

第一章 緒論

1.1 研究背景

隨著時代巨輪的演進,人類社會文明自 1991 年網際網路從學術領域轉為商業應用後,正式帶領全世界進入一個資訊傳遞超越時間、空間限制的第四波網路社會【1】。網路時代的來臨不僅突破不同國界、文化、語言之屏障,其無遠弗屆與資訊同步的特性,更徹底顛覆傳統思維、加速全球化競爭,以及企業數位電子化的趨勢。如今企業面臨前所未有的急劇轉變,例如:市場經濟自由化、產品生命週期縮短、客戶需求變化快速、風險管理不易、企業跨國合作經營等,均使得管理與決策的工作日益複雜【2】,讓傳統企業無法憑藉過去的經驗判斷與經營模式,迅速掌握商機、因應環境變化。

近年來企業間電子交易環境日益成熟,加上大陸、東南亞等低勞資條件國家的強烈競爭,企業價值鏈體系(Value Chain【3】)勢必受到嚴重衝擊,進而產生解構和重組的效應;國內企業也將從傳統勞力密集的 OEM (Original Equipment Manufacturer),逐漸轉變為知識密集的 ODM (Original Design and Manufacturer)、OBM (Original Brand Manufacturer)產業型態【4】,而轉型成敗的主要關鍵,則在於企業能否有效提升設計研發的實力,以及整合產品開發過程中的所有資源。

為追求產業升級效益、克服現今零阻力經濟環境的衝擊【5】,企業必須強化管理效能,整合散落於產品生命週期各階段中的資料,並將資訊擷取轉化為重要的營運知識,藉以提升整體競爭實力,創造資訊的價值。為此,企業不斷導入各式 e 化解決方案,大量採用網路資訊技術,如火如荼地針對各式內外流程,進行產品資訊電子化、網路應用化以及系統整合的工作。

然而資料交換標準、訊息傳遞協定,以及網路應用與分散式系統架構等理論技術,直至近年才逐漸發展成熟,加上國內中小企業資訊能力普遍不足、e 化觀念策略錯誤,因此當其導入整合性資訊管理系統時,往往採用傳統系統開發模式,或者盲目追求系統功能,導致企業無法有效彈性運用系統功能,使得系統整合與後續維護管理的工作極為艱難,不僅耗費大量投資成本,更因資訊化孤島(Island Of Information)的產生,反而成為企業 e 化沉重的負擔,此為企業產品 e 化整合失敗的主要原因之一【6】。

由於過去國內產業型態主要以製造代工為主，設計研發工作並不受重視，因此企業持續改善與 e 化的目標，往往偏重於製造、財會、採購與銷售領域等。然而根據 Iansiti 與 MacCormack【7】以及美國 Aberdeen Group 專業顧問公司的研究報告顯示，約有 80% 的產品開發成本，決定於製造生產前的流程【8】。這意味著無論企業於後續的供應鏈管理 (Supply Chain Management ; SCM)、企業資源規劃 (Enterprise Resources Planning ; ERP) 等方面再如何投資改善，至多也只能降低約 20% 的生產成本。反之，倘若在設計研發階段即能改善流程效率，兼顧上下游廠商與市場環境的實際需求，則將會比企業在採購、製造以及行銷等方面所作的努力，節省更多的營運成本、創造更大的利潤空間。

近年來協同產品商務 (Collaborative Product Commerce ; CPC) 的興起，提供給企業一個嶄新的管理思維，以及整合產品開發生命週期所有資訊的解決方案，如圖 1.1 所示【17】。它以設計研發為中心，利用網際網路與資訊技術，垂直整合產業價值鏈上的所有廠商，進行產品的協同設計，使產業間以核心能力互相連結，不僅改善了整體產品開發流程效率，更強化企業與客戶、上下游廠商之間的合作關係，藉由彼此分享營運知識經驗，發揮協同創新的最大效益【9】。因此，企業縱使在今日詭譎多變的競爭環境中，依然能夠形成一完整的虛擬企業，以前所未有的設計研發速度，開創一無限可能之新紀元。

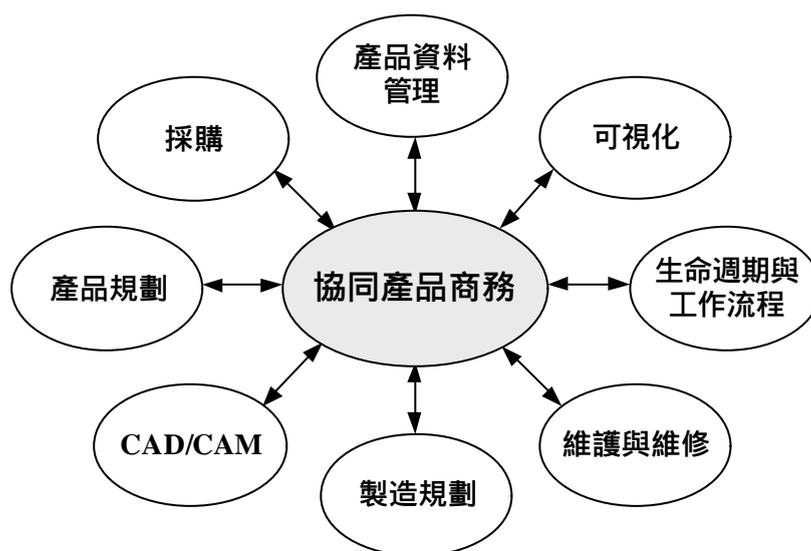


圖 1-1 產品協同商務功能關聯圖

1.2 研究動機

為有效解決產品資訊無法整合運用、降低 e 化風險與成本，企業應先溯本清源，探究 CPC 的本質，以及眾多 e 化方案的主要核心為何。其中，產品資料管理系統 (Product Data Management System ; PDMS) 所提供之結構與非結構化文件管理，以及權限控管等功能，對企業發展電子商務 (Electronic Commerce ; EC)、協同產品商務等，均扮演不可或缺的角色【10】。PDMS 不僅是製造企業 e 化的重要基礎，更可將其視為 CPC 與 ERP 的主要核心【11】。此外，PDMS 亦提供產品知識管理 (Product Knowledge Management ; PKM)、客戶關係管理 (Customer Relationship Management ; CRM)，以及供應鏈管理 (Supply Chain Management ; SCM) 等，一個產品資料來源與交換的媒介。

現今主流的 PDMS 軟體是以元件式【12, 13】Web-Based 系統架構為主，例如：PTC 參數科技的 Windchill、美國 Unigraphics 公司的 iMAN 等，由於其運用網路應用程式，開發系統功能與前端操作介面，因此使用者可利用瀏覽器(Browser)，透過 Internet 登入至 Web-Based PDMS 系統，進行遠端管理與產品資料查詢的工作。然而，雖然 Web-Based PDMS 已提供一完整平台，能夠讓企業夥伴以及上下游廠商，透過 Internet 登入取得與交換產品資訊，但絕大部分的工作依然需要人工來處理產品資訊的篩選與轉換，譬如：客戶欲查詢公司的產品資料，其必須先輸入帳號密碼登入至 Web-Based PDMS 平台，然後再從眾多功能選單中逐步操作點選，經過繁複的程序才能取得相關的產品資料項目，並且無法自行設定與運用產品資訊的內容。此外，由於各家 Web-Based PDMS 廠商採用特定元件模型技術 (如：COM+、EJB、COBRA 等)，以及不同的通訊協定標準，致使 PDMS 無法輕易地與其他企業內外異質系統相互整合運作，往往必須透過昂貴的橋接技術 (Bridging Technology【14】) 加以處理，才能達到自動化處理與跨平台整合的目標。歸納現今 Web-Based PDMS 系統架構與運作模式的主要缺點如下【15】：

1. 必須使用的橋接技術或者解譯轉換系統，才能與其他異質資訊系統相互運作。
2. 系統模組功能擴充與調整缺乏彈性，侷限於特定作業系統、程式語言與開發工具。
3. 難以滿足 B2B 電子商務與 CPC，對於產品資訊整合運用以及自動化處理之需求。
4. 未使用標準的軟體元件技術與資訊傳遞協定，致使系統元件功能無法再用，增加系統導入與整合的成本負擔。
5. 產品資訊難以跨越不同執行平台以及各式前端硬體裝置。

由上之論述可知，現今元件式 Web-Based PDMS 對於產品資訊的使用彈性、自動化處理，以及跨異質平台系統能力等，事實上並無法完全滿足現今與未來電子化商務環境需求【16】。產品資料管理系統架構的良窳是企業 e 化與協同產品商務導入成敗的主要關鍵，因此如何克服上述 Web-Based PDMS 的主要缺點，發展一具備高度彈性、整合性、跨平台，以及自動化處理能力的產品資料管理系統，滿足各式產品資訊服務需求，是現今企業 e 化系統整合應用的發展趨勢與重點，也是引發本文研究「Web Service-Based PDMS」的主要動機所在。

1.3 研究目的

企業在知識經濟的時代裡，必須整合所有營運資源並且有效管理，才能創造最大的商業利潤與企業競爭實力。產品資料管理系統是眾多 e 化解決方案的重要基礎，因此本研究將以元件模型技術 (Component Model Technology)、分散式系統 (Distributed System)、網路服務 (Web Service)等理論標準作為應用基礎，提出一符合協同產品商務內涵【17】的 Web Service-Based PDMS 理論架構與開發模式，其主要目的除了探討 PDMS 何以對國內企業轉型為 ODM/OBM 如此重要之外，另一方面則希望能夠利用元件模型與網路服務的優勢，發展出一具備跨平台、跨語言，易於彈性使用的產品資料管理系統，以有效地解決資訊化孤島 (Island Of Information)之問題，進而降低企業導入 e 化，以及產品資訊整合的阻礙與成本，期能提供企業與學術作一參考與改良的依據。為達到研究目的，本研究將完成以下三項研究目標：

1. 比較分析 Web-Based 系統與 Web Service-Based 系統的主要差異：

此目標主要探究 Web-Based 系統的整體架構，並論述 Web Service-Based 系統如何運用 Web Service 標準，克服 Web-Based 系統缺點，以及減少系統整合的困難度。

2. 提出元件式 Web Service-Based PDMS 理論架構與開發模式：

論述 Web Service-Based PDMS 如何滿足企業 e 化與商務環境需求，應採用何種系統架構、標準協定與開發模式，才能使 Web Service-Based PDMS 發揮整合效益。

3. 深入訪談企業及實作 Web Service-Based PDMS：

藉由訪談了解企業現況與未來發展方向，以及於設計研發管理上之問題及需求，以作為後續系統功能開發的基礎，進而運用 Web Service-Based PDMS 理論架構與開發模式，將 PDM 系統予以元件化、網路服務化，提供企業一實作參考案例。

1.4 研究方法與流程

本研究整體內容分為三大階段，各階段所應用之方法與步驟如下表 1-1 所示：

表 1-1 本研究各階段所應用之方法與步驟

階段	主要研究內容	方法應用與步驟
一	<ul style="list-style-type: none"> ● 企業經營與 e 化發展 ● 產品開發模式的改變 ● 產品資料管理系統的演進 ● 資訊系統架構與開發模式之發展 ● Web Service 	<p>透過文獻的整理與探討，了解現今企業經營環境所面臨的衝擊與發展趨勢，而企業的管理目標與產品開發模式變化為何。接續則探究協同產品商務的內涵及 PDMS 的演進過程，企業又應以何種資訊系統架構與開發模式發展 PDMS，以及 Web Service 觀念、技術、標準何以是下一波資訊革命的主軸。</p>
二	<ul style="list-style-type: none"> ● Web-Based 與 Web Service Based System 的主要差異 ● Web Service-Based PDM 理論架構與開發模式 	<p>此階段根據上述之文獻探討，以各種角度切入論述 Web-Based 與 Web Service-Based System 於功能特性、系統架構上的差異為何。後續則據此配合元件模型技術、分散式系統架構，以及 CPC 環境下系統整合之觀念，推論與建構一具備彈性、易於整合的 Web Service-Based PDMS 理論架構與開發模式。</p>
三	<ul style="list-style-type: none"> ● 企業深入訪談 ● 企業產品資訊整合現況分析 ● 系統實作步驟： <ol style="list-style-type: none"> (1)整理設計管理問題及需求 (2)設計規劃 PDMS 模組功能 (3)元件化 PDMS 模組功能 (4)網路服務化元件模組功能 	<p>以深入訪談的方式，了解企業 PDMS 與產品資訊整合現況，並整理設計研發管理問題作為系統開發基礎，而後則選擇一適當企業，使用其產品相關資料與導入本研究理論架構與開發模式，實際建置 Web Service-Based PDMS。系統開發主要方法技術如下：</p> <p>系統分析與設計工具：UML【18，19】 元件模型標準 / 主要開發工具：COM+ / Delphi6 作業平台 / 網頁伺服器：Windows 2000 / IIS5 後端產品資料庫管理系統：MS SQL 2000 Server 前端使用者介面設計工具：Dreamweaver 後端動態伺服器語言：ASP XML 文件物件模型：MS XMLDOM</p>

綜合整理表 1-1 各階段之主要研究內容與方法，本文整體架構共分為六大章節，各章節內容與研究流程如圖 1-2 所示。由於本研究屬實務性之論題，因此除了提出基本理論架構外，亦深入訪談 10 家企業了解現況，而後實際開發 Web Service-Based PDMS，最終則整理歸納研究成果與後續建議事項，提供各界應用與深入研究之參考。

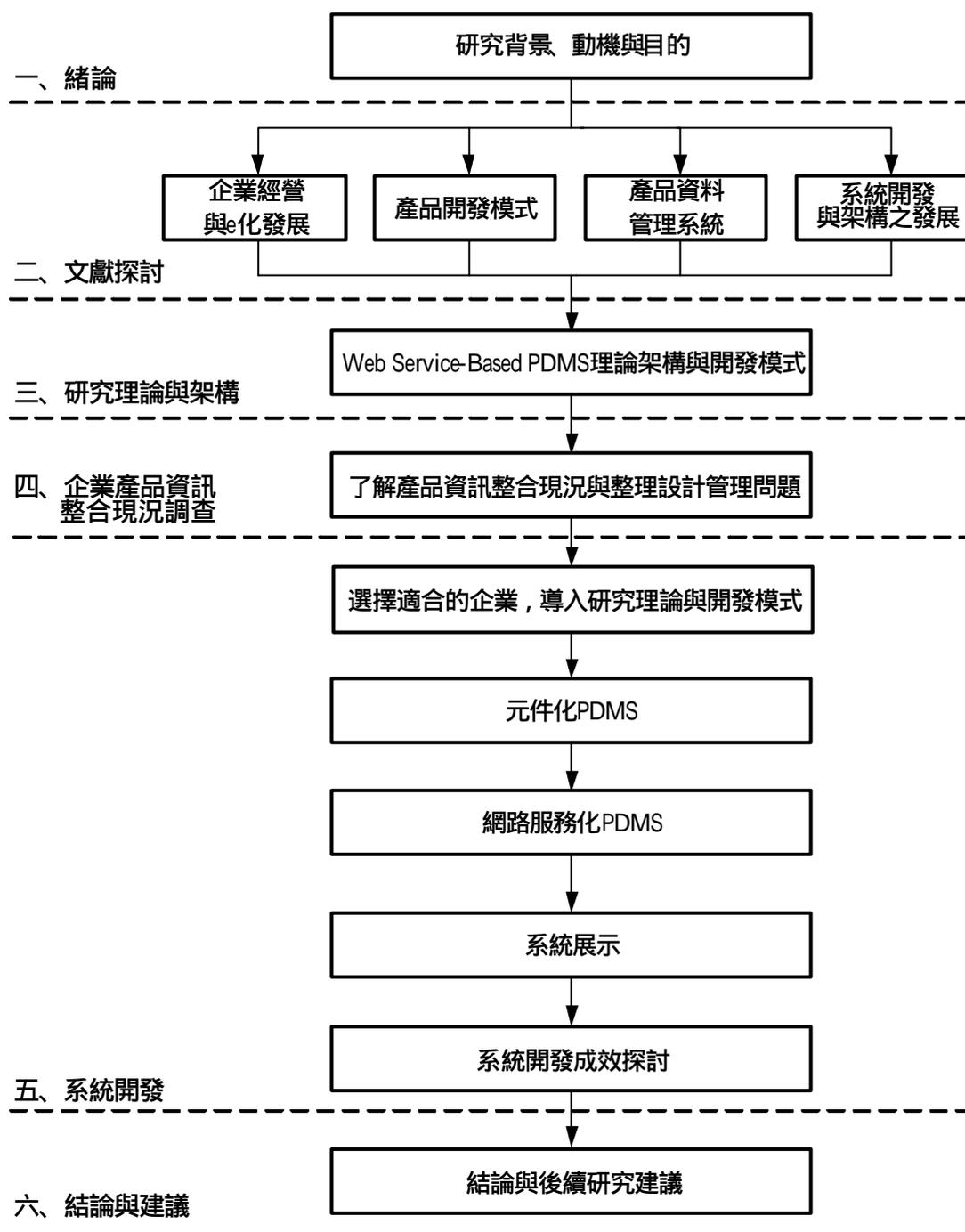


圖 1-2 研究流程圖

第二章 文獻探討

2.1 企業經營與 e 化發展

自從 1991 年網際網路開放應用於商業領域，以及高能物理學家 Tim Berners-Lee 提出 WWW (World-Wide Web) 通訊協定後【20】，企業電子化的發展，正式宣告進入嶄新的時代，成為資訊區域運用與廣域運用一明顯的分水嶺。微軟總裁比爾 蓋茲曾指出「1980 至 1990 年代是企業追求高品質，實施各種品質控制技術的時代；1990 至 1999 年代是企業再造、組織重整的時代；2000 年代則是邁向快速變遷、轉化迅速的數位革命時代。」因此，在瞬息萬變、資訊爆炸的網路數位時代中，「速度」決定了產業競爭力的關鍵。

為了在十倍速時代中保持競爭優勢、有效解決經營管理之問題，企業不斷調整其競爭策略與管理方法，如表 2-1 所示【21】。

表 2-1 企業競爭策略與管理方法之演變

	1970~1980 年	1980~1990 年	1990~2000 年	2000 年以後
競爭策略	成本	品質 成本	速度 品質 成本	創新 速度 服務 品質 成本
經營理念	為庫存而生產、僅考量生產與消費	為訂單而生產，納入研發為考量	所有營運管理包含設計、生產、市場預測、訂單等接納入整合性考量	為產品創造價值，重視客戶服務與環保考量
主要方法	大量製造 製程品管 數值控制 單機自動化	彈性生產 工廠自動化 品保制度	企業資源規劃 企業流程再造 產品資料管理 快速回應	供應鏈管理 知識管理 電子商務 全球資訊運籌系統

表 2-1 主要將近代企業經營與 e 化的過程分為四個主要階段：

一、1970~1980 年代：此時期的生產方式主要是以機械化大量生產，藉由生產效率的

提昇，達到降低成本的目的。由於市場消費需求遠大於供給，因此生產導向以追求規模經濟、少樣多量的生產的模式為主。

- 二、1980~1990 年代：此一階段市場需求逐漸趨向產品多樣化與品質要求，在少量多樣的生產模式下，企業管理的重心在於強調彈性化生產以及改善流程效率。由於生產技術與管理方法不斷改善，品質成為企業拓展業務與爭取訂單的利器，而產品的設計研發也逐漸納入經營策略的考量。
- 三、1990~2000 年代：由於全球化的衝擊與網路資訊科技的崛起，致使企業競爭進入戰國時代。此時代的市場導向以大量客制化生產模式為主，為維持競爭優勢，企業必須快速反應環境趨勢，完成全球客戶要求。因此，企業內所有營運管理皆須加以整合，以便提昇整體營運效能，而導入企業資源規劃、企業流程改造、產品資料管理及快速回應，則是此階段提升競爭力的主要方法。
- 四、2000 年以後：由於網路資訊科技快速普及，加速全球化競爭發展趨勢，如今企業除了必須快速反應市場需求之外，更須整合產業價值鏈體系。因此，企業藉由供應鏈管理與電子商務，整合上下游廠商與其合作夥伴；藉由知識管理保存營運經驗，提昇企業創新能力與開創產品價值；藉由資源的整合與全球資訊系統的建置，提供客戶迅速與高品質的服務；藉由環境的保護與發展零污染、高經濟價值產品等，都將是企業永續經營與提昇競爭力的因素。

經濟部商業司長劉坤堂於「經營現代化與網際網路商業應用」一文中【22】，歸納企業經營現代化的五大特徵如下：

1. 服務個人化：採用一對一行銷與 BTO(Build To Order)的方式，針對消費者設計出一連串的服務，以爭取忠實的客戶。
2. 經濟複合化：採用複合化的經營，使得服務的內涵增加，提昇客戶滿意度。
3. 通路效率化：加速新產品推廣速度，滿足市場多變的需求。
4. 業際整合化：利用整合行銷與聯合促銷的方式，讓不同領域之企業得以結合，提供客戶更廣泛、更多元的專業服務。
5. 市場國際化：面對我內需市場不足以及產品流行資訊普及的狀況，擴展國際市場與發展多樣化產品，是現代化企業經營的重要課題。

由上述五大現代化經營特徵與企業 e 化發展趨勢來看，成本、品質、速度等管理目標，已成為現今企業生存的必備條件，而企業未來能否於激烈競爭環境中嶄露頭角、

提昇整體競爭實力的關鍵則在於「服務」與「創新」兩大經營策略。若以資訊系統的角度觀之，則企業資源規劃與供應鏈管理系統，勢必須要作一整合運用，才能確保產品生產製造的正確性與流暢性，而未來企業資訊運用發展的重點，將趨向於客戶關係管理、電子商務以及知識管理等相關解決方案上，主要方向在於追求跨組織間的協同合作、大量客制化、客導化，以及提昇企業知識產能等。

2.1.1 企業電子化應用架構

英特爾董事長 Andy Grove (1999)曾說：「五年後，沒有什麼所謂的網路公司，因為所有存活下來的公司都將會是網路公司。」當顧客與供應商都已建立 e 化平台，加入數位科技的產銷體系後，企業已無法再用傳統營運模式，維持商業關係與保持競爭優勢。因此，縱使企業目前並沒有絕對的 e 化壓力，但不久的未來，終將隨著整個產業的 e 化速度前進，學習如何運用網路虛實結合的方式，適應全新的商務環境。除了前述現代企業經營策略之外，企業整體的電子化應用架構大致如圖 2-1【23】。

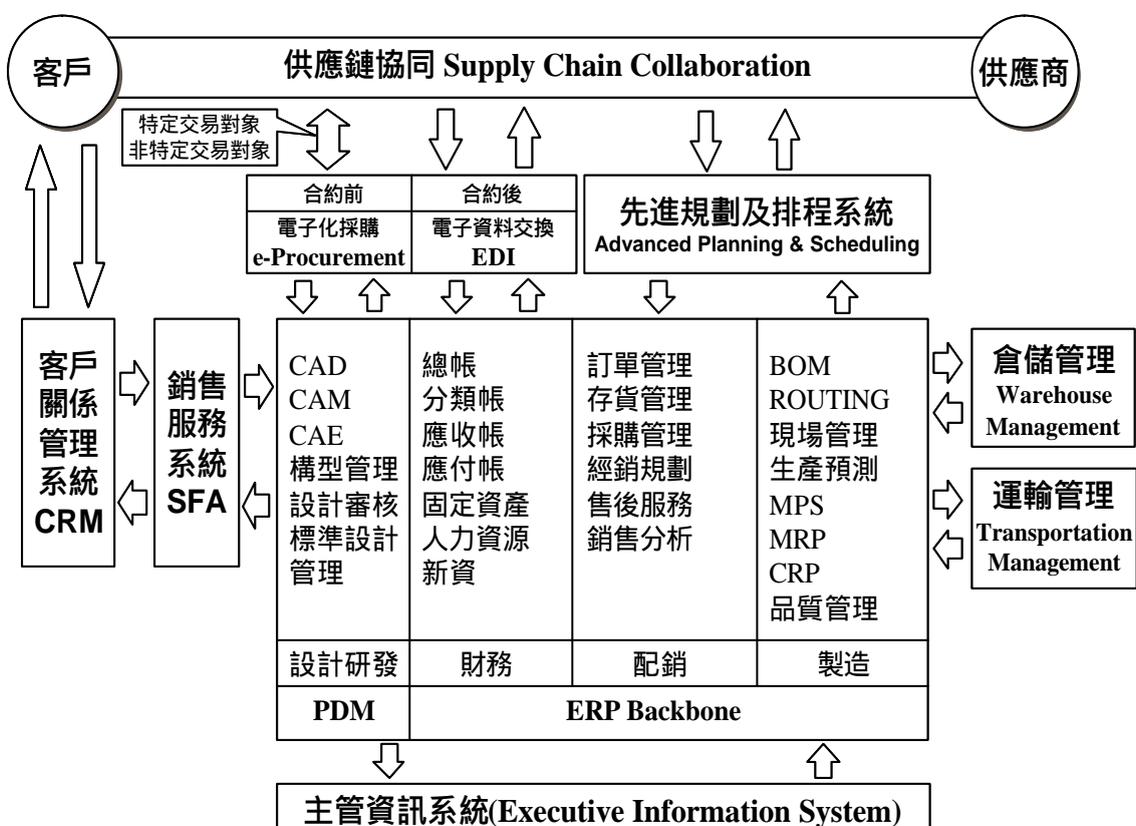


圖 2-1 企業電子化應用架構

圖 2-1 企業內部 e 化核心是以設計研發部門的 PDMS，以及結合製造、銷售、財務三大部門系統的 ERP 為主。一般 PDM 系統主要是將企業內所有產品相關資訊與程序，利用資訊技術加以整合管理與運用，如產品 CAD/CAM 設計資料、設計參數、製程參數資料等。ERP 系統則主要負責生產製造的管理，包括標準化之物料清單 (Bill of Material ; BOM)展開、工廠生產資料的收集與網路監控現場管理系統、主生產排程 (Master Planning System ; MPS : 生產順序、時程、人力、設備及物料計畫)、物料需求計畫 (Material Requirement Planning ; MRP : 由物料清單、庫存及生產排程訂出物料需求數量及次序)、產生製造工令及採購清單、零組件需求計畫 (Component Requirement Planning ; CRP : 管理零組件設計、製造屬性，以及庫存等資訊)，若再搭配銷售服務系統，以及倉儲管理系統，則可整合企業內所有商業活動與資訊流通。

在整個企業 e 化架構中，往上可繼續發展即時的生產排程規劃系統、建立企業間電子化採購環境，用以處理上下游供應鏈體系與顧客之相關資訊。在取得企業整體營運資訊並加以分析之後，所得到的數據可提供給決策支援系統，讓高階管理階層得以參考，並且訂定企業營運策略。面對企業外之顧客與供應商，則以電子資料交換 (Electronic Data Interchange ; EDI)與網際網路整合上下游供應鏈廠商，對特定及非特定對象，達成電子商務交易的目的。最後再透過 CRM 系統的運用，提供顧客 24 小時全方位服務與最新產品資訊等，強化客戶對企業的忠誠度。

由上之論述可知，所謂企業電子化，主要是將營運流程利用網路資訊科技加以改良，讓流程與傳統工作項目達到自動化處理、增加整體效能。而企業 e 化的主要精神則在於整合 (Integration)，整合的範圍包括企業內之產品資料管理系統、企業資源規劃、物流與銷售系統，以及企業外之供應鏈管理、電子商務系統等，再配合客戶關係的管理，完整收集整個產品生命週期中的所有資訊，提供企業經營者一迅速且正確的決策支援，構成一綿密、具快速反應能力的企業數位神經系統，如圖 2-2 所示【24】。

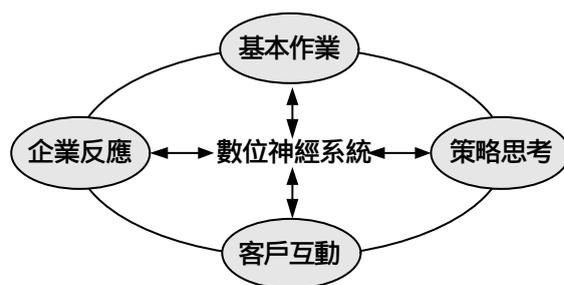


圖 2-2 數位神經系統

2.1.2 電子商務發展

90 年代初期，企業電子化的思考方向主要是從顧客需求、經營策略、以及企業流程再造 (Business Process Reengineering; BPR) 的角度，重新思考、設計企業流程，運用資訊與網路通訊技術，積極開發電子流程管理系統，以求在重要的績效衡量上，如成本、品質、服務與速度等，達到最大的改善。正當企業積極著手進行組織再造與內部流程電子化的同時，另一因網路資訊科技的推廣和應用，逐漸成熟的重大趨勢變革：電子商務 (E-Commerce)，則蓄勢待發地推翻與影響企業傳統經營模式，開創出嶄新的通路與機會，流動於網路上數以千億計的商機，正等待能夠以新觀念、新方法打破傳統經濟理論思維的企業去創造。

電子商務 (Electronic-Commerce; EC) 是第二次工業革命下的產物，也是現今前衛的商業潮流。廣義而言，電子商務是一種使用電腦網路來搜尋與取得資訊的商業行為，其目的在於因應公司組織與客戶的需求，達到降低成本、增進產品與服務的品質，並且提升整體營運流程效率【25】。由於眾多學者對於電子商務的廣義定義均包含商業行為與網際網路，因此可客觀地定義電子商務為結合電腦科技與通信技術，透過網際網路連線，以電子化的方式從事商業活動。電子商務依應用範疇之不同主要分為三大類【26】，如圖 2-3 所示【3】。

一、企業對企業 (Business To Business; B2B)：

主要以電子資料交換 (EDI) 作為工作進行之核心，用以發展上、中、下游企業間資訊的整合，首要應用於產業間的供應鏈管理以及電子資金移轉作業等，又稱為快速回應 (Quick Response; QR) 或者效率化顧客回應 (Efficient Customer Response; ECR) 機制。

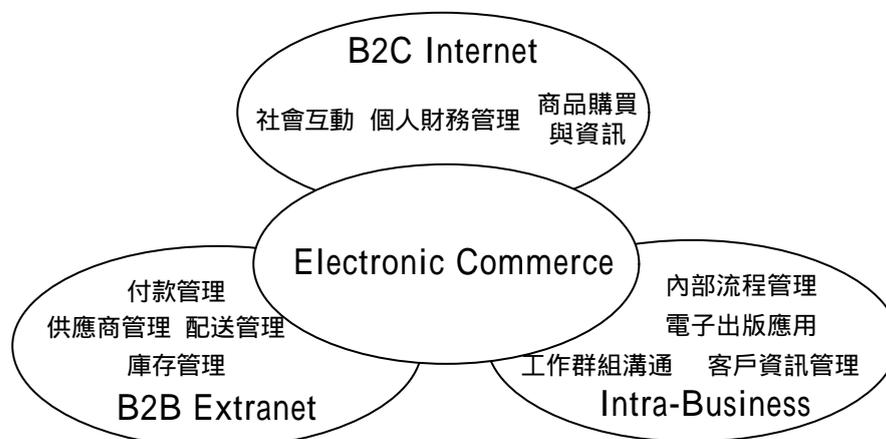


圖 2-3 電子商務的分類

二、企業對個人 (Business To Customer ; B2C) :

企業以建構電子商店的方式，讓消費者或企業客戶經由遠端電腦的瀏覽器，透過網際網路直接查詢及訂購商品，同時利用傳統物流通路運送實體貨品。此一模式又稱為電子商業 (Electronic Business)。

三、企業組織內部應用 (Intra-Business 或 Intra-Organization) :

一般中大型企業組織內部之活動，符合電子商務相關機能，以作業通訊為主的流程均可歸併於此一應用範圍，例如：公司之銷售點與總公司內的各項業務工作，以及部門間各類電子文件之瀏覽與交換等。

電子商務正式席捲全球，主要起源於 1997 年 7 月 1 日美國政府對外發布「電子商務白皮書」，至此之後所有國家無不以極快的步調建構屬於自己的電子商務空間，網路商業化已成為一股不可抵擋的趨勢【27】。我國政府為因應電子商務時代的來臨，自 1992 年起便積極推動各項資訊建設以及電子商務相關政策(如表 2-2 所示，資料來源：整理自【53】)，其主要目的為建立一資訊化社會，並協助企業運用網際網路技術發展高效率、低成本的商業營運模式，讓企業能以最小的阻力進入全球電子商務圈。

表 2-2 政府推動電子商務主要相關政策

年份	政策名稱	主要目的
1992	商業自動化十年計畫	推動商業自動化，期使商業交易活動能效率化，降低通路成本、提升商業品質，建立合理商業環境。
1996	國家資訊通訊基礎建設	建構資訊化社會基礎，擴展網路資訊基本建設，讓使用者能夠過網路與整個社會溝通，傳遞與取得各種資訊。
1998	電子商務政策綱領	確立我國發展 EC 原則，希望憑藉現有專業人力以及高科技產業的支援，透過電子商務的應用，提升產業總體競爭力，建設台灣成為二十一世紀的亞太商務中心。
	網際網路商業應用計畫	建立 EC 發展環境，推動產業應用網路，四年內達到四十個行業、五萬家上、中、下游企業導入應用網際網路。
1999	產業自動化與電子化推動方案	提出具體方案獎勵企業發展電子商務，推動產業自動化結合電子化，以加速協助企業發展 B2B 電子商務體系。
2002	促進中小企業電子化五年中長程計畫	協助企業建置產業網路資料庫、成立 e 化服務團、人才培訓、建置電子交易市集、提升網路系統供應商能力。

電子商務的時代已讓傳統中小企業無法憑藉過去的經驗判斷與經營模式，迅速掌握商機、因應環境重大轉變。經濟部中小企業處處長賴杉桂即表示「國內中小企業因規模小、資金少、缺少資訊科技人才，要運用電子商務拓展國內外市場較為困難。再加上資訊及網路科技蓬勃發展、全球化商業競爭市場的改變、產業經營模式的改變，中小企業正面臨空前的挑戰，亟需加速中小企業電子化，以迎接挑戰!【54】」。因此從表 2-2 可得知近年來政府對於國內中小企業網路資訊科技的發展以及電子商務的推廣相當重視，尤其是 B2B 的企業商務應用領域。

根據美國 Forrester 公司的研究顯示，B2B 的交易金額在 1998 年為 4300 億美元，到了 2003 年將成長至 1.37 兆美元，年複合成長率為 99%；資策會的報告則評估 2004 年全球幾個主要國家的 B2B 交易規模將約為 B2C 規模的 10 倍；經濟部所提出的資料顯示，台灣 B2B 市場的發展，預估到了 2003 年交易金額將高達 174.6 億元，年複合平均成長率為 81%【28】。因此，由圖 2-4【29】與上述研究可知，目前企業電子商務的發展趨勢與應用，主要著重在 B2B 商業市場上。隨著企業間電子交易環境的日漸成熟，以及面臨大陸、東南亞等低成本國家的強烈競爭，企業體系間的商務交易價值鏈將勢必受到衝擊，進而產生解構和重組效應；國內企業也將開始轉型為以服務、研發設計為導向的產業型態。

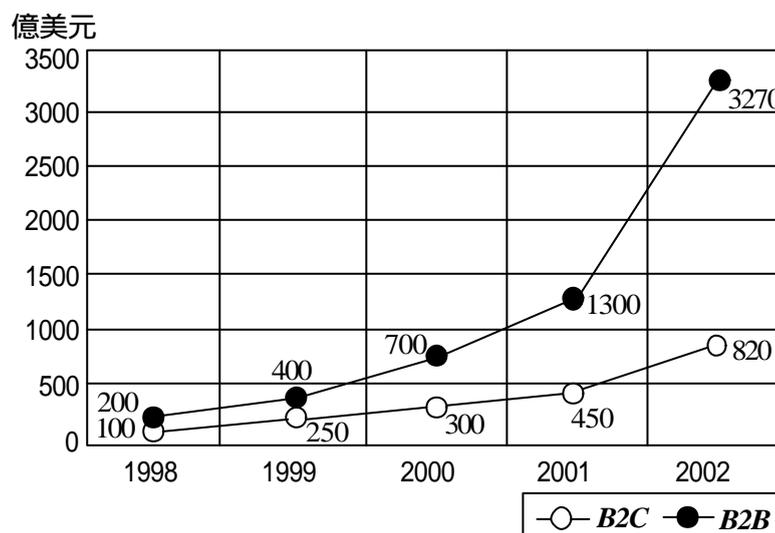


圖 2-4 全球 B2B、B2C 市場規模

由於網路商業環境在於強調跨組織之間的競爭，而非個別企業競爭【16】，加上我國有 85% 以上的企業屬於中小型企業，產業型態又以垂直專業分工為主，因此國內

企業欲發展 B2B 電子商業模式，則必須利用網際網路，透過供應鏈管理，虛擬整合上中下游廠商，其整合的範圍包含客戶服務、產品設計、品管、行銷、物流配送、技術支援與售後服務等，形成一具備彈性、快速反應的虛擬企業，協同追求價值鏈資源整合(如圖 2-5 所示，資料來源：本研究)、充分掌握中心廠與衛星廠的狀況，達到產品資訊快速流通之目的。

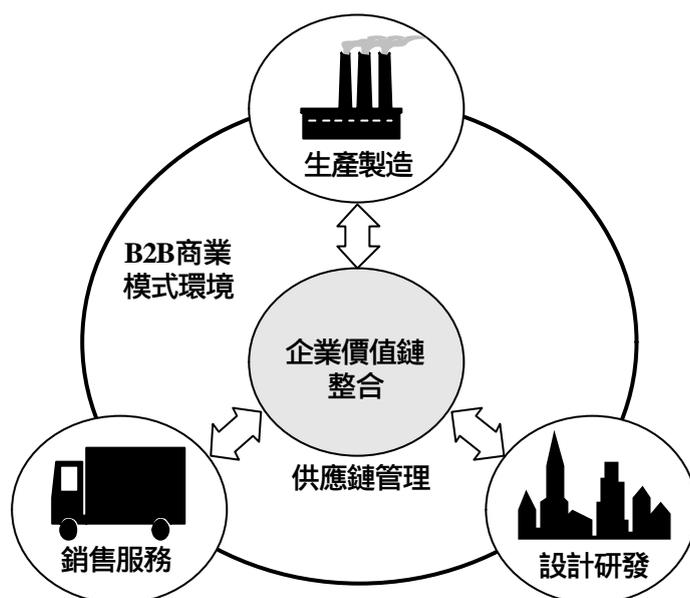


圖 2-5 國內中小企業發展 B2B 之模式

2.2 產品開發模式

今日企業正處於產業型態轉變及科技日新月異的快速變遷時代中，對製造業而言，降低生產成本、縮短新產品上市時間，以及兼顧品質改善與客戶滿意，獲得成本與時效上的效益，以成為企業永續經營的必備條件。而在新產品研發上，企業經常遭遇到的問題與挑戰包括：

1. 全球化競爭與產品生命週期縮短。
2. 難以保持與發展核心競爭能力。
3. 缺乏設計研發人才、產業環境不佳。
4. 組織內外網絡缺乏彈性與流暢性，難以整合溝通。
5. 產品開發時程掌握不易。

6. 客戶對產品功能、品質、上市速度、價格、服務之要求日益嚴苛。

為了克服日愈嚴峻的產業環境問題與挑戰，企業開始採用新式產品開發模式，利用網路資訊科技，克服因為時間距離所造成的隔閡與差距，且藉由產品開發資訊的管理輔助，以往許多繁雜耗時的開發流程與工作內容，也逐漸予以自動化處理，大幅降低資料、流程管理的難度與產品開發的時間。

產品開發模式依據 Nonaka 與 Takeuchim【30】的研究，大致可分為循序式(Sequential Engineering)與同步工程式(Concurrent Engineering；CE)兩大類。然而近年來因為網路資訊科技的崛起，使得眾多學者以不同的觀念與角度，重新審視同步工程的概念，進而衍生許多符合時代潮流的理論方法，其中整合性產品開發模式與協同產品開發模式，為近代較為學術與企業探討應用的主要方向。

2.2.1 同步工程

傳統產品開發模式稱之為序列式產品開發(如圖 2-6，修改自【11】)，藉由市場的分析研究及蒐集顧客需求後，展開一系列的產品設計、製造、測試、組裝與上市。由於以往在產品設計研發階段時，設計活動侷限於設計部門，僅考量產品的性能(Performance)及功能(Functionality)，而有關製造程序、製造限制、組裝程序、物料需求規劃，以及市場特性等問題，礙於組織、環境之隔閡，使得設計研發人員未能及早考量後續處理程序，其他部門人員亦無法提供專業意見，導致產品開發作適時、適材、適人、適作等考量。一旦問題產生時，僅能回溯至設計研發階段，進行設計變更，不但造成人員與成本上的極大浪費，更使得整個開發時程不斷延長、喪失上市時機。

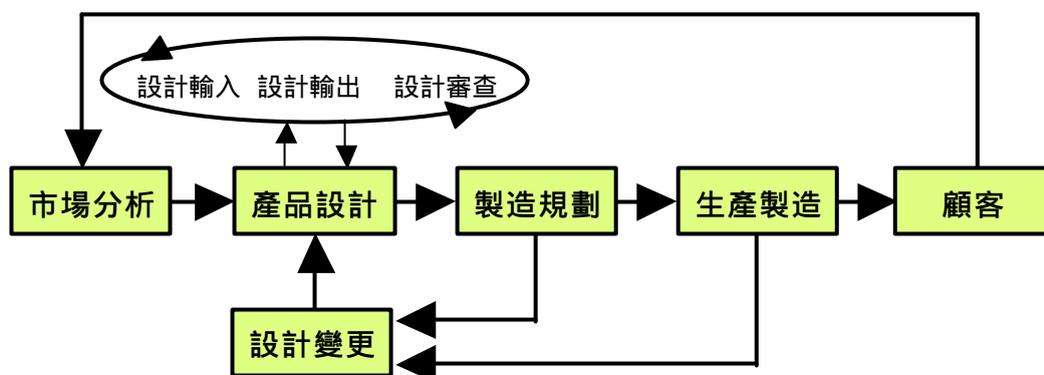


圖 2-6 序列式產品開發

而所謂的同步工程研發，主要是有一同步工程轉軸環境，如圖 2-7 所示。轉軸的組成是以產品研發生命週期(Life-Cycle)為主的任務編組，凡是與產品研發相關的專業部門，例如：製造、分析、組裝、後勤等，都需參與設計研發的工作。其主要目的是讓各領域的專業人員能夠在產品設計階段時，提早參與產品開發並提供改良建議，讓後續的開發流程得以順利進行，縮短產品上市時間以及減少設計變更成本。

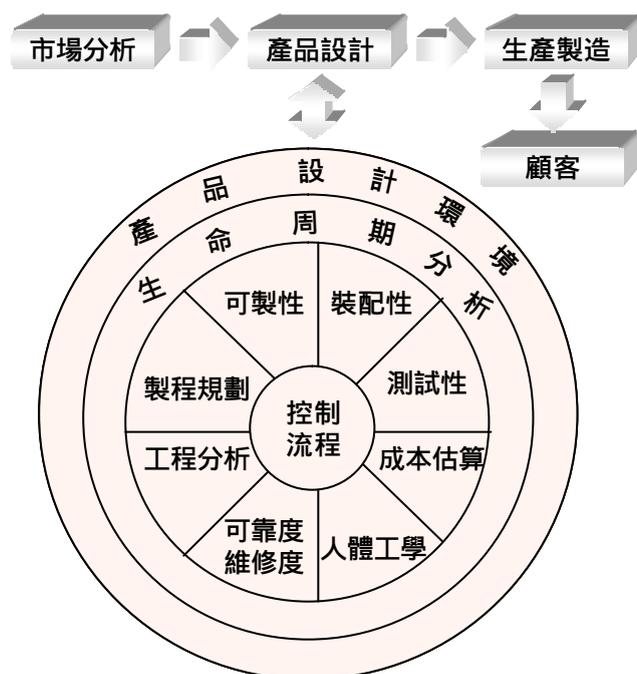


圖 2-7 同步工程研發環境

根據美國 L.P.Sullivan American Supplier Institute 的研究調查顯示，同步工程與傳統序列式的產品開發，在設計研發階段時所產生的設計變更總數大致相同，其兩者最大的不同點在於設計變更所發生的時間點，如圖 2-8 所示。同步工程設計變更的高峰，主要出現在初步設計階段，由於一開始即需考量後續處理，因此絕大多數的問題均在

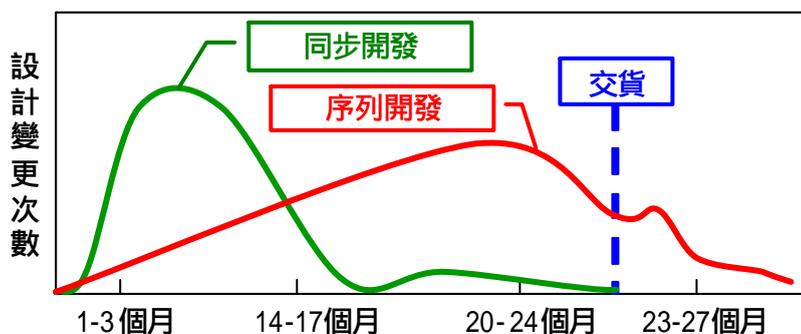


圖 2-8 產品開發模式與設計變更

此階段解決；傳統序列式產品開發的設計變更高峰，則出現在產品製造組裝，以及交付客戶兩大階段。

然而，根據長鞭效應 (Bullwhip Effect【55】)理論可知，產品在設計研發開發階段時，愈早發現設計錯誤與後續階段可能產生之問題，將可減少大量開發成本；倘若錯誤發現的時間點愈晚，則所需付出的成本代價，將以等比級數成長。如圖 2-9 所示【11】，若企業在初步設計階段處理設計變更所花費的成本為 1 倍，則設計變更發生於產品生產階段時，企業必須付出相對 1000 倍的處理成本。因此，同步工程的研發模式不但可有效減少產品開發成本，其對於整個產品開發專案時程的控管，也比序列式產品開發更為優越與準確。建立同步工程產品開發環境，必須達到以下五個條件【56】：

1. 工作與工具互相配合。
2. 電腦整體環境互相配合。
3. 產品資料管理。
4. 自動化程序管理。
5. 自動化決策支援。

由此觀之，要完全滿足上述之條件，企業必須先建立完善的網路環境，導入彈性且易於整合的資訊管理系統，將產品開發過程中的資料、程序、應用程式以及軟體系統等，作適當的串聯與運用，如此產品資訊才能有效地傳遞分享，若開發過程受到阻礙，也才能快速自動回饋訊息，警告通知企業人員作出適當地處理。

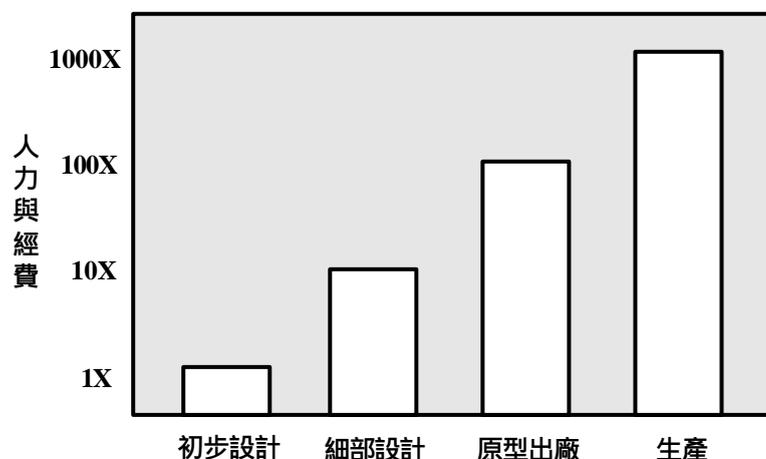


圖 2-9 各階段設計變更成本

2.2.2 整合性產品開發

同步工程的主要概念是希望建置一協同合作環境，讓產品開發案的所有成員，包括設計師、工程師、採購人員等，均能適時地參與討論專案內容。然而在實務上，要求企業內外各個產品開發成員都必須配合同步工程的產品開發，對一般企業而言通常是難以達成且不切實際的，而整合性產品開發的概念主要是利用資訊科技，讓設計研發人員得以適時取得產品生命週期中的相關資訊，進而實現同步工程的理想。

針對整合性產品開發系統，以往已有許多研究以不同角度切入並提出相關論述，如表 2-3 所示(整理自【33】)。

表 2-3 整合性產品開發相關研究

學者	論述角度	主要觀點
Parsad, <i>et al.</i>	資訊管理	以資訊整合的觀點綜論同步工程環境之特質，實現同步工程之工具/方法，及資訊管理的架構。
Liau, <i>et al.</i>	專家系統	建構以互動式限制模型(Interactive Constrain Modeling)系統，解決設計與製造間雙向整合之問題。
Parsad	同步化機制	以更廣義的方式，定義同步之產品與流程組織(Product and Process Organization; PPO)，與整合性產品開發(Integrated Product Development; IPD)兩機制之特質，來達成產品設計、開發與上市之同步化系統(Current Product Design, Development, and Delivery (PD ³) System)之要求。
Tsuda	品質技術	以品質機能展開(Quality Function Deployment; QFD)方式構建同步工程產品開發模型，並以 2 階段品質圖(2-Storied Quality Chart)觀念簡化 QFD 在複雜系統之規劃模式。
Scheer, <i>et al.</i>	企業流程再造	以整合性資訊系統架構之企業工程屋(House of Business Engineering)觀念，進行整合性產品開發流程模型構建與作業流程技術之整合應用，達成研發專案之流程建立與分析，及資訊系統之建立。

從表 2-3 可知，現今整合性產品開發的主要發展與應用方向為資訊整合管理、品質整合管理與流程整合管理等，王魁生等學者(2000)則以上述之研究為基礎，再納入品質標準、決策支援系統，以及資訊共享等觀念，提出一整合性產品開發管理系統基本架構，如圖 2-10 所示。

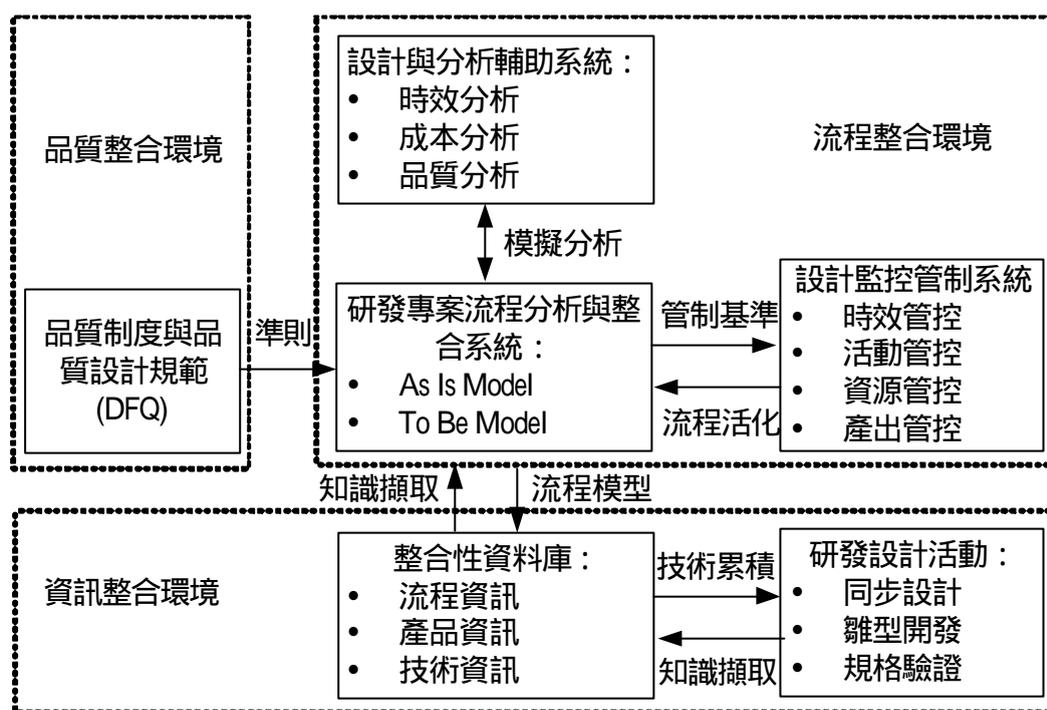


圖 2-10 整合性產品開發環境基礎模型架構

圖 2-10 建立品質整合環境的目的，在於以導引產品開發人員具備品質觀念，使新產品開發流程、活動與產出具備「品質是設計出來的」的特質，而整合的目標包括品質管理，品質設計與品質分析等相關技術實務；建構流程整合環境則是以研發專案流程為整合核心，透過模擬分析工具與開發流程的控管，明確定義各部門於開發活動中的整合與分工，在有限資源限制條件下，能適當調配營運資源，以有效達成研發專案目標；建立資訊整合環境，主要是強調企業技術知識管理與資訊共享，以產品生命週期資訊管理為基礎，定義產品資訊、流程資訊與技術資訊之資料架構(Data Schema)與表達方式，使開發經驗與資訊得以有效共享與交換，達到縮短開發時程之效益。比較同步工程與整合性產品開發模式的最大差異在於，整合性產品開發較強調以系統化管理的方式，整合產品生命週期中的相關資料、資訊、流程與知識，並利用自動化處理的機制，達到產品高品質、低成本與縮短交期的目標。

2.2.3 協同產品開發

協同產品開發 (Collaborative Product Development ; CPD)主要發展於 20 世紀 90 年代末期，其主要概念是將產品生命週期管理的概念，由企業本身擴及整個產業價值鏈體系，藉由資料、資訊、知識、流程的集中化運作與管理，提升企業間協同商務效率，使價值鏈體系內所有廠商的藍圖、設計變更、文件、產品資料、資訊等均能全面整合與共享，包括企業本身內部的各種營運到公司外部供應商、客戶、協力廠的資源，皆能有效協同合作與整合，如圖 2-11 所示(本研究)。

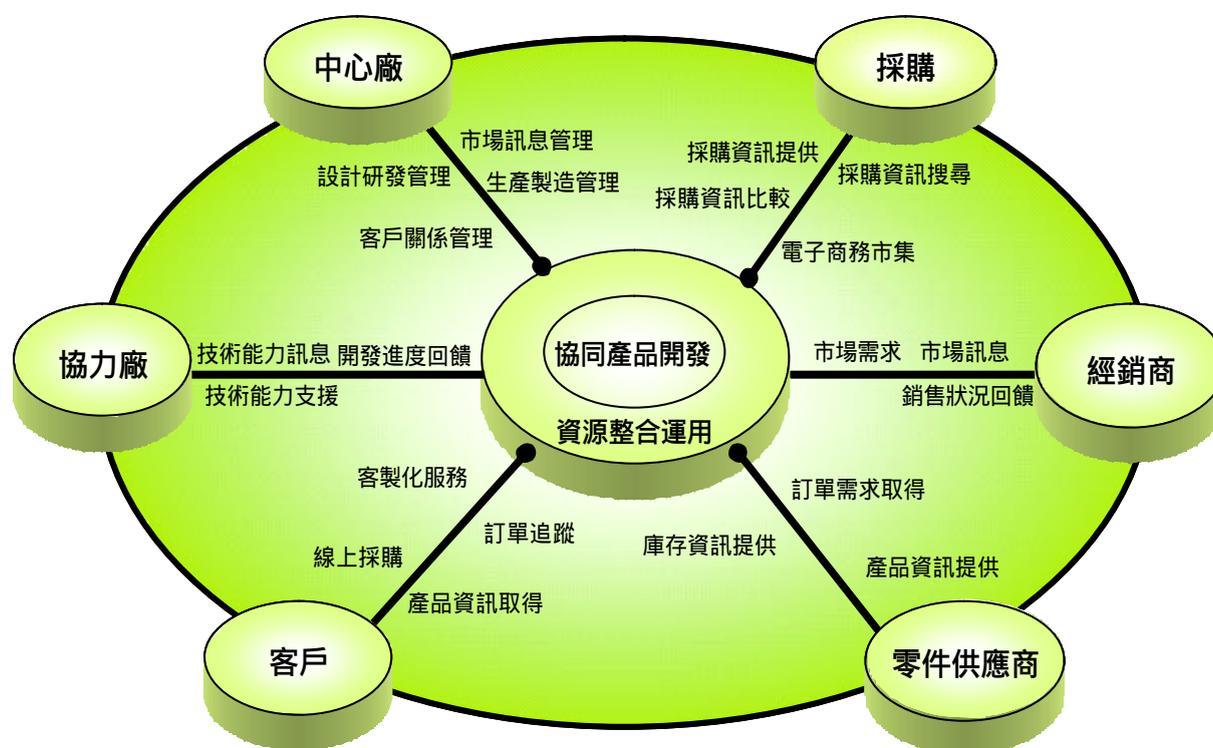


圖 2-11 協同產品開發概念

協同產品開發與整合性產品開發的主要差異在於，協同產品開發除了具備整合性產品開發的概念之外，其更強調企業之間的協同合作關係與電子商務平台的建立。藉由通用性協定、標準、媒介與工具，讓價值鏈上經授權的成員可以透過網際網路取得各式資源與進行協同開發。美國 META Group 認為協同產品開發所包含的領域除了銷售鏈、供應鏈廠商內外電子化與流程整合之外，更包括了設計、規劃、預測等不同層面，姚威宏則進一步認為 CPD 尚需包括製造協同商務及客服協同商務等領域較為完整，如圖 2-12【32】。

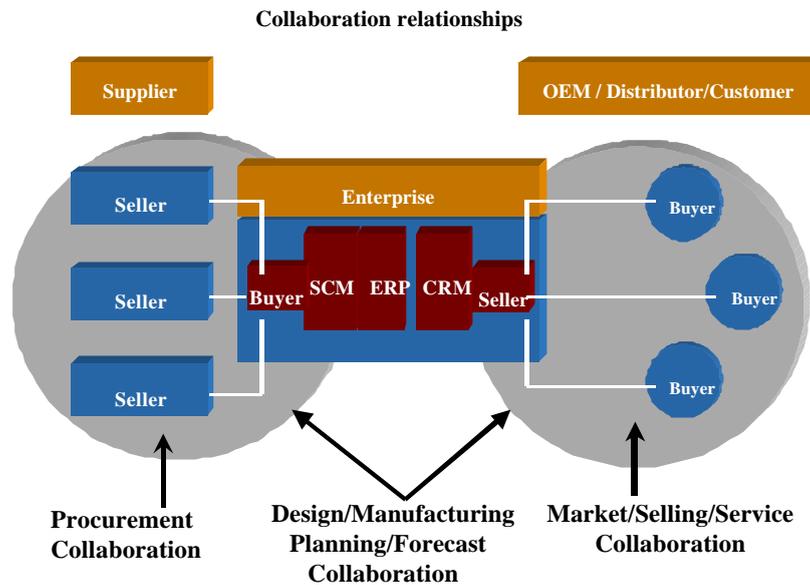


圖 2-12 協同產品開發之領域與關係

如圖 2-12，協同產品開發除了整合企業內部 SCM、ERP、CRM 管理系統之外，尚須與左邊的供應鏈體系進行協同採購，而與右邊的下游廠商、客戶等，進行銷售與服務的協同合作。採用協同產品開發模式對於企業產品開發工作的影響如下【33】：

1. 不再侷限於研發部門：
隨著產品的發展，整合與鏈結分散於各處的資源，建構共同的開發模式，讓成員在不同的階段中可透過網際網路，將產品知識有效的供應與傳遞給相關人員。
2. 全企業的開發團隊：
透過跨企業文件管理、結構管理、工作流程管理、生命週期管理、以及系統管理等常用的功能來支援整個產品團隊。另外，具彈性的存取控制權限設定，保障了產品生命週期內各階段資訊存取的安全性。
3. 針對客戶的需求 (Sense-and-Respond)：
因應全球客戶要求而進行生產所必須具備的敏捷性，同時讓產品的設計變更減至最低，並可以讓企業將組態和客製化邏輯納入產品定義內，進而製造出富有彈性且具備多樣性的產品。
4. 企業協同工作：
整合來自價值鏈體系上不同專業領域的資訊與資源，讓產品開發成員得以分工合作，創新與共享產品資訊，而完全不受來源系統的限制。

5. 快速取得所需資源：
透過零件資料與供應商資訊之標準化與合併，大幅降低全球採購和產品開發的成本。把零件選擇程序標準化，並將下游供應鏈最佳化，使得可以重複使用的零件、商用零件、和偏好的零件供應商等都變得很容易辨識。藉著以網路為基礎的架構，成為跨產品線、跨事業單位、和跨供應鏈的理想環境。
6. 產品資料管理：
完整的PDM產品資料管理，包括整個產品與流程生命週期間的組態和變更管理
7. 製造流程計劃：
建立與維護詳細的流程計畫，並執行生產分析與流程模擬作業。
8. 資訊即時整合：
透過與企業資源規劃系統整合，傳遞高價值的生產相關資訊，確保系統內有完整而一致的產品相關資料，完全不受個別資訊存在位置之限制，使企業經營效率大幅提高，可以讓零件主檔、物料清單、和工程設變等資訊在系統之間相互交換。
9. 充份運用企業知識：
藉由協同的產品開發，讓企業得以迅速將開發成員的專業知識，透過系統化的方法，加以擷取、管理與分享，提高產品開發效率及品質。

CPD 的主要效益在於設計變更的減少與上市時間的縮短，例如惠普科技於導入 CPD 與相關技術之後，其設計變更減少約 20%，平均交期由原本 34 天減為 18 天，而根據 US Air Force 的研究報告顯示 CPD 比傳統產品開發所需時間縮短約 40%，尤其在設計修改製程規劃階段，因資料再利用、製造重用率高、大幅縮減重製率的浪費，縮短產品開發週期及產品上市之時間，如圖 2-13 所示【32】。

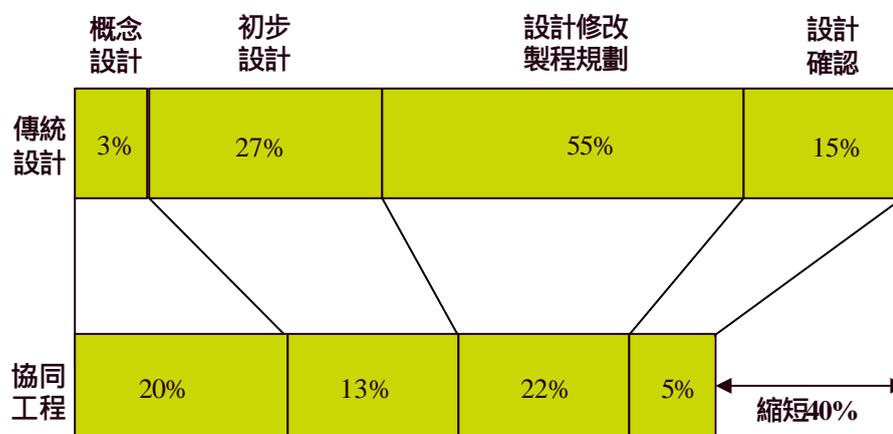


圖 2-13 協同產品開發優勢與效益

2.2.4 產品開發模式比較

本文將產品開發模式分為四大類別，分別為序列式、同步工程式、整合式，以及協同式，企業採用每個類別均有其優缺點，茲整理如表 2-4 所示。

表 2-4 各式產品開發模式比較

開發模式	主要優點	主要缺點	實施建議
序列式	<ul style="list-style-type: none"> • 管理單純化 • 無須於設計階段投資製造相關設備 	<ul style="list-style-type: none"> • 各單位之間缺乏互動 • 設計變更頻繁出現於後段開發流程導致開發成本呈倍數增加 • 產品開發時程增加設計人員與客戶缺乏互動，難以滿足客戶期望 	制定標準產品開發流程與負責單位，明定設計變更欄位與處理方式，規範產品與零件的編碼方式，確保編碼與產品資料管理方法的一致性，以降低未來導入同步工程與企業電子化的阻礙
同步工程式	<ul style="list-style-type: none"> • 減少製造階段的產品設計變更 • 縮短產品開發流程 • 提高生產效能 • 提高產品品質 • 知識經驗交換 	<ul style="list-style-type: none"> • 設計階段即需投資相關製造設備，增加成本支出 • 難以跨部門、組織協調溝通 	評估組織架構與產品開發成員工作內容，成立產品研發任務編組，使用電腦輔助軟體與分析方法達到資料共享以及產品設計、製造最佳化的目的
整合式	<ul style="list-style-type: none"> • 快速反應 • 自動化處理機制 • 知識經驗保存 • 系統化管理 • 更完善的同步工程環境與效益 	<ul style="list-style-type: none"> • 難以將流程與資料標準化與電子化 • 需具備跨部門的網路資訊設備 • 組織與人員需配合 e 化作適當的改變 	制定標準化開發流程與資料結構，建立完善的網路架構與自動化處理機制，藉由資訊科技與統一的使用者介面，達到開發知識經驗的管理及資訊快速反應
協同式	<ul style="list-style-type: none"> • 開發時程大幅降低 • 資源整合利用 • 發揮整體競爭實力 • 分享專業領域知識經驗 • 降低生產成本 	<ul style="list-style-type: none"> • 跨企業協同合作不易 • 資訊技術門檻高、整合不易 • 導入成本高 • 自動化處理機制難以建立 	採用符合國際標準的通訊協定與資料交換機制，建立完善網際網路環境與產品資料控管中心，讓使用者可透過瀏覽器直接查詢產品資料、開發進度以及召開線上會議等

產品開發模式的導入與實施，必須根據企業的組織架構與內部資源而決定，先了解產業環境之要求，再評估與參考成功案例的方法及執行步驟，以模組化的方式逐步導入。然而，導入時所採用的資訊系統與網路架構等，應使用廣為使用的標準協定與開發工具，以避免造成後續模組擴充與系統整合上的困難。

2.3 產品資料管理系統

產品資料管理的需求起始於 80 年代初期，由於電腦科技快速發展，產品開發的 CAD/CAM/CAE 技術也隨之不斷演進，從早期 2D 平面到 3D 設計、製造、模擬、測量等，廣泛地採用電腦化處理，進而產生出大量的電子資料，也造成企業下列的現象：

1. 各部門採用不相容的電腦系統，造成管理維護困難。
2. 各應用軟體之間無一致的標準介面，使得轉換工作負擔增加。
3. 工程部門與製造部門對材料表與產品資料各有不同的需求，容易造成混淆。
4. 產品開發所產生之大量資料，難以用人工的方式處理。

上述之現象使得產品資料管理的工作日益艱難，面對競爭環境地不斷挑戰，企業已無法再用傳統方式管理產品開發資料，以及克服資訊化孤島之問題，因此企業急需一套資訊系統，用以管理產品開發過程中所產生的龐大資料與相關作業程序，產品資料管理系統即在此需求下誕生，並且配合資訊科技的演進不斷改良。

產品資料管理的定義與名詞不勝繁舉，例如：PDM(Product Data Management)、PLM(Product Lifecycle Management)、EDM(Engineering Data Management)、PIM(Product Information Management)等，其中以 PDM 最廣為使用。廣義地說 PDM 是一種工具，可管理產品生命週期內之所有產品定義資料，包括：工程圖、材料規格、材料需求表等等。其主要功能分為兩大類【11】，分別是使用者功能與實用功能。

在使用者功能上，PDM 具備：

1. 資料及文件管理：提供設計資料與文件之安全與存取控制，建立資料間之相互關係，資料的轉出(Check-out)及轉入(Check-in)，及組織與使用者名單建立等功能。
2. 流程與程序 (Workflow & Process)管理：提供流程定義、版本、版次之關係與控制，以及工作流程與其他程序等管理。

3. 產品結構 (Product Structure)管理：提供零件清單(Part List)、材料表、零件定義、零件相關附屬資料(Attributes)，零件、結構、基準、產品與零件之關係，以及相關型態管理等功能。
4. 分類(Classification)及取用(Retrieval)：提供查詢工具以取得標準件與設計資料。
5. 計畫管理 (Program Management)：建立分工結構與時程管制功能。

在實用功能上，PDM 提供：

1. 基層結構通訊 (Communication Infrastructure)：提供訊息傳遞及與外部系統相互連結之功能，建立完整的通訊架構組織。
2. 資料移轉 (Data Transport)：提供資料在使用者、應用軟體、及系統或產品間轉換之移轉功能。
3. 資料格式轉換 (Data Translation)：提供資料在不同應用軟體，如 CAD/CAM/PDM 間之格式轉換功能。
4. 影像服務：提供設計藍圖審核時之影像觀看功能，也可與外部影像註解軟體結合，作為設計審核時之意見交換。
5. 系統管理：提供系統安裝、使用者自設畫面以及 PDM 系統管理等功能。

整個 PDM 的系統架構如圖 2-14 所示【34】：

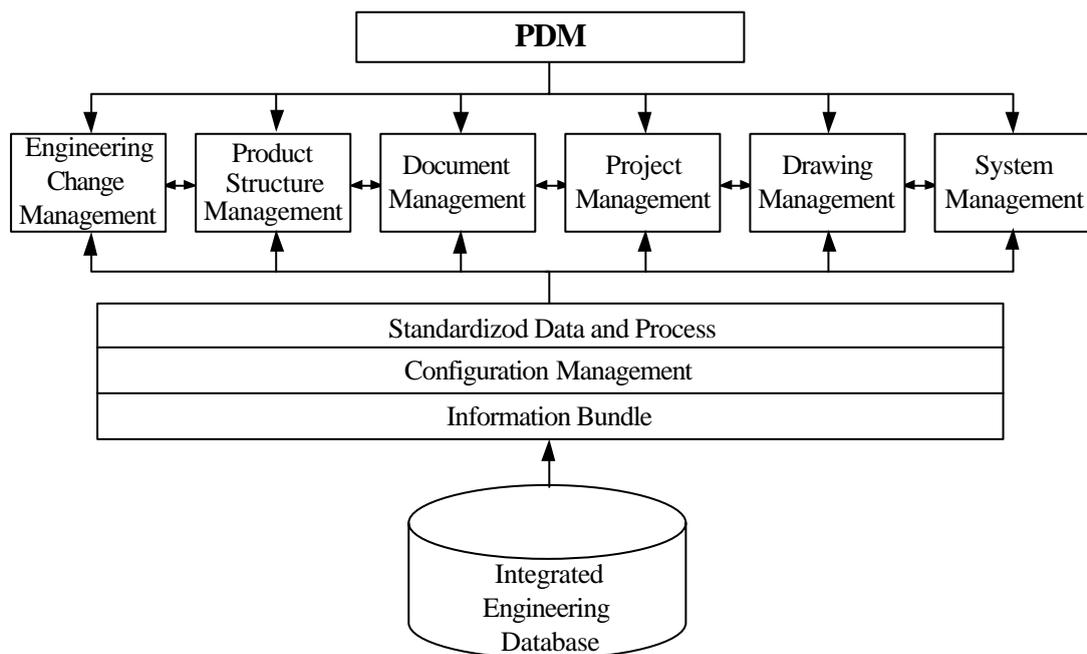


圖 2-14 PDM 系統架構圖

由於 PDM 以系統化的方式管理產品開發的所有資料與程序，因此導入 PDMS 能為企業帶來以下之效益：

1. 縮短產品上市時間。
2. 提高設計效率與品質。
3. 增加產品開發的正確性。
4. 輔助最佳化設計。
5. 輔助專案管理，控制開發時程。
6. 維護資料的安全性。
7. 跨企業或軟體的產品資料交換。

近年來政府積極輔導傳統產業，期望能提升企業設計研發能力，建立我國成為高附加價值的設計研發中心，產品資料管理系統的應用與導入，對於國內企業從製造導向轉向設計導向的過程，扮演著成功關鍵的角色。根據 CIMdata 公司於 2001 年的 PDM 市場研究報告顯示(圖 2-15)，PDM 市場於 2000 年已有 30 億美元的產值，在 2001 年成長了約 44%，預估到 2005 年將達到 130 億美元，而 Aberdeen Group 的研究報告(2000)更預期 2005 年 PDM 市場規模將達到 500 億美元之多，相較於趨向飽和的 ERP 市場，可見未來 PDM 勢必成為一發展重心與新興市場需求。

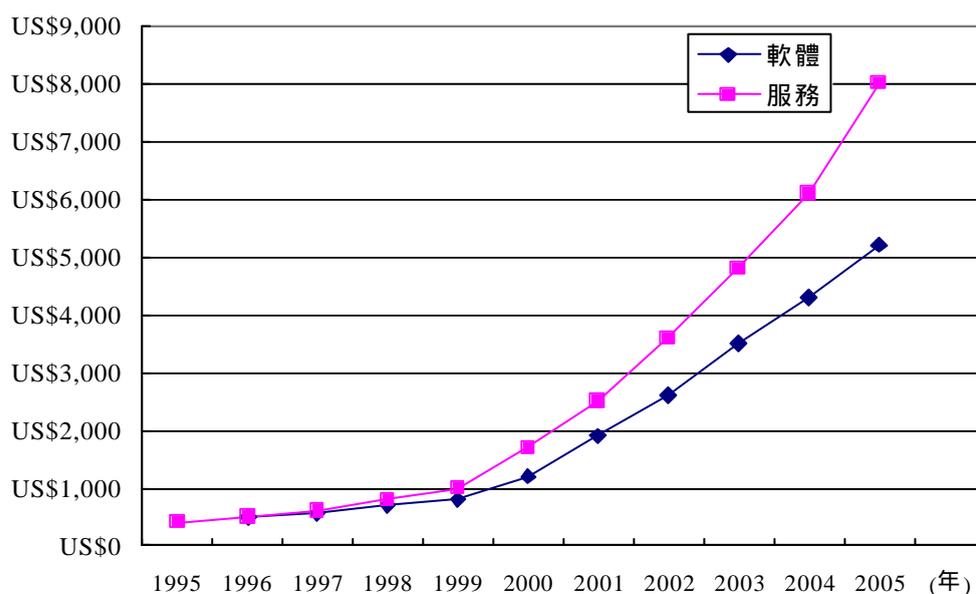


圖 2-15 PDM 市場規模趨勢圖

2.3.1 PDMS 的演進

在資訊快速變遷下，管理資訊系統的架構愈顯重要，它除了需考慮易用性、易維護性、相容性、易擴充性等重要條件因素外，開發成本、流行(新技術)將主導企業競爭力的強弱，因此如何建構出合乎未來趨勢的資訊架構，是現有資訊系統調整方向的首要目標，否則縱使有最優良的資訊人員，最後可能還是徒勞無功【35】。

產品資料管理系統是製造企業 e 化的主要核心，因此其架構良莠與否勢必會對未來企業電子化形成重大的影響。由於近年來資訊技術不斷發展，產品開發環境日益嚴峻，因此傳統的主從架構(Client/Server)PDMS，已無法處理龐大的產品資訊需求及符合環境趨勢，進而逐漸採用更具彈性、延展性的多層式(N-Tier)與網路式(Web-Based)系統架構，採用分散式系統架構使得 PDMS 的執行效率與穩定性均大幅提升，而採用網路式架構則讓使用者利用一般瀏覽器即可透過 Internet 操作 PDMS。

現今 PDMS 的應用範圍已非侷限於單一部門或者產品設計資料的管理，其整體應用策略從最前端的市場反應，到設計研發部門的電子資料管理、資源規劃與庫存管理系統、銷售訂單管理系統、製造系統、排程系統、品質管制以及供應鏈管理系統等，均與產品資料管理系統有著密不可分的連結關係，如圖 2-16 所示(CADesigner, 1998)。

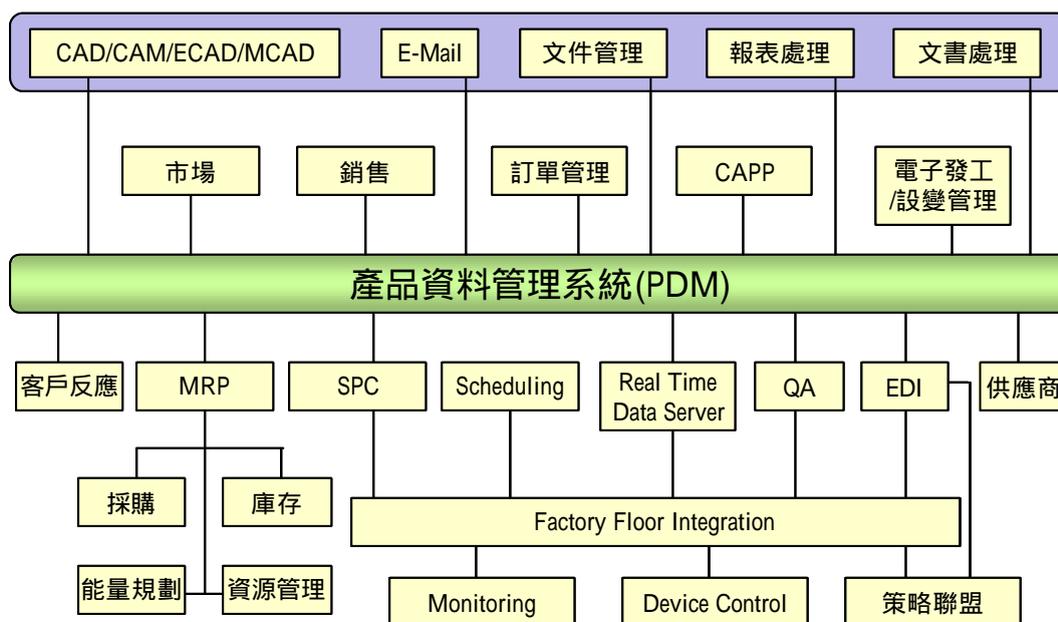


圖 2-16 PDMS 整體應用策略

PDMS 的發展已正式進入到協同產品商務(或稱為第三代 PDMS)的時代。如表 2-5 所示(修改自【36】), 第一代的 PDMS(產品資料管理)起源於 80 年代初期, 大多是由各 CAD 軟體廠商為搭配其 CAD 產品所推出的管理系統, 主要功能侷限於工程圖檔的管理, 解決了大量工程圖、技術文件以及設計圖檔的電腦管理問題, 屬於單一部門專用; 第二代 PDMS(虛擬產品開發管理)發展於 90 年代初期, 主要應用於整個產品開發團隊, 為企業提供各式產品工程資訊, 以 SDRC 公司的 Metaphase、UGS 公司的 iMAN 為典型代表。第二代 PDMS 已開始採用公開的標準, 加強系統對產品資訊的整合能力, 使得 PDMS 開始成為企業產品資訊與開發流程的集合中心, 透過權限的控管, 提供各部門專屬的產品資訊服務, 同時由於第二代 PDMS 的架構與功能逐漸發展成熟, 在商業領域獲得廣泛的使用, 因此造就 PDMS 成為一項新興開發市場, 帶動 PDMS 應用風潮; 第三代 PDMS 發展於 21 世紀初期, 它是一個完全建立在 Internet、元件技術基礎上的產品, 其主要目的是希望能透過標準化的資訊傳遞方式與系統架構模式, 整合產品生命週期中的所有資源與資訊, 讓合作夥伴、上下游供應商、海內外部門等, 能夠在第一時間內獲知並且運用產品開發資訊, 進而形成一營運共同體與全球知識網, 創造最大的商業利潤空間, 屬於產品價值鏈上所有成員協同使用, 以 PTC 公司的 Windchill 和 MatrixOne 公司的 eMatrix 為典型代表。

從表 2-5 與上述之探討可知, PDMS 的未來發展趨勢將會朝向更開放的系統架構與更多元的產品資訊服務, 如此才能與企業其他資訊管理系統, 例如 ERP、CRM、SCM 等, 作無縫的整合與運用, 避免資訊化孤島的狀況產生。

表 2-5 PDMS 系統之演進

	PDM	PDMII	CPC
興起時間	80 年代	90 年代	21 世紀初
管理模式	各專業開發模型	同步工程	協同產品開發
競爭焦點	追求利潤	市場佔有率	市場整體容量
產品開發策略	低成本	縮短產品上市時間	產品創新
技術焦點	提高生產率	資料、資訊分享	資源整合與協同合作
流程焦點	連續的設計過程	整合性產品開發	跨企業協同
系統架構	主從架構	分散式、元件式系統	網路式、開發式系統
應用範圍	企業部門	跨部門開發團隊	價值鏈體系

2.3.2 協同產品商務

2000年3月，3個距離遙遠的工程小組曾在7天內合作開發出一輛新型賽車。他們分別從美國波士頓(側重於動力系統)、德國漢諾威(側重於底盤)和日本東京(側重於車身)，利用Internet分享彼此的資訊與資源，在全球數萬人的見證之下，完成了此次產品開發案。上述即是運用CPC的著名實例【36】。

CPC是90年代末期出現的一類新式企業管理解決方案，其概念最早由Aberdeen Group於1999年10月提出，根據Aberdeen Group的定義：「CPC是指一類軟體和服務，它使用Internet技術，使開發成員在產品生命週期內互相地對產品進行開發、製造和管理，不管這些人員在產品商業化過程中擔任什麼樣的角色、使用什麼電腦工具、身處的地理位置或處在供應鏈的什麼環節。」由於Aberdeen Group是以當時已出現的一些軟體服務為基礎，結合企業實際需求提出此概念，因此其將CPC視為一項軟體或服務。但嚴格地說，CPC不僅是一套軟體系統或服務，更是一種企業管理理念的創新。它充分利用Web技術，增強企業間的協同合作能力，靈活適應動態的商業環境並提供產品全生命週期的資訊管理，能夠大幅改善離散型製造企業的核心業務過程，實現產品客制化以及快速推出創新產品。若從技術層面來看，CPC將成為下一代協同商務的基本框架，進而實現產品生命週期中各式e化解決方案邁向整合統一的目標【37】。

根據學者曾令衛博士的研究，CPC主要由協同、產品、商務三大方面的技術和理論發展、演化以及相互融合而來，如果用三維空間的座標軸來描述，那麼CPC實際就是三軸的交彙點，如圖2-17所示(修改自【37】)。以下分述圖2-17之三大座標軸：

- 一、產品軸：代表產品生命週期中各種解決方案的發展。從最初的單點解決方案，如：2D、3D CAD等，到多點之間的集合，如：採用通用特徵、零件實現設計、製造的集合等，再到統一資料模型，以主模型架構的方式，集合更大範圍的方案。這些集合的思路促進了PDM、ERP、CRM、SCM、EIP等資訊管理系統的出現和應用，各系統以不同角度，滿足產品生命週期中各種資訊和產品知識的管理需求。
- 二、協同軸：產品開發過程中合作關係的發展。從最初的缺乏交流，到企業內不同部門之間注意協調和溝通(同步工程概念)，再到強調不同企業之間的合作，直到現今跨企業、組織的全球性虛擬企業。由於客戶需求的充分表達均離不開協同工作

環境的支援，因此 CPC 提供了實現廣泛協同合作開發的基礎。

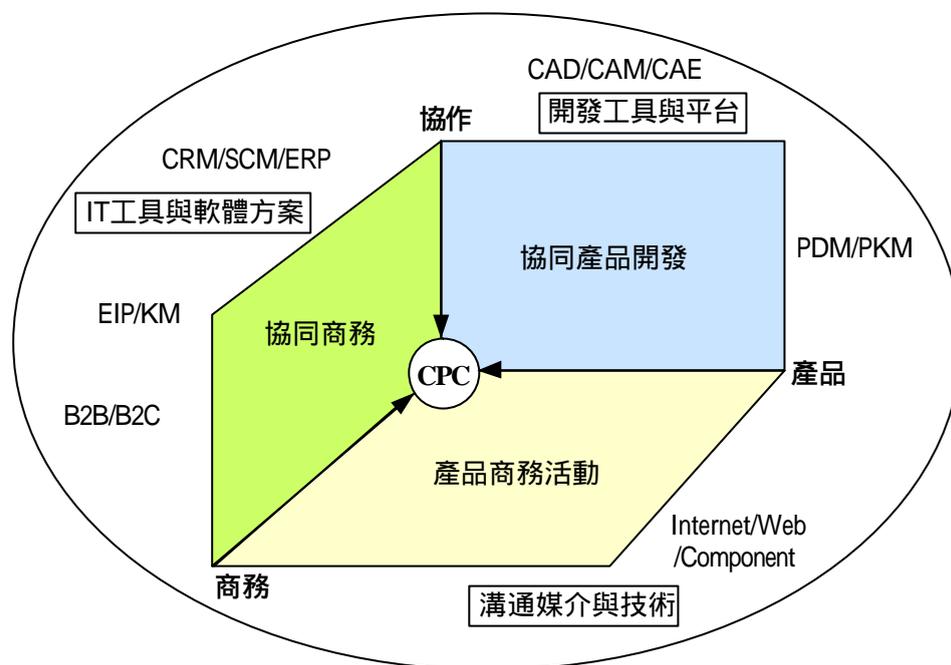


圖 2-17 協同產品商務發展演變

三、商務軸：商務可以看作具有商業活動能力的實體(如：企業、金融機構、政府機關、消費者等)之間，透過契約關係所進行的價值交換活動。商務最初大都透過人工與面對面的處理方式來完成，隨著網路通訊技術的演進，電子商務從最初的型錄展示、網路零售、線上交易等，直到現今企業間協同商務環境的建立，說明了電子商務不斷成熟發展，以及價值鏈體系間協同合作的趨勢。

協同產品商務的建立過程，可分為以下七個主要活動【38】，而整個活動的流程如圖 2-18 所示：

1. 挖掘商機：商業機會是一個相對的概念，如何搜集客戶的需求從中發現商機，並根據企業的策略和能力，確立與取得有效商機。
2. 制定策略：企業為了將商機轉化為利潤，根據商機和企業現有能力的分析，提出對合作夥伴的需求。
3. 發佈合作資訊、搜索潛在合作夥伴：通過有效渠道將協同合作的資訊發佈，另一方面則是可在此平台上搜尋潛在夥伴。
4. 與潛在夥伴開始接觸：與潛在夥伴進行初步的接觸了解與選擇。

5. 合作夥伴選擇與強化：按照需求選擇合作夥伴，並對整個合作聯盟的整體關係進行強化。
6. 合作聯盟建模：建立聯盟的合作模式。
7. 確立合作規範：確立企業間合作的產品模式、過程模式以及合作企業模式，企業間的合作過程中，一方面要確保合作企業的關鍵技術不會洩漏，另一方面則要保證企業間合作所需的必要資訊能夠交換與共用。

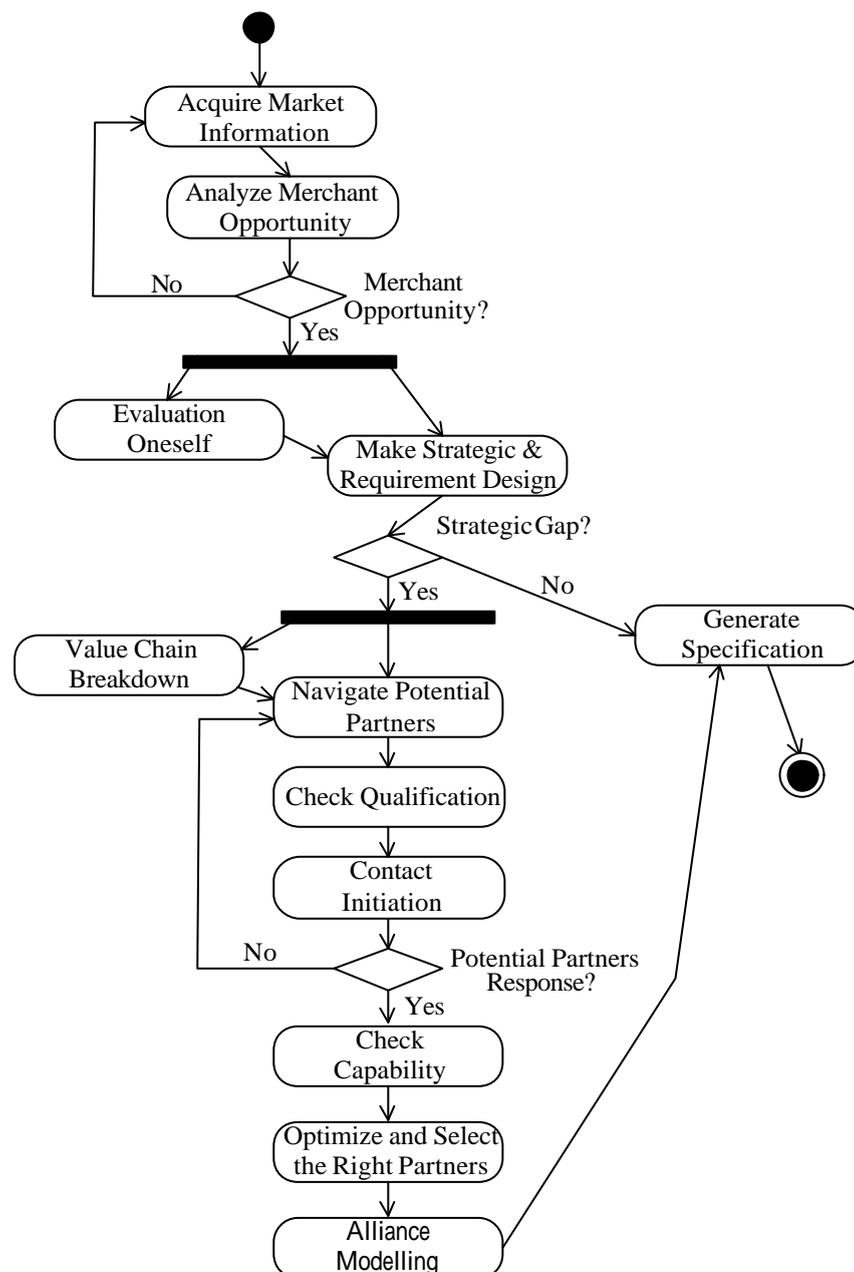


圖 2-18 協同產品商務建立過程(UML 活動圖)

PTC 參數科技的技術長及副總裁 James Heppelmann 認為協同設計的內涵為 3C，分別是創造(Create)、協同(Coordinate)及掌控(Control)，製造供應商除了接受上遊廠商所出之設計方案，更有加入自身設計的主控權。一個完整的 CPC 解決方案除了具備上述協同設計內涵之外，更需整合運用供應鏈管理系統、企業資源規劃系統以及客戶關係管理系統，如此才能建構製造企業之間的電子商務解決方案，如圖 2-19 所示【17】：

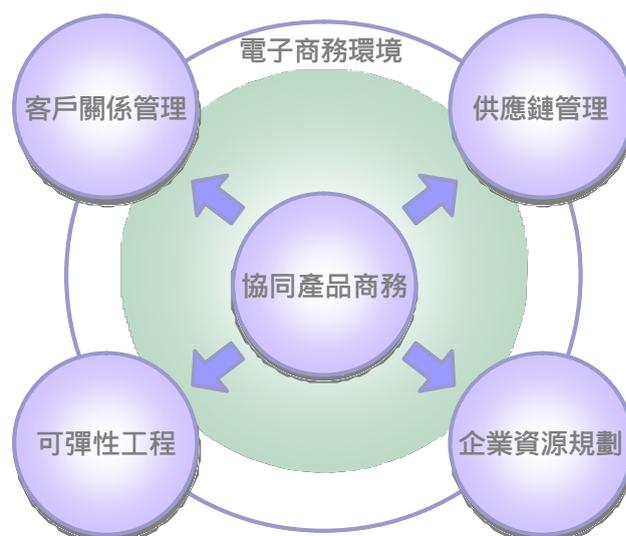


圖 2-19 CPC 發展電子商務架構圖

CPC 以 Internet 作為商務溝通平台，運用 ERP、SCM、CRM 等企業 e 化解決方案，實現產品全生命周期管理與全新的電子商務模式，其可帶給企業的效益如下【17】：

- 降低成本：透過即時的分析、修改與資訊的交換，製造商可以事先做好生產規劃，進而減少廢料、返工次數以及滯銷庫存等，同時避免出現代價昂貴的錯誤。
- 提高效率：因為 CPC 解決方案可以為製造商減少廢料和返工次數，所以很少會出現延誤情況，進而縮短了監視和追蹤失控況狀的時間。此外，由於企業決策能快速施行，因此準時交貨率大大提高，從而減少對人工的需求量。通過提高生產率，製造商甚至可以在產量增大的情況下，減少主管人數或避免增加僱員。
- 提高管理能力：員工可以致力於創新性的事務，而非專注於解決問題，更少的會議或更少的報告，代表著更高的效率和更完善的底線。
- 提高質量：透過嚴格的流程強化與整合，可以確保產品改善工作精確和準時的被執行，使得各部門人員將自己的實踐經驗融入到核心資料中，以便讓其他開發成員能夠快速參考與學習，進而提高設計與製造的質量。

- 提高核心競爭能力：透過與合作夥伴和供應商的商務協作，可以將非核心服務外包，整合與強化整體供應鏈與價值鏈，專注於增進企業核心競爭力。

近年我國製造業面對大陸低勞資條件的競爭，逐漸失去優勢，PTC 參數科技的技術長及副總裁 James Heppelmann 即指出，強化垂直產業價值鏈間的產品協同設計能力，乃是企業從 OEM 轉變為 ODM 的關鍵，唯有擁有最好的產品，以及產品的上市時間最短，台灣製造業才能在全球化競爭環境下生存，產業型態也應該從原本代工製造的 OEM 型態，發展成具備設計研發能力的 ODM。由於 ERP 系統的整合與管理，著重於製造與財會領域，而 SCM 主要負責製造物料上的整合與流通，因此如何整合運用 ERP、SCM 並配合客戶關係的管理，結合價值鏈上的所有廠商進行協同產品商務，使產業間以核心能力互相連結，發揮創新能力與提升產品價值，將是企業間共同創造市場容量及取得絕對競爭優勢的重要關鍵。

2.4 系統開發與架構之發展

自從 Internet 的應用快速興起之後，軟體技術的發展與系統架構的演進就逐漸由 Web 的應用所主導，各式開發工具無不新增加入 Internet/Intranet 應用程式開發的功能，也由於 Web 的應用範圍不斷快速擴展與成長，因此造就各種新式 Script 程式語言的誕生，例如：ASP、JSP、Perl 和 PHP 等。由於 Script 程式語言具備簡單、開發迅速、擁有常用函式等優點，因此目前已廣泛運用於各式平台，成為開發 Web 應用程式的主軸之一。

然而正因為 Internet 與 Web 的應用發展過於快速，使得不同的實作技術與解決方案往往遭遇無法整合的窘境，因此許多企業開始要求公司內部必須使用相同的開發工具、元件模型與 Web 應用程式，以便解決資訊管理系統無法整合運用的狀況。此種系統整合方案雖然確實可行，但通常僅能適用於企業本身，一旦跨出企業自身範圍，需要與其他上下游廠商相互配合時，除非公司屬於大型企業可以強勢要求下游廠商絕對地配合，否則無法整合的狀況便又再次發生。

為了避免無法整個的狀況不斷發生，資訊業界也逐漸開始形成共識、結盟合作，試圖訂定一個標準協定，用來解決 Internet/Intranet 及 Web 應用方案無法整合溝通之問題，讓資訊系統與資源可以藉由 Internet/Intranet 輕易地整合，而不必顧慮自身與其

他企業是否使用同樣作業平台、程式語言以及開發工具等，而這使資訊領域又邁入一嶄新紀元的標準協定便是網路服務(Web Service)。表 2-6(修改自【14】)說明了整個 Web 系統架構、程式開發，以及 Web 應用架構的發展過程：

表 2-6 資訊系統發展過程

演進項目	第一波	第二波	第三波	第四波
程式開發	資料處理 Data	程式語言 Code	物件導向 Data+Code	元件設計 Component Design
Web 應用	文字導向 HTML	程式導向 ASP/JSP/CGI	服務導向 Service/Data+Code	服務設計 Service Design
Web 系統架構	主從架構 Client/Server	三層式架構 3-Tier	Web 分散式架構 Web+3-Tier	Web 架構設計 Web Tier Design

表 2-6 中的程式開發從 60/70 年代以資料處理為中心的應用，到 70/80 年代由於各式程式語言快速發展，轉變為以程式語言為中心，到 90 年代開始盛行的物件導向開發模式，程式開發轉而趨向為以軟體元件服務為中心。Web 的應用則是從最初的靜態文字呈現(HTML)，到具備動態呈現效果與快速開發的 Script 語言時代，再到現今 Web Service 協定標準的出現，轉而趨向以網路元件設計服務為主。Web 系統架構從最初的前端瀏覽器與後端 Web 伺服器的主從架構，到增加應用程式伺服器(Application Server)的三層式架構，再到現今以全球資訊網路為基礎，建構出無數階層型態的 Web 分散式架構與 Web 架構設計。

從表 2-6 與上述之探討可發現，整個 Web 系統架構、程式開發，以及 Web 應用架構的發展過程事實上是相當雷同且相輔相成的，其中程式開發與系統架構約略經歷了三、四十年才發展到今日的地步，然而 Web 的應用在短短十年間便已演進革新了數次，可見 Web 的應用發展的相當迅速，並且是資訊科技發展不可避免的主流趨勢。未來系統開發與資訊科技的發展，將會朝向以網際網路為核心，具備更多元、更具彈性、以使用者需求為導向的服務模式(Service-Based)。所謂服務模式即是以滿足使用者各式需求為依歸，而服務者的責任則是為使用者設計出最適當的服務流程與服務項目，讓使用者可以儘管享受服務的內涵與品質，而不必煩惱各個服務項目的真正供應者以及取得的方式為何。資訊科技要達到服務的模式，則全球軟體供應商勢必要訂立

彼此能夠整合溝通的標準協定，傳統系統架構與開發模式也將作局部的調整或整體的變革，如此才能因應整個資訊發展的趨勢，而本文將於後面章節以開發模式、系統架構的角度繼續探討此一趨勢的形成背景以及應用領域。

2.4.1 系統開發模式的優缺點

系統開發模式發展至今已有相當多的理論及方法出現，其發展演化的主要目的在於降低系統開發成本、提高軟體品質以及縮短開發時程等，茲將主要的開發模式與其優缺點整理於表 2-7(整理修改自【39】)。

表 2-7 系統開發模式的演進

模式名稱	優點	缺點
瀑布式	階層式開發 任務明確	對使用者需求不了解 系統再用性低、風險高
擴展模式 (漸進模式)	快速產生系統雛型 改良瀑布式於系統完成前任何產品可使用之缺點	系統必須具備高度彈性 後續版本可能曲解系統目的 系統版本功能不完善
螺旋模式	降低系統開發風險 減少開發費用	開發人員必須具備風險評估知識與技術 風險因素需完善考量
結構化模式	使用資料結構圖形描述系統 系統架構、功能明確	結構太過嚴謹複雜 使用者需求增加時，修改開發成本龐大
物件導向模式	系統軟體再用、資訊封裝 降低系統建置、維護複雜度	系統分析、設計較為困難 系統開發前置時間較長
元件式	降低系統設計、管理複雜度 降低維護修改成本 平行開發、縮短時程 系統更具彈性、降低整合複雜度	需使用標準或特定的技術規範 對物件導向技術與系統功能需求充分了解 採用非一致元件技術將無法整合溝通

表 2-7 主要列出六種系統開發模式，茲簡述如下：

1. 瀑布式：此模式為最傳統的系統開發模式，屬於階層式的發展模式。它將軟體開發的程序主要劃分為六個階段，分別是系統工程、分析、設計、程式撰寫、測試與維護，如圖 2-20 所示。此開發模式必須先完成現有階段的工作項目才能繼續往下一階段進行，由於開發人員於系統分析階段時，往往無法完全了解使用者需求，因此使得後續階段的處理成本大幅增加，此外由於在軟體開發完成之前並沒有進行階段性的系統驗證，因此也使得系統開發的風險偏高。

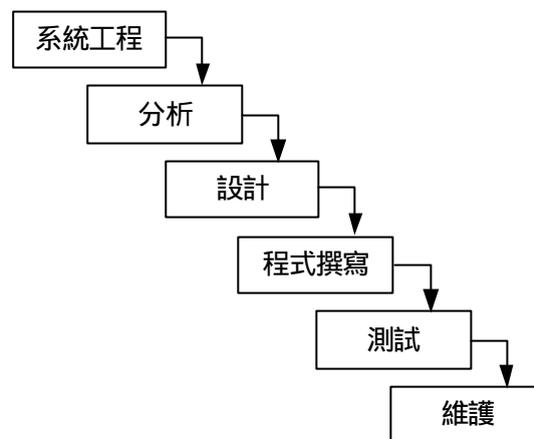


圖 2-20 瀑布式系統開發

2. 擴展模式：此模式強調使用者導向的系統開發，由學者 Gilb 於 1970 年初所提出。開發者先根據使用者需求開發出一系統雛型，然後再根據使用者階段性的測試與驗證結果，逐步地擴展改良系統，直到此階段性的系統完全滿足使用者需求為止，然後再進行下一階段的循環，下一階段的系統均包含上一階段所開發的功能，且每一階段的系統開發均包含完整的系統分析、設計、建置與測試，如圖 2-21 所示。

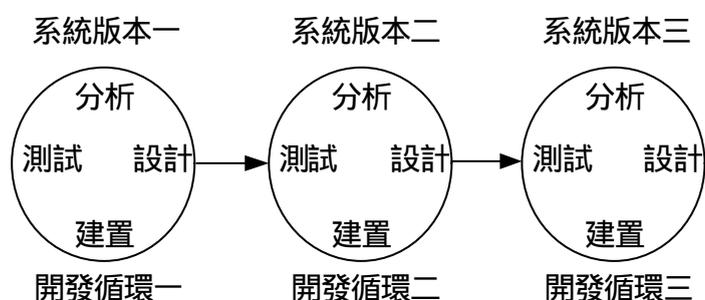


圖 2-21 擴展式系統開發

由於擴展模式以漸進的方式逐步開發系統，因此可避免傳統瀑布式無法作階段性測試與使用的缺點。為了達到階段性開發及完成後續整合工作，擴展模式必須具備足夠的彈性，以及採用鄉湯開法的系統架構，才能支援往後系統開發的需求，此外倘若在初期版本，對使用者需求考量不夠完整，則後續開發階段很有可能會曲解系統原本開發的目的與使用者真正的需求。

3. 螺旋模式：螺旋模式是 Boehm 於 1986 年所提出，此模式採取瀑布模式及擴展模式的優點，並增加了風險分析與評估，主要開發模式的重點在於不斷循環地查核與規劃，如圖 2-22 所示。在開發過程中，當一個階段完成時，就必須為下一階段明確定訂目標、風險評估後再選擇是否執行。採用螺旋模式的主要優點在於能夠建立雛形系統，查看是否與使用者的需求一致，此外由於每一階段均進行風險的評估與下一階段開發的考量，因此能夠有效發現錯誤並減少開發成本。此模式的缺點即在於開發人員須具備更多的專業知識技能，如此才能有效進行評估與規劃，另外若有風險因素未被及時考慮發現，則將對後續階段造成開發上的困難。

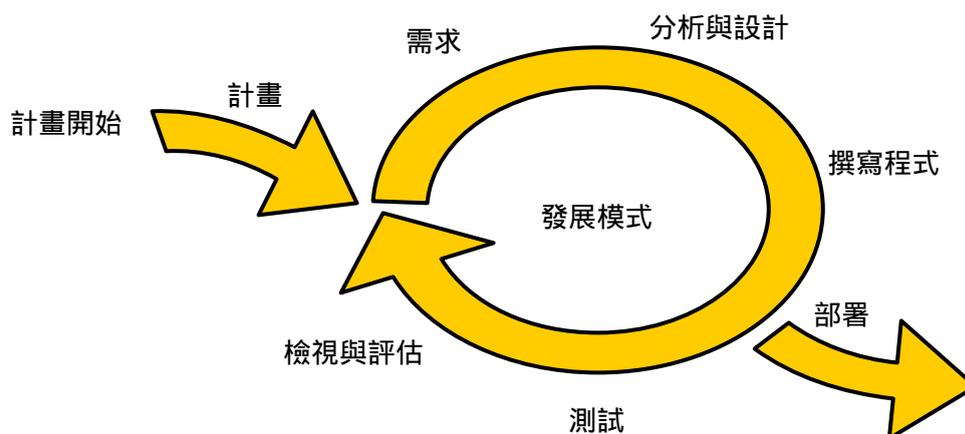


圖 2-22 螺旋式系統開發

4. 結構化模式：此模式是先進行系統的分析，以適當的資料結構加以描述，然後再建立相對應的執行流程，進而產生程式邏輯的架構。結構化的系統開發模式採用資料流程圖及結構表來描述系統的功能面；使用實體關係圖及狀態轉換圖描述系統的資料面和時間面。由於使用者需求的不斷增加，致使軟體系統的複雜度與變動性不斷增大，結構化模式的開發方法也愈來愈受到限制，一旦使用者需求產生任何變動時，結構化系統的變動成本將大幅增加，整個系統品質也會相對下降。

5. 物件導向模式：物件導向模式主要運用物件(Object)的四項基本特質來進行系統的開發，分別是(1)封裝(Encapsulation)：將物件的內部隱藏起來，只提供外部的特性供其他物件存取，以達到保護內部運作的目的。(2)繼承(Inheritance)：物件間以現有類別為基礎，組合成其他的物件。物件會擁有父類別的所有特性與功能，因此可以減少重覆建立相同類別功能，進而達到程式再用的目的。(3)抽象化(Abstract)：所謂抽象化是擷取物件中所需的性質，同時忽略不相關的細部性質，進而降低整體系統開發的複雜度。(4)多型(Polymorphism)：使用同一種方式處理許多不同型態的物件，它允許各種不同的運算使用相同的名稱，而期望的運算方案則需依據所使用的物件作動態地選擇。採用物件導向的優點在於可以使軟體系統功能再用、降低系統建置與後續維護的複雜度。缺點則是系統分析與設計較為困難，導致系統開發前置時間較長。
6. 元件式：元件式系統開發指的是將具備特定功能且可獨立運作的程式單元，相互連結成為應用系統，使得不同程式設計師所開發的軟體元件能彼此呼叫使用。而所謂元件是指一個透過標準介面以供存取相關服務的軟體，其具備三種特性：(1)自我包含：又稱為內聚性，元件專注於單一、明確的目的。(2)完成特定功能：元件具有特定的功能用來解決特定的問題。(3)定義完整的介面：元件提供介面與外界溝通，透過這些介面使用元作內部的功能。採用元件式系統開發的優點在於可降低系統設計與管理的複雜度、提昇系統彈性與延展性，以及降低開發成本等，缺點則是元件開發需遵循特定標準，否則不同類型的元件無法整合溝通。

2.4.2 元件式系統

近年來網路資訊科技發展與應用一日千里，使得企業對軟體系統的彈性與效率要求遠勝於過去，如今企業不僅要求軟體系統要能支援分散於世界各地的經營管理，更需整合上下游廠商資訊系統，用以累積企業營運知識及強化資訊的即時分析與管理。由於使用者的需求日益增多，造成系統模組不斷快速膨脹，模組間錯綜複雜的連結關係，使得後續維護管理工作極為困難，浪費龐大的系統開發資源。因此，傳統系統開發模式已無法解決現今企業的問題與需求，在有限的開發預算下，企業必須採用新的開發模式並設法降低建置成本與風險，以及提高整體系統效率與軟體生產力。

物件導向技術是產學界一致認為用來解決上述企業資訊系統問題與需求的不二法門，而以物件導向技術為基礎的元件式系統開發，在相關標準制定與支援廠商逐漸增加的情況下，已經成為一個具體可行的解決方案。物件與元件導向開發模式的主要差異在於物件導向的分析與設計是以物件為中心，並強調封裝與繼承等物件特性，藉由反覆來回地以各種靜態、動態的角度分析與設計，定義出物件的抽象類別和屬性，以及隸屬此類別的方法為何，最終分析設計出一符合彈性延展性、可重複使用的系統，而元件導向除了承襲了物件導向的開發觀念與主要特性之外，更進一步的改進物件導向的潛在缺點。

如表 2-8 所示(修改自【39】)，由於物件以類別繼承為主，因此若物件要重複使用或修改功能時，必須先了解物件內部的物件導向機制與原始碼，這增加了處理上的困難，以 Notebook 為例，如圖 2-22 所示(本研究)，當採用物件導向之 Notebook 要更換其他廠牌滑鼠時，必須先了解 Notebook 內部對於滑鼠這項裝置的相關處理機制為何，才能更換其他廠牌的滑鼠。而由於元件以介面繼承為中心，因此不需了解元件內部的運作機制，只需了解元件介面的內容即可重複使用元件，如圖 2-22 中的元件導向式 Notebook，只要符合相同的介面規格，使用者不但可任意更換各廠牌的滑鼠，更可替換滑鼠裝置，改為使用相同介面的鍵盤。另外，由於元件的開發並無限定程式語言，且對於系統的分割較為適中，因此元件比物件更具彈性，同時也降低了整體系統的複雜度。

表 2-8 物件與元件的差異

比較項目	物件	元件
重複使用性	強調類別(Class)與實作繼承(Implementation Inheritance)	介面繼承
語言相依性	有	無
就管理層面與分散機制而言	系統切割粒度 (granularity)太小	系統切割粒度恰當

歸納採用元件式系統開發的主要優點如下：

- 可將系統切割成較易設計的功能元件，藉以降低系統設計的複雜度。
- 各元件功能彼此獨立，可平行開發、縮短開發時程。
- 可以發展出更具彈性的系統。

- 分別掌握各元件功能之開發與設計，使管理工作更為容易。
- 可依循設計標準組裝，利於元件外包或購買現成元件使用。
- 採用標準的元件介面，可使不同系統相互整合操作。
- 可單獨對個別元件進行測試，減低系統整合測試的複雜度。
- 更容易且低成本地維護軟體元件。
- 利用元件間的組合，有效建構系統延伸功能。

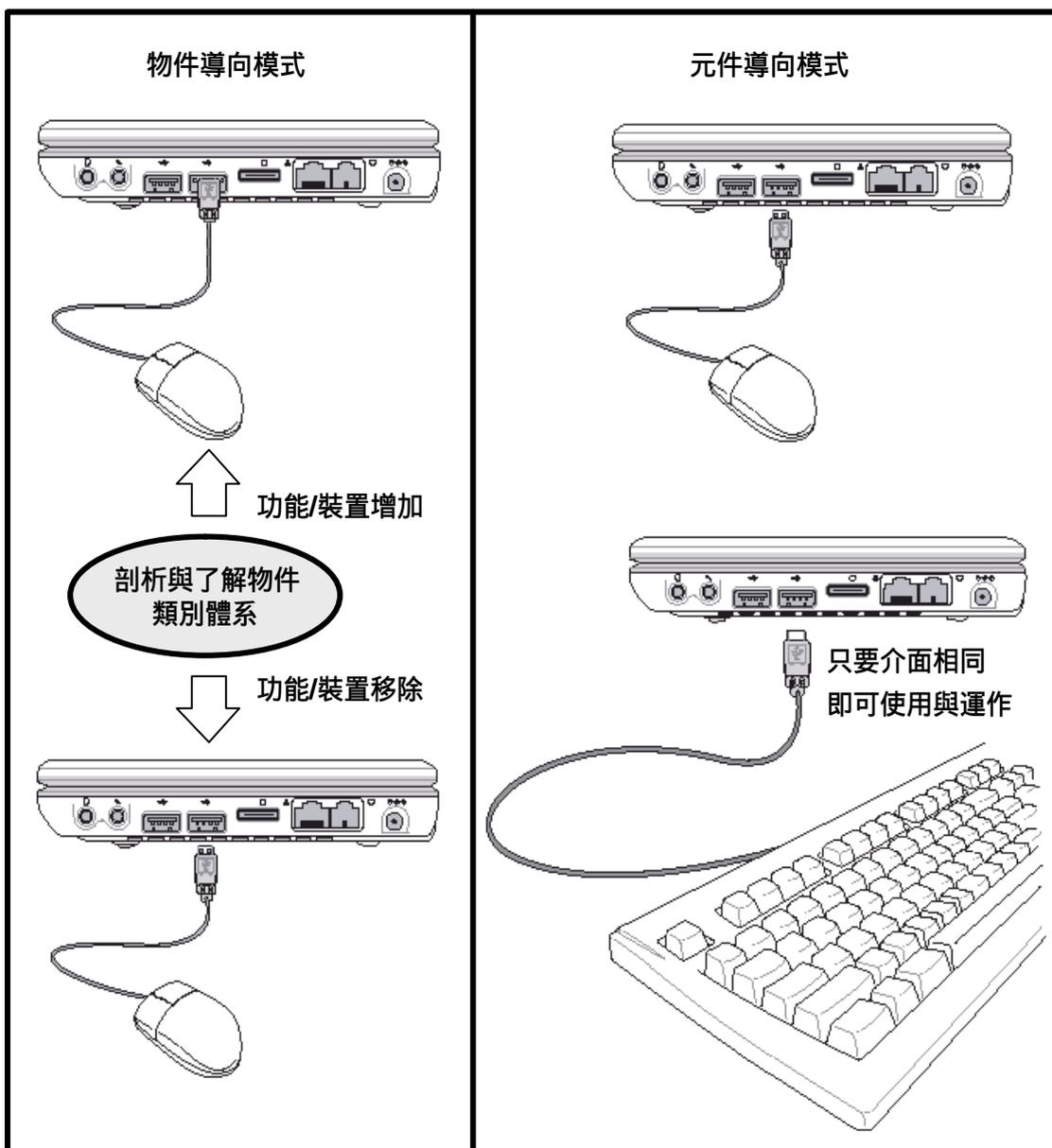


圖 2-23 物件與元件軟體再用與延展差異

元件的製作是以「軟體再用」為目標，利用組裝的方式來建構系統，這種開發模式使得系統具備修改、更新及允許增加應用功能等優點，可大幅降低成本、提高生產力與品質【12】。目前主流的元件技術規格為 CORBA、COM+與 EJB，其比較如表 2-9 所示(整理自【13,14】)。

表 2-9 主流元件模型比較表

比較項目	CORBA	COM+	EJB
主導單位	IBM&OMG	Microsoft	Sun、IBM etc.
標準訂定	透過一定程序 經由成員認可	微軟主導	透過一定程序 經由成員認可
技術複雜度	高	較低	較低
延展性/通透性	高	較低	高
安全性	高	較低	較低
實作規格分歧	是	無	是
執行效率	高	高	較低
實作程式語言	不限	不限	Java

表 2-9 的三種元件模型技術各有其優缺點以及應用領域，例如國內中小企業多使用 COM+元件模型，國外則以 EJB 為主，至於 CORBA 則是發展最久、功能最強的元件技術，不過由於 CORBA 過於複雜，所以除非是超大型應用系統(例如：銀行總帳系統、電訊系統等)，否則一般企業使用 COM+或 EJB 即可應付所有系統開發的功能需求。

目前元件的發展以 COM+與 EJB 為主，CORBA 的應用已逐漸退出市場。COM+技術的發展完全掌握於微軟公司，其優點在於程式語言只要符合 COM+的技術規格，便可確保 COM+元件間能夠整合溝通，但缺點即是 COM+對於其他系統平台的相容性差，難以達到跨平台的需求；EJB 元件技術主要由 SUN 與 IBM 所主導，藉由 Java 語言跨平台、免費使用、開放標準等絕對優勢，使得 EJB 可以形成完整的跨平台分散式系統，符合現今環境多元多變的需求，但由於 SUN、IBM 等廠商對於 EJB 的技術規格僅是功能規範而已，每一家廠商對於 EJB 的實作方式並不一致，使得不同廠商所開發出的 EJB 元件並不能完全確保能夠彼此溝通，加上 EJB 目前僅能透過 Java 程式語言開發，因此使用 EJB 元件依然存在許多問題需要解決，如此才能適合應用於企業實際商務環境之中。

2.4.3 分散式系統架構

以往主從架構的運作主要是由用戶端執行前端程式(Client Program)，然後連結到後端資料庫伺服器中進行資料的存取或者商業邏輯的運算，如圖 2-24 所示。

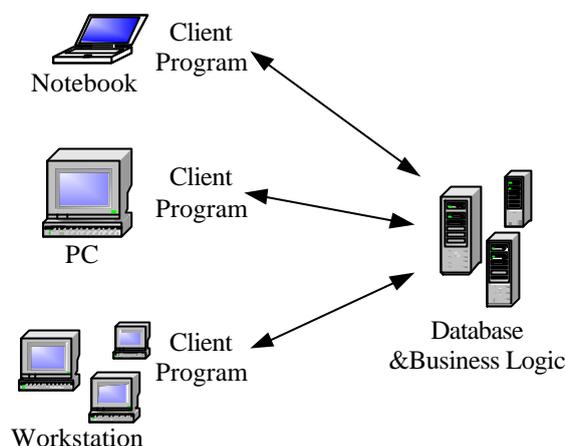


圖 2-24 主從系統架構圖

基本上主從系統架構在一般小型或少數人使用的系統上，可以執行的很流暢，但其缺點在於主從架構經常將企業邏輯撰寫於用戶端應用程式之中，因此一旦企業邏輯需要改變時，則所有用戶端的應用程式均需要變更，造成後續維護管理上的困難，因為如果一家企業有 100 台前端電腦，則必須花費相當龐大的人力成本去作維護與更新的動作。有些企業會將主從架構的企業邏輯撰寫於後端資料庫伺服器中，雖然這可降低維護前端程式的難度，但卻也增加資料庫伺服器的負擔，不但整個應用程式系統綁死在特定的資料庫上，更嚴重拖累資料庫的執行效率。

近年來 Internet/Intranet 的不斷興起，使得企業的經營模式與資訊系統架構逐漸改變，如今企業大量導入各式資訊系統及增加系統功能，使得傳統主從系統架構，再也無法負擔企業龐大的需求。為了因應全球客戶對於企業及產品相關資訊的需求，企業開始將資訊管理系統的邏輯功能，重新撰寫並發布於 Web 伺服器中，讓客戶或公司人員能夠藉由一般瀏覽器，透過網際網路直接查詢與取得企業營運相關資訊。除此之外，由於企業對於資訊系統的執行效率與穩定性要求日益嚴峻，因此開始採用所謂的三層式系統架構(如圖 2-25)，將整體資訊系統劃分為展示層(Presentation Tier)、商業邏輯層(Business Tier)，以及資料層(Data Tier)。前端的展示層通常稱為「Thin Client」，主要負責使用者介面的呈現，以及簡單的邏輯處理等，如瀏覽器便可視為展示層的前端應

用程式，而瀏覽器中 VB Script、Java Script 等稿本語言則可處理相關的程式邏輯。商業邏輯層主要負責企業物件的執行與交易控制(Transaction Control)，由於絕大部分的邏輯運算均存放於此，因此當邏輯變動時，系統管理者並不需要到各個前端電腦維護前端應用程式，僅需更改邏輯層的企業元件即可，如此不但可大幅降低管理維護的成本，更可重複使用企業元件，提高軟體功能的再用率，更可增進系統執行效能與穩定性，其中企業元件模型以 COM+、EJB 為代表。資料層主要負責資料的管理工作(查詢、新增、修改、刪除等)，由於不像主從架構需額外處理商業邏輯，因此執行效能與系統安全性較高，其中資料庫管理系統以 MS SQL Server 與 Oracle 為代表。

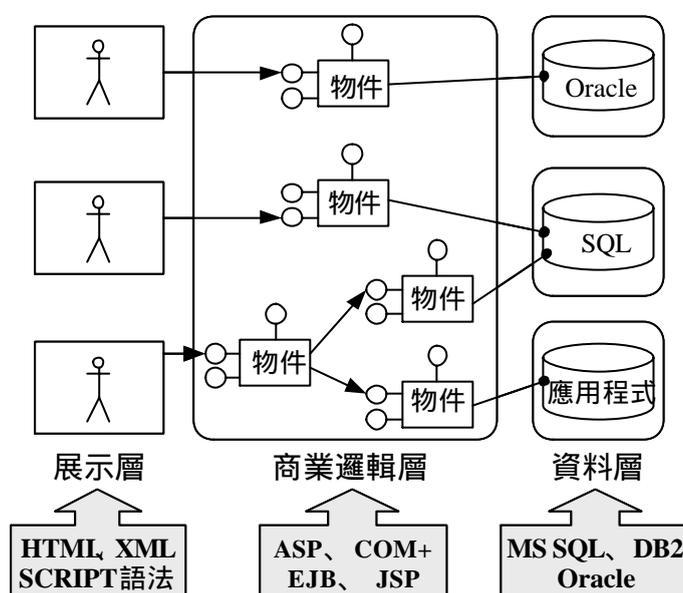


圖 2-25 三層式架構與常見應用技術

現今企業資訊管理系統(Management Information System; MIS)除了提供企業內部營運事務支援之外，更需處理網際網路上的各式資訊需求，如前述所言，原本採用主從架構的企業通常會增加一台 Web Server 用來解決對外之問題，但這樣的架構存在許多問題，那就是對外的資訊系統必須採用 Web 技術重新包裝企業邏輯，而並非與企業內部系統採用同一來源的企業邏輯，造成系統功能的重覆開發，此外 MIS 人員必須同時管理對外對內兩套系統，不僅提高系統後續維護管理的難度，更增加整個系統的運作成本。因此，企業開始思考利用三層式系統的概念，設計與規劃符合企業內外使用且易於維護管理的分散式系統架構，如圖 2-26 所示。圖 2-26 一般稱之為分散式 Web 系統架構，前端展示層分為兩大類，分別是瀏覽器以及傳統前端應用程式，瀏覽器主

要應用於企業對外資訊的呈現，而企業內部則使用傳統前端應用程式。架構中的商業邏輯層包括 Web Server 以及 Application Server 兩大部分，Web Server 負責接收與傳送外部環境的資訊需求，而 Application Server 主要執行企業的商业邏輯與法則。在此架構中，Application Server 內的企業邏輯功能同時提供給內部系統與 Web Server 所使用，因此可統一管理與運用企業邏輯，一方面降低管理的複雜度，另一方面則可避免系統功能重複開發以及資源的浪費。此外，由於此分散式 Web 系統架構提供了容錯能力(Fault Tolerance, 【註 1】)與平衡負載(Load Balancing, 【註 2】)機制，因此可大幅提高系統的穩定性與執行效率。

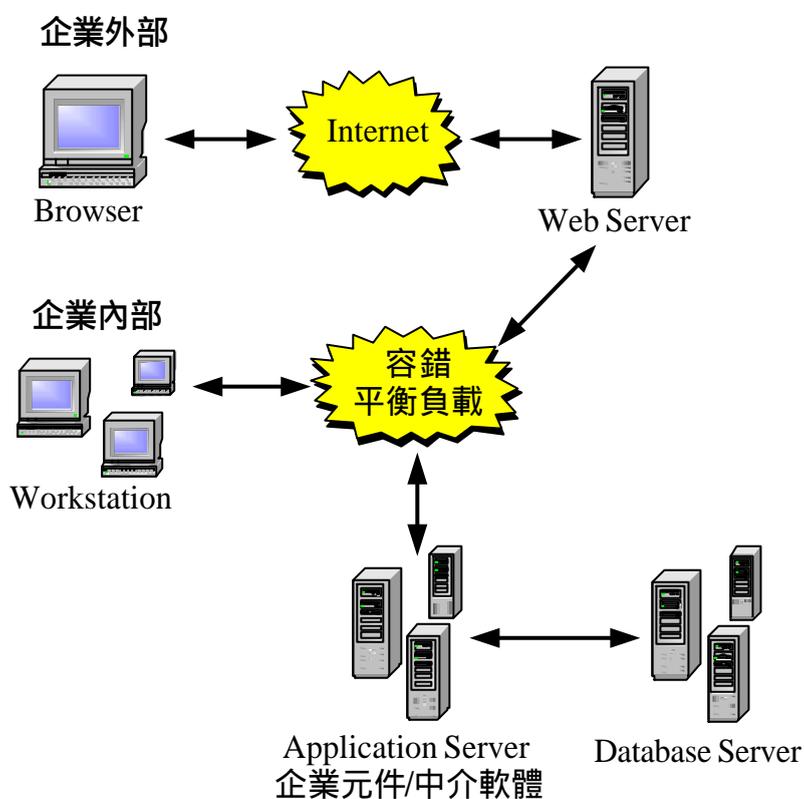


圖 2-26 企業分散式系統運用架構

註 1：讓應用程式伺服器在多台機器上執行，用戶端程式可連結至任何一台應用程式伺服器要求服務，若應用程式伺服器發生任何問題而無法繼續執行時，用戶端程式將可立刻連結到其他機器中的應用程式伺服器，繼續要求新的應用程式伺服器提供服務

註 2：指的是當用戶要求應用程式伺服器提供服務時，平衡負載機制會自動分配不同的用戶端程式到每一台應用程式機器中，以便平衡每一個應用程式伺服器的負荷。

綜合整理採用分散式系統架構的主要優缺點如下【40】：

主要優點：

- 高執行效率。
- 高穩定性。
- 高延展性。
- 可具備容錯能力。
- 可具備平衡負載能力。
- 可使用各種 Pooling 技術，例如：Database pooling、Object pooling、Resource pooling 等，增加系統執行效能，減少系統資源浪費。
- 可讓開發人員以企業物件的角度開發系統元件，藉以提昇系統功能再用率。
- 商業邏輯層的企業物件可提供企業內外系統使用。
- 可將企業物件統一管理，降低系統維護之難度。
- 可減少資料庫連線數目，進而大幅節省資料庫連線授權的金額。

主要缺點：

- 必須增加一台以上的應用程式電腦伺服器，增加硬體預算費用。
- 若對系統的效能要求較高的話，則必須將 Web Server 與 Application Server 分別裝置於不同的電腦伺服器之中，也就是需要購買兩台以上的伺服器，以便進行容錯與平衡負載的機制。
- 開發人員必須具備分散式系統開發的相關技術與觀念，開發導入門檻較高。
- 開發人員必須具備物件導向或者元件導向的開發觀念。

2.4.4 Web 應用程式的演進

Web 的應用與發展是近年來成長最快速的資訊技術之一，絕大部分企業內外系統，例如：ERP、SCM、PDM 等，均已具備 Web 的相關功能，由此可見 Web 技術的重要性。基本上 Web 應用程式大致可分為五種型態，分別是 1.CGI (Common Gateway Interface) 2.API (Application Program Interface) 3.Script 4.Script 結合中介軟體 5.Web Service，茲將此五種型態的優缺點，以及系統主要流程分述如下(修改自【14，41】)：

1. CGI 型態：

CGI 型態是最傳統的 Web 應用程式，同時也是最沒有效率的型態，其主要原因是 CGI 程式屬於 EXE 型態的應用程式，無論是啟動、執行、連結資料庫等動作，均會大量消耗作業系統資源，且無法使用絕大部分的 pooling 技術，因此一旦前端網站使用者人數增多時、資料庫存取動作頻繁，則系統執行效率將大幅下降，甚至造成整個系統當機，如圖 2-27 所示。

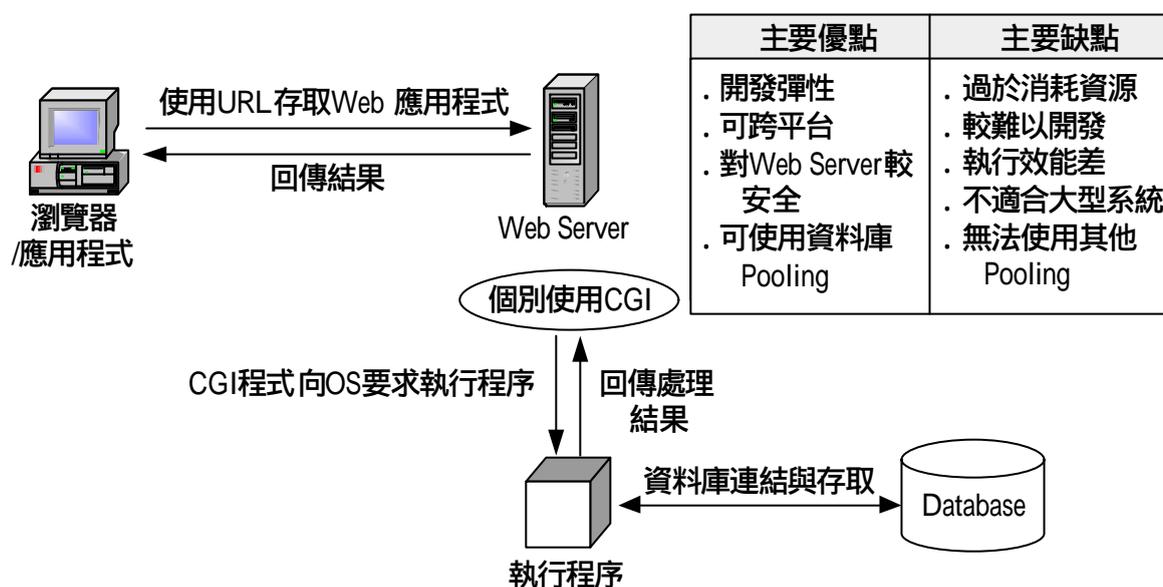


圖 2-27 CGI 型態 Web 應用程式

2. API 型態：

API 型態的 Web 應用程式主要是將系統功能與企業邏輯均撰寫放置於 Web 伺服器中，這樣的好處是可以降低應用程式的啟動 載入系統消耗成本，執行效率將比 CGI 型態的應用程式要高，但缺點即在於倘若 API 型態一旦發生錯誤時，將會造成整個 Web 伺服器執行不正常，使得系統當機的風險增加。此外 API 型態的應用程式依然需要作業系統提供執行緒來處理前端要求，因此倘若需要存取後端資料庫時，應用程式仍然必須針對每一次的前端要求，進行個別的資料庫的連結動作，使得資料庫的連線成本還是無法降低，如圖 2-28 所示。

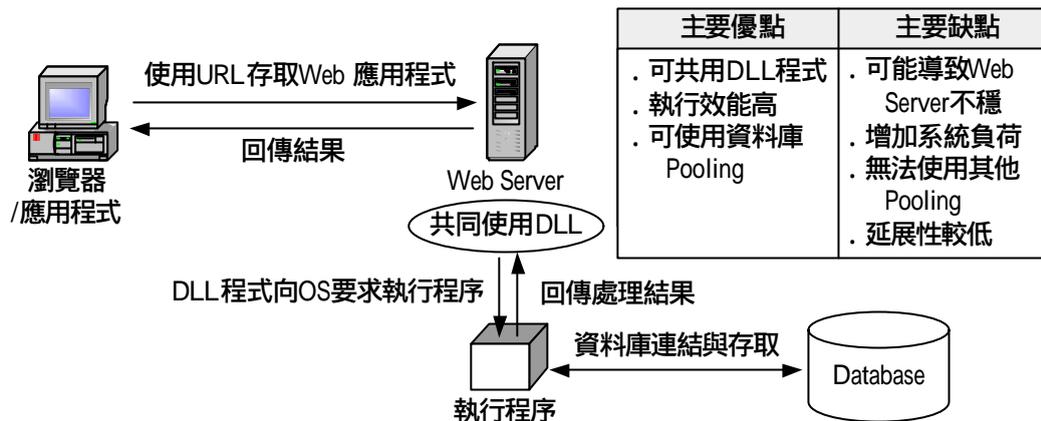


圖 2-28 API 型態 Web 應用程式

3.Script 語言型態：

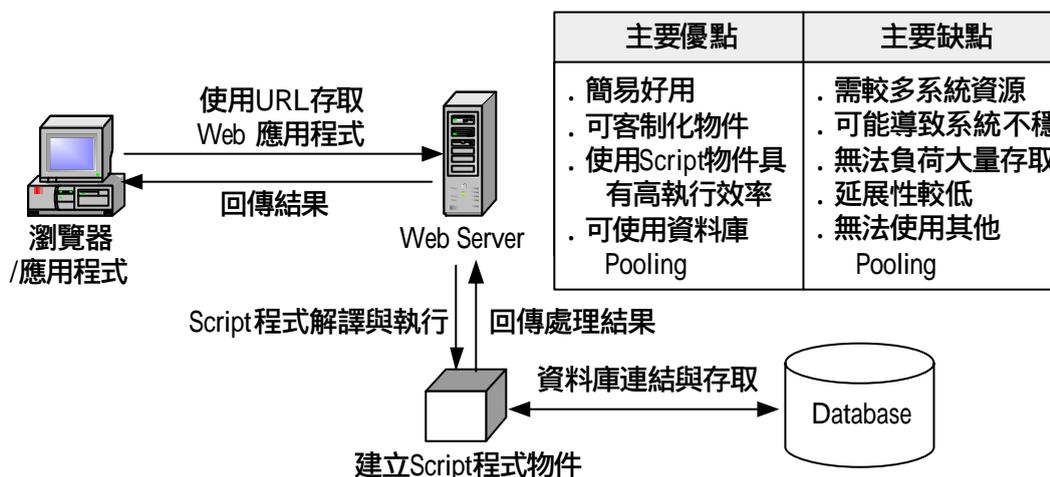


圖 2-29 Script 語言型態 Web 應用程式

Script 語言型態的 Web 應用程式具備簡易開發的特性，只要懂得運用 Script 語言所提供的各式物件介面，即可呼叫與使用物件的功能，無須自行撰寫許多低階的處理機制。然而，採用此型態的缺點在於當 Script 型態的 Web 應用程式使用各種物件功能時，物件會不斷重覆地被建立與釋放，因此造成執行時間的浪費。另外，由於 Script 語言屬於直譯式語言，只有當被執行時才會開始進行解譯的動作，因此執行效能會比一般編譯過的程式語言較低，雖然採用客制化 Script 物件可解決上述之問題，但由於客制化 Script 物件是執行於 Web 伺服器中，因此會提高 Web 伺服器出現錯誤或當機的风险，此外若將整個企業邏輯均放置於 Web 伺服器中，將會造成整個系統功能的延展性與再用性降低。

4. Script 語言結合中介軟體型態：

此種型態的 Web 應用程式主要結合 Script 語言簡單快速開發的特性以及中介軟體元件的各式優點。於此架構中，Web Server 上的 Script 程式語言僅負責傳送/接收前端程式的需求，以及簡單的邏輯運算，因此並不會造成 Web Server 龐大的負擔，而企業所有的商業邏輯程式碼則均執行於中介軟體元件環境中，藉由軟體元件環境所提供的各式 Pooling 資源、容錯機制、平衡負載機制，以及編譯過的元件程式碼等，可讓整個 Web 應用程式的執行效能、穩定性與延展性大幅提高。雖然此架構所需分析與設計的時間較長，但良好完整的各式支援與高執行效能的表現，勢必為現今企業開發電子商務與中大型系統所採用的主要系統架構，如圖 2-30 所示。

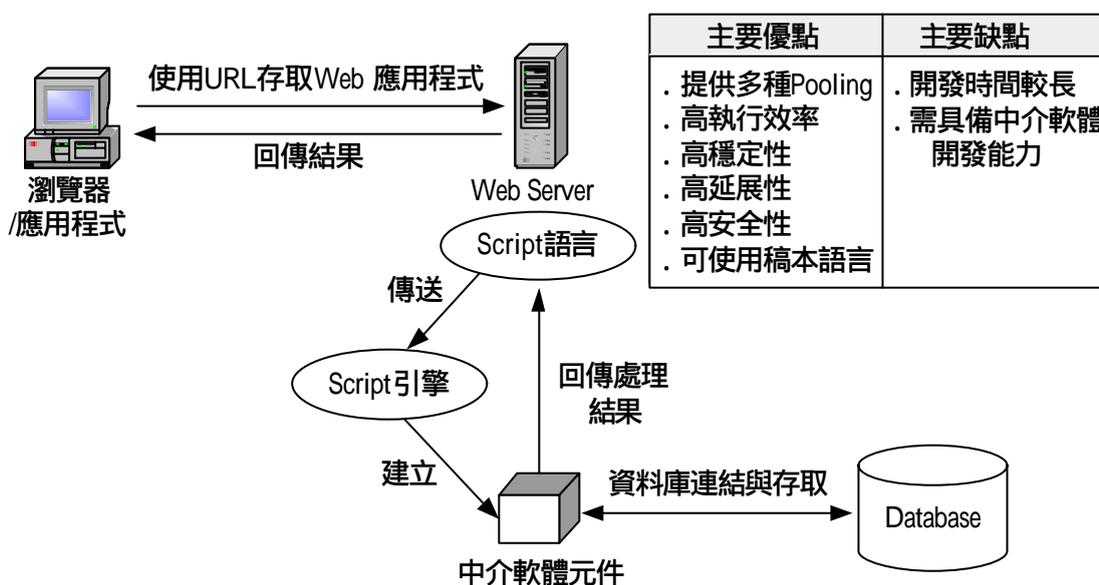


圖 2-30 Script 語言結合中介軟體型態

5. Web Service 型態：

Web Service 是近年來資訊界的一項重大革命，其應用與發展十分快速，但整體應用架構均大致如圖 2-31 所示。如圖 2-30，Web Service 服務主要存放於 Web 伺服器中，其可以是先前所提的 CGI、API 或者 Script 語言型態。在此架構中，Web Service 主要負責提供標準的遠端程序呼叫與回應，絕大部分的邏輯運算與資料庫存取動作均交由後端的中介軟體與資料庫伺服器負責，因此除了執行相關 Web Service 協定使得系統效能會稍微降低之外，其餘均與「Script 語言結合中介軟體型態」Web 應用程式相

同，同樣可以擁有各式完整的支援與分散式系統的優點，而採用 Web Service 型態的主要目的在於利用 Web Service 標準協定的特性，讓應用程式或系統功能達到跨平台、跨程式語言，以及適合各式前端裝置使用的絕對優勢，如圖 2-31 所示。

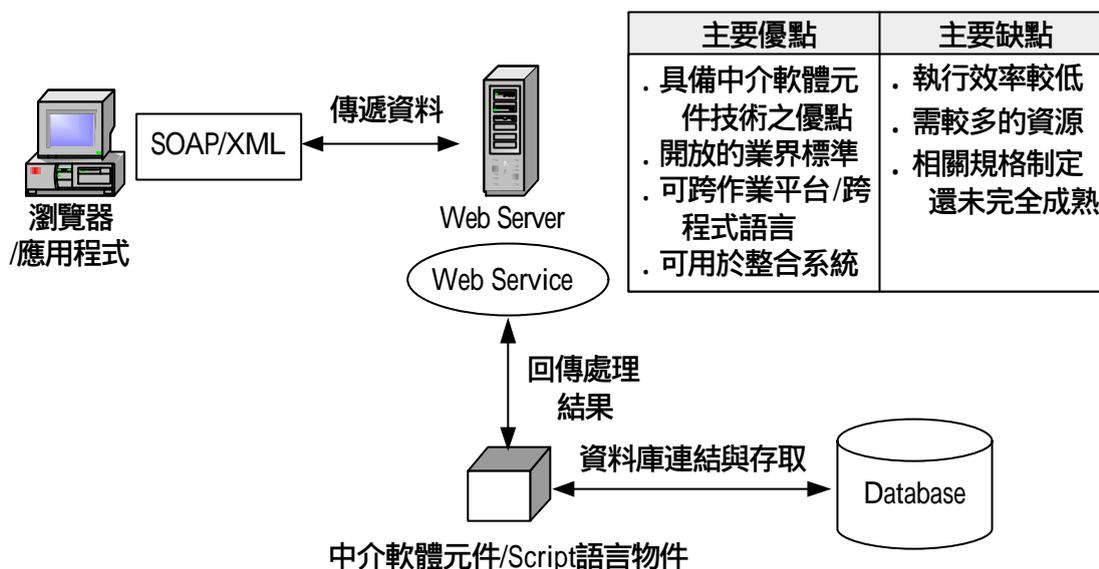


圖 2-31 Web Service 型態 Web 應用程式

2.4.5 Web Service

Web Service 是一種觀念，其觀念的形成與相關技術的誕生，主要來自兩大因素：

1. WWW 的發展：

根據統計資料顯示，全球網站的數量於 1998 年約有 3 百多萬個，而到了 2002 年卻已高達約 4 千萬個，可見自 1991 年 WWW 發明以來，每年均以驚人的速度成長與發展。WWW 的發展使得全球資訊內容不斷倍增，人們可透過標準的方式，取得全世界各式各樣的資訊需求與服務。然而，既然人們可透過標準方式取得全世界各種資訊服務，那麼倘若應用程式也可如此的話，則人類於 WWW 上取得資訊服務的所有操作過程，均可以利用應用程式將其自動化處理。例如：倘若您今天想買一本書，基本上您只要填寫送貨地址、書名以及選擇付款方式等，然後啟動買書的 Web Service 應用程式，則此 Web Service 應用程式將自動替您搜尋全球的網路書店，甚至於會幫您進行比價、議價以及處理所有金流、物流、資訊流等

商務流程，最終則確保書會由專人送達到您手中。上述的例子，明顯表達出 Web Service 的主要精神在於流程與資訊的自動化處理機制，以 WWW 上各式服務的角度來說，Web Service 試圖讓這些服務轉換為標準的軟體元件，這些軟體元件可透過標準的方式達到跨平台、跨程式語言、跨任何裝置無阻礙地使用，並且可相互結合成另一個功能更強大的 Web Service 軟體元件，高度自動化地滿足 WWW 上各式服務需求。

2. 企業資訊整合：

90 年代各種元件技術不斷發展(例如：COM+、EJB 等)，為了便是讓資訊系統更有效率，能夠重複使用企業邏輯程式碼，進而增加企業系統間的整合程度。然而，每一種元件技術的發展均希望擁有百分之百的市場占有率，進而達到資訊溝通無障礙、全球統一的目標，因此元件發展的過程並不注重是否能與其他元件整合溝通的問題。然而在現今網路商務環境中，要求所有企業、客戶、供應商等均使用同一種類的元件模型技術，事實上是不切實際且不大可能的，因為就算是採用同一類型的元件模型技術，也很有可能會因為不同軟體廠商的實作，而產生無法整合溝通的問題。為了讓不同種類的元件可以整合溝通，以往的解決方案是使用特殊、昂貴且複雜的橋接技術(Bridge)進行轉換，如圖 2-32 所示【14】。雖然使用橋接技術可解決不同元件模型整合溝通之問題，但在成本、效益、維護管理等考量之下，橋接技術最終不是一個良好的解決方案。

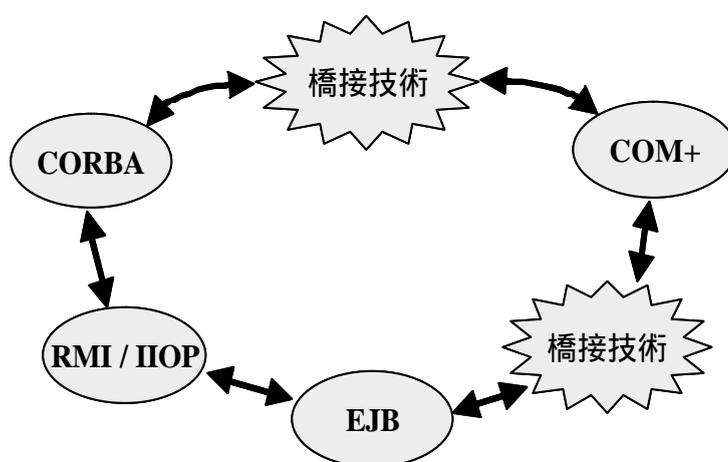


圖 2-32 不同元件使用橋接技術整合溝通

因此，單靠元件模型與橋接技術並無法真正解決企業資訊整合之問題，反而會增加系統整合的難度與成本。企業真正需要的是一個不受任何特定限制的元件模型標準。藉由此模型標準，企業內外所有的資訊系統均可以輕易地整合溝通，而 Web Service 即是在此需求之下所誕生。根據 Web Service 觀念所演衍伸的相關技術標準，不但可讓各種元件模型技術相互整合溝通，更可使用各式程式語言進行 Web Service 軟體元件的開發，如圖 2-33 所示(修改自【14】)。

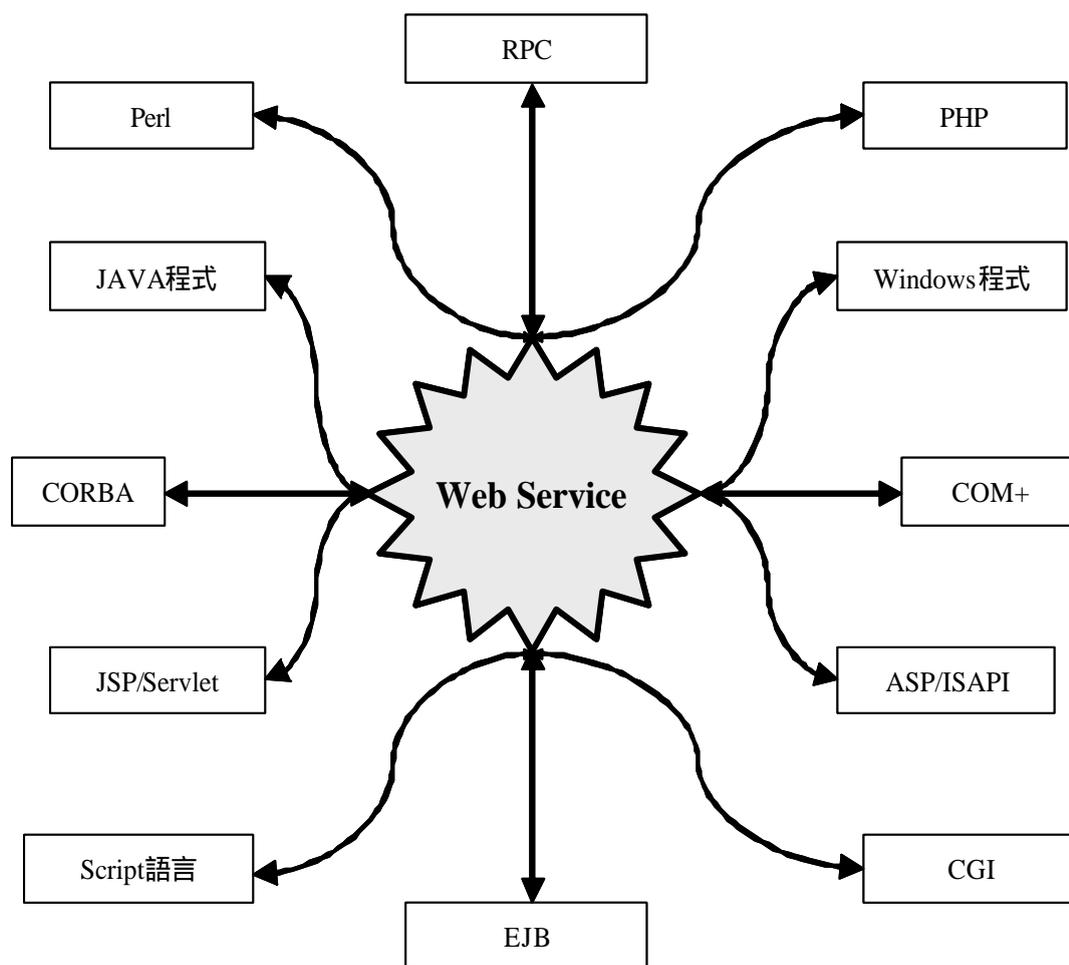


圖 2-33 Web Service 整合應用概念圖

Web Service 的相關實作技術標準最初是由許多世界軟體大廠所共同制定的，例如：IBM、Microsoft、Borland、HP 等，如今這些協定技術已成為 W3C(World Wide Web Consortium)上的公開標準(<http://www.w3.org/2002/ws/>)。Web Service 的實作主要有三大協定標準【42,43,44,45】：

一、簡單物件存取協定(Simple Object Access Protocol ; SOAP) :

SOAP 是一項傳輸協定的標準，主要結合 XML 和各式各樣的傳輸協定，例如 HTTP、SMTP、FTP 等。整個 SOAP 協定中包含三個部分，(1) Envelope 訊息封包：定義訊息封包的架構，用來描述 SOAP 訊息封包的組成，以及訊息處理等機制 (2) Header: 定義一些 SOAP 內文 SOAP 資料型態 SOAP 編碼等之 Namespace 位址。(3) Body：定義傳送 Client Request 與 Server Response 的訊息內容。

二、網路服務描述語言(Web Service Description Language ; WSDL) :

是一種以 XML 來描述 Web Services 的標準。描述內容包括(1) 描述 Server 所提供的 Web Services，以及 Web Service 可進行的各種操作方法。(2) 描述如何與 Web Service 軟體服務溝通，內容包括傳輸協定、格式、參數等。

三、通用描述、探索與整合(Universal Description Discovery and Integration ; UDDI) :

由 IBM、Ariba、Microsoft 共同主推的 Web Service 註冊及搜尋機制，主要制定的服務內容為(1) Web Service 之註冊、描述及搜尋的標準。(2) 提供介面讓公司、企業透過網路登錄 Business 相關資料。

整個 Web Service 三大協定標準的關係如圖 2-34 所示【43】。

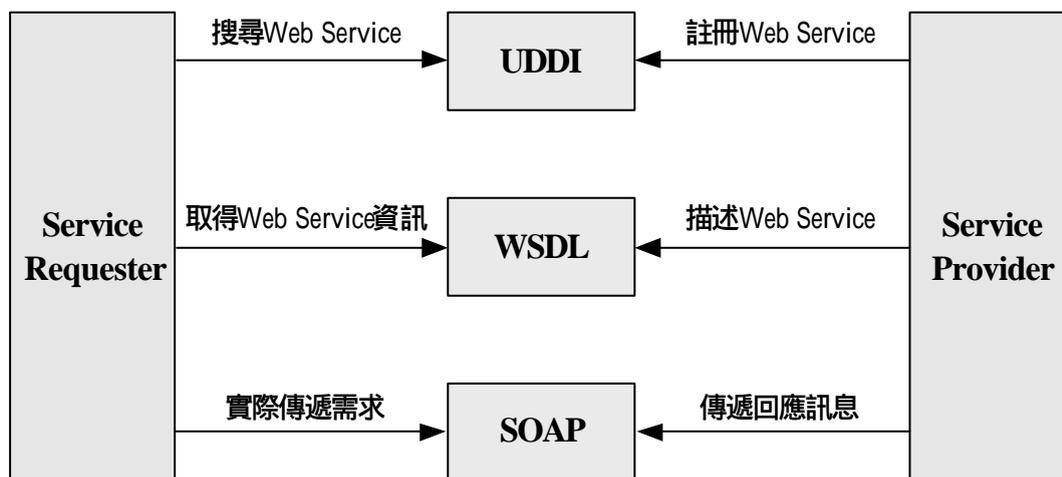


圖 2-34 Web Service 主要技術標準

UDDI 主要提供了註冊及搜尋 Web Service 的儲存資料庫，WSDL 檔案則是 Service Provider 及 Requester 之間互相連繫溝通(Binding)的語言，SOAP 負責整體架構中相關底層訊息遞送的格式標準。

就企業資訊系統整合而言，採用 Web Service 具有下列幾項優勢【46】：

1. 簡單化(Simple)：
典型系統整合方式是採用元件技術或是透過資料交換的模式來達到整合，而一旦採用 Web Service 的系統架構已建置完成，即可輕易建立一橫跨多個應用系統的新企業流程，故 Web Service 在設計、發展、維護及使用上皆較為簡單化。
2. 開放式的標準(Open Standards)：
相較於傳統式的系統整合，Web Services 所採用的標準為開放式標準，利用許多廣為運用的通訊協定與標準，企業無需再投資更多的成本來支援網路協定。此優點為 Web Service 可被廣泛使用的原因之一。
3. 有彈性(Flexible)：
傳統的系統整合多採用點對點(Point-to-Point)的整合方法，系統與系統會緊密結合在一起，故一但一端有所異動則會影響另一端，造成維護上的不便性；相較於此，Web Service 所採用的整合方式為透過使用被公佈出來的服務來與其他應用系統做結合，充份保留系統與系統間整合的彈性。
4. 整合範圍廣(Scope)：
就傳統式的系統整合而言，一個應用系統僅提供一個整合入口；然就 Web Service 而言，其可將應用系統區分成多個獨立的邏輯單位，以提供多個整合的連結點。以訂單管理系統為例，Web Service 的整合方式可提供接單、訂單狀態查詢、訂單確認、收款及付款等服務做為系統整合的入口。
5. 整合效率高(Efficient)：
Web Service 可為單一應用系統提供多種不同的整合方式，因此 Web Service 的整合方法比傳統方式更有效率且可行性較高。
6. 具動態性(Dynamic)：
企業系統整合的過程中，若採用 Web Service 的方式來進行，則只需從服務提供者(Services Provider)所公佈出來的服務中選擇適用的商業服務，並納入企業流程中，即可達到系統整合的目的。相較於傳統式的整合方法，Web Service 是以動態整合界面來整合系統，較具動態性。

2.5 小結

從文獻探討之企業經營管理、電子商務發展，以及產品開發模式演變可知，現今企業的產品開發，已非單一部門或者企業本身所能獨立自主與完成。隨著產業型態垂直、水平專業分工的發展趨勢，企業內部的產品開發流程與外部設計鏈、供應鏈廠商相互協同合作的機會不斷增加，正如同 Autodesk 設計部門執行副總裁 John Sanders 所說：「客戶當前面臨的挑戰不再只是如何利用科技來完成一項工作，而是一套解決方案，用以結合設計鏈當中的其他夥伴，讓工作流程更自動化與一氣呵成，並能反映同僚、夥伴和客戶協同作業的方式」。因此，一個能夠跨國界、跨領域，即時同步整合產品開發相關資訊的協同產品商務，已是企業發展的主要目標。

PDMS 是 CPC 的主要核心以及眾多 e 化解決方案的基礎，因此企業欲發展協同產品商務，建立完善的電子化企業，必須先具備易於整合、擁有良好基礎架構的 PDMS。於各式系統開發模式中，元件式開發是最符合現今企業的需求，藉由元件模型的標準與特性，企業可建置具備高度彈性、延展性，並且可重複使用的資訊系統；於各式系統架構中，分散式 Web 系統架構是最能滿足現今網路交易環境的要求，藉由分散式系統的容錯與平衡負載機制，企業可開發一穩定、高執行效率的商務系統；於各式資訊傳遞標準與系統整合方案中，Web Service 是最具成本效益與整合效能的解決方案，藉由 Web Service 跨平台、跨程式語言等特性，企業得以輕易整合各式資訊系統，發揮整體 e 化效益。

綜觀上述所言，PDMS 的導入與應用能大幅提昇產品的開發效率，而一個結合運用元件模型技術、分散式系統架構，以及 Web Service 標準協定的 PDMS，才能真正滿足企業 e 化與電子商業環境的需求，並有效解決存之已久的資訊化孤島問題，此不僅為本文研究探討之重點，更是現今企業 e 化發展與提昇整體營運效能的主要關鍵。本研究收集近年學術界於設計管理及產品資訊整合之相關研究(如圖 2-35)，定位出本研究內容主要是以高度資訊化應用，滿足中小企業產品資訊整合需求，試圖符合企業實質營運應用及奠定 e 化整合基礎。

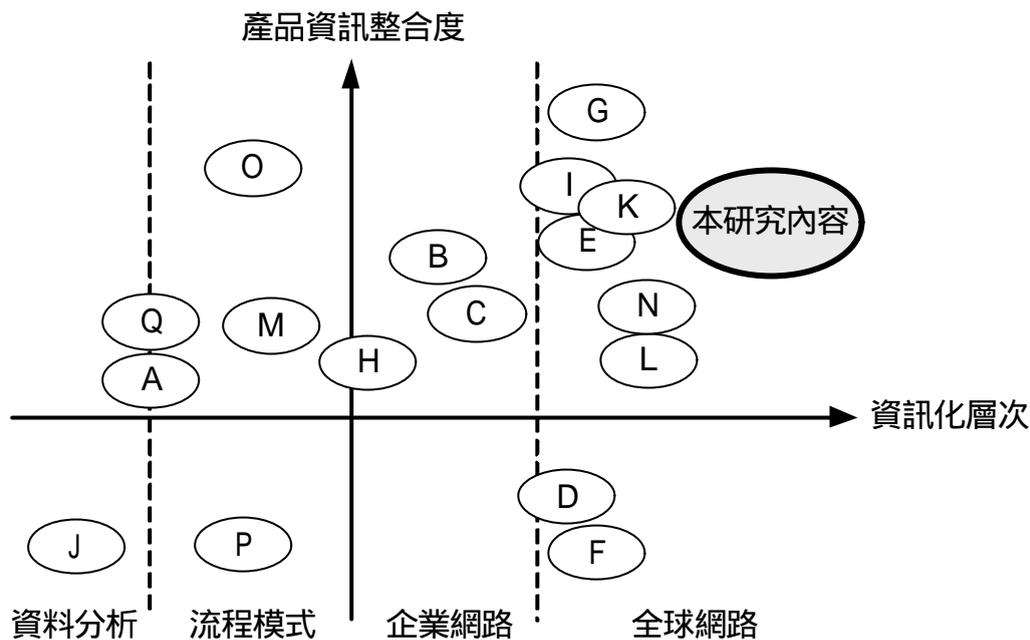


圖 2-35 國內相關研究與本研究定位

- A.彭泉 蔡禎騰 邱正祥 整合式智慧型產品開發資訊系統之研究 1999
- B.林崇宏 產品設計管理模式之研究 2001
- C.宋肇源 整合性產銷系統介面之設計與分析 2001
- D.黃弘毅 設計概念階段之產品知識建構與應用-以案例式推理法為例 2001
- E.黃致理 產品知識管理系統之建構 以組裝性製造業為例 2000
- F.李詩韻 委外設計專案之互動式顧客關係管理模式建構 2001
- G.鄭再添 建構企業產品商務系統之研究與應用 2000
- H.徐宏文 設計管理中多重設計專案之全面審議架構 2000
- I.姚威宏 產品資料管理於協同產品開發之整合-以少量多樣化產品為例 2001
- J.王錫中 運用關聯法則技術於產品開發設計之研究 2002
- K.盧永晟 創新式協同產品設計系統 2001
- L.張瑞芬 姚銀河等人 體系協同產品設計之研究 2002
- M.江清水 陳樂惠 產品開發流程模型建構研究 1999
- N.王魁生 蔡志弘 李榮貴 整合性產品開發管理系統之構建 2001
- O.曾令衛 協同產品商務CPC研究 2002
- P.蔡耀鴻 工程設計製造資訊系統整合物件庫觀念的資料模型規畫 1999
- Q.鄭錡新 溫肇東 同步工程在開發資訊系統時的應用 2001

第三章 研究理論與架構

本章將首先比較分析 Web-Based System 與 Web Service-Based System 的主要差異為何，藉由功能的分析與系統架構的比較，整理歸納出一個完善的 Web Service-Based System 應具備那些系統功能與特質，後續則據此再輔以協同產品商務之理論觀念，配合元件模型技術與分散式系統架構，提出一具備彈性、易於整合的 Web Service-Based PDMS 理論架構與開發模式。

3.1 Web-Based 與 Web Service-Based System 之比較

Web-Based System(網路式系統)與 Web Service-Based System(網路服務式系統)兩者之所以產生隔閡與差異的主要原因在於訊息傳遞的標準。最初的 Web-Based 系統僅能單純地以 HTML 語言呈現靜態的文字資料，而後則藉由後端伺服器語言(例如：CGI、ASP、PHP、JSP 等)，以及前端物件技術(例如：ActiveX、Java Applet 等)，使得 Web-Based 系統瞬間具備多樣的使用者介面與強勁的系統功能，此時 Web-Based 系統與傳統使用區域網路之系統，於操作介面與系統功能上已無太大的區別，加上 Web-Based 系統可透過網際網路，使用一般瀏覽器進行遠端的操作管理，因此各式企業內外資訊系統的發展，逐漸朝向 Web-Based 系統的開發與應用，同時也興起電子商務與企業入口網站(Enterprise Information Portal；EIP)的趨勢熱潮。

如今，電子商務的發展主要著重於 B2B 商業模式上，企業與其客戶、上下游廠商間的電子化交易行為日益頻繁，而由於傳統的電子資料交換技術過於複雜昂貴，因此已不符合經濟成本效益，取而代之的是 XML(Extensible Markup Language；可延伸標記語言)這套資料交換的標準協定，由於 XML 具備簡單、架構嚴謹、支援性高，以及跨平台等特性，因此目前已廣泛應用於各式領域。

為解決商務間交易之問題，Web-Based 系統也開始納入 XML 的應用，讓企業間可透過 XML 資料交換標準與業界專屬語辭標籤，進行線上商務交易或者產品資料交換的運作。倘若企業間彼此建立合作關係，共同制定資訊傳遞標準的話，則企業間絕大部分的商業行為與資料交換過程，均能達到自動化處理的目標。然而，企業間欲共同制定資訊傳遞標準，達到系統資料、資訊整合的目的，並非一般企業能力所及，因為標準的制定、資訊系統的運作流程與轉換等，企業間均需有高度的共識與資訊整合

能力。況且，即便企業間有能力建構一跨組織的整合性系統，由於此系統屬於封閉式架構，因此後續維護管理工作將極為艱難，對於未來系統的管理成本與功能擴充而言，最終不是一個良好的解決方案。

目前 Web-Based 系統無法再持續發展與應用的主要原因，便是缺乏一符合國際標準通用的訊息傳遞標準，因此無論 Web-Based 系統功能如何強大、執行效能如何優異，倘若缺乏此標準機制，最終也只是眾多資訊化孤島的一環，無法輕易地與其他系統整合溝通。雖然目前已有許多系統整合方案相繼提出，但這些解決方案大多僅侷限於各大策略聯盟廠商所提出的軟體系統，除了無法自由選擇合適的產品與整合方案之外，龐大的導入成本更是企業沉重的負擔。

若以使用者的角度來比較 Web-Based 與 Web Service-Based 兩者之差異，可發現 Web-Based 的主要觀念為使用者可透過網際網路取得各式資料與資訊，而 Web Service-Based 則是讓使用者或系統(以 Actor 為統稱)可透過網際網路取得各式服務。前者於觀念上偏向於使用者自己去尋找符合本身需求的資源，而後者偏向於 Actor 去接受別人已整理好的資源。以目前眾多的網路電子報為例，若使用者只想閱讀每份電子報中的財經消息，以 Web-Based 的觀念來說，則使用者只能自己去整理所有電子報中的財經消息，但若以 Web Service-Based 的觀念來看，Actor 可直接取得網路上已經過整理歸納，甚至附上各式專業分析圖的財經消息，如圖 3-1 所示(本研究)。

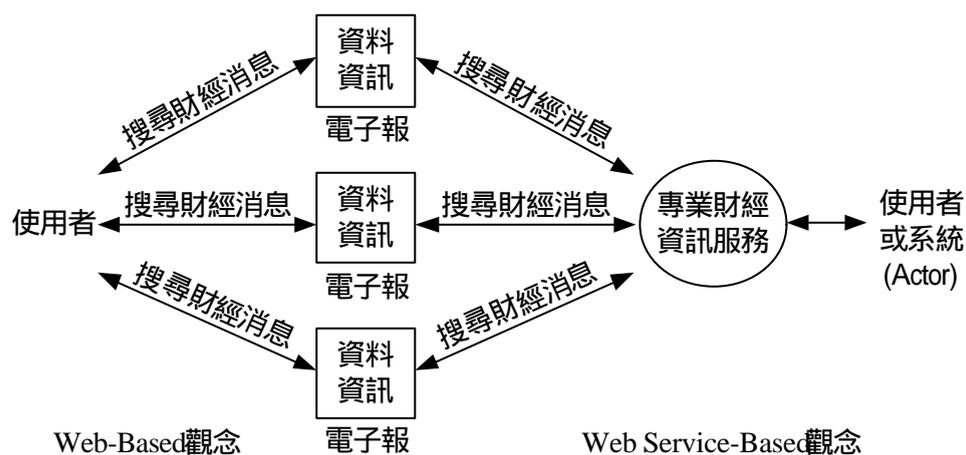


圖 3-1 Web-Based 與 Web Service-Based 於使用上之差異

因此，一個 Web-Based System 建立的主要功能在於提供使用者一個操作平台，讓使用者可透過網路取得此平台上的各式資料與資訊，而一個 Web Service-Based System

建立的主要功能則是提供 Actor 一個服務平台，讓 Actor 可透過網路取得此平台上各式資訊與服務。

以資訊技術的角度觀看，由於 Web Service-Based 基本上是 Web-Based 的延伸，再加上運用了 Web Service 的三大協定標準(SOAP、WSDL、UDDI)，因此其不僅具備 Web-Based 系統架構的所有優點，更能滿足現今環境對於系統整合與自動化處理機制的的需求，強化 Agent-Based System 訊息傳遞溝通與整合的通透性。茲將 Web Service-Based 系統與 Web-Based 系統的主要差異整理於表 3-1(整理自【14,15,42,43】)。

表 3-1 Web Service-Based System 與 Web-Based System 比較表

比較項目	Web Service-Based System	Web-Based System
系統觀念	提供一服務取得平台	提供一資料/資訊取得平台
發展年代	90 年代末期	90 年代初中期
主要技術	Web 程式語言+Web Service	Web 程式語言
訊息傳遞標準	SOAP	無
自動搜尋機制	UDDI	無
描述訊息機制	WSDL	無
跨異質平台 整合與支援	高(採用通用的協定標準,且絕大多數作業平台均有支援)	低(僅有少數程式語言可跨平台,且採用其專用的語法與標準)
跨程式語言能力	高(大部分程式語言均支援)	低(需使用橋接技術或轉換系統)
彈性與可延展性	高	低
主要應用對象	應用程式、使用者	使用者
主要通訊協定	HTTP、FTP、SMTP、TCP/IP 等	HTTP、TCP/IP
資料格式	XML	多採用 HTML, 使用 XML 較少
自動化處理能力	高	低
執行效能	中	高
系統開發時間	較長	較短
系統功能與目標	讓系統或使用者可在任何平台,使用任何程式語言或裝置,透過網路取得資訊服務,進而達到資訊、系統自動整合運用的目標	讓使用者可透過前端程式,例如:瀏覽器,透過網際網路查詢符合自己需求的資料與資訊

目前較先進的 Web-Based 系統均已採用元件模型技術，搭配元件模型技術可增加系統的執行效能、彈性、延展性等，而整個系統架構與運作情形如圖 3-2 所示(修改自【14,40,41】)。

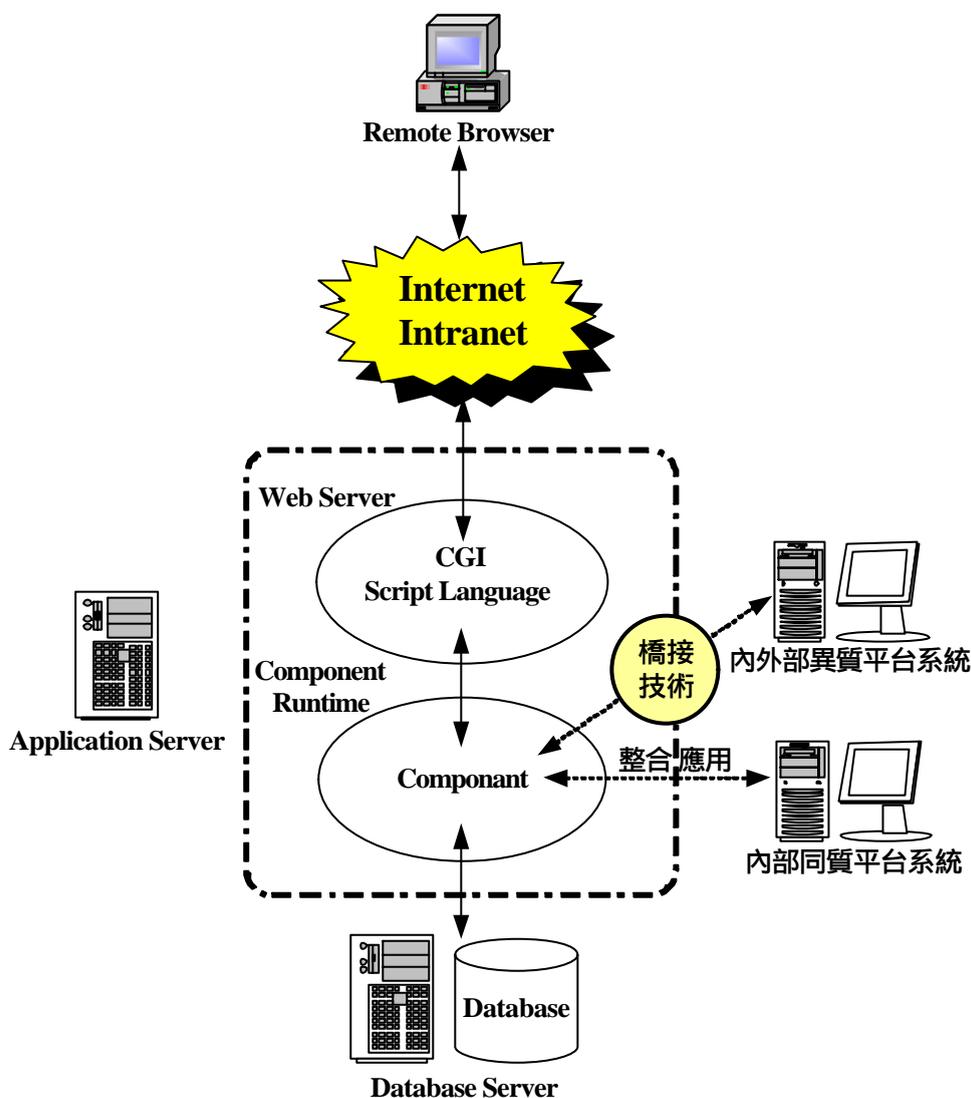


圖 3-2 元件式 Web-Based 系統架構圖

圖 3-2 採用的是三層式架構，前端展示層以瀏覽器程式為主，而整個系統運作流程為：(1) 使用者利用 Browser 透過 Internet/Intranet 啟動後端應用程式伺服器中的 Web Server。(2) Web Server 收到參數後直接回應或者進一步呼叫元件服務。(3) 元件服務執行企業邏輯或者資料庫存取與處理。(4) 元件處理結果傳送至 Web Server，再由 Web Server 整理回應給前端瀏覽器程式。

圖 3-3 為元件式 Web Service-Based 系統架構圖(修改自【14,40,41】), 其運作流程大致與 Web-Based 系統雷同, 而最大不同點在於 Web Service-Based 系統在元件軟體外面增加一層 Web Service, 此層的主要用途是將元件功能的呼叫介面轉為 Web Service 的協定標準, 如此不但可提供上層 Web Server 中的程式語言運用, 更可讓企業內外的各式程式語言、系統平台、硬體裝置等, 透過通用的標準協定使用企業元件所提供的功能。因此, 圖 3-3 對於異質系統整合的方式主要是透過 Web Service 標準協定, 而圖 3-2 僅能使用橋接技術加以轉換與整合。

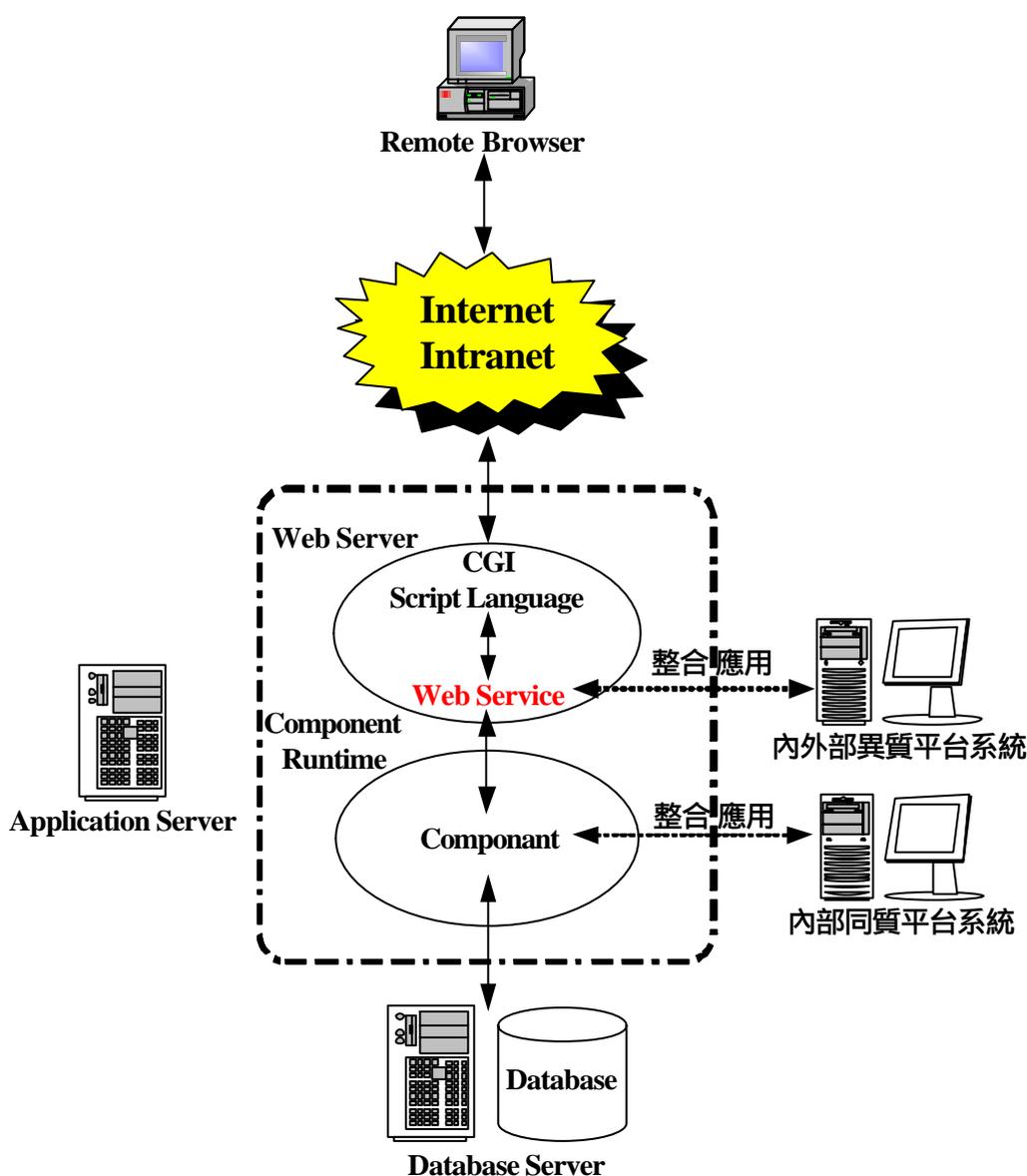


圖 3-3 元件式 Web Service-Based 系統架構圖

3.2 Web Service-Based PDMS 理論架構

處於知識經濟時代的今日，企業必須整合所有營運資源並有效管理，才能創造最大的商業利潤與企業競爭力。PDMS 不僅是製造企業 e 化的核心，更是協同產品商務的根本，因此勢必與產品開發流程和其他資訊系統有著直接、間接的連結關係，如圖 3-4(整理自【11,17,33】)。

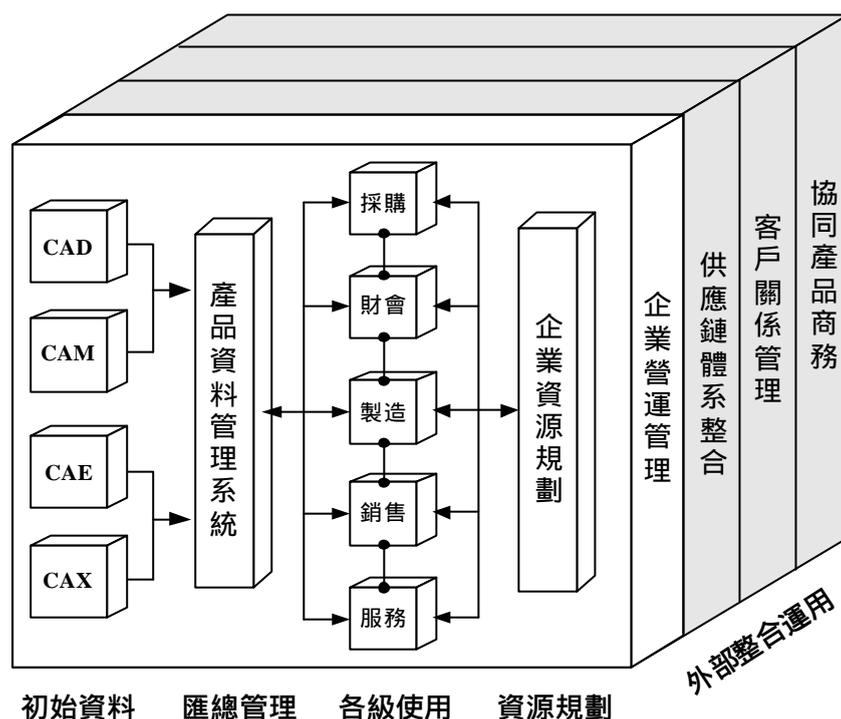


圖 3-4 PDMS 與產品開發系統之關係架構

由圖 3-4 之架構可知，CAD/CAM/CAE 等電腦輔助軟體為 PDMS 的主要初始資料來源，當產品設計研發的電子化資料匯入 PDMS 之後，PDMS 便會依據企業事先所設定的流程邏輯與組織架構，將產品資料分門別類地儲存與管理，其他部門如：採購、財會、製造、銷售等，則由人員或資訊系統取用與回存產品的相關開發資訊，最後再由 ERP 系統將後續產品開發階段的資訊與流程作一歸納與控管，進而結合企業營運流程、供應鏈管理，以及客戶關係管理等，與客戶、夥伴、上下游廠商進行協同產品商務的合作。

若從企業營運資訊系統的角度來看，前端的銷售服務系統必須取得 PDMS 的產品型錄資料，作為客戶參考選購的依據；協同設計、同步工程系統需使用 PDMS 所管理的設計圖檔、文件等資料進行產品開發；ERP 系統需要 PDMS 所控管的 E-BOM(工程物料清

單)、M-BOM(製造物料清單)資料來進行企業資源的規劃；進銷存貨系統需要擷取核對 PDMS 的產品資料；產品知識管理系統(Product Knowledge Management System : PKMS) 需要轉化 PDMS 所控管的作業程序與資料，將其提升為產品知識營運價值；SCM 需要功能架構符合國際標準的 PDMS 作為基底，才能整合交換上下游之產品相關資料；發展電子商務則需要一完整功能的 PDMS，才能有效串聯管理金流、物流以及產品資訊流。換言之，企業因 e 化所導入的系統，絕大部份均與 PDMS 相關，不僅使用 PDMS 所控管的產品資料，甚至經常使用 PDMS 的商業邏輯或者模組功能。如圖 3-5 所示(本研究)，一個完整的 PDMS 是由許多 PDMS 模組功能所組成，而這些模組所包含的產品資料與系統功能，於理論上應可完全提供給其他資訊管理系統使用。

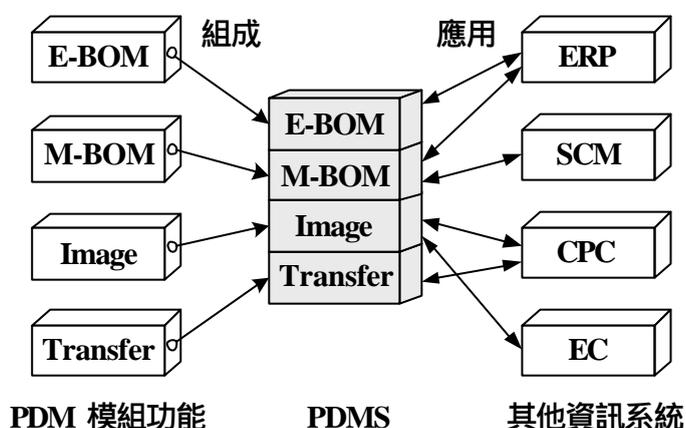


圖 3-5 PDMS 模組組成與應用概念圖

若從系統規劃設計以及物件再用(Object Reuse)的角度重新思考【47】，可推論出一個符合現今環境趨勢的 PDMS，需具備以下幾點主要特質：

1. PDMS 的組成必須是可元件化、模組化並具備一定的系統彈性，才能動態地修正系統功能，符合外界需求。
2. PDMS 必須可跨多種異質作業平台與程式語言，並且具備簡易操控的介面。
3. PDMS 的建置必須採用符合世界通用的標準技術協定和資料交換機制，以避免系統無法整合溝通之問題。
4. PDMS 必須充分結合網際網路(如：提供符合 W3C 標準的前端管理系統、結合無線通訊等)，以便快速傳遞產品資訊，掌握環境變化。
5. PDMS 應具備企業邏輯法則，運用人工智慧與專家系統的方法技術，使 PDMS 具備一定程度的商業智慧，自動分析判斷產品資訊，並且即時彙整所有訊息，反應

給相關系統或人員。換言之，新一代的 PDMS 必須與企業數位神經系統結合，形成綿密的知識傳遞網路環境。

PDMS 要滿足上述特性、達到整合運作之目標，則須採用新的技術與通用的協定標準，其中元件式系統開發模式(Component-Based System)、XML、SOAP、Web Service 以及分散式系統架構等，為新一代產品資料管理系統的主要開發技術標準。採用元件式系統的主要目的是確保系統能夠彈性模組化，因為元件模型不但具備物件導向之特性，更克服傳統物件之缺點，因此可以有效降低系統開發與後續維護的困難度。XML 則提供了系統間的資料交換格式；SOAP 訊息傳遞協定、Web Service 標準技術，其成本低廉、簡易、跨平台、跨語言等特性，可讓異質元件和系統，均能輕易地整合溝通；分散式系統架構則確保系統的穩定性以及高負載能力，如此才能在資訊交換頻繁的環境下正常運作。綜合上述之觀念與系統開發方法，本文所提出之 Web Service-Based PDMS(簡稱 WS-PDM)理論架構如圖 3-6 所示(本研究)。

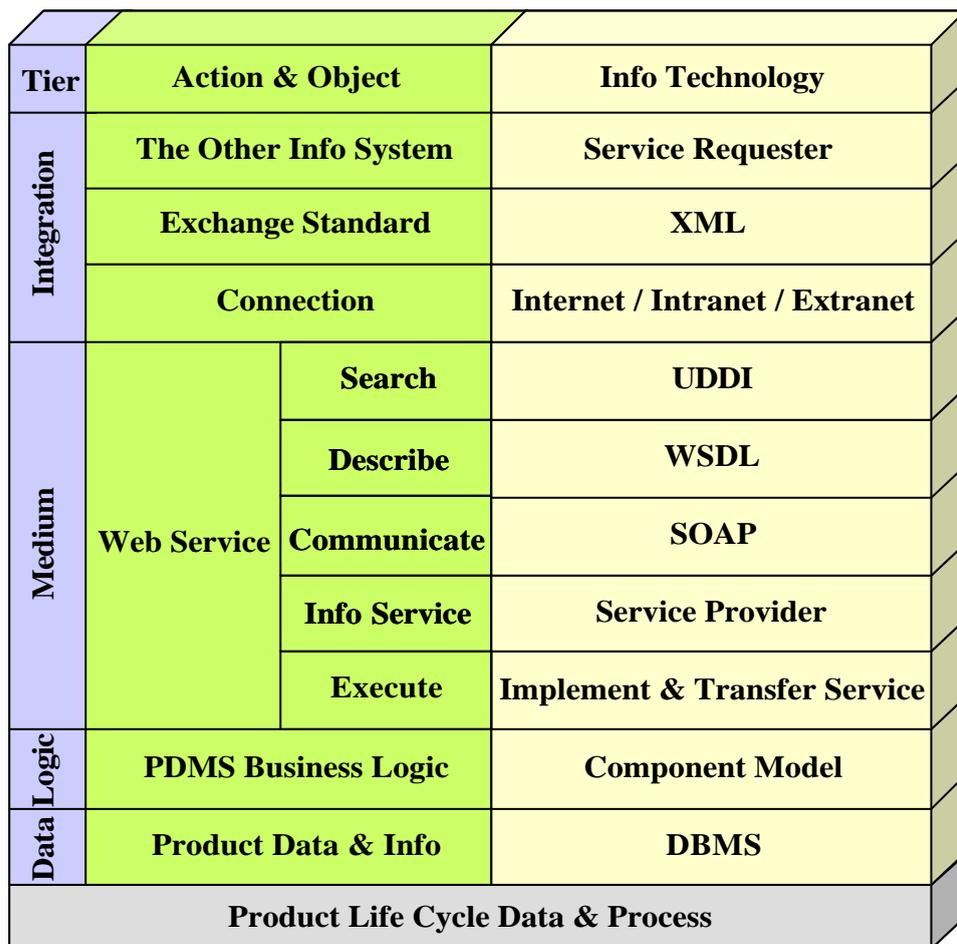


圖 3-6 Web Service-Based PDMS 理論架構

圖 3-6 架構主要將 Web Service-Based PDMS 分為四個層次，由下至上分別是：

- 產品資料服務層 (Product Data Tier)：
產品資料庫為整體架構的主要核心，於此理論架構中，產品資料的管理，僅能透過 PDMS 所提供的功能邏輯及權限定義來操控使用，其理由是避免錯誤或蓄意的使用程序，造成產品資料的嚴重損毀。此外，透過單一窗口使用產品資訊，除了將有效降低企業整體系統架構與流程的複雜度，更能夠精確控管產品資料、資訊的輸出與輸入。現今較普遍使用的 DBMS (Database Management System)，例如 MS SQL、Oracle 等，均提供完善的資料庫連結協定標準與資料操作語言，如 ODBC、JDBC、SQL 等。一些 DBMS(如：MS SQL)甚至可將資料庫裡的資料表自動整理轉出為 XML 格式，以便進行資料交換，因此在此層級較無資料整合與轉換之問題。
- PDMS 商業邏輯層 (PDMS Business Logic Tier)：
PDMS Business Logic 的觀念主要是將 PDMS 的商業邏輯與功能模組，轉換為以元件技術為基礎的應用程式。採用元件技術，將會使 PDM 系統更具彈性，例如：公司認為目前的設計審議流程控管元件，除了需要經過設計部門主管審核之外，還必須同時交付給製造部門主管審核，要滿足此一功能需求，系統開發人員只需將舊有元件抽掉，改換上新的設計審議流程控管元件即可，就如同將舊電池抽掉，更換為一個新電池一樣。能夠將元件式系統模組功能隨時抽離更換的主要原因，是因為元件是以介面標準(Interface)作為與外部程式互動的唯一路徑，因此不同的元件，只要具備定義相同的介面，即可作相互整合或替換。另外，元件具備跨語言、資訊封裝、介面繼承等特性，因此能夠滿足 PDMS 未來系統功能的擴充，以及內部元件功能的可再用性。由於此層中的元件存在於 Intranet 內，因此區域網路內的系統可直接取用 PDMS 的元件功能，但倘若為異質系統平台(泛指採用不同的作業系統、元件模型技術、程式語言、硬體裝置等)，則依然需透過 SOAP/Web Service 的介面才能使用。PDMS Logic 的元件類別主要可分為三種：
 - (1) 資料元件：負責擔任存取資料的工作，為最接近產品資料庫的元件。其他元件只能透過資料元件所定義的方法來存取產品資料庫，另外當產品資料庫或者資料庫語法有任何變更時，基本上僅需更改資料元件即可，如此將可降低維護系統的成本。
 - (2) 功能元件：功能元件主要負責企業邏輯與系統功能的執行，例如查詢使用者的權限、查詢設計圖檔資料、轉換圖檔格式、搜尋產品資料、計算產品成本等。功能

元件又分為基本功能與整合功能兩種，所謂基本功能元件指的是此元件所擁有的資料與方法原則上是不會變更的，主要功能是提供整合功能元件或系統，最基礎的功能服務，例如權限管理、基本資料查詢等。所謂整合功能元件主要是將數個基本功能元件所提供的基本服務串聯起來，用以提供更完整的系統功能元件，例如前端系統需要查詢產品銷售資料、型錄資料以及庫存資料，則整合功能元件可將提供產品銷售資料、型錄資料、庫存資料的基本功能元件統整起來，並加以整理設計形成一整合功能元件，滿足前端系統真正的需求。

(3) 協調元件：協調元件又稱為控制元件，用以執行企業法則或者企業流程的邏輯。例如企業的設計審核流程、設計變更流程、產品資料管理流程、交易控制流程等，都屬於協調物件所應實作的。除了負責協調與管理各式實體系統流程之外，協調元件也負責整合各式功能元件，形成對外使用系統功能的唯一窗口，其性質類似整合功能元件，但整體層次更高，因為它可能使用了許多整合功能元件、基本功能元件，甚至於其他的協調元件，目的則在於完成一個系統功能。茲將此三大元件與系統之互動以圖 3-7 表示(本研究)。

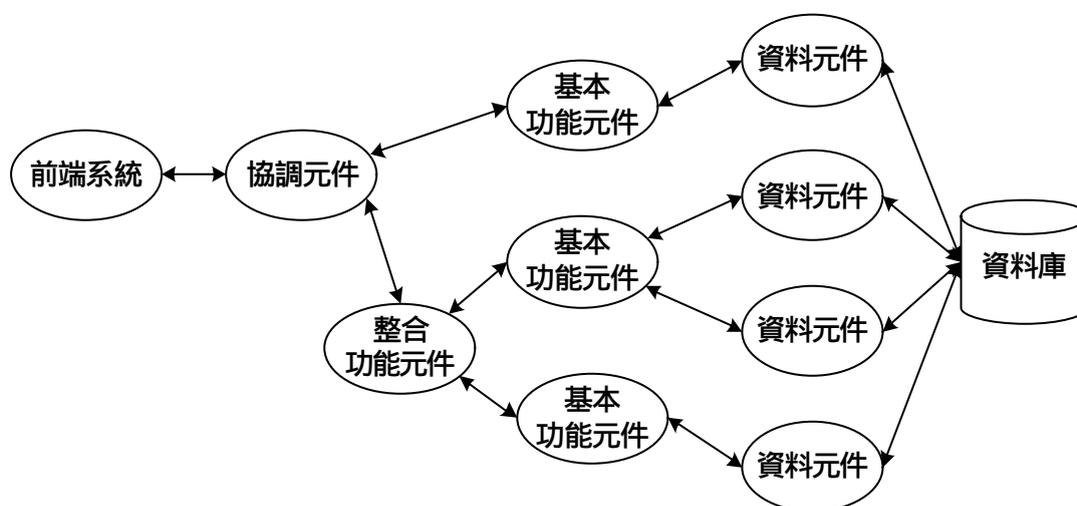


圖 3-7 元件式系統三大元件運作流程

- 網路服務中介整合層 (Web Service Medium Tier)：
元件技術雖然可以跨程式語言，但是卻難以跨不同元件，其主要原因是各家廠商所制定的元件標準不同，因此倘若不同的元件(例如：COM+與 EJB)需相互整合運用

時，則須透過橋接技術做轉換處理，由於橋接技術有版次之分，加上費用昂貴、難以使用，因此往往成為企業資訊整合的一大阻礙。於此理論架構中，利用 Web Service 的特性與標準，作為內層元件與外界互動溝通的橋樑，將可使元件所提供的功能更具融通性。換言之，無論外界採用何種元件模型、作業平台、應用程式、硬體裝置等，均可透過 Web Service 中介層，使用元件所提供之功能與服務。因此，於此層的主要工作為定義與實作 Web Service 的功能服務，將內層的 PDMS 商業邏輯整理分析，把外部系統所使用到的功能介面，轉換為 Web Service 的呼叫介面，如此外部系統便可透過標準的 SOAP 協定使用 Web Service，而 Web Service 的實際功能則由內層系統的元件予以實作與回傳。Web Service 的功能介面主要是透過 WSDL 所發布，若系統要具備自動化搜尋 Web Service 機制，則可透過 UDDI 的運用來完成。

- 產品資訊整合層 (Integration Tier) :

此層級的觀念是企業內外系統，例如：ERP、SCM、CRM 等，可使用 SOAP 訊息傳遞標準協定，透過 XML 資料交換標準格式，取用 PDMS 所提供的模組功能以及產品資訊，或者進行系統之間的整合工作。然而，SOAP 並非完全適用於所有的資訊傳遞與整合。由於採用 SOAP 必須針對資料做額外的處理與轉換(為了符合標準協定)，因此系統的執行效能會比系統間直接互動較差。所以，倘若系統之間可使用元件技術加以整合，則應以元件技術作為第一考量，SOAP 協定則作為輔助以及主要處理元件技術難以整合或運用的部分。

以往的產品資料管理系統均以主從架構或者 Web-Based 系統為主，如今以元件技術、Web Service 標準為核心的 Web Service-Based PDMS，主要採用分散式系統架構，其系統架構如圖 3-8 所示(本研究)。整個系統運作流程如下：

1. 前端程式(瀏覽器)透過 Internet/Intranet，連結到 Web Server 並啟動 Web Server 中的 Web Program(例如：ASP、JSP、ISAPI 等)。
2. 傳遞符合 SOAP/XML 標準格式的參數資料至 Web 伺服器，然後 Web Program 判斷此參數需求，直接回應前端程式或者呼叫後端的產品資訊 Web Service。
3. 後端 Web Service 根據前端所呼叫的方法，直接處理使用者需求或者啟動後端應用程式伺服器上的 PDMS 模組元件。
4. PDMS 模組元件執行與處理前端需求，若前端程式需存取後端產品資料庫的資

料，則將由協調元件或整合功能元件啟動基本元件，再由基本元件呼叫資料元件，利用資料元件連結後端產品資料庫，然後再執行特定的 SQL 語法取得或管理產品資料。

5. 將所取得的資料或經過元件邏輯處理後的結果，轉換為 SOAP 與 XML 的標準格式，回傳至 Web Server 中的 Web Program，待 Web Program 執行完畢之後，再由 Web Server 透過通訊協定回傳給前端程式。
6. 於此架構中，同質系統(Homogeneous System)直接整合或應用 PDMS 元件，異質系統(Heterogeneous System)則透過 Web Service 中介層進行整合與應用。

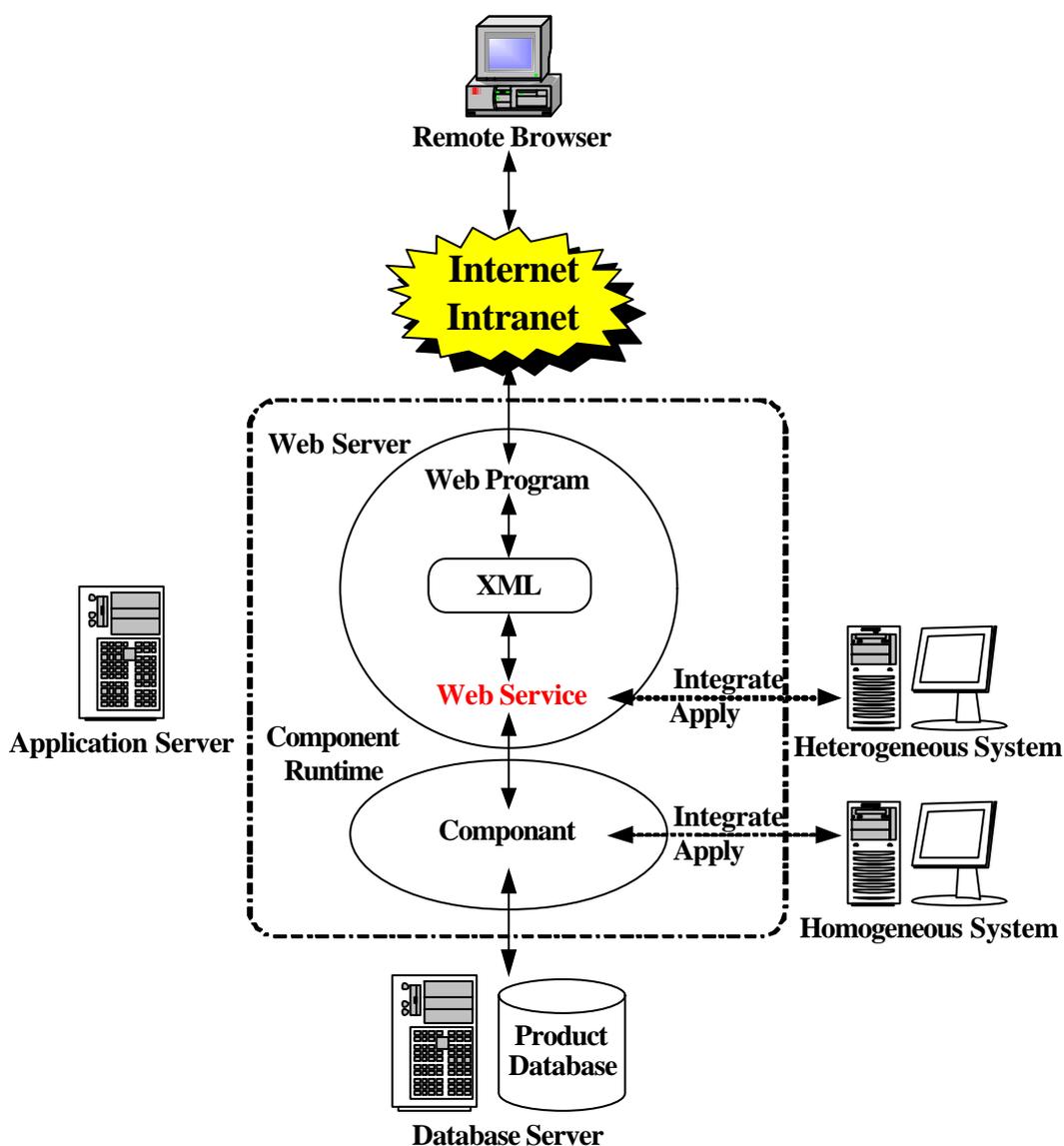


圖 3-8 Web Service-Based PDMS 系統架構

3.3 Web Service-Based PDMS 開發模式

開發 Web Service-Based PDMS 除了需具備前述之觀念技術之外，還需先充分了解使用者需求，以及企業軟硬體系統的狀況。整個系統開發模式流程，如圖 3-9 所示。

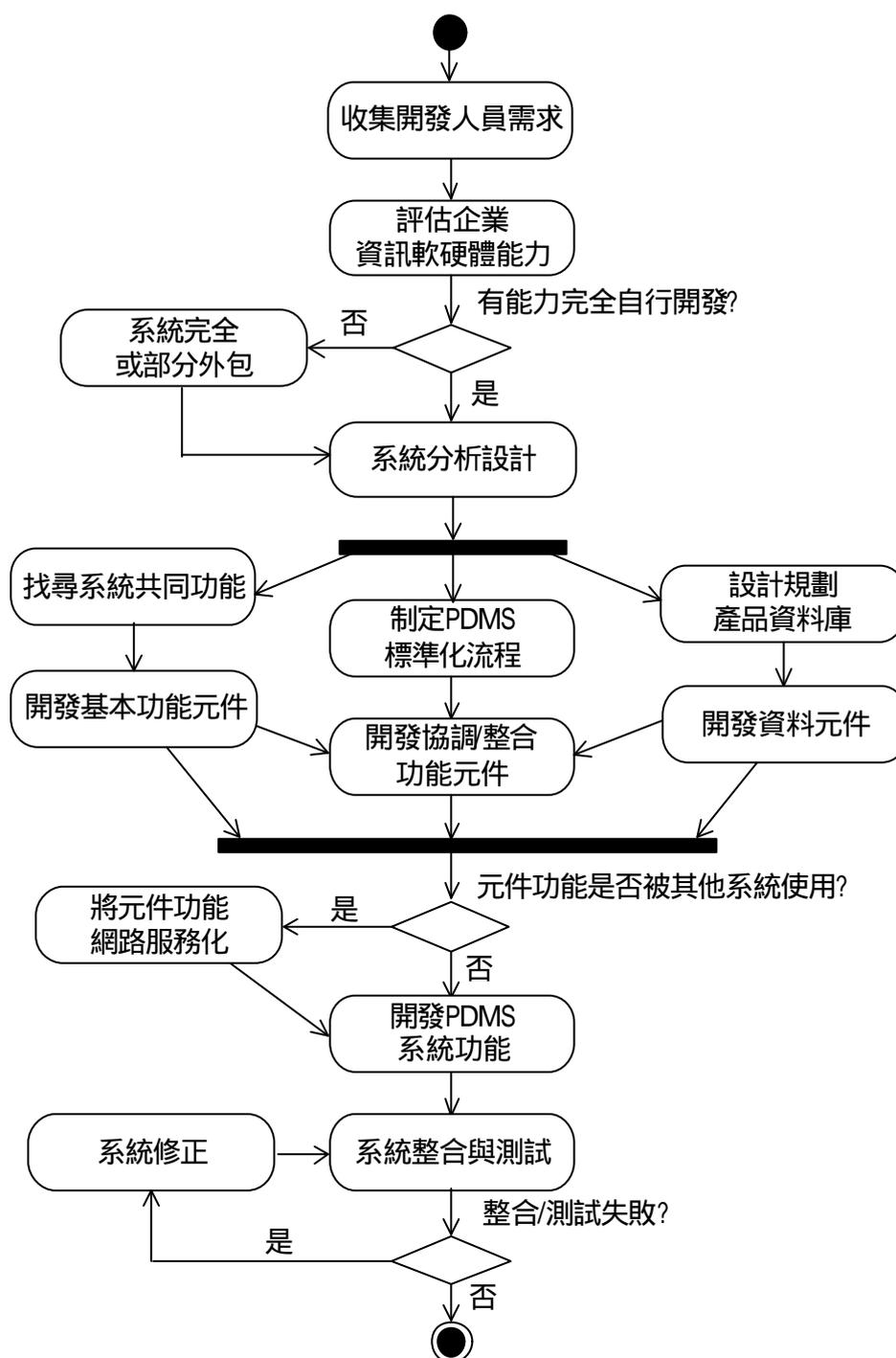


圖 3-9 Web Service-Based PDMS 開發模式

圖 3-9 是以 UML 活動圖描述 Web Service-Based PDMS 開發模式的流程，茲將整個流程活動步驟分述如下：

步驟一、收集開發人員需求

由於 PDMS 的主要功能與應用範圍在於設計研發，因此應先從設計研發部門人員的需求訪談開始。常見的設計研發管理問題有設計圖檔編碼混亂、設計圖檔重覆、設計資料存取無標準規格、設計資料管理方式缺乏效率、存取資料程序過於繁雜，以及未將設計流程標準化等，系統開發人員可從這幾個大方向訪談設計師、研發人員，以及相關主管等，了解使用者真正需求以及目前產品設計研發的流程為何。除此之外，若企業欲擴展產品資料管理範圍，發展同步工程或者協同產品開發模式，則也應蒐集其他部門，甚至於客戶、上下游廠商之資料與流程，以便後續作適當的元件與網路服務設計。

步驟二、評估企業資訊軟硬體能力

國內大多屬於中小企業，其資訊能力往往無法不足以完全自行開發系統，加上企業資訊軟硬體設備或許不足，因此於正式開發 Web Service-Based PDMS 之前，需先針對企業資訊能力與目前軟硬體設備進行了解與評估，然後選擇是否全部外包或者部份外包，以及企業應增添那些資訊軟硬體等。評估資訊能力的重點在於人員是否具備分散式系統開發、元件式系統、Web Service，以及網站開發的知識經驗與技術觀念。評估軟硬體設備的重點在於了解系統開發範圍內，欲整合運作的資訊管理系統，其軟體平台、開發語言、使用元件模型為何，而企業現行的資訊系統架構能否支援系統的整合。無論部分外包或者全部外包，企業均應結合內外部成員組成開發團隊，一方面外部人員可直接經由此團隊取得系統開發相關資料與協助，另一方面藉由內外部開發人員的交流互動，達到知識經驗分享與學習的目的，提昇企業人員系統開發能力。

步驟三、系統分析設計

根據所收集的使用者需求、系統開發資料與流程，進行系統的分析與設計。分析設計的原則將以制定 PDMS 標準化程序、找尋系統模組間的共同功能，以及規劃設計資料庫欄位表格為主。分析的觀點應以元件和物件導向的角度切入，利用元件功能獨立、介面繼承、軟體再用等觀念，進行系統功能的切割與組合的規劃。找尋系統模組間共同功能的目的是在於設計基本功能元件；設計規劃資料庫關聯表格的用途在於後續

資料元件的建立；制定 PDMS 標準化流程主要為了開發協調與整合功能元件。

步驟四、元件功能網路服務化

一旦 PDMS 系統已利用元件之觀點設計規劃完成後，則再以系統整合和物件再用的角度，思考那些元件功能可利用 Web Service 加以封裝，並將元件的介面轉換為 Web Service 的服務介面，以提供其他作業平台、系統、程式語言、前端裝置所使用。

步驟五、開發 PDMS 系統功能

藉由已設計並開發完成的軟體元件(包括以 SOAP/WSDL 封裝元件功能介面的 Web Service 軟體元件)，進行元件的組成與運用，以完成先前規劃好的系統功能與流程。在此階段可藉由元件獨立運作之特性，進行系統部份功能的組成與執行測試，以便確保單一系統功能的正確性與穩定性。另外，也可於此階段將元件組合或執行測試的工作分配給各個小組同時進行，如此平行開發的模式將可縮短系統建置完成的時間。

步驟六、系統整合測試

將各個系統功能依事前所規劃設計之 PDMS 標準流程加以整合測試。整合的重點在於驗證系統模組間，是否能依照系統所制定的標準流程執行運作，而運作的結果能否滿足企業實際需求。測試的重點在於了解 PDMS 系統的穩定性以及處理產品資料的正確性，藉由讓使用者實際上限測試的結果，了解是否系統功能不足或者有例外狀況產生。

步驟七、系統修正

藉由系統的整合測試了解問題點所在之後，便進行系統的修正。系統修正的方向應由上而下，主要原因是越底層的軟體元件，其所提供的功能與保存的資料/變數就越基本。因此倘若愈往下修正基本流程與資料，則上層所牽扯到的元件數量就愈多，雖然元件模型技術可藉由介面繼承或功能修正來解決此問題，但能盡量避免變動基本流程與資料，將可大幅減少系統修正與開發的時間。茲將修改系統的順序整理如下：(1) 修改 Web Service 所封裝的軟體元件。(2) 修改協調元件。(3) 修改整合功能元件。(4) 修改基本功能元件。(5) 修改資料元件。(6) 修改資料庫欄位或資料表格。

第四章 企業產品資訊整合現況調查

由於近年來協同產品商務和各式 e 化解決方案的興起，促使企業不斷運用資訊技術，試圖整合產品生命週期中的所有資源，進而發揮整體營運效益。本研究雖可從文獻理論與產業趨勢的探討，了解現今及未來企業於產品開發模式、企業經營管理，以及企業 e 化發展的主軸方向為何，但並無法得知目前企業的實際發展狀況。有鑑於此，本研究即透過問卷與深入訪談的方式，進一步了解國內企業於此環境趨勢下的發展策略，以及產品資訊整合的現況為何，希望能將訪談的結果與文獻理論作相互的探討，並了解企業於設計研發管理上常見之問題與需求，作為後續系統功能開發的依據。

4.1 調查方法與步驟

本研究之調查方法係採用質性研究，亦即「個案研究探索法」(Case Study Research)，採用此法的主要原因是藉由深入企業訪談，除了可取得結構化問卷表單上的資料數據外，更能以非結構化的問題訪問企業人員，讓受訪者能根據企業實際營運狀況，更精確且充分地回答問題。因此，本研究於調查問卷上的設計主要有二，其一為結構化問卷(詳見【附錄 A】)，其二為非結構化問卷(如表 4-1 所示)。調查方法的步驟如圖 4-1 所示。

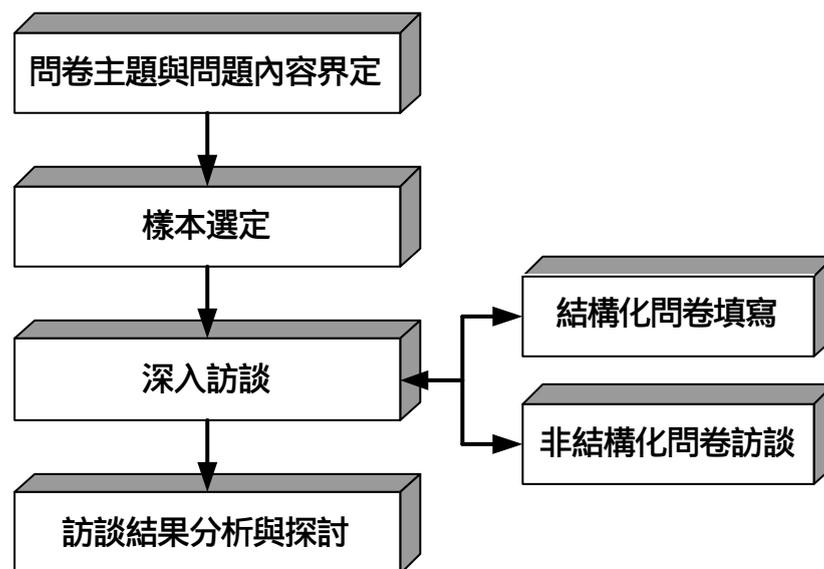


圖 4-1 企業訪談調查步驟

表 4-1 企業訪談非結構化問卷內容

主題	主要問題
主題一： 企業發展策略	<ul style="list-style-type: none"> • 於目前全球化的激烈競爭環境中，公司永續經營的管理目標為何？ • 公司目前 OEM / ODM / OBM 的百分比為何？未來將會如何調整？ • 公司的電子商務發展目標為何？目前有那些相關專案計畫在進行？ • 企業發展 CPC 的目標或準備為何？導入模式為何？ • 公司將以那一資訊系統作為發展企業電子化與協同產品商務的核心？
主題二： 企業電子化現況	<ul style="list-style-type: none"> • 目前公司已導入那些資訊系統？導入的順序為何？採購外包與自行開發的百分比為何？導入單一系統的預算上限為何？ • 公司導入資訊系統或者進行系統整合時，最常發生的困難點有那些？ • 公司導入 e 化的主要成效有那些？解決那些傳統營運上的問題點？
主題三： 產品開發模式	<ul style="list-style-type: none"> • 公司目前主要的產品開發模式為何？ • 工程設計變更之頻率為何？最常出現於那一階段？以何種方式傳遞訊息？ • 海內外部門人員使用產品相關資料的狀況為何？(就資料取得的難易程度、取得方式、資料共用，以及安全控管程度而言) • 公司產品開發流程或開發專案以何種方式控管？
主題四： 產品資料管理	<ul style="list-style-type: none"> • 公司於設計、製造階段的產品資料及開發文件利用何種方式管理？ • 設計研發人員取得產品設計資料(設計文件與圖檔等)的方式為何？ • 公司與合作廠商產品資料文件往來的頻繁度為何？主要透過那些方式？ • 公司工程物料清單(E-BOM)如何轉換為製造物料清單(M-BOM)？ • 公司是否已導入 PDMS？導入的模式為何？系統架構、開發方式、主要技術為何？其所具備的功能為何？未來對於 PDMS 的功能需求為何？ • 公司產品開發相關資訊系統(例如：PDM、ERP、SCM 等)的整合程度為何？利用何種方式整合？未來產品資訊整合的目標為何？ • 公司是否考慮使用 Web-Based PDMS？主要原因與其優缺點為何？
主題五： 企業產品資訊整合趨勢的看法或建議	(由受訪者填寫)

圖 4-1 主要分為四個步驟，依序為(1)界定問卷主軸與問題內容。(2)樣本選定。(3)深入訪談。(4)訪談結果分析與探討。茲分述如下：

(1) 問卷主題與問題內容界定

本調查目的在於了解企業面對環境衝擊之管理目標、e化成效，以及產品開發資訊整合現況，因此問卷所界定的主題有四個主要方向：一、企業發展策略 二、企業電子化現況 三、產品開發模式 四、產品資料管理，後續則依此四大方向，分別設計結構與非結構化問卷內容。結構化問卷的設計，主要先讓受訪者了解本調查的方向與問題內容，進而藉由勾選問卷上所預設的答案，快速回答與提供問卷資料。然而，結構化問卷的缺點在於受訪者可能會對問題或預設答案產生誤解，抑或無心填寫敷衍了事，因此再輔以非結構化問卷訪談，藉以提高訪談內容的精確性與真實性。

(2) 樣本選定

由於國內企業有 85%以上屬於中小企業，產業類別又以製造業為主，因此調查對象以中小製造企業為主。此外，調查對象還須具有設計研發部門、資訊管理部門，以及企業e化基礎。選定樣本之後則以電話聯繫訪談意願，了解受訪意願後再確定受訪單位與主要受訪者，受訪單位以設計研發部、文件資料管理部、資訊部為主，受訪者則以工作年資超過兩年以上的中高階主管與資深職員為主。

(3) 深入訪談

於受訪意願確定後，便開始進行深入的訪談調查。訪談的程序為：一、請主要受訪對象填寫結構化問卷。二、訪問主要受訪對象非結構化問卷內容。三、針對結構化與非結構化問卷內容，倘若主要受訪者不清楚問題內容或無法回答問題，則請其交由其他部門人員處理與回覆，以增加問卷答案的正確性。四、蒐集問卷主題相關資料、實際了解企業e化狀況，例如：蒐集產品開發流程圖、觀看企業e化系統等。

(4) 訪談結果分析與探討

藉由上述深入訪談程序，整理歸納結構與非結構化訪談問卷資料。整理的方式主要先從結構化問卷中，將受訪者的答案轉換為可量化之數據，而根據非結構化問卷訪談結果，修正與強化結構化問卷答案的品質及正確性，再利用微軟 Excel 軟體處理結構化問卷的量化數據，呈現基本的統計結果與圖示，然後再據此與文獻理論和產業環境趨勢作一探討。

4.2 訪談對象及其相關資料

由於國內中小製造企業集中於中南部地區，因此訪談對象選擇主要先以中南部地區之製造企業為主，經營型態以 ODM/OBM 為優先，除此之外企業還須具備設計研發與資訊管理部門。經初步條件篩選與訪談意願確定後，本調查主要深入訪談 10 家企業，其中 8 家位於中南部地區，2 家位於北部地區，訪談對象平均於受訪企業的工作年資為 7.2 年。茲將其相關資料整理如表 4-2 所示：

表 4-2 訪談對象及其相關資料

A			
公司名稱	主要產品	公司電話	訪談日期
彥豪	自行車煞車器、手把及零件等	886-4-8652164	2003/01/17
主要受訪對象	所屬部門	職稱	工作年資
蔡美慧	資訊管理部	工程師	10 年
公司地址：彰化縣埔鹽鄉彰水路二段 338 號			
B			
公司名稱	主要產品	公司電話	訪談日期
美耐敏	系統櫥櫃	886-4-24254488	2003/01/18
主要受訪對象	所屬部門	職稱	工作年資
楊恆輔	設計研發部	經理	4 年
公司地址：台中市西屯區文心路三段 289 號			
C			
公司名稱	主要產品	公司電話	訪談日期
美隆	各式揚聲器與相關產品	886-3-3261611	2003/01/29
主要受訪對象	所屬部門	職稱	工作年資
李龍潛	設計研發部	副理	20 年
蕭煌奇	文管中心	副理	3 年
公司地址：桃園市汴洲里春日路 1775 巷 77 號			

D			
公司名稱	主要產品	公司電話	訪談日期
迪友	電腦週邊產品塑膠射出	886-2-29093285	2003/01/29
主要受訪對象	所屬部門	職稱	工作年資
黃俊凱	設計研發部	設計師	4 年
公司地址：台北縣泰山鄉全興路 313 號			
E			
公司名稱	主要產品	公司電話	訪談日期
喬福	塑膠、泡棉相關產品開發	886-4-23596666	2003/02/12
主要受訪對象	所屬部門	職稱	工作年資
邱信華	文管中心	課長	8 年
公司地址：台中市工業區 26 路 1 號			
F			
公司名稱	主要產品	公司電話	訪談日期
精元	電腦鍵盤與周邊產品	886-4-25690248	2003/02/13
主要受訪對象	所屬部門	職稱	工作年資
余元懿	設計研發部	課長	7 年
劉侑因	設計研發部	設計師	4 年
公司地址：台中縣大雅鄉上楓村中正路 188-1 號			
G			
公司名稱	主要產品	公司電話	訪談日期
揚譜	各式揚聲器與相關產品	886-4-23580951	2003/02/14
主要受訪對象	所屬部門	職稱	工作年資
洪曉華	設計研發部	經理	10 年
公司地址：台中工業區 21 路 39 號			

H			
公司名稱	主要產品	公司電話	訪談日期
福裕	精密工具機	886-4-7991126	2003/02/14
主要受訪對象	所屬部門	職稱	工作年資
林成興	設計研發部	副理	7 年
公司地址：彰化縣伸港鄉興工路 34 號			
I			
公司名稱	主要產品	公司電話	訪談日期
瑞振	自行車相關零件	886-4-7616188	2003/02/17
主要受訪對象	所屬部門	職稱	工作年資
鄭玉情	設計研發部	設計師	5 年
陳立智	資訊部	工程師	3 年
公司地址：彰化縣秀水鄉義興村三義巷 50 號			
J			
公司名稱	主要產品	公司電話	訪談日期
穩正	高爾夫球車與相關產品	886-6-2536766	2003/04/10
主要受訪對象	所屬部門	職稱	工作年資
鄭自強	資訊部	經理	13 年
林耿輝	設計研發部	設計師	3 年
公司地址：台南縣永康市民東路 5 號			

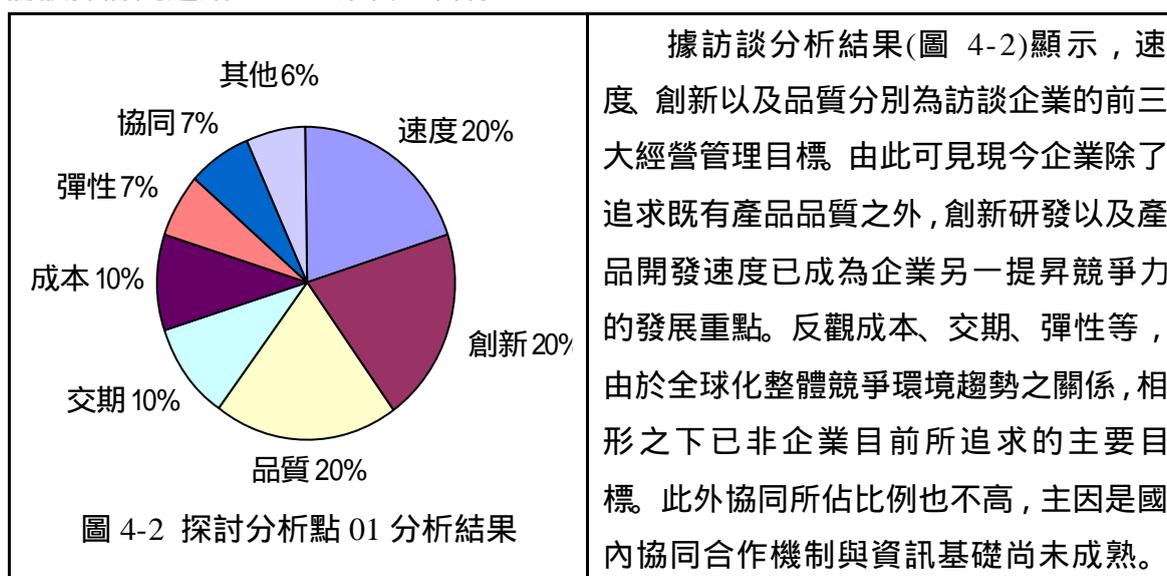
4.3 調查結果分析與探討

經結構與非結構化問卷訪談之後，根據本調查之主題，整理歸納幾個主要訪談探討問題點，如表 4-3 所示。

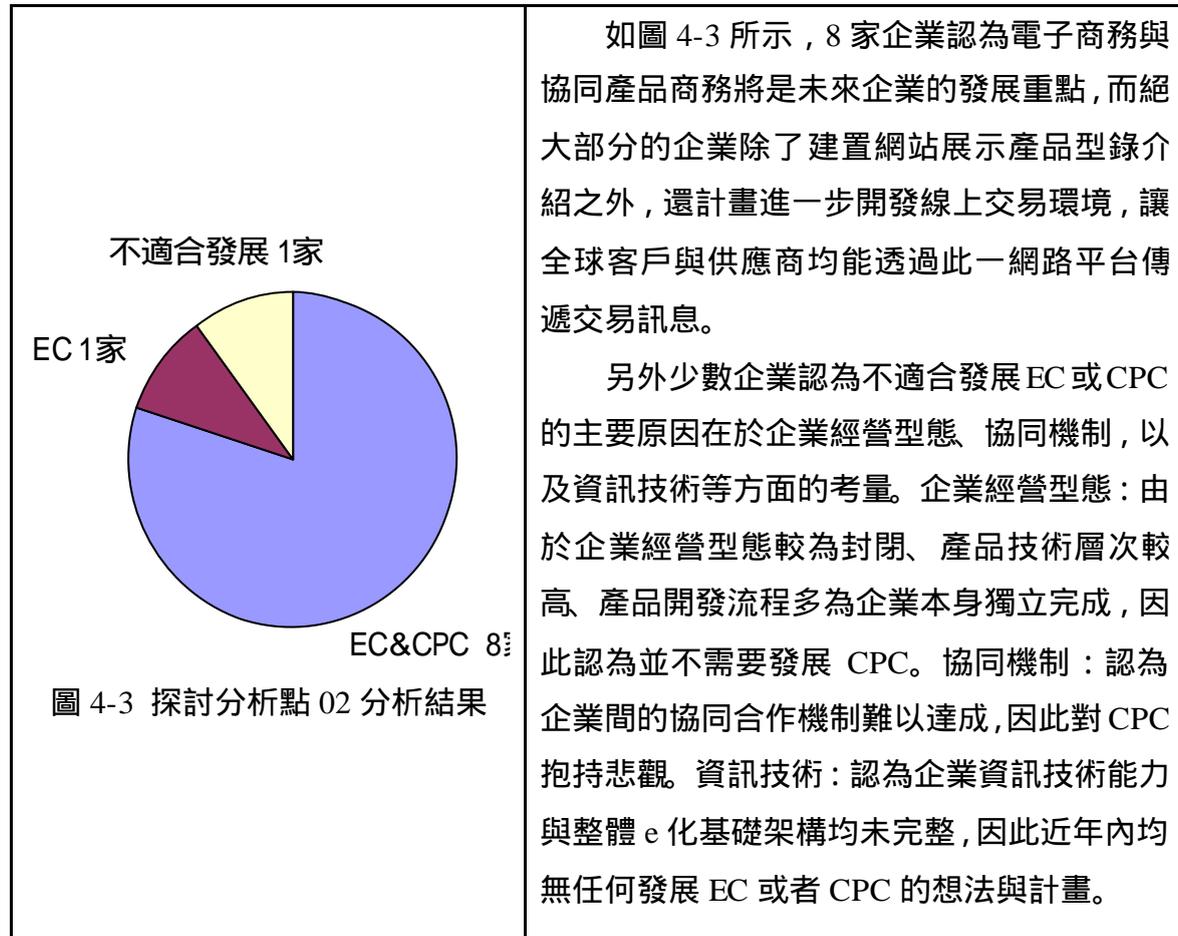
表 4-3 主要訪談探討問題點

主題	訪談探討問題點
企業發展策略	01. 企業管理目標 02. 電子商務與協同產品商務 03. 企業 e 化核心系統
企業電子化現況	04. 企業 e 化系統投資預算 05. 企業 e 化困難點 06. 企業 e 化成效
產品開發模式	07. 產品開發模式 08. 工程設計變更
產品資料管理	09. 產品資料管理方式 10. 設計資料使用方式 11. 產品資料管理系統 12. 企業產品資訊整合趨勢看法

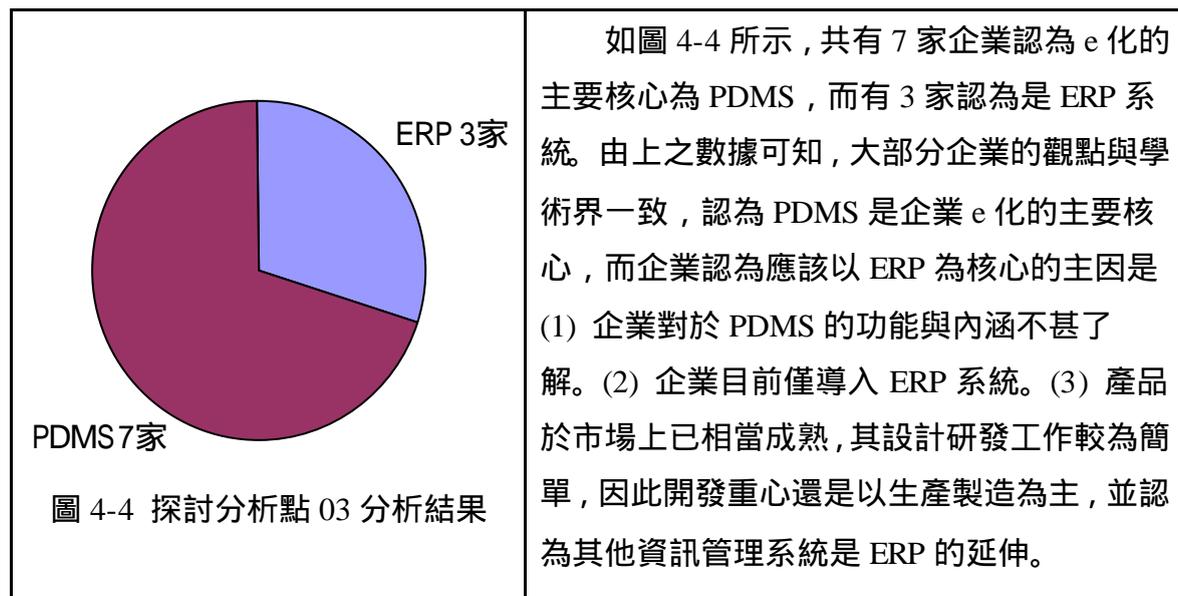
訪談探討問題點 01：企業管理目標



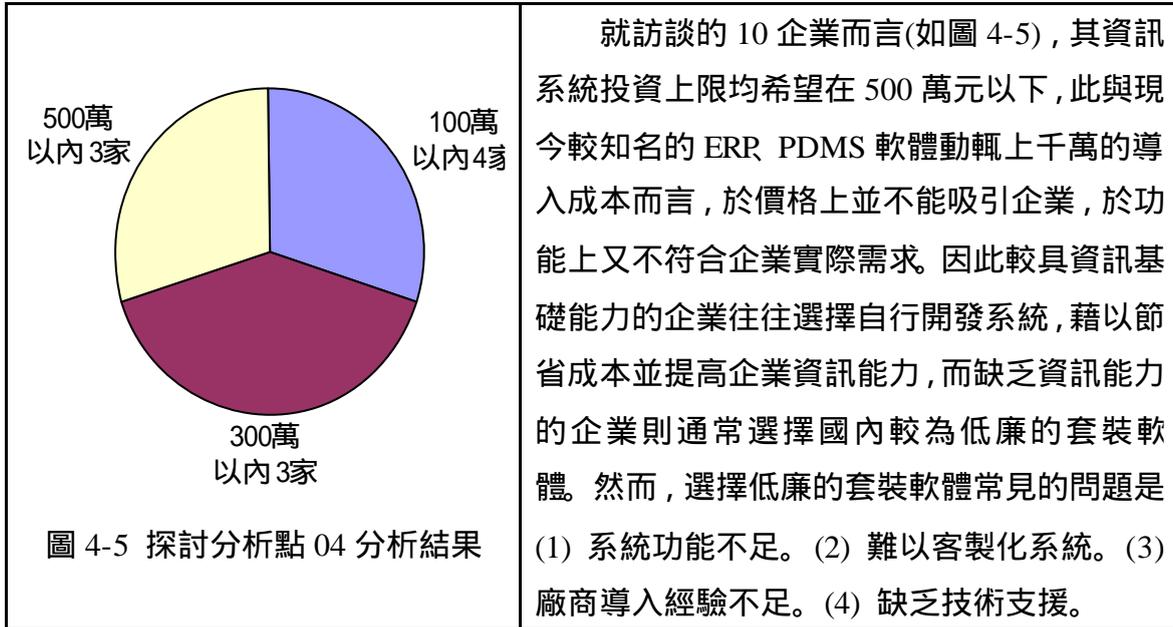
訪談探討問題點 02：電子商務與協同產品商務



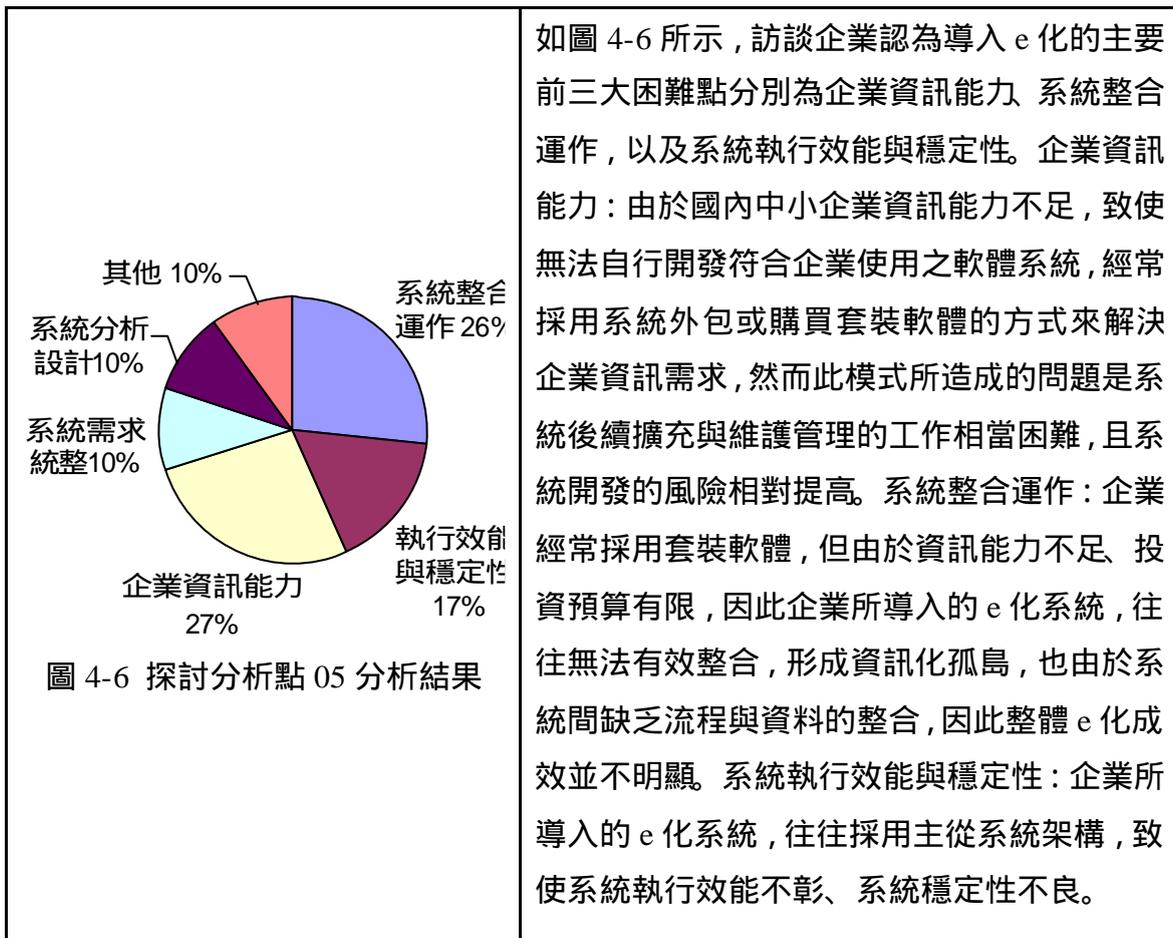
訪談探討問題點 03：企業 e 化核心系統



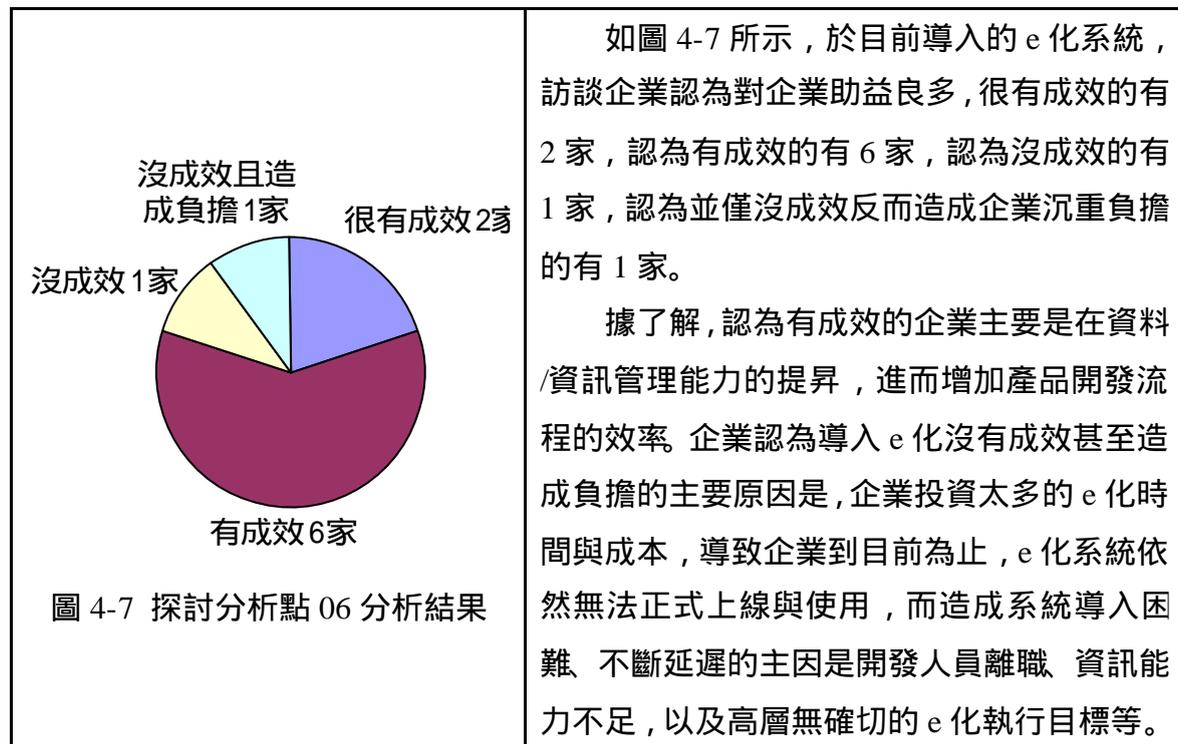
訪談探討問題點 04：企業 e 化系統投資預算



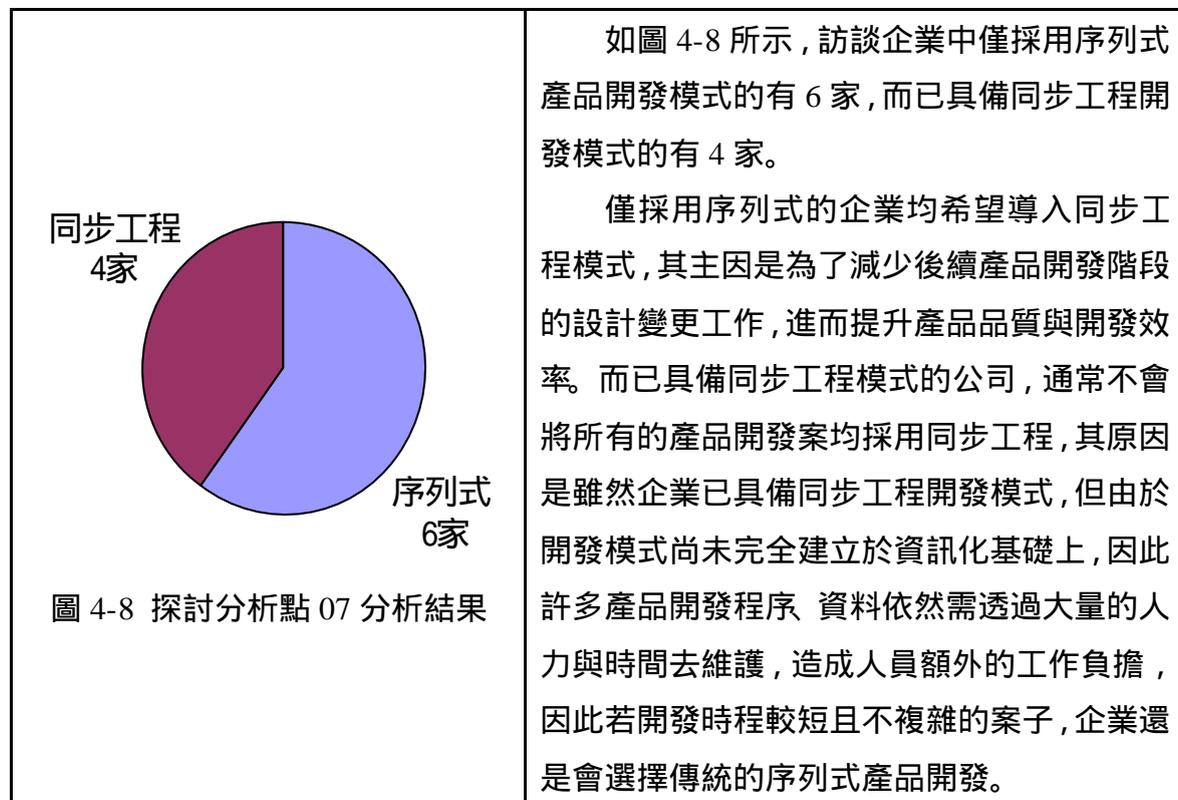
訪談探討問題點 05：企業 e 化困難點



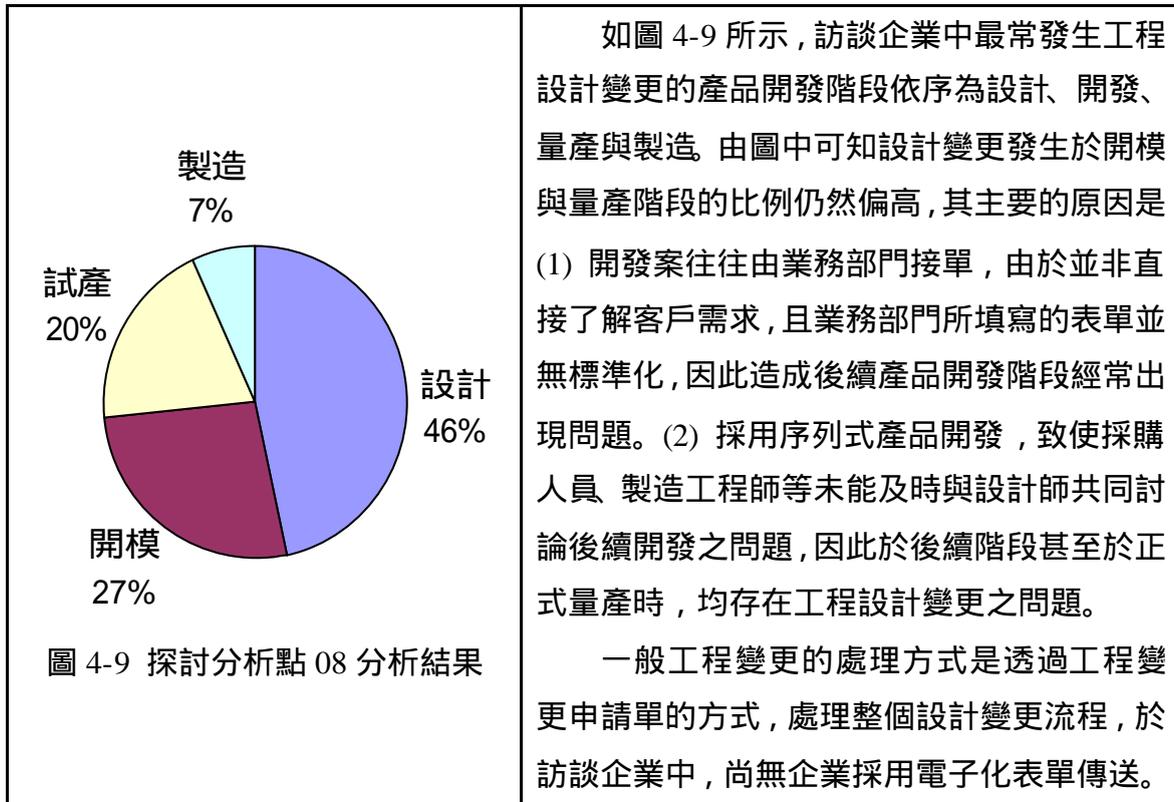
訪談探討問題點 06：企業 e 化成效



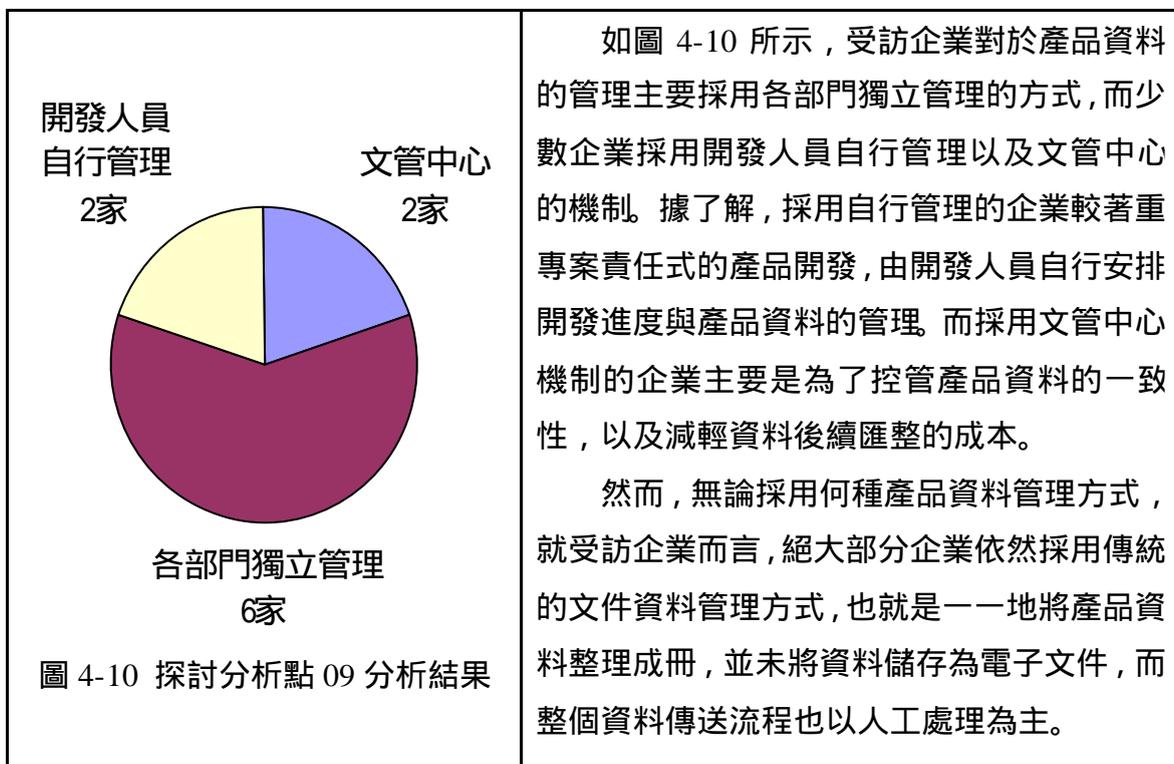
訪談探討問題點 07：產品開發模式



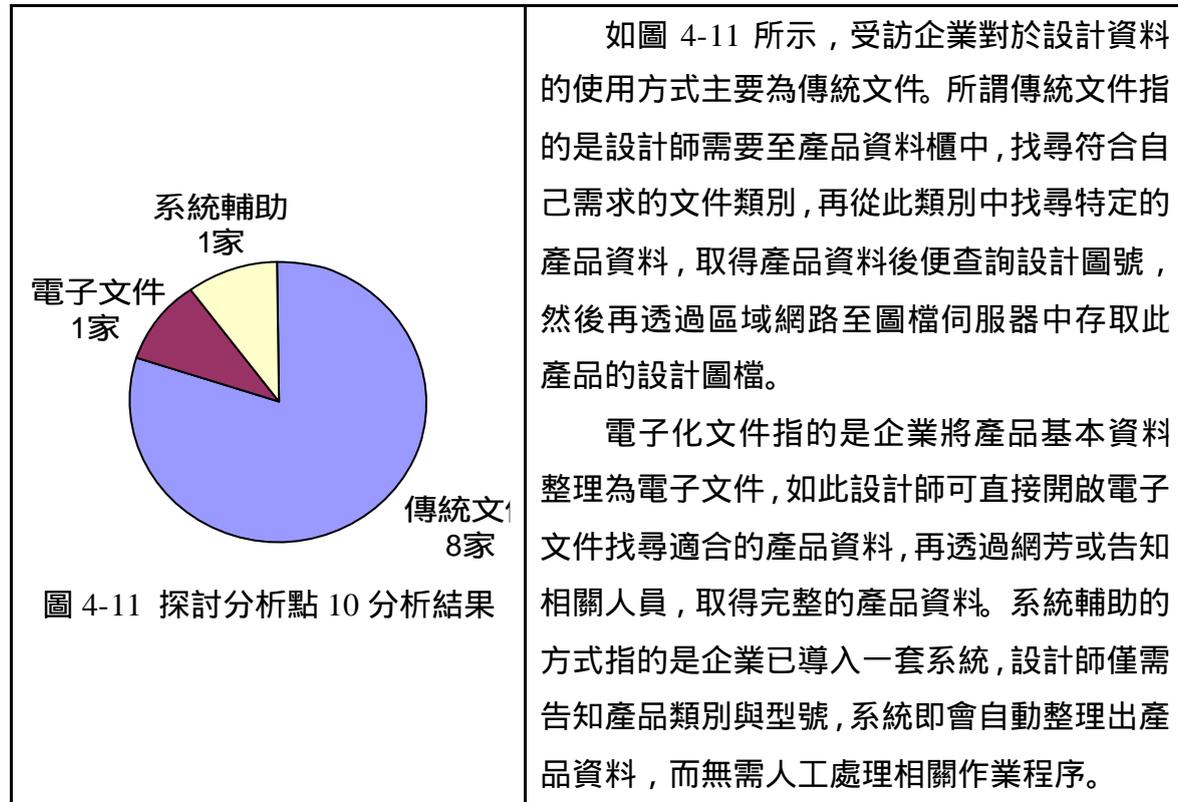
訪談探討問題點 08：工程設計變更



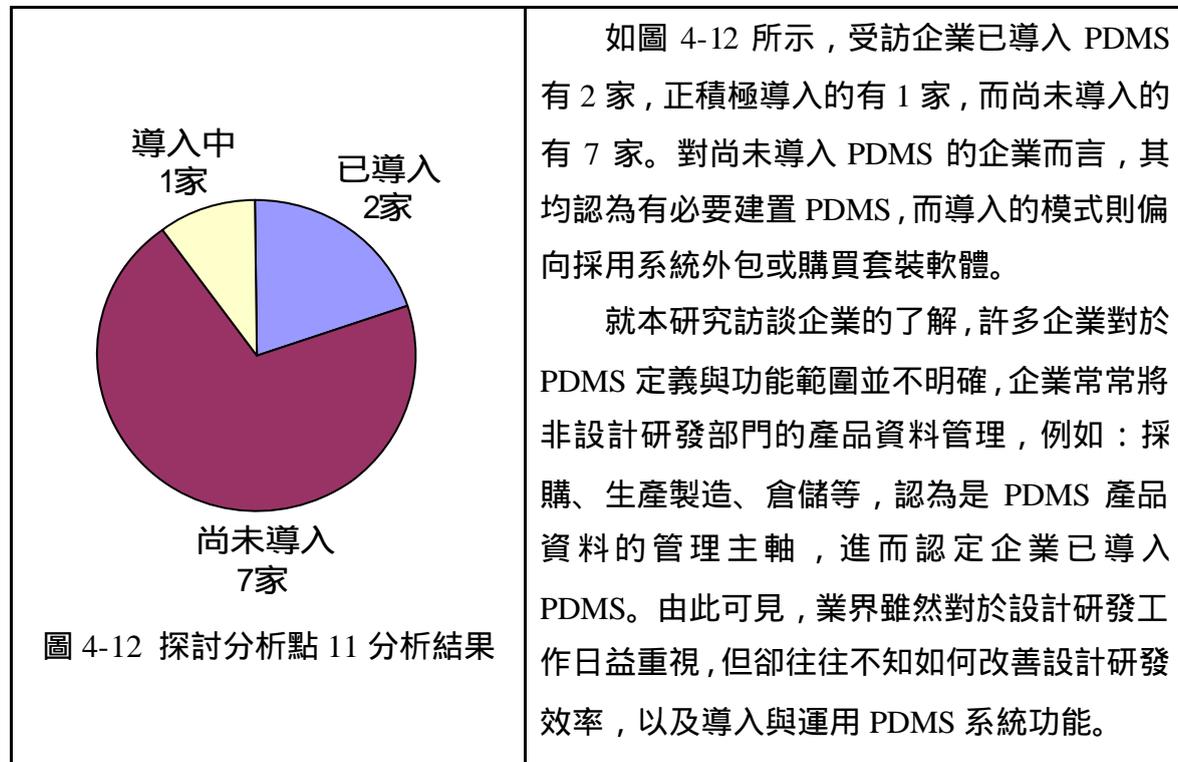
訪談探討問題點 09：產品資料管理方式



訪談探討問題點 10：設計資料使用方式



訪談探討問題點 11：產品資料管理系統



訪談探討問題點 12：企業產品資訊整合趨勢看法

茲就訪談企業所提出之看法整理如下：

- 產品資訊整合為現今與未來不可避免的趨勢，且將提升企業營運效率。
- 若企業採用套裝軟體，則難以達到資訊整合運用的目標。
- 上下游廠商能否密切配合為產品資訊整合的關鍵。
- 產品資訊整合的目的在於提昇產品開發效率，快速推出新式產品滿足客戶需求。
- 開發人員需深入了解產品作業流程及系統溝通介面，才能達到產品資訊整合。
- 高層主管需多參與產品資訊整合開發，提供開發資源與需求意見。
- 產品資訊整合困難，主因是企業與上下游廠商缺乏資訊能力，系統難以整合。

4.4 訪談調查結論

就本研究訪談的 10 家國內中小企業與上述 12 項探討分析點而言，企業意識到傳統代工產業已無法於全球化競爭環境中繼續保持優勢，如今企業已逐漸將發展的重心轉移到產品的設計研發上，管理的目標也趨向於追求產品創新、品質提昇，以及縮短上市時間為主。對於電子商務與協同產品商務的發展，絕大部分企業抱持樂觀的態度，認為企業有必要將目前的實體營運結合電子商務，藉以擴展行銷通路、降低商業交易成本；企業發展 CPC 的主因是希望能減少產品資料、資訊流通的阻礙，進而縮短整個產品開發的時程。

於企業經營管理目標與 e 化發展策略上可發現，大部分企業的主要策略與觀念與學術理論差異不大，然而於實際執行面上，往往因資訊能力不足，實行步驟或方法錯誤等因素，導致不但無法達到 e 化成效目標，反而浪費許多建置時程與成本。就 PDMS 的導入與運用而言，目前絕大多數企業尚未具備基本的產品電子型錄、搜尋引擎以及資料管理等系統機制，依然以人工處理產品文件資料為主，不僅浪費許多產品開發時程，更增加設計研發人員的工作負擔，進而形成企業提升設計研發效能的一大阻礙。

由於本調查以案例研究為主，因此於樣本數上並無法完全代表國內中小企業，然而在問題點探討及分析上，可真實反應與了解受訪企業的實際營運情形，而本調查所整理之企業現況與問題點，可提供企業與學術作一參考，用以檢視企業或者作進一步的量化與統計分析。

第五章 系統開發

本章將統合前述企業訪談所了解之常見設計管理問題，作為系統功能開發的背景基礎，並以瑞振企業的產品資料與實際狀況為例，運用元件模型技術、Web Service 以及分散式系統架構，建置一 Web Service-Based PDMS。首先會針對使用者需求作深入的訪談與資料的收集，而後根據系統開發範圍，運用統一塑模語言進行系統的分析與設計，最終則進行實際系統開發，將系統功能元件化、網路服務化，再針對整個系統的功能與架構，探討其對企業的主要效益為何。

5.1 深入訪談與系統功能界定

一個完整的產品資料管理系統，其功能範圍涉及整個產品開發生命週期，然而 PDMS 主要是以設計研發資料與流程的管理作為系統發展之核心。因此，本研究系統開發範圍將以瑞振企業的設計研發部門為主，以產品資料管理、作業流程改善、資訊整合運用等角度，深入訪談與了解企業問題以及使用者需求為何。

表 5-1 整理歸納本研究深入訪談企業所發現之問題點與需求點，從表中 12 項之敘述可知，問題產生的主要原因在於企業未以電子化、系統化的方式管理產品設計研發資料、零件資料、模具資料以及相關電子圖檔資料等，造成開發時程與人力資源龐大的浪費。設計師花費於整理、搜尋產品相關資料的時間，遠比其專心於產品設計研發工作上的時間還多，目前的產品資料管理方式，不僅增加設計人員的工作負擔，更使得設計研發效能無法有效提昇。為解決表 5-1 所整理之問題與使用者需求，本研究所開發之 Web Service-Based PDMS 主要具備以下系統功能(詳細說明如表 5-2 所示)：

1. 圖檔系統化管理功能。
2. 產品型錄資料展示功能。
3. 設計資料整合。
4. 搜尋引擎及線上 2D 圖檔瀏覽。
5. 零件資料查詢。
6. 模具資料查詢。
7. 產品資料管理功能。
8. 元件化及網路服務化。

表 5-1 瑞振企業設計研發部問題或需求點

編號	問題或需求點說明
P.01	產品設計資料的管理方式主要以紙張整理成冊，收納於資料櫃中。日積月累之後造成資料龐雜難以找尋與管理。
P.02	公司當初所導入的 ISO 制度，對於產品、零件等圖檔編碼方式過於繁雜，且存在許多不合理的狀況，例如：類別編號重複、編碼方式不明確等。
P.03	設計師必須先查詢圖檔編碼原則，然後再仔細編寫複雜的圖檔圖號，造成設計師額外的負擔。
P.04	同一份圖檔資料，可能有不同的圖號路徑，導致資料重複建置或不知正確存放位置為何。
P.05	由於圖檔的存放方式是以圖號為主，因此設計師必須先根據圖號路徑，於圖檔伺服器中建立許多資料夾，然後再存放圖檔資料，造成圖檔存放程序繁雜且費時。
P.06	設計師僅能從產品型錄或資料櫃中，找尋相關產品資料作為設計參考，若欲取得設計圖檔，則需根據圖號，透過區域網路登入至圖檔伺服器一一搜尋，然後再將圖檔取回開啟，若不符合參考之用，則重複此運作流程，造成時間與人力的浪費。
P.07	公司網站僅提供型錄資料之瀏覽，未具備設計資料之檢視與管理，因此對於設計師的參考價值與使用效益較低。此外網站產品資料與公司內部產品資料，分別獨立存放於不同的資料庫中，因此造成資料不一致以及維護管理成本增加等問題。
P.08	由於網站的產品資料庫採用 MS Access，系統執行效能與安全防護較低，因此難以滿足商業環境以及日後系統擴充之需求。
P.09	公司以 MS Excel 存放模具相關資料，較難以維護管理與應用。此外設計師希望能有一介面，用以查詢及管理模具資料。
P.10	目前公司所導入的科展系統主要以製造、財會、採購等方面的功能為主，其所提供之設計資料管理功能相當薄弱，無法滿足設計研發部門之需求。此外由於科展公司並未提供必要的協助，因此難以於科展系統上額外增加產品資料管理功能。
P.11	設計師為了製作工程物料清單，其必須花費相當多的時間整理相關產品、零件資料，以及製作、傳送表單。
P.12	設計師希望能有一系統，整合與呈現產品相關資料，例如：設計圖檔、設計概念、產品零件資料等。

表 5-2 Web Service-Based PDMS 主要系統功能

編號	功能名稱	功能說明
F.01	圖檔系統化管理功能	將原本複雜的編碼資料與處理原則【如附錄 B 所示】，分別存放於資料庫與系統程式中。設計師使用系統時，僅需填寫產品或零件的型號，系統將根據編碼原則，自動整理出所有圖檔的命名格式，而設計師僅需根據此命名格式，上傳符合檔名的檔案，後續的資料夾建立與相關圖檔存放等動作，系統均將自動化處理完成。
F.02	產品型錄資料展示功能	將同一類別的所有產品型錄照片與基本資料，以分頁的方式呈現於同一系統畫面，提供設計師快速參考與找尋符合其需求條件的產品資料。
F.03	設計資料整合	將型錄、設計、圖檔、工程物料清單等產品資料，整理呈現於單一系統畫面，如此不僅能減少設計師找尋產品資料的操作程序，更可取得完整的設計研發參考資料。此外，由於所有設計相關資料均已由單一系統畫面所管理，因此也可降低管理者維護資料的難度。
F.04	搜尋引擎及線上 2D 圖檔瀏覽	提供產品、零件、模具資料搜尋，讓設計師與管理者能透過此功能介面輸入型號，快速找尋相關資料。提供一介面功能，讓使用者可直接透過網路瀏覽產品、零件、模具等 2D 圖檔以及相關資料。
F.05	零件資料查詢	將零件資料與圖檔分門別類系統化管理，讓使用者可透過多種方式找尋符合條件的零件資料。
F.06	模具資料查詢	將模具廠商、模具資料、模具圖檔統整作系統化的管理，讓使用者可透過操作介面，快速查詢模具相關資料。
F.07	產品資料管理功能	提供型錄、設計、零件、模具、圖檔等產品相關資料一新增、修改、刪除的系統化管理功能。
F.08	元件化及網路服務化	將系統功能元件化與網路服務化，使系統整體架構及功能更具彈性、通透性，藉以降低後續維護管理與系統整合工作的複雜度與困難度。

5.2 系統開發環境

茲將本系統開發環境、工具、技術整理如表 5-3，系統架構圖如圖 5-1 所示：

表 5-3 系統開發環境

項目	開發技術、方法或工具
前端使用者操作介面	微軟 Internet Explorer 6 瀏覽器
後端伺服器語言	Active Server Page(ASP)
後端伺服器作業系統	Windows 2000 Server
網頁伺服器	Internet Information Services 5(IIS5)
元件模型標準	COM+
元件與 Web Service 開發工具	Delphi6 SP2
資料庫管理系統	MS SQL 2000 Server
網頁設計工具	Dreamweaver 4
XML 文件物件模型	MS XMLDOM
系統分析與設計工具	Unified Modeling Language(UML)
SOAP 版本	1.1
DWF 檔案瀏覽軟體	Autodesk Express Viewer

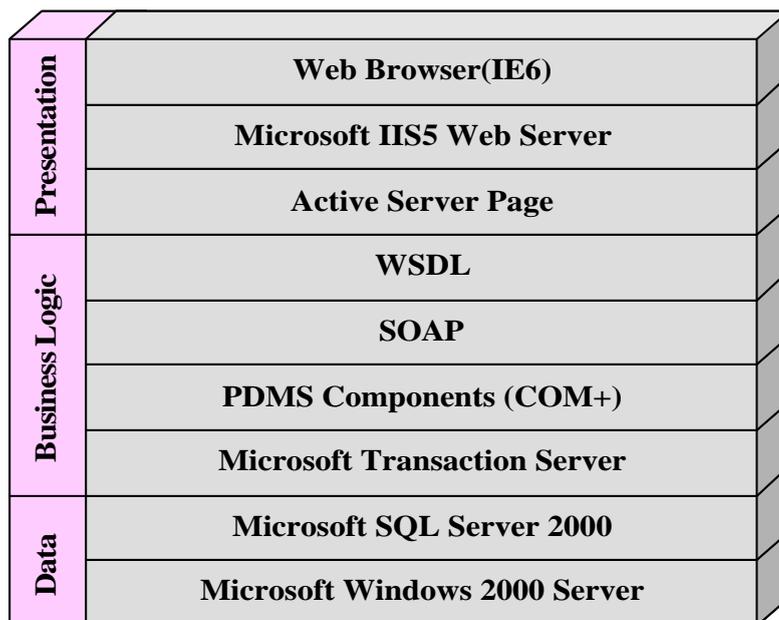


圖 5-1 系統架構圖

5.3 系統分析與設計

由於 UML 已是現今描述系統的標準語言【49】，因此在 5.1、5.2 節取得使用者需求與系統主要開發功能之後，本節將以 UML 發源地 Rational Software 所提倡的 UML 4+1 View (如圖 5-2，修改自【48】)，進行 Web Service-Based PDMS 系統分析與設計。

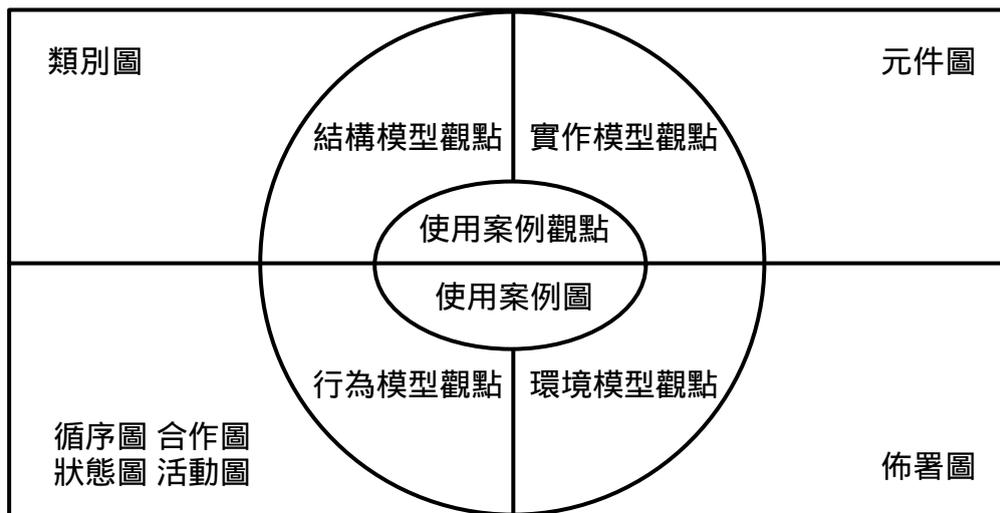


圖 5-2 UML 4+1View

圖 5-2 的 4+1 View 主要觀點內容如下：

- 使用案例觀點(Use Case View)：以系統的外部使用者角度觀看系統。使用案例觀點所描述的系統功能，主要由外部使用者或者其他系統所觸發，而經觸發啟動的系統功能，則對外部使用者或其他系統提供服務。使用案例觀點是 UML 4+1 觀點的核心和基礎，也是其他 UML 圖形的主要發展來源。
- 結構模型觀點(Architectual Model View)：又稱為邏輯觀點(Logic View)，主要描述物件類別如何互動運作以完成系統功能，屬於系統的靜態觀點或內部觀點。
- 行為模型觀點(Behavioral Model View)：描述一個程序或一群工作任務如何執行運作以完成系統功能，又稱為動態、程序、同時，以及合作觀點。
- 實作模型觀點(Implementation Model View)：以模組或元件表示動靜態觀點的物件是如何被實作，並以元件圖表示出它們之間的相依性。
- 環境模型觀點(Environment Model View)：又稱為佈署觀點，主要運用佈署圖充分表示系統軟硬體運作的實際狀況，例如：需要哪些電腦設備及如何連接等。

UML 之所以成為塑模標準的主要原因，在於其所提供之觀念、圖形及符號可以完整描述真實系統，作為系統開發的標準藍圖，讓設計者、實作者與使用者均能透過 UML 視覺化圖形表達方式，了解整個系統功能與架構。UML 各個觀點所定義的圖形與圖形功能如表 5-4 所示。

表 5-4 UML 各個觀點所定義之圖形與功能(資料來源：修改自【48】)

觀點	圖形	功能及目的
使用案例觀點	使用案例圖(Use Case Diagram)	以使用者的角度描述系統功能，找出 Use Case 與 Actor，並界定系統的範圍
結構模型觀點	類別圖(Class Diagram)	表示物件類型之間結構上的關聯。
行為模型觀點	循序圖(Sequence Diagram) 合作圖(Collaboration Diagram)	表示物件之間動態得溝通模式。
	狀態圖(State Diagram)	表示物件狀態的變化。
	活動圖(Active Diagram)	表示業務流程或一類型中單一運作(Operation)的邏輯流程。
實作模型觀點	元件圖(Component Diagram)	表示模組或元件之間的關聯。
環境模型觀點	佈署圖(Deployment Diagram)	表示程序(Process)或執行緒(Thread)對應到主機或裝置的實際狀況。

以下將以 UML 的各個觀點與其圖形描述本研究之 Web Service-Based PDMS 的整體架構與功能。

使用案例觀點：

根據表 5-2 所整理之系統功能，Web Service-Based PDMS 有 4 個主要子系統：產品資料管理系統、零件資料管理系統、模具資料管理系統，以及搜尋引擎；4 個主要動作者：產品資料需求者、產品資料管理者、企業內外異質系統，以及產品資料庫。由整個系統功能與 4 個主要動作者所建構而成的 UML Use Case View 如圖 5-3 所示。

圖 5-3 中的資料需求者可使用產品資料、搜尋引擎、零件資料，以及模具資料等功能，一旦使用系統功能後，系統將啟動 Web Service 與 PDMS 元件提供服務；企業內外異質系統也可啟動 Web Service 間接使用 PDMS 元件模組；資料管理者主要透過管理產品資料案例管理整個產品資料庫，而產品資料庫主要滿足外界各式產品資料需求。

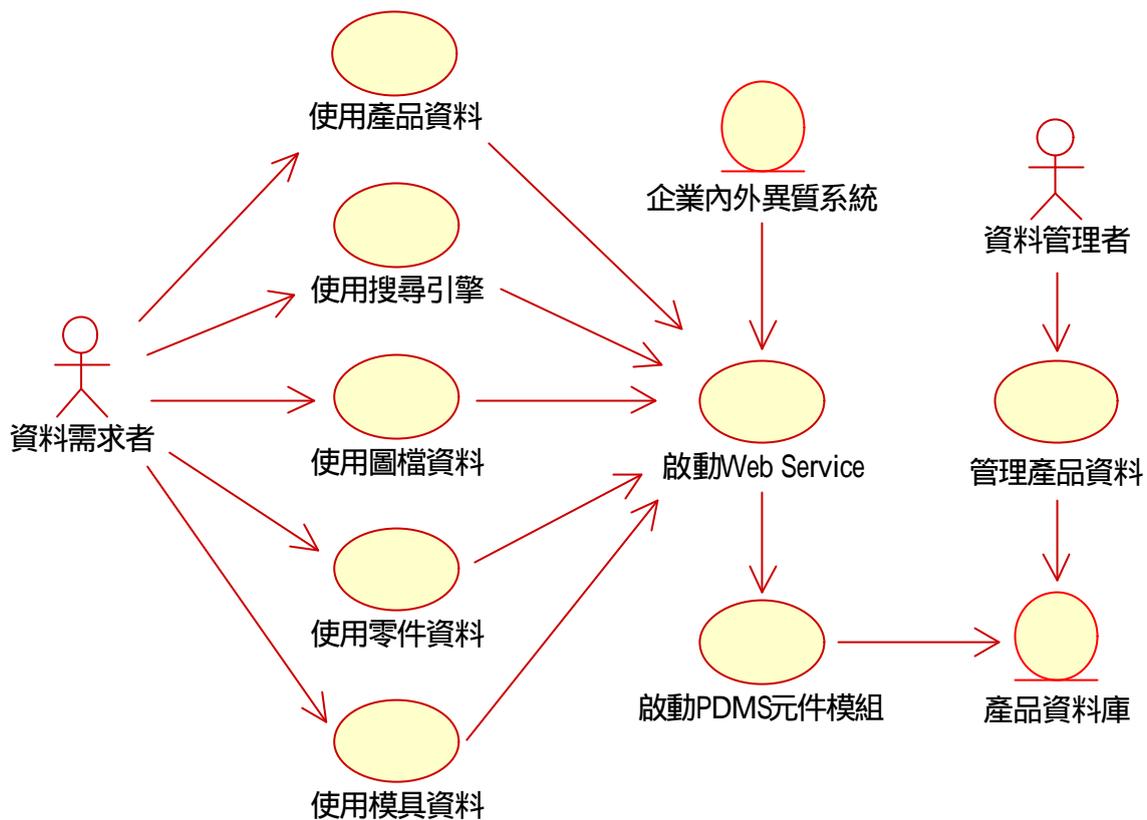


圖 5-3 Web Service-Based PDMS 使用案例圖

結構模型觀點：

完成使用案例圖形之後，則可根據 UML 的 Boundary-Control-Entity 模型(UML B-C-E Model)，找尋三層架構中的物件類別，以下分別說明 B-C-E 模型所代表的意義與找尋的方法：

- 一、 邊界類別(Boundary Class)：邊界類別的功能在於釐清使用者在系統邊界上的需求，或者塑造動作者與系統之間的互動。可將其視為系統的邊界，而一個系統的邊界可能是與其他系統之間互動的實際介面，也可能是與使用者互動的使用者介面(User Interface：UI)。
- 二、 控制類別(Control Class)：用以協調物件之間的交易或將複雜的計算、企業邏輯加以封裝，通常即代表三層式系統架構中，商業邏輯層的類別物件。
- 三、 實體類別(Entity Class)：代表系統永久保存的資料或資訊，通常以資料庫的實體關係模型作為找尋實體類別的媒介。

表 5-5 即根據上述之觀念與方法，整理出本系統 UML B-C-E 模型中的各個類別。

表 5-5 UML B-C-E 模型

Use Case	Boundary Class	Control Class	Entity Class
使用產品資料	Product UI	Product Process	產品/零件/E-BOM/設計圖檔
使用搜尋引擎	Search UI	Search Process	產品/零件/模具
使用圖檔資料	Design File UI	Design File Process	設計圖檔
使用零件資料	Part UI	Part Process	零件/設計圖檔
使用模具資料	Mold UI	Mold Process	模具/模具廠商/設計圖檔
啟動 Web Service	WS Interface	WS Process	Null
啟動 PDMS 元件模組	PDMS Interface	PDMS Process	產品/零件/模具/模具廠商/設計圖檔/E-BOM
管理產品資料	Management UI	Management Process	產品/零件/模具/模具廠商/設計圖檔/E-BOM

根據表 5-5 所整理的 UML B-C-E 各個階段的物件類別，即可繪製三層式架構類別圖。為避免圖形過於龐大，本文省略重覆部分，僅表達主要系統類別，如圖 5-4 所示。

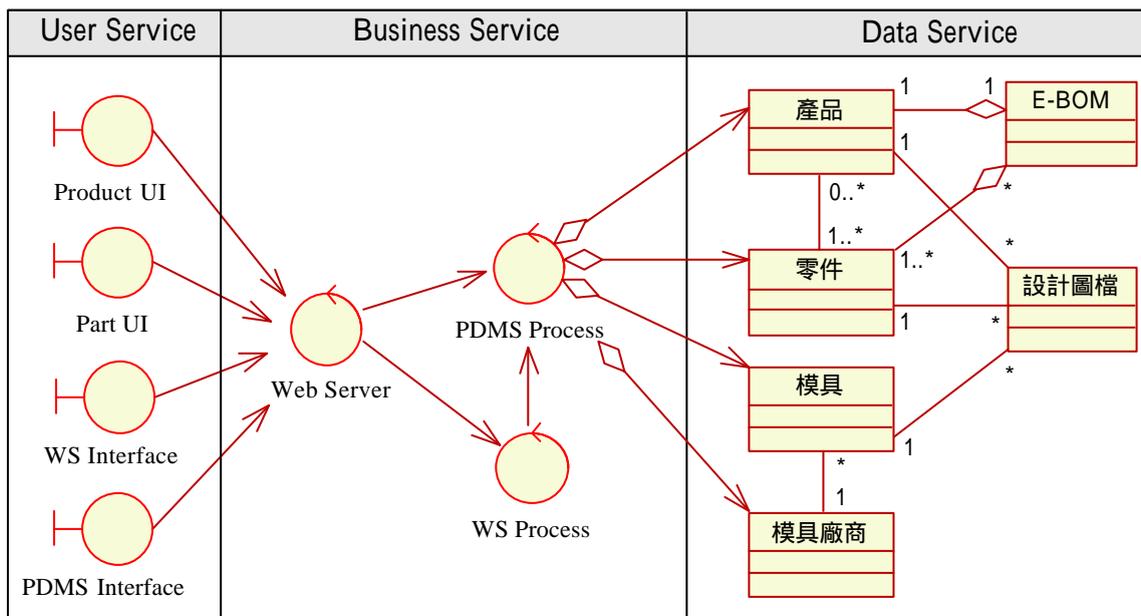


圖 5-4 Web Service-Based PDMS 類別圖

行為模型觀點：

行為模型觀點的圖形有循序圖、合作圖、狀態圖，以及活動圖，其中循序圖與合作圖是由使用案例圖所延伸，狀態圖與活動圖則由類別圖所延伸。行為模型觀點圖形的主要功能在於表達使用案例圖、類別圖中，物件類別間的時間關係以及順序關係。

由於本系統中許多物件或使用案例的行為幾乎完全一樣，例如：使用產品資料、使用零件資料、使用模具資料等，因此為避免圖形重覆且過於龐大難以表示，本節將以系統中的零件相關類別與使用案例作為主要分析設計樣本，茲將行為模型各式圖形分別表示如下：

1. 循序圖：

本文以圖 5-5 作為系統循序圖之代表，其為一取得零件資料的順序，其流程如下：

- (1).使用者登入至零件頁面(part.asp),而後 Web Server 回應(Response)使用者介面
- (2).呼叫 GetPartData 後，系統先啟動 Web Service，再由 Web Service 呼叫 PDMS 元件中的函式。
- (3).PDMS 元件函式使用零件與設計圖檔資料庫，最終將處理完的結果透過 Web Server 回應前端。

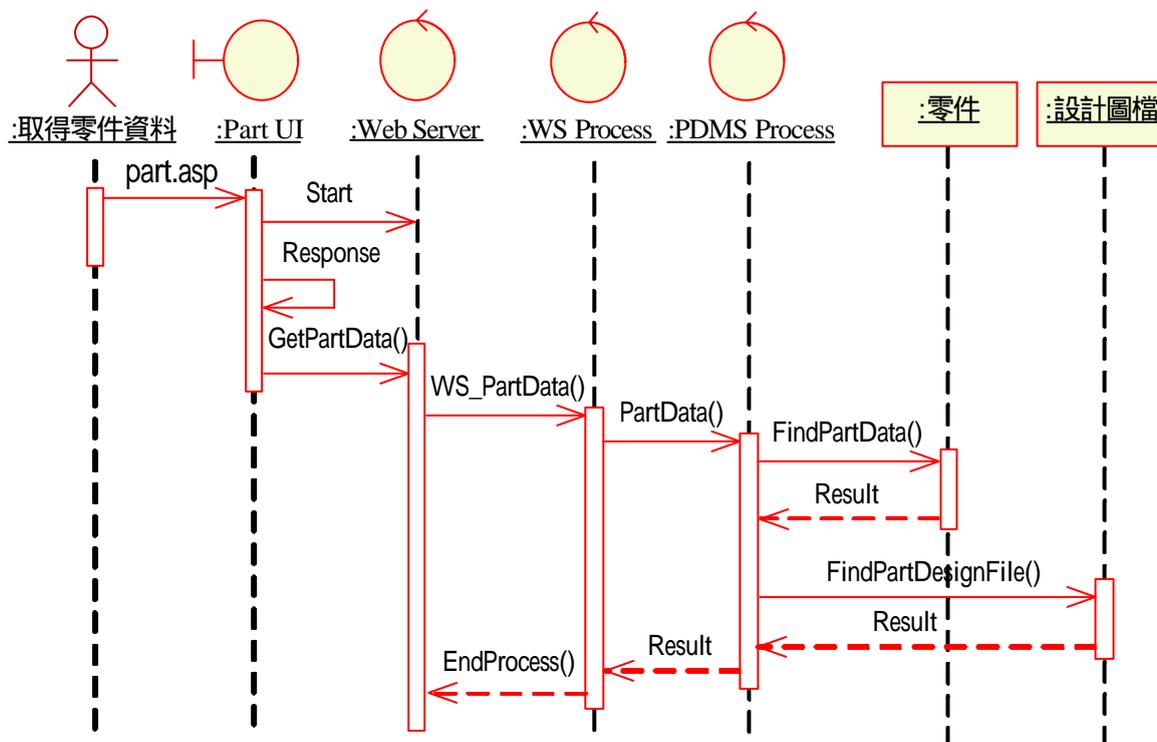


圖 5-5 零件資料取得循序圖

2. 合作圖：

基本上合作圖與循序圖所表達的內容相同，但表達的方式不同。循序圖偏向強調的是物件互動的時間順序關係，而合作圖強調的是物件互動的訊息傳遞關係。以圖 5-5 為基礎的合作圖如圖 5-6 所示：

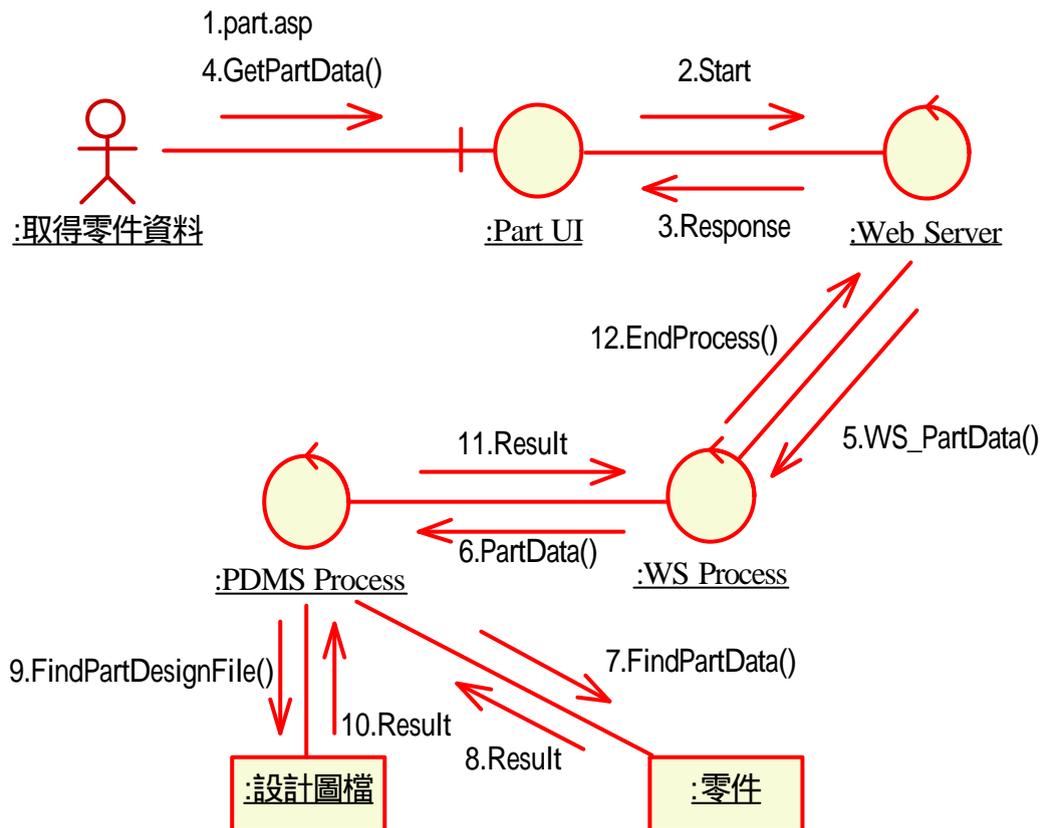


圖 5-6 零件資料取得合作圖

3. 狀態圖：

零件資料取得狀態圖如圖 5-7 所示，主要狀態分為：

- (1).執行 Web 程式：啟動伺服器端程式，並且讓使用者等待處理結果。
- (2).系統處理狀態：透過取得 Web Service 與 PDMS 元件模型介面之後，即呼叫系統所提供之功能，取得零件資料與零件圖檔。
- (3).整合資料與資訊：將系統處理狀態最終所傳回之查詢結果加以處理，賦予使用者介面。
- (4).回應使用者：網路程式處理完畢之後，透過 Web Server 回應前端使用者。

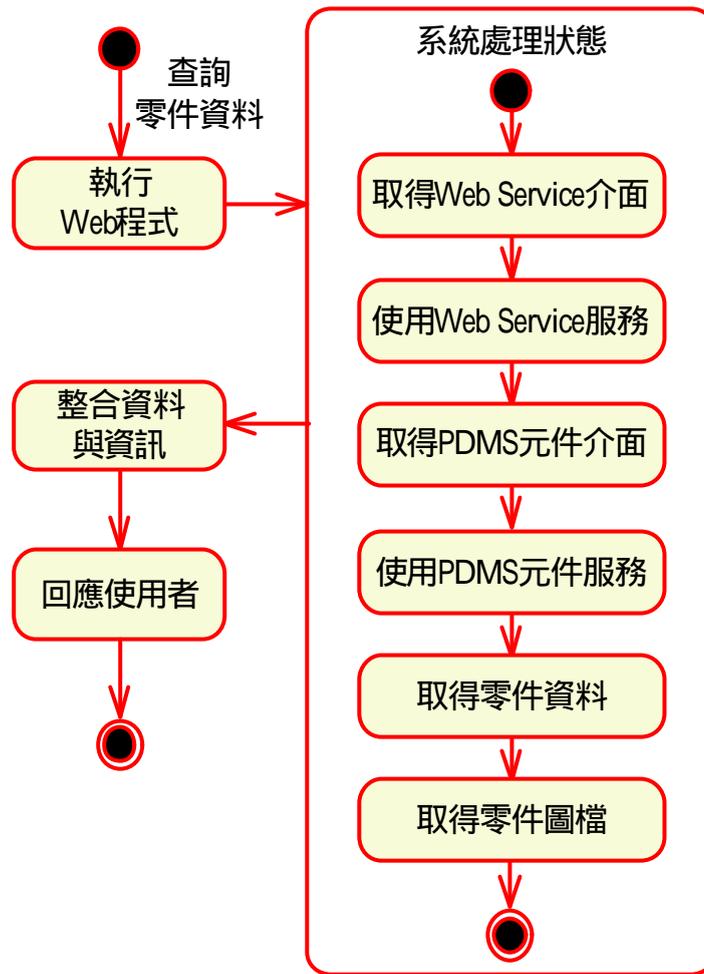


圖 5-7 零件資料取得狀態圖

4. 活動圖：

活動圖主要描述系統內外部流程與行為。利用前述之 UML B-C-E 模型與三層式系統架構觀念，可將整體系統劃分為三個主要類別環境，分別為 User Service(表現層)、Business Service(商業邏輯層)，以及 Data Service(資料層)。依照這三大主要層級觀念，可分別找出屬於其環境範圍的活動流程，茲將整個零件資料取得的活動過程以圖 5-8 表示。圖 5-8 與圖 5-7 之系統邏輯流程大致一樣，然而圖 5-8 之活動圖除了可描述流程活動位於那一類別環境中，更可加入判斷式、分支、分叉、會合等流程符號，提供更詳盡且豐富的系統流程活動設計。

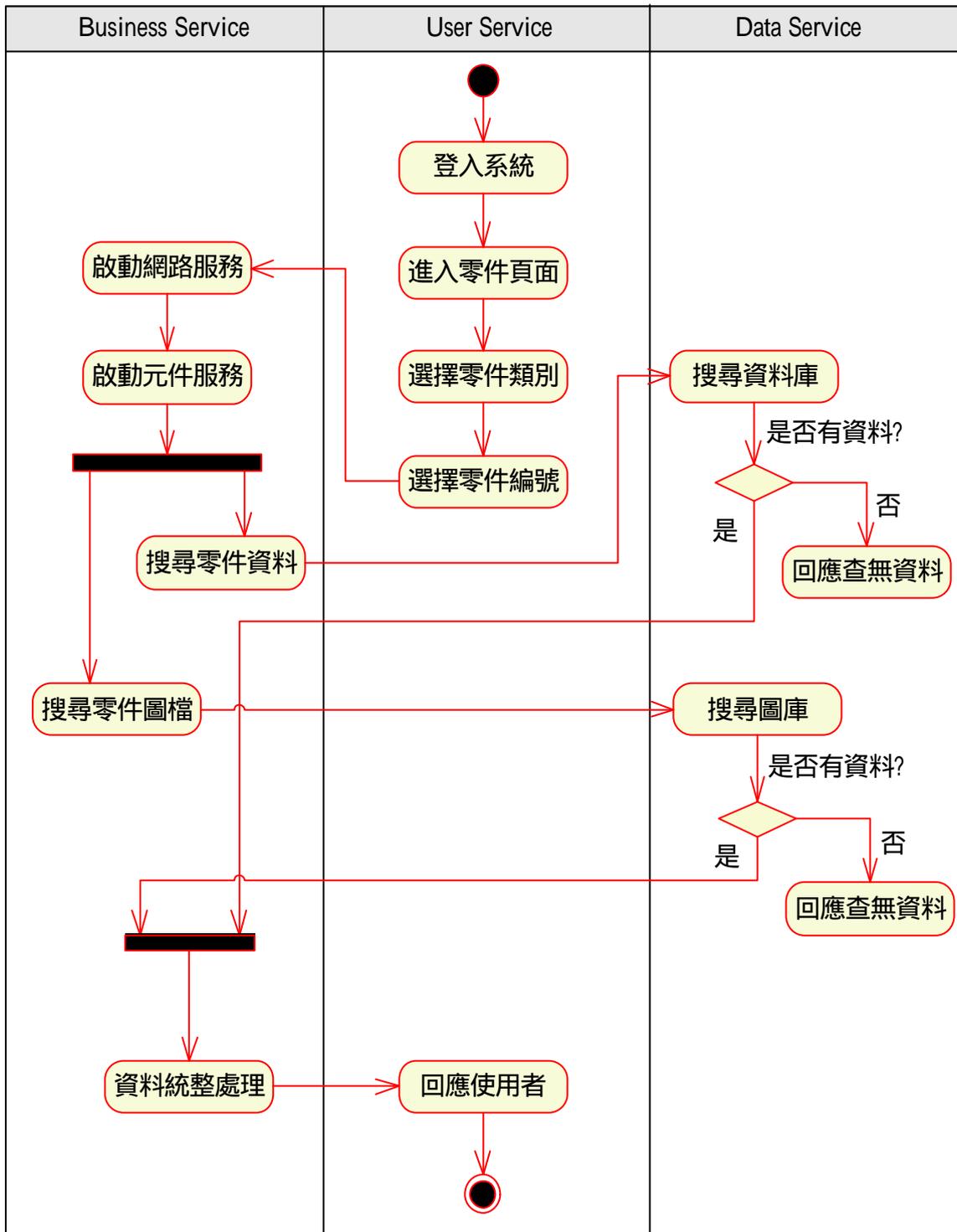


圖 5-8 零件資料取得活動圖

實作模型觀點

實作模型觀點主要是透過元件圖，表達各軟體元件於編譯或執行期間的相依性。所謂元件指的是具有完整介面的軟體模組，例如：二進位碼元件、原始碼元件、可執行元件、DLL 元件等，而一般元件圖的描繪範圍在於用戶端主機與伺服器端主機上執行的程式。根據三層式架構與 UML B-C-D 模型，以使用零件資料為例之元件圖如圖 5-9 所示：

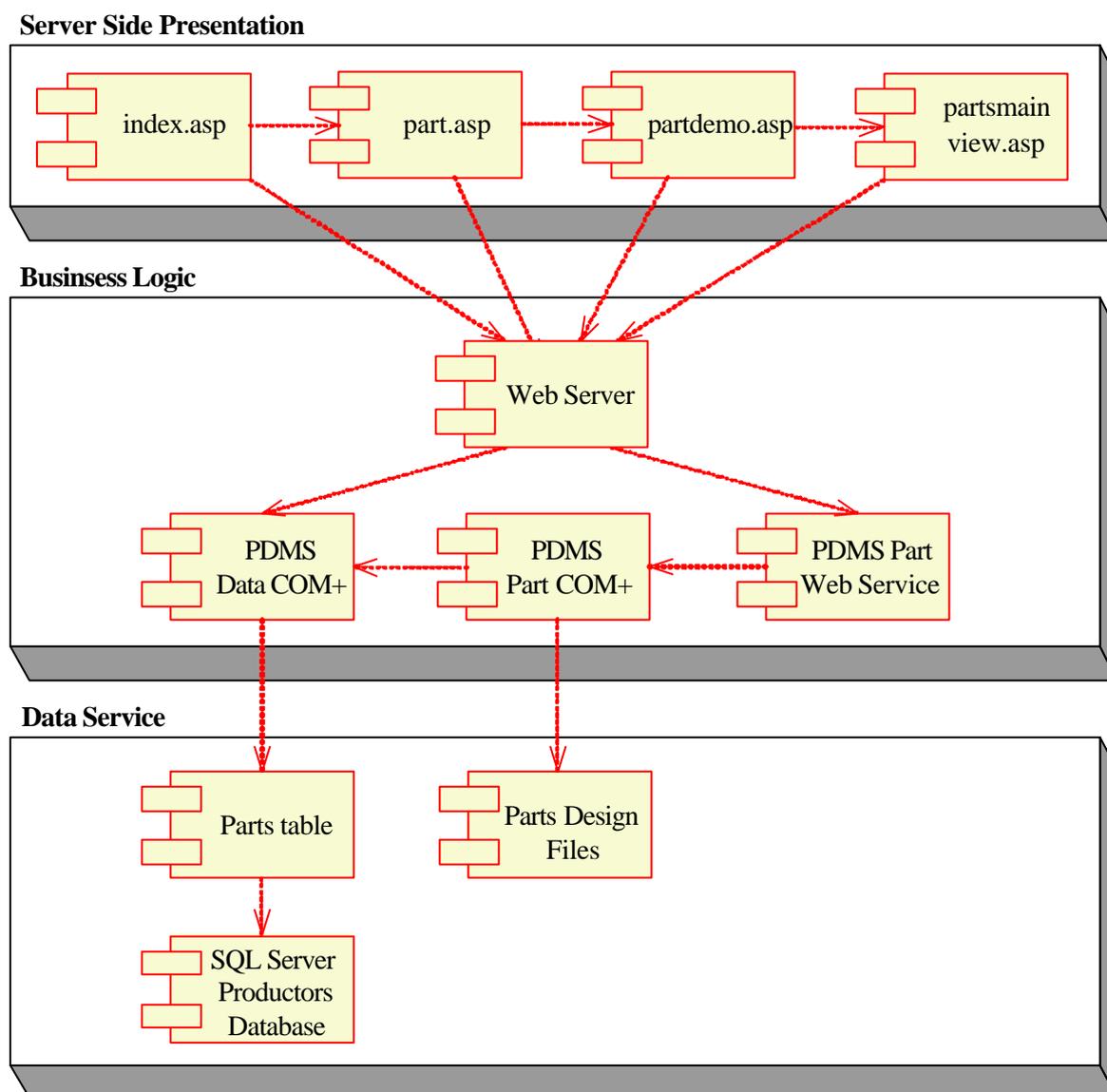


圖 5-9 零件資料取得元件圖

環境模型觀點

環境模型觀點以佈署圖描述整體系統軟硬體裝置與其架構，本系統的佈署圖如圖 5-10 所示：

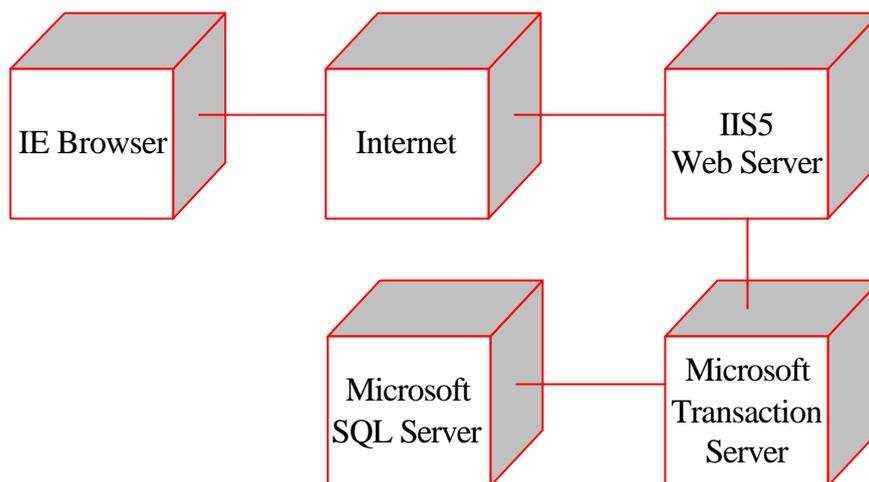


圖 5-10 系統佈署圖

5.4 系統實作

整體系統架構

經系統的分析與設計後，本研究實作四個主要子系統，分別是 1. 網站入口系統。2. 產品資料管理系統。3. 零件資料管理系統。4. 模具資料管理系統。整體網站系統組成架構如圖 5-11 所示，產品資料表關聯請見【附錄 C】，各子系統之功能、操作流程、系統畫面將於 5.5 節說明。

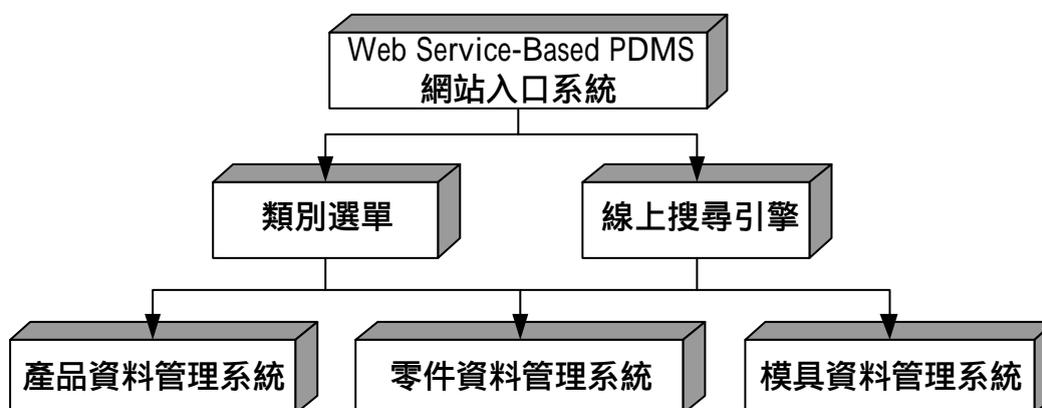


圖 5-11 網站系統架構圖

系統元件與其 Web Service

整個 Web Service-Based PDMS 系統後端所使用到之元件與 Web Service 如表 5-6 所示。所有系統元件註冊於 Microsoft Transaction Server 之中，而 Web Service 則放置於 IIS Web Server 的虛擬目錄中。

表 5-6 Web Service-Based PDMS 之元件與 Web Service

元件	元件類別	元件主要功能	Web Service
DB_Object.DataObject	資料元件	提供程式或其他元件存取後端產品資料庫之功能。	P_WS_DataObject.dll
P_Product_Type.Product_Type	功能元件	提供前端程式或元件查詢與取得產品類別資訊功能。	P_WS_Product_Type.dll
P_Product_Data.Product_Data	功能元件 協調元件	提供程式或其他元件取得產品相關資料、資訊，以及判斷與取得設計檔案。	P_WS_Product_Data.dll
P_mold.mold	功能元件 協調元件	提供程式或其他元件取得模具相關資料、資訊功能。	P_WS_Mold.dll
P_pair.pair	功能元件	提供程式或其他元件取得產品相關配件資料之功能。	P_WS_Pair.dll
P_GUI.GUI	功能元件 協調元件	整合後端程式或元件所處理之結果，並賦予使用者介面回傳前端。	Null

系統開發與使用方法

本系統建構於 Web 架構之上，主要由後端程式語言 ASP，負責處理使用者介面與呼叫 Web Service 或系統元件，系統功能的核心以設計圖檔管理為主，其詳細的開發模式與程式碼如【附錄 D】所示。

整個呼叫與使用 Web Service 介面功能的步驟如下：

- 一、取得 Web Service 的 WSDL 連結網址。WSDL 檔案是以 XML 格式來描述後端 Web Service 的服務介面，因此取得 WSDL 之後即可呼叫與使用 Web Service。

- 二、由於 ASP 程式語言並無提供呼叫 Web Service 的物件,因此可先安裝微軟的 SOAP ToolKit 程式【50】 ,再以 ASP 建立 SOAP Toolkit 中的 SoapClient 物件,並指定 WSDL 位置,即可呼叫 Web Service 之介面功能。
- 三、Web Service 呼叫 PDMS COM+元件功能,COM+元件處理企業邏輯或產品資料後,再將結果回傳給 Web Service。
- 四、ASP 程式直接處理 Web Service 所回傳的 XML 格式訊息,或者再透過 MS XML DOM (本系統使用 3.0 版) 與 ASP 本身的 Recordset 物件,處理 Web Service 回傳的相關產品資料與邏輯。

整個系統使用的方法如圖 5-12 所示,除了 ASP 程式存取系統功能的方法與程序之外,於此系統架構中,其他異質系統主要透過 Web Service 存取系統元件,而同質系統則可直接使用 COM+元件中的系統功能。

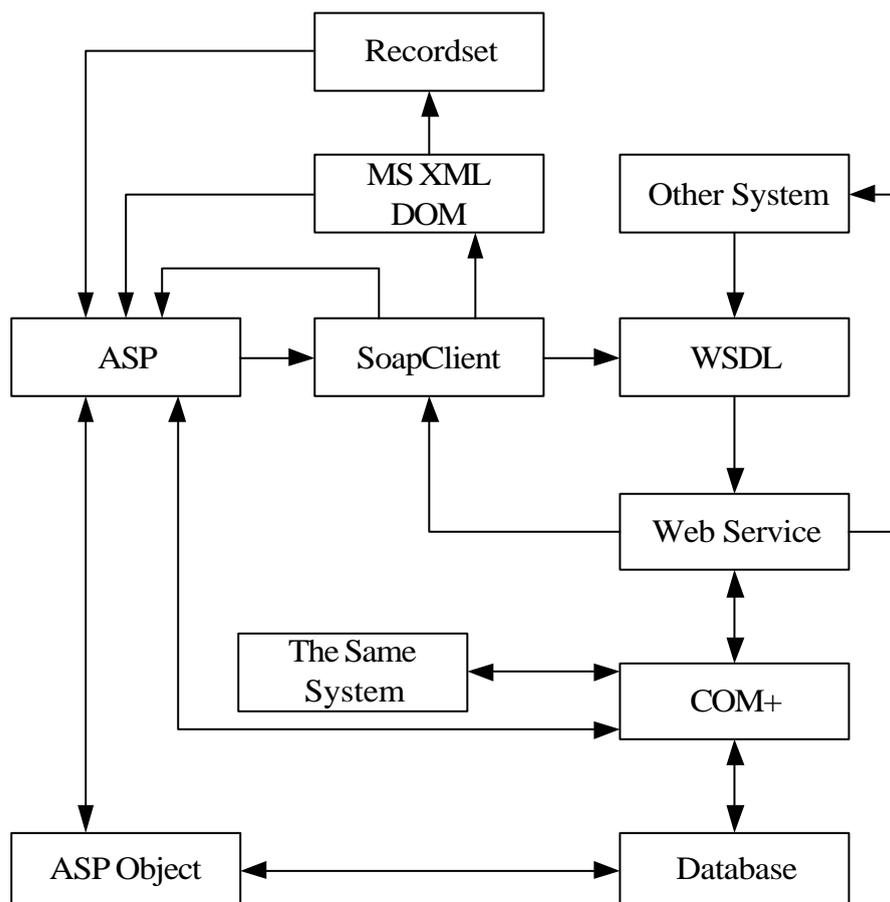


圖 5-12 系統使用方法及程序

以下說明 ASP 程式呼叫 Web Service 與處理回傳資料之程式碼：

```
‘取得 PDMS Web Service 的 WSDL 連結網址，並將其放置於 WSDL_URL 變數之中。  
WSDL_URL = "http://127.0.0.1:8008/flinger/component/WS/P_WS_Product_Data.dll" &  
wsdl/IWS_Product_Data"  
  
‘建立 MS SOAP ToolKit 中的 SoapClient 物件。  
Set PData = Server.CreateObject("MSSOAP.SoapClient30")  
  
‘開始連結遠端 Web Service。  
PData.ClientProperty("ServerHttpRequest") = True  
PData.mssoapinit WSDL_URL  
  
‘Web Service 連結完畢，此時 PData 物件已具備遠端 Web Service 的功能介面，可讓程  
‘式使用功能介面所定義之方法。  
  
‘建立 MS XML DOM 物件：XmlObj  
Set XmlObj = Server.CreateObject("MSXML2.DOMDocument.3.0")  
  
‘呼叫 Web Service 之介面方法：Get_PNO_Data(table_name,pno)，並將回傳之 XML  
‘格式訊息，載入於 XmlObj 物件中。  
XmlObj.async = "false"  
XmlObj.loadXML(PData.Get_PNO_Data(table_name,pno))  
  
‘建立 ASP Recordset 物件：RS。  
Set RS = Server.CreateObject("ADODB.Recordset")  
  
‘利用 Recordset 物件中的 Open 方法，將 XmlObj 物件所儲存的 XML 格式資料載入至  
‘Recordset 物件中。  
RS.Open XmlObj  
  
‘載入成功之後，即可利用 Recordset 物件的方法來處理資料。若不想以 Recordset 物件  
‘處理 Web Service 回傳之資料，則也可透過 XMLDOM 所提供之 XML 資料操作方法  
‘加以處理。
```

5.5 系統展示

如上節圖 5-11 所示，本系統主要由 4 個子系統所構成，茲分述如下：

一、網站入口系統

系統功能：

- 提供前端使用者登入系統之單一入口平台。
- 整合所有產品類別與其子類別，呈現於單一系統畫面，藉由視覺化圖示點選項目，讓使用者可快速取得與連結各產品類別的完整資訊。
- 提供產品、零件、模具搜尋引擎，使用者僅需輸入型號，系統將自動搜尋與處理，最終回應前端資料搜尋結果。

系統流程(如圖 5-13 所示)：

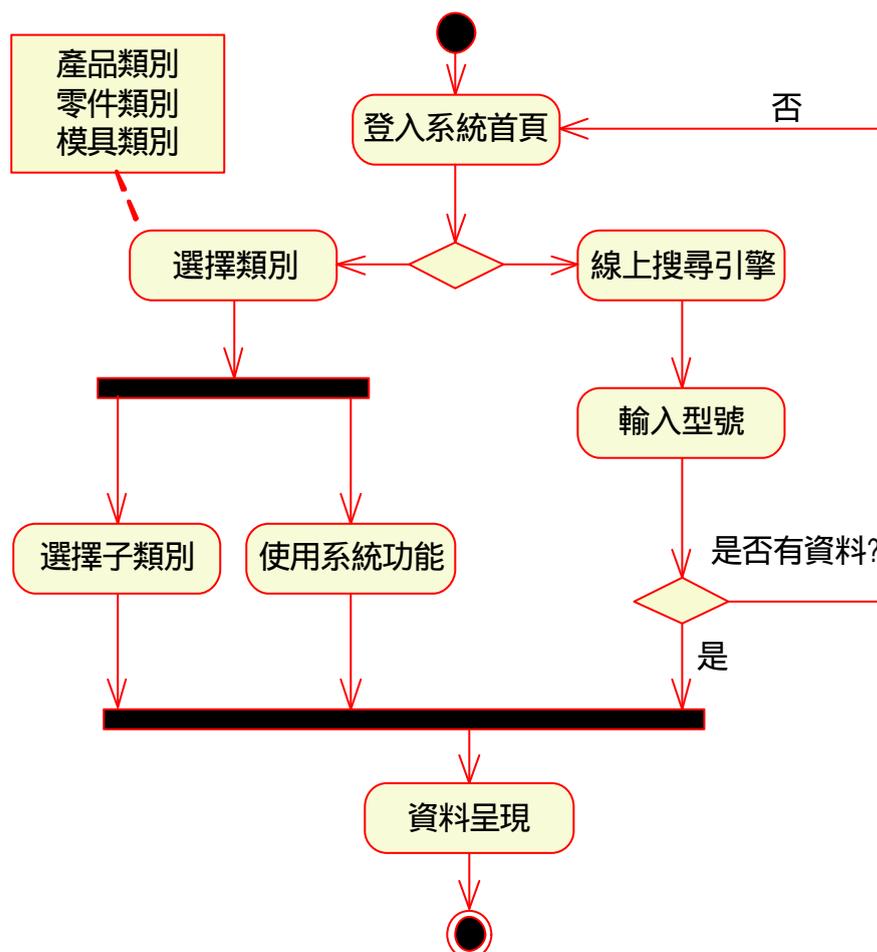


圖 5-13 網站入口系統流程

系統畫面：

如圖 5-14 所示，系統包括產品、零件、模具三大類別。產品類別共有擋泥板、鏈蓋、齒輪蓋、反光板架、兒童輔助輪、兒童安全座椅、前後置物箱，以及其他等八大類別，每一類別可能包含子類別，本系統以圖層的方式顯示(如圖 5-14 中之 A 部分)，可讓使用者於圖層中連結各個子類別頁面。除了類別查詢產品相關資料外，使用者也可利用線上搜尋引擎輸入型號，快速找尋產品、零件，以及模具等資料，如圖 5-14 中之 B 部分。



圖 5-14 網站入口系統畫面

二、產品資料管理系統

系統功能：

- 提供產品型錄分頁瀏覽，可讓使用者快速得知產品外型與基本資料。
- 提供產品型錄資料、設計資料、工程物料清單、設計圖檔等一整合介面，可讓使用者透過瀏覽器取得完整產品資料，或進行資料的維護與管理。

系統流程(如圖 5-15 所示)：

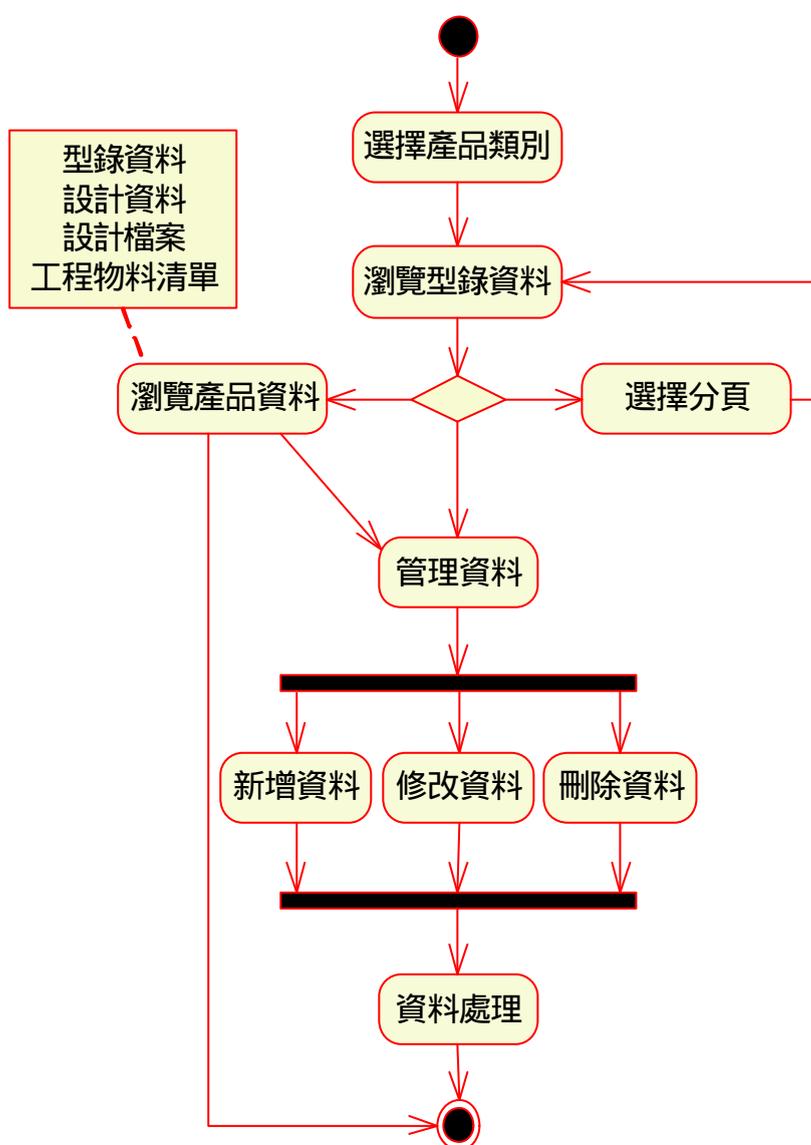


圖 5-15 產品資料管理系統流程

系統畫面：

當點選產品類別後，系統便進入各類別型錄瀏覽畫面，如圖 5-16、5-17 所示。



圖 5-16 輔助輪產品型錄瀏覽畫面

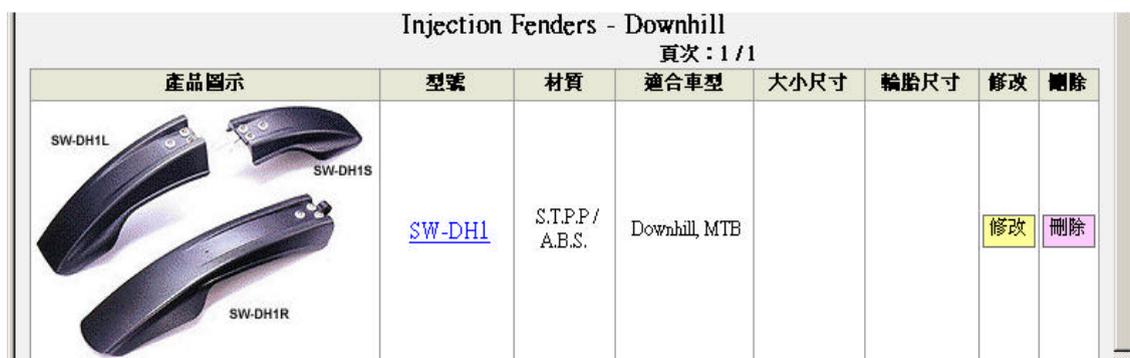


圖 5-17 土除產品型錄瀏覽畫面

點選產品圖形後(例如 NB-1)，即進入產品資料整合展示畫面，如圖 5-18 所示：

產品資料管理系統 - Microsoft Internet Explorer

地址: http://127.0.0.1:8003/lingbo/Data.asp?no=NB-1&table=training_wheel&subclass=1&subclassname=Training%20wheel%2001&page=1

同主產品類別 > 同上一頁 > 您搜尋的結果如下: [管理](#)

型錄資料

兒童輔助輪 - Training wheel 01

型號: NB-1
材質: ABS
Bike size: 12",14",16"童車

可拆式輪座,具強化支架,適合12",14",16"童車使用

Tire:

設計資料

主要負責人: [施依婷](#) 開發完成時間: 2002/10/24

客戶: [黃寶](#)

設計概念或需求: [造型新穎且避震效果要好](#)

發案原因: 無

檔案連結: [詳圖](#) [簡圖](#) [組立圖](#) [總圖](#) [流程圖](#) [組立流程圖](#) [NC檔](#) [客戶提供草稿](#) [2D圖](#) [3D圖](#)

設計圖展示: [2D平面圖](#) [流程圖](#) [展示圖](#)

研發部產品資料提供

No	型號	名稱	材質	單位	標準 用量	供應商	規格	單價	加工廠	處理	單價	加工廠	處理	單價	備註
1	NB101DL-0	下取附(左)	鐵	PCS	1	新瑞	11	1.0	平治	ED	1.0				重214.2g
2	NB101DR-0	下取附(右)	鐵	PCS	1	新瑞	11	1.0	平治	ED	1.0				重214.2g
3	NB101U-0	上取附保護套	Polycarbonate	PCS	2	新瑞	08								重3.0g
4	NB101UL-0	上取附(左)	鐵	PCS	1	新瑞	11	1.0	平治	ED	1.0				重109.2g
5	NB101UR-0	上取附(右)	鐵	PCS	1	新瑞	11	1.0	平治	ED	1.0				重109.2g
6	NB102LD-0	平行連接桿(左下)	鋼	PCS	1	新瑞	黑色	1	聯勝	聯合					重34.64g
7	NB102LU-0	平行連接桿(左上)	鋼	PCS	1	新瑞	黑色	1	聯勝	聯合					重34.64g
8	NB102RD-0	平行連接桿(右下)	鋼	PCS	1	新瑞	黑色	1	聯勝	聯合					重34.64g
9	NB102RU-0	平行連接桿(右上)	鋼	PCS	1	新瑞	黑色	1	聯勝	聯合					重34.64g
10	NB103-0	兩重彈簧	鋼	PCS	4	新瑞	1.2								重1.6g
11	NB104-0	橡膠彈簧塊	橡膠	PCS	2	新瑞	紅色	1.0							紅61.4g
12	NB105-0	輪胎	膠	PCS	2	新瑞	1.5	1.0	聯勝						64.8g
13	NB106-0	輔助輪(左)	鐵	PCS	2	新瑞	1.5	1.0	聯勝						64.8g
14	NB107-0	輔助輪(右)	鐵	PCS	2	新瑞	1.5	1.0	聯勝						64.8g
15	NB108-0	固定夾	鋼	PCS	2	新瑞	白色	1.0							球內有(重13.2g)
16	NB109-1	上取附螺絲	鐵	PCS	4	新瑞	3/8	1.0	平治	ED	1.0				
17	SCREWS-1	輔助輪螺絲帽蓋	鐵	PCS	6	新瑞	黑色	0.5							球內有(重0.6g)

圖 5-18 產品資料整合展示畫面

圖 5-18 中 A、B、C、D 分別為：

A：型錄資料

使用者可於此查詢產品的基本資料，包括產品材質、適用車型、產品簡介，以及產品配件等。當使用者點選產品配件後，系統將開另一視窗來顯示配件資料，如圖 5-19 所示：



圖 5-19 瀏覽配件資料畫面

B：設計資料

使用者可於此查詢設計師、產品完成日期、客戶名稱、設計概念、設計需求，以及設計變更原因等。

C：設計圖檔資料

本系統共將圖檔資料分為 14 種類別，包括產品簡圖、組立圖、2D、3D 圖形，以及客戶所提供之草稿、2D、3D 圖檔等，只要產品圖形檔案存放於固定目錄資料夾內，系統即將整理與顯示，提供使用者下載開啟，或者透過 Autodesk Express Viewer 直接線上瀏覽。

D：工程物料清單

列出此產品所有零件、零件基本資料、標準用量、供應商、價格、加工廠、加工價格等，使用者透過此清單可了解產品與其零件的相關資訊，若直接點選零件型號，系

統將開出一新視窗顯示零件資料，如圖 5-20 所示：

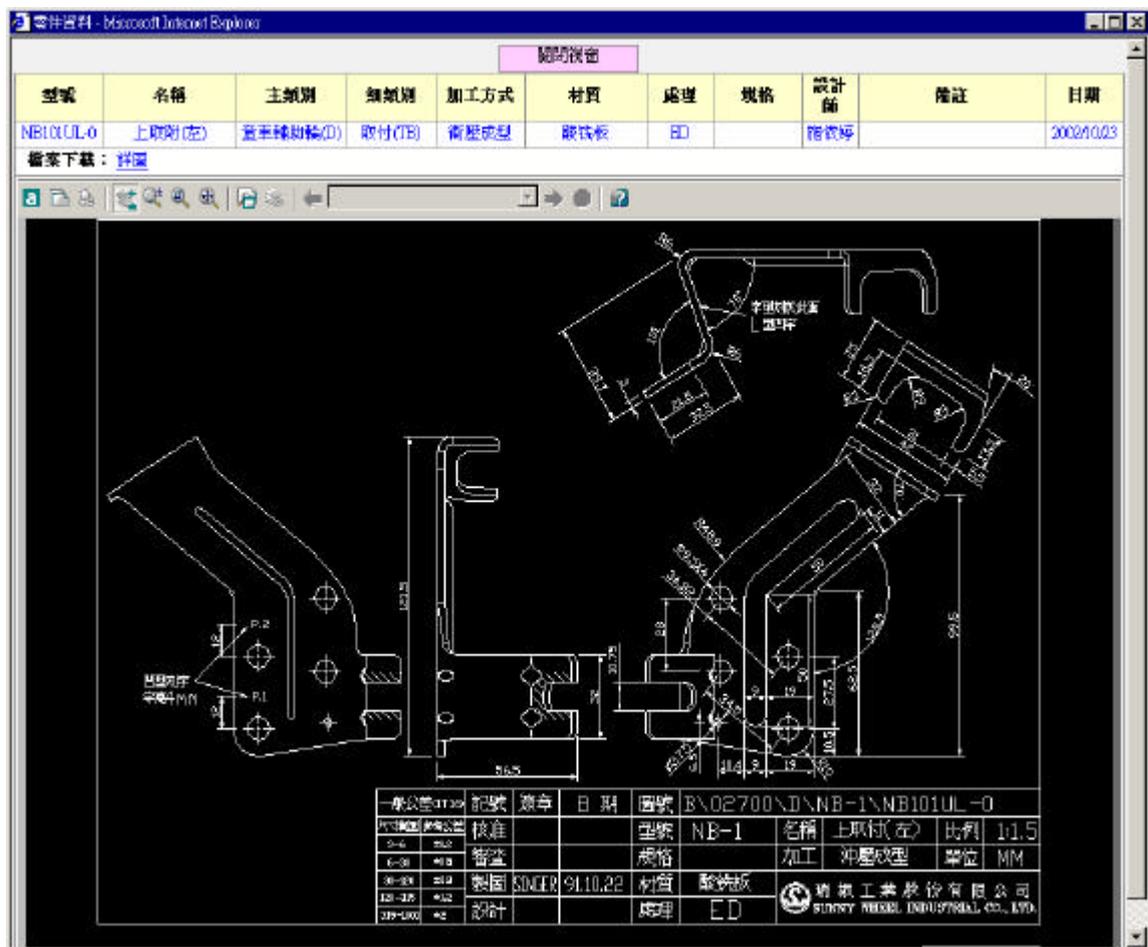


圖 5-20 零件資料線上瀏覽畫面

若於產品型錄瀏覽畫面點選新增產品資料，則會進入產品資料新增畫面，如圖 5-21 所示。當使用者輸入產品型號後，系統將自動命名所有圖檔的名稱(如圖 5-21 中之 A 所示)，使用者必須使用檔案上傳的功能上傳圖檔，而在上傳圖檔之前，系統將自動檢查圖檔名稱是否符合命名規則，若不符合則系統將不允許新增產品資料。若此產品資料有相關配件，則其可點選「新增配件」按鈕，系統將出現此類別的所有配件清單視窗(如圖 5-21 中之 B)，讓使用者以點選的方式輸入配件資料。若要談寫產品工程物料清單，則可先「選擇零件類別」，再「新增零件資料」，系統將列出符合類別條件的所有零件清單(如圖 5-21 中之 C)，讓使用者可以點選的方式輸入。當所有資料正確輸入完畢並送出後，系統將依內定之規則，系統化自動處理所有產品資料與設計圖檔。

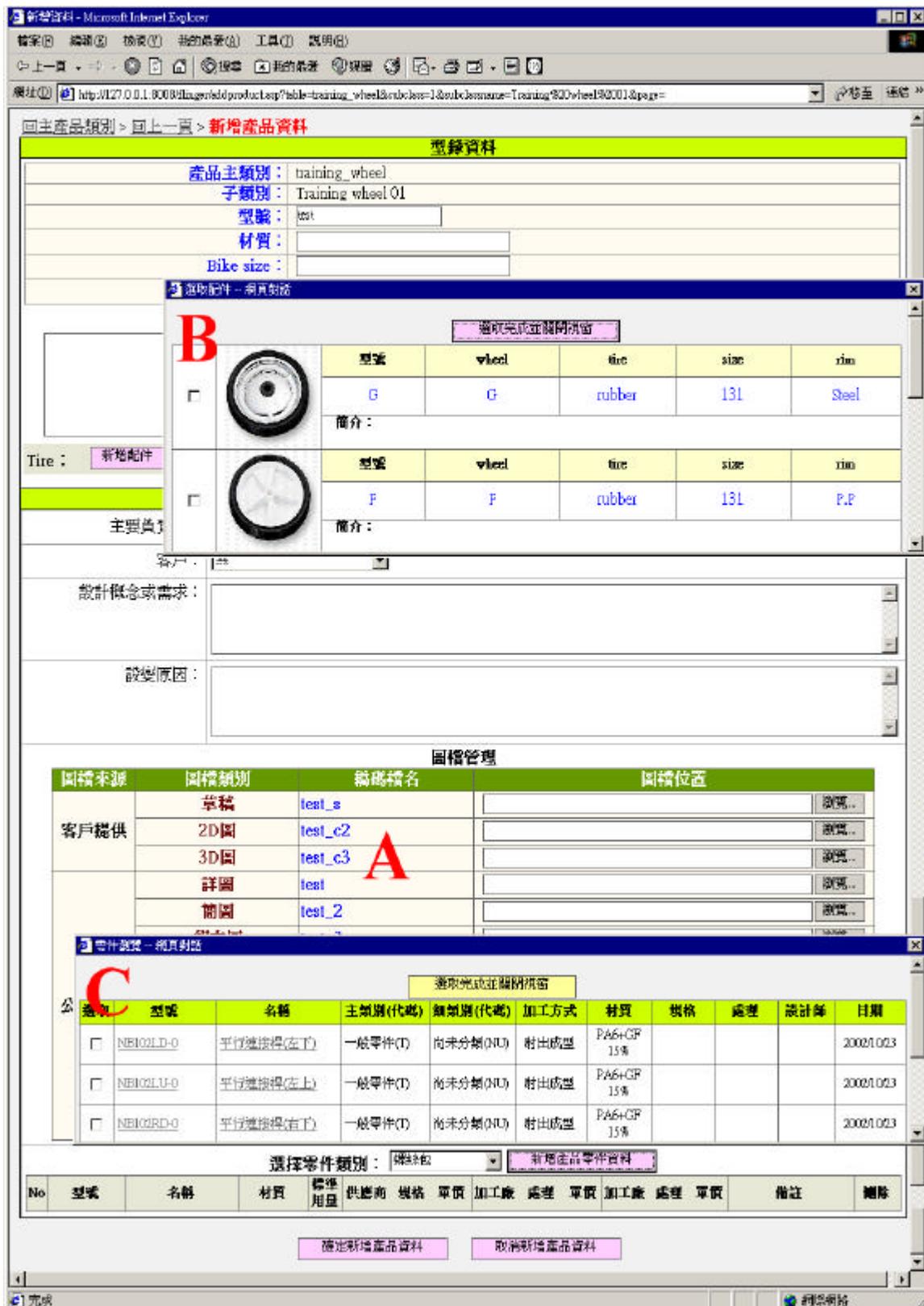


圖 5-21 產品資料新增畫面

產品資料的修改、刪除等管理功能畫面與圖 5-18 相似，不同之處在於將所有單純產品資料顯示，改為以表單欄位的方式顯示，讓使用者可修改或刪除表單欄位內的資料，當點選「儲存所有變更」或「刪除所有資料」按鈕後，系統將處理前端所傳送之資料與管理需求。

三、零件資料管理系統

系統功能：

- 提供以類別名稱與主從類別兩種方式找尋零件類別。
- 零件基本資料分頁顯示。
- 零件資料線上瀏覽與管理。

系統流程(如圖 5-22 所示)：

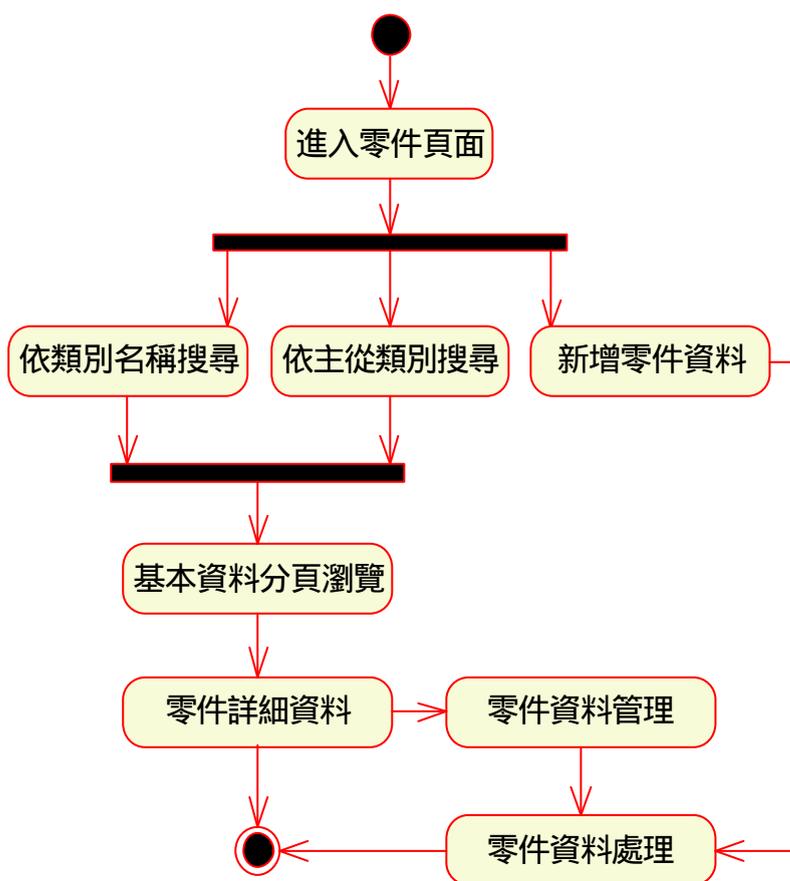


圖 5-22 零件資料系統流程

系統畫面：

零件類別搜尋方式有二，如圖 5-23、5-24 所示。採用主從類別搜尋可讓使用者精確地找出符合條件的零件類別，而採用類別名稱搜尋方式則可讓使用者取得較為廣泛的零件參考資料，而不限定其零件資料屬於哪一零件類別。

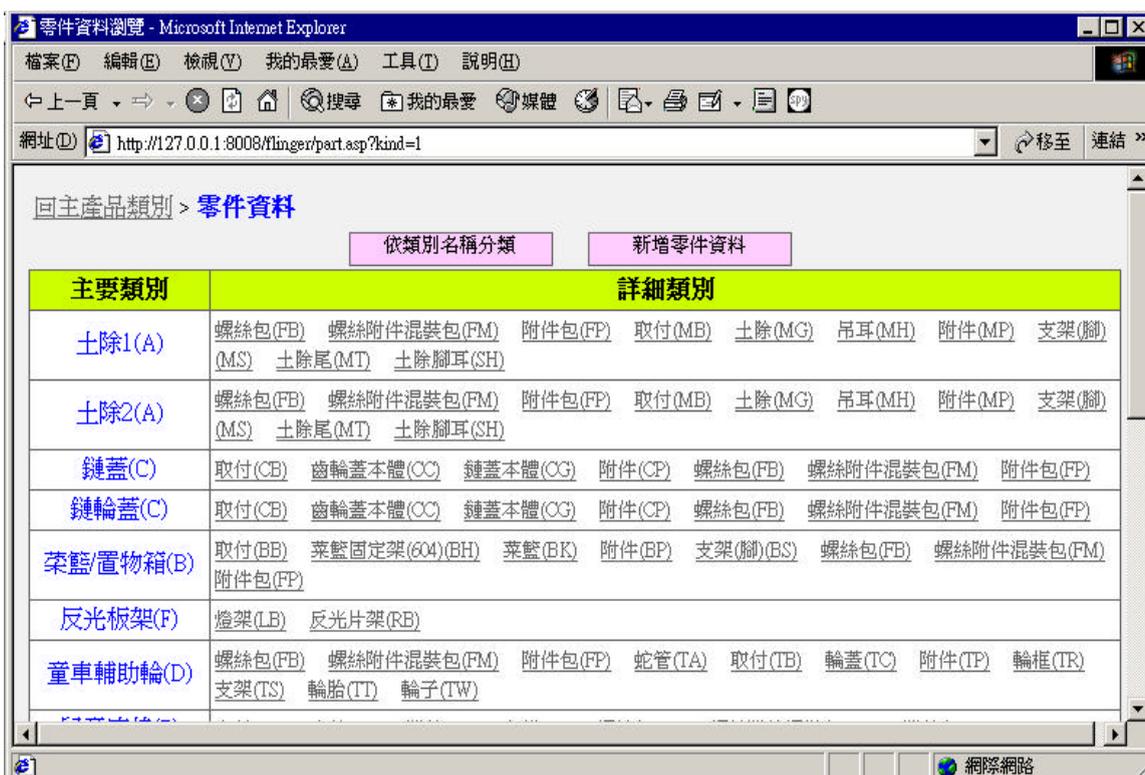


圖 5-23 零件資料主從類別搜尋

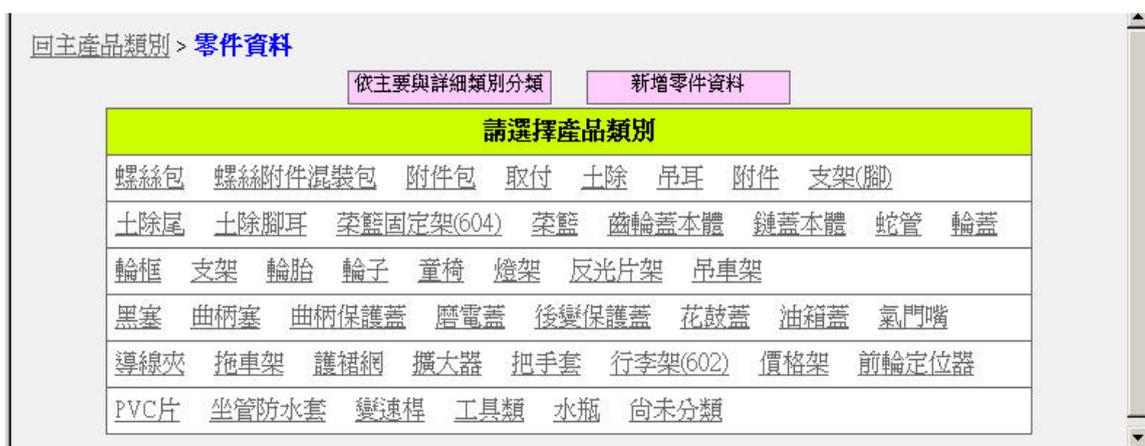


圖 5-24 零件資料類別名稱搜尋

自圖 5-23、5-24 點選零件類別之後，系統即進入零件基本資料分頁瀏覽畫面，如圖 5-25 所示。使用者可於此畫面快速了解零件的特徵及基本資料，並可進行資料的修改與刪除等管理工作。

圖 5-25 顯示了零件基本資料分頁瀏覽的畫面。畫面中有一個表格，列出了零件的詳細信息。表格的表頭如下：

型號	名稱	主類別(代碼)	細類別(代碼)	加工方式	材質	規格	處理	設計師	日期	修改	刪除
NB102LD-0	平行連接桿(左下)	一般零件(T)	尚未分類(NU)	機加工	Al6061				2002/10/23	修改	刪除
NB102LU-0	平行連接桿(左上)	一般零件(T)	尚未分類(NU)	機加工	Al6061				2002/10/23	修改	刪除
NB102RD-0	平行連接桿(右下)	一般零件(T)	尚未分類(NU)	機加工	Al6061				2002/10/23	修改	刪除
NB102RU-0	平行連接桿(右上)	一般零件(T)	尚未分類(NU)	機加工	Al6061				2002/10/23	修改	刪除
NB103-0	葫蘆型彈簧	一般零件(T)	尚未分類(NU)	熱處理	彈簧鋼	φ2	鍍黑		2002/10/23	修改	刪除
NB104-0	橡膠彈簧塊	一般零件(T)	尚未分類(NU)	機加工	橡膠				2002/10/23	修改	刪除
NB108-0	固定夾	一般零件(T)	尚未分類(NU)	機加工	PC				2002/10/23	修改	刪除

圖 5-25 零件基本資料分頁瀏覽

若使用者點選圖 5-25 中的零件名稱連結，則即可下載此零件的 2D 圖檔(於本系統中為 Autocad 檔)。若使用者點選零件型號，則系統將進入顯示零件詳細資料畫面，此畫面之介面與資料與圖 5-20 相同，可讓使用者取得此零件的所有資料，包括可以下載零件設計圖檔以及線上瀏覽零件 2D 圖。

圖 5-26 為零件資料的管理畫面，系統設計的原則是盡量採用自動化處理與選取的方式來達到資料管理的功能。如圖 5-26 中的 A 部分，系統將類別編碼方式、加工方式、設計師等資料，設計為下拉式選單，一方面加快處理效率，另一方面則避免人為輸入的錯誤。圖 5-26 中的 B 部分與產品圖檔自動命名功能相同，提供零件圖檔自動命名，俟資料均正確輸入且送出之後，系統也將依既定的邏輯法則，自動處理零件資料以及歸納設計圖檔。零件新增的畫面與圖 5-26 的管理畫面相似，差異只在新增零件畫面上的所有表單欄位資料均為空白。

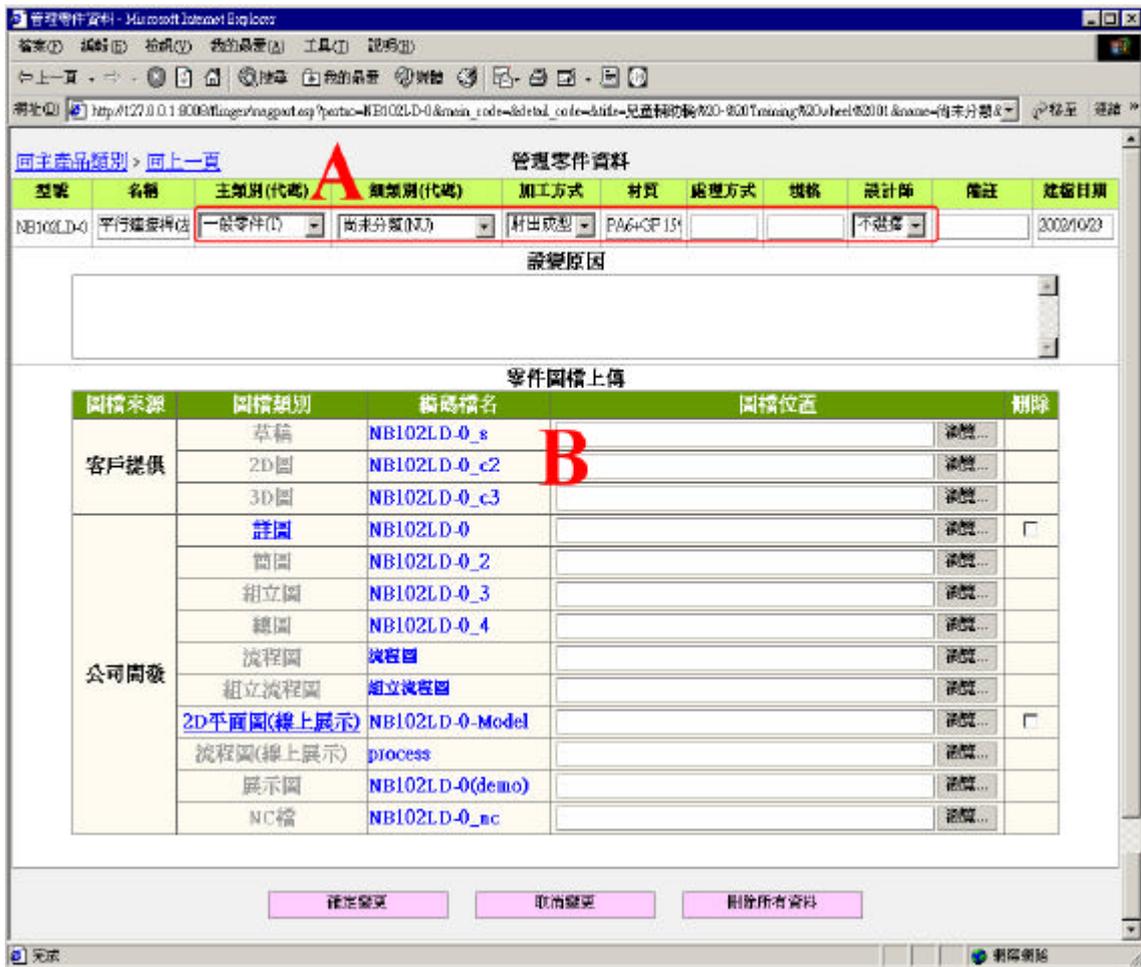


圖 5-26 零件資料管理畫面

四、模具資料管理系統

系統功能：

- 模具廠商資料管理：可管理模具廠商資料，並提供分年表列模具資料等功能。
- 模具資料管理：提供線上查詢與管理模具資料。
- 廠商模具資料搜尋引擎：讓使用者可選擇模具廠商以及查詢年份，而後搜尋符合條件的模具資料。

系統流程(如圖 5-27 所示)：

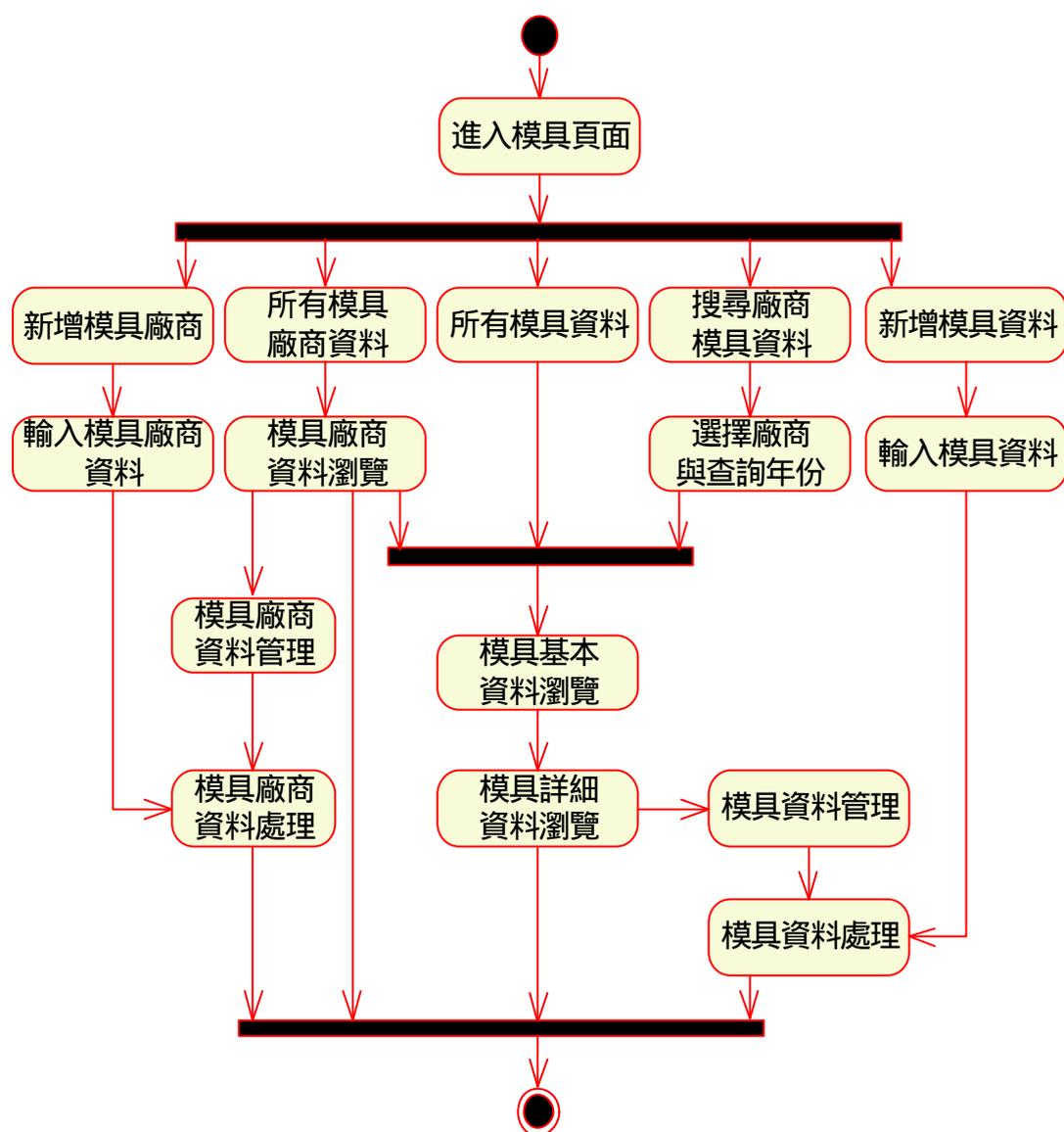


圖 5-27 模具資料管理系統流程

系統畫面：

當使用者點選「所有模具廠商資料」，系統即出現圖 5-28 之畫面，可查詢與管理模具廠商資料。當使用者點選「所有模具資料」，系統即出現圖 5-29 之畫面，列出所有模具資料，並可以依年份排序。圖 5-29 中之 A 部分即為廠商模具資料搜尋引擎，可讓使用者快速找尋特定廠商的所有模具資料。

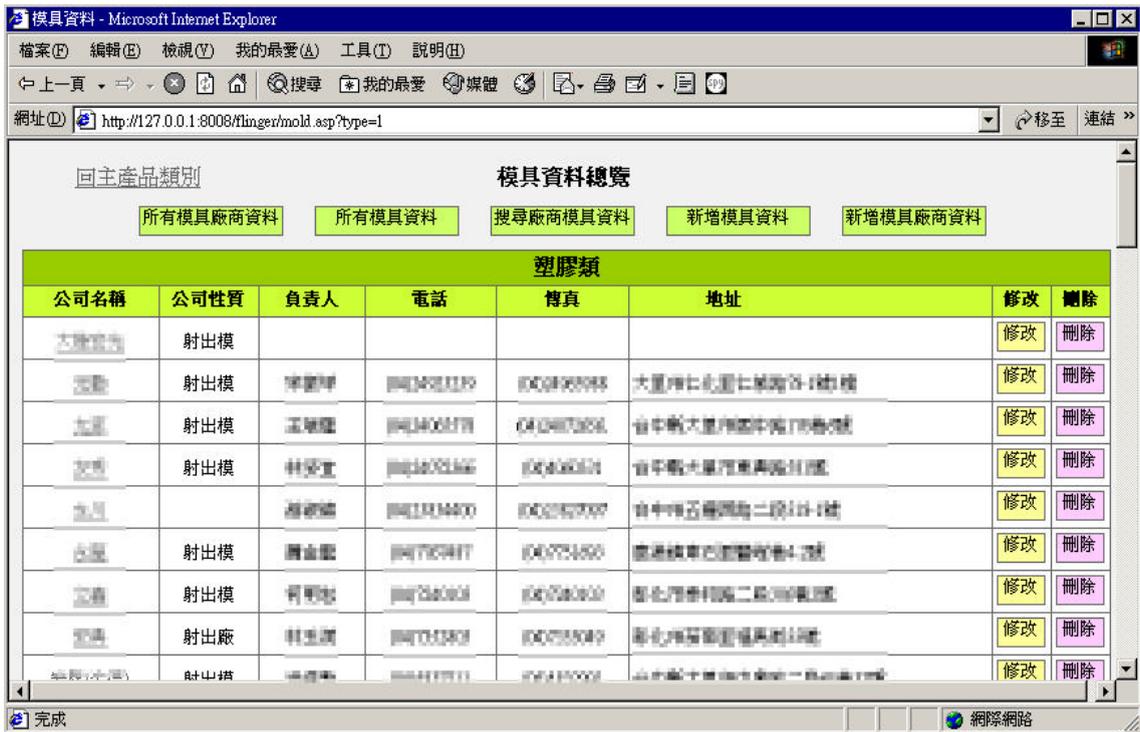


圖 5-28 模具廠商資料管理畫面

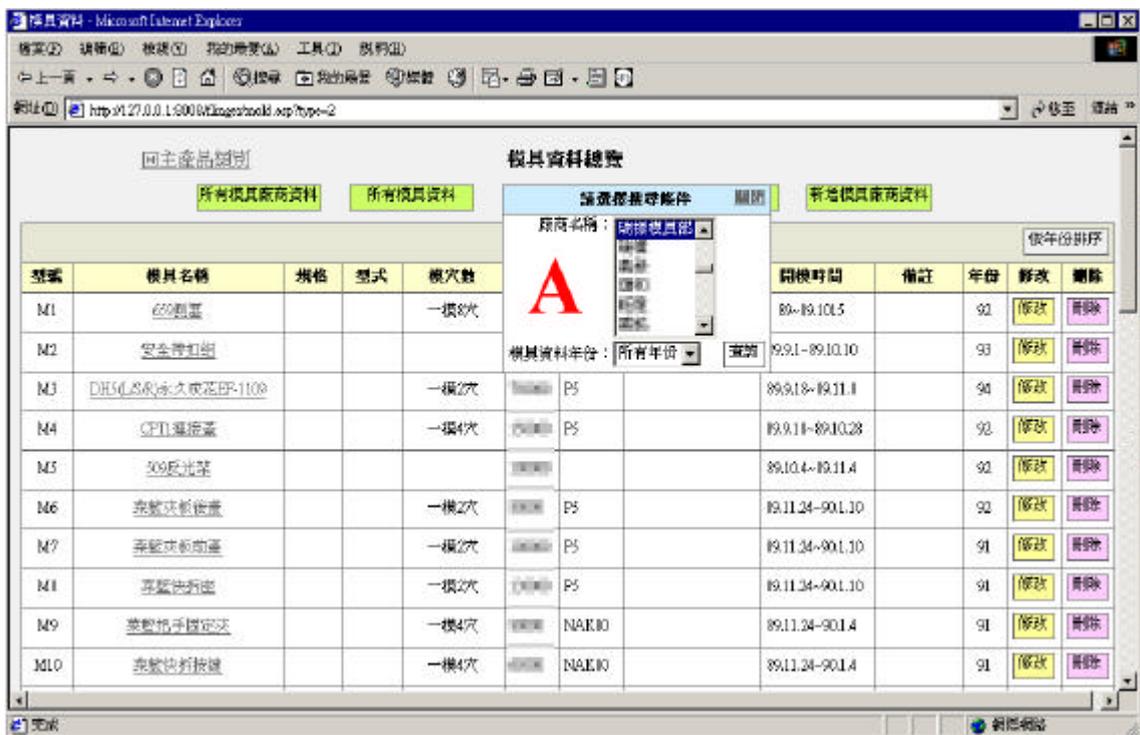


圖 5-29 模具資料管理系統畫面

當使用者點選圖 5-29 中的模具名稱，系統即進入詳細模具資料畫面，如圖 5-30 所示。使用者可於此介面取得所有模具資料，包括模具圖檔、NC 檔、線上 2D 圖形瀏覽，以及查詢此模具適合那些產品或零件使用。而當使用者點選「管理」按鈕，系統即進入模具資料的管理畫面，畫面類似於圖 5-30，差異僅在將模具資料的呈現，轉換為表單欄位呈現，並且盡量讓使用者採用點選的方式維護資料，資料送出之後，系統將啟動交易管理機制，自動處理所有資料與圖形檔案管理等。至於模具資料、模具廠商資料的新增也類似於產品、零件資料的新增方式，均以盡量降低人為的輸入與操作、利用程式達到自動化處理為目的，確保所有資料格式與管理方式符合既定標準，並降低出現錯誤的風險。

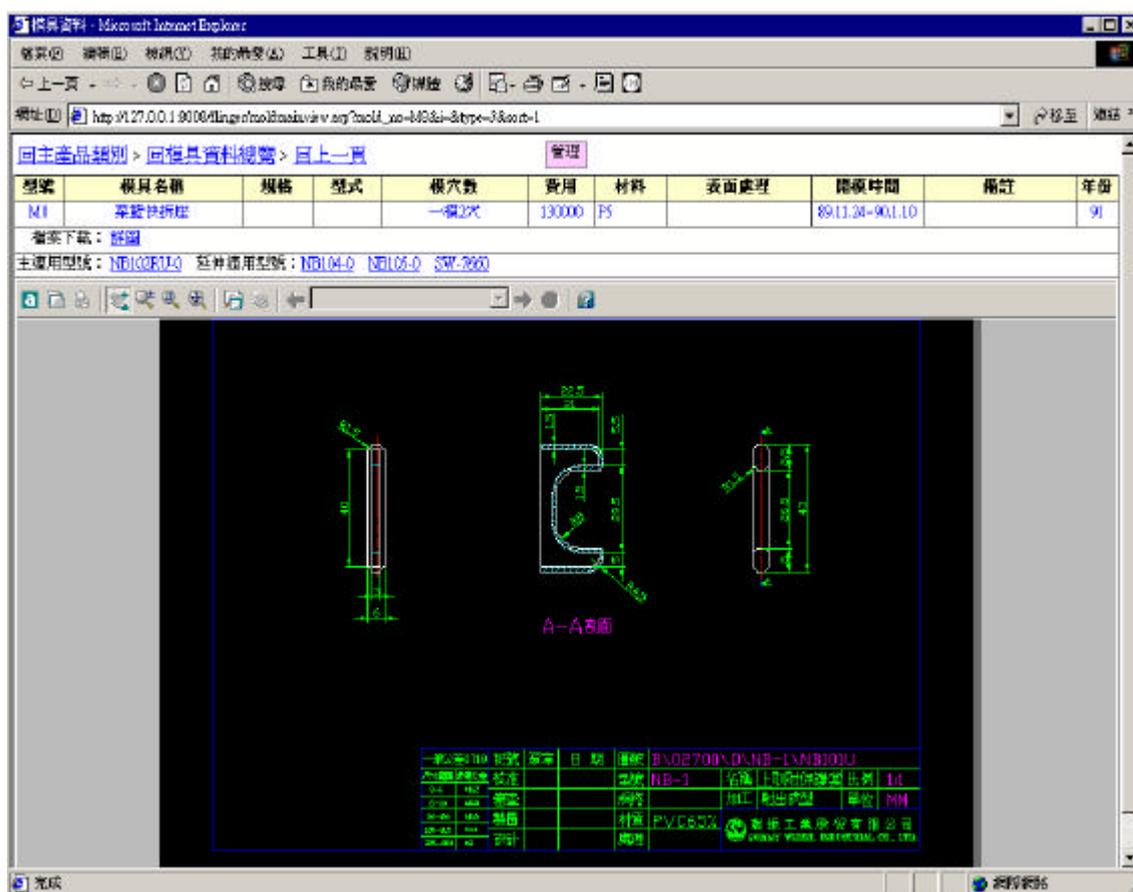


圖 5-30 詳細模具資料畫面

5.6 系統開發效益探討

本研究所提之 Web Service-Based PDMS 於系統功能開發上，實際訪談企業了解現況問題與使用者需求；在系統整體架構上，主要採用元件技術、Web Service、分散式系統架構，以及網路應用程式開發建置。對於企業的主要效益可從這兩方面予以探討：

一、系統功能面主要效益

- 大幅降低圖檔管理工作的複雜度與人力時間成本浪費。

以往開發人員對於圖檔的查詢、新增、修改、刪除等動作，必須耗費大量的時間處理，整個流程大致如圖 5-31 所示：

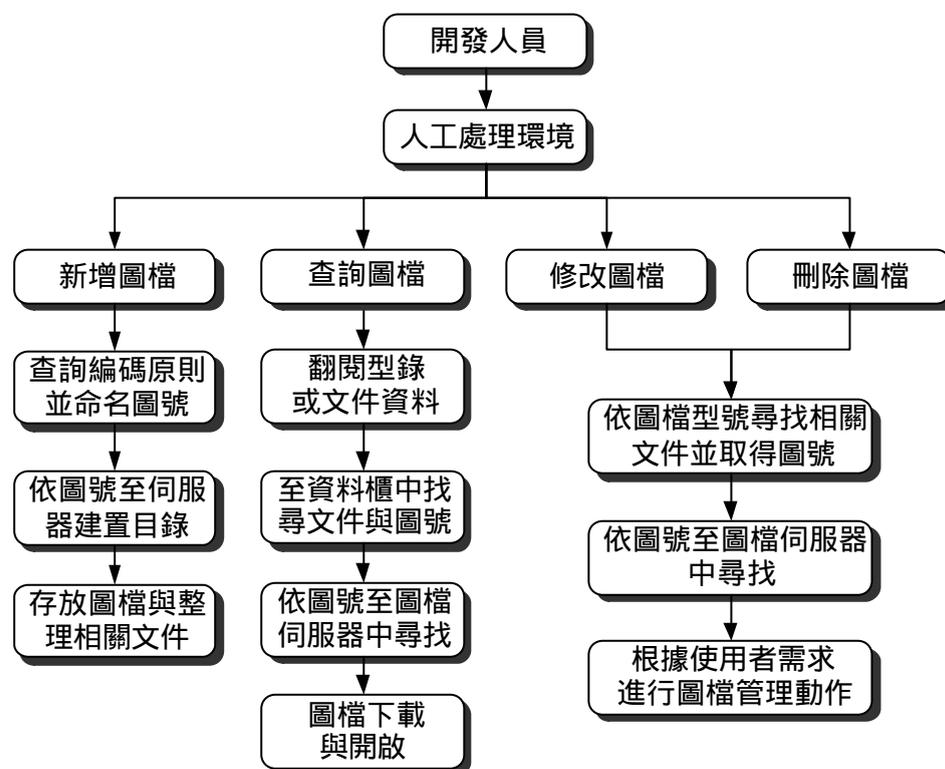


圖 5-31 圖檔傳統管理方式

由圖 5-31 可知，傳統人工處理圖檔管理的程序步驟相當繁雜，開發人員必須經常前至資料櫃中找尋相關文件，再從紙張文件中得知圖檔圖號，然後利用網路芳鄰搜尋或者告知相關人員至圖檔伺服器中找尋圖檔，而後進行圖檔的使用與管理等動作。本系統圖檔管理方式如圖 5-32 所示，系統將複雜的編碼原則與圖號存放路徑【如附錄 B 所示】，均採用系統化的管理方式處理，利用標準的操作介面與流程，配合後端系統

的控管，讓開發人員得以簡易地使用設計圖檔，甚至透過瀏覽器線上審閱瀏覽。由於系統將圖檔的管理流程予以簡化、自動化，因此不僅可降低開發人員的工作負擔，更能避免檔案重覆建置、儲存路徑混亂等問題發生，進而有效提升圖檔管理效率。

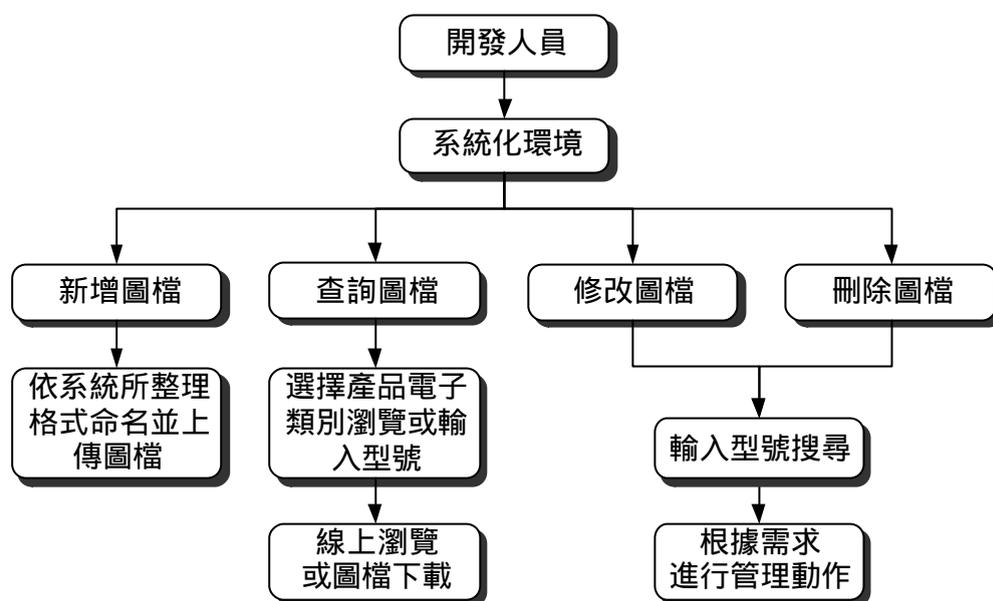


圖 5-32 系統圖檔管理方式

- 改良產品資料使用方式，減少資料收集與整合的人力資源浪費。

圖 5-33 為常見的傳統產品資料使用方式，開發人員藉由型錄與開發文件，先找尋符合搜尋條件的產品，再利用區域網路或者由人員至各部門找尋相關資料使用。

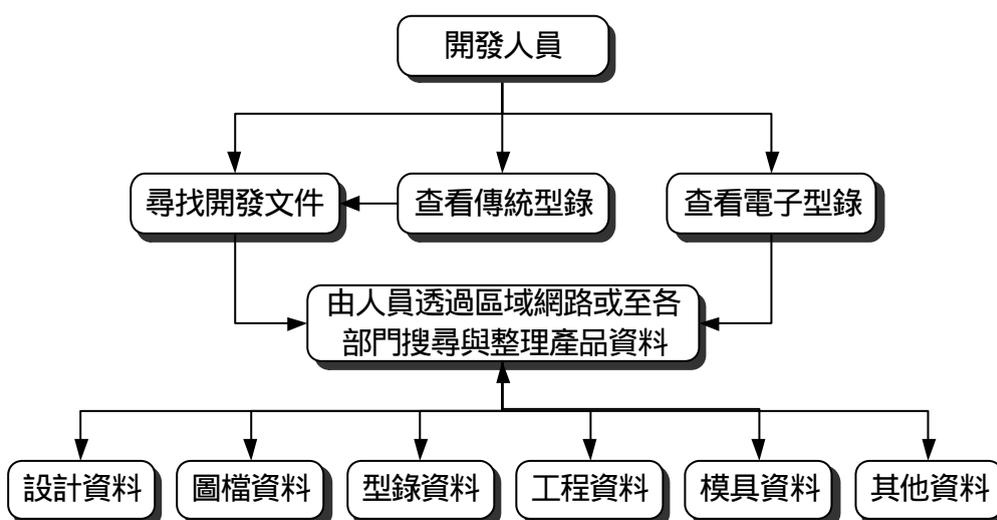


圖 5-33 產品資料傳統使用方式

傳統產品資料使用方式主要透過大量人力找尋與維護資料，而系統化產品資料使用方式則大量利用程式系統，達到資料自動化整合與處理的目標。如圖 5-34 所示，開發人員可透過系統化的產品型錄或者搜尋引擎，快速參考並取得符合條件的完整產品資料，例如本系統之產品資料管理系統，即利用單一介面整合型錄資料、設計資料、配件資料、設計圖檔，以及工程物料清單，讓使用者可於同一畫面中，取得所有產品相關資料，進而節省大量人工處理的時間與成本。

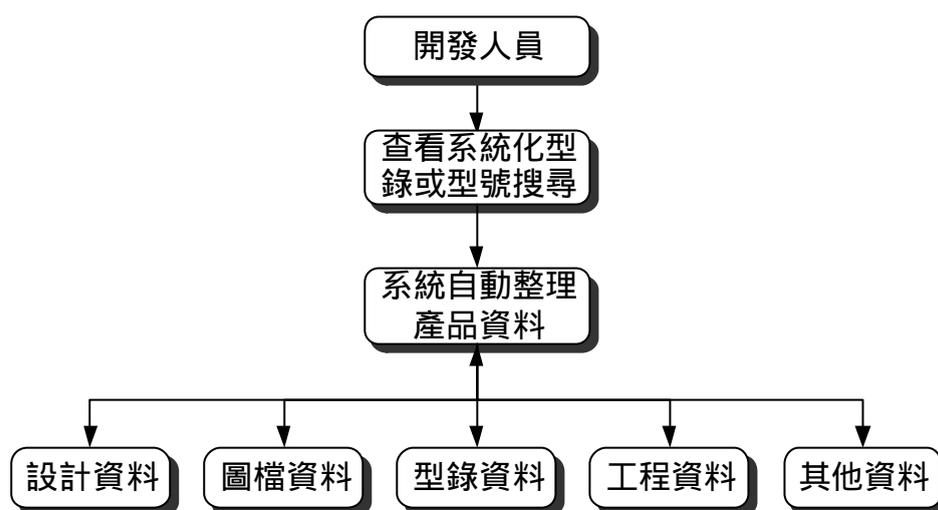


圖 5-34 系統化產品資料使用方式

- 網路式操作介面與系統自動化處理機制。

開發人員可於世界各地不限時間、空間的限制，在完善的安全防護體制下，利用瀏覽器透過 Internet 使用產品開發相關資料，此對於跨國企業或海外部門而言，可降低資訊傳遞交換的隔閡及成本，若進一步擴展與規劃系統安全防護，則可提供客戶、上下游廠商查詢、交換企業產品資料之介面，進而形成一價值鏈體系產品資訊聯繫網絡，共同提昇產品開發效率與企業開發資訊的交流。

採用網路式前端介面的另一優點，即無須特別針對前端程式進行維護，若有特殊軟體物件需要安裝，也僅需透過伺服器提供下載更新服務即可。因此，系統的功能與前端使用者介面，完全操控於企業本身，換言之，企業可不斷擴充後端系統功能，利用登入權限與操作介面控制，提供使用者更多元、更廣泛的產品資訊服務。例如圖 5-34 所示，最底層為產品資料庫，理論上所有產品相關資料均於此層級集合彙整，進而提

供上層系統功能開發使用。若將所有產品相關資料與流程整合於單一系統管理介面，則系統即可進一步發展為產品資料管理系統；若將產品資料結合線上銷售服務系統，則系統可再擴展為電子商務平台，提供網路商業交易，如圖 5-35 所示。

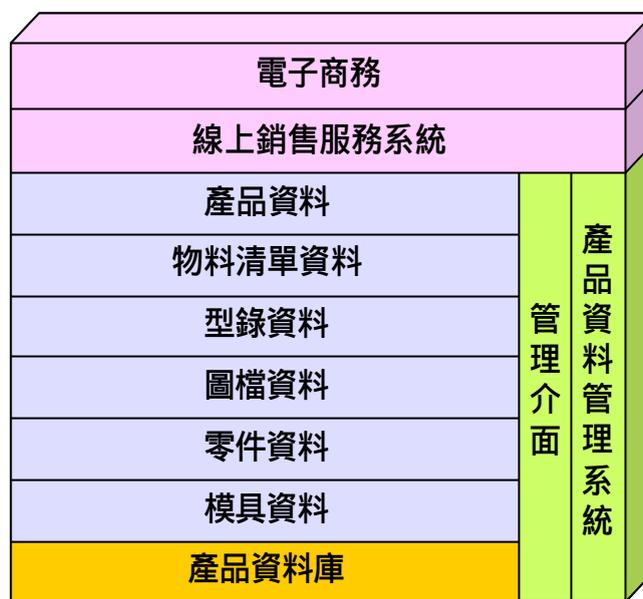


圖 5-35 網路式系統功能介面擴充

除網路式操作介面外，系統自動化處理機制亦為系統重要之功能。透過適時適當的輔助資訊與處理程序，可降低使用者操作之重複性與複雜度，例如本系統提供圖檔原則自動編碼、圖檔命名格式判斷、存放路徑系統化處理、輸入資料選擇、資料比較搜尋與整理等機制，均能使整體操作環境更為簡易且人性化。

二、系統架構面主要效益

- 元件模型可降低維護成本與難度，分散式系統則可確保系統穩定性與執行效能。

利用元件模型跨語言、功能獨立、介面標準等特性，將企業邏輯、系統功能封裝於元件之中，進而達到軟體功能再用之目的，可避免系統功能重複開發，降低系統建置成本以及後續維護管理難度。以本系統為例，後端元件模型是利用 Delphi 的 Object Pascal 程式語言開發，利用元件標準呼叫介面，集合各獨立元件組合成為一系統功能，用以服務前端程式需求。若將來前端需求改變，需變更後端元件功能邏輯，則可重新組合獨立元件，甚至利用其他程式語言，例如 VB、VC、C++ 等重新開發元件功能，而非一定得使用原先的 Delphi 與 Object Pascal。因此採用元件模型技術，將使系統

更具彈性與未來擴充能力。分散式系統的容錯與平衡負載機制，可讓應用程式伺服器具備高度可靠性與穩定性。配合元件模型技術，可將元件分散於各個應用程式伺服器中，進而強化整體系統執行效能，用以克服日益複雜、使用頻繁的網路商業環境需求。

- 以 Web Service 封裝元件模型之功能介面，可使元件功能具備整合性與通透性。

利用 Web Service 相關技術標準封裝元件模型之功能介面，可避免不同元件模型(例如：COM+、EJB、CORBA)無法整合溝通之問題。由於 Web Service 已成為各界通用之標準，加上其使用簡易、成本低廉，以及採用現行標準協定之優點，使企業得以將其取代昂貴的橋接整合技術，進行企業內外資訊系統整合工作。採用 Web Service 包裝系統元件，除了可克服上述異質元件系統整合問題之外，更能將元件功能提供前端各式裝置、作業平台、程式語言使用，設計完善的 Web Service 與系統元件，甚至可達到完全自動化處理，大幅減少人為操作流程與處理成本。由於絕大多數的程式語言、開發工具、作業系統均可建置 Web Service，因此後續系統功能擴充將較為簡易，維護管理成本也相對降低。

將 PDMS 的主要核心以元件模型與 Web Service 標準技術開發，可使 PDMS 系統功能具備高度整合性與通透性。以本系統狀況為例(如圖 5-36 所示)，瑞振企業目前所使用的科展系統主要管理製造、採購、銷售部門之資料與流程，而本系統則以設計研發部門為主。由於科展系統內部資料結構與資訊流程尚未釐清，因此目前無法進行整合工作，然而未來兩套系統將可透過元件模型與 Web Service 相互整合運用，使產品開發從設計研發乃至後續製造銷售均完整串聯，進而減少資訊隔閡與開發資源浪費。

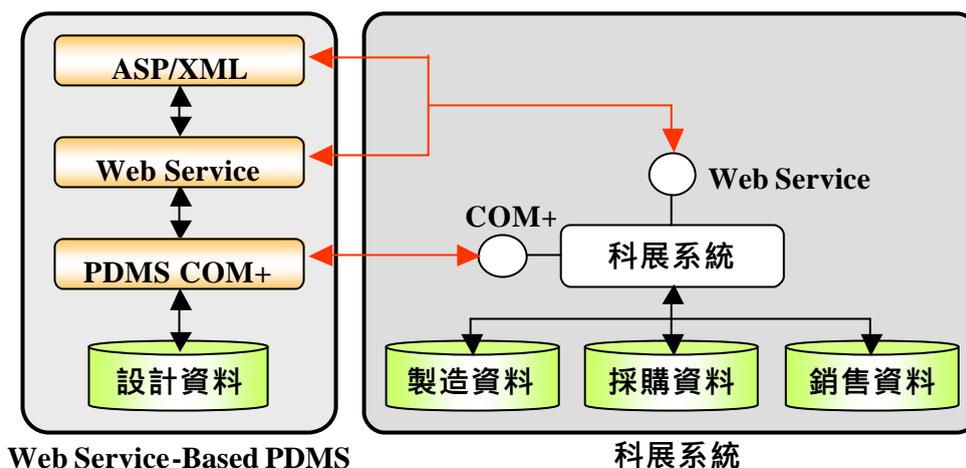


圖 5-36 本系統資訊整合方式

第六章 結論與建議

網路經濟之衝擊與全球化競爭時代的來臨，使得企業傳統經營模式備受考驗，如今企業不僅需面對大陸、東南亞等低生產成本國家之強烈競爭，更需整合處理不斷流動於網路上的商業資訊。因此，如何促進產業升級、增強設計研發能力、建置價值鏈體系 e 化環境，進而提升產業整體競爭實力，已成為產官學界共同討論之焦點。

產品開發的成本耗費(或總價值)約有 80% 決定於設計研發階段【51】，因此強化設計研發階段的工作質量，將是企業提升競爭力之重要關鍵。PDMS 不僅為管理設計研發資料與流程的重要工具，更是製造企業 e 化主要核心，故其系統功能及整體架構良衰與否，勢必對產品開發效能與企業 e 化成效影響甚深。目前 PDMS 以元件式 Web-Based 系統架構為主，由於元件模型技術侷限於特定平台與軟體廠商，加上 Web-Based 系統缺乏訊息傳遞標準，難以達到資訊自動化處理目標，因此 Web-Based PDMS 尚無法完全滿足現今網路商業環境需求。

於 Web Service 協定標準制定以前，產品生命週期開發資訊整合尚無一良好解決方案，因此本研究基於 Web Service 之優勢與發展前景，以元件模型、SOAP、分散式系統等理論技術作為應用基礎，研究與實作 Web Service-Based PDMS，試圖提供一思考方向與解決方案，使 PDMS 具備跨平台、高適應性、易於整合管理等系統特性，進而克服現今 Web-Based PDMS 的缺點。此外，本研究深入訪談企業所得之調查分析結果，以及 Web Service-Based PDMS 理論架構、開發模式與系統建置範例，亦期能作為學術研究參考改良依據，實際解決企業產品資訊整合與管理之問題。

6.1 研究結果

以下即針對本研究訪談企業、實作 Web Service-Based PDMS 之研究結果，整理分述如下：

一、 企業訪談結果

1. 企業經營管理目標與 e 化發展策略：

根據文獻資料與企業訪談結果顯示，企業經營管理目標已趨向追求產品創

新，以及縮短上市時間(Time to Market)為主；e 化發展策略則以建置產品開發資訊整合環境為目標，中小企業願意投資單一系統金額上限以 300 萬元為主。對於電子商務與協同產品商務之發展，多數企業抱持樂觀態度，認為企業實體營運需充分結合網際網路，藉以擴展全球行銷通路，以及整合產品開發資源。

2. 企業產品資訊整合：

雖然多數企業之管理目標及 e 化方向與學術理論差異不大，亦符合知識經濟時代之競爭條件，然而於實際執行面上，卻往往因缺乏設計製造整合觀念、資訊開發能力不足、e 化實行步驟方法錯誤等因素，使得產品資訊整合不但無法達到預期成效，反而因企業內外資訊化孤島的產生，耗費企業龐大的資源與成本。

3. 產品資料管理系統：

國內企業已普遍認知產品設計研發之重要性，但絕大部分依然侷限於發展僅有 20%改良效益的生產系統，而忽略擁有 80%潛能空間的設計研發領域。本研究訪談 10 家企業中，僅有一家中型企業正式導入 PDMS，但由於其所導入的 PDMS 為套裝軟體系統，因此目前尚未與其他系統整合運作。歸結企業未能導入 PDMS 的主要因為(1) 企業不知 PDMS 的主要用途與應用範圍。(2) 企業資訊系統開發能力不足，無法建置與導入 PDMS。(3) PDMS 套裝軟體導入費用昂貴，一般中小企業無法負擔。(4) 企業所使用的 PDMS 系統功能不符合使用者實際需求。

4. 設計資料管理模式：

絕大多數企業設計資料管理模式以文件資料櫃與區域網路傳輸為主，透過企業所導入的 ISO 文件檔案編碼規則，整理與歸納設計文件及圖檔資料。由於圖檔編碼原則過於複雜、相關資料取得不易、設計資料使用與管理程序尚未系統化，使得開發人員必須耗費大量時間成本，處理與搜尋設計相關資料。此管理模式除了額外增加開發人員的工作負擔外，亦使其無法全力專注於設計研發工作，進而造成設計研發部門與產品開發管理效能低落，不僅浪費企業資源，更影響產品開發時程與上市時間。

二、Web Service-Based PDMS 實作結果

1. 系統功能符合設計師實際需求：

本研究之 Web Service-Based PDMS 實作了(1) 網站入口系統。(2) 線上搜尋引擎。(3) 產品資料管理系統。(4) 零件資料管理系統。(5) 模具資料管理系統。由於系統功能是以多位設計師進行設計研發工作時，所實際面臨到之問題點與需求點作為開發依據，加上內建多種自動化處理機制，盡量減少人為操作程序，因此可確切改善傳統產品文件、圖檔管理方式，進而達到增進企業設計研發效率之目的。

2. Web Service-Based 系統可取代 Web-Based 系統：

研究發現 Web Service-Based 系統與 Web-Based 系統之差異，僅在於 Web Service 介面的呼叫與使用，其餘使用者介面、資料之呈現方式並無太大差異。本研究分別建置了同樣功能的 Web-Based PDMS 與 Web Service-Based PDMS，其系統執行結果並無不同，然而於執行效能上，Web Service-Based PDMS 則較為低落，其主要原因為 Web Service-Based PDMS 需額外傳遞與處理介面呼叫資訊。Web Service-Based 系統效能相對之下雖然較差，但其資訊之通透性與自動化處理能力卻遠高於 Web-Based 系統。由於目前已有眾多作業平台與開發工具支援 Web Service，因此建置一 Web Service-Based 系統應無太大的阻礙與技術困難點。此外，若將 Web Service 結合元件技術，除可補足元件無法與其他異質元件整合之缺點外，更可增加系統執行效能，以及後續維護管理的彈性。

3. 採用瀏覽器介面可使產品資訊應用更為多元：

Web Service-Based PDMS 可採用多種前端使用者介面，然而若以瀏覽器 (Browser)操作介面為主的話，則系統將具備以下之優勢(1) 能利用權限控管以及後端介面處理等方式，使系統依照不同的使用權限與用途，呈現適當地產品資訊與系統功能。(2) 開發人員可不限時間空間之限制，利用 Internet 使用及管理產品開發資訊。(3) 無須一一分發前端程式，大幅降低後續系統開發及管理之難度。(4) 可於後端結合其他 e 化系統，並將產品資訊與開發流程整合於同一系統畫面上，降低企業內外 e 化系統的使用複雜度。(5) 可

進一步應用與擴展 PDMS 系統功能，使其支援電子商務、協同產品商務，以及其他線上產品資訊相關系統之發展。

4. 企業應加強產品資訊整合觀念及系統開發能力

本研究所開發之 PDM 系統範圍與功能，雖遠低於專業的 PDM 套裝軟體，但本系統之功能卻可解決企業常見之設計資料管理問題。然而就本研究所訪談之企業而言，尚未有企業認知僅需開發一小系統，即可大幅改善傳統產品資料整合與管理問題，探究其原因為國內中小企業缺乏產品開發資訊、流程整合觀念、系統開發能力不足，以及未能充分了解使用者實際需求等，一味採用各式套裝軟體或者將系統外包，最終雖系統可能導入成功，但企業卻難以從中學習經驗，進而提昇自身資訊開發能力。本系統若屏除元件技術、分散式系統架構，單就 Web Service 與 ASP 而言，雖然其資訊技術門檻不高，但依然可完成本系統所有功能，發展出一套基本且適用的 Web Service-Based PDMS，除了可有效降低開發成本外，更重要的是企業可藉此培育產品開發資訊整合人才，以因應企業未來發展之需求。

6.2 研究貢獻

本研究訪談國內 10 家中小企業，整理歸納產品資訊整合與設計研發管理問題，最終以瑞振企業之開發需求與產品資料為例，導入本文之理論觀念所實際建置之 Web Service-Based PDMS，可為企業帶來之效益如下：

1. 強化產品開發資訊整合運用觀點，檢視企業 e 化系統與導入成效。
2. 增進產品資料管理效能、縮短開發時程與人力資源成本浪費。
3. 減少企業 e 化系統整合阻礙與資訊化孤島產生之風險。
4. 擴展 PDMS 應用領域，發揮產品資訊整合效益。

本文所提之 Web Service-Based PDMS 除可應用於企業產品開發實務外，在學術研究領域上之貢獻如下：

1. 比較分析現今主要產品開發模式之優缺點，並提出企業實施導入建議。

2. 透過整理歸納 PDMS 之演進發展及功能架構，釐清 PDMS 於企業 e 化系統中之定位與應用範圍，避免企業盲目導入不適當之 PDMS 系統功能與架構。
3. 比較分析現今各式 Web 系統架構與開發模式之優缺點，說明 Web 系統運行架構與未來發展趨勢。
4. 深入訪談 10 家國內中小製造企業，了解企業營運發展現況，整理產品資訊整合之問題點與需求點。訪談調查分析之數據及結論，可提供學術領域作進一步地研究與探討。
5. 提出 Web Service-Based PDMS 理論架構與開發模式，作為產品全生命週期資訊整合之基礎，提供各界後續研究改良之依據。
6. 實際建置一基本的 Web Service-Based 產品資料管理系統，說明其系統開發觀念、資訊架構、軟體技術及功能流程等，提供企業與學術界作進一步地發展與延伸。

6.3 研究限制

基於種種研究限制條件因素，本研究尚未能完善解決之處如下所述：

1. 本研究企業訪談係採用案例調查方式，由於樣本數僅有 10 家國內中小企業，因此調查所整理分析之結論，僅能就此 10 家企業之現況加以說明及推論，無法完整代表其他企業之狀況，此為質性研究之盲點所在。本調查的主要目的在於了解企業現況，雖樣本數的代表性不足，但訪談結果可真實反應受訪企業的實際情形，藉由本文所整理的企業現況與問題點，可提供後續研究作進一步的量化與統計分析。
2. 於 Web Service-Based PDMS 理論架構中，強調的是利用元件模型與 Web Service 標準，達到產品開發資訊的整合與運用，然而由於受限於科展系統資料庫欄位尚未釐清，以及產品加工程序並未整理歸納，因此本研究尚未整合科展系統，達到設計製造資訊、流程整合管理，以及實際驗證 Web Service-Based PDMS 具備跨平台、跨系統與資訊自動處理之特性。
3. 本文雖已論述分散式系統平衡負載與容錯機制，對於企業發展網路商務之重要性，然於本系統開發建置環境中，尚未適合導入採用，未來若系統開發範圍擴大，使用者連線需求增加，則可針對此一部份加以驗證與探討。

4. 本研究之重點與範圍主要在於系統整合平台架構的建立，以及設計資料、圖檔的管理等，雖已完成基本的 Web Service-Based PDMS 系統功能，但未來應再納入更多 PDMS 管理功能與國際標準，並考量其他產品開發階段資訊整合之需求。

6.4 後續研究建議

本研究雖已運用 Web Service-Based PDMS 理論架構與開發模式，實際建置一基本的產品資料管理系統，然而因為研究之限制，尚未能使 Web Service-Based PDMS 發揮整體效益，提供各界更完善的參考範例與驗證結果，因此於後續研究發展方向，本研究之建議如下：

1. 企業整合方向分為資訊整合與流程整合兩種，本研究主要偏向於前者並提出產品資訊整合基本架構與範例，後續研究則可依此再進一步發展設計相關管理系統，例如：設計審議管理、設計概念發展系統、產品專利管理、開發成本管理、產品知識管理【52】、同步工程、協同產品商務等開發流程，使產品開發資訊、流程均能在完善權限控管機制下加以整合與發揮企業整體效能。
2. Web Service-Based PDMS 主要以資料、功能、協調等三大元件為基礎，運用元件模型技術與 Web Service 協定標準開發整體系統功能，後續研究可依系統開發範圍，持續深入設計分析 PDMS 系統功能，尋找與切割此三大元件，作為後續 PDMS 開發與 e 化系統整合的重要基礎建設。
3. 跨企業產品資料整合與協同設計為未來企業不可避免之趨勢，後續研究可運用國際性的資料表達交換標準，例如：STEP(STandard for the Exchange of Product data)、IGES(Initial Graphics Exchange Specification)等，結合本研究所建議之系統架構與開發模式，建置系統應用範圍更廣泛的產品開發資訊整合平台。
4. 由於本研究資源、開發技術大多侷限於特定平台、裝置以及系統開發工具，因此後續研究可嘗試採用各式系統平台(例如：Linux、Unix、Mac OS、OS/2 等)、前後端裝置(例如：PDA、Mobile Phone、CNC 等)、系統開發工具(例如：Java、Visual Studio.Net 等)，建置與運用 Web Service-Based PDMS，發展更為多元且符合企業實體商業環境需求的產品資料管理系統。

參考文獻

- 【1】 eTown 網路論文, “透過網際網路, 讓世界更美好!”,
<http://www.ETOWN.com.tw/event/document/doc1.htm>
- 【2】 陳光榮, “網路科技發展對企業經營模式及國際競爭態勢影響之分析”, 經濟部
經濟情勢暨評論季刊, 第五卷第二期, 1999.09
- 【3】 郭炳宏、鄭再添, 建構企業產品商務系統之研究與應用, 私立東海大學工業設
計學系碩士論文, 第 18~22 頁, 2000.07
- 【4】 Michael E. Porter, 競爭優勢(上), 天下文化出版, 1999
- 【5】 T. G. Lewis, THE FRICTION-FREE ECONOMY, 1998
- 【6】 郭炳宏、鄭再添, “數位化產品設計支援系統”, 中華民國設計學會, 第 929-934
頁, 2000.05
- 【7】 Iansiti, M. and MacCormack, A., “Development Products on Internet Time”, Harvard
Business Review, 1997.10
- 【8】 CNET 新聞專區報導, <http://taiwan.cnet.com/news/ec/story/0,2000022589,20021019,00.htm>, 2001.08
- 【9】 CNET 新聞專區報導, “協同設計為台灣轉型 ODM 關鍵”,
<http://taiwan.cnet.com/news/ec/story/0,2000022589,20033085,00.htm>, 2002.06
- 【10】 郭炳宏、蔡智銘, 企業產品整合開發資訊控管系統之研究, 國科會專題研究成果
報告, 2002.10
- 【11】 鄧世剛, “產品資料管理系統在同步工程之應用”, CADesigner, VOL.121, 第 42~48
頁, 1998
- 【12】 曾淑峰, “元件式系統開發新思維”, 軟體產業通訊, VOL.30, 第 12~16 頁, 2000.08
- 【13】 軟體產業服務團, “軟體元件定義與發展現況”, 經濟部工業局, 2000
- 【14】 李維, Delphi6/Kylix2 SOAP/Web Service 程式設計篇, 旗標出版社, 2002.01
- 【15】 蔡智銘, 企業產品資訊服務控管系統之研究, 工業工程學會學生論文競賽, 2003
- 【16】 任文瑗, “電子化企業--E-Business”, 資訊與教育, 76 期, 第 78~86 頁, 2000.04
- 【17】 PTC 技術專題, “產品協同商務-利用 Internet 促進產品創新”,
<http://www.ptc.com/hongkong/products/windchill/2001/sec1/index.htm>, 2001
- 【18】 張裕益, UML 理論與實作, 博碩文化, 2001。

- 【19】Booch, G., Rumbaugh, J. and Jacobson, I., *The Unified Modeling Language User Guide*, Addison-Wesley, 1999。
- 【20】Robert H'obbes' Zakon, “Hobbes' Internet Timeline”,
<http://www.zakon.org/robert/internet/timeline/>, 2003
- 【21】陳鴻仁, “長纖梭織業之產業電子化發展與未來趨勢”, 絲織園地季刊第 33 期, 2000.07
- 【22】劉坤堂, “經營現代化與網際網路商業應用”, 今日會計, 72 期, 第 105~111 頁, 1998.09
- 【23】柯獻凱, “e 世代企業-企業電子商務(e-Business)介紹”, 財團法人中衛發展中心專業文摘, 2000.03
- 【24】Bill Gates 著, 樂為良譯, 數位神經系統, 商周出版社, 1999
- 【25】艾一資管股份有限公司-ERP 論壇, “ERP 的延伸功能- 電子商務介紹”,
<http://www.iemis.com.tw/>
- 【26】劉坤堂, “我國電子商務的政策與應用現況研析”, 電工通訊, 第 6~23 頁, 1999.09
- 【27】黃廷合, “論電子商務的發展與推廣”, 商業職業教育, 74 期, 第 45~51 頁, 1999.08
- 【28】林淑禎, “B2B EC 之發展模式探討”, 永昌投顧季刊, 第三季季刊, 2000。
- 【29】台灣微軟, “電子商務產業趨勢導讀”,
http://www.microsoft.com/taiwan/e-commerce/ec/ec_01.htm, 2000
- 【30】Nonaka, I., Takeuchi, H., and Umenoto, K., “A theory of organizational knowledge creation” *Int.J.Technology Management*, Vol.11, pp.833-845, 1996.
- 【31】王魁生, 蔡志弘, 李榮貴, “整合性產品開發管理系統之構建”, 中華管理評論, 3 期, 第 149 171 頁, 2000.03
- 【32】姚威宏, 產品資料管理於協同產品開發之整合-以少量多樣化產品為例, 東海大學工業設計學系碩士論文, 2002.07
- 【33】參數科技, “數位化資訊時代的產品開發”, 0 與 1 BYTE 雜誌, 2000.02
- 【34】中衛發展中心, “漢翔 PDM 及同步工程應用”, <http://cals.csd.org.tw/1-3-2.html>, 2002
- 【35】王台平, “產品資料管理系統之發展趨勢”, *CADesigner*, 121 期, 第 36~41 頁, 1998
- 【36】陳恬立, “用 CPC 改善流程”, 參數科技網站, 2002.04

- 【37】曾令衛，“協同產品商務 CPC 研究”，IT 經理世界，2002.05
- 【38】孟繁晶，“協同產品商務(CPC)及其建立過程之研究”，
<http://www.e-works.net.cn/ewkArticles/Category41/Article12723.htm>，2003.01
- 【39】陳柏辰，以元件式設計建構晶圓代工廠 CRM 架構 - 實作換單管理系統，交通大學工業工程與管理學系碩士論文，2001.06
- 【40】李維，Delphi5 分散式多層應用系統篇，旗標出版社，1999.11
- 【41】李維，分散式多層應用電子商務篇，旗標出版社，2000.02
- 【42】黃文石、陳惠淳，“Web Services 之規格現況探討”，資策會 EC 實驗室，2002.04
- 【43】喻瀚寬、蔡澤銘，“解析 Web Service 的技術內容與意涵”，資策會電子商務技術實驗室，2001.06
- 【44】The UDDI information，<http://www.uddi.org>。
- 【45】The SOAP information，<http://www.w3.org/2000/xp/Group/>。
- 【46】劉遠威、黃雯汝，“Web Services 帶來整合革命”，資策會 EC 實驗室，2002.04
- 【47】顏春煌，“系統化的軟體元件再使用方法”，管理與資訊學報，第 241~267 頁，1998.04
- 【48】周斯畏，物件導向系統分析與設計/使用 UML 與 C++，全華出版社，2002.01
- 【49】柳宏昌，“認識 UML”，元擎軟體股份有限公司專題報導，2000.10
- 【50】Microsoft，“SOAP Toolkit3.0”，<http://www.microsoft.com/downloads/details.aspx?FamilyID=c943c0dd-ceec-4088-9753-86f052ec8450&DisplayLang=en>，2003.04
- 【51】參數科技，“先 CPC，後 ERP”，http://www.shapinginnovation.com/hongkong/news/clippings/internet_0402.htm，2002.04
- 【52】黃致理，產品知識管理系統之建構-以組裝性製造業為例，私立東海大學工業設計學系碩士論文，2000.07
- 【53】工業局自動化與電子化小組網站，<http://proj.moeaidb.gov.tw/iaeb/>
- 【54】賴杉桂，“權變管理”，台灣通訊-高峰論壇，<http://www.taiwanetelcom.com.tw/pj30pg232.asp?Tag=8468>
- 【55】Lee,H.L.,V.Padamanabhan,and S.Whang，“Information Distortion in a Supply Chain：The Bullwhip Effect”，Management Science. Vol.43.No.4，1997.04
- 【56】Carter, D. E. and Baker B. S.，Concurrent Engineering: The Product Development Environment for the 1990s，Addison-Wesley, NY，1992

審查委員問題與建議

審查委員	主要問題與建議	修正與處理
林 銘泉老師	內文、表格、圖形、文獻之格式與排版建議。	絕大部分內容均已採用林老師之建議格式修改論文，唯少部分因屬於特殊專有格式或用法，所以較難以符合其格式要求。
	問卷調查分析缺乏使用統計方法。	由於本研究之企業訪談調查，其主要目的在於了解企業現況，以及找尋產品資訊整合的問題點，因此現階段的重點為透過質性研究(亦即案例式的深入訪談)，先取得企業的問題點與需求點，作為系統建置背景與功能開發依據。至於欲以不同的統計方法擷取分析訪談結果，則可能需先將訪談問題作廣泛的量化，取得相當的企業樣本數，如此統計分析之結果才能具備一定程度的指標性與完整度。本文因研究限制之關係，尚未能作廣泛的量化與統計分析，此點於研究限制及後續研究建議中已再加以論述。
	系統功能如何建置應予以說明。	已針對系統功能的核心「設計圖檔管理」的整個開發模式與程式碼作詳細的說明，如附錄 D 所示。
	系統開發應更強調設計研發領域。	本研究主要以企業常見之設計研發管理問題與需求，作為系統開發的重點與核心，因此系統建置的範圍主要以設計研發領域為主。然而或許因為本系統尚未完整具備所有設計資料、流程管理功能，使得於設計領域之系統功能較為缺乏，關於此點已於後續研究中更加詳細地說明。

吳 銜容老師	產品資料表達與交換應採用通用標準，如 STEP 等。	STEP 與 IGS 等標準，主要用於資料的表達與交換之用，其對於圖檔的轉換、產品資料交換等，均扮演不可或缺之角色。然而，本研究的重點與範圍主要在於系統整合平台架構的建立，以及設計資料、圖檔的管理，對於 STEP、IGES 等資料交換標準的運用，可能需等系統發展至整合供應鏈管理、協同產品開發等範圍，才能突顯採用資料表達交換標準的效益，此點於研究限制與後續研究建議中，已將其列入並加以論述。
	系統流程與圖形應反應設計概念或設計活動的形成。	運用人工智慧的方法，發展設計概念與處理設計活動流程，已日益受到各界之重視。由於本研究範圍主要是以建構整合性設計資料管理平台為主，因此系統的流程與圖形，對於設計概念、活動之發展較為不符，而關於此點建議已將其納入後續研究中。
	強調 Web Service 的功能用途。	此點於 2.4.5 節(Web Service)、3.1(Web-Based 與 Web Service-Based System 之比較)、3.2(Web Service-Based PDMS 理論架構)已補強論述。
張 炳騰老師	再強調研究目的與系統開發範圍。	此點於論文摘要 1.3(研究目的)、1.4(研究方法與流程)中已再補強說明。
	界定清楚異質系統的涵義。	此點於 3.2(Web Service-Based PDMS 理論架構)、3.3(Web Service-Based PDMS 開發模式)中已再加以論述說明。
	理論架構應更突顯說明。	已修正第三章(研究理論與架構)之圖形，使整個圖形更能清楚且詳細地表達研究理論架構，系統開發模式亦採用老師之建議，以步驟程序的方式加以說明。

附錄 A

企業產品資訊整合現況調查問卷

各位先進您們好!由於近年來協同產品商務以及各式 e 化方案之興起,促使企業不斷運用 IT 技術,試圖整合產品開發過程中的所有產品資訊,發揮企業整體營運之最大效益。此外,企業面對此一趨勢衝擊,也有著各自的想法與因應之道,因此本研究盼望能透過此問卷,了解目前企業對於產品資訊整合的需求與策略方法,進而整理分析,提供各界一研究之參考。以下有關產品資訊整合運用之問題,還望各位不吝指教,給於本研究寶貴的答案,謝謝!

1. 就您的實際經驗與體會,您認為於目前的競爭環境中,現今企業的管理目標與策略,其重要程度順序為何?(請依重要程度填寫 1-9)

速度	交期	創新
全球化	服務	彈性
協同整合	成本	品質

2. 您是否認為與上下游廠商或合作夥伴,共同推行協同產品商務、分享產品資訊,是不可避免之趨勢?

是 否

3. 您是否認為企業除了目前實體營運之外,還需與網路電子商務作緊密結合?

是 否

4. 您認為下列哪一資訊管理系統為發展協同產品商務或 e 化的主要核心?

供應鏈管理(Supply Chain Management : SCM)。

企業資源規劃(Enterprise Resources Planning : ERP)。

產品資料管理(Product Data Management : PDM)。

客戶關係管理(Customer Relationship Management : CRM)。

知識管理(Knowledge Management : KM)。

其他 _____。

5. 您所服務的公司是否已導入類似協同產品商務,與其他企業或上下游廠商,透過國際網路一起協同合作開發產品?

是的,已具備完整設計管理功能

是的,但尚未具備完整設計管理功能

尚未導入,但認為未來有其必要

尚未導入,認為不適合導入

6. 您所服務的公司，設計研發部門是否能夠簡易地使用一套系統，查詢與使用產品相關資料作為設計研發之參考？
- | | |
|------------|-------------------|
| 沒有，認為沒有必要 | 有的，但功能不足，對設計師效益不大 |
| 有的，對設計師有效益 | 沒有，但認為有其必要 |
7. 您是否認為，若設計師能將產品相關資訊作個人化的設定，以便其隨時查詢與使用產品資訊，將會對其設計研發工作有所助益？
- | | |
|-------|---------------|
| 非常有助益 | 有助益 |
| 沒有助益 | 沒有助益，反而造成工作負擔 |
8. 您所服務的公司，是否有導入類似同步工程的產品開發模式，讓產品開發的相關人員(例如：製造工程師、測試工程師等)，得以參與設計研發的工作，並提出其專業意見？
- | | |
|--------------|------------|
| 是的，已導入同步工程模式 | 規劃建置中 |
| 沒有，但認為有其必要 | 沒有，認為不適合導入 |
9. 您所服務的公司是否已具備產品資料管理系統，用以系統化管理產品設計與製造等開發流程所產出之資料？
- | | |
|--------------|--------------|
| 是的，由公司自行開發 | 是的，購買套裝軟體並導入 |
| 規劃建置中 | 尚未建置，但認為有其必要 |
| 尚未建置，認為不適合導入 | |
10. 您所服務的公司，其產品資料管理系統(PDMS)所採用的架構為何？
- | | | |
|------------|---------|----------|
| 元件式網頁形式的架構 | 網頁形式的架構 | 主從架構 |
| 三層式或分散式架構 | 不清楚 | 其他 _____ |
11. 您所服務的公司，目前所導入的所有資訊管理系統，是否已彼此串連，相互整合運用？
- | | |
|---------------|-------------------|
| 是的，系統間已彼此串連運作 | 是的，但僅有少部分系統尚未串連整合 |
| 沒有，但認為有必要 | 沒有，認為沒有必要 |
12. 您認為您所使用的資訊管理系統，有那些系統操作面或功能面上的缺點?(請依缺點大小，依序填寫 1,2,3,4...)
- | | |
|--------------|---------------|
| 無法將資訊作個人化的設定 | 難以與其他資訊系統整合運用 |
| 使用方式或操作介面太複雜 | 後續功能增加或維護管理困難 |

難以達到自動化處理之目的 系統執行效率不彰或不穩定
 安全性不佳 其他_____

13. 就您所知，您所服務公司曾經投資建置單一系統的金額上限為何？

100 萬元以內 300 萬元以內 500 萬元以內
 1000 萬元以內 1500 萬元以內 其他 _____

14. 您認為企業欲整合運用企業內外所有產品資訊的主要動機順序為何？

(請依動機順序填寫 1 ~ 5)

降低產品開發成本 提升產品開發效率 提升產品資訊管理能力
 提高產品質量 快速得到投資回報 其他 _____

15. 請您寫下您對產品資訊整合趨勢的看法，或是對企業 e 化的感觸：

16. 最後請您告訴我們，您的背景資料：

公司類別	製造業	資訊業	服務業	零售業	營造業
	金融業	政府機關	其他 _____		
經營型態	OEM(原廠設備製造)		ODM(原創設計製造商)		
	OBM(自有品牌製造)		其他 _____		
服務年資	1 年以下	1 ~ 2 年	2 ~ 4 年	4 ~ 6 年	6 年以上
工作職務	一般員工	中階主管	高階主管		
工作部門	設計部	生產部	資訊部	其他 _____	

再次感謝您填寫本問卷以及提供給我們的寶貴意見，敬祝您順心如意!

東海大學工業設計研究所
 產品開發設計研究室

附錄 B

物料、圖面編碼原則

物料編碼原則

第一碼 大類

A:土除類	B:菜籃類	C:鏈蓋/齒盤蓋	D:輔助輪類	E:童椅類	F:燈架/反光架
G:吊車架	H:蓋類	I:其他類	J:模具類	K:	L:
M:金屬材料	N:	O:其他材料	P:塑膠材料	Q:包裝材料	R:
S:標準零件	T:	U:	V:	W:	X:說明書
Y:	Z:				

第二、三碼 細分類

A:土除類

FB:螺絲包	FM:螺絲附件混裝包	MB:取付	MG:土除	MH:吊耳
MP:附件	MS:支架(腳) 土除尾	SH:土除腳耳	FP:附件包	

B:菜籃類

BB:取付	BH:菜籃固定架(604)	BP:附件	BS:支架(腳)	FB:螺絲包
FM:螺絲附件混裝包	FP:附件包	BK:菜籃		

C:鏈蓋、齒盤蓋

CB:取付	CC:齒盤蓋本體	CG:鏈蓋本體	CP:附件	FB:螺絲包
FM:螺絲附件混裝包	FP:附件包			

D:輔助輪類

FB:螺絲包	FM:螺絲附件混裝包	FP:附件包	TA:蛇管	TB:取付	
TC:輪蓋	TP:附件	TR:輪框	TS:支架	TT:輪胎	TW:輪子

E:童椅類

BB:取付	BC:童椅	BP:附件	BS:支架	FB:螺絲包
-------	-------	-------	-------	--------

FM:螺絲附件混裝包

FP:附件包

F:燈架/反光片架

LB:燈架

RB:反光片架

G:吊車架類

BH:吊車架

BP:附件

FB:螺絲包

FM:螺絲附件混裝包

FP:附件包

H:蓋類

BC:黑塞

CC:曲柄塞

CP:曲柄保護蓋

DH:磨電蓋

DP:後變保護蓋

HC:花鼓蓋

OP:油箱蓋

I:其他類

BN:氣門嘴

CG:導線夾

CJ:拖車架

FA:擴大器

LC:行李架(602)

PT:前輪定位器

PF:價格架

RP:PVC 片

SD:坐管防水套

SL:變速桿

W:水瓶

BP:附件

TL:工具類

DG:護裙網

GP:把手套

圖面管理辦法

圖面圖號依下列兩種編碼方式，架構如下：

編碼方式一(基本編碼)：

\ \ \ \ -

圖 產 產 加 型 變
別 品 品 工 號 更
 別 主 別 別 別
 件
 別

圖別：

圖別代碼	1	2	3	4	5
類別	分圖	簡圖	組立圖	總圖	其他

編碼方式二(依客戶編碼)：

\ \ \ \ -

圖 客 產 產 加 型 變
別 戶 品 品 工 號 更
 編 別 主 別 別 別
 號 件
 別

圖別：

圖別代碼	A	B	C	D
類別	國外客戶	貿易商	組車廠	其他

加工別代碼如下表所示：

代碼	1	2	3	4	5	6	7	8	9	10	11	12	13
零半 組成 件品 及類 別	射出 成型	押出 成型	衝壓 成型	? 合	車床 加工	吹出 成型	真空 成型	硫化	組裝	壓鑄	擠型	擠出	其他 加工

產品別代碼如下表所示：

代碼	A	B	C	D	E	F	G	H	I	J	M	O	P	Q	S	X
產品 別	土除 類	菜籃 類	鏈蓋 齒盤 蓋	輔助 輪類	童椅 類	燈架 反光 架	吊車 架	蓋類	其他 類	模具 類	金屬 材料	其他 材料	塑膠 材料	包裝 材料	標準 零件	說明 書

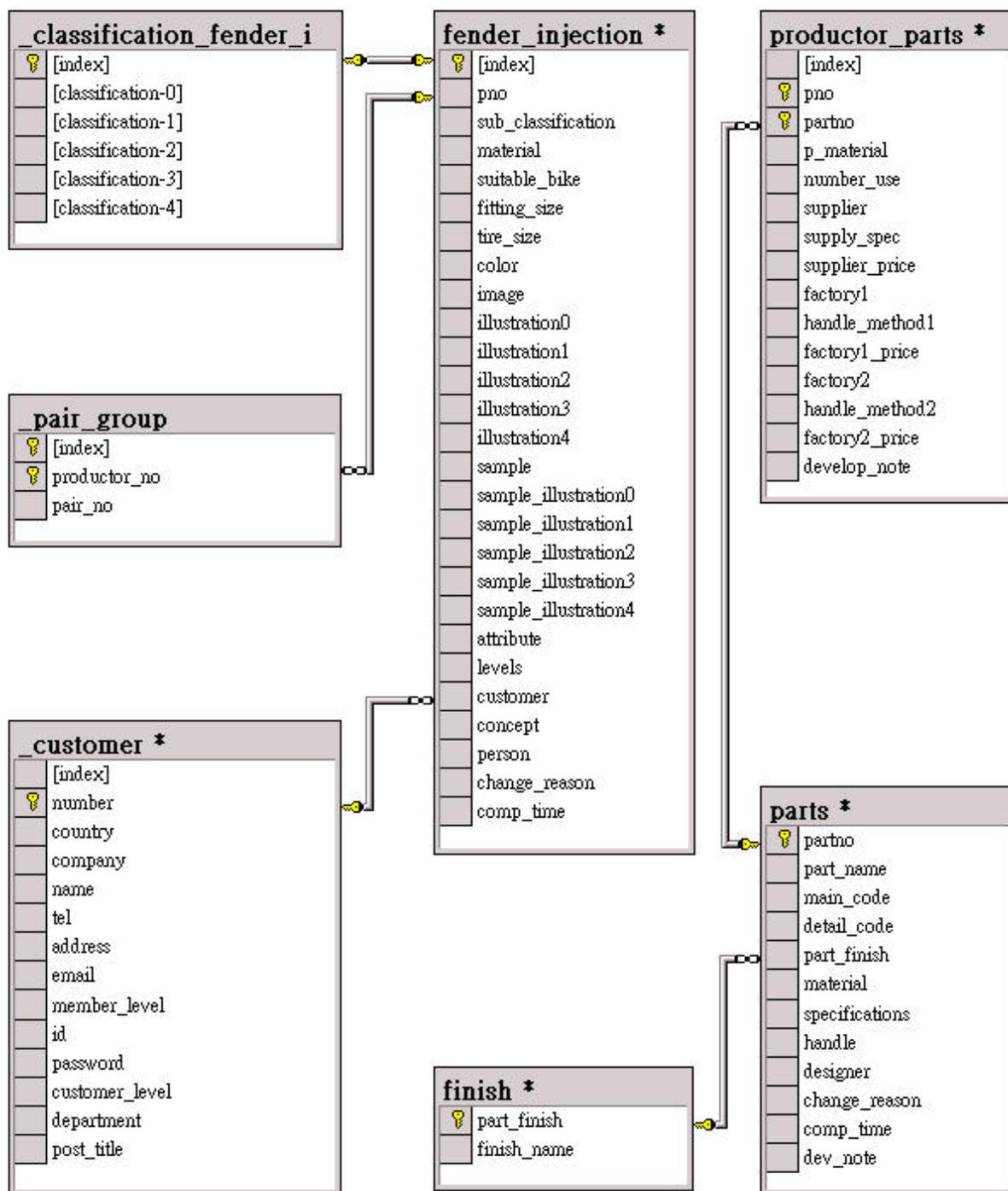
附錄 C

由於本系統產品類別眾多，因此以下僅列出單一產品類別(fender_injection)，以及其他系統重要資料表格，以下即為本系統產品資料庫表格簡述：

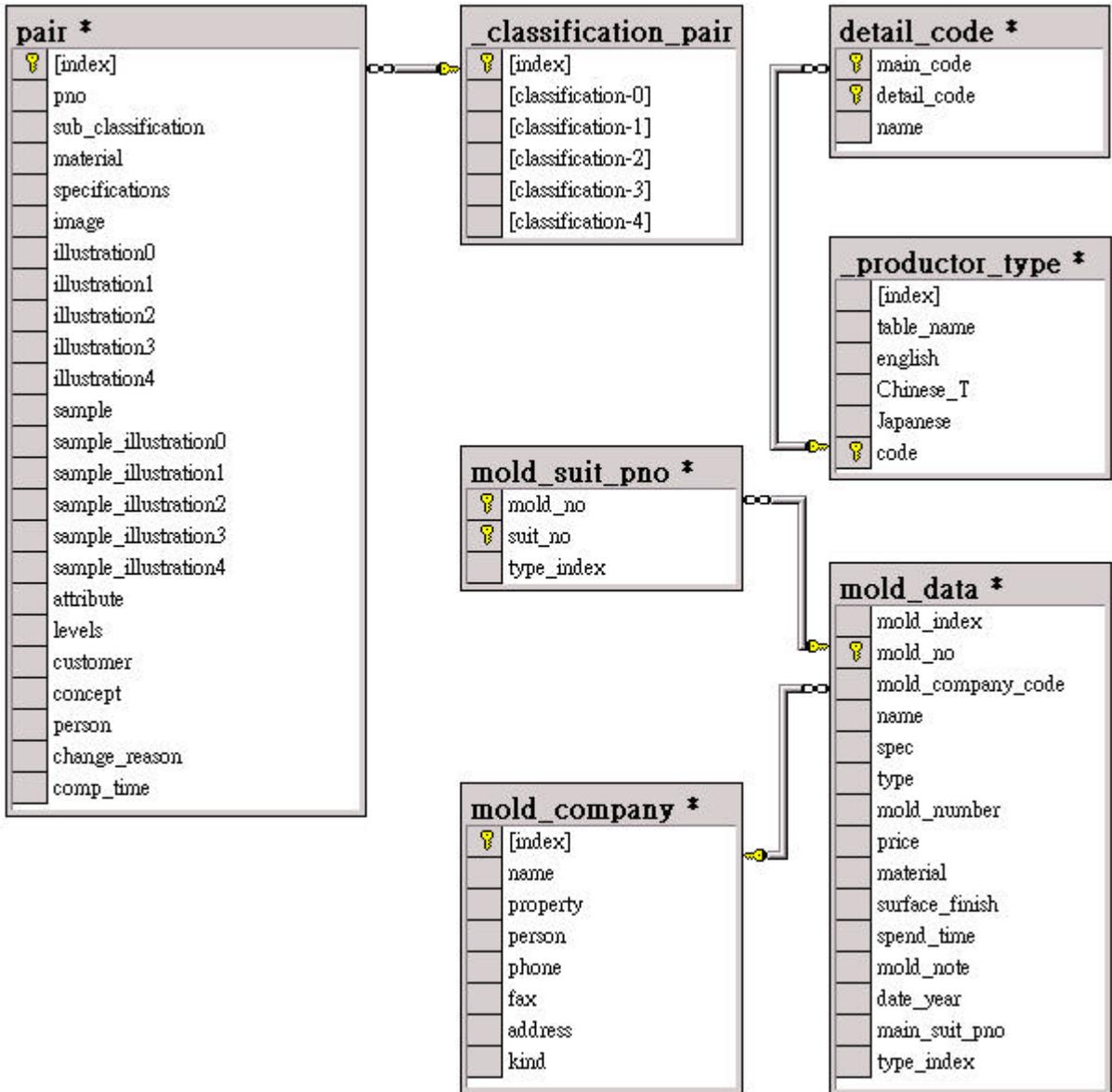
本系統資料表簡述

資料表名稱	簡述
fender_injection	擋泥版產品類別，紀錄產品主要資料內容。
_classification_fender_i	擋泥版子類別。
productor_parts	產品零件物料清單，紀錄產品由那些零件所組成。
parts	零件資料表，紀錄零件主要資料內容。
pair_group	配件組合資料表，紀錄產品擁有那些配件組合。
customer	客戶資料表，紀錄客戶與內部成員主要資料內容。
finish	加工方法資料表。
pair	配件資料表。
_classification_pair	配件子類別資料表。
_productor_type	產品類別資料表。
detail_code	詳細產品類別代碼資料表，紀錄企業 ISO 制度所規範之產品類別資料代碼。
mold_data	模具資料表。
mold_company	模具公司資料表。
mold_suit_pno	模具可適用於開發產品、零件型號資料表。

本系統主要產品資料表格關聯如下所示：



接續下頁



附錄 D

圖檔管理模式與系統開發說明

本系統圖檔主要分為產品、零件、模具三種，雖圖檔類別項目不同，但管理模式與系統處理機制均大致相同，因此以下僅以產品圖檔分別說明一、圖檔目錄結構與編碼原則 二、圖檔搜尋機制與程式碼 三、Web Service 轉換圖檔搜尋資訊 四、圖檔上傳機制與程式碼 五、圖檔刪除機制與程式碼 六、線上瀏覽機制與程式碼。

一、圖檔目錄結構與編碼原則

目錄路徑	編碼原則	用途說明
根目錄\image\ productor\	型號-變更別.副檔名	網站上顯示的型錄圖形
	型號-變更別(demo).副檔名	網站上顯示的展示圖形
根目錄\images\productor\型號-變更別\	流程圖.副檔名	產品開發流程圖
	process.dwf	線上瀏覽用之流程圖
	組立流程圖.ppt	組立流程圖
	型號-變更別.副檔名	詳圖檔案
	型號-變更別_2.副檔名	簡圖檔案
	型號-變更別_3.副檔名	組立圖檔案
	型號-變更別_4.副檔名	總圖檔案
	型號-變更別_5.副檔名	其他圖示檔案
	型號-變更別-Model.dwf	線上瀏覽 2D 平面圖
	型號-變更別_nc.副檔名	NC 檔案
	型號-變更別_s.jpg	掃描的草稿圖形
	型號-變更別_c2.副檔名	客戶所提供的 2D 圖檔
	型號-變更別_c3.副檔名	客戶所提供的 3D 圖檔

二、圖檔搜尋機制

以下即分別說明採用 Delphi Pascal 與 ASP 程式語言，如何搜尋設計圖檔：

Delphi Pascal 圖檔搜尋程式碼

```
//建立一函式 Get_Design_File ，第一個輸入參數為搜尋類別，第二個輸入參數為型號，而
//系統回傳的參數以字串為主，其主因為字串型態於 Web Service 中較無型態轉換之問題。
function TProduct_Data.Get_Design_File(const Search_Type, pno: WideString): WideString;
//目錄路徑
var dirpath:string;
//搜尋紀錄
var sr:TSearchRec;
//暫存名稱
var tempname:string;
//字串陣列
var DFArray:array[0..13] of string;
//整數 i
var i:integer;
begin
    //如果搜尋類別為 productor
    if Search_type='productor' then
        begin
            //指定圖檔目錄位置
            dirpath:=ExpandFilename('D:\flinger\images\productor\'+'PNO');
            //查看目錄是否存在，如果不存在則新增此目錄
            if not DirectoryExists(dirpath) then
                CreateDir(dirpath);
            //利用迴圈的方式，找尋此目錄內的所有設計圖檔
            dirpath:=dirpath+'*. *';
            if FindFirst(dirpath,faArchive,sr)=0 then
                begin
                    DFArray[0]:= "";
                    repeat
                        tempname:=copy(sr.name,pos('\',sr.name)+1,pos('.',sr.name)-1);
                        if tempname=pno+'_s' then
                            DFArray[1]:=sr.Name
```

```
    else if tempname=pno+'_c2' then
        DFArray[2]:=sr.Name
    else if tempname=pno+'_c3' then
        DFArray[3]:=sr.Name
    else if tempname=pno then
        DFArray[4]:=sr.Name
    else if tempname=pno+'_2' then
        DFArray[5]:=sr.Name
    else if tempname=pno+'_3' then
        DFArray[6]:=sr.Name
    else if tempname=pno+'_4' then
        DFArray[7]:=sr.Name
    else if tempname='流程圖' then
        DFArray[8]:=sr.Name
    else if tempname='組立流程圖' then
        DFArray[9]:=sr.Name
    else if tempname=pno+'-Model' then
        DFArray[10]:=sr.Name
    else if tempname='process' then
        DFArray[11]:=sr.Name
    else if tempname=pno+'(demo)' then
        DFArray[12]:=sr.Name
    else if tempname=pno+'_nc' then
        DFArray[13]:=sr.Name;
    until FindNext(sr)<>0;
    FindClose(sr);
end;
end;
//將陣列所儲存之結果串聯為一字串型態，並回傳給前端
result:=DFArray[0];
for i:=1 to 13 do
    begin
        result:=result+'#'+DFArray[i];
    end;
end;
```

ASP 圖檔搜尋程式碼

```
<%  
'設立圖檔目錄  
  dirtowalk="images/productor/"&request("pno")&"/"  
'取得目錄的絕對路徑  
  thepath=server.mappath(dirtowalk)  
'建立 FileSystemObject 物件  
  Set fs = CreateObject("Scripting.FileSystemObject")  
'判斷目錄是否存在，若不存在的話則新建目錄  
  if not fs.folderexists(thepath) then  
    fs.createfolder(thepath)  
  end if  
'取得目錄與檔案  
  Set f = fs.GetFolder(server.mappath(dirtowalk))  
  Set fc = f.Files  
'建立陣列  
  Dim picfiles(15)  
'利用迴圈的方式搜尋於此目錄內的所有設計圖檔，並將搜尋的結果以圖檔檔名儲存於陣列中  
  for each whatever in fc  
    filename=left(whatever.name,instr(whatever.name,".")-1) '取得檔名  
    select case filename  
    case request("pno")&"_s"  
      picfiles(1)=whatever.name  
    case request("pno")&"_c2"  
      picfiles(2)=whatever.name  
    case request("pno")&"_c3"  
      picfiles(3)=whatever.name  
    case request("pno")  
      picfiles(4)=whatever.name  
    case request("pno")&"_2"  
      picfiles(5)=whatever.name  
    case request("pno")&"_3"  
      picfiles(6)=whatever.name  
    case request("pno")&"_4"
```

```

picfiles(7)=whatever.name
case "流程圖"
picfiles(8)=whatever.name
case "組立流程圖"
picfiles(9)=whatever.name
case request("pno")&"-Model"
picfiles(10)=whatever.name
case "process"
picfiles(11)=whatever.name
case request("pno")&"(demo)"
picfiles(12)=whatever.name
case request("pno")&"_nc"
picfiles(13)=whatever.name
end select
next
‘釋放 FileSystemObject 物件
set fc=nothing
set f=nothing
set fs=nothing
%>

```

三、Web Service 轉換圖檔搜尋資訊

介面功能定義
function Get_Design_File(const Search_Type, pno: String):WideString; safecall;
介面功能實作
<pre> //輸入參數 1：搜尋類別，輸入參數 2：產品型號 function TWS_Product_Data.Get_Design_File(const Search_Type, pno:String):WideString; safecall; var //定義物件 obj : Variant; begin CoInitialize(nil); try </pre>

```

//建立 COM+物件，呼叫 Get_Design_File 方法，回傳搜尋結果
obj :=CreateOleObject('P_Product_Data.Product_Data');
result:=obj.Get_Design_File(Search_Type,pno);
finally
CoUninitialize;
end;
end;

```

Delphi6 所轉換之 WSDL 結果

```

<?xml version="1.0" ?>
- <definitions xmlns="http://schemas.xmlsoap.org/wsdl/"
xmlns:xs="http://www.w3.org/2001/XMLSchema" name="IWS_Product_Dataservice"
targetNamespace="http://tempuri.org/" xmlns:tns="http://tempuri.org/"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/">
+- <message xmlns="http://schemas.xmlsoap.org/wsdl/" name="Get_Design_FileRequest">
<part xmlns="http://schemas.xmlsoap.org/wsdl/" name="Search_Type" type="xs:string" />
<part xmlns="http://schemas.xmlsoap.org/wsdl/" name="pno" type="xs:string" />
</message>
- <message xmlns="http://schemas.xmlsoap.org/wsdl/" name="Get_Design_FileResponse">
<part xmlns="http://schemas.xmlsoap.org/wsdl/" name="return" type="xs:string" />
</message>
- <portType xmlns="http://schemas.xmlsoap.org/wsdl/" name="IWS_Product_Data">
- <operation xmlns="http://schemas.xmlsoap.org/wsdl/" name="Get_Design_File">
<input message="tns:Get_Design_FileRequest" />
<output message="tns:Get_Design_FileResponse" />
</operation>
</portType>
- <binding xmlns="http://schemas.xmlsoap.org/wsdl/" name="IWS_Product_Databinding"
type="tns:IWS_Product_Data">
<soap:binding style="rpc" transport="http://schemas.xmlsoap.org/soap/http" />
- <operation xmlns="http://schemas.xmlsoap.org/wsdl/" name="Get_Design_File">
<soap:operation
soapAction="urn:WS_Product_DataIntf-IWS_Product_Data#Get_Design_File" style="rpc"
/>
- <input>
<soap:body use="encoded" encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="urn:WS_Product_DataIntf-IWS_Product_Data" />

```

```

</input>
- <output>
  <soap:body use="encoded" encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
  namespace="urn:WS_Product_DataIntf-IWS_Product_Data" />
</output>
</operation>
</binding>
- <service xmlns="http://schemas.xmlsoap.org/wsdl/" name="IWS_Product_Dataservice">
- <port xmlns="http://schemas.xmlsoap.org/wsdl/" name="IWS_Product_DataPort"
  binding="tns:IWS_Product_Databinding">
  <soap:address
  location="http://127.0.0.1:8008/flinger/component/WS/P_WS_Product_Data.dll/soap/
  IWS_Product_Data" />
</port>
</service>
</definitions>

```

四、圖檔上傳機制與程式碼

本研究係採用「upload_5xsoft」元件處理圖檔上傳，由於現階段設計圖檔存放於 Web 伺服器上，因此可透過 ASP 程式與上傳元件相互配合，達到設計圖檔管理的目的。

前端使用者上傳介面程式碼
<pre> <form name="f" method="post" action="handledata.asp?pno=<%=request("pno")%>" enctype="multipart/form-data"> <table width="534" border="0" cellspacing="1" cellpadding="1" bgcolor="#FFFFFF" id="pictable" align="center"> <tr> <td bgcolor="#669900">圖檔類別</td> <td bgcolor="#669900" align="center">新增或覆寫</td> </tr> <tr> <td> 草稿 </td> <td> <input type="file" name="file1" size="51" > </td> </tr> <tr> <td> 2D 圖 </td> <td> <input type="file" name="file2" size="51" > </td> </tr> <tr> <td> 3D 圖 </td> <td> <input type="file" name="file3" size="51" > </td> </tr> <tr> <td> 詳圖 </td> </pre>

```

    <td> <input type="file" name="file4" size="51" > </td> </tr>
<tr> <td> 簡圖 </td>
    <td> <input type="file" name="file5" size="51" > </td> </tr>
<tr> <td> 組立圖 </td>
    <td> <input type="file" name="file6" size="51" > </td> </tr>
<tr> <td> 總圖 </td>
    <td> <input type="file" name="file7" size="51" > </td> </tr>
<tr> <td> 流程圖 </td>
    <td> <input type="file" name="file8" size="51" > </td> </tr>
<tr> <td> 組立流程圖 </td>
    <td> <input type="file" name="file9" size="51" > </td> </tr>
<tr> <td> 2D 平面圖(線上展示) </td>
    <td> <input type="file" name="file10" size="51" > </td> </tr>
<tr> <td> 流程圖(線上展示) </td>
    <td> <input type="file" name="file11" size="51" > </td> </tr>
<tr> <td> 展示圖 </td>
    <td> <input type="file" name="file12" size="51" > </td> </tr>
<tr> <td> NC 檔 </td>
    <td> <input type="file" name="file13" size="51" > </td> </tr>
</table> <p align="center"> <input type="submit" name="Submit" value="上傳圖檔"> </p>
</form>

```

後端圖檔上傳處理程式碼

```

納入 upload_5xsoft 上傳元件檔案
<!--#include FILE="upload.asp"-->
<%
‘建立 upload_5xsoft 物件
    set upload = new upload_5xsoft
‘取得產品型號
    pno=request.querystring("pno")
‘取得產品型號目錄路徑
    dirtowalk="images/productor/"&pno&"/"
    thepath=server.mappath(dirtowalk)

```

‘取得產品目錄路徑

```
dirtowalk2="images/productor/" '目錄  
thepath2=server.mappath(dirtowalk)
```

‘圖檔上傳處理

```
for each formName in upload.objFile  
set file=upload.objFile(formName)
```

‘如果所上傳的圖檔為產品型錄圖檔的話，則將圖檔放置於產品目錄底下

```
if formname="webpic" then  
if file.FileSize>0 then  
file.SaveAs Server.mappath(dirtowalk2&file.FileName)  
webpicname=file.FileName
```

```
end if
```

‘如果所上傳的圖檔為產品展示圖檔的話，則將圖檔放置於產品目錄底下

```
elseif formname="file12" then  
if file.FileSize>0 then  
file.SaveAs Server.mappath(dirtowalk2&file.FileName)  
webdemopicname=file.FileName  
end if
```

‘其餘圖檔全部上傳至產品根目錄底下

```
else  
if file.FileSize>0 then  
file.SaveAs Server.mappath(dirtowalk&file.FileName)  
end if  
end if
```

```
next
```

‘釋放 upload_5xsoft 物件

```
set file=nothing
```

```
%>
```

五、圖檔刪除機制與程式碼

‘建立 Scripting.FileSystemObject 物件

```
Set fs = CreateObject("Scripting.FileSystemObject")
```

‘取得圖檔檔名(先前程式以儲存於 session 變數中)

```
filename=split(session("tempfilename"),",")
```

‘取得目錄路徑

```
dirtowalk="images/productor/"&pno&"/"  
thepath=server.mappath(dirtowalk)
```

‘利用迴圈的方式判斷前端使用者是否有勾選要刪除圖檔

```
for i=1 to 13
```

‘如果有勾選要刪除圖檔則先判斷此圖檔存在於否，再進行圖檔刪除的動作

```
if upload.Form("dfile"&i)="1" then  
    if fs.FileExists(thepath&"\"&filename(i)) then  
        fs.deletefile thepath&"\"&filename(i),false  
    end if  
end if  
next
```

六、線上瀏覽機制與程式碼

線上 2D 瀏覽主要是利用 Autodesk 所提供的免費軟體 Express Viewer，一旦安裝完成之後，系統及將自動註冊線上瀏覽 DWF 檔格式的物件，於網頁中呼叫與使用此物件的方法如下：

```
<html>  
<head>  
<title>線上瀏覽</title>  
<meta http-equiv="Content-Type" content="text/html; charset=big5">  
</head>  
<body bgcolor="#FFFFFF" text="#000000" leftmargin="0" topmargin="0" marginwidth="0"  
marginheight="0">  
<div align="center">  
<!-- 呼叫 Express Viewer 物件，並設定長寬數值 -->  
<object id = "viewer" classid = "clsid:A662DA7E-CCB7-4743-B71A-D817F6D575DF"  
width="1000" height="750">  
<!-- 傳入 DWF 檔路徑，Express Viewer 物件即顯示 2D 設計圖檔內容 -->  
    <param name = "Src" value="<%=request("path")%>">  
</object>  
</div>  
</body>  
</html>
```

簡歷

姓名：蔡智銘

出生：民國六十七年一月二十八日

籍貫：台灣省基隆市

學歷：私立東海大學工業設計學系碩士

私立東海大學工業工程學系學士

省立基隆高級中學

經歷：東海大學企業講座-電子商務課程助教

穩正企業產學合作案-線上銷售服務系統主要負責人

九十一學年度研究所班級副代表

國科會計畫「企業產品整合開發資訊控管系統之研究」主要研究人員

國科會計畫「企業設計鍊之整合控管系統之研究」主要研究人員



兩年內主要學術發表及研究所參與事務

研究所期間：九十學年度 九十二學年度，歷時兩年

一、學術發表論文

- [1] 郭炳宏、蔡智銘，企業網路銷售管理系統模式之建構-以台灣中小企業為例，2001年科技與學術研討會，2001.10。
- [2] 蔡智銘、郭炳宏，彈性管理於設計鏈系統整合之應用，中華民國設計學會第七屆設計學術研究成果論文集，pp.731~736，2002.05。
- [3] 郭炳宏、蔡智銘，企業產品整合開發資訊控管系統之研究，行政院國家科學委員會專題研究計畫成果報告，計畫編號：NSC 90-2218-E-029-007，2002.10。
- [4] 蔡智銘，企業產品資訊服務控管系統之研究，中國工業工程學會九十一年度學生論文競賽，2002.10。
- [5] 郭炳宏、蔡智銘，企業設計鍊之整合控管系統之研究(1/3)，行政院國家科學委員會專題研究計畫期中報告，計畫編號：NSC 91-2213-E-029-007，2003.05。
- [6] 郭炳宏、蔡智銘，建構網路服務式產品資料管理系統之研究，2003年產業電子化運籌管理學術暨實務研討會，2003.05。
- [7] 蔡智銘，建構網路服務式產品資料管理系統之研究，東海大學工業設計學系碩士論文，2003.06。

指導教授：東海大學工業設計學系副教授 郭炳宏老師

口試委員：東海大學工業工程與經營資訊學系教授 張炳騰老師

成功大學工業設計學系副教授 林銘泉老師

明道管理學院資訊管理學系主任暨副教授 吳銜容老師

二、研究所參與事務

- 參與穩正企業產學合作計畫，主要負責「線上銷售服務系統」之研究與開發。
- 負責建置開發企業產品商務系統(Enterprice Product Commence：EPC，鄭再添)之E-Service子系統，2000-2001。
- 主要協助與建置產品開發知識案例式推理系統(The Product Development Knowledge Using Case-Based Reasoning，黃弘毅)，2002。

- 負責國科會計畫「企業產品整合開發資訊控管系統之研究」主要研究與事務。
- 負責國科會計畫「企業設計鍊之整合控管系統之研究(1/3)」主要研究與事務。
- 協調規劃產品開發設計研究室定期發表與相關事務。
- 積極參與組內研究討論，與各研究成員相互研討與學習。
- 負責擔任系上電腦教室管理及行政單位事務性工作。
- 協助瑞振企業開發產品資料管理系統並提供 e 化改良建議報告。