TSEG

Tamkang University Software Engineering Group
淡 江 軟 體 工 程 實 驗 室
http://www.tkse.tku.edu.tw/

軟體專案管理

本教材僅供修習學生閱讀使用，敬請尊重智慧財產權，勿非法使用本教材內容及相關參考資料

Presented by : Ying-Hong Wang
E-mail : inhon@mail.tku.edu.tw
Date :3/18/2012

基本規定

- **上課用書**
  - 軟體專案管理　林信惠等 著　智勝文化出版
  - 請尊重智慧財產權，勿非法影印使用教科書與參考書籍及使用盜版軟體
- **參考資料**
  - 軟體工程聯盟編撰教材
- **成績評定**
  - 出席15%、平時作業40% 、期中報告15%、期末報告20%
  - 出席成績僅考核出席與否，故不接受任何請假，但每人每學期有三次免責，全勤者於學期成績另加3分
  - 演講一場：需繳交隨堂聽講報告(10%) ，若無法安排演講，此項分數併入平時作業成績
  - 一般作業遲交每24小時扣10分、隨堂作業、期中與期末報告不可遲交
  - 期中報告繳交時間：4/12 1600前、期末報告繳交時間：5/17 1600前
- **上課方式**
  - 投影片爲主、板書爲輔
- **上課規定**
  - 手機改設震動或關機、不要私下講話

課程目標

- 瞭解軟體專案開發流程運作程序
- 學習軟體專案開發程序中各種『管理』議題與方法、可使用之工具軟體
- 輔助軟體工具：**Openproj**
  - 下載網站http://tw.openinstall.org

調整上課心態與學習態度

- 以前修課的想法可能是求過關就好
- 以前的學習態度可能是可以順利畢業就好
- 現在妳(你)應該要爲即將就業作準備
- 現在的妳(你)應該要知道爲什麼要學、要學什麼
- 現在的妳(你)應該要開始思考如何面對競爭
- 現在的妳(你)應該要準備如何打贏一場又一場的競賽
- 現在的妳(你)應該思考如何成爲企業需要的人才

對同學們的建議

- 在專業養成上 (Hard Skills)
  - 奠定紮實的基礎
  - 養成終身學習的習慣
- 在人格養成上 (Soft Skills)
  - 建立正確的態度
  - 處事三態：真誠、負責、合群

TSEG Tamkang University Software Engineering Group 淡江軟體工程實驗室 http://www.tkse.tku.edu.tw/

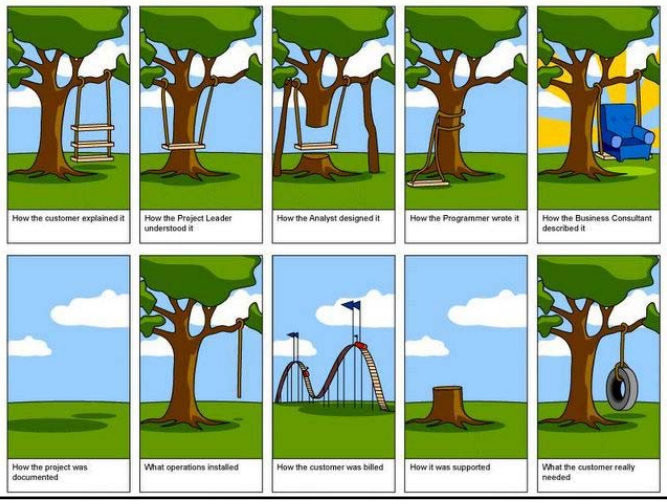
Agenda

- Overview
- Challenges of Software Industry and MIS Department
- Recall Software Development Models
- Integration Management of Software Management
- 軟體成本估計
- 時程規劃與管理

Overview

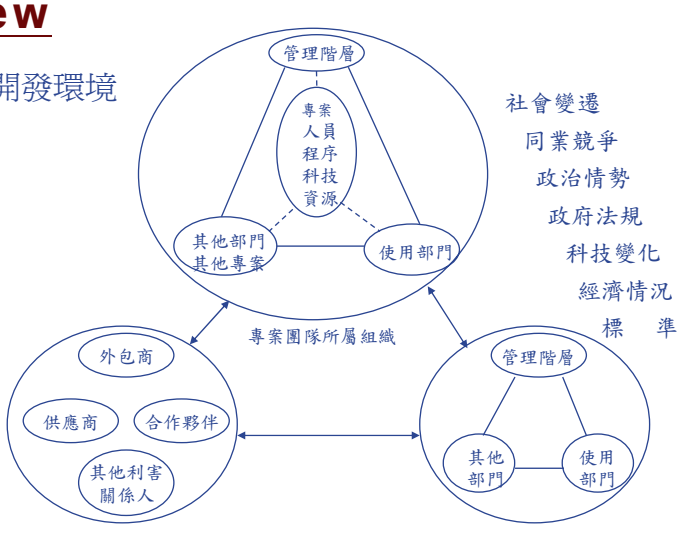
- 軟體專案開發環境

Copyright from: United States Department of Agriculture



Overview

- 軟體專案開發環境



社會變遷
同業競爭
政治情勢
政府法規
科技變化
經濟情況
標　準

Overview

- 軟體的定義與特性
  - 定義
    - 軟體是電腦程式的統稱。廣義的軟體包括電腦程式、資料庫和文件
  - 特性
    - 軟體是百分之一百由人類所設計出來的產物。
    - 軟體是知識和智慧的成果。
    - 軟體的開發，強調設計開發而非製造。
    - 軟體沒有老化現象。
    - 人們無法證明軟體完美無缺。

TSEG Tamkang University Software Engineering Group 淡江軟體工程實驗室 http://www.tkse.tku.edu.tw/

Overview

- 專案的定義與特性
  - 定義
    - 專案是一項暫時性的工作，其目的在於創造特定的產品或服務。
    - 專案是一連串的活動，以符合既定的目標並產生既定的結果。
  - 特性
    - 有預定的目標與時程。
    - 具獨特性。
    - 是暫時性的工作。
    - 是資源的整合。
    - 專案的成敗受外在因素影響甚大。
    - 專案的變動性大。

Overview

- 專案的定義與特性
  - 範例
    - 機場的興建。
    - 捷運系統的興建。
    - 軟體開發。
    - 影片製作。
    - 尋找工作。

Overview

- 管理的意涵
  - 管理功能可分爲下列五種活動：
    - 規劃
      - 目標形成的過程以及選擇目標達成的策略。
    - 組織
      - 設計角色扮演的結構和活動間相互的關係。
    - 任用
      - 人員的招募、訓練、晉用、獎懲。
    - 領導
      - 引導或指引員工以達成組織目標。
    - 控制
      - 設定目標、衡量績效、改正偏差。

Overview

- 管理的意涵
  - 若將管理視爲問題解決則可分爲下列步驟：
    - 發覺問題。
    - 瞭解問題。
    - 定義問題。
    - 設計解決問題的方案。
    - 選擇方案 。
    - 方案執行的規劃。
    - 方案的執行。
    - 回饋與預測。

Overview

- 軟體專案管理
  - 軟體專案管理所涵蓋的範圍可能從
    - 管理主題構面；
    - 作業流程構面；
    - 開發程序構面；

    等三方面來說明。

Overview

- 軟體專案管理
  - 管理主題構面
    - 管理者關心的主題，如
    - 成本、時程、品質、人員、 型態、風險、資訊等
    - 並以此維度與管理功能維度架構成的專案管理架構

管理功能 / 管理主題	規劃	控制	組織	任用	領導	決策
成本	✓	✓			✓	✓
時程	✓	✓			✓	✓
品質	✓	✓	✓		✓	✓
人員	✓	✓	✓	✓	✓	✓
型態	✓	✓	✓		✓	✓
風險	✓	✓	✓		✓	✓
資訊	✓	✓	✓			✓

Overview

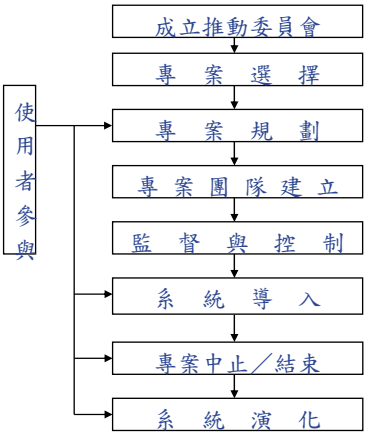
- 軟體專案管理
  - 作業流程構面
    - 作業流程構面是從專案形成到結束的流程，依時間的先後順序，找出軟體專案開發所涉及的重要作業，每一作業表示一個階段性的工作。
    - 它包括專案選擇、專案規劃、專案團隊建立、外包、專案監督與控制、系統導入與維護、專案中止與結束。
    - 管理功能的核心：
      - 規劃
      - 執行
      - 控制

Overview

- 軟體專案管理
  - 作業流程構面
    - 作業流程構面是從專案形成到結束的流程，依時間的先後順序，找出軟體專案開發所涉及的重要作業，每一作業表示一個階段性的工作。
    - 它包括專案選擇、專案規劃、專案團隊建立、外包、專案監督與控制、系統導入與維護、專案中止與結束。
    - 管理功能的核心：
      - 規劃
      - 執行
      - 控制

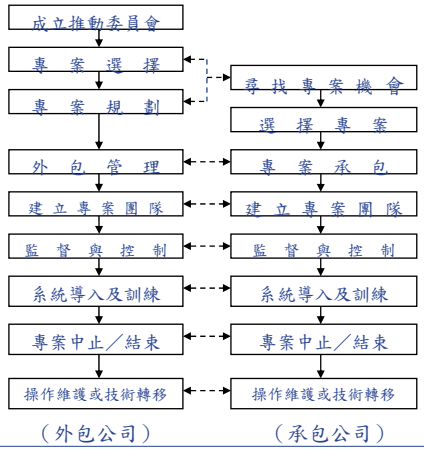
Overview

- 軟體專案管理
  - 作業流程構面 – 自行開發的作業流程



Overview

- 軟體專案管理
  - 作業流程構面 – 外包開發的作業流程

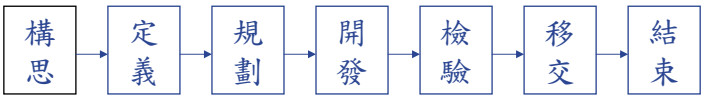


Overview

- 軟體專案管理
  - 開發程序構面
    - 開發程序構面將專案管理視爲開發程序的管理。
    - 程序是達到某特定目標的一系列活動。
    - 一個程序將某些輸入轉換爲輸出，使得附加價值得以提升。

Overview

- 軟體專案管理
  - 開發程序的階段



TSEG Tamkang University Software Engineering Group 淡江軟體工程實驗室 http://www.tkse.tku.edu.tw/

Overview

- 軟體專案管理
  - 開發程序的階段
    - 構思階段的主要內容
      - 尋找機會。
      - 瞭解狀況。
      - 瞭解利害相關人。
      - 可行性研究。

Overview

- 軟體專案管理
  - 開發程序的階段
    - 定義階段的主要內容；清楚界定專案的目標、範圍及預期的結果
      - 定義目標。
      - 定義範圍。
      - 定義交付項目。
      - 定義策略。
      - 定義利害關係人。

TSEG Tamkang University Software Engineering Group 淡江軟體工程實驗室 http://www.tkse.tku.edu.tw/

Overview

- 軟體專案管理
  - 開發程序的階段
    - 規劃階段的主要內容：定義細部的執行內容
      - 需求分析。
      - 工作分解圖。
      - 時程規劃。
      - 資源。
      - 組織圖。
      - 人事安排。
      - 品質。
      - 風險。
      - 控制。
      - 方案評估。
      - 政策、規劃及指引。

Overview

- 軟體專案管理
  - 開發程序的階段
    - 開發階段的主要內容：涵蓋專案執行的核心
      - 需求規格。
      - 設計。
      - 編碼與單元測試。
      - 管理。

TSEG Tamkang University Software Engineering Group 淡江軟體工程實驗室 http://www.tkse.tku.edu.tw/

Overview

- 軟體專案管理
  - 開發程序的階段
    - 檢驗階段的主要內容：認證專案是否符合品質目標。
      - 衡量
      - 評估
      - 審查
      - 測試

Overview

- 軟體專案管理
  - 開發程序的階段
    - 移交階段的主要內容：將完成的系統移轉給顧客
      - 系統安裝。
      - 系統轉移。
      - 訓練。

TSEG Tamkang University Software Engineering Group 淡江軟體工程實驗室 http://www.tkse.tku.edu.tw/

Overview

- 軟體專案管理
  - 開發程序的階段
    - 結束階段的主要內容
      - 專案後的審查。
      - 人員安置。
      - 獎勵。
      - 維護與支援。

Overview

- 軟體專案成功的關鍵因素
  - 專案的成敗可從不同角度來探討
    - 滿足合約的要求，並且符合需求規格。
    - 符合成本效益。
    - 由顧客滿意度來判定。
    - 以長期的利益爲考量。

Overview

- 軟體專案成功的關鍵因素 -- Phan 等，1995

成功因素	類別
1.及時回應最終使用者的意見。	溝通、更改管理
2.在開發階段重大的設計更改應該降到最少。除非萬不得已，否則應延至下個版本再更改。	更改管理
3.溝通與協調的問題應該降到最低。	溝通、領導
4.專案領導者應具備良好的管理技巧。	領導、管理
5.人員流動應儘量降低。	任用、人力資源管理
6.專案間的人員安排要盡量平衡。	任用
7.外來的回饋應儘速與專案團隊人員溝通，以立即採取更正的行動。	溝通、風險管理
8.不該爲了符合預算與時程而犧牲品質。	品質管理
9.符合顧客需求規格。	品質管理
10.保持與顧客及外在環境良好的介面	溝通、人際關係
11.專案的問題和目標必須清楚定義	規劃
12.開發團隊應有足夠的專業人才	任用
13.應有效利用推動委員會及使用者諮詢小組。	組織、溝通

Overview

- 軟體專案成功的關鍵因素 -- Jones 等，1996

成 功 因 素	類 別
1.有效的專案規劃	規劃
2.有效的成本估計	規劃、成本估計
3.有效的專案管理	規劃、控制
4.有效的追蹤里程碑	時程管理
5.有效的品質保證	品質管理
6.有效的更改管理	更改管理
7.有效的開發程序	程序管理
8.有效的溝通	溝通
9.有能力的專案經理	任用
10.有能力的技術人員	任用
11.有效利用專家	任用
12.充足的再用模組	技術管理

Challenges of Software Industry and MIS Department

- 資訊軟體業經營環境的新挑戰
  - 追求規模經濟
  - 建立專業技術
  - 提高生產力
  - 培養創意
  - 國際化

Challenges of Software Industry and MIS Department

- 資訊軟體業經營環境的新挑戰
  - 追求規模經濟
    - 軟體業的經營策略漸漸由訂製型軟體轉爲套裝軟體。
    - 套裝軟體的固定成本高但變動成本低，規模經濟的效果極爲明顯。
    - 網際網路的發達擴大了規模經濟的影響。
    - 利用低價或免費軟體來擴大市場占有率及蒐集市場資訊，以作爲系統更新的參考及套牢使用者的策略。

TSEG Tamkang University Software Engineering Group 淡江軟體工程實驗室 http://www.tkse.tku.edu.tw/

Challenges of Software Industry and MIS Department

- 資訊軟體業經營環境的新挑戰
  - 建立專業技術
    - 建立專業技術是避免價格競爭的有效方法。
    - 在垂直分工的趨勢下建立專業技術，才能維持競爭優勢。尋找利基市場、系統整合及發展垂直分工的專業技術都是朝向專業化的做法。

Challenges of Software Industry and MIS Department

- 資訊軟體業經營環境的新挑戰
  - 提高生產力
    - 軟體的成本不斷提高，但生產力卻沒有明顯改善。
    - 提高生產力的方法有：
      - 引進先進開發技術。
      - 改善專案管理。
      - 吸引及培養人才。
    - 軟體專案的開發要更重視團隊的生產力，而非個人突出的表現。

TSEG Tamkang University Software Engineering Group 淡江軟體工程實驗室 http://www.tkse.tku.edu.tw/

Challenges of Software Industry and MIS Department

- 資訊軟體業經營環境的新挑戰
  - 培養創意
    - 創意可以提高產品的價值。
    - 營造自由開放的環境、尊重智慧財產權、鼓勵嘗試新方法的價值觀等都有利於創新。
    - 兼顧個人創意和集體創意。

Challenges of Software Industry and MIS Department

- 資訊軟體業經營環境的新挑戰
  - 國際化
    - 爲追求規模經濟，應將產品定位在更大的市場，如華文或國際市場。
    - 國際資源整合、國際分工合作是必然的趨勢。
    - 企業應儘早培養跨國專案管理人才。

TSEG Tamkang University Software Engineering Group 淡江軟體工程實驗室 http://www.tkse.tku.edu.tw/

Challenges of Software Industry and MIS Department

- 企業資訊部門的新挑戰
  - 技術落差
  - 組織變動
  - 人才覓集困難
  - 資訊部門的重新定位

Challenges of Software Industry and MIS Department

- 企業資訊部門的新挑戰
  - 技術落差
    - 新科技不斷更新使得資訊人員對外的依賴日深。
    - 預算及人力的增加無法跟上新科技的進步。
    - 當外部資訊公司的替代性增加時將面臨內部人員的轉業、重組、淘汰等問題。

TSEG Tamkang University Software Engineering Group 淡江軟體工程實驗室 http://www.tkse.tku.edu.tw/

Challenges of Software Industry and MIS Department

- 企業資訊部門的新挑戰
  - 組織變動
    - 企業快速成長或企業國際化將帶來資訊系統擴充、人員調派、跨國專案管理等問題。
    - 企業改組、購併、多角化經營將帶來系統修改、系統整合等問題。
    - 專案管理者應多充實企業經營的知識，才能瞭解專案在組織中所扮演的角色。
    - 專案管理者應更重視專案與組織目標的契合。

Challenges of Software Industry and MIS Department

- 企業資訊部門的新挑戰
  - 人才覓集困難
    - 資訊人員流動大。
    - 資訊人力供需失調，人才難覓。
    - 資訊人才養成時間長。

TSEG Tamkang University Software Engineering Group 淡江軟體工程實驗室 http://www.tkse.tku.edu.tw/

Challenges of Software Industry and MIS Department

- 企業資訊部門的新挑戰
  - 資訊部門的重新定位
    - 由軟體開發的角色轉爲資訊提供的角色。
    - 由技術執行的角色轉爲技術管理的角色。
    - 強化對所屬企業的瞭解及對使用者需求的瞭解，這些是外部資訊公司難以取代的功能。

Software Development Models

- 軟體開發模式
  - 軟體開發模式描述軟體開發一系列的步驟或階段。
  - 軟體開發模式可分爲
    - 生命週期模式(Life-Cycle Models)，
    - 程序模式(Process Models)。

TSEG Tamkang University Software Engineering Group 淡江軟體工程實驗室 http://www.tkse.tku.edu.tw/

Software Development Models

- 生命週期模式(Life-Cycle Models)
  - 將軟體開發的階段及各階段的關係以概念性的模式表示。
  - 隱含著開發程序的時間順序。
  - 方法有：瀑布模式、快速雛型法、漸進模式、演化雛型模式、螺旋模式和同步模式。
  - 每一種方法代表一種系統開發的策略。

Software Development Models

- 程序模式(Process Models)
  - 爲了某一特定目的而設計一系列的活動。
  - 活動內容包括：
    - 軟體開發程序
    - 品質改善
    - 演化程序
    - 專案管理程序
    - 顧客導向程序
    - 需求程序
    - 維護程序
    - 同步程序

TSEG Tamkang University Software Engineering Group 淡江軟體工程實驗室 http://www.tkse.tku.edu.tw/

Software Development Models

- 生命週期模式的最終結果是軟體系統；
- 程序模式的最終結果則是某一管理目標。

Software Development Models

- 軟體專案依循某一開發模式有多方面的優點：
  - 統一的名詞及概念有助於溝通、規劃及管理。
  - 有利於標準、規範與政策的建立及推行。
  - 可提供評估、檢核及里程碑的參考特點。
  - 簡要地描繪重要的功能、活動及特性。
  - 開發過程更結構化、更容易管理。
  - 提供一個不斷改善的基礎。

TSEG Tamkang University Software Engineering Group 淡江軟體工程實驗室 http://www.tkse.tku.edu.tw/

Software Development Models

- Recall 生命週期模式 (Life-Cycle Models)：
  - 以下僅以瀑布模式、快速雛型法與螺旋模式三種生命週期模式的軟體開發方法作爲介紹。
- Recall程序模式 (Process Models)：
  - 以下分別以開發程序模式、反覆定義與改進程序模式、連續改進程序模式三種程序模式的開發方法作爲介紹

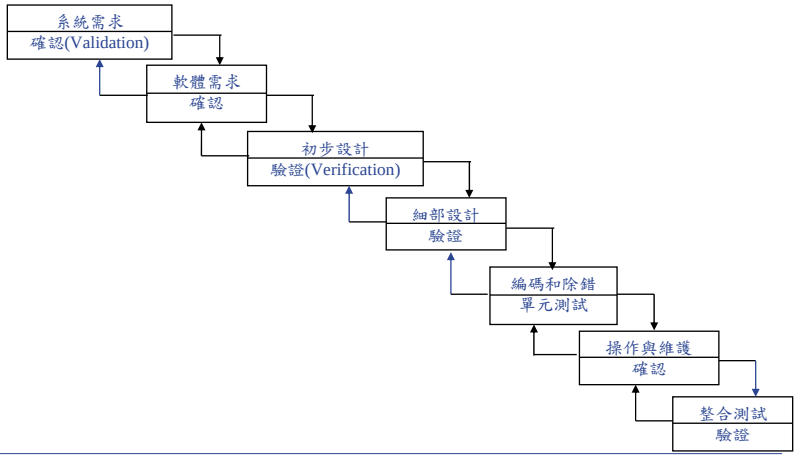
Software Development Models

- 瀑布模式
  - 瀑布模式是基於下列的構想而設計：
    - 軟體開發應依照一序列的階段進行。
    - 一個階段的產出必須經過驗證或確認才能視爲完成。
    - 任何更改、錯誤或爭議都必須回溯到前面相關的階段加以修正。
    - 若發現錯誤或新需求時，必須回溯到前面相關的階段。

TSEG Tamkang University Software Engineering Group 淡江軟體工程實驗室 http://www.tkse.tku.edu.tw/

Software Development Models

- 瀑布模式



Software Development Models

- 瀑布模式的管理意義
  - 它鼓勵依生命週期階段來進行規劃。每一階段的結束正好成爲管理控制的里程碑。
  - 每一階段的產出都必須經過確認、驗證或測試。確認(validation)用來檢驗產出是否符合顧客的需求，以真實世界的問題爲檢驗的對象。驗證 (verification)是檢驗系統是否依規格正確地執行。
  - 從事前階段工作的人，有責任將正確、完整、可行且容易瞭解的產出移交給下一階段的人員。
  - 開發的程序變得更結構化且更容易管理。

TSEG Tamkang University Software Engineering Group 淡江軟體工程實驗室 http://www.tkse.tku.edu.tw/

Software Development Models

- 瀑布模式的的缺點
  - 必須到了最後階段才能看到可執行的軟體系統，風險太高。
  - 過於複雜與費時。
  - 若需求不明確，而每一階段又要求非常結構化、嚴謹、完整的開發方法，將使得開發時程拉得很長。

Software Development Models

- 快速雛型法
  - 快速雛型法主要是基於下列的構想：
    - 需求變更無可避免。
    - 一個可看得到、可操作的雛型是開發者與顧客溝通的良好工具。
    - 雛型的建造及修改應該要非常快速，以因應顧客的要求。
    - 提高使用者參與的意願，進而改進使用者滿意度。

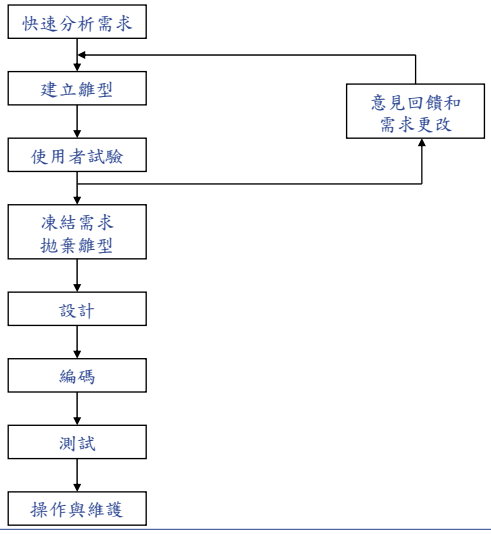
TSEG Tamkang University Software Engineering Group 淡江軟體工程實驗室 http://www.tkse.tku.edu.tw/

Software Development Models

- 快速雛型法的層次
  - Cherveny 提出了一個三階層的雛型法架構：
    - 第一階雛型
      - 又稱輸入/ 輸出設計(Input / Output Design)，它只產生輸入/ 輸出的螢幕及列印的報表。
    - 第二階雛型
      - 包含使用介面及系統功能，經由第四代語言及關聯式資料庫快速地設計一個可執行的系統。
    - 第三階雛型
      - 發展一個適應環境的雛型，這種策略將系統永遠當作一個雛型，以因應外在環境的不確定性。

Software Development Models

- 快速雛型法



Software Development Models

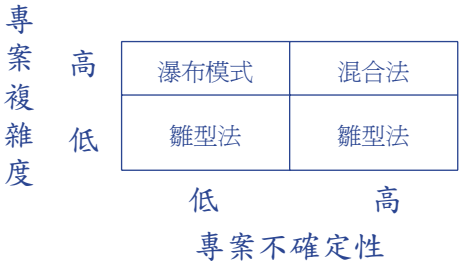
- 快速雛型法的優點
  - 雛型法可增進使用者參與。
  - 可看得到、可操作的雛型成爲開發者與使用溝通的基礎。
  - 需求分析的時間及成本可以降低。
  - 系統的正確性提高。

Software Development Models

- 快速雛型法的缺點
  - 由於溝通較複雜、專案開發較動態，因此，專案的管理及控制也較爲困難。
  - 較難建造大型系統的雛型。
  - 缺乏深入的分析及開發的工具，因此系統的效率較差。

Software Development Models

- 開發策略與專案特性
  - 雛型法可以和傳統的瀑布模式混合或獨自運作，Burns與Dennis提出了一個二維的架構，將專案依不確定性與複雜性兩個維度來區分選擇的策略。



Software Development Models

- 系統開發策略的權變架構
  - Louadi 合併 Burns 與 Dennis 的架構及 Chrveny 的三階層分類，提出了權變 (Contingency)架構。

專案複雜度		低	高
	高	交易處理系統(TPS) 瀑布模式輔以第一階雛型法	資訊回報系統(IRS) 瀑布模式輔以第二階雛型法
	低	使用者開發系統(EUDA) 第三階雛型法	決策支援系統(DSS) 第三階雛型法輔以瀑布模式

專案不確定性

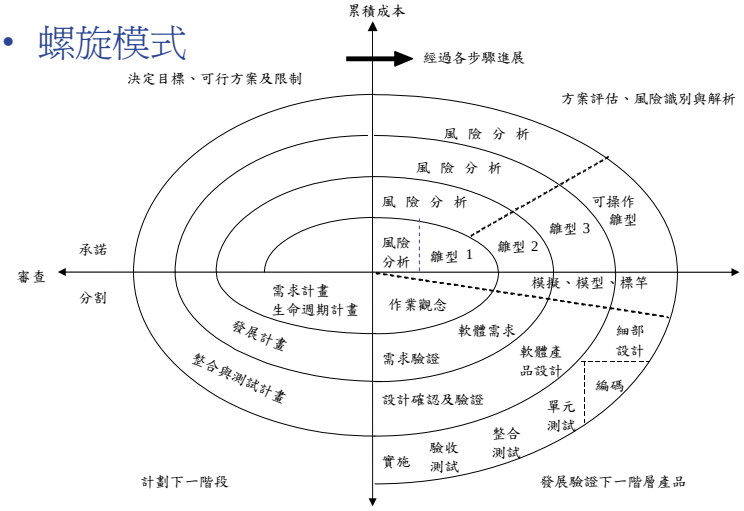
TSEG Tamkang University Software Engineering Group 淡江軟體工程實驗室 http://www.tkse.tku.edu.tw/

Software Development Models

- 螺旋模式
  - 螺旋模式是基於下列的構想而產生：
    - 在生命週期的每一階段，應該主動發覺風險並設法解決。
    - 運用模擬、雛型、模式建立、標竿(Benchmarks)等方法來降低風險。
    - 每一階段都必須經過確認或驗證。
    - 考量每一階段的目標、可行的方案及限制條件。

Software Development Models

- 螺旋模式



Software Development Models

- 螺旋模式的特性
  - 結合瀑布模式、雛型模式、風險分析和逐步規劃的精神。
  - 強調每一階段的風險分析。
  - 考慮因素非常廣泛，適用於大型而高風險的專案，對於小型或需求明確的系統此法會太複雜，且成本太高。

Software Development Models

- 軟體開發程序模式(Software Process Model)是爲達成某一特定目標之一系列活動，其定義如下
  - 「程序是一個系統，它將輸入、活動、工作流程、資訊流程及其他相關的項目轉換成特定的結果與產品。」
  - 「程序是一系列經過特殊安排的步驟以達到特定的目標，軟體開發程序的目標是完成或改進符合品質要求的軟體產品。」

TSEG Tamkang University Software Engineering Group 淡江軟體工程實驗室 http://www.tkse.tku.edu.tw/

Software Development Models

  - 「軟體程序是一組的工具、方法、做法以製造軟體產品。」
  - 「程序模式描述一個開發階段中的工作，各工作間的關係，以及啓動工作的條件。」

Software Development Models

- 開發者依不同的目的而設計不同類型的程序模式，分別有：
  - 開發程序模式
  - 反覆定義與改進程序模式
  - 連續改進程序模式

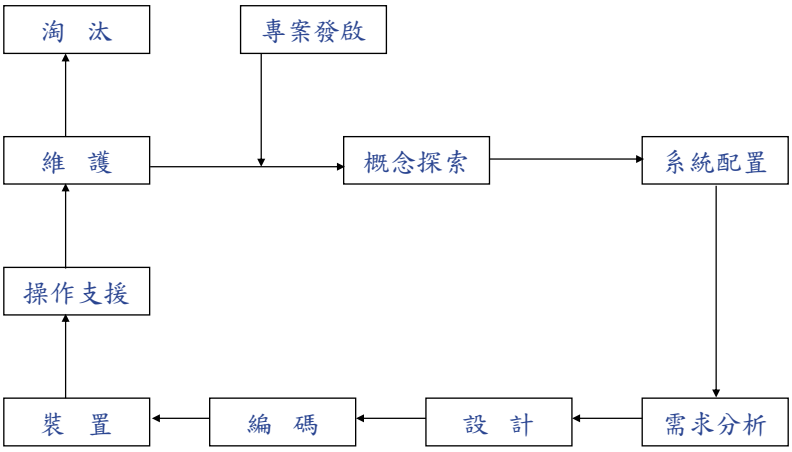
TSEG Tamkang University Software Engineering Group 淡江軟體工程實驗室 http://www.tkse.tku.edu.tw/

Software Development Models

- 開發程序模式
  - 開發程序模式就像生命週期模式，這種模式的重要概念如下：
    - 程序是一個連續的循環，是一個持續進行、不斷改進的過程。
    - 一個階段的輸出成爲下一階段的輸入。
    - 程序中的每一階段都包含一系列的活動。
    - 每項活動都有負責的人員。
    - 每項活動都必須依循設定的標準與程序，每一階段都必須定義它所包含的活動。

Software Development Models

- 開發程序模式



Software Development Models

- 反覆定義與改進程序模式 (或稱 反覆模式)
  - 反覆模式是基於下列的構想：
    - 將程序視爲產品。程序產品(Process Product)可包含套裝軟體、文件指引、訓練、諮詢顧問與工具開發等。
    - 程序產品應配合顧客的特殊需求以達成其企業目標。

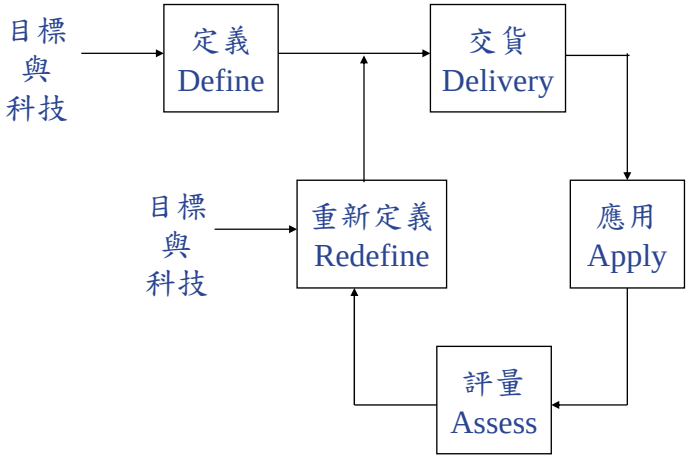
Software Development Models

- 反覆模式
  - 反覆模式是基於下列的構想：
    - 將程序視爲產品。程序產品(Process Product)可包含套裝軟體、文件指引、訓練、諮詢顧問與工具開發等。
    - 程序產品應配合顧客的特殊需求以達成其企業目標。
    - 強調反覆的定義和改進，並透過下列的方法來達成：
      - 使用者使用程序產品後將資訊回饋。
      - 對程序問題的即時反應。
      - 專案計畫與專案程序間密切的協調。
      - 適應環境的改變並引進新科技以創造優勢。
    - 程序的定義必須符合組織的目標並適合所處的環境。組織的目標如時程控制、成本降低、品質提升。環境因素如科技、組織型態、產品類別等。

TSEG Tamkang University Software Engineering Group 淡江軟體工程實驗室 http://www.tkse.tku.edu.tw/

Software Development Models

- 反覆模式



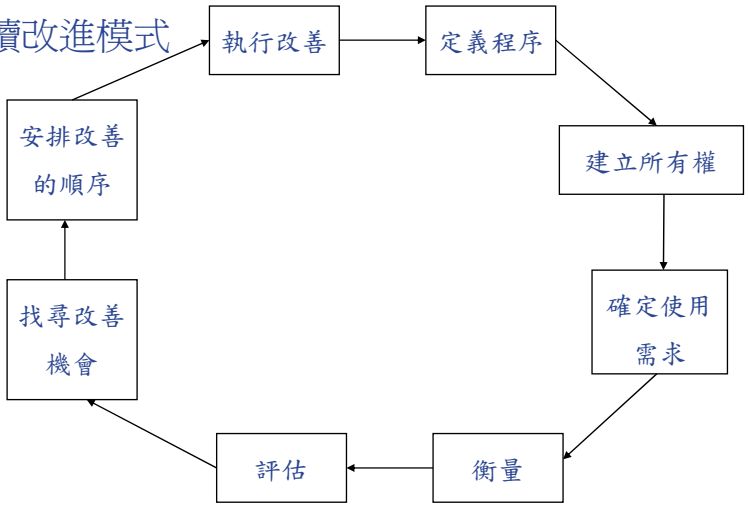
Software Development Models

- 持續改進模式
  - 此模式共分爲八個階段，用以反覆且連續地改善程序
    - 定義程序
    - 建立所有權
    - 確定使用需求
    - 衡量
    - 評估
    - 找尋改善的機會
    - 安排改善的順序
    - 執行改善

TSEG Tamkang University Software Engineering Group 淡江軟體工程實驗室 http://www.tkse.tku.edu.tw/

Software Development Models

- 持續改進模式



補充—eXreme Programming

- eXtreme Programming (XP, 極致編程、終極制程)
  - 90年代初期，由Kent Black與Ward Cunningham一起合作所提出，其成功自於強調客戶滿意及促進團隊合作。
  - 主要以下列四個方向來促進軟體開發：
    - 溝通(communication)
    - 簡化(simplicity)
    - 回饋(feedback)
    - 勇氣(courage)

TSEG Tamkang University Software Engineering Group 淡江軟體工程實驗室 http://www.tkse.tku.edu.tw/

eXreme Programming

- XP事實上是一種嚴謹且有規律的軟體開發方式，其發展已有5 年的歷史，它已在一些對成本重視的公司中被實證。
- XP授權開發人員自信的回應客戶需求的改變，既使是在軟體生命週期的後期。
- XP同時強調團隊合作，管理人員、客戶及開發人員都是貢獻於開發高品質軟體團隊中的成員。

eXreme Programming

- 溝通
  - XP 的程式設計師與其客戶及程式設計師同僚溝通
- 簡化
  - 保持其設計簡單且清晰
- 回饋
  - 一開始便不斷從測試軟體獲的回饋，儘可能提早交付系統給客戶並且立即依據其建議改善軟體。
- 勇氣
  - 基於這個基礎XP 的程式設計師能夠勇於回應持續改變中的需求及技術。

Integration Management of Software Project

- 軟體專案的啓始
- 軟體專案計畫書的規劃
- 軟體專案計畫書的內容
- 軟體專案計畫書的撰寫與發展
- 軟體專案的整體變更與控制

Integration Management of Software Project

- 軟體專案的啓始
  - 正式宣告一個新軟體專案的存在或是已存在的專案應持續進入下一個階段的一序列程序
  - 造成專案之授權原因有：
    - 市場需求
    - 商業需求
    - 顧客需求
    - 追求新科技
    - 法律要求
    - 社會需要

TSEG Tamkang University Software Engineering Group 淡江軟體工程實驗室 http://www.tkse.tku.edu.tw/

Integration Management of Software Project

- 軟體專案啓始的投入
  - 軟體產品說明
    - 工作條款(Statement Of Work, SOW)
  - 策略計畫
  - 專案選擇準則
  - 歷史性資料

Integration Management of Software Project

- 軟體專案啓始的工具與技術
  - 軟體專案選擇方法
    - 利潤衡量法
    - 有限最佳法
  - 專業判斷

TSEG Tamkang University Software Engineering Group 淡江軟體工程實驗室 http://www.tkse.tku.edu.tw/

Integration Management of Software Project

- 軟體專案啓始的產出
  - 軟體專案核準證明
  - 專案經理之確認與指派
  - 限制條件
    - 如預算與人力
  - 假設事項

Integration Management of Software Project

- 專案規劃重要考量因素
  - 一般而言，專案規劃的主要對象爲3P，它包括
    - 「人」(People)、
    - 「產品」(Product) 、 與
    - 「程序」(Process) 。

TSEG Tamkang University Software Engineering Group 淡江軟體工程實驗室 http://www.tkse.tku.edu.tw/

Integration Management of Software Project

- 專案規劃時必須考慮的重要問題如下：
  - 進行一個專案的範疇爲何?
  - 採用何種品質標準？
  - 工作順序及時間的需求爲何?
  - 一個專案所需的專案成本；
  - 專案所面臨的風險；
  - 一個專案所需的軟體設計人員。

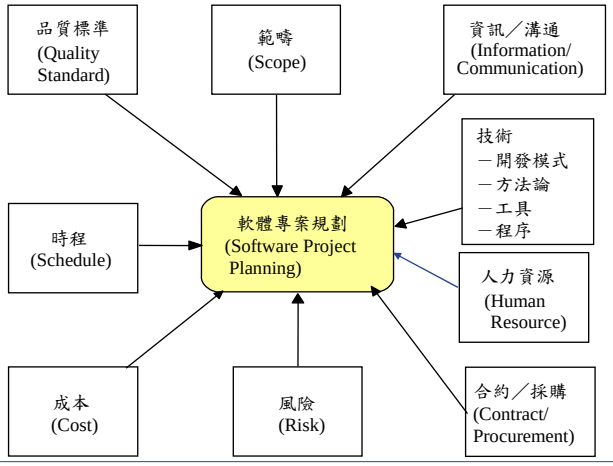
Integration Management of Software Project

- 專案規劃時必須考慮的重要問題如下：(續)
  - 一個專案所需的需求資源與採購
  - 專案所需的人員溝通與資訊傳遞格式
  - 一個專案所需的軟體設計人員
  - 一個專案所需的需求資源與採購
  - 專案所需的人員溝通與資訊傳遞格式

TSEG Tamkang University Software Engineering Group 淡江軟體工程實驗室 http://www.tkse.tku.edu.tw/

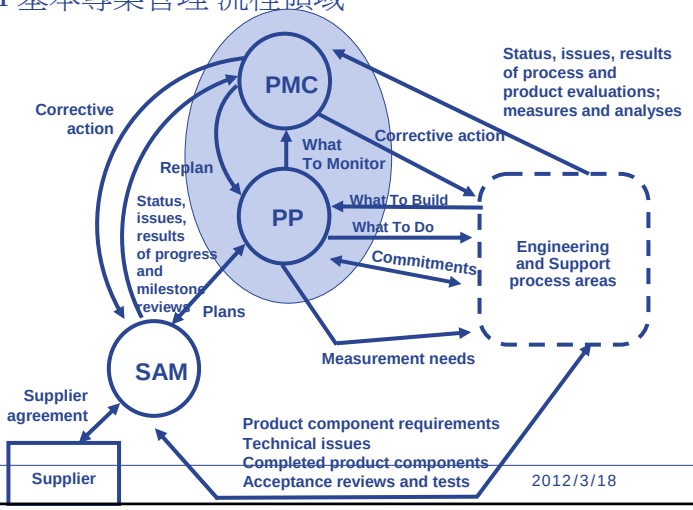
Integration Management of Software Project

- 軟體專案規劃



Integration Management of Software Project

- CMMI 基本專案管理 流程領域



Integration Management of Software Project

- CMMI Level 2 軟體專案規劃
  - 軟體專案規劃的目的是爲執行軟體工程和管理軟體專案制定合理的計畫
  - 軟體專案規劃包含所要執行工作的發展評估、建立必要的承諾和定義執行工作的計畫。

Integration Management of Software Project

- CMMI Level 2 軟體專案規劃
  - 軟體專案規劃流程包括以下步驟：估計軟體工作產品的規模及所需的資源、制定時間表、鑒別和評估軟體風險以及協商的承諾。
    - 爲了建立軟體專案計畫（即軟體開發計畫），可能需要重複地採取這些步驟。
  - 該計畫提供執行和管理軟體專案活動的基礎，並按照軟體專案的資源、約束和能力，闡述對軟體專案顧客所作的承諾。

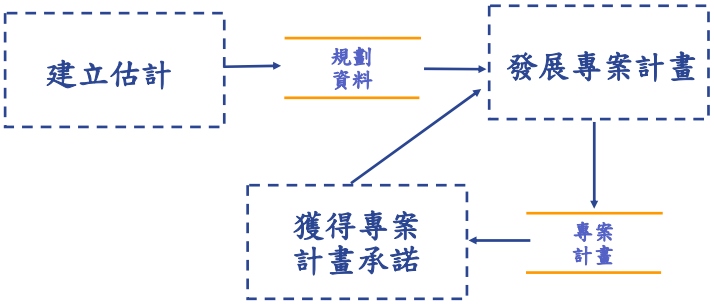
TSEG Tamkang University Software Engineering Group 淡江軟體工程實驗室 http://www.tkse.tku.edu.tw/

Integration Management of Software Project

- 軟體專案規劃步驟
  - 階段一：建立評量
    - 專案規劃評量參數建立與維護
  - 階段二：發展專案計畫
    - 專案計畫建立與維護，以做爲管理專案之基準
  - 階段三：獲得專案計畫承諾
    - 專案計畫承諾建立與維護

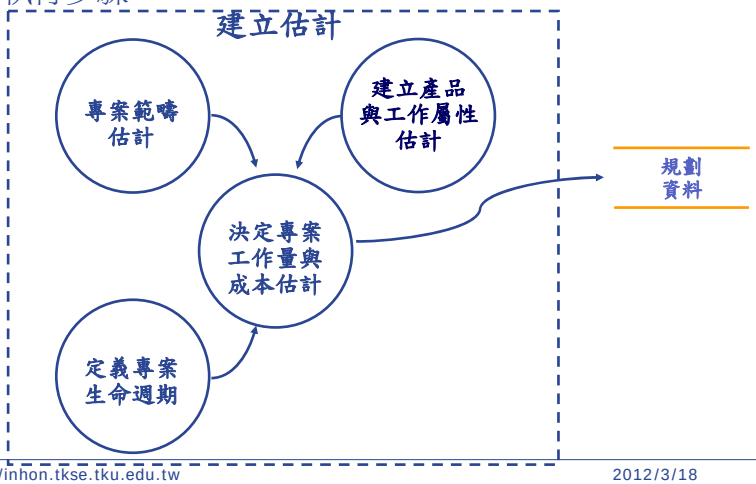
Integration Management of Software Project

- 軟體專案規劃執行步驟 一



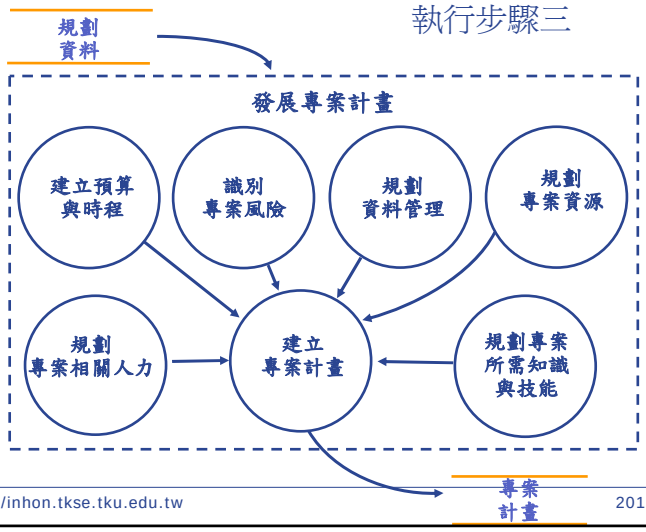
Integration Management of Software Project

- 執行步驟二



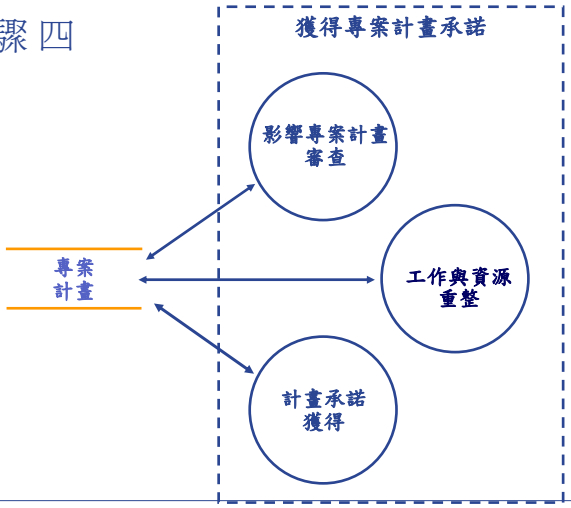
Integration Management of Software Project

執行步驟三



Integration Management of Software Project

- 執行步驟 四



Integration Management of Software Project

- 專案計畫書
  - 一般而言，軟體專案計畫書至少須包含下列幾個重要項目(Phillips, 1998)：
    - 作業目標與範圍
    - 專案工作內容，包括工作項目、時程表、負責人、完成產品項目、開發方式、技術與工具。
    - 資源估計與分配，包括專案組織、人力、硬體與軟體需求等。
    - 相關計畫，包括品質計畫、型態管理計畫與整合計畫。

TSEG Tamkang University Software Engineering Group 淡江軟體工程實驗室 http://www.tkse.tku.edu.tw/

Integration Management of Software Project

- 專案計畫書
  - 基本上，專案計劃與資源分配是不可分開，一個計畫須涵蓋工作、工作網路、時間與時程表等幾個項目。
  - 其次，預算是指完成專案所需資源的成本，它包含硬體、軟體、網路、訓練與其他發展費用。除此以外，風險評估亦是軟體專案計畫不可欠缺的重要項目，針對計劃中各項工作可能發生問題的機率與可能產生的影響作適當的預測與評估，並提出一些因應的措施。最後的活動就是將它述之於文字，以清晰的文字表示。

Integration Management of Software Project

- IEEE 軟體專案管理計畫書標準 IEEE1058

【軟體專案管理計畫書】

1. 導論
   - 1.1 專案摘要
     - 1.1.1 目的與範圍
     - 1.1.2 假設與限制條件
     - 1.1.3 專案交付項目
     - 1.1.4 期程與預算摘要
   - 1.2 專案管理計畫沿革
2. 參考規範
3. 定義

TSEG Tamkang University Software Engineering Group 淡江軟體工程實驗室 http://www.tkse.tku.edu.tw/

Integration Management of Software Project

- IEEE 軟體專案管理計畫書標準 IEEE1058

4. 專案組織
   - 4.1 對外溝通管道
   - 4.2 內部架構
   - 4.3 角色與職責
5. 管理流程規劃
   - 5.1 專案啟動規劃
     - 5.1.1 預估規劃
     - 5.1.2 人力規劃
     - 5.1.3 資源獲得規劃
     - 5.1.4 專案人員訓練規劃

Integration Management of Software Project

- IEEE 軟體專案管理計畫書標準 IEEE1058

5.2 工作規劃

5.2.1 工作項目

5.2.2 時程規劃

5.2.3 資源規劃

5.2.4 預算規劃

TSEG Tamkang University Software Engineering Group 淡江軟體工程實驗室 http://www.tkse.tku.edu.tw/

Integration Management of Software Project

- IEEE 軟體專案管理計畫書標準 IEEE1058

5.3 管制規劃

5.3.1 需求管制規劃

5.3.2 時程管制規劃

5.3.3 預算管制規劃

5.3.4 品質管制規劃

5.3.5 回報管制規劃

5.3.6 度量蒐集規劃

5.4 風險管理規劃

5.5 專案結案規劃

Integration Management of Software Project

- IEEE 軟體專案管理計畫書標準 IEEE1058

6. 技術流程規劃
   - 6.1 流程模式
   - 6.2 方法、工具與技術
   - 6.3 基礎建設規劃
   - 6.4 產品接收規劃

TSEG Tamkang University Software Engineering Group 淡江軟體工程實驗室 http://www.tkse.tku.edu.tw/

Integration Management of Software Project

- IEEE 軟體專案管理計畫書標準 IEEE1058

7. 支援流程規劃
   - 7.1 構型管理規劃
   - 7.2 確認與驗證規劃
   - 7.3 文件製作規劃
   - 7.4 品質保證規劃
   - 7.5 審查與稽核規劃
   - 7.6 問題解決規劃
   - 7.7 下包商管理規劃
   - 7.8 流程改善規劃
8. 附帶事項規劃

Integration Management of Software Project

- 專案監控
  - 專案監控的目的，提供對專案進展的了解，當專案執行重大偏離計畫時，能採取適當的矯正措施以因應。
  - 特定目標：
    - SG1依計畫監控專案：依專案監控實際執行績效與進度。
    - SG2 管理矯正措施直到結案：當專案執行績效或結果與原訂計畫產生重大偏離時，管理矯正措施直到結案。

TSEG Tamkang University Software Engineering Group 淡江軟體工程實驗室 http://www.tkse.tku.edu.tw/

Integration Management of Software Project

- SG1依計畫監控專案

SG 1	依計畫監控專案
SP 1.1	監控專案規劃之各項參數：依專案計畫監控專案參數之實際值。
SP 1.2	監控承諾事項：依專案計畫監控所界定的承諾。
SP 1.3	監控專案風險：依專案計畫監控所界定的風險。
SP 1.4	監控資料管理：監控專案資料管理。
SP 1.5	監控相關的關鍵人員的參與：依專案計畫監控相關的關鍵人員參與。
SP 1.6	進行進度審查：定期審查專案的進度、執行績效及問題。
SP 1.7	進行里程碑審查：進行專案里程碑審查。

Integration Management of Software Project

- 監控專案規劃之各項參數
  - 依時程表監控專案進度
    - 定期度量、比較、界定重要偏差
  - 監控專案成本與耗用的投入
  - 監控專案工作產品與任務之屬性
  - 監控專案資源的提供與使用
  - 監控專案人員的知識與技能

TSEG Tamkang University Software Engineering Group 淡江軟體工程實驗室 http://www.tkse.tku.edu.tw/

Integration Management of Software Project

- 監控承諾事項
  - 定期審查專案承諾
  - 界定尚未完成之承諾或具高風險之承諾
  - 文件化專案承諾審查結果

Integration Management of Software Project

- 監控專案風險
  - 在現行環境下定期審查風險
  - 有新增資訊時，應將變化納入並修訂風險文件
  - 將風險狀態通報有關人員

TSEG Tamkang University Software Engineering Group 淡江軟體工程實驗室 http://www.tkse.tku.edu.tw/

Integration Management of Software Project

- 監控資料管理
  - 定期審查資料管理動作
  - 界定重大問題及其影響並文件化
  - 文件化審查結果

Integration Management of Software Project

- 監控關鍵人員的參與
  - 定期審查關鍵人員參與情形
  - 界定重大問題及其影響，並予以文件化
  - 文件化參與情形審查結果

TSEG Tamkang University Software Engineering Group 淡江軟體工程實驗室 http://www.tkse.tku.edu.tw/

Integration Management of Software Project

- 進度審查
  - 定期通報相關人員指定活動與工作產品之狀況
  - 審查用以管制專案之分析與度量結果
  - 界定並文件化專案重大問題與偏差
  - 記錄所有工作產品及流程的變更需求及已知之問題
  - 追蹤需求變更及問題報告至結案為止

Integration Management of Software Project

- 里程碑查核
  - 選擇專案時程的重要環節進行審查
  - 審查專案的承諾、計畫、狀況及風險
  - 界定及文件化專案的重大問題及其影響
  - 文件化審查結果及行動決策
  - 追蹤行動項目到結案

TSEG Tamkang University Software Engineering Group 淡江軟體工程實驗室 http://www.tkse.tku.edu.tw/

Integration Management of Software Project

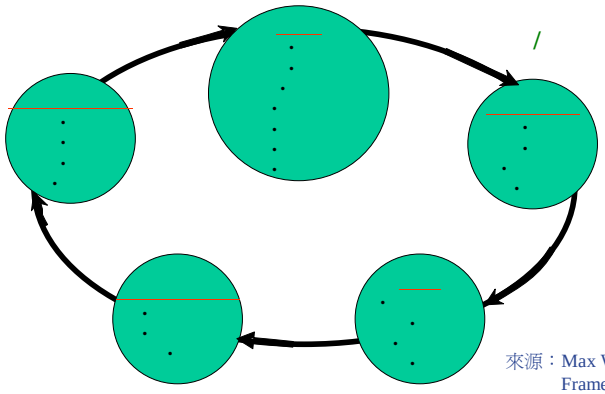
- SG2管理矯正措施

SG 2	管理矯正措施直到結案
SP 2.1	分析問題：蒐集、分析專案的問題以決定採取必要之矯正措施。
SP 2.2	採取矯正措施：對已界定之問題採取矯正措施。
SP 2.3	管理矯正措施：管理矯正措施直到結案。

TSEG Tamkang University Software Engineering Group 淡江軟體工程實驗室 http://www.tkse.tku.edu.tw/

Integration Management of Software Project

- 專案控制週期



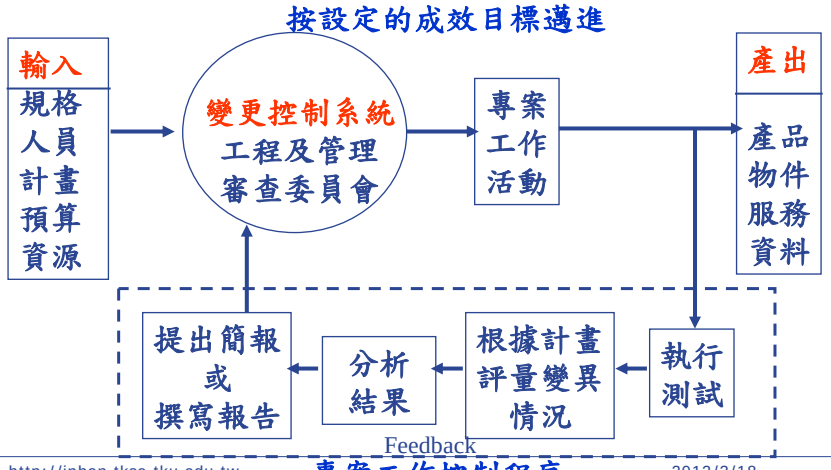
來源：Max Wideman, "A Framework for PM Integration"

規劃
需求
產品
標準
里程碑
成本預測
不確定性
採購策略
成功因素

TSEG Tamkang University Software Engineering Group 淡江軟體工程實驗室 http://www.tkse.tku.edu.tw/

Integration Management of Software Project

- 專案控制的路徑圖



專案工作控制程序

因素
物
質
控制

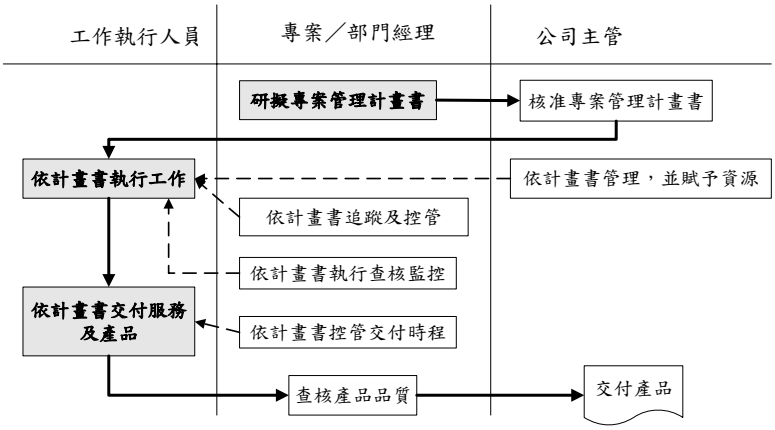
Integration Management of Software Project

- 成功控制專案的考量因素
  - 使用專案計畫書作爲監控專案執行的基準
  - 持續控制及更新專案計畫書與其他專案文件
    - SOW、需求規範 、功能規範、藍圖
  - 有效溝通是專案控制成敗的關鍵
  - 全面參與、積極投入整體控制作業
  - 掌握專案時程、成本與品質
  - 紀錄專案進度、各項更動並告知所有專案成員

TSEG Tamkang University Software Engineering Group 淡江軟體工程實驗室 http://www.tkse.tku.edu.tw/

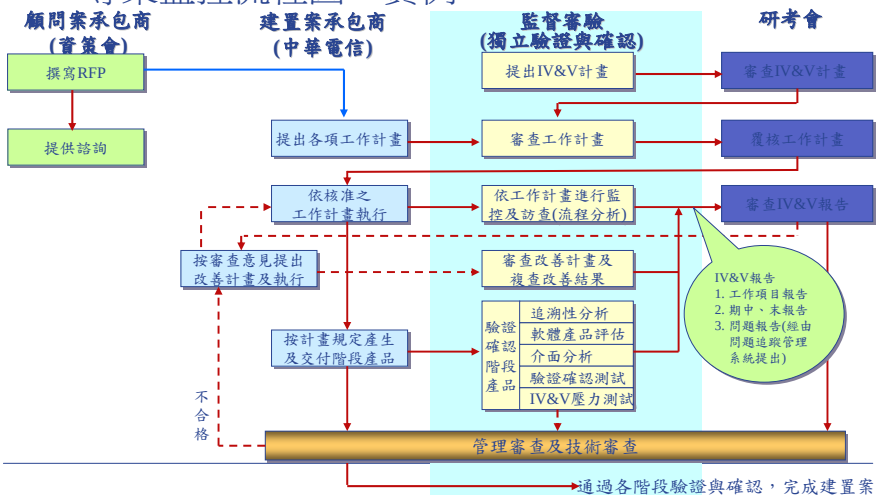
Integration Management of Software Project

- 專案監控流程圖



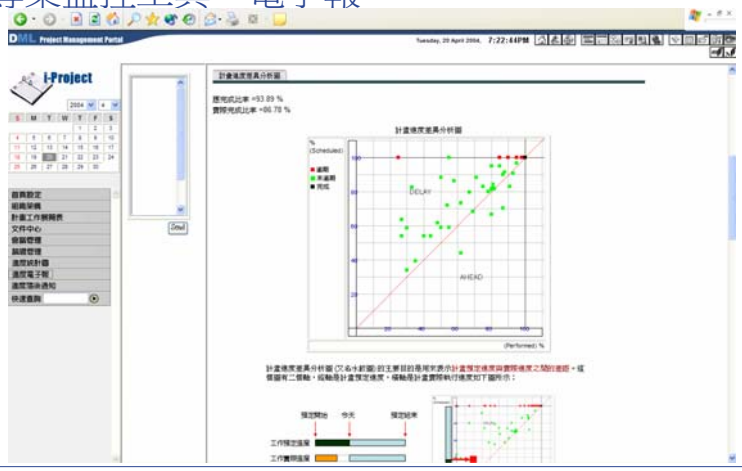
Integration Management of Software Project

- 專案監控流程圖 – 實例



Integration Management of Software Project

- 專案監控工具—電子報

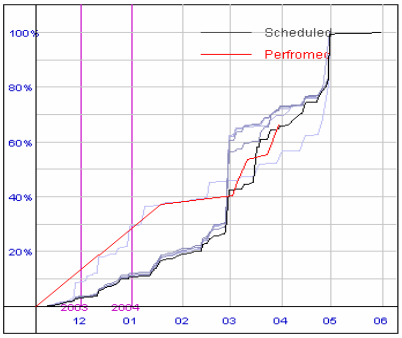


Integration Management of Software Project

- 專案監控工具—專案進度趨勢分析圖

計畫進度趨勢分析圖

計畫進度趨勢分析圖



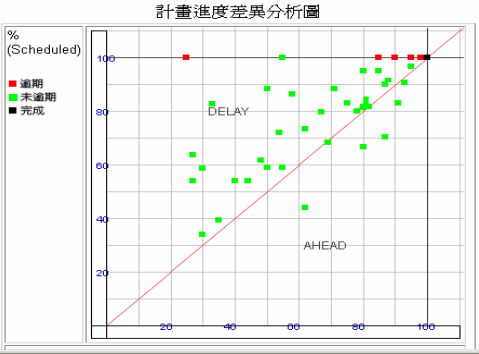
Integration Management of Software Project

- 專案監控工具—水餃圖(俗稱)

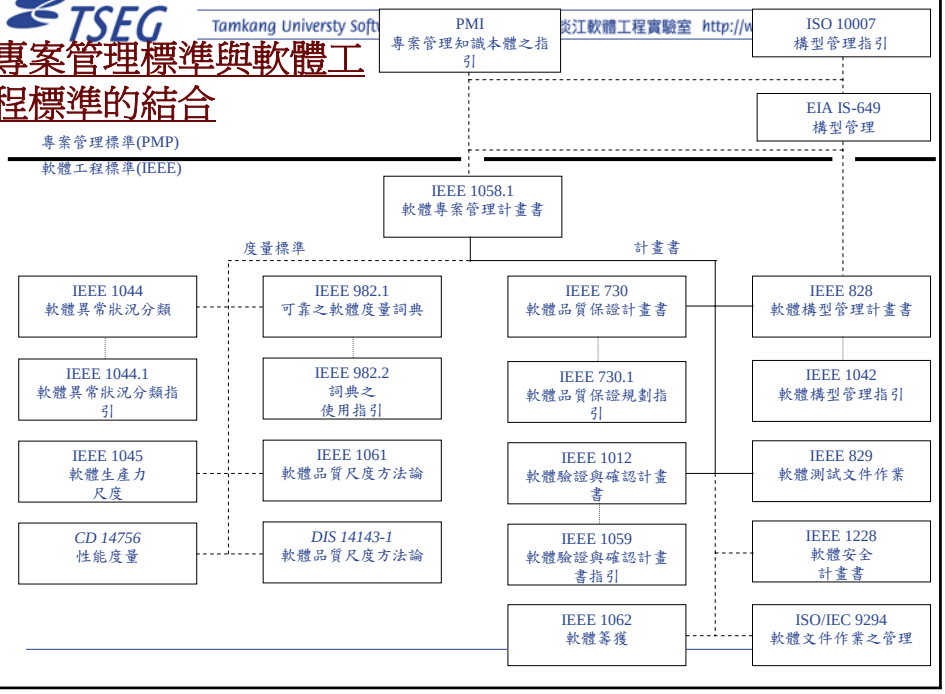
計畫進度差異分析圖

應完成比率 =84.15 %
實際完成比率 =82.25 %

計畫進度差異分析圖



專案管理標準與軟體工程標準的結合



Integration Management of Software Project

- 小結
  - 軟體專案規劃工作建議從以下幾個構面進行分析：
    - 技術構面—選擇與採用軟體開發模式、方法論及工具等。
    - 資源構面—人力、物力與時間的估計與安排。
    - 資訊構面—開發人員彼此間的資訊傳遞方式、工具。
    - 管理構面—時程、風險與人力資源的管理模式。
  - 專案規劃涉及的因素非常廣泛，加上外在環境的變動，規劃工作並非只是一次就能完成，它必須經由不斷且漸進的修正過程才能臻於完美的境界。

軟體成本估計

- 軟體成本估計存在著下列的問題：
  - 低估了軟體開發及導入的成本。
  - 影響軟體成本的因素很多，精確的估算並不容易。
  - 不同的估算法所得結果差異非常大。
  - 低價搶標策略使得軟體成本的問題更加嚴重。

時程規劃

- 時程規劃就是專案開發的「時間管理」。
- 除了運用傳統專案管理技術如PERT/CPM外，同時也應考量軟體專案的特性，適度地加入緩衝時間；對於超過負荷的工作分派加以平滑，如果需要壓縮時程時，則應選擇適用的方法，並且考量其限制與可能的影響；當專案無法在規劃時間內完成時，則應調整時程規劃。