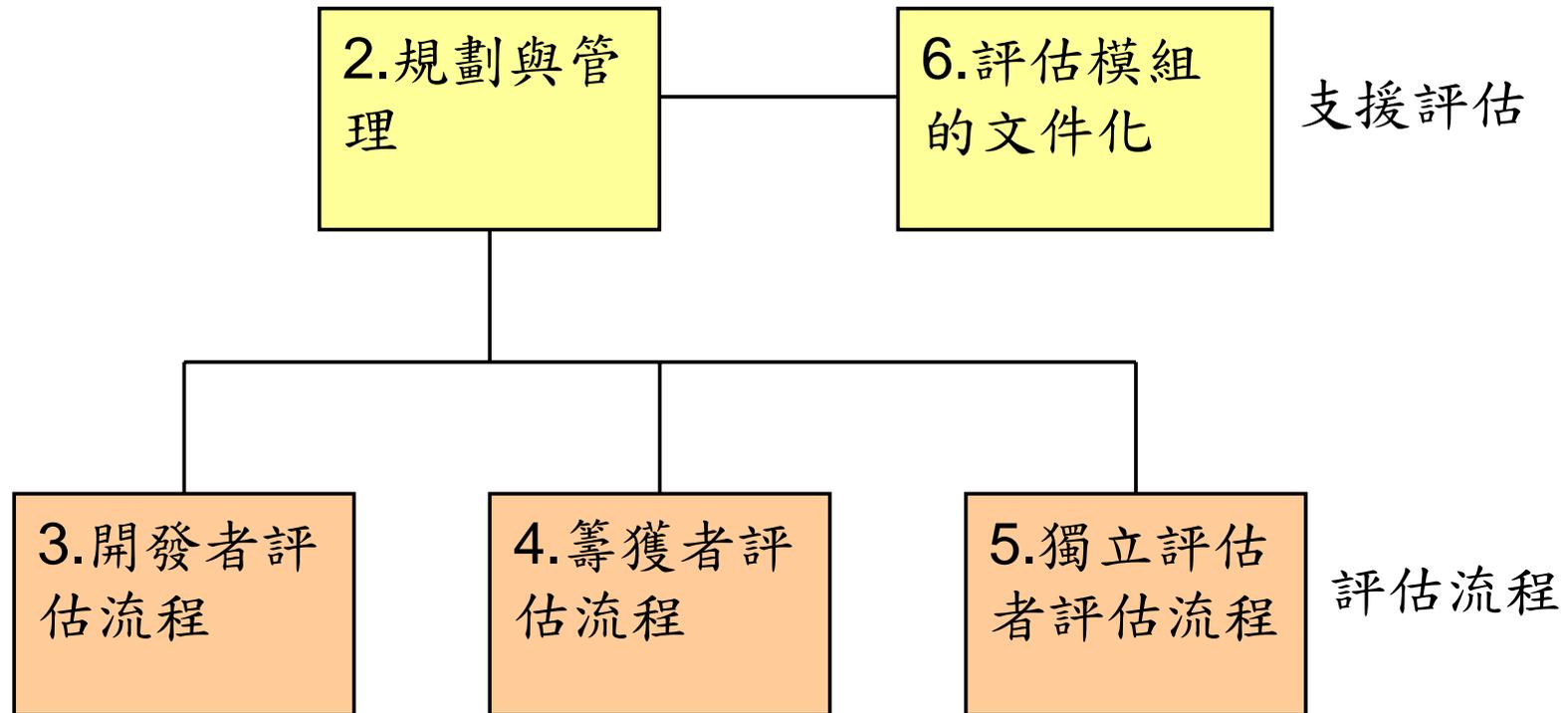
ISO/IEC 14598 (1/2)

- ISO/IEC 14598 Software Product Evaluation 是一套用來評估軟體產品品質的國際標準，它主要是提出對軟體產品品質度量及評估的方法，著重在產品的度量方面，而不是用來度量及評估軟體開發流程或是預估軟體開發成本
- 使用ISO/IEC 14598產生的評估結果能讓管理者、開發者或者是維護者了解軟體產品符合使用者需求的程度，以及作為品質改善的依據

ISO/IEC 14598 (2/2)

- ISO/IEC 14598系列中有六個單元，其中14598-1是定義評估的概念及架構，它提供了一個可以用於各類軟體產品評估架構及說明欲度量、評估一個軟體產品時的基本需求
- ISO/IEC 14598的評估作業主要是提供給發展者、籌獲者、獨立評估者所使用，不同的角色會有不同的評估用途。例如：發展者可以使用軟體產品評估來判斷軟體是否符合使用者需求，籌獲者則是使用軟體產品評估來比較同類型的軟體產品以決定是否購買

系列中各單元的關係 (1/3)



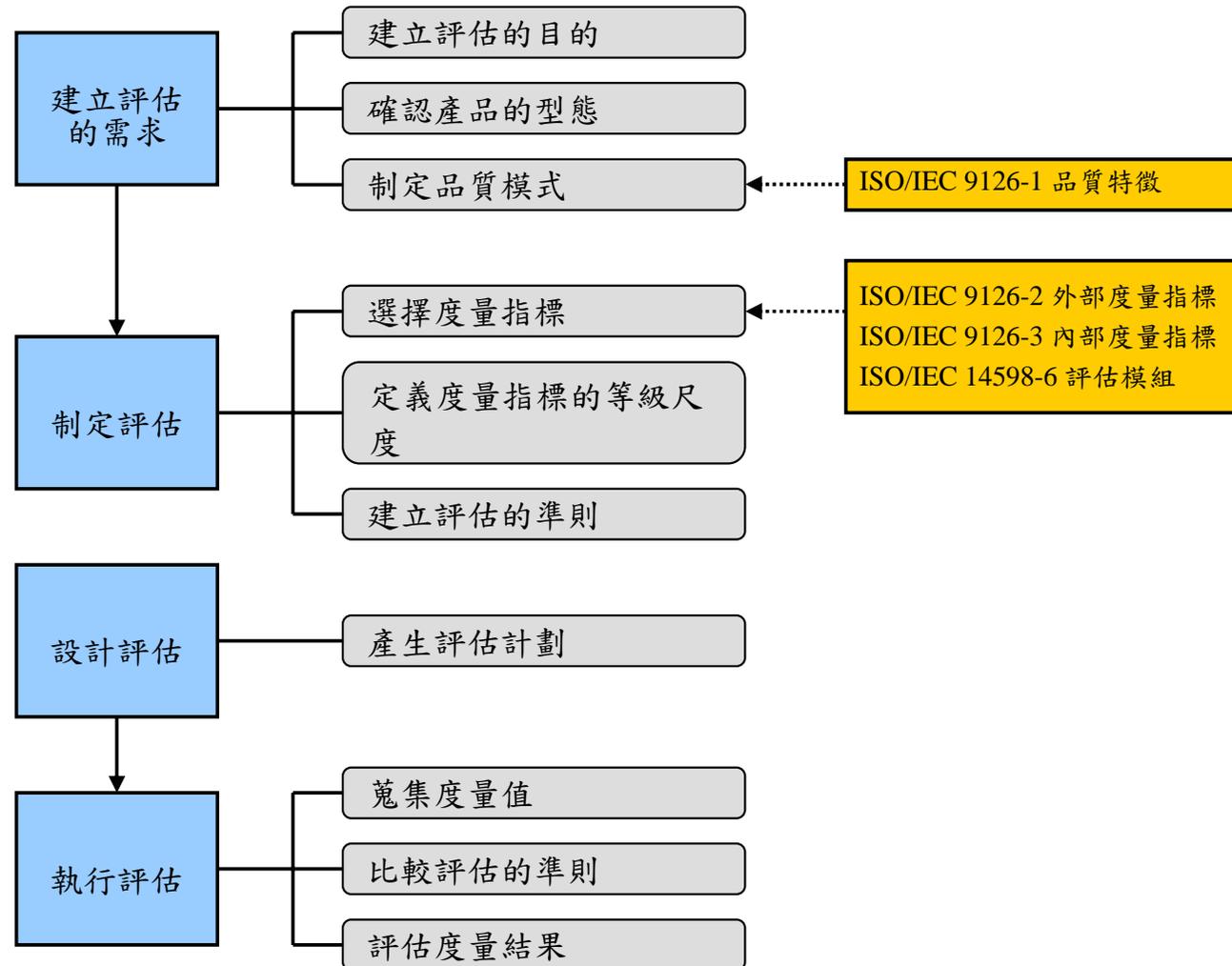
系列中各單元的關係 (2/3)

- ISO/IEC 14598提供了三種角色的評估流程（14598-3、14598-4、14598-5），如圖所示，這三個評估流程都必須跟14598-2及14598-6結合應用。
- 14598-2及14598-6是一個支援評估的單元，單元二是規劃與管理，它是用來讓管理者作評估流程活動的計劃及管理，包含了評估技術的發展、籌獲、標準化、控制、轉換、及回饋，目的是讓專案能夠產生一個量化的評估計劃（Quantitative Evaluation Plan）。

系列中各單元的關係 (3/3)

■ 單元六是評估模組的文件化，它是提供一個如何紀錄評估模組的導引；所謂的評估模組就是指一特定的品質特徵或子特徵之一系列相關的評估資訊，裡面會包含評估的方法、技術、輸入的文件、度量及蒐集的資料、支援的程序或是工具的使用等等，使得執行度量與分析的人員能了解如何實際的應用定義好的品質模式及度量指標，讓資料的蒐集更加精確及有效率。

軟體產品評估架構 (1/2)



軟體產品評估架構 (2/2)

- 評估一個軟體品質如上圖所示，首先必須先建立評估的需求，然後才能規劃、設計、執行評估的作業。在不同情況及角色的評估下，會有不同的評估細節，但是整個流程還是要經過這四個階段。
- ISO/IEC 14598它是配合ISO/IEC 9126的品質模式來建立軟體產品品質評估模式，使用者也可以選擇其它的品質模式來建構，但是所使用的品質模式或是度量指標最好是已驗證過的或是其它的國際標準所定義的品質模式。

軟體流程評估



軟體流程評估的三種觀點

■ 以模式為基礎的評鑑

例CMM、BOOSTRAP與CMMI等。

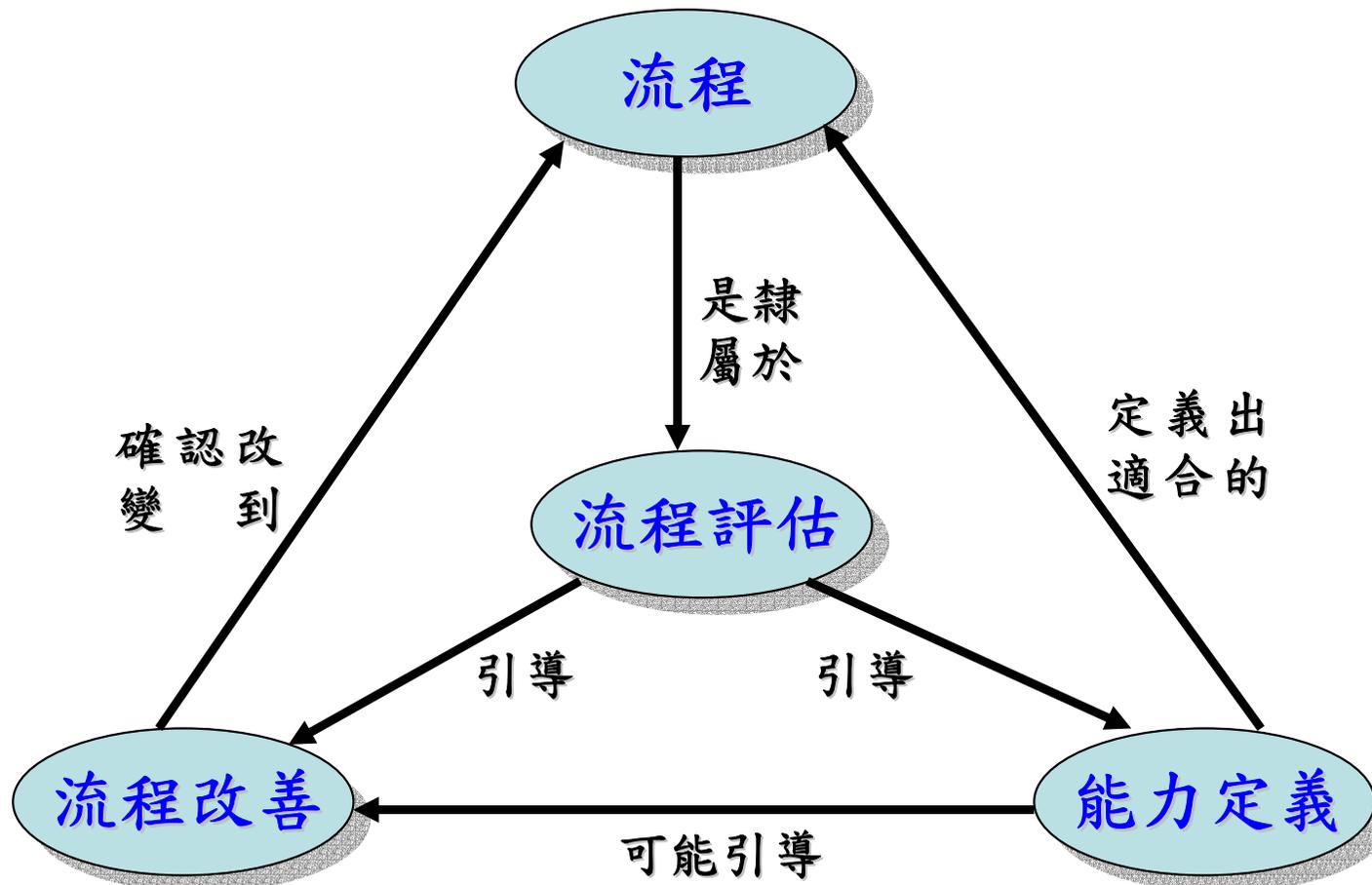
■ 以標準為基礎的評鑑

ISO/IEC TR 15504、ISO 9000等。

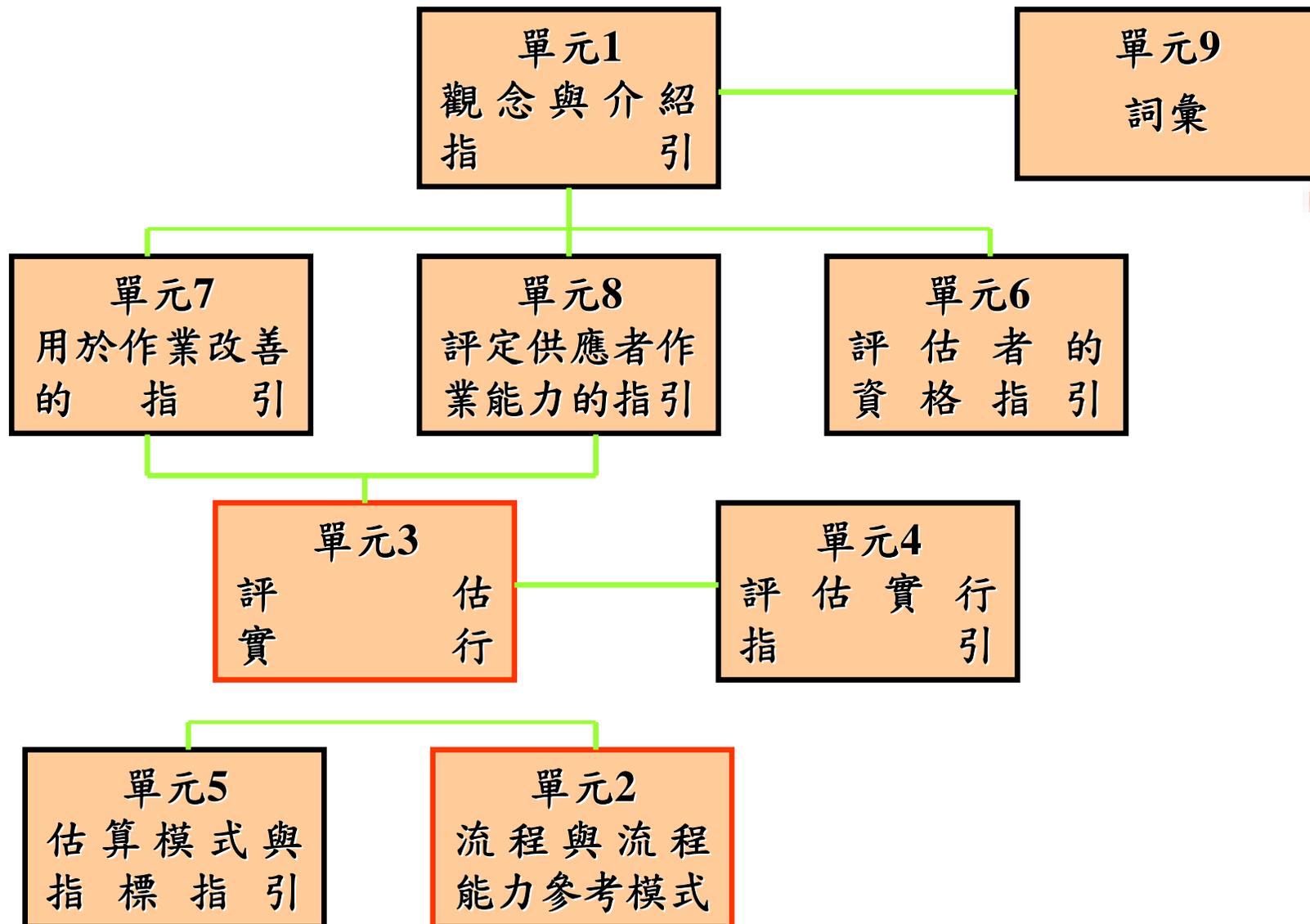
■ 以業界基準為基礎的評鑑

參考業界的平均流程能力。

軟體流程評估的概念



ISO/IEC TR 15504 的九大單元



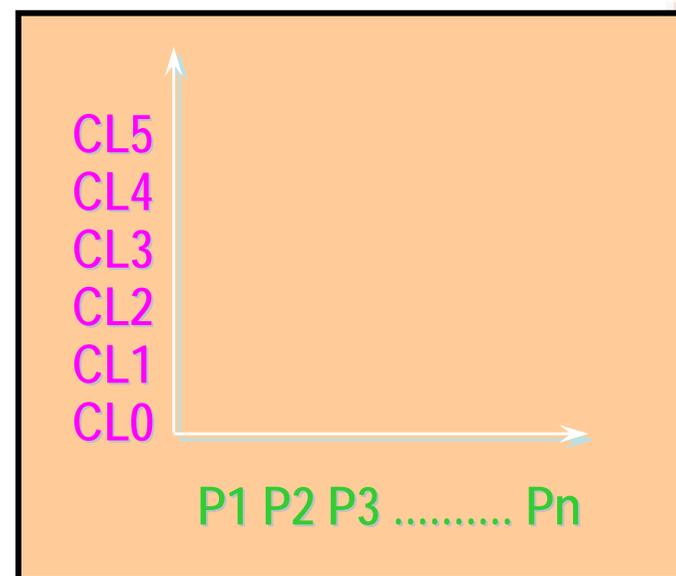
ISO/IEC TR 15504的參考模式

流程維度

由流程中一些不可或缺與可度量的目標所描繪而成，其流程的預期結果指出它的制度化程度。

流程能力維度

由一系列的流程屬性描繪而成，可適用於任何流程，且呈現了管理一個流程與改進自身執行能力的特徵需求。



ISO/IEC TR 15504 五個類別

主要的

客戶供應者 (CUS)

工程 (ENG)

支援的

支援 (SUP)

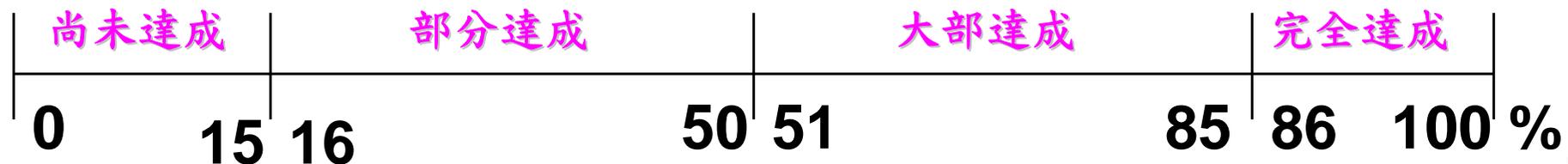
組織的

管理 (MAN)

組織 (ORG)

屬性評估的尺度

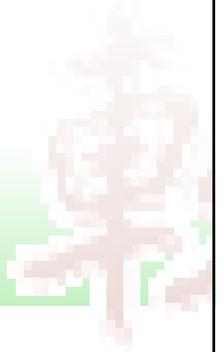
- N尚未達成(Not achieved)：
定義的屬性沒有達成證明
- P部分達成(Partially achieved)：
定義的屬性有些已經達成
- L大部達成(Largely achieved)：
定義的屬性絕大部分都已顯著達成
- F完全達成(Fully achieved)
定義的屬性完全達成



ISO/IEC TR 15504-2能力等級的符合程度

等級 屬性	一	二	三	四	五
流程效能	L or F	F	F	F	F
效能管理		L or F	F	F	F
工作產品管理		L or F	F	F	F
流程定義與裁適			L or F	F	F
流程資源			L or F	F	F
流程度量				L or F	F
流程控制				L or F	F
流程改變					L or F
不斷改變					L or F

軟體流程評估報告



ARC

- CMMI的評鑑需求 (Appraisal Requirements for CMMI, ARC)是SEI針對如何設計CMMI評鑑方法所制定的需求標準。
- 該評鑑需求定義了三個評鑑需求等級，以配合各種不同的目的，例如組織內部的流程改善評鑑、對供應商的能力評估與流程監控之需求。

評鑑需求的等級

特 徵	Class A	Class B	Class C
所需蒐集的客觀證據數量(相對)	高	中	低
評比的產生	有	無	無
所需資源(相對)	高	中	低
評鑑小組規模	大	中	小
評鑑小組主評鑑人的資格	CMMI主稽核員	CMMI主稽核員或經過訓練且有經驗的人員	經過訓練且有經驗的人員

評鑑需求的格式

-  責任
-  評鑑方法的文件化
-  評鑑的計畫與準備
-  評鑑資料的蒐集
-  資料的整理與確認
-  評比
-  報告

SCAMPI

❖ 流程改善的標準CMMI評鑑方法(Standard CMMI Appraisal Method for Process Improvement, SCAMPI)是目前唯一符合ARC需求的CMMI Class A 的評鑑方法。

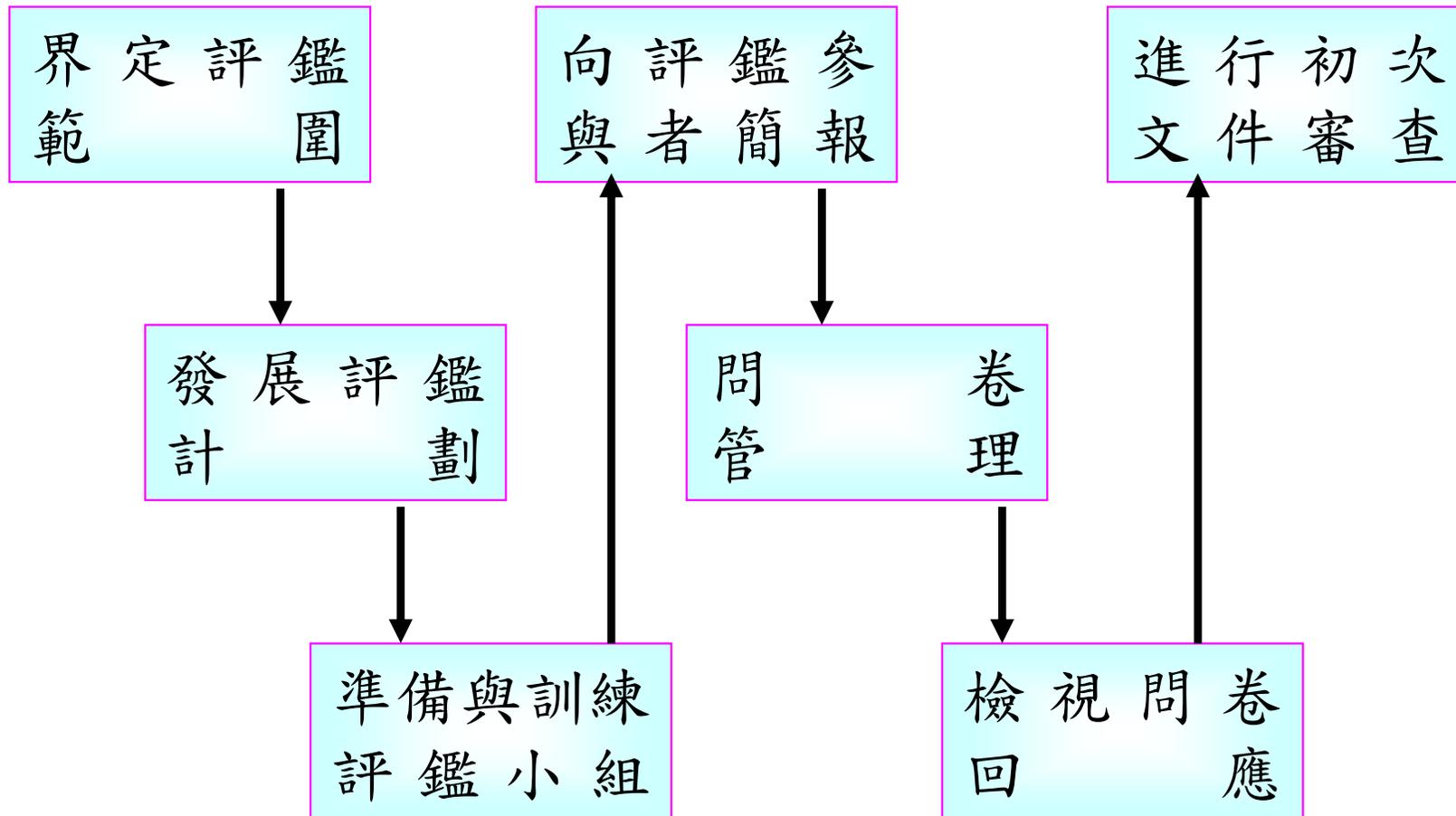
❖ SCAMPI的目的

- 組織藉由界定目前流程的優點、缺點，瞭解自己的發展能力。
- 依照CMMI模型，說明所發現的優缺點。
- 排定改善計畫的優先順序。
- 根據組織目前的成熟度或流程能力，以及考量組織的業務目標，專注於對組織最有利的改善活動。
- 評比。

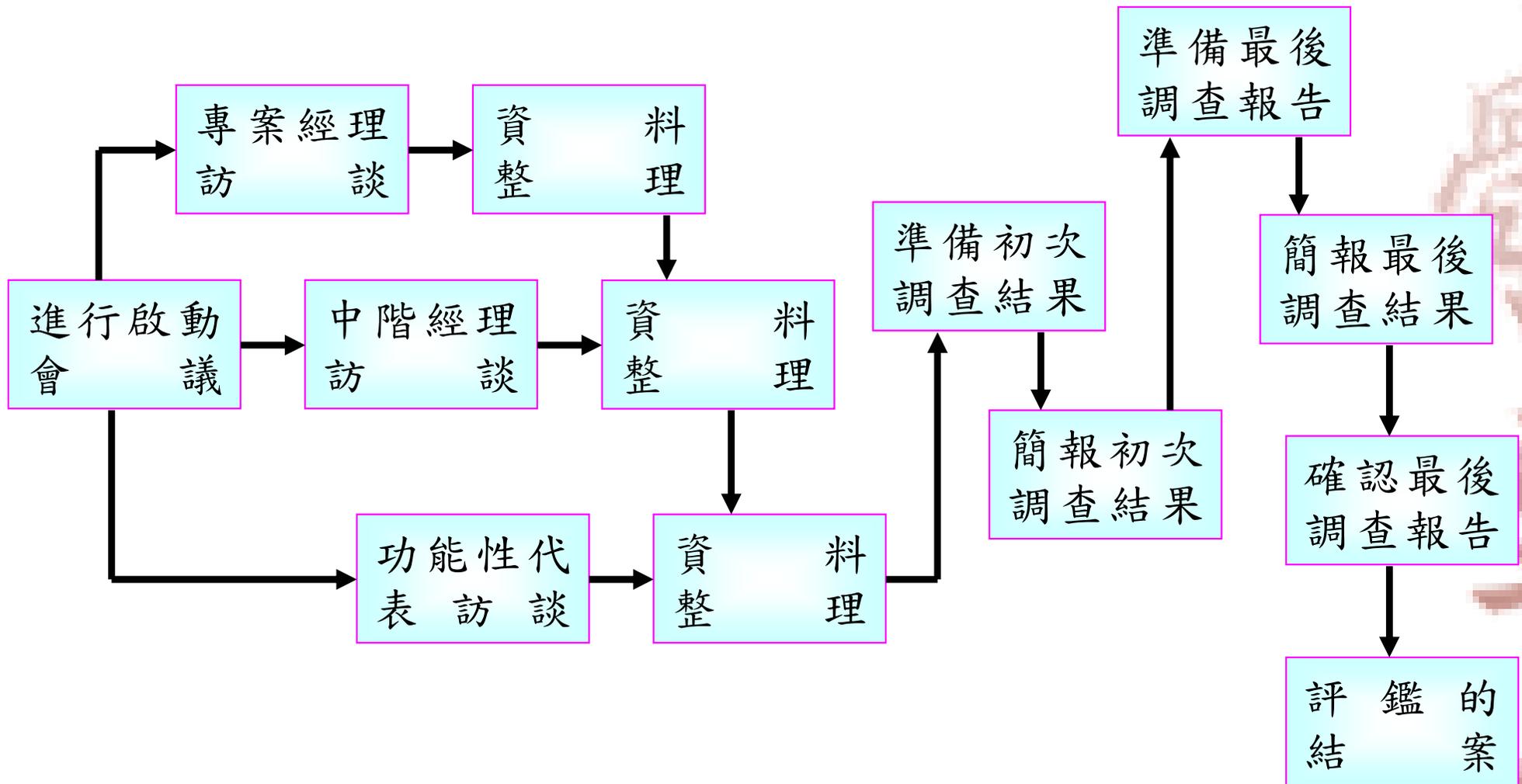
SCAMPI的三個階段

-  規劃準備階段
-  進行評鑑
-  報告結果

規劃準備階段



進行評鑑



第一天

第N天

報告結果

- 評鑑結果會向評鑑贊助人與SEI報告。通常是由評鑑小組的主任評鑑師向評鑑贊助人進行簡告，評鑑贊助人擁有該份調查報告的所有權，並可自由使用。簡報者必需確認組織確實瞭解該評鑑結果。
- 受評鑑單位的資深經理可以要求舉行高階會議，以確認他們有瞭解評鑑小組的調查報告，並得到評鑑小組的各種改善建議。
- 評鑑小組主評鑑人，會將所有評鑑過程所蒐集而來的資料，彙整後向SEI報告。

SCAMPI的評估方式

評鑑值	說明
全部達成(Fully Implemented, FI)	有直接成果且符合CMMI要求。
大部分完成(Largely Implemented, LI)	最少一個間接成果或陳述，證明執行方法的實踐無重大缺失。
部分完成(Partially Implemented, PI)	有直接成果且符合CMMI要求。
未達成(Not Implemented, NI)	最少一個間接成果或陳述，證明執行方法的實踐有發現缺點。

結論

- 本章介紹軟體產品度量值、軟體流程度量值。
- 以上相關指標可依組織收集的相關資料調適成符合組織可用的指標，並可藉由這些量化的度量指標來輔助管理者決策參考之用。
- 本章介紹軟體產品評估、軟體流程評估。
- 依循以上相關的評估標準所產生的評估結果能讓管理者了解軟體專案符合使用者的程度和作為改善的依據。

自我評量

- 軟體流程度量值與軟體產品度量值差別何在？
- 請舉例各說明五個軟體產品度量值與軟體流程度量值。
- 試說明量化度量的目的。
- 軟體產品評估的目的為何？
- 有那些軟體產品評估的標準？
- 軟體流程評估的目的為何？
- 有那些軟體流程評估的標準？

參考資料

- 軟體專案管理，林信惠，智勝出版社
- 陳俊彥，以量化軟體度量指標支援CMMI模式的導入與評鑑，國立台灣科技大學碩士論文
- ISO/IEC 9126:1991-- Product quality
- Practical Software and Systems Measurement ,Version 4.0c,Department of Defense and US Army
- Daniel Galin, Software Quality Assurance,Addison Wesley,2004
- Guidelines to Software Measurement,IFPUG,2004
- Software Process Improvement:Metrics,Measurement,and Process Modelling,Springer,2001

參考資料

- S.M. Henry and D. Kafura, Software Structure Metrics Based on Information Flow, IEEE Transactions on Software Engineering SE-7, Sep. 1981, pp. 510-518
- D.N. Card and R.L. Glass, Measuring Software Design Quality, Prentice-Hall, 1990
- Simmons, D.B, N.C. Ellis, H. Fujihara and W. Kuo Software Measurement for Project Control and Process Improvement, Prentice-Hall, 1998.
- T.J. McCabe, A Complexity Measurement, IEEE Transactions on Software Engineering SE-2, Dec. 1976, pp. 20-45
- Musa, J.D., A. Lannino and K. Okumoto, software Reliability-Measurement, Prediction, Application, McGraw-Hill, 1987

參考資料

- Hetzel Bill, Making Software Measurement Work: Building an Effective Measurement Program, QED publishing Group, QED Software Evaluation Series, 1993
- Putnam Lawrence H., Myers Ware, Measures for Excellence: Reliable Software on Time within Budget, Yourdon Press Computing Series, 1992