

行政院國家科學委員會專題研究計畫進度報告

以仲介者為基礎之廣域軟體再利用研究與工具開發 Research and Design of a Broker-based Wide-Area Software Reuse Process and Environment

計畫類別：個別型計畫

計畫編號：NSC 89-2213-E-009-013

執行期間：88年 8 月 1 日至 89年 7月 31 日

計畫主持人：鍾乾癸 教授

共同主持人：

處理方式： 可立即對外提供參考
 , 一年後可對外提供參考
 , 兩年後可對外提供參考
 (必要時，本會得展延發表時限)

執行單位：國立交通大學資訊工程研究所

中華民國 八 十 九 年 五 月 二 十 三 日

摘要

軟體再利用技術運用效益之大小與元件庫中所包含的元件數量以及其所屬應用領域多寡有密切關係。近來由於 Internet 及 WWW 環境下相關技術的蓬勃發展，軟體產品的發展與行銷策略已引起重大變革。在眾多的軟體發展技術中，軟體再利用技術已被認為是有效提升軟體產能與品質的方法。本計畫將軟體再利用的觀念與現行的 Internet/WWW 環境結合，創造出一個廣域再利用的環境，讓使用者透過 Broker 的幫助，再利用 Internet 上眾多 Vendor 所提供的再利用資源。在第一年的研究成果中，本實驗室已完成一“以仲介者為基礎之 Internet/WWW 環境下的軟體再利用 (Broker-based Software Reuse on Internet/WWW) 之環境架構”。延續此研究成果，我們將繼續對在 Internet/WWW 此一廣域環境下進行需求規格再利用進行之性質、再利用的流程以及所需之關鍵技術進行研究，並以第一年完成的 Broker 雛形系統為基礎，擴充此雛形系統以支援廣域的需求規格再利用。

目錄

一、計畫背景及目的介紹

由於資訊硬體產品的效能及功能大幅提升及價格大眾化，導致各類軟體需求愈來愈多，功能也愈多樣化且複雜，因而造成發展軟體時常遭遇開發過程費時、交貨延遲、產能不易提升，且品質不易確保等問題。為解決軟體發展之問題，許多軟體工程技術相繼提出以提升軟體的生產力與品質，其中軟體再利用技術（Software Reuse）已是目前公認最能提升軟體發展之產能及品質之軟體工程技術[1] [3]。

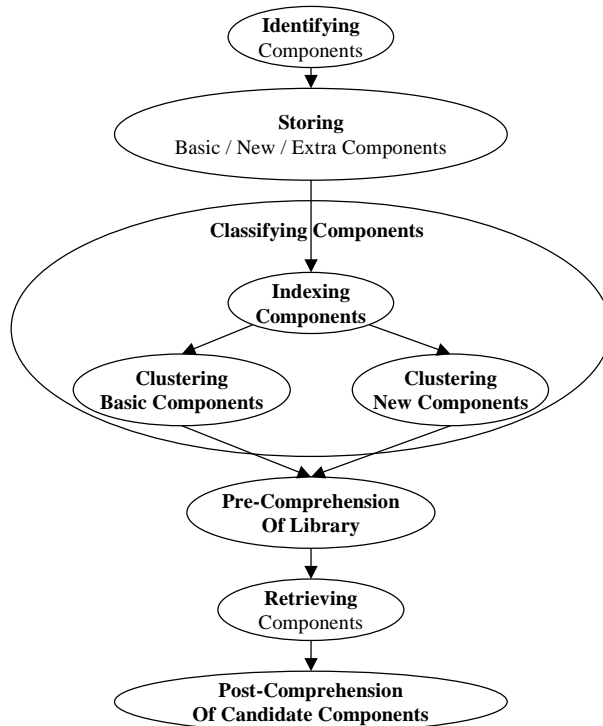
所謂軟體再利用技術意指在軟體發展過程中軟體開發者有系統的利用以往所發展完成的軟體產物（Software Artifacts）加以組合發展新的軟體系統。軟體開發者首先透過搜尋引擎（Search Engine）至元件庫（Reusable Component Repository）中尋找合用的軟體元件並了解其功能行為，進而篩選出最合適的軟體元件後予以決定直接使用或修改後再利用，因此軟體開發者不需自行開發軟體系統之所有組成單元，不但可以節省開發成本、時間及提升產能，且由於現存之軟體元件均已透過驗證及測試，對整體系統之品質驗證也可節省不少時間。

軟體再利用技術運用效益之大小與元件庫中所包含的元件數量以及其所屬應用領域多寡有密切關係。近來由於 Internet 及 WWW 環境下相關技術的蓬勃發展，使用者可以很容易的透過 Internet/WWW 快速取得所需的資訊，此一新興的通訊傳輸管道對於軟體產品的發展與行銷策略已引起重大變革，愈來愈多的軟體公司都會將自己所開發的軟體產品及相關說明文件放置在其網路伺服器上讓使用者免費或付費後下載使用。在此環境下，軟體開發者在軟體發展的各個階段中，除了可以再利用自身或公司內部軟體元件庫所提供的元件外，也可以透過 Internet/WWW 以及相關的再利用機制，至其他公司的軟體元件庫中搜尋合適的軟體元件並購買使用。如此一來，可再利用的軟體元件不再只侷限於自身或公司內部的元件庫中，取而代之的是一群分散於各地的軟體元件庫，軟體元件的數量及所涵蓋的領域大增，因此，如果可以利用 Internet/WWW 技術將全世界各式各樣不同領域的軟體元件庫連結以建立一數量龐大、涵蓋領域豐富的軟體元件庫，應可解決軟體元件數量及涵蓋領域不夠豐富之瓶頸，大幅提升實施軟體再利用技術的效益。

可再利用的軟體產物的種類包含軟體發展過程各階段中所產生的結果，例如軟體需求規格文件、設計規格文件、軟體架構、程式碼等等皆可視為軟體產物，因此從軟體發展過程各階段中所產生產物的角度來看，軟體再利用的層次主要可以分為：程式碼再利用、設計再利用以及需求規格再利用。在軟體再利用以往的研究成果中，以程式碼再利用所獲得的成果最為豐碩[4][5][6]，且在軟體產業界，也有許多程式碼再利用應用成功的案例，例如 IBM 的 systematic reuse[4]策略，

及日本的軟體工廠（Software Factory）[7]等等。

有關在 Web 上進行軟體再利用的研究，目前已有學者針對程式碼元件再利用的層次提出了一個 SOORLS（Summary-based Object-Oriented Reuse Library System）[8] 的系統。SOORLS 是一個 Web 上的物件導向軟體元件庫，主要提供給軟體發展者一個能在 Web 上再利用現有的軟體元件開發軟體的軟體元件庫，由於 Web 上軟體元件的數目眾多，因此 SOORLS 提供一套自動分類的方法，其再利用的流程如圖一-1 所示，軟體元件先經過元件識別（Identifying）萃取出可供再利用的資訊後將元件儲存（Storing）到 SOORLS 的元件庫中，透過自動化的群組分類（cluster-based）技術將元件庫中的元件分類及建立索引（Index）。所謂的 cluster-based 分類技術，是先將元件庫中的軟體元件依照元件彼此間的相似程度自動分成數個群組（cluster），而對於新加入元件庫的元件，則會與先前挑選出代表各群組的種子元件（seed）比較彼此間的相似性並再歸類。當軟體開發者欲透過 SOORLS 尋找可再利用的軟體元件時，則再透過 summary-based 擷取機制，先了解每個群組的摘要內容（summary），此摘要內容是由在分類時所挑選出代表群組的一群關鍵字所組成，在選取出想要的群組並輸入欲尋找元件之查詢條件後，系統便會自動搜尋比對並擷取出元件庫中符合查詢條件的軟體元件供軟體開發者選擇。

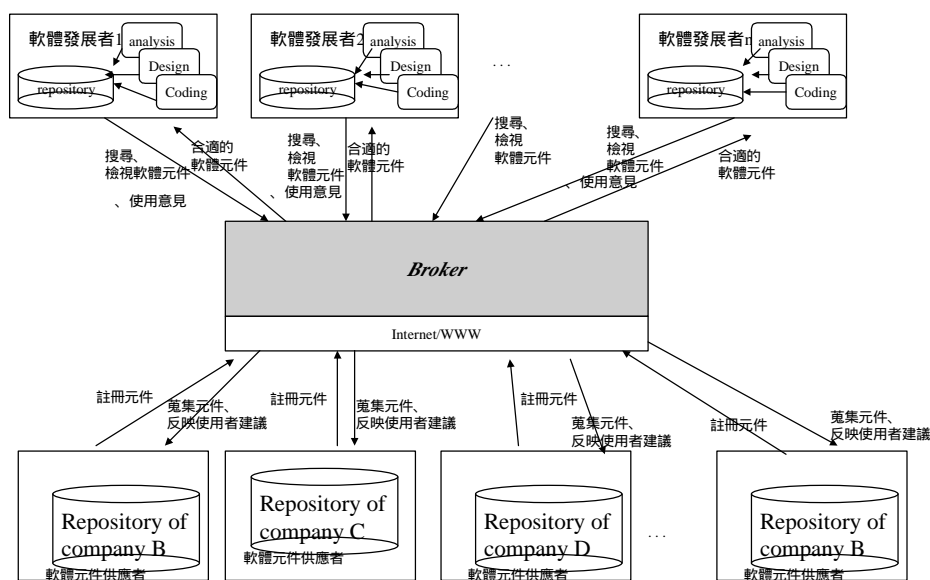


圖一-1 Reuse process model in SOORLS

除了上述之 SOORLS 軟體元件庫之外，本計畫第一年之研究成果即是針對在 Internet/WWW 環境下進行軟體再利用提出了一個“以仲介者為基礎之

Internet/WWW 環境下的軟體再利用 (Broker-based Software Reuse on Internet/WWW) 之環境架構 “，如圖一-2 所示。此架構主要是以程式碼再利用為研究對象，研究在 Internet/WWW 環境下，一個支援軟體再利用的仲介者系統 (Broker) 所需的有關程式碼元件封裝、分類、搜尋、檢視、品質測量、註冊、目錄庫管理等關鍵技術。所謂的仲介者 (Broker) 即是負責蒐集及管理軟體元件供應商所生產的軟體元件及相關資訊，軟體開發者透過此 Broker 系統便可找到各類應用領域的軟體元件，而不需再一一搜尋遍佈各地的軟體元件庫。

此 Broker 系統之目的便是在 Internet/WWW 環境下提供軟體開發者一個具有標準化的軟體元件封裝、嚴格的元件品質管制、良好的元件分類、搜尋與檢視措施及健全的元件交易管理制度的再利用環境。由於有著良好的元件分類架構，軟體開發者可以很快速的找到他想要的軟體元件，配合著標準的封裝格式，使得發展者不需學習不同的元件搜尋及檢視方式，不但可縮短選擇元件的時間，同時也使得元件易於使用。再加上這些元件已經過 Broker 嚴格的品質管制，發展者使用元件的信心與意願也隨之增強。然後在健全的交易制度下，製造商會更踴躍開發各類元件，進而豐富了軟體元件的種類。



圖一-2 Framework of Broker-based Wide Area Software Reuse on Internet/WWW Environment

由本計畫第一年之研究成果已可證實在 Internet/ WWW 環境下進行軟體再利用確實可行，並已針對程式碼再利用的層次設計一在 Internet/WWW 下，支援程式碼再利用之 Broker 系統，然而程式碼撰寫只不過是佔整個軟體開發成本的 8% 到 10% [9]，若要提升軟體再利用的效益，其解決之道，便是要將軟體再利用的層次提高，即要做到軟體需求規格、設計規格元件的再利用；也就是在開發軟體時，若能越早再利用現成的軟體產物，所獲得的效益也就越大 [10]。例如，在制定新軟體的軟體規格時，若能利用大部分現存的軟體規格元件，不僅可以省下分析成本，且其對應的軟體架構、演算法、程式碼及測試資料等皆有可能被再利

用，進而可以省下不少的設計、製作的人力與成本。然而目前仍尚未發現到在 Internet/WWW 的環境下，探討有關軟體需求規格層次再利用技術方面的研究。

因此，延續第一年的研究成果，我們將繼續研究在 Internet/WWW 此一廣域環境下，軟體發展者如何在分析階段再利用現存的需求規格元件以降低軟體分析階段所耗費的時間與成本。由於需求規格元件亦與程式碼元件經歷相同的再利用程序，如註冊、品質檢測、分類、搜尋、檢視等，但因需求規格的內容以及再利用的時機、先決條件均不相同，我們將據此研究需求規格元件在 Internet/WWW 環境下進行再利用之需求為何，並比較與程式碼元件之封裝、分類、搜尋、檢視、品管等關鍵技術需求之相同及相異處，以制定完整的再利用程序以及系統架構。並以第一年完成的 Broker 雛形系統為基礎，增加或調整相關的關鍵技術以及工具，擴充此雛形系統以支援廣域的需求規格再利用。

二、在 Web 上進行需求規格再利用之探討

從前述之背景介紹可知，在 Web 上進行需求規格再利用是一新興的研究方向，目前尚未發現任何在 Web 上的需求規格再利用系統，本章先簡單介紹目前有關於需求規格再利用系統方面的研究現況，並從中歸納出一般公司或組織內部應用需求規格再利用技術的流程，然後再對傳統的需求規格再利用環境與在 Web 環境下進行需求規格再利用之差異做一比較，以作為接下來在設計一 Web 上之需求規格再利用系統時之參考。

2.1 現有與需求規格再利用相關的研究

由於系統需求文件是用來作為客戶與系統分析師彼此溝通的媒介，其表示方式通常並無統一規定的格式，為求溝通的方便，系統需求往往是以自然語言的形式或圖形的方式表示。因此，欲達到需求規格再利用，便必須研究如何從非制式的需求文件抽取出有用的資訊以供再利用。目前有關於這方面技術的研究，主要可分成直接對系統需求規格的文字內容進行處理 (text processing) 及分析並從系統需求規格文件中萃取出有用的知識 (knowledge management) 兩個方向[11]。

第一種方式主要是針對以自然語言描述的系統需求規格文件，利用自動化的方式進行如語法 (syntactic) 及語意 (semantic) 分析、使用語彙 (lexical) 的統計及分析等處理，並據此建立索引 (index) 以利系統需求的搜尋及擷取。例如[12]的軟體再利用系統便提供一自動化的工具可用來處理以自然語言描述的系統需求規格及使用者的查詢條件。此種方式最大的好處是透過程式的自動處理，可大幅減輕管理再利用系統的人力負擔，但是系統的實用程度便視於該程式比對的精確度而定。

而第二種方式則是從系統需求規格文件中抽取出可作為代表的資訊儲存到元件庫中，並依其所屬的應用領域建立起對應的領域模型以作為分類及索引的依據。相關的研究早期有類比式 (Analogy) [13]及 F³[14]再利用方法的技術。類比法主要是將規格之抽象模式 (Abstract model) 存於儲存庫中，規格之抽象模式主要是在建立抽象化的系統模型，利用因果關係圖 (causal graph) 表達系統物件間的關係及系統目的，分析師再利用此抽象模式，以類比的推理技巧尋找與新問題的解決流程、目的及運用策略相似的抽象領域模式，並取出合適的需求規格元件進行類比推理，應用先前解決問題的知識，建立可解決新問題的需求規格文件。而 F³ 再利用技術是利用 Modeling 的觀念，根據問題的需求定義出對應的物件綱要模式 (class schema)，再根據此物件綱要模式到儲存庫中尋找相同或相似之物件綱要模式，並將其轉換到可再利用之軟體元件。

然而上述兩種方法存在以下兩種缺點：(1)將合適的模型轉換成具體的需求規格之成本及困難度相當高。(2)在系統分析的初期通常無法立即獲得完整的系統需求，因此很難建立完整的領域模型。在實際應用上，無法滿足部份需求就能再利用之要求。因此本實驗室的學長即針對早期需求規格再利用技術缺乏之處，發展了一個架構在 FDOOA 需求分析方法上的物件導向需求規格再利用程序與環境[15][16]，再由本實驗室范姜永益學長加以發展改良並提出一需求規格再利用系統[17]，以下便針對此需求規格再利用系統作一簡單介紹。

此需求規格再利用系統所研究的再利用的環境主要是自身或公司內部所擁有的規格元件庫，規格元件再利用的流程及方式則是架構在由本實驗室吳介銘學長所提出的“改良式 FDOOA 軟體需求分析方法”[18]之上。此外，本系統並針對實現規格再利用所需之規格元件的分類搜尋、驗證、編輯與修改等關鍵技術進行研究與探討，並提出一規格再利用之程序如下：

(1) 領域知識再利用

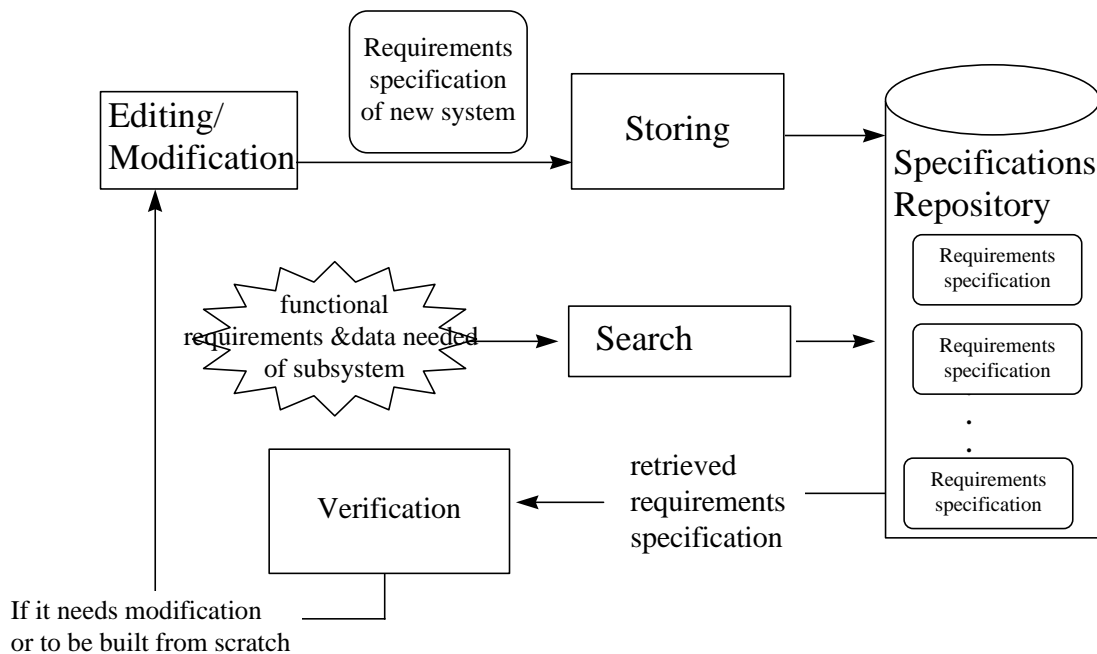
根據新系統的領域特質，參考相同或類似領域之下的系統領域知識，加速分析師訂定系統之服務需求。

(2) 規格再利用

根據子系統之服務需求，定義一組轉換特徵作為子系統規格的查詢語言，尋找相似的系統規格以資利用。若子系統中的系統功能，無法經由此系統規格加以修改定義，則分析師亦可進行功能規格之再利用程序。

(3) 組合再利用

當所有的子系統規格定義完畢，分析師可以擷取相同或相似的系統規格，參考其交互關係描述，將子系統由下往上，依序組成子系統及系統規格。



圖二-1 規格再利用系統架構圖

本實驗室並依此再利用程序，設計並製作一規格再利用環境，其組成包括規格元件儲存庫、規格元件儲存分類模組、快速雛形執行器及規格編輯器。其中，規格元件儲存庫負責儲存領域知識、規格單元內容、規格單元之間的連結關係以及分類搜尋用的索引架構。而規格元件儲存分類模組負責將新規格文件加以切割並建立適當的索引連結。而快速雛形執行器及規格編輯器則整合現有的 FDOOA 環境，提供規格元件驗證及編輯修改之能力。

2.2 需求規格再利用流程之探討

觀察目前已有具體成果之需求規格再利用相關研究，可發現其再利用系統使用的環境多半是位於組織或公司內部，可供再利用的系統需求規格的來源主要是自己或是公司內部自行開發的元件或系統，此種再利用方式，我們稱之為組織內再利用。接下來，藉由參考先前所研究之需求規格再利用系統[17]，我們將歸納出在傳統的組織內再利用環境下分析師進行需求規格再利用之流程。然後我們便可從此流程中找出在傳統環境下與在 Web 環境下進行需求規格再利用之差異。

要談需求規格再利用的流程，我們必須先從軟體需求分析的程序探討起，然後我們再從中找出進行需求規格再利用的時機以及再利用之軟體產物，最後便可歸納出需求規格再利用的流程。

2.2.1 軟體需求分析流程

軟體需求分析的目的，即是在找出客戶對軟體系統的需求，並將其撰寫成制式的軟體需求規格文件，以做為軟體設計階段及軟體開發完成後進行驗證的依據與標準。

分析師對一個系統作軟體需求分析時，首先會根據客戶提出的問題，收集與問題所在之應用領域相關的資料，藉由以前的發展經驗、相關文獻的閱讀及訪問問題領域上專家等方式，以對客戶的問題及所需的系統有一初步的認知與了解。然後透過與客戶間之溝通，分析師可徹底瞭解以往使用者處理問題之流程，並對包括軟體系統的目標、特性、以及軟體系統的設計限制等問題產生共識。接下來分析師便會與客戶討論客戶對系統之功能性的需求，並訂出系統必須提供之服務（Services）。若系統的問題範圍太大或太複雜，以致於不易很明確的決定出整個系統的需求，則分析師將對系統做進一步的分解，把大系統分解成幾個較小的子系統，子系統所牽涉的功能需求及資料範圍較小，也因此對其行為特性較容易瞭解，所提供之服務也較易訂出，藉由對子系統的充分瞭解來幫助整個系統的瞭解。例如一個學校的校務行政系統所包含的範圍很大，要直接制定其系統之功能需求並不十分容易，因此在討論一個學校的校務行政系統之系統需求時，我們可以先將其分解成教務行政子系統、訓導行政子系統及圖書館管理子系統等子系統，並分別討論其系統需求。

當分析師對於系統或子系統的目的、特性及所需服務有了初步的瞭解之後，便可以開始制訂子系統所需要提供的功能及所使用的資料，並分別描述各系統及子系統功能的限制條件，功能規格的制定主要是依據功能處理劇本（scenario）來訂出，劇本包括對正常行為及不正常行為之描述。接下來便是要實際撰寫這些系統及子系統功能的規格，規格文件可用圖示、正規語言（formal specification language）或普通語言的方式來描述。當每個功能之規格皆定義完畢後，須進一步將這些子系統組合成我們所需要的系統，組合過程必須考慮不同子系統功能間的互動關係及資料之交換方式等。

由上所述，軟體系統需求分析的過程主要可以分成三個階段：

(1) 問題認知階段

認清客戶的問題及所屬之應用領域，收集與領域相關的知識，使分析師可以對客戶的問題及所需的系統有一初步的認知與了解。

(2) 需求蒐集階段

從客戶所提供之資料及與客戶的訪談間蒐集客戶對系統之服務需求。

(3) 需求分析及制定規格階段

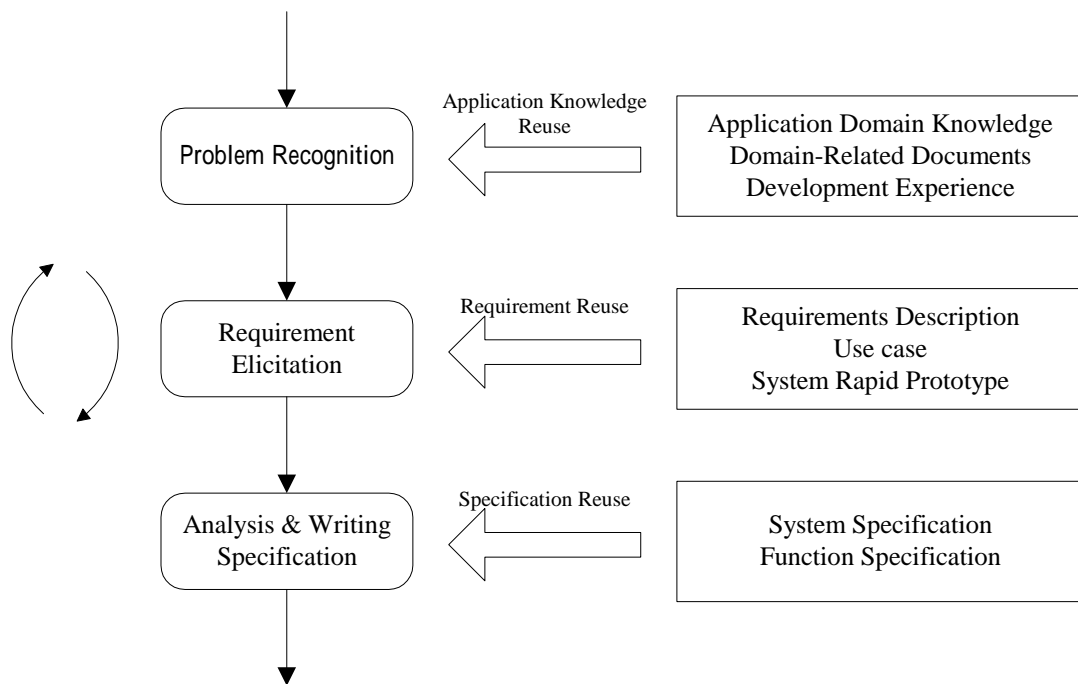
當客戶的需求被了解清楚後，則分析師依客戶的需求制定系統所需的服務，若系統太複雜則將系統分解成數個子系統訂定其提供的服務，最後再對各個功能分別制定其功能規格，並針對整合各個子系統以制定完整之系統規格。

軟體系統需求分析是一個反覆的過程，在分析初期有些系統需求可能未被發現，或在制定規格的過程中發現有些需求需要被修改，或是分析師制定之規格與客戶的認知有差距，此時上述的步驟會一直反覆進行直到客戶認可所制定完成之規格文件為止。

2.2.2 軟體需求分析階段再利用之時機與軟體產物

根據前一節所述，軟體需求分析的過程主要可以分成問題認知、需求蒐集及規格制定三個階段，欲加速需求分析的過程及需求規格文件之制定，分析師可再利用舊有的需求規格文件進行新系統之需求分析及撰寫規格，以下我們分別針對需求分析的三個階段探討可以運用再利用技術的時機及可再利用的軟體產物。

在問題認知階段分析師可藉由以往開發相同領域系統之經驗來幫助其了解新的問題，當進入需求蒐集階段時，分析師可參考以往開發的系統之需求制定過程及文件來幫助客戶制定新系統之需求，最後在規格制定階段若從前發展之系統與新系統有類似的系統需求，則以往制定之系統規格及功能規格也可被再利用，或被參考以制定新系統之規格。因此，我們可從軟體需求分析的三個階段找出以下三個可運用再利用技術的時機，如圖二-2 所示：



圖二-2 需求分析階段之再利用時機示意圖

(1) 應用領域知識再利用

當客戶欲開發的軟體系統是屬於分析師不熟悉的應用領域時，分析師此時可能不具備足夠的能力去了解客戶的問題及與客戶就系統需求進行溝通，因此分析師可以根據客戶對問題的描述去尋找與該問題相似之應用領域，閱讀跟該領域相關之文件資料，藉由這些資訊，分析師可對客戶的問題及所需的系統有一初步的認知與了解，尤其愈複雜之系統，所相關之應用領域愈多範圍愈廣、資訊收集愈費時，因此若能再利用相同或類似系統之領域知識內容，將有助於分析師對新系統之了解及進行分析。

此階段主要被再利用的軟體產物是應用領域知識，包括欲了解該領域需參考之文件資料或文件索引。

(2) 需求再利用

當進入需求分析階段時，分析師主要的工作即是與客戶將系統之需求談清楚，由於客戶對於軟體系統可能不是很熟悉，因此剛開始可能很難敘述清楚其想要系統提供的服務，此時若分析師有開發過類似領域系統之經驗，則分析師可以經由展示以往開發過的系統或是系統雛形讓客戶對類似的軟體系統有初步的了解，然後可和客戶從中尋找其所需之功能，或從以往開發系統之客戶訪談記錄及需求分析文件中尋找可再利用

之系統需求。

若分析師本身也對客戶欲開發的軟體系統不熟悉，亦可從相關的應用領域中尋找別人開發過類似的系統，經由了解其客戶訪談記錄及需求分析文件的過程，分析師也可以從中找出可能符合客戶所需之功能與客戶進行溝通。經由再利用前人所開發系統之需求分析的結果可大幅縮減當客戶對軟體系統不熟或分析師無相關軟體系統開發經驗時在進行需求蒐集時時間的花費。

此階段被再利用的軟體產物主要是需求分析相關的資訊，可能包含有系統的客戶訪談記錄、需求分析文件如系統、子系統的服務需求描述、系統之 use case 等。

(3) 規格再利用

當系統的服務需求已被談清楚時，則分析師開始定義系統的功能及制定功能及系統之規格，此時若是可以從以往開發過的系統規格中找到服務需求相近之系統，則由於服務需求相似，則系統規格必定也相似，此時直接再利用現成的系統規格，或是經由部份的修改後再利用會比重新制定新的系統規格來的節省時間及人力。

此階段被再利用的軟體產物即是系統的規格文件，包括系統規格、功能規格、類別規格文件等。

2.2.3 需求規格再利用流程

由於組織內部再利用之系統需求規格的來源是自己或是公司內部自行開發的元件或系統，對於未開發過之應用領域便無法進行再利用，因此在應用領域知識再利用及需求再利用階段之再利用效益並不明顯，本小節僅就組織內再利用進行規格再利用之程序進行探討。

根據前述制定系統規格之流程，若是系統複雜的話，可將系統分解成數個子系統，當各個子系統之功能需求確定之後，分析師應定義子系統之功能模型及資訊模型，並撰寫規格，此時分析師應可根據子系統之功能需求，尋找合適的系統規格或子系統規格進行再利用，由於可供再利用之候選規格元件可能不只一個，因此分析師必須先檢視規格元件內容，以判斷此規格元件是否真符合所需，並從中挑選出最接近新系統或子系統功能需求之規格元件進行再利用，若所有候選規格元件與實際所需差異太大，則分析師必須自行開發該部份之規格。當各子系統之功能規格定義完畢之後，分析師須依各子系統之間的交互關係，進一步定義各子系統規格以使各子系統可組合成完整的系統規格。此種規格再利用方式，即是組合式規格再利用。

組合式規格再利用可以子系統為單位，根據子系統之功能需求描述，尋找合適的系統或子系統規格予以再利用，以構建各子系統規格，由於子系統之服務需求是透過一組系統功能來達成，因此子系統規格再利用，可透過系統功能規格之再利用來達成。同理，複雜的系統功能規格，也可由一些子功能規格所組成，因此可運用子功能來定義複雜系統功能之再利用。此外，在組合子系統規格成完整的系統規格之階段，分析師主要是要定義各子系統之間的交互關係（例：資訊交換，執行順序限制），此時若有一現存系統所包含的子系統組成與新系統類似，則在組合的過程中，可參考此系統規格所記錄的交互關係描述，以進行子系統規格之整合。同理，在組合系統功能規格成為子系統，或組合子功能規格成為系統功能規格的過程當中，也可運用相同方式。

因此，組合式規格再利用主要是以子系統、功能或子功能規格為再利用對象，根據其功能需求描述，尋找合適的規格文件予以再利用，以下便歸納組合式規格再利用之步驟：

(1) 搜尋合適的規格元件

根據功能需求描述，到元件庫中搜尋有無合適的規格元件可資利用。所謂合適的規格元件，意指是與新規格有相同或相似的功能需求。

(2) 檢視候選規格元件內容，判斷是否合乎所需

由於可供再利用之候選規格元件可能不只一個，因此分析師必須先檢視規格元件內容，以判斷此規格元件是否真符合所需，並從中挑選出最接近新系統或子系統功能需求之規格元件進行再利用，若所有候選規格元件與實際所需差異太大，則分析師必須自行開發該部份之規格。

(3) 修改規格元件內容

規格元件經過檢視判斷之後，分析師須修改規格元件內容中不符合之處，才予以再利用。

(4) 組合各規格元件成為完整的系統規格

當各子功能、功能及子系統規格定義完畢之後，分析師須定義各功能、子功能及各子系統之間的交互關係，由下往上組成完整的系統規格。在規格元件組合的過程當中，分析師亦可尋找合適的規格元件，參考其組成單元間的交互關係定義，以加速規格元件之組合。例如，在組合各子系統規格元件為完整的系統規格之時，分析師可以尋找與新系統有相同或極為類似的子系統元件的系統規格予以再利用。

2.3 傳統需求規格再利用環境 vs. 廣域需求規格再利用環境

觀察目前與需求規格再利用相關之研究，可發現其再利用系統使用的環境多半是位於組織或公司內部，可供再利用的系統需求規格的來源也多半是自己或是公司內部自行開發的元件或系統，我們稱此種再利用方式為組織內再利用。此種再利用方式與我們所欲提出的廣域再利用 - 透過一個仲介系統在 Web 的環境下進行需求規格再利用，兩者存在許多差異，其主要差異有：

(1) 再利用的系統所屬的應用領域不限於以往開發過應用領域

由於以往的需求規格再利用的對象主要是自己或公司內部開發過的元件或系統，會再利用的系統所屬的領域必是自己已熟悉的應用領域，而當透過仲介系統在 Web 環境下進行需求規格再利用時，由於仲介系統中包含的應用領域範圍很廣，因此對於自己不熟悉的應用領域可能也可以找到可供再利用的系統需求規格，更可進行應用領域知識再利用及需求再利用，對於此種情形，舊有的需求規格再利用系統的再利用流程便需要做適當修改，才能符合實際情形。

(2) 再利用的系統需求規格可檢視的內容有限

由於以往的需求規格再利用的對象擁有者為自己或同一組織或公司，因此在進行需求規格再利用的過程中可以檢視系統全部的內容，而當透過仲介系統在 Web 環境下進行需求規格再利用時，對於未決定購買的系統並無法檢視其全部的內容，因此在比較及挑選欲再利用的系統需求規格時，其步驟也須視情形做適當的修正。

(3) 再利用的系統需求規格格式不固定

當透過仲介系統在 Web 環境下購買所需的系統需求規格元件時，由於製造的廠商可能不同，而需求規格文件的製作方式並無統一，所以其格式可能不同，因此對於再利用系統的設計也須考量此一因素。

為了要更進一步提升軟體再利用技術的效益，我們希望能在 Web 環境下進行需求規格再利用，即達到廣域需求規格再利用。由於廣域需求規格再利用與組織內需求規格再利用有著上述之差異，因此舊有適用於組織內再利用之需求規格再利用系統便無法被直接套用於 Web 環境。本計畫第一年研究之在 Web 環境下支援程式碼再利用的仲介者系統 (Broker) 已經獲得了良好的成果。因此，延續此在 Web 環境下支援程式碼再利用的仲介者系統以及 2.1 小節所介紹的架構在 FDOOA 需求分析方法上的物件導向需求規格再利用程序與環境技術，我們將提出一個在 Web 環境下支援需求規格再利用之仲介者系統。接下來我們會先介紹一般在 Web 上之 Broker based 的軟體再利用系統的基本架構及其再利用流程，

然後再探討在 Web 上之需求規格再利用 Broker 系統中進行需求規格再利用之流程並從中找出此支援需求規格再利用之 Broker 系統的服務需求，以做為接下來設計系統的依據。

三、 Web 上需求規格再利用系統之需求分析

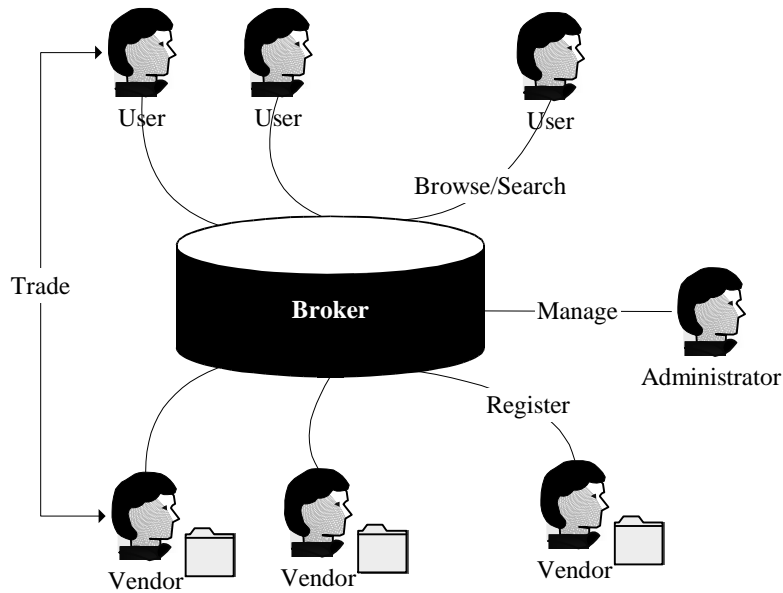
根據前一章討論之結果，欲在 Web 環境下進行軟體需求規格再利用，我們需要設計一個支援軟體需求規格再利用的仲介者系統（以下簡稱 Spec. Reuse System）。本計畫第一年之研究成果已成功建置一在 Web 環境下支援軟體程式碼元件再利用之 Broker 系統（以下簡稱 Code Reuse System），由於我們想要發展之 Spec. Reuse System 其基礎架構及基本的系統需求應會與 Code Reuse System 相類似，只是再利用的層次及對象不同，因此本節將先簡介一在 Web 環境下支援軟體再利用之仲介者系統的基本架構及其再利用流程。

3.1 Web 環境下軟體再利用系統簡介

在 Web 環境下之軟體再利用系統，其參與者共可分為三類：User（元件使用者）、Administrator（Broker 管理者）及 Vender（元件提供者），其間之相互關係如圖三-1 所示。當 Vendor 開發出新的軟體元件想透過網路銷售，便可將軟體元件登錄至專門蒐集可再利用軟體元件之 Broker，經過 Broker 之 Administrator 進行確認及分類的手續後，便完成了元件登錄之工作。而當 User 想利用現有之軟體元件開發新的軟體系統時，User 便可進入 Broker 中，並透過 Broker 提供之瀏覽及搜尋工具尋找所需之軟體元件，在找到符合所需的軟體元件後，User 可透過 Broker 與 Vendor 進行交易並下載軟體元件進行再利用。

一般 User 使用 Broker 之基本流程如下：

- (1) User 希望應用軟體再利用技術來加速軟體的開發，因此 User 進入 Broker 網站，經過 Broker 確認身分後，便可開始使用 Broker
- (2) User 可經由下列兩種方式在 Broker 中尋找可能符合所需的元件
 - (2.1) User 可以透過瀏覽元件分類架構的方式一一尋找可能符合所需的軟體元件
 - (2.2) User 利用 Broker 提供之元件搜尋工具，依其所需填入搜尋條件後，由搜尋工具幫 User 尋找符合條件之軟體元件，並依照相似程度排列
- (3) User 可找到一或多個候選軟體元件，User 先瀏覽每個候選元件之基本資訊，並選取部份有興趣之軟體元件檢視其詳細資訊



圖三-1 Web 環境下軟體再利用系統之參與者

- (4) User 在檢視元件詳細資訊後決定購買某個元件，透過 Broker 提供之交易工具，直接於線上進行交易
- (5) 完成交易程序後，User 可利用 Broker 提供之工具直接於線上下載購買的軟體元件

軟體元件 Vendor 使用 Broker 之流程如下：

- (1) 軟體元件 Vendor 新開發了一個軟體元件想要透過 Broker 銷售，於是 Vendor 進入 Broker 網站，經過 Broker 確認身分後，便可開始使用 Broker
- (2) 軟體元件 Vendor 透過 Broker 所提供之元件資訊登錄工具，將軟體元件相關的資訊上傳並登錄至 Broker
- (3) 在完成登錄後會由 Administrator 進行進一步之處理，Vendor 在收到通知後，元件便可開始於 Broker 上銷售

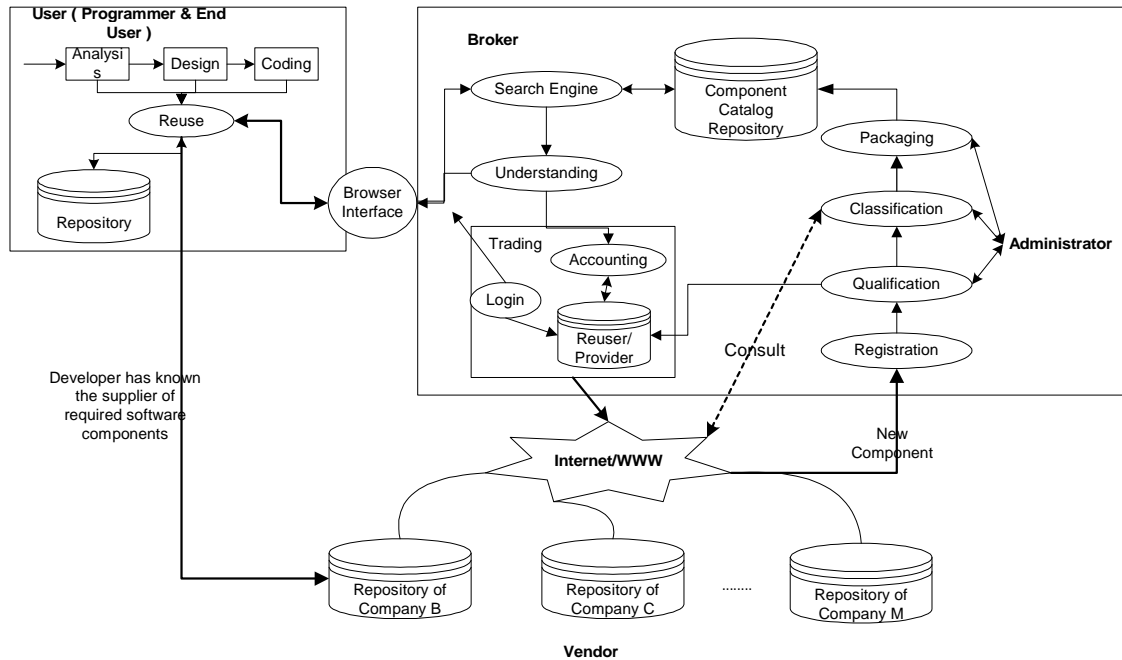
Administrator 需要負責管理 Broker 之工作，其工作包括有管理使用者資訊、確認並歸類 Vendor 登錄之軟體元件，管理並維護交易資訊等。

我們可分別從這三種系統參與者的角度，找出在 Web 環境下支援軟體再利用的 Broker 之系統需求，歸納如下：

- } 從 Vendor 的角度
 - € Broker 必須要能吸引眾多 User 上網
 - € Broker 必須提供方便好用之元件註冊工具

- € Broker 必須提供健全的交易制度
- } 從 User 的角度
 - € Broker 必須包含種類豐富的軟體元件 Broker 必須提供快速準確之元件搜尋工具
 - € Broker 中的元件必須包含足夠的元件資訊
 - € Broker 必須提供方便好用之元件檢視工具
 - € Broker 中的元件必須具有一定的品質
 - € Broker 必須確保交易過程安全
 - € Broker 必須提供 User 在 User 端管理元件的機制與工具
 - € Broker 必須追蹤 User 之瀏覽及購買行為，並提供個人化之服務
- } 從 Administrator 的角度
 - € Broker 必須提供使用者身分確認功能
 - € Broker 必須制定統一的元件註冊格式
 - € Broker 必須要求 Vendor 提供足夠之元件資訊

根據上述之系統需求，我們可以再進一步歸納出一個在 Web 環境下支援軟體再利用的 Broker 主要可依其所需提供的服務分成 Registration、Qualification、Classification、Packaging、Component Catalog Repository、Search Engine、Understanding、Trading 等子系統，其系統架構如圖三-2 所示。



圖三-2 Broker 系統架構

以下便分別就每個子系統所負責之工作做一簡單介紹：

(1) Registration

Registration 子系統依據元件封裝所需資訊中 Vendor 必須提供的部份，發出註冊表格要求 Vendor 填寫欲存入元件之相關資訊，並檢查是否符合填寫格式以及有無遺漏，若有錯誤，則應要求廠商重新更正，而後驅動 Qualification 子系統。

(2) Qualification

Qualification 子系統會針對欲註冊的軟體元件，提供給 Administrator 檢視其資料之正確性、測試其是否能夠正確運作及是否滿足 Vendor 所宣稱的功能，並量度軟體元件之各項品質，排除不良的軟體元件。元件通過品保手續則驅動 Classification 子系統。

(3) Classification

Classification 子系統負責將通過 Qualification 之元件提供給 Administrator 進行歸類，Administrator 按元件所屬之應用領域及元件特性抽取搜尋所需之分類特徵。Classification 必須仰賴 Administrator 之分類經驗，倘若 Administrator 遇到不熟悉之應用領域元件，亦可請提供此元件之 Vendor 提供分類的建議。

(4) Packaging

Packaging 子系統將 Classification 子系統所抽取之元件分類特徵、各項元件資訊及品質測試結果封裝成一元件儲存單位而存入 Component Catalog Repository。

(5) Component Catalog Repository

Component Catalog Repository 即元件目錄庫負責儲存封裝後的元件資訊。並提供目錄中元件項目的刪除、修改以及目錄結構調整之運作。

(6) Search Engine

Search Engine 提供 User 輸入所欲搜尋之元件之領域及特徵條件，而後依 User 輸入之搜尋條件至 Component Catalog Repository 找出合適的候選元件並依序列出。Search Engine 必須提供一套簡單易用的查詢語言以及相似性計算公式。查詢語言應讓使用者很容易的定義所欲搜尋之元件，同時又能確實地表示出該元件的特性。而相似性計算公式可幫助使用者在眾多的候選元件中，迅速排除相似程度值低的元件，節省驗證元件是否符合所需之檢視時間。

(7) Understanding

經由 Broker 之 Search Engine 取出的候選元件數量可能很多，User 經由 Understanding 子系統檢視元件之封裝資訊以確認元件是否符合所需，並從中比較挑選出最適合之元件。

(8) Trading

Trading 子系統用以支援元件之交易、追蹤 User 上線查詢記錄、紀錄元件成交資料及撥付給 Vendor 之相關費用情況。Login 是提供 User 以及 Vendor 之私密性檢查，User/Vendor 資料庫則記錄 User 與 Vendor 的基本資料以及交易記錄等。Trading 機制可確保 Vendor 收到提供元件之經濟效益。

3.2 Web 環境下之需求規格再利用系統

上一節簡介了一在 Web 上支援軟體再利用的 Broker 系統的系統架構，及使用者使用此 Broker 系統之流程。然而，不同的軟體再利用層次對於其 Broker 系統之設計及使用者之使用流程也會有所不同，本小節便先對使用者使用支援需求規格再利用之 Broker 系統進行軟體需求分析的流程做一探討。

3.2.1 Web 上進行需求規格再利用之流程

當分析師接獲客戶的委託欲開發一軟體系統時，首先必須對客戶欲解決的問題進行了解，才有辦法與客戶進行進一步的溝通，此時分析師可能會遇到下列兩種情形：

} 分析師對於該軟體系統所屬領域不熟悉

由於在廣域再利用的情況下，再利用的系統所屬的應用領域不限於以往開發過應用領域，因此第一種情形可能發生在分析師以往沒有開發過與客戶的問題相關的應用領域的軟體系統，對於該領域的專業知識並不熟悉，若直接與客戶進行訪談，可能不易了解客戶真正的問題與需求。因此分析師必須先收集並學習與系統所屬領域相關的知識，然後才有能力了解客戶的問題，並與客戶再針對系統的需求進行進一步的溝通。

此時，分析師可透過下列步驟透過 Broker 進行應用領域知識再利用：

- (1) 分析師進入 Broker 網站，透過瀏覽的方式尋找與客戶的問題相近的應用領域
- (2) 檢視各個應用領域的基本描述，判斷是否可能是相近的應用領域，或是可參考應用領域下包含之軟體系統之系統概述，了解該應用領域下所包含之系統類型，以增加對應用領域之判斷能力
- (3) 選取一與客戶問題最接近之應用領域，可以從中找到欲了解該應用領域需要參考之書籍、文獻資料索引
- (4) 在學習相關之領域知識後，分析師便有與客戶溝通系統需求的能力

} 分析師對於該軟體系統所屬領域很熟

第二種情形可能是分析師對於開發與客戶問題類似的系統已經具有某些經驗，對於該領域已很熟悉，因此已具備有了解客戶問題、與客戶進行蒐集系統需求的工作的能力。

在分析師對問題所屬之領域已有充分了解後，接下來便需要與客戶就系統的需求進行溝通與討論，由於不同的客戶對電腦系統之了解程度不同，因此分析師與客戶進行訪談之流程也會有所不同，針對客戶對電腦系統的了解程度，以下分成下列兩種情形進行討論：

} 客戶對電腦系統不熟

第一種情形是客戶本身對電腦系統不熟悉，可能客戶想到要將某項工作電腦化但以前未有操作過相關的軟體系統的經驗，因此客戶在描述

問題及其需求時應是從其原本做事之方式及流程思考。此時分析師之工作便是將客戶對問題的描述及對做事流程之描述轉化為其與軟體系統之間的互動，並依此與客戶溝通各項需求。

在這種情形下，若分析師具有開發過此領域類似系統之經驗，則分析師與客戶進行溝通的過程會較順利，因為對於客戶對需求的描述，分析師可利用以往開發過的系統經驗從中尋找對應的系統服務與客戶進行更進一步的討論。此外，從以往開發過的系統的客戶訪談記錄或需求分析文件中分析師也能夠找出客戶可能也會需要但尚未發現之系統功能，或是可以向客戶展示其雛形系統以找出更多客戶的需求。若是分析師本身並無開發過此領域類似系統之經驗，則分析師可透過下列步驟利用 Broker 系統進行需求再利用：

- (1) 分析師到 Broker 中與客戶問題相近之應用領域下，瀏覽應用領域下之系統可能提供之系統功能
- (2) 檢視各項系統功能之描述，選取客戶可能需要之系統功能
- (3) 與客戶就所選取之功能逐一討論，並記錄客戶所需之功能
- (4) 利用 Broker 提供之工具，尋找是否已經存在有與客戶的功能需求相近的系統
- (5) 由 Broker 挑選出一到數個基本需求接近之系統，分析師檢視其基本資訊以決定是否購買更詳細之系統需求文件

} 客戶對電腦系統很熟

此種情形可能發生在客戶已有使用相關軟體系統之經驗，欲開發的是類似的軟體系統，或是已有現成類似之軟體系統存在，但並非完全符合客戶的需求。此時分析師在與客戶談系統需求時可能會先從現有的系統，或以往操作系統之經驗著手，先從客戶舊有的經驗或現有的系統中找出客戶需求之功能，再討論需要被修改或新增之功能需求。同樣的，若是分析師本身也具有開發過此領域類似系統之經驗，則可再利用以往開發過的系統之相關需求分析文件來加速與客戶談論系統需求之過程。若是分析師本身並無開發過此領域類似系統之經驗，則分析師可透過下列步驟利用 Broker 系統進行需求再利用：

- (1) 分析師到 Broker 中與客戶問題相近之應用領域下，瀏覽應用領域下之系統可能提供之系統功能，並與客戶提出對系統的需求進行比對
- (2) 利用 Broker 提供之工具，記錄與客戶需求相近之系統功能，並尋找是否已經存在有與客戶的功能需求相近的系統

- (3) 由 Broker 挑選出一到數個基本需求接近之系統，檢視候選系統所提供之系統功能，並對客戶之前未想到之功能進行討論
- (4) 分析師檢視與客戶需求較接近之候選系統的基本資訊以決定是否購買更詳細之系統需求文件

當系統需求談清楚以後，分析師則會依據客戶的服務需求一一制定系統所需提供的服務，此時，若系統的範圍太大或太複雜，以致不易明確訂出系統所有的服務，則分析師可將系統分解成數個較小之子系統進行分析，針對每個子系統分別定出其系統的目的及所提供之功能，然後再針對各個功能制定規格及針對各系統及子系統分別制定其系統規格。此外，分析師可到 Broker 中尋找是否已存在可以滿足客戶需求的系統，若存在則分析師便不須重新制定新系統之規格文件，可考慮直接購買現成之系統規格文件進行再利用，其再利用之步驟如下：

- (1) 從之前的應用領域知識再利用及需求再利用階段，分析師可在 Broker 中找到與客戶問題相近之應用領域
- (2) 瀏覽應用領域下，系統可能提供之功能，利用 Broker 提供之工具記錄符合客戶需求之系統功能
- (3) 搜尋 Broker 中是否有可滿足客戶需求之現成的系統，由 Broker 挑選出候選系統
- (4) 分析師檢視候選系統之基本資訊，及系統功能描述，若是現有系統與客戶需求之系統很接近，則可購買現成之系統規格而不需重新制定新的系統規格文件

以上便是利用支援需求規格再利用之 Broker 系統進行需求分析之流程，由於軟體系統需求分析是一個反覆的過程，因此上述的步驟會一直反覆進行直到客戶認可所制定完成之規格文件為止。

3.2.2 Web 上需求規格再利用系統之關鍵技術

從上一小節探討之在 Web 上進行需求規格再利用的流程，我們可以發現欲設計一 Web 上支援需求規格再利用之 Broker 系統，必須先針對下列關鍵問題進行更深入的研究，其中包括：

- } 軟體需求規格再利用單元之分析

我們必須先釐清在 Broker 系統中，Vendor 登錄及 User 欲再利用之需求規格再利用單元其組成為何，需要包含的資訊有哪些。

- } 軟體需求規格再利用單元之分類架構設計

然後我們可以根據 User 使用 Broker 之流程及需求規格再利用單元所包含的內容設計 Broker 系統中需求規格再利用單元之分類方式。

} 軟體需求規格再利用單元之搜尋、檢視及擷取機制

再依據再利用單元之分類架構、再利用單元包含的內容及 User 進行需求分析之流程與習慣設計 Broker 之搜尋工具，其中包括搜尋之機制及相似度計算的公式。然後再依搜尋可得知結果及元件所包含的內容設計元件檢視工具。

} Broker 系統儲存庫之設計

最後我們必須再研究設計一可儲存需求規格元件之元件儲存庫。

四、 結論及未來工作項目

近年來，軟體再利用技術已被認為是有效提升軟體產能與品質的方法。軟體再利用意指在軟體發展過程中，軟體發展者利用現存的軟體產物，加以組合發展新的軟體系統，發展者透過搜尋引擎至元件庫中尋找合用的軟體元件並了解其功能行為，進而篩選出最合適的軟體元件予以決定直接使用或修改利用。本計畫將軟體再利用的觀念與現行的 Internet/WWW 結合，創造出一個廣域再利用的環境，讓使用者透過 Broker 的幫助，再利用 Internet 上眾多 Vendor 所提供的再利用資源。本計畫第一年已完成一“以仲介者為基礎之 Internet/WWW 環境下的軟體再利用 (Broker-based Software Reuse on Internet/WWW) 之環境架構”。

延續第一年之研究成果，我們繼續對如何在 Internet/WWW 此一廣域環境下進行需求規格再利用進行研究，我們先前已就目前與需求規格再利用相關之研究做過探討，並經由研究本實驗室先前設計之組織內部的需求規格再利用系統，找出了在廣域的需求規格再利用環境下與在傳統的組織內部進行需求規格再利用有何不同之處，並歸納出在 Web 環境下進行需求規格再利用的流程，以及設計一支援需求規格再利用 Broker 系統所需研究的關鍵技術。接下來，在本計畫的剩餘時間內，我們將對之前提到設計一支援需求規格再利用 Broker 系統所應研究之關鍵技術做更深入的探討並提出解決方案，然後我們將以第一年完成的 Broker 雛形系統為基礎，增加或調整相關的關鍵技術以及工具，擴充此雛形系統以支援廣域的需求規格再利用。

參考文獻

- [1] Charles W. Krueger, "Software Reuse," *ACM Computing Surveys*, Vol. 24, No. 2, pp. 131-183, June 1992.
- [2] Even-André Karlsson, *Software Reuse: A Holistic Approach*. NY: John Wiley & Sons, 1995.
- [3] Ivar Jacobson, Martin Griss, and Patrik Jonsson, *Software Reuse: Architecture Process and Organization for Business Success*, New York: ACM Press; Harlow, England: ACM Wesley Longman, 1997.
- [4] M. L. Griss, "Software Reuse: From Library to Factory," *IBM Systems Journal*, Vol. 32, No. 4, pp. 548-566, 1993.
- [5] Charles W. Krueger, "Software Reuse," *ACM Computing Surveys*, Vol. 24, No. 2, pp. 131-183, June 1992.
- [6] O. Nierstrasz, S. Gibbs, and D. Tsichritzis, "Component-Oriented Software Development", *Communications of the ACM*, vol. 35, No. 9, pp. 160-165, Sept. 1992.
- [7] W. Frakes and S. Isoda, "Success Factors of Systematic Reuse," *IEEE Software*, Sept. 1994, pp. 15-19.
- [8] Sung-Koo Lee and Joseph E. Urban, "SOORLS: A Software Reuse Approach on the Web," *International Journal of Software Engineering and Knowledge Engineering*, Vol. 9, No. 3, pp. 279-296, 1999.
- [9] Roger S. Pressman, "Software Engineering: A Practitioner's Approach, Forth Edition", 1997.
- [10] Jacob L. Cybulski, Ralph D. (Butch) Neal, Anthony Kram and Jeffrey C. Allen, "Reuse of early life-cycle artifacts: workproducts, methods and tools," *Annals of Software Engineering 5*, pp. 227-251, 1998.
- [11] Jacob L. Cybulski, Karl Reed, "Requirements Classification and Reuse: Crossing Domain Boundaries", *Sixth International Conference on Software Reuse*, Vienna, Austria, June 27-29, 2000.
- [12] Girardi, M.R. and B. Ibrahim, "A software reuse system based on natural language specifications," *5th Int. Conf. On Computing and Information*, Sudbury, Ontario, Canada, p. 507-511, 1993.
- [13] N. A. Maiden and A. G. Sutcliffe, "Exploiting Reusable Specifications Through Analogy", *Communications of the ACM*, 35, 55-64, 1992.

- [14] Silvana Castano and Valeria De Antonellis, "The F³ Reuse Environment for Requirements Engineering," *ACM SIGSOFT Software Engineering Notes*, Vol. 19, No. 3, July 1994, pp. 61-65.
- [15] S. C. Chou and C. G. Chung, "The Study of Specification Reuse", Technical Report of National Science Council, ROC, No. NSC81-0408-E009-542, 1993, 1994.
- [16] 劉耀華, "軟體規格儲存庫之設計", 國立交通大學資訊工程研究所, 碩士論文, 1994.
- [17] 范姜永益, "需求規格再利用系統之設計", 國立交通大學資訊工程研究所, 碩士論文, 1996.
- [18] 吳介銘, "改良式FDOOA分析方法", 國立交通大學資訊工程研究所, 碩士論文, 1996.