

序

本書架構共分成五篇（共有 15 章）。前 12 章為傳統之系統分析與設計之方法論；自第 13 章開始為目前資訊通訊科技（Internet Communication Technology，ICT）的重要相關應用領域。第 13 章之供應鏈管理、第 14 章之企業資源規劃、第 15 章之知識經濟 - 資料倉儲、資料採礦、商業智慧、海量資料，均為在校同學畢業後，極有可能在以上的應用領域中，去執行系統分析與設計之專業技能。

本書之主旨在於使用簡單、清晰以及容易了解的專業術語，來協助同學們學習系統分析與設計的基本技術，儘量避免使用艱澀的措辭，造成同仁們學習上的障礙與阻撓。本書以實務導向，減少純理論而無法在業界活用的冗言長篇大論。降低純理論造成學生空泛知識，在企業界無法使用之窘態。

在本書的架構當中，我們使用了非常豐富的實際案例，讓同學們充份了解。而系統分析與設計這門課，其實是可以非常容易和實務作結合的。在本書當中，也提供了大量企業界中所使用的表格、流程、同時也引用實際公開招標的案件，來當作學習的活教材，我們希望系統分析與設計這門課程所傳授的技術，可以讓同學們輕易地使用到實務界應用上，這是我們最大的心願。

以我們的實務經驗，很多剛畢業的同學們，在公司上班的時候，都是著重於最新技術的鑽研，卻忽略了一個專案的開發與執行，是需要清楚和明確的系統分析與設計的專業知識，來作為執行專案的最佳穩固基礎。

對於本書之出版，在此也感謝碁峰資訊公司之全力協助，以及各大專院校教授長久之支持與推薦，在此深表謝意。

最後要感謝的是國立臺中教育大學，提供本人完善之教學、研究環境。本人身為國立臺中教育大學研發長兼國際暨兩岸事務長，我以國立臺中教育大學百年名校為榮。

朱海成

2014 年秋 於台灣台中市國立臺中教育大學管理學院

資訊系統發展模式

本章學習重點

- 資訊系統發展模式概論
- 瀑布(Waterfall)式之資訊系統發展模式
- 漸增(Incremental)式之資訊系統發展模式
- 雛型(Prototype)式之資訊系統發展模式

3-1 資訊系統發展模式概論

資訊系統發展模式是系統發展專案過程中，十分重要的步驟及執行規範。當系統發展專案依循邏輯化、系統化之方式運作時，整個系統發展專案就較容易進行管理，開發的過程將會更具有效能與效率。相對地，開發出的新系統專案之容錯程度與品質控管，也將大幅提昇。資訊系統發展模式已經有很多專家學者，在過去五十多年中，陸續地提出，以期適用於不同情況的系統發展專案，目標都是希望藉由較適合的發展模式，以期能以更系統化的全面考量分析，運用更適當之設計技術，使用更短之時間與更少之資源投入，來完成系統之構建，根據這些相關學者所提出的精闢見解，本書將介紹一些較具代表性之資訊系統發展模式，提供同學們學習上的參考。

3-2 瀑布(Waterfall)式之資訊系統發展模式

約在 1970 年代，有專家學者提出此種資訊系統發展模式。瀑布式資訊系統發展模式之運作是將系統發展專案分成數個先行定義好之階段，並且詳細指出每階段所必須達成之工作目標，各階段僅進行一次之迴圈運作，並循序執行之前先行定義好之各階段。而究竟要區分成多少之定義好之階段呢？則要視資訊系統發展專案之規模與複雜程度。當系統發展專案規模較小時，甚至只需要區分成三個階段，如系統分析、系統設計、系統建置等階段，就可以輕易的掌握整個新系統專案的建置情況。相對地，如果系統發展專案規模較大時，之前的系統分析、系統設計、系統建置就可能再被細分成更多個階段，例如：將分析階段再切割成為系統需求分析(System Requirement Analysis)、系統分析(System Analysis)、可行性分析(Feasibility Analysis)，將設計析階段再切割成為概括性設計(General Design)、細部性設計(Detail Design)，將建置階段再切割成為程式模組(Module)撰寫與測試(Testing)、子系統整合與測試(Subsystem Integration and Testing)、最終系統整合與測試、系統安裝(System Installation)與障礙排除(Trouble Shooting)、使用手冊(User Manual)與教育訓練(Training)、系統使用者目饋(Feedback)與系統修正(System Modification)、系統維護(System Maintenance)，等十二個階段，如圖 3-1 所示，即為瀑布式資訊系統發展模式，該圖顯而易見，像是一座瀑布的形狀，以 cascading 方式呈現。若瀑布式資訊系統發展模式，在各階段察覺到錯誤，即應儘量早以修正，以期減少修改系統或重新再做之額外成本。此外，在各階段已明確定義，且應做之工作或必須交付的文件，則務必履行，如此，才可使系統發展專案之開發，更明確也更容易掌握。

瀑布式資訊系統發展模式是一種相當符合邏輯的運作方式，它依系統發展生命週期(Software Development Life Cycle, SDLC)來進行規劃，每一階段的結束，就可作為系統發展專案在管理上之一個里程碑(Milestone)。它在執行運作上，每一階段的負責人員承擔明確的責任，所有參與系統發展專案之相關人員，必須正確而完整地將該階段之輸出，正式移交給下一階段的人員，而後一階段的相關人員在接收前一階段的輸出時，也必須有簽收確認之程序。如此一來，系統發展專案的開發過程，將會更具結構化特性，且在系統發展專案之管理層面上，也比較容易掌控。在本書中的後面章節，就有專章來介紹專案管理(Project Management)的重要性與相關的概念，對於專案管理有興趣的同學，可以現在就跳到本書的後面章節，直接了解專案管理的重要性。

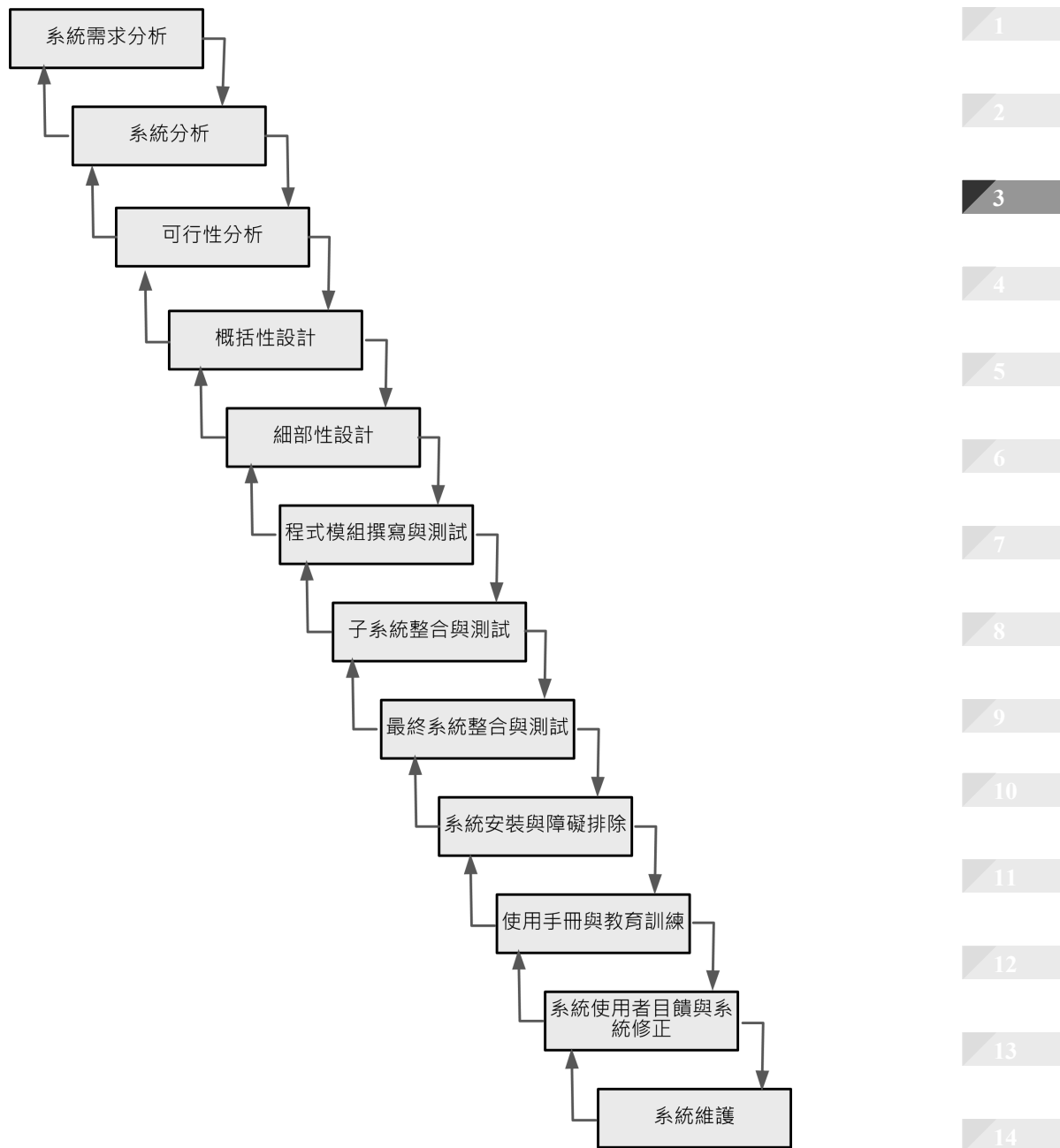


圖 3-1 為瀑布式資訊系統發展模式，該圖顯而易見，像是一座瀑布的形狀，以 cascading 方式呈現

- 在實際運作上，瀑布式資訊系統發展模式應用在系統發展專案時，每一個階段必須仔細考量出完整的功能需求，而且前一階段必須完全滿足系統需求，才可進入下一階段。如此累積下來，在品質控管完整的情況下，整個系統才被建置出來。在執行過程中，每一階段的輸出，必須通過相關人員

之審核，以確保軟體品質(Software Quality)，只要該階段審核通過，則該階段所輸出的相關系統文件將被凍結(Freeze)，如果有異動的需求狀況，則相關人員必須要經過正式的申請程序，並獲得批准，才能更改，如此才能避免任意之異動，將造成系統規格(System Specification)版本不同，前後不一致(Non-consistency)的窘境。此種模式將重點放在系統發展專案過程中，需有完整的分析、規劃、設計、測試及文件等控管機制，因此才能有效的確保系統軟體品質，在業界，很多承接系統發展專案之公司，都以瀑布式資訊系統發展模式為軟體開發之標準。從實際的經驗告訴我們，瀑布式資訊系統發展模式，在面臨以下情況時，在運作上，可能會遭遇一些問題：

- 如果系統發展專案之開發週期較長，一般而言，使用者在需求分析階段，均會樂於參與，但若因時間、地點無法長時間配合，且在爾後此過程中，使用者因故而無法踴躍參與，而到了後期之系統安裝階段才多花時間參與，此時，系統修正之成本也就相對地提高許多。
- 在著手進行程式撰寫之前，有時系統開發過於重視完整的分析與相關設計文件，因此，一旦系統需求異動，相關文件就必需大幅隨之修改，進度就非常有可能延後。
- 如果程式之撰寫，在系統開發週期之後段才開始，如此會有較高之風險，例如系統使用者需求異動或所欲使用之資訊科技已過時(Phase Out)，此時異動所造成之成本額外負擔(Overhead)，有時是相當高的，建置團隊也必須要承擔這樣的額外負擔。

我們不難了解，在面臨以上情況時，如果在系統發展專案開始時，系統需求(System Requirement)仍遲遲無法清楚描述時，瀑布式資訊系統發展模式就無法適用。因此，以下漸增(Incremental)式資訊系統發展模式與雛型(Prototyping)式資訊系統發展模式，就可以解決上述情況，以進行系統開發。

3-3 漸增(Incremental)式之資訊系統發展模式

如上所述，為了彌補瀑布式資訊系統發展模式在先天上的缺失，在 1970 年左右，有相關專家學者提出漸增式資訊系統發展模式。它也是一種系統發展專案開發方法論，原則上，該方法論會將系統需求切割成幾個部分，每一部分之內，再切割為細部系統需求，並定義該部分為一系統發展專案週期，依循上述方法，

系統需求也就進一步被切割成幾個系統發展專案週期。每個週期，可依序(Serial)或平行(Parallel)來進行系統發展專案開發，在每個週期之階段，均明確定義出有那些系統功能與目標必須要達成，有那些相關之文件，必須要交付，在每一個階段內依序進行，而且在該階段內，僅反覆(Iteration)一次，如圖 3-2 所示，即為漸增式資訊系統發展模式之運作示意圖。

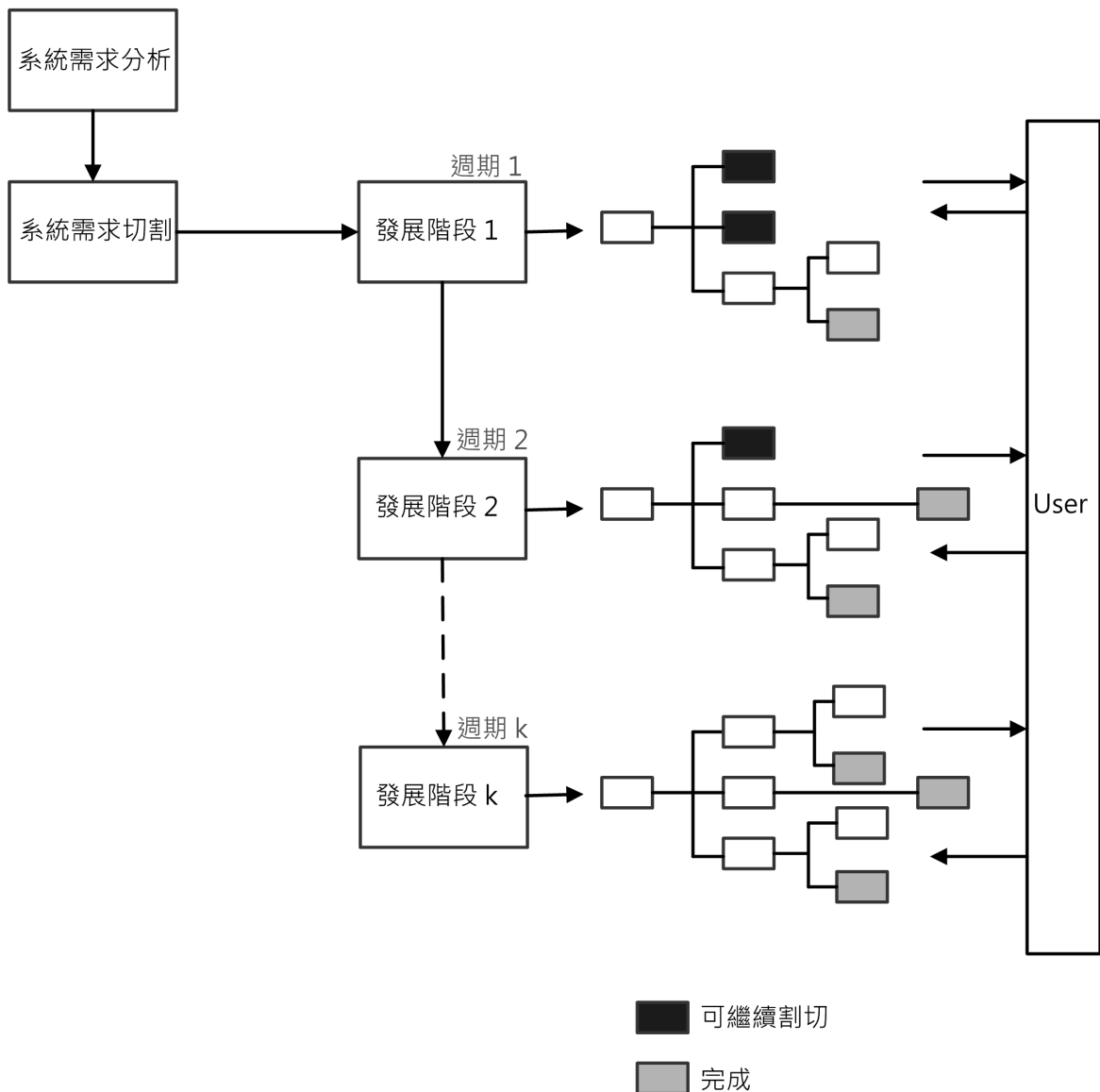


圖 3-2 為漸增式資訊系統發展模式之運作示意圖

漸增式資訊系統發展模式可視為是瀑布式資訊系統發展模式之擴充，該模式之重點在於系統需求可被分切割成數個部分，且各部分，可再依瀑布式資訊系統發展模式，繼續進行。換言之，漸增式資訊系統發展模式必需先仔細進行系統需求分析，並要十分確認系統需求，並進行系統需求切割。在漸增式資訊系統發展模式中，對系統發展專案開發之系統分析與系統設計採用由上而下(Top Down)之模式，將系統需求切割成幾個部分，每一部分之內，再切割為細部系統需求，按照以上計畫持續進行，最終就可完成系統之開發。

漸增式資訊系統發展模式和瀑布式資訊系統發展模式很類似，但以下情況，兩者就不太相同：

- 漸增式資訊系統發展模式中，將系統需求切割成幾個部分，每一部分之內，再切割為細部系統需求，各細部系統可獨立依序開發，等到該細部系統上線測試成功後，再進行另一系統之開發週期，可參考圖 3-2 之示意圖，而瀑布式資訊系統發展模式，在系統建置階段，事先切割定義好之模組，需同時進行開發。
- 漸增式資訊系統發展模式中，由數個週期循環完成，每個週期，象徵不同之系統建置版本，而重點在於每個週期中，系統使用者均會參與使用並提供回饋(Feedback)，在此情況下，系統開發所冒之風險會較低，因為在每個週期中，使用者寶貴的回饋資訊，可以讓一些隱藏的問題，早點浮上檯面，並尋求因應之方法，這一個切入點，在風險管理之觀點上，相對佔優勢。
- 在很多實際案例中，預算採取分期(每年或每季)編列之方式，在此情形下，運用漸增式資訊系統發展模式更佳，原因是經由先期之預算，可先將系統發展專案作整體性之規劃，再分期(每年或每月)執行系統發展專案，就算在後期沒有預算之奧援，先前開發完成的部分，其系統功能依然可以執行，如此，組織企業避免一次投入大筆金額，可以較保守性地進行系統發展專案，以期降低組織企業之財務負擔，執行較好之風險管理。

3-4 雛型(Prototype)式之資訊系統發展模式

瀑布式資訊系統發展模式與漸增式資訊系統發展模式，一般而言，均需在系統發展專案著手時，使用者可以提供明確與清楚之系統需求描述。然而，在台灣

之中小企業在進行電子商務或企業 e 化時，專案承辦人往往只是承受上級之指示，至於系統需求之細節，常常是無法提供明確與清楚之系統需求描述，在此情況下，正因使用者本身都無法將系統需求清楚且完整的呈現，所以更不用奢望相關資訊人員能憑空解決問題，就算是經驗老到的系統分析師，也可能因不明瞭使用者之系統需求，而壯志難伸。雛型(Prototype)式資訊系統發展模式是一種系統開發方法，它是先將系統使用者需求較明確的部分，或是資訊人員較有把握的部分，按照系統分析、系統設計、系統建置等大方向步驟，在短時程內，快速地開發出一個簡單的系統雛型(System Prototype)，再以此雛型朝向最終系統邁進。在電子商務實作環境中，我們舉實例來說明雛型式之資訊系統發展模式：

實際案例

大榮鞋業股份有限公司是家傳統中小企業，為了與海外協力廠商配合，以留住國際大廠 OEM 之訂單，加入了該產業的供應鏈管理系統(Supply Chain Management)體系，總裁下令由承辦人銜命執行，但是專案承辦人只是承受上級之指示，系統需求之細節，無法提供明確之系統需求描述，只是希望該系統與國際廠商連線後，資料庫欄位之名稱可藉由使用者介面，隨時均可自行更改，而輸出格式初步分析，大約有 300 多種，但也必須藉由使用者介面，輕易地改成國際廠商所要求之格式，承辦人不希望程式經常性地修改，希望藉由良好而完善之使用者介面，來達成不必日後，一直麻煩系統建置廠商。

系統分析師在面臨以上問題時，坦白說，承辦人及相關承辦人員之系統需求所做之描述算是清楚，而系統開發建置人員，也算是清楚的瞭解使用者之系統需求，但是，系統分析師對於 SCM 中，具有高度彈性之資料庫與和國際廠商所要求之格式，並不確定要使用那些資訊科技來達成該資訊系統，短時間內，可能找不出適合之設計模式方法或相關可運用之開發工具派上用場。因此，有專家學者，大約在 1977 年左右，提出雛型(Prototype)式資訊系統發展模式，來解決上述問題，因為在系統需求無法提供明確與清楚之系統需求描述時，瀑布式資訊系統發展模式和漸增式資訊系統發展模式均不適用。

雛型式資訊系統發展模式是新系統專案發展時所使用的一種系統開發方法，它是先將系統使用者需求較明確的部分，或是資訊人員較有把握的部分，按照系統分析、系統設計、系統建置等大方向步驟，在短時程內，快速地開發出一個簡單的系統雛型(System Prototype)，並以該雛型當作系統使用者或專案承辦人與系統分析師、程式設計師溝通之橋樑，兩造雙方藉由系統雛型之運作，使用

者或專案承辦人提供回饋資訊，系統分析師藉以釐清、變更、延伸系統需求，同時不斷地修改系統雛型，直到期望之系統功能與目標，符合專案承辦人之理念為止。由以上可知，雛型式資訊系統發展模式適合運用在新系統專案之專業領域並不熟悉、具高風險等之情況的新系統發展專案，尤其適用在系統發展專案開發時，系統需求很可能改變，或系統需求較大且複雜的情況下，而且在新系統發展專案開發時程中，系統使用者可以高度參與。相對的，如果新系統發展專案已具有明確且不易更動之系統需求，或在應用資訊科技之技術上，沒有易見之困難，則不需使用雛型式資訊系統發展模式。

由於雛型式資訊系統發展模式並不強調以系統文件(System Document)當作系統使用者與資訊人員溝通之主要媒介，相對的，而是以系統雛型作為兩造雙方溝通之工具，而兩造雙方在不斷演進之系統雛型中，藉由使系統用者之高度參與，可以不斷修正新系統發展專案方向，共同執行系統雛型之開發、運作與檢討，並藉由系統使用者不斷地提供回饋，將系統雛型不斷地進行修正。而最終之系統，在客觀角度上，使用者對最終系統之滿意度與接受度，相對地也大大提高，有助於新系統發展專案之驗收與簽結。

然而，雛型式資訊系統發展模式並不是沒有缺點，承如上述，此系統發展模式並不強調以系統文件當作系統使用者與資訊人員溝通之主要媒介，因此，系統文件可能較不完整，在日後，如果系統團隊有人員異動或要進行維護、更新，都可能因為系統文件不完整，或原系統開發工程師離職，而造成維護困難度提昇，長期來看，甚至導致系統失敗之隱憂。

另一方面，此系統發展模式以快速建構系統雛型為目標，因此，較缺乏整體性宏觀式之分析、規劃、設計，如此一來，很明顯的，雛型式資訊系統發展模式就比較不適用於較大型系之系統發展專案。如圖 3-3 所示，即為雛型式資訊系統發展模式之運作示意圖。

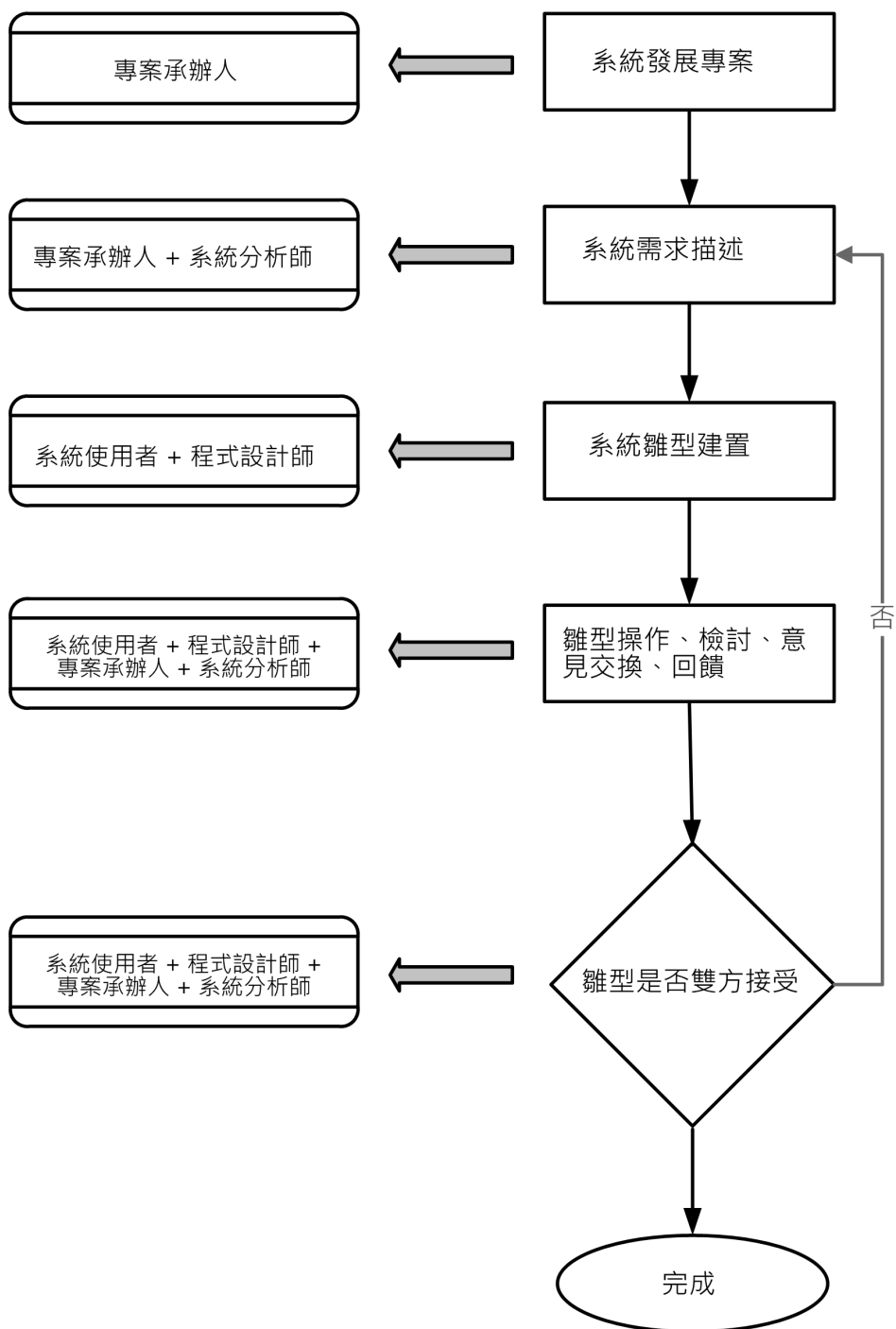


圖 3-3 為雛型式資訊系統發展模式之運作示意圖

除了以上所介紹的資訊系統發展模式之外，當然還有一些學者提出其它相關的研究方法論，來提升資訊系統發展模式的效益，而其目標，都是希望能夠使用一個比較適切的方法論，針對新系統專案的開發，有所助益，而我們目前就先向各位同學們介紹以上的幾種方法，希望以上所介紹的幾種資訊系統發展模式，在各位同學們將來就業的時候，有多方面的幫助。



學習評量

一、問答題

1. 何謂瀑布(Waterfall)式之資訊系統發展模式？
2. 何謂漸增(Incremental)式之資訊系統發展模式？
3. 何謂雛型(Prototype)式之資訊系統發展模式？

二、選擇題

1. () 瀑布式資訊系統發展模式之運作是將系統發展專案分成數個先行定義好之階段，並且詳細指出每階段所必須達成之工作目標，就可以輕易的掌握整個新系統專案的建置情況，以下何者無助於前開之敘述？
 - (a) 系統整合
 - (b) 系統分析
 - (c) 系統設計
 - (d) 系統建置
2. () 將分析階段再切割成為三個子分析系統，以下何者不應屬於前開之敘述？
 - (a) 系統需求分析(System Requirement Analysis)
 - (b) 系統整合分析(System Integration Analysis)
 - (c) 系統分析(System Analysis)
 - (d) 可行性分析(Feasibility Analysis)