

國立交通大學

資訊管理研究所

博士論文

一種以代理人為基礎支援客戶關係管理的即時整合
方法設計



An Agent-Based Design for Real-Time Integration of
Customer Relationship Management
Development Process

研究生：葉文宏

指導教授：黃景彰 博士

中華民國九十五年七月

一種以代理人為基礎支援客戶關係管理的即時整合方法設計

An Agent-Based Design for Real-Time Integration of Customer
Relationship Management Development Process

研究生：葉文宏

Student : Yeh, Wen-Hung

指導教授：黃景彰

Advisor : Hwang, Jing-Jang

國立交通大學

資訊管理研究所

博士論文



Submitted to Institute of Information Management

College of Management

National Chiao Tung University

in Partial Fulfillment of the Requirements

for the Degree of

Doctor of Philosophy

in

Information Management

July 2006

Hsinchu, Taiwan, Republic of China

中華民國九十五年七月

一種以代理人為基礎支援客戶關係管理的即時整合方法

學生：葉文宏

指導教授：黃景彰

國立交通大學資訊管理研究所

中文摘要

本論文主要在研究如何利用企業內部已經存在的資訊系統，滿足企業在運行客戶關係管理時的需求。我們提出一個方法，運用軟體代理人的技術，整合存在於相關資訊系統內部的資訊與流程，快速提供即時、整合的資訊和服務給客戶，滿足客戶的需求。



我們的方法，包含三個主要的步驟，作為執行時的參考。首先，企業必須先定義哪些是客戶真正需要的服務，接著，要分析有哪些相關的資訊系統必須參與，如何參與，共同來支援這些服務，最後，需要制定資訊系統服務介面來做為資訊系統之間彼此整合的基礎。在執行上，我們提出以軟體代理人的方式來實踐資訊系統服務介面。

透過中華電信公司的案例，我們總共設計了六種不同類型的軟體代理人，來實現我們所提出的方法，包括(1)服務提供者代理人(Service-Provider Agent, SPA)，用來提供資訊服務或程序服務，(2)服務請求者代理人(Service-Request Agent, SRA)，用來請求資訊服務或程序服務，(3)資訊發佈代理人(Information Publish Agent, IPA)，提供即時資訊主動通知的服務，(4)資訊訂閱代理人

(Information Subscribe Agent, ISA)，接受資訊發佈代理人的資訊服務，(5) 資料整合代理人(Data Consolidation Agent, DCA)，用來整合不同來源的資訊服務，以及(6)流程控制代理人(Process Control Agent, PCA)，用來整合和控制不同來源的程序服務。代理人之間透過 XML 格式的訊息來相互溝通，只需花費些許的功夫和時間，就可以迅速達到整合資訊系統的目的。此外，我們還設計了資訊安全代理人來保護未授權資料的存取。

多數的企業都像中華電信一般，為了每天的運作，必須投入許多資源在設計、開發、以及維護眾多的資訊系統上。本論文提出了一個架構和方法，滿足企業實際的需要，引導企業充分利用企業內現有的資訊系統，來加強客戶服務。本論文所實作的案例，雖然以中華電信內部的應用為基礎，但我們所提出的架構和方法，也可以適用在其他企業的應用。

關鍵詞：客戶關係管理，企業應用整合，軟體代理人，即時整合，延伸式標記語

言

An Agent-Based Design for Real-Time Integration of Customer Relationship Management Development Process

Student : Yeh, Wen-Hung

Advisor : Hwang, Jing-Jang

Institute of Information Management
National Chiao Tung University

ABSTRACT

This thesis focuses on how to leverage the existing information systems to meet the needs for customer relationship management. The author proposes a method for integrating the existing business processes, which are identified according to the needs to satisfy the customer. The method utilizes a programming technique, known as software agent, to meet the needs for providing the desired information in a real-time manner.

The method includes three main tasks: (1) defining the services to satisfy the customer, (2) identifying the business processes that can provide information to support the service, and (3) defining the interfaces for integrating the information systems associated with the identified processes. The interfaces are implemented with the technique of software agent.

In one implementation at Chunghwa Telecommunication Co., six types of

software agent have been designed: (1) Service Requestor, (2) Service Provider, (3) Information Subscriber, (4) Information Publisher, (5) Data Consolidator, and (6) Process Controller. The input/output data of these six types of software agent are defined according to the syntax of Extensible Markup Language (XML) such that the integration of the identified information systems can be achieved with less costly effort. In addition, an agent for information security is further included to prevent the information from unauthorized access.

Like Chunghwa Telecommunication Co., most organizations have spent tremendous amount of resources on designing, developing and operating a variety of information systems for their daily businesses. It is a practical need to leverage the legacy systems for customer relationship management. This thesis has devised a method and framework for this purpose. The implementation described is exemplary; applications of the method and framework are not restricted to Chunghwa Telecommunication Co.

Keywords : Customer Relationship Management(CRM), Enterprise Applications Integration(EAI), Software Agent, Real-time Integration, Extensible Markup Language (XML)

誌 謝

八年的博士班生涯，終於將結束。首先要感謝我的指導教授黃景彰老師，在過去這兩千九百多個日子裏，不斷地給我指導和鼓勵，尤其當我的研究工作陷入低潮時，老師仍不停地給我打氣，給我信心，幫助我順利完成論文。從老師身上，我學習到嚴謹的治學精神和研究態度，也學習到許多做人處事的方法，受益匪淺。

也非常感謝口試委員吳壽山教授、陳安斌教授、劉敦仁教授、鍾青萍教授、陳鴻基教授對論文的詳細審閱與費心指導，讓我的博士論文得以更加完善。

能來唸博士班，我非常感謝嚴劍琴小姐和詹凱統先生，因為他們的提攜及鼓勵，讓我在工作之餘，還能有一個進修的機會，當然，也要感謝陪我走過這些年來的同事，特別是過去在中華電信研究所 860 專案的同事和主管，在我實踐理論的過程中，提供了許多寶貴的意見和協助。

還要感謝這些日子以來，一起分享、相互鼓勵的夥伴，劉興華學長、廖耕億學長、李鍾斌學長、林之寅學長、敏華、慈章、曉芸、怡鎮，還有所上的助理淑惠，及其他已經畢業的學長姐、學弟妹，讓我的博士班生活留下許多美好回憶。

最後要感謝我的爸媽，我的太太秀娟，還有我那可愛的兒子宥廷，你們的支持，是我最大的精神支柱，也將此篇論文獻給我最摯愛的家人，與你們共同分享這喜悅的一刻。

目 錄

中文摘要.....	i
ABSTRACT.....	iii
誌 謝.....	v
圖目錄.....	viii
表目錄.....	ix
第 1 章、緒論.....	1
1.1. 研究動機.....	1
1.2. 研究目的.....	3
1.3. 研究方法.....	5
1.4. 研究範圍與限制.....	6
1.5. 論文章節概述.....	6
第 2 章、輔助客戶關係管理的資訊技術探討.....	8
2.1. 資訊科技的重要性.....	8
2.2. 客服中心相關技術.....	11
2.3. 知識管理系統.....	14
2.4. 資料倉儲與資料探勘技術.....	17
2.5. 資料與流程的即時整合議題.....	21
第 3 章、以軟體代理人為基礎的即時整合方法設計.....	24
3.1. 整合方法的演進.....	25
3.2. 軟體代理人技術探討.....	27
3.3. 以軟體代理人為基礎的即時整合方法設計.....	29
第 4 章、實例探討.....	38
4.1. 背景介紹.....	38
4.2. 定義服務.....	39
4.3. 系統分析.....	40

4.4. 制定資訊系統服務介面.....	43
4.5. 服務介面的再利用.....	49
4.6. 服務介面設計分析.....	51
第 5 章、資料保護機制的設計.....	54
5.1. 資訊訂閱代理人的註冊程序.....	54
5.2. 提昇資訊系統安全稽核能力.....	55
5.3. 資訊安全代理人的設計.....	55
第 6 章、結論和未來的研究方向.....	59
6.1. 研究貢獻.....	59
6.2. 未來研究方向.....	61
參 考 文 獻.....	62
附錄、資訊系統服務介面設計參考資料.....	69



圖目錄

圖 2-1、知識螺旋.....	15
圖 2-2、五種不同的資料倉儲架構.....	19
圖 2-3、TOP 10 Technology & Management Priorities.....	22
圖 3-1、企業資源規劃應用系統的集中式資料庫架構.....	25
圖 3-2、以客戶服務為核心的模型.....	32
圖 3-3、資料整合代理人架構圖.....	36
圖 3-4、流程控制代理人架構圖.....	37
圖 4-1、支援大客戶服務中心服務的系統關係圖.....	43
圖 4-2、提供『查詢障礙處理進度』服務的各資訊系統代理人關係圖.....	45
圖 4-3、流程控制代理人與其它代理人之間的關係圖.....	48
圖 4-4、障礙處理資訊系統与其它資訊系統之間的關係圖.....	49
圖 5-1、輸出資訊過濾程序.....	58

表目錄

表 2-1、客戶關係管理成功的五項要素，以及資訊技術如何協助.....	10
表 3-1、不同類型的企業應用整合拓撲架構.....	26
表 3-2、不同屬性的軟體代理人.....	33
表 4-1、流程控制資料庫範例.....	48
表 4-2、適用各種整合的拓撲架構的設計說明.....	52
表 5-1、資訊發佈列表.....	55
表 5-2、資訊安全政策資料庫設計：系統名稱和安全性等級.....	57
表 5-3、資訊安全政策資料庫設計：安全性等級和被限制存取的資訊.....	57
表 5-4、資訊安全代理人的訊息處理範例.....	58
表A- 1、請求『障礙申訴』服務的合約內容.....	69
表A- 2、『障礙申訴』服務回應的合約內容.....	70
表A- 3、請求『查詢障礙處理進度』服務的合約內容.....	72
表A- 4、『查詢障礙處理進度』服務回應的合約內容.....	73
表A- 5、發佈『電路服務狀態』服務的合約內容.....	76
表A- 6、發佈『資源調整作業資訊』服務的合約內容.....	78
表A- 7、請求『擷取用戶障礙資訊』服務的合約內容.....	80
表A- 8、『擷取用戶障礙資訊』服務回應的合約內容.....	82
表A- 9、請求『障礙處理回報』服務的合約內容.....	86
表A- 10、『用戶障礙資訊發佈』服務的合約內容.....	88

第 1 章、緒論

1.1. 研究動機

客戶不僅僅是企業最重要的資產，也是企業生存的依據，沒有客戶，企業就失去了它存在的意義。Reichheld 和 Teal(1996)在他們的著作中明白指出，開發一個新的客戶所需要的成本，比留住一個現有的客戶還要高出五倍，客戶的忠誠度，才是決定企業能否永續經營的關鍵。的確，當一個客戶已經對一個企業感到失望時，企業所流失的不單單只是一個客戶而已，而是更多更多，因為，這一個客戶，會把他不好的經驗擴散出去，告訴他所有的親朋好友，一傳十，十傳百，對企業造成難以估計的傷害。Reichheld, F. F. (2004) 在他後續的研究中也提出了『詆毀者』與『宣揚者』對企業的影響。從他們的案例分析，美國線上(American On-Line, AOL)在 2002 年時，其顧客中有 42%為『詆毀者』，『宣揚者』只有 32%，導致其新客戶(其中多是衝著免費試用促銷活動而來)增加的速度趕不上客戶流失的速度，而發生經營危機。相對的，MSN 為提升服務，投入資金研發，改善功能，到 2003 年時，MSN 的『宣揚者』達到顧客總數的 41%，『詆毀者』比例則為 32%，是 MSN 得以繼續成長的主因，這說明了『詆毀者』口訊傳播對企業的損傷效果。因此，如何客戶提供更好的服務，創造更高的價值，把客戶轉變成宣揚者，建立客戶與企業雙贏共榮的關係，是企業在經營客戶關係管理時，必須重視的一個問題。

隨著技術的演進，企業與客戶接觸的管道及方式，變得更加多元化。從過去，客戶必須親自到企業所設立的營業櫃檯，面對面接觸，才能得到企業所提供的服務，演變到後來，客服中心的概念成型，企業紛紛設立專職的客服人員，透過電話來接受客戶的抱怨，並提供各種諮詢的服務。隨著資訊技術的進步，發展出了互動式的語音回覆系統（Interactive Voice Response, IVR），提供客戶自助服務的管道，最近幾年，隨著網際網路的發展，企業開始可以透過網站(Web)及電子郵件(e-mail)來接觸客戶，與客戶進行雙向的互動。甚至，企業會有所謂的客戶代表，主動去接觸客戶。這種接觸管道多元化的轉變，迫使企業必須思考如何在各種不同類型的接觸管道，都能提供良好的服務給客戶。而其關鍵就在於，如何將前端的接觸管道與後端的資料和流程，作良好的整合。



想像下列的情境：某電信公司的客戶葉先生因為沒有繳費，所以被停話了，家裡電話無法使用，同時也無法使用 ADSL 上網，於是，葉先生就親自到該電信公司的營業櫃檯繳費，同時也詢問櫃檯人員，何時可以復話？何時可以使用 ADSL 上網？如果客戶得到的答案是馬上就可以復話，回到家就可以使用 ADSL 上網，相信，客戶一定會很高興，對這麼快的作業效率感到滿意。相反的，如果客戶得到的答案是一天後才可以復話，三天後才可以使用 ADSL 上網，而該電信公司的競爭對手卻只要四小時就可以復話，和使用 ADSL 上網，可以想見的是，客戶一定開始抱怨，對這樣的效率感到不滿和無法理解，甚至有可能轉而投向該電信公司的競爭對手，而且變成該電信公司『詆毀者』。而這問題的背後，代表著一連串作業流

程的即時整合。企業要能夠敏捷的，迅速的反應客戶需求，提高競爭力，就必須提昇流程的效率，縮短反應時間。

再想像另一個情境：當葉先生到該電信公司的營業櫃檯繳完費，櫃檯人員說馬上就可以復話，而且回到家就可以使用 ADSL 上網，可是葉先生回到家以後，卻發現家裡電話還是無法使用，也無法使用 ADSL 上網，於是葉先生就打電話到該電信公司的客服中心抱怨。如果這時候，客服人員的回答是：『抱歉，葉先生您尚未繳費。』可以想見的是，葉先生一定非常生氣，明明才剛到該電信公司的營業櫃檯繳完費，現在電信公司卻不承認。相反的，如果客服人員的回答是：『抱歉，葉先生，為了進行 ADSL 升速，您家附近正在進行管線抽換的工程，預計下午六點可以施工完畢，對不起，造成您的不便。』相信這樣的說明，葉先生比較可以接受，心情也會比較平和。而這問題其實就突顯出即時整合各種資料的重要性，客服人員在接觸客戶的同時，就必須能夠得到即時完整的客戶資訊，例如知道客戶已經完成繳費，知道管線抽換工程的進行會影響到該用戶，才能提供客戶滿意的答案。

因此，如何有效地將企業內部的資訊及作業流程即時整合起來，提供給客戶最滿意的服務，讓客戶成為『宣揚者』，而不是『詆毀者』，是非常值得研究的一個問題。

1.2. 研究目的

企業在發展的過程中，在不同的時空背景，根據當時的需求，而發展出符合

當時需求的資訊系統。這些資訊系統，由不同的部門或不同的分公司所管理維護，擁有部分的客戶資訊，也各自負責部分的交易處理流程，可以提供給客戶某些特定的服務。換句話說，客戶資訊是破碎的，散落在企業內各個相關的資訊系統之中，而各部門的作業也是獨立的，並未有效整合。

很多時候，客戶的需求常常需要動用多個部門的資源，需要多個部門一起合作才能完成，但是，因為各個部門所使用的資訊系統，並沒有整合起來，使得部門間資訊的傳遞工作，變成人工作業，可能使用傳真的方式、電話通知的方式、或者其它的人工方式，將客戶需求的資料，從一個部門傳遞到另一個部門，再輸入到該部門的資訊系統中。可以想見的是，人工作業將同樣的資料，重複輸入到各資訊系統，不但沒有效率，也增加了輸入錯誤的機率，而且，整個流程也無法有效掌控，沒辦法知道哪些部門已經完成相關工作，花了多少時間，哪些部門還未完成，也沒辦法回答客戶查詢進度的要求。

要整合這些已經存在的資訊系統，無疑是一件難事。特別是，這些資訊系統在開發時，只為了滿足當時的需求而設計，很少考慮到以後如何與其他的系統作整合，因此常自成一封閉的系統。但是，如果不能打破系統的界限，就無法將各自獨立的資訊與流程整合起來，這將是企業在執行客戶關係管理時的一個很大致命傷。

本研究的目的，就是希望從客戶服務的觀點切入，架構在企業內部現有資訊

系統的基礎上，提出一個合適的架構和方法，有效地整合企業內部的資訊和流程，
提昇服務效率及客戶滿意度，滿足企業對即時整合的需求。

1.3. 研究方法

本研究希望能以企業內部現有資訊系統為基礎，探討如何運用資訊技術來進行資料與流程的即時整合，協助企業有效做好客戶關係管理，整個研究的過程中，我們所採用的步驟如下：

- (1) 探討資訊技術在企業執行客戶關係管理策略的過程中，所扮演的角色，探討現有哪些相關資訊技術，已經成功的運用在客戶關係管理上，同時也分析這些資訊技術的應用目的及範圍；
- (2) 探討有哪些觀念和方法是可以應用在即時整合的議題上；
- (3) 從客戶服務的觀點切入，架構在企業內部現有資訊系統的基礎上，以系統化的思考方式，提出一個可以即時整合資料與流程的架構和方法；
- (4) 透過實例驗證，來說明我們提出的架構和方法的可行性。

1.4. 研究範圍與限制

本研究著重在解決客戶關係管理的問題，因此，所提出的即時整合的架構和方法，也是從客戶服務的觀點來切入的，而在實例的設計上，也是以客戶關係管理的問題為主，至於此架構和方法是否能適用於客戶關係管理外的其他領域，並不在我們的討論範圍內。與現行大家所熟知的資訊技術之間的關係，如知識管理、資料倉儲、資料探勘等等，我們所提出的方法並不是、也無法用來取代這些資訊技術，而是用來互補對方之不足，因為我們提出的架構和方法，著重在即時整合，並不適合用在大量資料的分析和處理上。



1.5. 論文章節概述

本論文分為六章，各章的內容分別敘述如下：

第一章、緒論，主要敘述研究動機、研究目的、研究方法、研究範圍與限制，

以及論文章節的概略敘述；

第二章、輔助客戶關係管理的資訊技術探討，探討企業在執行客戶關係管理

的策略時，資訊技術的重要性，以及目前已經被普遍使用的資訊技

術，他們的應用範圍和目的；

第三章、以軟體代理人為基礎的即時整合方法設計，探討即時整合的

觀念及軟體代理人技術，並提出我們所設計的架構和方法；

第四章、實例探討，透過應用實例來說明，如何從客戶服務的觀點切入，一步一步將相關的資訊系統透過軟體代理人的方法連接起來，讓彼此之間的資訊和流程得以整合，驗證我們所設計的架構和方法；

第五章、資料保護機制的設計，分享我們在實例驗證的過程中，所遭遇到的有關資訊安全的議題，以及我們如何面對相關問題並提出解決方案；

第六章、結論和未來的研究方向，對本研究內容作一簡單的結論，並提出未來可以繼續深入的研究方向。



第 2 章、輔助客戶關係管理的資訊技術探討

客戶關係管理的本質是一種企業文化，一種策略，讓公司的行銷、服務與作業流程，都能以顧客為核心，能否成功的做好客戶關係管理，絕大部分的原因也取決於企業是否已經擬定正確的客戶策略，以及是否已經建立以客為尊的企業文化。但是，除了企業文化或策略上的思考方式外，如何妥善運用資訊科技，也是很值得思考的一件事，事實上，企業在執行客戶關係管理的過程中，常常需要蒐集大量客戶資訊，並對這些資訊作進一步的分析和運用，這些工作，需要各種資訊技術的協助。不善用資訊技術，企業根本無法整理散佈在各個角落的客戶資料，更遑論把這些客戶資料轉變成有用的知識了。透過資訊科技的協助，企業可以很容易的管理大量客戶資訊，精確地衡量成效，讓執行更有效率。在本章中，我們將探討資訊科技對企業在執行客戶關係管理時的重要性，並探討現有已經被企業所採用的資訊科技。

2.1. 資訊科技的重要性

日本人力資源學院在其所出版的『CRM 戰略執行手冊』一書中，對顧客關係管理下一個定義：『所謂顧客關係管理，就是在顧客和組織的關係上，建立相互培育關懷與尊敬的架構』（楊鴻儒譯，日本人力資源學院著，2001）。並提出七個

步驟及五項要素，來思考推動顧客關係管理方案，七個步驟為：(1) 分析顧客關係管理環境，(2) 建構顧客關係管理願景，(3) 制定顧客關係管理策略，(4) 展開顧客關係管理／企業流程重整，(5) 建置顧客關係管理系統，(6) 運用顧客關係管理資訊，(7) 利用顧客關係管理知識管理來形成完整的執行週期；而五項要素則是：(1) 顧客關係管理相關者，(2) 顧客關係管理通路，(3) 顧客關係管理資訊科技工具，(4) 顧客關係管理一對一資料庫，(5) 顧客關係管理合作關係。其中，包含建置顧客關係管理系統、顧客關係管理資訊科技工具、顧客關係管理一對一資料庫等等，都是需要透過資訊科技的協助，才能夠達成。



Gillies、Rigby 和 Reichheld 等學者(2002)在分析過許多企業執行客戶關係管理的成功和失敗案例後，明白指出，那些成功的客戶關係管理的案例都具備三項要素：第一是建立一個健全的客戶策略，知道那些客戶是必需跟他建立良好關係的，那些客戶是不必需花太多的成本去服務他們；第二是要調整組織，使企業流程能夠與客戶策略相結合，讓客戶策略可以得到最大的支持；最後才是要運用適當的技術或工具來協助，例如亞馬遜書店(Amazon.com)使用協同過濾(Collaborative filtering)的技術，分析具有相同品味的客戶的購買行為，來產生推薦書單。

Rigby、Reichheld 和 Schefter(2002)在探討如何避免掉入客戶關係管理的四個陷阱的研究中，提出了客戶關係管理成功的五項要素，這五項要素包括了：(1)必須獲得正確的客戶資訊、(2)必須雕塑出正確的價值、(3)必須制訂最佳的

流程、(4) 激勵員工、(5) 學習如何留住好的客戶，而資訊技術如何協助企業達成這五項要素，其功能如下表所示。

表 2-1、客戶關係管理成功的五項要素，以及資訊技術如何協助
(本研究整理，資料來源：Rigby, Reichheld, and Schefter 2002)

客戶關係管理成功的五項要素				
獲得正確的客戶	雕塑出正確的價值	制訂最佳的流程	激勵員工	學習如何留住好的客戶
資訊技術如何協助企業達成				
<ul style="list-style-type: none"> ● 幫助分析客戶的貢獻度和成本，找出具有高價值的客戶 ● 提升目標行銷的效果 	<ul style="list-style-type: none"> ● 幫助紀錄產品與服務的相關資訊 ● 建立新的銷售管道 ● 開發新的定價模式 ● 建立社群 	<ul style="list-style-type: none"> ● 幫助加速交易處理流程 ● 提供第一線人員更好的資訊 ● 有效管理庫存及供應鏈 ● 催化協同商務 	<ul style="list-style-type: none"> ● 調整獎勵與制度 ● 建置知識管理系統 	<ul style="list-style-type: none"> ● 幫助追蹤客戶流失與留住的程度 ● 追蹤客戶服務的滿意程度

綜合上述，資訊科技雖然不是客戶關係管理成功的關鍵，但也扮演了一個不可或缺的重要角色(a key enabler)，它能夠幫助企業有效地分析客戶資料，加速處理流程，節省成本和時間，讓企業在執行客戶關係管理時，更有效率。在接下來的章節中，我們將探討有哪些資訊科技已經被運用在處理客戶關係管理的議題上，以及這些資訊科技被應用的方式。

2.2. 客服中心相關技術

隨著科技的進步，企業與客戶接觸的管道及方式，變得更加多元化，從營業櫃檯面對面的接觸方式，到透過電話、傳真來互動，一直到最近隨著網際網路的發展，可以透過網站(Web)及電子郵件(e-mail)來與客戶進行雙向的互動。ARC 遠擎管理顧問公司對客服中心(Call Center)作了一段簡短的定義：『客服中心』為企業中一群專職利用各種溝通管道與顧客接觸，處理顧客問題，雖不以營利為目的，但卻負有協助業務銷售，行銷任務的團隊(吳姝倩，2001)。而 Meta Group 公司更把客服中心定位成客戶接觸中心，或客戶互動中心，認為客服中心是結合多重與客戶接觸的管道，提供包括客戶服務、支援，現場服務派遣，品質管理，智慧型路由，案例式推理(Case-Based Reasoning)及知識庫等在內的重要職責(黃貝玲，2001)。隨著接觸的管道越來越多元化，企業也引進了許多的新技術，這些新技術，包含提供客戶自助的服務，也用來提升客服中心的整體效率，和降低成本，以下我們簡單介紹幾種客服中心技術的演進。

- **電腦電話整合技術 (Computer Telephony Integration, CTI)**，主要是將資訊系統的資料處理能力和即時檢索功能，與電話服務功能整合起來，讓客服人員在接到客戶電話的剎那間，就能得到客戶的基本資訊，了解客戶的過去的行為和習慣，對客服人員而言，可以很快的認識客

戶，對客戶而言，他(她)也有受到尊重的感覺（李瓊芬譯，2000），甚至，若辨別出的重要用戶，還可以轉給特定的客服人員處理。

- **互動式語音服務系統 (Interactive Voice Response, IVR)**，主要是透過語音的導引，利用電話按鍵，引導客戶可以一步一步取得所需資訊，或完成交易。例如，證券公司的語音下單服務，醫院的語音掛號服務，以及銀行的語音轉帳和信用卡開卡都是很普遍的服務應用，當然，這些服務的背後，需要與相關資訊系統整合才行。互動式語音服務系統可以提供客戶自助式服務，自然也減輕的客服人員的工作負擔，可以為企業降低不少人工處理的成本，但是如果語音選單設計不良的話，容易造成客戶失去耐性，降低了服務品質。此外，若搭配**技能基礎來話分配 (Agent Skill-Based Routing)**的技術，就可以依據客戶所選擇的服務內容，將電話轉給具有適當能力的客服人員接聽。
- **自動話務分配 (Automatic Call Distribution)**，主要是能將客戶的來電平均分配給每個客服人員，平衡每個客服人員的工作量，也能夠將客戶的來電分配給有空閒的客服人員，減少客戶的等待時間。
- **電子郵件回應管理系統**，主要是利用人工智慧技術去辨識電子郵件中的關鍵字，然後依照優先處理順序自動轉送給專業的客服人員來回應，甚至可以從企業內部的知識庫中，抓取相關的資訊自動回覆給客戶（黃貝

玲，2001)。

- **網路驅動客服中心(Web-Enabled Call Center, WECC)**，主要將網站與客服人員結合起來的概念，當客戶在瀏覽網站的過程中需要協助，或者需要更進一步的資訊，可以馬上與客服人員聯絡。透過網站的設計，客戶可以透過點選某一個按鈕的方式，就可以馬上與客服人員進行電話的溝通，客服人員也可以與客戶透過網頁同步的方式瀏覽相同網頁，達到線上即時互動的目的。根據分析報告(李瓊芬譯，2000)，建構網路驅動客服中心的主要成效有二，一是網路可以做為客戶取得電話客服中心服務的入門捷徑，二是可以運用網路提高電話客服中心的效能，當電話客服中心的服務提供成本過高時，可以用網路來替代，提升整體服務效益。
- **虛擬客服中心**，是指一部分或全部的服務人員以兼職或外包的方式工作(李瓊芬譯，2000)。這些服務人員，可以在家上班，也可以在某個特定時間上線服務，瑞典鐵路局及是一個很好的虛擬客服中心的案例，瑞典鐵路局的售票人員，在沒有旅客需要購票服務時，就充當電話客服人員，接聽客戶的查詢電話。

客服中心的相關系統，扮演著與客戶接觸的第一線角色，但是他無法單獨存在，必須與後端的資訊系統相結合，才能發揮最大的功效。如果不能與後端的資

訊系統整合起來，那麼就只能處理有限的少量業務，對超出業務外的相關問題，無法立即服務客戶，必須將輾轉至其他第二線，甚至是第三線的人員來處理，隨著等待的時間拉長，客戶的耐心，也隨之消磨殆盡，對企業的滿意程度，自然也隨著消失。相反的，若第一線的接觸，即能夠解決大部分的問題，提供給客戶滿意的服務，相信，不但服務客戶的成本可以下降，客戶的滿意度也會提高。因此，企業如何創造自己的競爭優勢，其關鍵點應在於如何整合企業內部資源，將各個接觸管道與企業內部的資訊及流程緊密結合。

2.3. 知識管理系統



知名的知識管理研究學者野中郁次郎(Nonaka, Ikujiro)率先提出內隱(tacit)知識和外顯(explicit)知識的知識分類方式，所謂內隱知識即是潛藏在人類大腦裡的知識，不容易表達出來，而外顯知識則容易被記錄下來，容易被整理、分類(Nonaka 1991)。同時，他也提出組織知識的創造是一種螺旋的過程，稱為『知識螺旋』，由社會化(Socialization)、外化(Externalization)、系統化(Combination)、內化(Internalization)等四種知識轉換模式不停的循環所組成(Nonaka and Takeuchi 1995)。不管是從組織外部或是從組織內部獲得知識，這些知識逐漸在大腦中累積，就是一種社會化的過程；慢慢地，這些內隱知識透過分析、整理、對話的方式，轉換成一種別人容易理解的型態，例如文字或圖像，成

為外顯知識，即是外化；這些外顯在知識開始在組織內散佈、傳播，不同來源的外顯知識也可以加以整合並形成新的外顯知識，即是系統化的過程；透過外顯知識的實踐，可以獲得新的經驗，形成新的內隱知識，即是內化。下圖顯示知識螺旋四種知識轉換模式的循環情形。

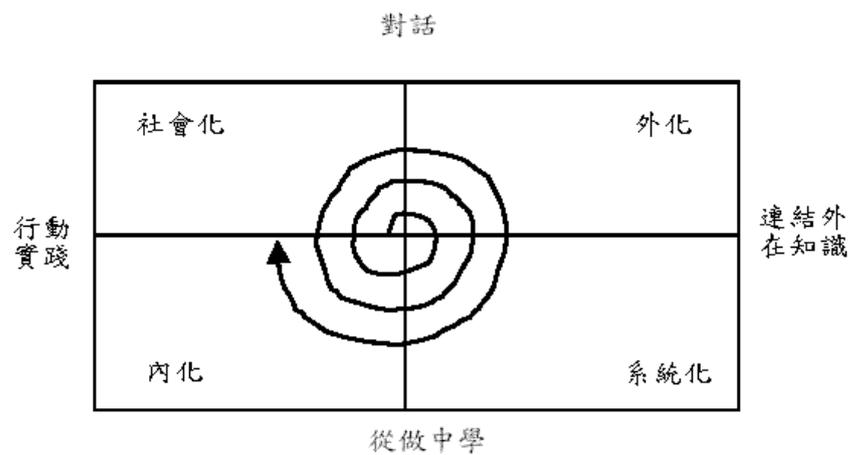


圖 2-1、知識螺旋

(資料來源：Nonaka and Takeuchi 1995)

知識管理的流程，一般可以分為三個程序，即知識的產生(knowledge generation)、分類(knowledge codification)、和傳遞(knowledge transfer/realization)。知識的產生包含了知識的獲得(acquisition)和發展(development)，知識的分類則強調如何把知識轉換成容易存取和應用的形式，而知識的傳遞則專注在如何把知識送到知識使用者的手上，讓需要的人在需要的時候，可以取得他需要的知識(O' Dell and Grayson 1998; Davenport and Prusak 1998; Grover and Davenport 2001)。知識管理系統的發展，較注重外顯知識的

管理，以及知識分類和知識傳遞的程序，希望建立一套系統化的機制，方便外顯知識的儲存、使用、和傳遞。因此，透過知識管理系統，將組織內各個領域專家的外顯知識儲存下來，建立好的使用介面和傳遞機制，也可以應用在客戶關係管理，協助第一線的客服人員，迅速有效地處理客戶問題。

研究學者 Davenport、Long 和 Beers(1998)在分析 24 家公司，31 個知識管理專案後發現，許多專案嘗試用知識管理系統來縮短客戶在電話中等待的時間，或是提供客戶自助服務，來提高客戶滿意度。而惠普公司(Hewlett-Packard, HP)，成功的運用知識管理系統，將客戶來電的數量在兩年內減少了 50%，因此也讓惠普公司得以聘僱較沒有技術背景，以及較沒有經驗的客服人員，減少人力成本。



IBM 公司也有把知識管理運用在客戶關係管理的成功案例(Massey, Montoya-Weiss, and Holcom 2001)。IBM 公司的『Inside IBM』案例，成功地把組織內部的知識資源整合起來，用來提供更好的客戶服務，在解決客戶問題的流程上，得到很顯著的改善，不但提高了客戶滿意度，也促進了組織內部各領域專家的知識交流。

某些探討知識管理系統的學者，是運用過去的個案加以推理，希望能夠解決目前遭遇到的同一類問題，即所謂的 Case-Based Reasoning (Ketler 1993；Watson and Marir 1994)。而這樣的方法，也可以運用在詢問櫃檯。包含了知識取得與知識表現的方法， Chan、Chen 和 Geng(2000)等學者基於 Case-Based

Reasoning 理論基礎，發展出一套自動化的詢問櫃檯系統，幫助客戶及客服人員解決問題，不僅有效縮短處理問題的時間，也縮短新進人員的訓練時間。

我們發現，其實透過知識管理系統的協助，客服人員在解決問題的過程中，會更有效率，而客戶也可以更快得到滿意的答案，對企業而言，可以用更少的員工服務更多的客戶，對客戶而言，問題能被迅速解決，滿意度提高，創造了企業與客戶雙贏的結果。

2.4. 資料倉儲與資料探勘技術



資料倉儲這個概念的出現，遠早於客戶關係管理，可以追溯到一九六〇年代，美國麻省理工學院為了研究計量經濟學而發明了多維度模型 (Multidimensional Modeling)，但一直到一九九〇年代，這個概念才受到重視(蘇隄, 1998)。一般而言，資料倉儲是用來儲存各個部門、不同用處的大量資料，從顧客檔案、產品規格、到地區性資料等，其最有價值部分是提供整合性資料，或者是說提供統一版本的確實資料(a single version of the truth)。

資料倉儲的運用，可以說是一連串的资料處理程序，包含了萃取 (extracting)、轉換(transforming)、和載入(loading)。萃取的程序主要是把資料從各種不同異質性(heterogeneous)的資料來源，把必要的資料抓取出來，轉換的程序主要是把這些從不同來源抓取出來的資料，轉換成資料倉儲資料庫儲存的

格式，轉換的方法可以利用中介資料檔(metadata)來描述，而轉換的過程也要進行資料清洗(cleaning)的程序，將錯誤的資料、不一致的資料剔除，最後才進行載入程序，將清洗後的資料倒入資料倉儲資料庫中，供後續的分析應用。另外，也必須定期執行更新(refresh)的程序，可能一天一次，一週一次，或一個月一次，才能持續蒐集到最新的資料(Chaudhuri and Dayal 1997)。資料倉儲的專家 Inmon 曾說(2002)：『資料倉儲是一種主題導向、整合的、長時間的資料蒐集，提供管理者進行決策的程序。』其中，他特別強調長時間的資料蒐集的重要性，認為長時間去蒐集各個不同來源的不同資料，才有辦法滿足經營者的各種不同面向的決策需求。



在架構的設計上，則大致可以區分成兩種，一是企業資料倉儲(enterprisewide data warehouse)，另一是資料超市(data mart)。資料超市架構設計上，是採取由下而上的設計策略(bottom-up strategy)，他的目標是希望先建立各特定領域、或者是各組織部門的資料倉儲系統，而企業資料倉儲指的便是這些資料超市的聯集，當然，這些資料超市的內容及設計，必須與企業資料倉儲的設計一致。圖 2-2 顯示了五種不同的資料倉儲架構設計(Sen and Sinha 2005)。

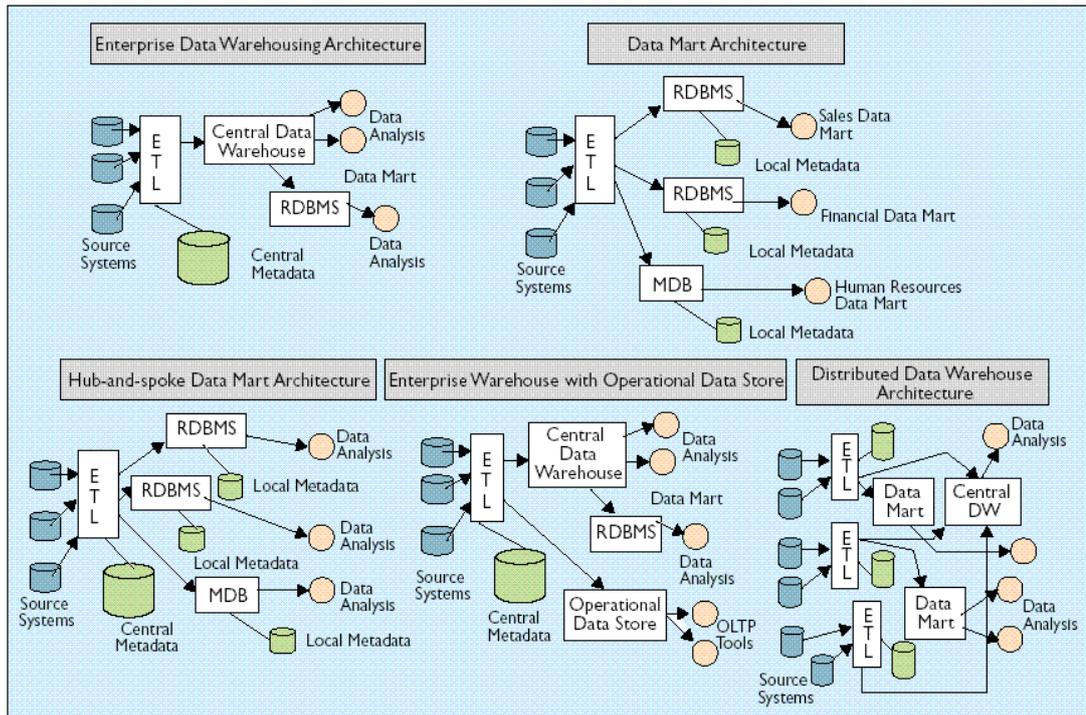


圖 2-2、五種不同的資料倉儲架構

(資料來源：Sen and Sinha 2005)



在使用上，管理者是透過線上分析處理(On-Line Analysis Processing, OLAP)

工具的幫助(Kimball and Ross 2002)，來進行各種不同面向的資料檢視。透過這些分析檢視，企業可以去分析各種商業活動的表現，了解到哪些是貢獻度高的客戶，客戶對哪些產品或服務的反應比較好，評估什麼樣的產品能夠吸引客戶，也可以了解個別客戶的消費行為和嗜好。客戶的歷史行為，有助於企業在進行目標行銷或銷售活動時作為參考，提供客戶適當的產品或服務水準(陳曉開譯，Jill Dyche 著，2003)。而另外一種使用方式，即是運用資料探勘的技術。

資料探勘是一種搜尋與分析資料的程序，用來挖掘隱而未見的、潛在的有用

資訊(Berry & Linoff, 1997)。他包含了對大量資料進行選擇、探測、和模型化的動作，藉以發現先前未知的樣板模型，將之轉化成可以理解的、有價值的新資訊，這對企業了解本身和顧客都非常有助益，有助於分析沒有假設條件、沒有頭緒的資料，並得到意想不到的發現。常見的資料探勘的應用可以歸納為三類：

- **預測(Forecasting/Predictive)**，用歷史資料來預測以後的行為，譬如說，預測模型可以根據客戶以往的購買記錄，預測出客戶下一個最可能購買的產品，或預測出同一族群的潛在客戶；
- **關聯性分析(Association Analysis)**，又經常被稱為採購籃分析，用來檢視哪些產品具有購買上的關連性，或是哪一類的客戶會是某一種產品的目標群，例如男人會同時買尿布和啤酒，藉由這些挖掘出來的關聯性，企業可以針對這些產品或客戶擬訂行銷策略；
- **順序性模式分析(Sequential Pattern Analysis)**，用來找出不同事件之間在某一段時間內的順序性，例如，買電視的人都會在六個月內買錄放影機。

有效的客戶關係管理，需要真正去了解客戶的需求和喜好，量身訂製符合客戶的產品和服務，而不是以大眾化的思考方式去假設每一個客戶都差不多(Shaw, et al. 2001; Peppers and Rogers 1999)。透過資料倉儲及資料探勘工具的幫助，企業能夠從大量的資料中，分析出每一個客戶的特性，發現一些過去沒有察覺的

客戶知識，並轉換成有效的行銷策略。透過對資料的整理、分析、預測，可以提供更明確的資訊，幫助企業更了解客戶，作出更正確的決定，對客戶關係管理有極大的幫助。

2.5. 資料與流程的即時整合議題

從接觸管道的角度切入，需要與後端資訊系統整合，才有辦法提供給客戶良善的服務，當然，這些系統包含了前面所提到資料倉儲系統和知識管理系統。資料倉儲系統或知識管理系統提供了一種資料整合與知識整合的解決方案，但是，它們所強調的都是在提供分析過的資訊，而非即時的資訊。建立資料倉儲的過程中，資料必須從各個資訊系統匯整到單一的資料倉儲，然後才能做徹底的分析，而知識也必須經過萃取、儲存、分享的程序，才能在企業內有效的分享與傳遞。這些程序，可能一天一次，一週一次，或一個月一次，都是週期性的，而不是即時的整合性資料。有些時候，企業在面對客戶時，也希望能取得最即時的客戶資訊，例如，客戶幾分鐘前是不是使用了某一種服務，是不是有任何不愉快的經驗，或是提供給客戶最即時、最完整的服務資訊，類似這樣對資訊即時整合的需求，都不是資料倉儲系統或知識管理系統所設定的目標。

此外，資料倉儲系統或知識管理系統是著眼於資料整合的部份，並沒有考慮流程的問題，而流程的議題，也是企業在執行客戶關係管理時，不能忽視的另一

個關鍵要素。在表 2-1 所列的客戶關係管理成功的五項要素中，除了提供第一線人員更好的資訊外，也需要加速交易處理流程，用最短的時間滿足客戶的需求。

而 CIO 雜誌的調查結果，也顯示了流程整合的重要性。

CIO 雜誌在 2005 年 6 月 16 日到 7 月 8 日之間，對企業的資訊部門主管作了線上問卷調查，在回收的 545 份有效問卷中，製造業佔 14%，醫療業佔 11%，保險業佔 9%，金融業佔 9%，教育佔 9%，零售業佔 8%，政府機關佔 8%。調查的結果顯示，參考圖 2-3，這些資訊部門主管認為系統與流程的整合，是他們心目中列為最優先要處理的技術議題，而透過資訊技術去改善流程，也在管理的議題

中名列第四。



圖 2-3、TOP 10 Technology & Management Priorities
(資料來源：State of the CIO 2006 - Research Reports)

綜合上述，資訊與流程的即時整合，都是企業在執行客戶關係管理時，所必須突破的兩個問題，但是，現有被廣泛應用在客戶關係管理上的資訊技術，並未對此問題提供解決方案，因此，我們必須再思考其它作法，來有效解決企業對即時整合的需求。



第 3 章、以軟體代理人為基礎的即時整合方法設計

如同前面章節所述，企業所面臨的一項很大的挑戰，就是要如何去整合哪些片段的資訊和流程。每個企業，隨著時間的累積，都會搜集、產生、儲存大量的資訊，問題在於，這些資訊並非集中儲存，而是散佈在各組織、各部門不同的資訊系統當中。這每一個資訊系統，都是為了某種目的而誕生，對滿足這些特定目的，具有極大的貢獻與價值。但是，要維護這些系統，卻必須付出昂貴的代價，包括：相同的資料必須重複輸入、資訊沒辦法自動同步更新、系統間資訊的傳遞工作變成人工作業等等，不但沒有效率，增加了輸入錯誤的機率，也嚴重影響到企業的競爭力。想像一下，假如一個企業的銷售系統與訂單系統，沒辦法與生產排程的系統溝通的話，企業就沒辦法很快地回應客戶的需求；或是說，假如銷售系統與行銷系統的資料與財務報表不一致的話，管理者就會因為沒辦法瞭解產品的獲利及客戶的價值，而做出正確的決策。因此，整合的需求應運而生。

以下的內容，將先介紹幾種整合方法的演進，包括企業資源規劃(Enterprise Resource Planning，ERP)、企業應用整合(Enterprise Application Integration，EAI)、及企業流程整合(Business Process Integration，BPI)等方法，然後介紹軟體代理人的觀念及應用情形，最後說明我們利用軟體代理人的觀念所發展出來的即時整合架構和方法。

3.1. 整合方法的演進

企業資源規劃(Enterprise Resource Planning, ERP)的應用系統是最先定位在整合企業內部相關資料與作業流程的系統，它運用集中式的資料庫(如圖 3-1)和標準的作業程序，來達到資料與作業流程整合的目的。企業資源規劃主要是應用在財務、會計、人力資源、物料、後勤支援、銷售、供應鏈等領域，而且企業必須調整他原有的作業方式，來配合企業資源規劃系統所定義標準的作業程序。所以，它並無法涵蓋所有的作業程序，而且，也並非每個企業都能夠調整原有的作業方式，來配合所謂的標準作業程序，例如，戴爾電腦(Dell Computer Corp.)，就宣稱他們非集中式的管理模式，不適合採用企業資源規劃這種集中式的架構(Davenport 1998)。

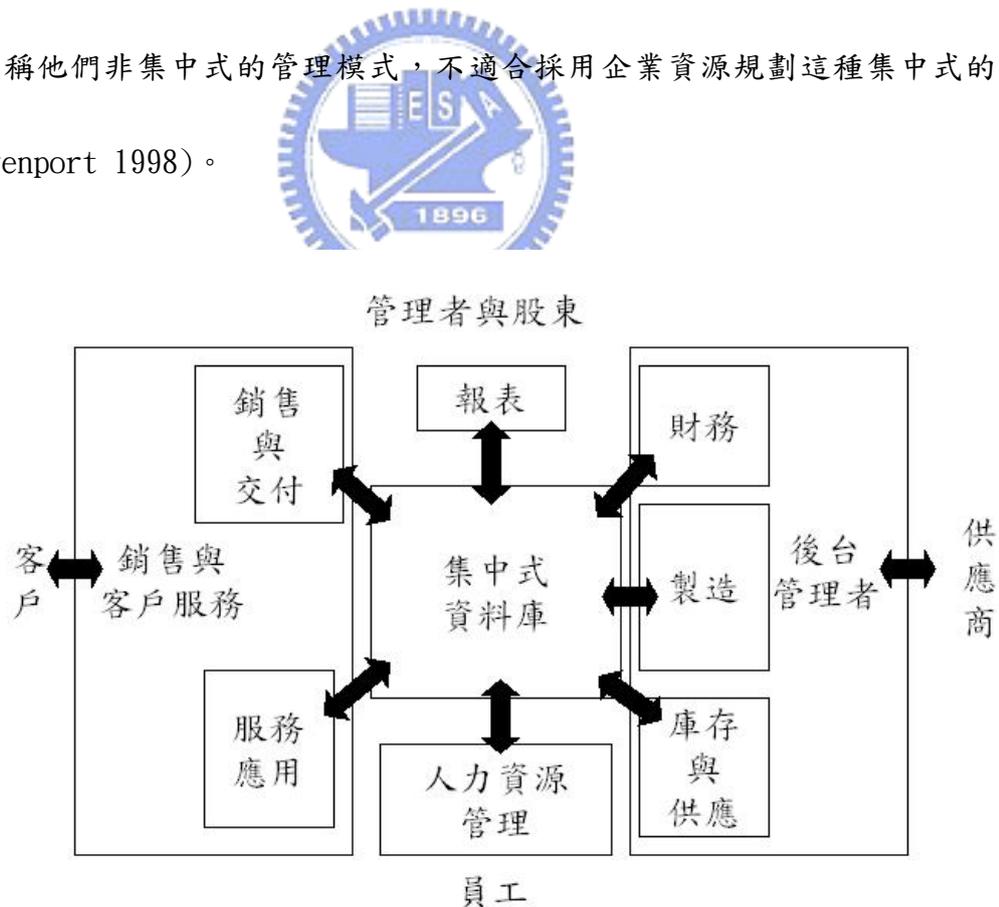


圖 3-1、企業資源規劃應用系統的集中式資料庫架構

(本研究整理，資料來源：Davenport 1998)

在 1990 年代中期，企業應用整合(Enterprise Application Integration, EAI)的觀念開始孕育成型。它的基本觀念是，利用已經存在的系統，以較低的成本和較少的編排來完成整合的目的，當有新的服務或新的應用要開發時，會去思考如何有效地利用這些已經存在的系統(Lee, et al. 2003)。從企業應用整合的角度來看，企業資源規劃的應用系統就相當於是一個『已經存在的系統』，也可以透過企業應用整合的方法來與其它已經存在的系統作整合。從企業應用整合的拓撲架構來看，可以歸納出五種不同類型的拓撲方式，如表 3-1 所示，選擇合適的拓撲架構，有助於降低維護成本與提高整合的效能。



表 3-1、不同類型的企業應用整合拓撲架構
(資料來源： Lee, et al. 2003)

拓撲架構	說明
Hub	星型架構，來源系統整合到某一中心點
Bus	佈告欄模式，來源系統將訊息送到匯流排上
Point-to-Point	應用系統彼此互連
Pipeline	使用先進先出(first in first out)的資訊流
Network	適合非同步的活動和獨立的交易

企業應用整合提供了一種整合的觀念，但並沒有特定的方法，它『利用已經

存在的系統』的想法，給企業帶來一種新的思維。Irani、Themistocleous、和 Love 等學者，在探討企業應用整合對資訊系統生命週期的影響時發現，採用企業應用整合的方法時，軟體被再利用的程度提高了，當資訊系統可以和其他資訊系統整合時，它變成了一個會跟著企業環境不斷成長的個體，隨著環境改變，新服務的推出，資訊系統的價值可以被再利用(Irani, et al. 2003)。

Kobayashi、Tamaki 和 Komoda 等學者，則把工作流程(workflow)與企業應用整合的概念相融合，提出了企業流程整合(Business Process Integration, BPI)的解決方案。他們的方法應用在供應鏈管理系統上，並且具有流程管理的特性，而且，從他們的評估結果，這種融合企業應用整合概念的方法，比從前的方式節省更多人力、時間，同時還提高了品質(Kobayashi, et al. 2003)。

綜合上述，輔以適當的架構和方法，企業應用整合的概念是可以用來解決即時資料整合與流程整合的問題。

3.2. 軟體代理人技術探討

所謂的軟體代理人，其實就是一個被設計出來的、可以執行的程式模組，具有定義清楚的介面規格，能夠解決某一種特定的問題，就好像是某個特定領域的專家一樣，透過制定好的開放介面及通訊協定，和其它的軟體代理人互相溝通、合作，共同解決複雜的問題(Papazoglou 2001； Jennings 2001)。因此，軟體代

理人是一種以目標為導向的設計，為了特殊的目的，而設計特殊的軟體代理人，可以應用的領域也非常廣泛，一般而言，可以分成兩大類型：一類是設計來作為個人助理(personal assistant)，另一類則是互相合作的，共同達成某一目的軟體代理人(Seung Baek, et al. , 1999)。

作為個人助理的軟體代理人，可以根據(或者學習)使用者的嗜好和習性，幫助個人過濾資訊、擷取資訊(Maes 1994)。在電子商務上的應用上，可以幫助買方和賣方在網際網路的環境下自動議價，完成交易(Maes, et al. 1999)。在網路社群的應用上， Hattori 等學者透過軟體代理人的協助，來建構一個社交軟體(socialware)，把真實世界中人與人的社交議題帶到網路上，協助在網路世界中，將具有相同喜好的人連接在一起(Hattori, et al. 1999)； Kautz 等學者透過軟體代理人建構一個『ReferralWeb』，協助人發現已經存在的社會網路(social networks)，以及搜尋自己所信賴的或認識的專家的推薦資料(Kautz, Selman, and Shah 1997)。

Jennings, N. R. (2001)提出了運用多個軟體代理人來發展複雜的、分散式的系統的觀念，並以三種不同功能的軟體代理人：個人通訊代理人(Personal Communication Agent, PCA)、服務提供者代理人(Service Provider Agent, SPA)、網路提供者代理人(Network Provider Agent, NPA)，來說明一個在網際網路上提供虛擬私有網路(Virtual Private Network, VPN)的電信應用。學者 Glushko、Tenenbaum 及 Meltzer (1999)則以 CommerceNet 的 eCO 系統為例，認為用 XML 作

為訊息的主體，是一種把訊息內容編碼(encode)的方法，經過編碼的訊息，軟體代理人可以很容易的從中解譯出必要的資訊，買方的軟體代理人藉此對不同供應商的產品規格及價格進行比較，一但完成交易也可以觸發一連串與供應鏈相關的事件，達到流程的自動化。而使用 XML 作為訊息主體的好處，就在於它的擴充性和無所不在，XML 是網際網路應用中一種新興的標準，為 W3C 標準組織所制定 (<http://www.w3.org/>)，應用在資料交換的方法上，而且被許多主要的廠商接受和支援，如 IBM、Sun、Microsoft 等等。

因此，我們認為可以應用軟體代理人的觀念來做為企業應用整合的架構和方法，並且用 XML 作為軟體代理人之間溝通的訊息主體。在下一節，我們將說明如何運用軟體代理人來作為資訊系統間溝通的橋樑，讓隱身在資訊系統背後的資訊與程序得以互相分享、連接，達到企業內部資訊與流程即時整合的目的。而我們所提的方法，是以客戶服務為核心，藉由客戶對服務的需求來驅動，逐步將企業內部的資訊系統串連起來，建立起即時整合的基礎架構。

3.3. 以軟體代理人為基礎的即時整合方法設計

整合，並不是一件容易的事，必須先清楚其目標，對企業才有正面的助益，盲目的整合，不但對企業幫助不大，有時會變成一種夢靨。因此，企業在進行企業應用整合之前，應該先思考整合的策略，才能發揮整合的功效。

經營客戶關係管理的最終目的，就是希望能為顧客創造最大效益，同時讓企業產生最高獲利，也就是創造一個雙贏的局面，因此，企業在擬定企業應用整合的策略時，也應該以服務客戶為思考的方向，就如同交通大學管理科學系朱博湧教授在『藍海策略－開創無人競爭的全新市場』一書的推薦序中所寫的：企業必須從顧客觀點著手，先了解客戶的需求在哪裡，將客戶重視的需求，加以提升或創造，造成有價值的差異化服務，成為企業獨特的競爭優勢(黃秀媛譯，2005)。

因此，我們提出幾個步驟，作為企業在規劃即時整合時的參考方法。我們所提的方法，是架構在企業內部現有的資訊系統之上，希望能善用這些已經存在的資訊系統，並且以客戶服務為核心，藉由客戶對服務的需求來驅動，逐步將企業內部的資訊系統串連起來。



(1) **定義服務**，這是我們所強調的客戶服務導向的觀念，企業必須先從顧客的觀點去思考，了解、分析客戶的需求，知道哪些是客戶重視的需求，哪些是對客戶有價值的服務，同時，企業也必須了解這些需求對客戶的重要性及優先順序，當受到資源的限制，企業通常沒辦法同時提供所有客戶需要的服務，必須按照規劃好的優先順序，逐步來提供。

(2) **系統分析**，在規劃一個服務時，要分析出需要哪些資訊系統配合，了解相關的資訊系統到底掌握了什麼樣的資源，需要提供什麼方式的支援，有些資訊系統可能只需要分享出它所擁有的一些資訊，有些資訊系統則

需要將它內部的處理流程與其它系統的處理流程連接起來。

- (3) **制定資訊系統服務介面**，將相關的資訊系統，對外所能提供的資源以及提供的方式，設計出對應的軟體代理人，提供公開的介面以及與其它軟體代理人溝通時所需要的通訊協定，透過軟體代理人的方式，資訊系統不需要變更原有的企業邏輯。在分散式的計算環境下，每一個獨立的應用系統都很複雜，有自己的人機界面和資料庫，在不同的作業平台上執行，要讓彼此之間能夠溝通，就必須透過開放的介面及通訊協定，不要受限於任何一種執行平台，也不受限於某一種特定的資訊技術，意即具有技術中立(Technique Neutral)的特性。這些制定出來的資訊系統服務介面是可以被再利用的，也就是說，它不是只能提供給某一項客戶服務使用，也不是只能跟某一特定的資訊系統溝通，任何的服務規劃，只要須要，都可以來與這些已經設計出來的軟體代理人溝通，取得必要的資源。

我們的方法是一種以客戶服務為核心的模型(Customer Service-Centric Model)，並不是一次就要把企業內部所有的資訊系統整合在一起。當企業規劃出一項新的服務要提供給客戶時，就驅動了整合的程序，企業開始分析有哪些資訊系統需要參與提供相關資源，並開始設計軟體代理人來提供相關介面，或是利用先前已經設計好的軟體代理人，來達到服務提供的目的，並逐步建立企業應用整合的基礎架構。隨著提供的服務愈來愈多，參與整合的資訊系統也愈來愈多，相

對的設計出來的軟體代理人也愈來愈多，表示可以被再利用的機率愈高，對後續開發的服務，也許只要重複利用現有的系統介面，不須要再設計新的軟體代理人來提供相關介面，即可提供新服務。在這個概念下，其實，資訊系統的生命週期已經改變了，因為資訊系統可以與其它資訊系統溝通、整合，他不再有固定的生命週期，他可以隨著企業不斷的成長，而持續貢獻他的價值。相較於傳統的軟體開發方式，會因為企業的目標改變，或客戶的需求改變，而必須重新開發一個新系統來滿足新的企業目標或客戶需求，無法快速建立一個可靠的、有彈性的的資訊系統(Irani, et al 2003; Avison and Fitzgerald 1995)，而我們的方法則是運用整合的方式來快速滿足新服務的需求，具備可靠性、彈性及可以再利用的特點。

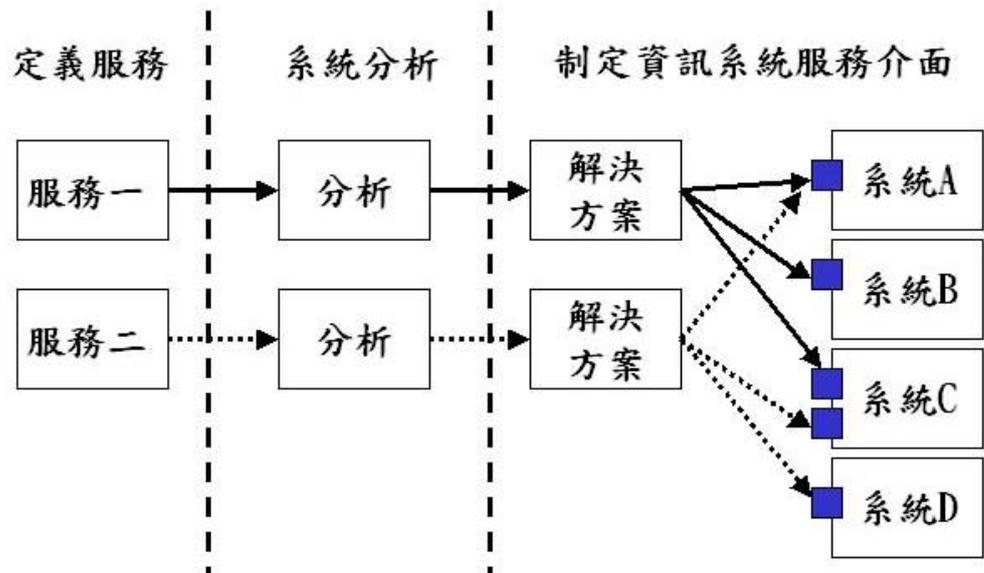


圖 3-2、以客戶服務為核心的模型

我們所設計出來的軟體代理人，依據其所扮演的角色，可以歸納成兩大類：

第一類是即時整合架構的基礎，負責封裝資訊系統對外所能提供的資源及方式，是每一個資訊系統最基本的資訊服務介面，第二類則是設計來整合這些基本的資訊服務介面，包含資訊的整合與流程的整合。在第一類中依照資訊服務提供的方式，又可分為兩種模式：請求／回應模式(Request/Response Model)，以及訂閱／出版模式(Subscriber/Publisher Model)。依此，我們定義出了六種不同屬性的軟體代理人，分別是服務提供者代理人(Service-Provider Agent, SPA)，服務請求者代理人(Service-Request Agent, SRA)，資訊發佈代理人(Information Publish Agent, IPA)，資訊訂閱代理人(Information Subscribe Agent, ISA)，資料整合代理人(Data Consolidation Agent, DCA)，以及流程控制代理人(Process Control Agent, PCA)，其分類如下表所示。



表 3-2、不同屬性的軟體代理人

軟體代理人角色	使用模式	軟體代理人類型
基本的資訊服務介面	服務請求／回應模式	服務提供者代理人
		服務請求者代理人
	資訊訂閱／出版模式	資訊發佈代理人
		資訊訂閱代理人
整合	資料整合	資料整合代理人
	流程整合	流程控制代理人

(1) **服務提供者代理人**，顧名思義就是當資訊系統需要對外提供服務時，所設計出來的代理人，提供的服務可以是資訊系統本身所擁有、所儲存的資訊，或者是資訊系統內部所執行的作業流程，因此，透過服務提供者代理人，資訊系統就可以把自身所擁有的資訊分享出來，或是把本身的流程與其它系統連結，達到整合的目的。服務提供的方式，是一組公開的介面和通訊協定，服務提供者代理人會定義服務的『合約』，描述它所提供的服務內容，介面格式，以及通訊協定。實際上，『合約』的內容是以 XML 的文件形式描述的，使用 XML 文件，主要是因為 XML 目前已經是大家所普遍接受且熟悉的一種標記語言，能夠描述具有彈性、可擴充性的訊息格式。

(2) **服務請求者代理人**，當資訊系統需要使用其它資訊系統所提供的服務時，就必須設計對應的服務請求者代理人來請求服務。服務請求者代理人必須遵循服務提供者代理人所制定的『合約』，來和服務提供者代理人溝通，並根據介面格式的定義，分析得到的資訊。實際上，服務請求者代理人會發出一個 XML 格式的訊息給服務提供者代理人，請求服務，服務提供者代理人分析過訊息內容之後，依據請求，執行服務，然後將執行結果同樣以 XML 格式的訊息傳回給服務請求者代理人。

(3) **資訊發佈代理人**，也是一種提供服務的代理人，提供主動告知的服務，當資訊發佈代理人發佈資訊時，同時可能會有一個或一個以上的資訊訂閱代理人在接受資訊，這個發佈的動作，可以單純只是提供即時資訊的通知，也可以主動觸發其他系統內部的執行程序。與服務提供者代理人之間的差異，就在於提供服務的主被動方式，服務提供者代理人是屬於被動的，是由系統外部的請求來驅動，而資訊發佈代理人則是主動的，可以主動發佈即時的資訊給其它相關的系統。同樣的，資訊發佈的方式，也是使用 XML 文件。

(4) **資訊訂閱代理人**，相對於資訊發佈代理人，是一種使用資訊服務的代理人，透過註冊的機制，接受資訊發佈代理人所發佈的資訊。與服務請求者代理人之間的差異，就在於使用資訊服務的主被動方式，資訊訂閱代理人是屬於被動的，不會主動去要求資訊，而是靜靜等待著即時資訊的發佈，而服務請求者代理人則是主動的，會主動請求其它系統提供資訊。

(5) **資料整合代理人**，是設計用來將得自各個資訊系統的零碎資訊，整合成完整的資訊。當所需要的資訊儲存在數個不同的系統時，就必須分別向每一個系統請求提供資訊服務，然後將這些資訊整合。在設計上，必須考慮到每個資訊系統的回應時間可能不同，因此，資料整合代理人必須能夠處理回應資料的同步問題，等待各系統都回應資料後再加以整合，或是捨棄掉逾時沒有回應的資料。這些整合後的資訊，除了提供資訊系

統內部使用外，也可以變成一種新的資訊服務，提供給其它系統使用。

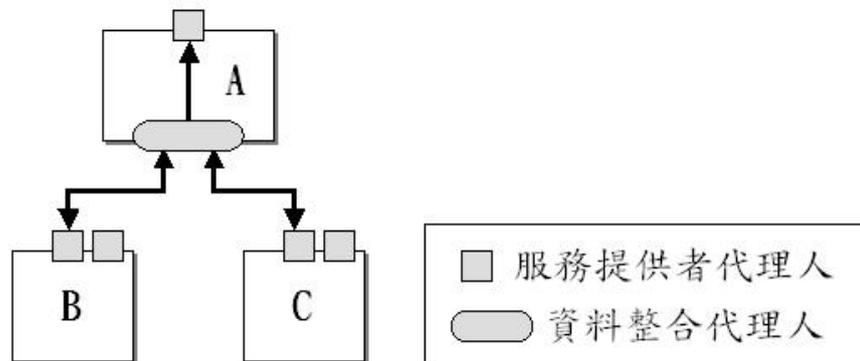


圖 3-3、資料整合代理人架構圖

- (6) **流程控制代理人**，是設計來請求不同的系統所提供的流程服務，然後將這些流程服務串聯起來執行，藉以達到作業流程管理、整合的目的。流程控制代理人能夠紀錄每一個流程服務執行的時間，紀錄整個執行的進度和過程，還必須能夠根據每一個流程服務的執行結果，決定下一個要執行的流程服務。控制整個流程服務運行的規則，可以定義在關聯式資料庫中，或是定義在 XML 格式的檔案中。同樣的，整合後的流程，也可以變成一種新的流程服務，再提供給其它的系統使用。

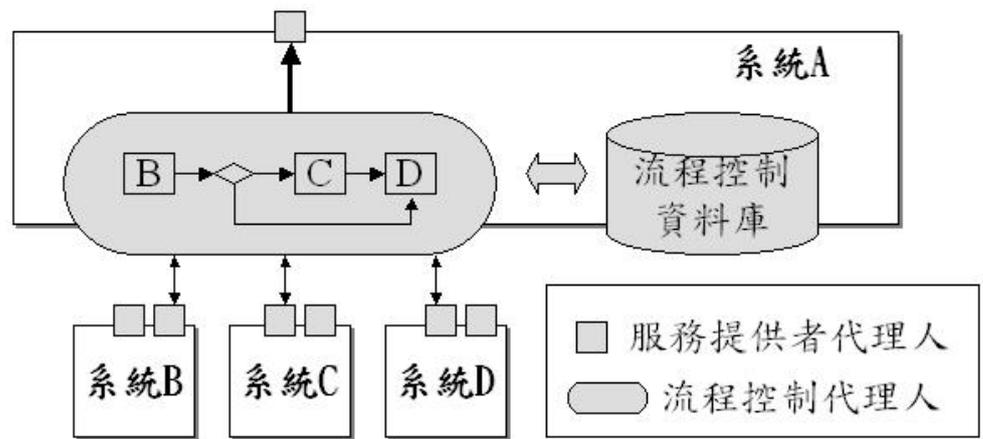


圖 3-4、流程控制代理人架構圖

在下一章節，將透過中華電信公司內部應用的實例來說明，如何應用這種以客戶服務為導向的方法，逐步來建立企業應用整合的基礎架構，透過這六種不同類型的軟體代理人設計，讓隱身在資訊系統背後的資訊與程序得以互相分享、連接，充分發揮現有資訊系統的投資效益，同時也滿足中華電信公司客戶服務作業上的需求。

第 4 章、實例探討

4.1. 背景介紹

中華電信股份有限公司，主要業務包括固網通信、行動通信，以及數據通信三大領域，總資本額為新台幣 964.77 億元，係台灣營運經驗最豐富、規模最大之綜合電信業者，也是國際間電信業者重要的合作伙伴，以成為「知識經濟時代的價值創造者」，「行動資訊社會的領航者」，和「客戶、投資人與員工的價值創造者」作為企業經營的願景。統計至民國 95 年 3 月，員工人數 25,537 人，市內電話用戶數超過一千三百萬，行動電話用戶數超過八百二十萬，網際網路用戶數超過四百一十萬(包含寬頻上網的用戶數三百七十六萬)，九十四年度營業收入新台幣 1833.8 億元，營業利益新台幣 585.6 億元 (<http://www.cht.com.tw>)。

對中華電信而言，企業客戶是重要的競爭市場，企業客戶雖然數量較少，但是對營收的貢獻度很大，而且可以作為市場的指標，因此，如何粘住企業客戶，提供更貼心的服務，以提高企業客戶的滿意度及忠誠度，是很重要的競爭策略。基於此，中華電信針對大企業客戶量身訂作，在 Internet 上建置大客戶服務中心，提供大企業客戶差異性服務。本章的實例探討，即針對大客戶服務中心的建置案例來加以分析，將我們所提出的架構和方法，應用在大客戶服務中心的部份服務和相關的資訊系統上。

本章所使用的資訊系統名稱，並非中華電信公司內部所使用的真正名稱，所

定義的服務內容，也並非中華電信公司內部所使用的全部內容，僅是擷取部份資料，幫助我們對所欲探討的架構、觀念和方法，作適當的闡述。而整個中華電信大客戶服務中心資訊體系架構，則可以參考吳秀蘭(2005)在她論文中的描述。

4.2. 定義服務

對企業客戶而言，不管是語音服務、數據服務、甚至是各種商業活動，都是相當依賴所租用的電信網路來進行，網路服務的品質間接影響到企業客戶日常的作業及交易。當發現到網路有問題，服務無法正常使用時，希望能夠有順暢的管道可以申訴，也希望問題能盡快解決。



對中華電信而言，維持網路服務的穩定與品質，迅速且有效的排除網路障礙，降低維運成本，是很關鍵的一項競爭要件。當客戶發現到網路有問題，報怨服務無法正常使用時，能夠在最短的時間內，找到問題點，迅速排除障礙，恢復正常服務，並掌控整個過程，是提高客戶滿意度與忠誠度，建立良好口碑與企業形象的一個關鍵。

因此，如何提供企業客戶順暢的申訴管道，以及如何迅速排除網路障礙，恢復正常服務，是一個能夠創造雙贏的方式，一方面能縮短障礙時間，一方面能夠提供給客戶更好的服務，同時增加了客戶的利益與企業的利潤。

經過對企業客戶需求的了解與分析，首先，我們在大客戶服務中心上建置兩

個服務，提供企業客戶使用。第一個是障礙申訴服務，當企業發現網路服務有問題時，可以透過大客戶服務中心立即向中華電信反應；第二個是查詢障礙處理進度的服務，透過此服務介面，企業客戶可以掌握所有承租網路的障礙情形，及處理進度。

4.3. 系統分析

完成了對企業客戶服務需求的分析，接下來我們分析，中華電信內部需要哪些資訊系統配合，才能滿足客戶的需求，並縮短障礙處理的時間。



首先，我們想到了障礙處理資訊系統。障礙處理資訊系統原本就是用來接受客服人員的障礙通報，也就是說當客服人員接到客戶的障礙申訴電話時，可以記錄這些事件的發生，並啟動相關的處理流程，記錄整個處理過程，過程中，能提供個別事件的障礙處理進度給相關人員。因此，由障礙處理資訊系統來提供上一節所描述的兩個服務給大客戶服務中心是非常適合的。

分析歷史資料，我們發現約有百分之十五的障礙申訴是屬於重複的申告，也就是說，在客戶發現服務無法使用，而欲進行障礙申告之前，已經有其他人將這個障礙申告進來；約有百分之五的障礙申訴是因為用戶本身帳單逾期未繳而遭到停止服務，以及網路資源重新調整所造成的服務暫時中斷。如果在受理用戶障礙申告之時，能夠得到這些資訊，就可以立即告訴客戶障礙發生的原因，或是目

前的處理情形，讓客戶明白，同時也可以減少百分之二十的障礙申告受理作業，提升作業效率。為了達成這樣的目的，需要帳務系統及網路資源管理系統的配合，帳務系統需要提供一个服務，主動更新電路的服務狀態，說明服務目前是可以正常使用，還是已經停止使用，而網路資源管理系統也需要提供一个服務，通報影響到用戶服務的資源調整作業，例如，設備升級，或是纜線工程等等。

為了能主動去關懷客戶，提供貼心感受，企業客戶代表希望，當客戶障礙發生及修復完畢時，都可以在第一時間知道，方便企業客戶代表主動去追蹤問題，關懷客戶，所以，也必須考慮和客戶關懷系統整合，讓障礙處理資訊系統可以主動發送障礙申告與障礙修復的資訊給客戶關懷系統。



在障礙處理的過程中，包含了障礙測試及障礙查修兩項作業。障礙測試作業主要的目的在釐清障礙原因及障礙位置，而障礙查修作業則必須派遣專業的查修人員到障礙點實際去修復障礙。過去的作法，在執行障礙測試工作時，測試人員必須先登入到障礙處理資訊系統，取得需要進行測試的用戶障礙資訊，然後再登入測試系統，釐清用戶的障礙原因後，再一次登入障礙處理資訊系統，回報測試結果，以更新障礙處理的狀態。而執行障礙查修工作時，查修人員也是必須先登入到障礙處理資訊系統，取得必須派修的用戶障礙資訊，到障礙點實際去修復障礙後，再一次登入障礙處理資訊系統，回報修復情形，更新障礙處理的狀態。障礙處理資訊系統則是根據這些回報，來記錄整個障礙處理的過程，直到障礙被排除為止。不難發現，這樣的程序，牽涉到許多人工作業，且必須一直重複，直到

障礙被排除為止，如果能減少人工作業，讓整個障礙處理過程更加自動化，就能縮短處理時間，代表能在更短的時間內，恢復提供服務，一方面能減輕人力的負擔，一方面也提高了客戶的滿意度。因此，必須再結合測試系統及無線派工系統，讓整個障礙處理過程更有效率。測試系統主要的功能在於能夠透過測試設備去分析和釐清障礙發生的原因，百分之五十以上的障礙原因，都能透過系統自動去診斷，只有少數的情形，非得由測試人員介入不可，如果讓測試系統可以主動來擷取需要進行測試的用戶障礙資訊，取代掉人工登入擷取資料的作業方式，大部分的測試工作就可以由系統自動完成並回報，剩餘的測試工作才由測試人員介入完成，可以節省許多人力和時間。而無線派工系統主要的功能在於能夠將工作任務，傳送到查修人員的行動裝置上，例如手機，或是個人數位助理(PDA)，同時，查修人員也可以透過行動裝置回報修復結果。同樣的，由於查修人員經常在外奔波，如果無線派工系統能夠主動擷取必須派修的用戶障礙資訊，然後將資訊主動傳送到查修人員的行動裝置上，就可以節省不少往來辦公室與障礙點之間的交通時間，提高障礙查修的時效性。經由以上的分析，我們可以得到參與大客戶服務中心服務提供的系統關係圖如下。

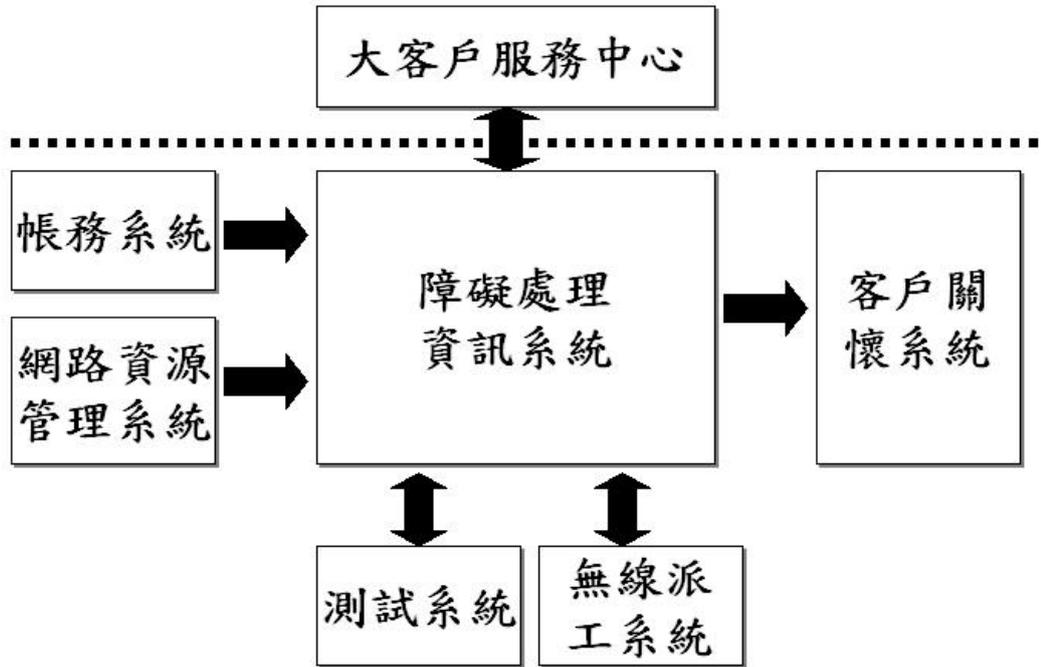


圖 4-1、支援大客戶服務中心服務的系統關係圖

4.4. 制定資訊系統服務介面



依據上一節的系統分析，我們可以開始定義服務的『合約』，也就是描述服務提供者代理人 and 資訊發佈代理人它們所能提供的服務內容及介面格式。首先，我們先替障礙處理資訊系統定義兩個服務合約，分別用來提供『障礙申訴』及『查詢障礙處理進度』的服務，然後實作服務提供者代理人，結合系統內的企業邏輯，對外(大客戶服務中心)提供資訊服務。而需要這兩項服務的系統(大客戶服務中心)，則可以依據服務合約內容，來實作對應的服務請求者代理人，來要求『障礙申訴』及『查詢障礙處理進度』的服務，提供給企業客戶使用。附錄中的表 A-1 和表 A-2 所定義的是『障礙申訴』服務的合約內容，服務請求者代理人依據表 A-1

的描述，傳送訊息給服務提供者代理人，請求『障礙申訴』服務，而服務提供者代理人則將受理『障礙申訴』的執行結果，依據表 A-2 描述的格式，回傳給服務請求者代理人。

附錄中的表 A-3 和表 A-4 則是定義『查詢障礙處理進度』的服務合約，服務請求者代理人可以依據表 A-3 的描述，傳送訊息給服務提供者代理人，查詢整個企業所承租的網路，目前哪些是有障礙的，以及目前處理的情形如何，而服務提供者代理人則將查詢結果，依據表 A-4 描述的內容，回傳給服務請求者代理人。



實際在執行上，中華電信公司內部組織的劃分，分成北中南三個區分公司，各分公司都有一套障礙處理資訊系統，且只擁有其維運區域內的障礙資訊。舉個例子來說，當大客戶服務中心需要查詢中國信託銀行目前所有已經發生的網路障礙時，障礙處理資訊系統除了查詢所屬分公司內的障礙資訊外，還需要向其他兩個區分公司查詢，才能得到完整的中國信託銀行所有目前已經發生的網路障礙資訊。因此，我們遵照表 A-3、A-4 所定義的合約規範，另外為障礙處理資訊系統設計了一個服務提供者代理人，用來提供該分公司內『查詢障礙處理進度』的服務，也設計了服務請求者代理人，向其它分公司查詢。另外，還設計了資料整合代理人，用來整合自身與其它兩個分公司的網路障礙資訊，對外提供資料整合的服務。各資訊系統之間代理人的關係，如圖 4-2 所示。

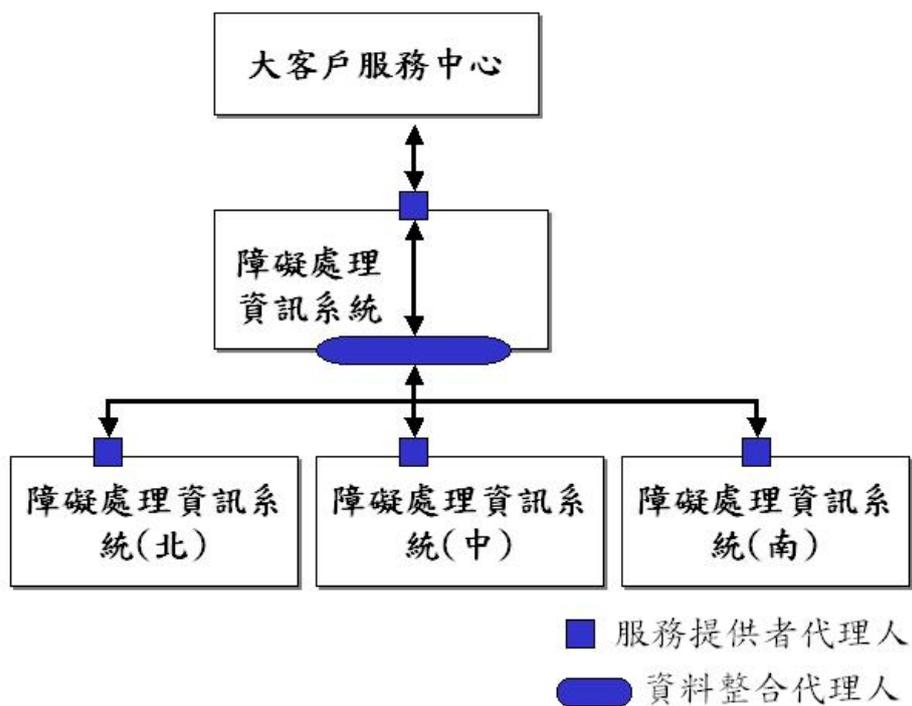


圖 4-2、提供『查詢障礙處理進度』服務的各資訊系統代理人關係圖



上一節提到，障礙處理資訊系統可以與帳務系統和網路資源管理系統來整合，藉以提升受理障礙申訴時的作業效率。因此，我們也請求負責維護帳務系統的單位，定義一個服務合約(參考附錄中表 A-5 的內容)，當電路的服務狀態有變動時，可以主動發佈，而障礙處理資訊系統則必須依據合約內容，設計對應的資訊訂閱代理人，來接受發佈的資訊。同樣的，我們也請求負責維護網路資源管理系統的單位，提供服務合約(參考附錄中表 A-6 的內容)，主動發佈影響到用戶服務的資源調整作業資訊，並為障礙處理資訊系統設計對應的資訊訂閱代理人，來接受發佈的資源調整作業資訊。

最後要說明的是流程控制代理人的設計及運作方式。在上一節系統分析的時
候談到，障礙處理資訊系統可以再結合測試系統及無線派工系統，以減少人工作
業，讓整個障礙處理過程更加自動化，更有效率。而且，為了能提供企業客戶貼
心感受，可以再和客戶關懷系統整合，當客戶障礙發生及修復完畢時，都可以在
第一時間通知企業客戶代表，方便企業客戶代表主動去追蹤問題，關懷客戶。從
客戶申訴，障礙處理，一直到通知企業客戶代表，整個過程，都是由流程控制代
理人來控制的。



為了讓整個障礙處理過程更加自動化，更有效率，障礙處理資訊系統必須提
供一個『擷取用戶障礙資訊』服務的介面，讓測試系統可以主動來擷取需要進行
測試的用戶障礙資訊，也讓無線派工系統可以主動來擷取必須派修的用戶障礙資
訊，取代掉人工登入擷取資料的作業方式。另外，障礙處理資訊系統還必須提供
一個『障礙處理回報』服務的介面，讓測試系統和無線派工系統可以將執行結果
回報給障礙處理資訊系統，取代掉人工登入回報的作業方式。為了能提供企業客
戶貼心感受，讓企業客戶代表可以在第一時間掌握客戶障礙發生及修復完畢的資
訊，障礙處理資訊系統還必須提供一個『用戶障礙資訊發佈』的服務介面，透過
資訊發佈代理人，對外發佈障礙發生及障礙修復的資訊。

而為了控制整個過程，流程控制代理人必須與提供『障礙申訴』的服務提供

者代理人，提供測試系統和無線派工系統回報服務的『障礙處理回報』服務提供者代理人，以及提供對外發佈障礙發生及障礙修復資訊的『用戶障礙資訊發佈』服務代理人密切互動，才能掌握從客戶申訴，障礙處理，一直到通知企業客戶代表的整個過程即時的障礙狀態，而透過流程控制資料庫的建立，流程控制代理人可以根據『障礙申訴』的內容，以及測試系統和無線派工系統的『障礙處理回報』結果，來控制整個障礙處理作業的流程。這些代理人彼此之間的關係，如圖 4-3 所示。流程控制資料庫的內容，描述一個有限狀態機(Finite State Machine)，由目前狀態，輸入資料，來推導出下一狀態(可以參考表 4-1 的例子)，狀態的類型包含了障礙申訴、測試、查修、及結案，而輸入資料的內容則包含障礙申訴的資料及測試系統與無線派工系統的障礙處理結果。上述各服務介面的合約內容，則可以參考附錄中表 A-7、A-8、A-9、A-10 所描述的內容。



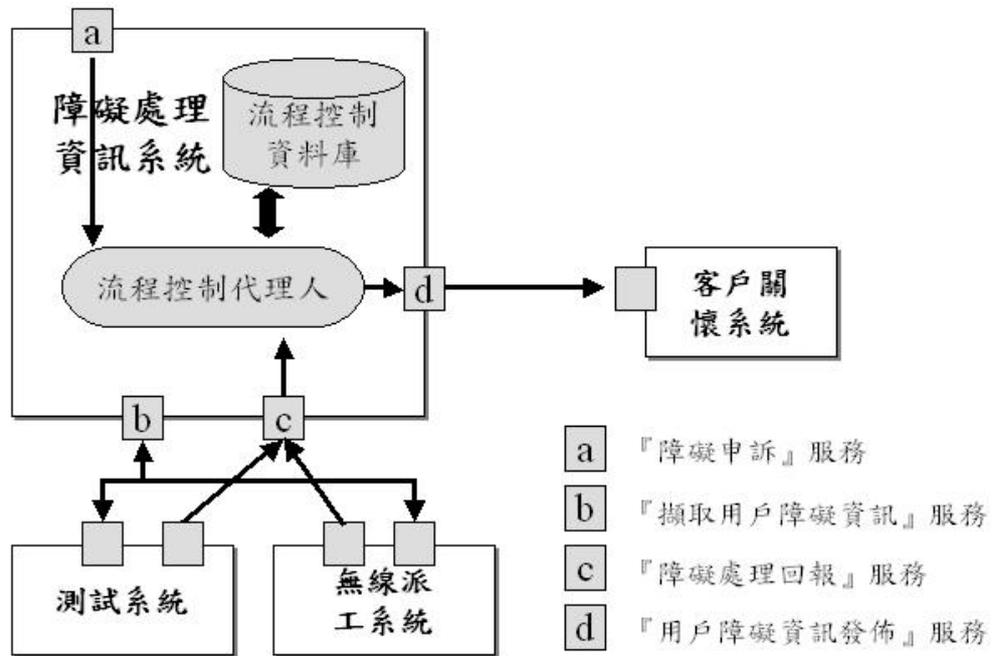


圖 4-3、流程控制代理人與其他代理人之間的關係圖

表 4-1、流程控制資料庫範例

目前狀態	輸入	下一狀態
0 (障礙申訴)	DC (斷線)	1
0	CPE (用戶設備故障)	2
0	PWD (密碼錯誤)	3 (結案)
1 (測試)	TIE (機房線路故障)	1
1	MOD (數據機故障)	2
1	TOK (測試結果良好)	3
2 (查修)	PBX (機房設備故障)	2
2	ROK (修復)	3

綜合上述的分析，障礙處理資訊系統所有對外提供的服務，以及軟體代理人

與其它資訊系統之間的關係圖，可以由圖 4-4 來表示。

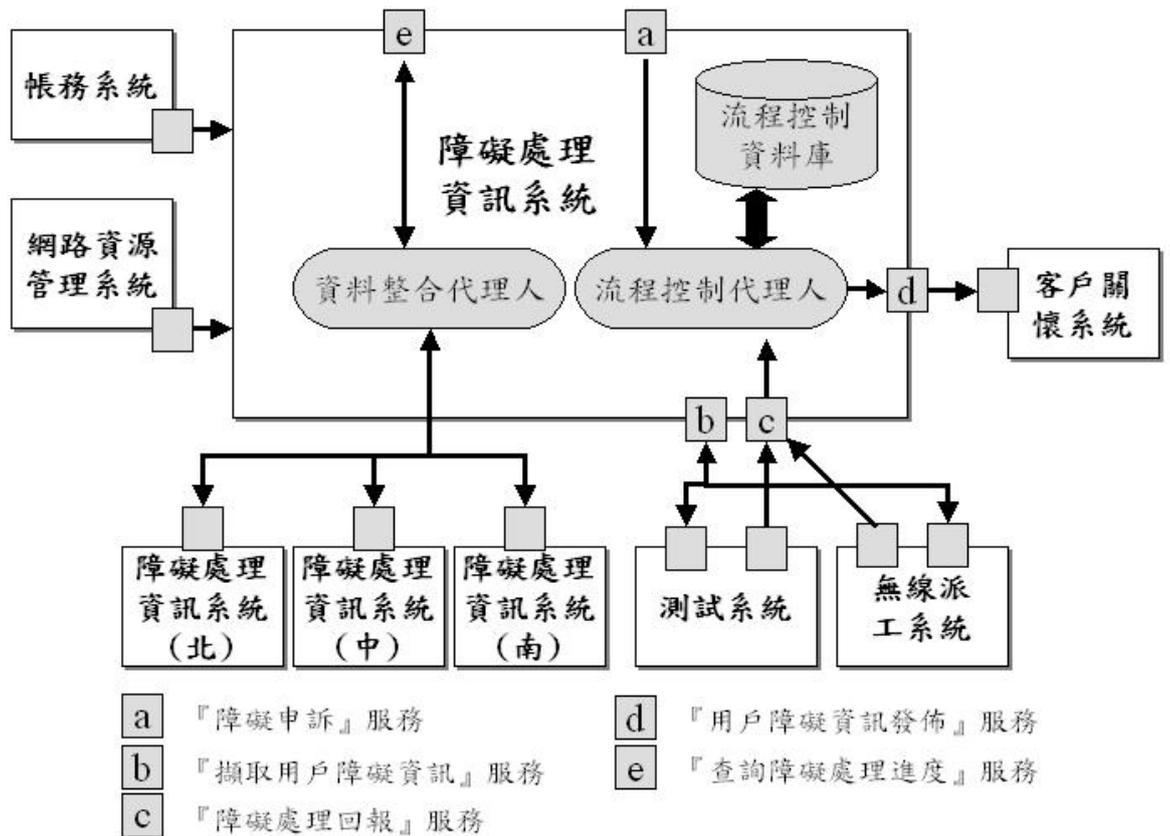


圖 4-4、障礙處理資訊系統與其它資訊系統之間的關係圖

4.5. 服務介面的再利用

障礙處理資訊系統為了滿足大客戶服務中心的需求，提供企業客戶障礙申訴及查詢障礙處理進度的服務，進而制定了相關的服務合約，並設計對應的軟體代理人來提供外界服務。這些已經完成的服務及軟體代理人除了提供給大客戶服務

中心使用外，也可以提供給其它需要的資訊系統來使用，達到新服務快速開發、重複利用的目的。舉例而言，某些企業客戶可能不喜歡自己上網來申訴障礙，或查詢障礙處理進度，反而喜歡打電話給企業客戶代表，抱怨障礙情形或詢問障礙處理進度，這些企業客戶代表平時可能都是在客戶關懷系統上作業，因此可能要求客戶關懷系統的維護單位提供相關功能，這時，客戶關懷系統的維護單位就可以利用障礙處理資訊系統所提供的服務，很快地開發出相關功能，滿足企業客戶代表的需求。

障礙處理資訊系統原本就可以接受客服人員的障礙通報和障礙查詢，也就是說當客服人員接到客戶的障礙申訴或障礙查詢電話時，需要登入到障礙處理資訊系統，記錄客戶申訴的相關資料，或是查詢障礙處理進度的資訊，然後回應客戶，因此，客服中心的資訊系統也可以整合障礙處理資訊系統所提供的服務，免除需要不斷地登入到障礙處理資訊系統的動作。

而『用戶障礙資訊發佈』服務除了用來服務企業客戶代表外，也可以提供給更多資訊系統使用。例如，帳務系統可以來註冊成為『用戶障礙資訊發佈』服務的訂閱者，當障礙修復之後，可以依據因障礙發生而暫停服務的時間，與當初跟用戶簽訂的服務等級協議(Service Level Agreement, SLA)，來決定是否需要扣減租費。

4.6. 服務介面設計分析

表 A-1、表 A-2 所描述的是一個『請求障礙申訴服務』的合約內容，表 A-1 描述外部資訊系統該如何請求障礙處理資訊系統來提供障礙申訴服務，而表 A-2 則是描述障礙處理資訊系統在接受請求後，回應的內容，在這一節，我們以表 A-1、表 A-2 為例，來分析服務介面的設計方式。

整個服務介面的設計，分為服務介面識別、與服務介面主體兩部份，分別用 XML 標籤<Head>和<Body>來包含。

```
<xs:sequence>  
  <xs:element ref="Head" />  
  <xs:element ref="Body" />  
</xs:sequence>
```

在服務介面識別的部份，又分為請求與回應兩種類型。請求服務時，服務介面識別的內容包含有來源系統識別碼、來源系統訊息序號、目的系統識別碼、及所請求的服務名稱，分別以<SourceSystemID>、<SourceMessageID>、<DestSystemID>、<ServiceName>等 XML 標籤來表示。

```
<xs:element name="Head">  
  <xs:complexType>  
    <xs:sequence>  
      <xs:element ref="SourceSystemID" />  
      <xs:element ref="SourceMessageID" />  
      <xs:element ref="DestSystemID" />  
      <xs:element ref="ServiceName" />  
    </xs:sequence>  
  </xs:complexType>  
</xs:element>
```

而回應服務請求的服務介面識別，則包含有來源系統識別碼、目的系統識別碼、目的系統的訊息序號、及所回應的服務名稱，分別以<SourceSystemID>、<DestSystemID>、<DestMessageID>、<ServiceName>等 XML 標籤來表示。

```
<xs:element name="Head">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="SourceSystemID"/>
      <xs:element ref="DestSystemID"/>
      <xs:element ref="DestMessageID"/>
      <xs:element ref="ServiceName"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

服務介面主體的內容，會依據不同的服務介面類型(請求與回應)，不同的服務名稱，定義出不同的內容。這樣的設計方式具有以下兩項好處：

- (1) 可以利用一個軟體代理人來實踐多個服務介面，因為軟體代理人可以從服務介面識別中的服務名稱，來辨別出外部系統所要請求的資訊服務，使用資訊系統內對應的企業邏輯來執行，然後回應服務請求。對外的介面減少了，可以減輕管理上的負擔。
- (2) 具備彈性，可以適用於表 3-1 所描述各種企業應用整合的拓撲架構。詳細說明參考表 4-2。

表 4-2、適用各種整合的拓撲架構的設計說明

拓撲架構	說明
Hub	星型架構，來源系統整合到某一中心點，透過來源系

	統識別碼、目的系統識別碼、及服務名稱的設計，訊息可以在此中心點互相交換。
Bus	佈告欄模式，來源系統將訊息送到匯流排上，透過來源系統識別碼、及目的系統識別碼的設計，各資訊系統可以從匯流排上取出屬於自己的訊息。
Point-to-Point	應用系統彼此互連，最簡單的方式，只需等待請求的回應，不需任何特殊設計。
Pipeline	使用先進先出(first in first out)的資訊流，類似 Bus Topology，只是訊息必須依序處理，透過來源系統識別碼、及目的系統識別碼的設計，各資訊系統可以取出屬於自己的訊息。
Network	適合非同步的活動和獨立的交易，透過訊息序號的設計，將請求時的訊息序號與回應時的訊息序號關連起來，可以滿足非同步的請求、回應方式。



第 5 章、資料保護機制的設計

對企業而言，妥善保護客戶資料，是一種義務，也是一種責任，要做好客戶關係管理，一定要更重視客戶資料保密。一旦客戶資料洩露出去，客戶會產生資料安全上的疑慮，對企業產生不信任感，不但競爭對手可能藉機搶走重要客戶，企業形象也會因而受損。除了保護客戶資料，員工的資料也要受到保護，可以避免員工受到外界不必要的干擾。

整合帶來了便利，卻也帶來了資料保密的問題，透過整合，客戶和員工的資料可以由很多管道取得，若缺乏適當的管控，很容易洩露出去。因此必須輔以適當的保護機制，以及監控的方法，才能在享受整合所帶來的好處的同時，也能達到資料保護的目的。



5.1. 資訊訂閱代理人的註冊程序

由於各資訊系統的管轄單位不同，考量到的資訊安全政策也不同，所以當資訊訂閱代理人需要資訊發佈代理人所發佈的資訊時，其所需的註冊程序並無法以自動化的方式來完成，而是必須透過人工的方式向該管轄單位提出申請，待批核後，該管轄單位才會將此資訊訂閱代理人的位置加入資訊發佈列表之中(參考表 5-1)，完成此註冊程序後，資訊訂閱代理人才能夠接收到資訊。未來，這也是一個可以研究的課題，看如何在符合組織資訊安全政策的前提下，建立一套自動化

的註冊機制。

表 5-1、資訊發佈列表

服務名稱	系統代碼	資訊訂閱代理人位置
TroubleNotify	CARE	http://10.144.1.2/servlet/SubscriberAgent
TroubleNotify	BILL	tcp://10.144.11.2:9501

5.2. 提昇資訊系統安全稽核能力

透過整合，資訊系統內部的資料得以與其它外部的資訊系統分享，但對於某些敏感性的資料，如客戶資料和員工資料，都要防止其被不當的存取，所以，資訊系統必須加強本身的稽核能力，隨時監控系統的存取紀錄，對於不正常的資料存取行為要能夠偵測出來，才能及時遏止資料被惡意竊取。例如，資訊系統可以對過去的存取紀錄，計算出一個合理存取範圍的臨界值，包含上班日、非上班日、上班時段和非上班時段，當監控系統的存取紀錄時，發生超過臨界值的情形時，即主動發出告警，提醒管理者追蹤、了解。

5.3. 資訊安全代理人的設計

雖然各資訊系統可以透過已經定義好的服務合約內容，來存取資料，但是從

資料保護的角度來看，我們必須要限制某些系統所能夠看到的資料內容，而不是所有的系統都擁有相同權限。例如，我們規範來自 Internet 應用系統的請求，不允許看到客戶的身分証字號、地址，也不允許看到員工的姓名、連絡方式等等，而來自企業內部客服人員及企業客戶代表所使用的資訊系統的請求，則可以看到客戶的地址，員工的姓名、連絡方式等等，因此，我們再提出一個資訊安全代理人(Security Agent)的設計方法，解決上述的問題。

在 4.4 節，制定資訊系統服務界面的方法中，我們在訊息的格式的設計上，定義了一個表頭的部份，參考附錄表 A-8 的範例：

```
<Head>  
  <SourceSystemID>PH</SourceSystemID>  
  <DestSystemID>FORCE</DestSystemID>  
  <DestMessageID>FO12345678</DestMessageID>  
  <ServiceName>ListTickets</ServiceName>  
</Head>
```

當資訊系統要回應請求時，從 XML 訊息標籤<DestSystemID>的內容可以判別出訊息接收端的系統是哪一個，資訊安全代理人可以利用這樣的訊息，遮住不被允許看到的資訊。

首先，需要先建立一個資訊安全政策資料庫(Security Policy Base)，存放相關的系統名稱、安全性等級、被限制存取的資訊等內容。如表 5-2、5-3 的設計。

表 5-2、資訊安全政策資料庫設計：系統名稱和安全性等級

系統名稱	安全性等級
* (其它系統)	9
FORCE (無線派工系統)	1
ITS (測試系統)	5
SC (大客戶服務中心)	9
CARE (客戶關懷系統)	1

表 5-3、資訊安全政策資料庫設計：安全性等級和被限制存取的資訊

安全性等級	被限制存取的資訊
9	Address
9	StaffName
9	StaffMobile
5	Address

當所有訊息要傳送出去時，都必須先經過資訊安全代理人的過濾，資訊安全代理人根據資訊安全政策資料庫中的定義，對於被限制存取的資訊內容，保留其 XML 標籤，但是遮住被限制存取的資訊內容，來達到資料保護的目的，其程序如圖 5-1 表示。例如，附錄表 A-8 的範例，根據資訊安全政策資料庫中的定義，當訊息回傳給大客戶服務中心時，客戶的地址、員工的姓名和連絡方式等資訊，都

會被遮住，如下表 5-4 所示。

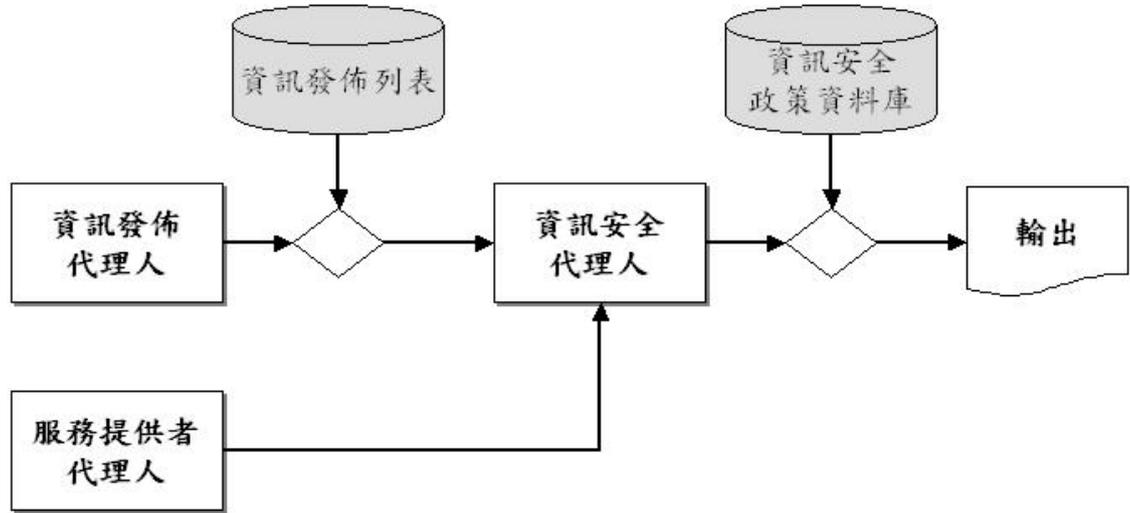


圖 5-1、輸出資訊過濾程序

表 5-4、資訊安全代理人的訊息處理範例

```

    <TroubleTicket>
      <TicketID>TP12345678</TicketID>
      <TicketTime>95-05-18 10:00</TicketTime>
      <ResourceNum>225D825123</ResourceNum>
      <RootCauseID>DC</RootCauseID>
      <RootCauseName>斷線</RootCauseName>
      <StaffName id="A02857">XXX</StaffName>
      <StaffMobile>XXXXXXXXXX</StaffMobile>
      <Customer>
        <EnterpriseID>A987654321</EnterpriseID>
        <EnterpriseName>中國信託銀行
      </EnterpriseName>
        <Contact>Mr. YEH</Contact>
        <Phone>02-22113456</Phone>
        <Address>XXXXXXXXXXXXXX</Address>
      </Customer>
    </TroubleTicket>
  
```

第 6 章、結論和未來的研究方向

6.1. 研究貢獻

成功的客戶關係管理，是企業成長與永續經營的關鍵之一，系統化的建立及維持客戶關係的能力很快就會成為競爭的必備武器。假如無法有效地提升服務客戶的水準，無論成本如何降低，我們能夠服務的客戶數量只會愈來愈少而已。過去以功能劃分的部門，會使許多服務客戶的活動無法整合，打破以功能區分組織的企業框架，改以客戶為核心，是本論文所要探討的重點。

本論文以客戶服務為核心，重新思考資訊系統的定位與價值，我們提出了三個步驟，即首先須先定義服務，瞭解客戶的服務需求，規劃要提供哪些服務給客戶，接著要分析有哪些相關的資訊系統必須參予，並分析每個資訊系統在參與提供服務的過程中，所扮演的角色，最後本著技術中立(Technique Neutral)及服務導向的精神，定義服務介面，並透過軟體代理人的方法來實踐，協助建立一個即時整合的企業，提供更好的服務給客戶。

傳統的軟體開發方法，為了滿足企業的目的或使用者需求，就必須再開發出一套新的資訊系統來支援，不但無法反應企業快速變遷的需求，資訊系統本身的價值和貢獻度也較低，生命週期亦短暫。本論文強調以企業內部現有的資訊系統為基礎，在非常異質性的資訊環境中，以漸進的方式，逐步建立企業內部一個即時整合的架構和環境。在分散式的計算環境下，透過軟體代理人的包裝，讓企業

內已經存在的資訊系統變成服務的提供者，資訊系統不再像過去一樣，只能夠用來提供某些特定的企業活動，原本只屬於系統內部的活動，開始可以提供給外部使用，原本只存放在系統內部的資訊，開始可以提供給其他系統分享，讓隱身在資訊系統背後的資訊與程序得以互相分享、連接，讓現有的系統能持續創造新的價值，充分發揮投資效益。

除了提出架構和方法之外，我們也將這些方法實際應用在中華電信公司的案例上，加以驗證，結果也證明，當新的服務提出時，不需要再開發新的資訊系統來提供解決方案，反而能善用各部門現有的資訊系統，來快速反應客戶需求，提升了服務品質和作業效率。透過服務介面的設計，除了點對點的系統連接方式外，也能適用其他不同類型的企業應用整合拓璞架構，具備彈性，且不侷限於特定的執行平台，而且可以用一個軟體代理人來實踐多個服務介面，透過服務名稱的設計，來辨別出外部系統所要請求的資訊服務，然後執行對應的企業邏輯來回應服務請求。

此外，在實際應用上，我們感受到保護客戶資料是企業很重要的一個資訊安全政策，因此，本論文也提出資訊安全代理人與資訊安全政策庫的設計，來解決資訊保護的議題，資訊安全代理人根據資訊安全政策資料庫中的定義，針對不同的訊息接收對象，過濾被限制存取的資訊內容，保留其 XML 標籤，但是遮住被限制存取的資訊內容，才能在享受整合所帶來的好處的同時，也能達到資料保護的目的。

6.2. 未來研究方向

最近，以 Web Service 技術(<http://www.w3.org/TR/ws-arch/>)為基礎的服務導向架構(Service Oriented Architecture, SOA)開始興起(Huhns and Singh 2005; Papazoglou and Georgakopoulos 2003)，希望每一個資訊服務都能像日常生活的電話和水電一般，能夠很方便使用它。

服務導向架構的基礎包含一個服務註冊(registry)與服務發現(discovery)的機制，期望所有可以對外提供的資訊服務都能註冊，期望能夠自動搜尋已註冊的服務，然後迅速開發新的服務出來，未來我們將思考如何融合軟體代理人的技術及服務導向架構的精神，在符合企業資訊安全政策的前提下，建立一個自動化的機制和一個整合的基礎環境，讓資訊系統可以去搜尋所有服務提供者代理人 and 資訊發佈代理人的所提供的服務，並可以自動向資訊發佈代理人註冊，訂閱資訊。

另外，制定服務描述語言，透過一種標準化的語意和語法來描述服務，進一步建立服務組裝工廠(或是智慧型決策中心)的機制，也是另一個值得研究的方向。在這樣的理想下，當外界有新服務的需求時，服務組裝工廠就能分析需求，然後派遣軟體代理人搜尋合適的服務介面，以完全自動化的方式組裝出新的服務來滿足新需求，可以讓整個新服務的開發，更快、更有效率。

參考文獻

1. 李瓊芬譯，2000，『新世代的客服中心趨勢』，電子化企業經理人報告，第七期，9-17 頁。
2. 吳秀蘭，2005，『探討中華電信導入顧客關係管理之研究』，國立交通大學，管理學院碩士在職專班經營管理組碩士論文。
3. 吳姝倩，2001，『台灣客服中心的現況發展與未來趨勢』，電子化企業經理人報告，第二十一期，10-24 頁。
4. 陳曉開譯，Jill Dyche 著，2003，客戶關係管理手冊，台北，台灣培生教育出版股份有限公司。
5. 黃秀媛譯，W. Chan Kim, and Renee Mauborgne 著，2005，藍海策略，開創無人競爭的全新市場，台北，天下遠見出版股份有限公司。
6. 黃貝玲，2001，『從 Call Center 到 Contact Center』，電子化企業經理人報告，第二十一期，29-38 頁。
7. 覃笑風譯，Frederick F. Reichheld 著，2004，『顧客忠誠成長之始』，哈佛商業評論中文版，第二十八期，46-56 頁。
8. 楊鴻儒譯，日本人力資源學院(HR Institute)著，2001，CRM 戰略執行手冊，台北，遠擎管理顧問股份有限公司。

9. 劉復苓、邱天欣譯，John G. Freeland 著，2003，CRM 關鍵 32 堂課，台北，美商麥格羅•希爾國際股份有限公司 台灣分公司。
10. 蘇隄，1998，『資料倉儲的應用與技術』，資訊與電腦，2 月號，78-89 頁。
11. Avison, D., and Fitzgerald, G., “Information Systems Development: Methodologies, Techniques, and Tools”, McGraw Hill, London, UK, 1995.
12. Baek, S., Liebowiz, J., Prasad, S. Y., and Granger, M., “Intelligent Agents for Knowledge Management-Toward Intelligent Web-Based Collaboration within Virtual Teams”, Knowledge Management Handbook, Chapter 11, CRC Press, 1999.
13. Berry, M.J.A., and Linoff, G., “Data Mining Techniques for Marketing, Sales, and Customer Support”, Wiley, New York, 1997.
14. Chan, C. W., Chen, Lin-Li, and Geng, Liqiang, “Knowledge Engineering for An Intelligent Case-Based System for Help Desk Operations”, Expert Systems with Application, vol. 18, 2000, pp.125-132.
15. Chaudhuri, S., and Dayal, U., ”An Overview of Data Warehousing and OLAP Technology”, SIGMOD Record, Vol. 26, No. 1, March 1997.
16. Davenport, T. H., “Putting the Enterprise into the Enterprise System”, Harvard

Business Review, July-Aug. 1998, pp.121-131.

17. Davenport, T. H., Long, D. W. D., and Beers, M. C., “Successful Knowledge Management Projects”, Sloan Management Review, Winter 1998, pp. 43-57.
18. Davenport, T. H., and Prusak, L., “Working Knowledge: How Organizations Manage What They Know”, Harvard Business School Press, Boston MA, 1998.
19. Gillies, C., Rigby, D. K., and Reichheld, F. F., “The Story Behind Successful Customer Relations Management”, European Business Journal, vol. 14, iss. 2, 2002, pp.73-77.
20. Glushko, R. J., Tenenbaum, J. M., and Meltzer, B., “An XML framework for Agent-based E-commerce”, Communications of ACM, Vol. 42, No. 3, March 1999, pp.106-114.
21. Grover, V., and Davenport, T. H., “General Perspectives on Knowledge Management: Fostering a Research Agenda”, Journal of Management Information Systems, Vol. 18, No. 1, Summer 2001, pp. 5-21.
22. Hattori, F., Ohguro, T., Yokoo, M., Matsubara, S., and Yoshida, S., “Socialware: Multiagent Systems for Supporting Network Communities”, Communications of ACM, Vol. 42, No. 3, March 1999, pp. 55-61.



23. Huhns, M. N., and Singh, M. P., “Service-Oriented Computing: Key Concepts and Principles”, IEEE Internet Computing, Jan./Feb. 2005, pp. 75-81.
24. Inmon, W. H., “Building the Data Warehouse”, 3rd Edition, Wiley, New York, 2002.
25. Irani, Z., Themistocleous, M., and Love, P. E.D., “The Impact of Enterprise Application Integration on Information System Lifecycles”, Information & Management, vol. 41, 2003, pp. 177-187.
26. Jennings, N. R., “An Agent-Based Approach for Building Complex Software Systems”, Communications of The ACM, vol.44, no.4, April 2001, pp.35-41.
27. Kautz, H., Selman, B., and Shah, M., “Referral Web: Combining Social Networks and Collaborative Filtering”, Communications of ACM, Vol. 40, No. 3, March 1997, pp. 63-72.
28. Ketler, K., “Case-based Reasoning: An Introduction”, Expert Systems with Applications, Vol. 6, 1993, pp. 3-8.
29. Kimball, R. and Ross, M., “The Data Warehouse Toolkit: The Complete Guide to Dimensional Modeling”, 2nd edition, Wiley, New York, 2002.
30. Kobayashi, T., Tamaki, M., and Komoda, N., “Business process integration as a

solution to the implementation of supply chain management systems”,

Information & Management, vol. 40, 2003, pp.769-780.

31. Lee, J., Siau, K., and Hong, S., “Enterprise Integration with ERP and EAI”,
Communications of ACM, Vol. 46, No. 6, Feb. 2003, pp. 55-60.
32. Maes, P., “Agents That Reduce Work and Information Overload”,
Communications of ACM, Vol. 37, No. 7, 1994, pp.31-40.
33. Maes, P., Guttman, R. H., and Moukas, A. G., “Agents That Buy and Sell”,
Communications of ACM, Vol. 42, No. 3, March 1999, pp.81-91.
34. Massey, A. P., Montoya-Weiss, M. M., and Holcom, K., “Re-engineering the
Customer Relationship: Leveraging Knowledge Assets at IBM”, Decision
Support Systems, vol.32, 2001, pp.155-170
35. Nonaka, I., “The Knowledge Creating Company”, Harvard Business Review, Vol.
79, No. 6, 1991, pp. 96-104.
36. Nonaka, I., and Takeuchi, H., “The Knowledge Creating Company: How
Japanese Companies Create the Dynamics of Innovation”, Oxford University
Press, 1995.
37. O’Dell, C., and Grayson, C.J., “If only we knew what we know: Identification

and Transfer of Internal Best Practices”, California Management Review, Vol. 40, No. 3, 1998, pp. 154-174.

38. Papazoglou, M. P., “Agent-Oriented Technology in Support of e-Business”, Communications of The ACM, vol.44, no.4, April 2001, pp.71-77.
39. Papazoglou, M.P., “Service-Oriented Computing: Concepts, Characteristics and Directions”, IEEE Proceedings of the Fourth International Conference on Web Information Systems Engineering (WISE’03), 2003.
40. Peppers, D. & Rogers, M., “Is your company ready for one-to-one marketing”, Harvard Business Review 1999, pp. 151-160.
41. Reichheld, F. F., and Teal, T.A., “The Loyalty Effect: The Hidden Force Behind Growth, Profits, and Lasting Values”, Harvard Business School Press, Boston, MA, 1996.
42. Rigby, D. K., Reichheld, F. F., and Schefter, P., “Avoid the Four Perils of CRM”, Harvard Business Review, Feb. 2002, pp.101-109.
43. Sen, A., and Sinha, A. P., “A comparison of Data Warehousing Methodologies”, Communications of The ACM, Vol. 48, No. 3, March 2005, pp. 79-84.
44. Shaw, M. J., Subramaniam, C., Tan, G. W., and Welge, M. E., “Knowledge

Management and Data Mining for Marketing”, Decision Support Systems, vol. 31, 2001, pp. 127-137.

45. Watson, I., and Marir, F., “Case-based Reasoning: A Review”, The Knowledge Engineering Review, Vol. 9, No. 4, 1994, pp. 327-354.

46. “Extensible Markup Language (XML) 1.0 (Second Edition)”, W3C Recommendation, 6 Oct. 2000 (See <http://www.w3.org/TR/2000/REC-xml-20001006>.)

47. “State of the CIO 2006 - Research Reports”, CIO Magazine, 2006,

<http://www2.cio.com/research/surveyreport.cfm?id=101>

48. “Web Services Architecture”, W3C Working Group Note, 11 Feb. 2004, (see

<http://www.w3.org/TR/ws-arch/>)

附錄、資訊系統服務介面設計參考資料

表 A- 1、請求『障礙申訴』服務的合約內容

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
elementFormDefault="qualified">
  <xs:element name="Request">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="Head"/>
        <xs:element ref="Body"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="Head">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="SourceSystemID"/>
        <xs:element ref="SourceMessageID"/>
        <xs:element ref="DestSystemID"/>
        <xs:element ref="ServiceName"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="SourceSystemID" type="xs:NCName"/>
  <xs:element name="SourceMessageID" type="xs:NCName"/>
  <xs:element name="DestSystemID" type="xs:NCName"/>
  <xs:element name="ServiceName" type="xs:NCName"/>
  <xs:element name="Body">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="ResourceNum"/>
        <xs:element ref="ProblemCode"/>
        <xs:element ref="ProblemNote"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

```

```

</xs:element>
<xs:element name="ResourceNum" type="xs:NMTOKEN" />
<xs:element name="ProblemCode" type="xs:NCName" />
<xs:element name="ProblemNote" type="xs:NCName" />
</xs:schema>

```

訊息範例

```

<Request>
  <Head>
    <SourceSystemID>SC</SourceSystemID>
    <SourceMessageID>SC1009876</SourceMessageID>
    <DestSystemID>PH</DestSystemID>
    <ServiceName>RegisterTrouble</ServiceName>
  </Head>
  <Body>
    <ResourceNum>225D825123</ResourceNum>
    <ProblemCode>NOP</ProblemCode>
    <ProblemNote>時斷時續</ProblemNote>
  </Body>
</Request>

```

表 A- 2、『障礙申訴』服務回應的合約內容

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
elementFormDefault="qualified">
  <xs:element name="Response">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="Head"/>
        <xs:element ref="Body"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="Head">

```

```

<xs:complexType>
  <xs:sequence>
    <xs:element ref="SourceSystemID"/>
    <xs:element ref="DestSystemID"/>
    <xs:element ref="DestMessageID"/>
    <xs:element ref="ServiceName"/>
  </xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="SourceSystemID" type="xs:NCName"/>
<xs:element name="DestSystemID" type="xs:NCName"/>
<xs:element name="DestMessageID" type="xs:NCName"/>
<xs:element name="ServiceName" type="xs:NCName"/>
<xs:element name="Body">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="Result"/>
      <xs:element ref="ResultNote"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="Result" type="xs:boolean"/>
<xs:element name="ResultNote" type="xs:NCName"/>
</xs:schema>

```

訊息範例

```

<Response>
  <Head>
    <SourceSystemID>PH</SourceSystemID>
    <DestSystemID>SC</DestSystemID>
    <DestMessageID>SC1009876</DestMessageID>
    <ServiceName>RegisterTrouble</ServiceName>
  </Head>
  <Body>
    <Result>>false</Result>
    <ResultNote>欠費停話</ResultNote>
  </Body>
</Response>

```

表 A- 3、請求『查詢障礙處理進度』服務的合約內容

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
elementFormDefault="qualified">
  <xs:element name="Request">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="Head"/>
        <xs:element ref="Body"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="Head">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="SourceSystemID"/>
        <xs:element ref="SourceMessageID"/>
        <xs:element ref="DestSystemID"/>
        <xs:element ref="ServiceName"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="SourceSystemID" type="xs:NCName"/>
  <xs:element name="SourceMessageID" type="xs:NCName"/>
  <xs:element name="DestSystemID" type="xs:NCName"/>
  <xs:element name="ServiceName" type="xs:NCName"/>
  <xs:element name="Body">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="EnterpriseID"/>
        <xs:element ref="EnterpriseName"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="EnterpriseID" type="xs:NCName"/>
  <xs:element name="EnterpriseName" type="xs:NCName"/>
</xs:schema>

```

訊息範例

```
<Request>
  <Head>
    <SourceSystemID>SC</SourceSystemID>
    <SourceMessageID>SC1001234</SourceMessageID>
    <DestSystemID>PH</DestSystemID>
    <ServiceName>TrackStatus</ServiceName>
  </Head>
  <Body>
    <EnterpriseID>A987654321</EnterpriseID>
    <EnterpriseName>中國信託銀行</EnterpriseName>
  </Body>
</Request>
```

表 A- 4、『查詢障礙處理進度』服務回應的合約內容

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
elementFormDefault="qualified">
  <xs:element name="Response">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="Head"/>
        <xs:element ref="Body"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="Head">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="SourceSystemID"/>
        <xs:element ref="DestSystemID"/>
        <xs:element ref="DestMessageID"/>
        <xs:element ref="ServiceName"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

```

    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="SourceSystemID" type="xs:NCName"/>
<xs:element name="DestSystemID" type="xs:NCName"/>
<xs:element name="DestMessageID" type="xs:NCName"/>
<xs:element name="ServiceName" type="xs:NCName"/>
<xs:element name="Body">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="TroubleList" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="TroubleList">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="unbounded"
ref="TroubleTicket" />
    </xs:sequence>
    <xs:attribute name="NumOfTickets" use="required"
type="xs:integer"/>
  </xs:complexType>
</xs:element>
<xs:element name="TroubleTicket">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="TicketID"/>
      <xs:element ref="TicketTime"/>
      <xs:element ref="ResourceNum"/>
      <xs:element ref="RootCauseID"/>
      <xs:element ref="RootCauseName"/>
      <xs:element ref="Status"/>
      <xs:element ref="StaffName"/>
      <xs:element ref="StaffMobile"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

```

<xs:element name="TicketID" type="xs:NCName"/>
<xs:element name="TicketTime" type="xs:string"/>
<xs:element name="ResourceNum" type="xs:NMTOKEN"/>
<xs:element name="RootCauseID" type="xs:NCName"/>
<xs:element name="RootCauseName" type="xs:NCName"/>
<xs:element name="Status" type="xs:NCName"/>
<xs:element name="StaffName" type="xs:NCName"/>
<xs:element name="StaffMobile" type="xs:string"/>
</xs:schema>

```

訊息範例

```

<Response>
  <Head>
    <SourceSystemID>PH</SourceSystemID>
    <DestSystemID>SC</DestSystemID>
    <DestMessageID>SC1001234</DestMessageID>
    <ServiceName>TrackStatus</ServiceName>
  </Head>
  <Body>
    <TroubleList NumOfTickets="2">
      <TroubleTicket>1896
        <TicketID>TP12345678</TicketID>
        <TicketTime>95-05-18 10:00</TicketTime>
        <ResourceNum>02-23445678</ResourceNum>
        <RootCauseID>DC</RootCauseID>
        <RootCauseName>斷線</RootCauseName>
        <Status>測試中</Status>
        <StaffName>葉文宏</StaffName>
        <StaffMobile>0933123456</StaffMobile>
      </TroubleTicket>
      <TroubleTicket>
        <TicketID>TC23456789</TicketID>
        <TicketTime>95-05-18 12:10</TicketTime>
        <ResourceNum>04-22445566</ResourceNum>
        <RootCauseID>DCX</RootCauseID>
        <RootCauseName>時斷時續</RootCauseName>
        <Status>查修人員處理中</Status>
        <StaffName>文宏</StaffName>
      </TroubleTicket>
    </TroubleList>
  </Body>
</Response>

```

```

        <StaffMobile>0933654321</StaffMobile>
    </TroubleTicket>
</TroubleList>
</Body>
</Response>

```

表 A- 5、發佈『電路服務狀態』服務的合約內容

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
elementFormDefault="qualified">
  <xs:element name="Publish">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="Head"/>
        <xs:element ref="Body"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="Head">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="SourceSystemID"/>
        <xs:element ref="SourceMessageID"/>
        <xs:element ref="DestSystemID"/>
        <xs:element ref="ServiceName"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="SourceSystemID" type="xs:NCName"/>
  <xs:element name="SourceMessageID" type="xs:NCName"/>
  <xs:element name="DestSystemID" type="xs:string"/>
  <xs:element name="ServiceName" type="xs:NCName"/>
  <xs:element name="Body">
    <xs:complexType>
      <xs:sequence>

```

```

    <xs:element ref="ResourceNum"/>
    <xs:element ref="Customer"/>
    <xs:element ref="CustomerAddr"/>
    <xs:element ref="ServiceState"/>
    <xs:element ref="TimeOfStateChanged"/>
  </xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="ResourceNum" type="xs:NMTOKEN"/>
<xs:element name="Customer">
  <xs:complexType>
    <xs:simpleContent>
      <xs:extension base="xs:NCName">
        <xs:attribute name="id" use="required"
type="xs:NCName"/>
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>
</xs:element>
<xs:element name="CustomerAddr" type="xs:NCName"/>
<xs:element name="ServiceState">
  <xs:complexType>
    <xs:simpleContent>
      <xs:extension base="xs:NCName">
        <xs:attribute name="code" use="required"
type="xs:NCName"/>
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>
</xs:element>
<xs:element name="TimeOfStateChanged" type="xs:string"/>
</xs:schema>

```

訊息範例

```

<Publish>
  <Head>
    <SourceSystemID>BILL</SourceSystemID>
    <SourceMessageID>BL12345678</SourceMessageID>

```

```

    <DestSystemID>*</DestSystemID>
    <ServiceName>ServiceStateChanged</ServiceName>
  </Head>
  <Body>
    <ResourceNum>225D825123</ResourceNum>
    <Customer id="A987654321">中國信託銀行</Customer>
    <CustomerAddr>台北市信義路100號</CustomerAddr>
    <ServiceState code="suspend">欠費停話</ServiceState>
    <TimeOfStateChanged>95-05-19
10:10</TimeOfStateChanged>
  </Body>
</Publish>

```

表 A- 6、發佈『資源調整作業資訊』服務的合約內容

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
elementFormDefault="qualified">
  <xs:element name="Publish">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="Head"/>
        <xs:element ref="Body"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="Head">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="SourceSystemID"/>
        <xs:element ref="SourceMessageID"/>
        <xs:element ref="DestSystemID"/>
        <xs:element ref="ServiceName"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

```

<xs:element name="SourceSystemID" type="xs:NCName"/>
<xs:element name="SourceMessageID" type="xs:NCName"/>
<xs:element name="DestSystemID" type="xs:string"/>
<xs:element name="ServiceName" type="xs:NCName"/>
<xs:element name="Body">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="StartTime"/>
      <xs:element ref="EstimateFinishTime"/>
      <xs:element ref="ClosedTime"/>
      <xs:element ref="ResourceState"/>
      <xs:element ref="EffectuatedResources"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="StartTime" type="xs:string"/>
<xs:element name="EstimateFinishTime" type="xs:string"/>
<xs:element name="ClosedTime" type="xs:string"/>
<xs:element name="ResourceState">
  <xs:complexType>
    <xs:simpleContent>
      <xs:extension base="xs:NCName">
        <xs:attribute name="code" use="required"
type="xs:NCName"/>
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>
</xs:element>
<xs:element name="EffectuatedResources">
  <xs:complexType>
    <xs:sequence>
      <xs:element maxOccurs="unbounded" ref="ResourceNum"/>
    </xs:sequence>
    <xs:attribute name="size" use="required"
type="xs:integer"/>
  </xs:complexType>
</xs:element>
<xs:element name="ResourceNum" type="xs:NMTOKEN"/>

```

```
</xs:schema>
```

訊息範例

```
<Publish>
  <Head>
    <SourceSystemID>RM</SourceSystemID>
    <SourceMessageID>RM12345678</SourceMessageID>
    <DestSystemID>*</DestSystemID>
    <ServiceName>ResourceStateChanged</ServiceName>
  </Head>
  <Body>
    <StartTime>95-05-19 10:00</StartTime>
    <EstimateFinishTime>95-05-19
18:00</EstimateFinishTime>
    <ClosedTime></ClosedTime>
    <ResourceState code="cablechanged">電纜改接
</ResourceState>
    <EffectuatedResources size="3">
      <ResourceNum>225D825123</ResourceNum>
      <ResourceNum>225D825124</ResourceNum>
      <ResourceNum>225D825125</ResourceNum>
    </EffectuatedResources>
  </Body>
</Publish>
```

表 A- 7、請求『擷取用戶障礙資訊』服務的合約內容

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
elementFormDefault="qualified">
  <xs:element name="Request">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="Head"/>
        <xs:element ref="Body"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

```

    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="Head">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="SourceSystemID"/>
      <xs:element ref="SourceMessageID"/>
      <xs:element ref="DestSystemID"/>
      <xs:element ref="ServiceName"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="SourceSystemID" type="xs:NCName"/>
<xs:element name="SourceMessageID" type="xs:NCName"/>
<xs:element name="DestSystemID" type="xs:NCName"/>
<xs:element name="ServiceName" type="xs:NCName"/>
<xs:element name="Body">
  <xs:complexType>
    <xs:sequence>
      <!-- 1 for Test, 2 for Repair -->
      <xs:element ref="TicketType"/>
      <xs:element ref="Turf"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="TicketType" type="xs:integer"/>
<xs:element name="Turf" type="xs:string"/>
</xs:schema>

```

訊息範例

```

<Request>
  <Head>
    <SourceSystemID>FORCE</SourceSystemID>
    <SourceMessageID>F012345678</SourceMessageID>
    <DestSystemID>PH</DestSystemID>
    <ServiceName>ListTickets</ServiceName>
  </Head>

```

```

<Body>
  <!-- 1 for Test, 2 for Repair -->
  <TicketType>2</TicketType>
  <Turf>*</Turf>
</Body>
</Request>

```

表 A- 8、『擷取用戶障礙資訊』服務回應的合約內容

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
elementFormDefault="qualified">
  <xs:element name="Response">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="Head"/>
        <xs:element ref="Body"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="Head">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="SourceSystemID"/>
        <xs:element ref="DestSystemID"/>
        <xs:element ref="DestMessageID"/>
        <xs:element ref="ServiceName"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="SourceSystemID" type="xs:NCName"/>
  <xs:element name="DestSystemID" type="xs:NCName"/>
  <xs:element name="DestMessageID" type="xs:NCName"/>
  <xs:element name="ServiceName" type="xs:NCName"/>
  <xs:element name="Body">
    <xs:complexType>
      <xs:sequence>

```

```

        <xs:element ref="TroubleList"/>
    </xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="TroubleList">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="unbounded"
ref="TroubleTicket"/>
        </xs:sequence>
        <xs:attribute name="NumOfTickets" use="required"
type="xs:integer"/>
    </xs:complexType>
</xs:element>
<xs:element name="TroubleTicket">
    <xs:complexType>
        <xs:sequence>
            <xs:element ref="TicketID"/>
            <xs:element ref="TicketTime"/>
            <xs:element ref="ResourceNum"/>
            <xs:element ref="RootCauseID"/>
            <xs:element ref="RootCauseName"/>
            <xs:element ref="StaffName"/>
            <xs:element ref="StaffMobile"/>
            <xs:element ref="Customer"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="TicketID" type="xs:NCName"/>
<xs:element name="TicketTime" type="xs:string"/>
<xs:element name="ResourceNum" type="xs:NMTOKEN"/>
<xs:element name="RootCauseID" type="xs:NCName"/>
<xs:element name="RootCauseName" type="xs:NCName"/>
<xs:element name="StaffName">
    <xs:complexType>
        <xs:simpleContent>
            <xs:extension base="xs:NCName">
                <xs:attribute name="id" use="required"

```

```

type="xs:NCName" />
    </xs:extension>
    </xs:simpleContent>
</xs:complexType>
</xs:element>
<xs:element name="StaffMobile" type="xs:string" />
<xs:element name="Customer">
    <xs:complexType>
        <xs:sequence>
            <xs:element ref="EnterpriseID" />
            <xs:element ref="EnterpriseName" />
            <xs:element ref="Contact" />
            <xs:element ref="Phone" />
            <xs:element ref="Address" />
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="EnterpriseID" type="xs:NCName" />
<xs:element name="EnterpriseName" type="xs:NCName" />
<xs:element name="Contact" type="xs:string" />
<xs:element name="Phone" type="xs:NMTOKEN" />
<xs:element name="Address" type="xs:NCName" />
</xs:schema>

```

訊息範例

```

<Response>
    <Head>
        <SourceSystemID>PH</SourceSystemID>
        <DestSystemID>FORCE</DestSystemID>
        <DestMessageID>FO12345678</DestMessageID>
        <ServiceName>ListTickets</ServiceName>
    </Head>
    <Body>
        <TroubleList NumOfTickets="2">
            <TroubleTicket>
                <TicketID>TP12345678</TicketID>
                <TicketTime>95-05-18 10:00</TicketTime>
                <ResourceNum>225D825123</ResourceNum>
            </TroubleTicket>
        </TroubleList>
    </Body>
</Response>

```

```
<RootCauseID>DC</RootCauseID>
<RootCauseName>斷線</RootCauseName>
<StaffName id="A02857">葉文宏</StaffName>
<StaffMobile>0933123456</StaffMobile>
<Customer>
  <EnterpriseID>A987654321</EnterpriseID>
  <EnterpriseName>中國信託銀行
</EnterpriseName>
  <Contact>Mr. YEH</Contact>
  <Phone>02-22113456</Phone>
  <Address>台北市信義路四段100號</Address>
</Customer>
</TroubleTicket>
<TroubleTicket>
  <TicketID>TP55667788</TicketID>
  <TicketTime>95-05-18 10:00</TicketTime>
  <ResourceNum>225D825321</ResourceNum>
  <RootCauseID>DC</RootCauseID>
  <RootCauseName>斷線</RootCauseName>
  <StaffName id="A02899">文宏</StaffName>
  <StaffMobile>0933123478</StaffMobile>
  <Customer>
    <EnterpriseID>A987654321</EnterpriseID>
    <EnterpriseName>中國信託銀行
  </EnterpriseName>
    <Contact>Mr. CHEN</Contact>
    <Phone>02-22112344</Phone>
    <Address>台北市杭州南路10號</Address>
  </Customer>
</TroubleTicket>
</TroubleList>
</Body>
</Response>
```

表 A- 9、請求『障礙處理回報』服務的合約內容

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
elementFormDefault="qualified">
  <xs:element name="Request">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="Head"/>
        <xs:element ref="Body"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="Head">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="SourceSystemID"/>
        <xs:element ref="SourceMessageID"/>
        <xs:element ref="DestSystemID"/>
        <xs:element ref="ServiceName"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="SourceSystemID" type="xs:NCName"/>
  <xs:element name="SourceMessageID" type="xs:NCName"/>
  <xs:element name="DestSystemID" type="xs:NCName"/>
  <xs:element name="ServiceName" type="xs:NCName"/>
  <xs:element name="Body">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="TroubleTicket"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="TroubleTicket">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="TicketID"/>
        <xs:element ref="ResourceNum"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

```

```

        <xs:element ref="RootCause" />
        <xs:element ref="StaffName" />
    </xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="TicketID" type="xs:NCName" />
<xs:element name="ResourceNum" type="xs:NMTOKEN" />
<xs:element name="RootCause">
    <xs:complexType>
        <xs:simpleContent>
            <xs:extension base="xs:NCName">
                <xs:attribute name="id" use="required"
type="xs:NCName" />
            </xs:extension>
        </xs:simpleContent>
    </xs:complexType>
</xs:element>
<xs:element name="StaffName">
    <xs:complexType>
        <xs:simpleContent>
            <xs:extension base="xs:NCName">
                <xs:attribute name="id" use="required"
type="xs:NCName" />
            </xs:extension>
        </xs:simpleContent>
    </xs:complexType>
</xs:element>
</xs:schema>

```

訊息範例

```

<Request>
  <Head>
    <SourceSystemID>FORCE</SourceSystemID>
    <SourceMessageID>F012345678</SourceMessageID>
    <DestSystemID>PH</DestSystemID>
    <ServiceName>UpdateTicket</ServiceName>
  </Head>
  <Body>

```

```

    <TroubleTicket>
      <TicketID>TP12345678</TicketID>
      <ResourceNum>225D825123</ResourceNum>
      <RootCause id="CPE">客戶端數據設備壞掉</RootCause>
      <StaffName id="A02857">葉文宏</StaffName>
    </TroubleTicket>
  </Body>
</Request>

```

表 A- 10、『用戶障礙資訊發佈』服務的合約內容

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
elementFormDefault="qualified">
  <xs:element name="Publish">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="Head"/>
        <xs:element ref="Body"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="Head">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="SourceSystemID"/>
        <xs:element ref="SourceMessageID"/>
        <xs:element ref="DestSystemID"/>
        <xs:element ref="ServiceName"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="SourceSystemID" type="xs:NCName"/>
  <xs:element name="SourceMessageID" type="xs:NCName"/>
  <xs:element name="DestSystemID" type="xs:string"/>
  <xs:element name="ServiceName" type="xs:NCName"/>
  <xs:element name="Body">

```

```

<xs:complexType>
  <xs:sequence>
    <xs:element ref="TroubleState"/>
    <xs:element ref="ResourceNum"/>
    <xs:element ref="StartTime"/>
    <xs:element ref="ClosedTime"/>
    <xs:element ref="Customer"/>
  </xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="TroubleState" type="xs:NCName"/>
<xs:element name="ResourceNum" type="xs:NMTOKEN"/>
<xs:element name="StartTime" type="xs:string"/>
<xs:element name="ClosedTime" type="xs:string"/>
<xs:element name="Customer">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="EnterpriseID"/>
      <xs:element ref="EnterpriseName"/>
      <xs:element ref="Contact"/>
      <xs:element ref="Phone"/>
      <xs:element ref="Address"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="EnterpriseID" type="xs:NCName"/>
<xs:element name="EnterpriseName" type="xs:NCName"/>
<xs:element name="Contact" type="xs:string"/>
<xs:element name="Phone" type="xs:NMTOKEN"/>
<xs:element name="Address" type="xs:NCName"/>
</xs:schema>

```

訊息範例

```

<Publish>
  <Head>
    <SourceSystemID>PH</SourceSystemID>
    <SourceMessageID>PH02040909</SourceMessageID>
    <DestSystemID>*</DestSystemID>
  </Head>
</Publish>

```

```
<!--Notify message when trouble open or closed-->
<ServiceName>TroubleNotify</ServiceName>
</Head>
<Body>
  <!-- state=open/closed -->
  <TroubleState>closed</TroubleState>
  <ResourceNum>225D825123</ResourceNum>
  <StartTime>95-05-19 10:00</StartTime>
  <ClosedTime>95-05-19 17:30</ClosedTime>
  <Customer>
    <EnterpriseID>A987654321</EnterpriseID>
    <EnterpriseName>中國信託銀行</EnterpriseName>
    <Contact>Mr. YEH</Contact>
    <Phone>02-22113456</Phone>
    <Address>台北市信義路四段100號</Address>
  </Customer>
</Body>
</Publish>
```

