行政院國家科學委員會補助專題研究計畫 ☑ 成 果 報 告 □期中進度報告

輕度障礙學生數學教學之知識管理導向數位學習平台研發---子計畫三：輕度障礙學生數學教學之數位學習平台核心模組與技術研發（3/3）

計畫類別：□ 個別型計畫　☑ 整合型計畫

計畫編號： NSC 96-2524-S-343-001-

執行期間：96 年 8 月 1 日至 97 年 7 月 31 日

計畫主持人：王昌斌　南華大學電子商務管理系

共同主持人：陳宗義　南華大學電子商務管理系

陳裕民　國立成功大學製造工程研究所

計畫參與人員：楊惠媚　大同商業專科學校

陳萌智　國立成功大學製造工程研究所

曾清義　南華大學資訊管理研究所

丁月琴　南華大學資訊管理研究所

林育弘　南華大學資訊管理研究所

蔡政宇　南華大學資訊管理研究所

陳育銘　南華大學資訊管理研究所

藍釭凱　南華大學資訊管理研究所

張堯榮　南華大學資訊管理研究所

執行單位：南華大學電子商務管理系

中　華　民　國　97　年　10　月　21　日

一、前言

數位學習的特色，在於受教者能打破時空限制，隨時隨地進行學習。經由數位學習，能使學習活動更有效率；加上教學內容可輕易大量複製，因能降低學習成本，達到「時時可學習（any time）」及「處處可學習（any where）」的學習目標。然而，目前尚無針對輕度障礙學生之教師、家長及相關領域專家之需求所發展之數學學科數位學習平台，能有效提供診斷與教學知識及教學策略與教材內容。是故，建構一知識管理導向之「輕度障礙學生數學教學之數位學習平台」實有必要性。

本子計畫主要任務是以輕度障礙學生數學教學之知識及知識管理之實現技術為基礎配合子計畫（一）、子計畫（二），來開發平台的核心功能模組與技術，並針對各模組之核心技術進行方法設計與元件開發，使平台具備：(1)提供教師適性化之教學策略、方法及教材之能力；(2)知識自動獲取、解析、儲存及自我學習之能力；(3)教學個案知識分析、學習樣板(Pattern)建立及預測之能力；(4)提供輕度障礙學生線上個人化自我學習之能力。

二、計劃緣由與目的

2.1 研究背景

教育是國家百年大計；在這個知識爆炸的時代裡，教學成效的良窳，不僅影響諸多學子的一生，更攸關國家未來的發展前途。我們認為，教育的目的，不僅在培育揚名世界的精英學生，也應把焦點放在弱勢學生──學習障礙學生的身上。

學習障礙可分為閱讀疾患(reading disorder)、數學疾患(mathematics disorder)與文字表達疾患(disorder of written expression)等三類(陳以青，2004);而其中又以數學障礙最為普遍。國內外正式或非正式研究報告均指出，數學是國民中、小學學生最感學習困難的學科之一(邱上真、詹世宜、王惠川與吳建志，1995)。另研究報告也指出，國民中、小學學生約有6%具有嚴重的數學障礙(Fleischner , Marzola, 1988)。

數學為科學之母，是一切科學的基礎；欠缺數學能力的學生，未來勢與高新科技產業

無緣。在可見的未來，勞力密集產業將從國內絕跡，欠缺數學能力者，極可能成為失業者的同義詞。因此，對數學學習障礙學生的誘導及訓練，使其能適應未來環境其具備獨立謀生之能力，不僅是當前特殊教育的努力目標之一，也是教育工作者的神聖使命與義務。

數學學習障礙難道真的無可挽救嗎？答案是否定的。研究指出：幾乎所有二年級數學學習障礙學生均有解題之潛能，惟教師必須發展適性化之教學策略，以增進學生成功的數學經驗（朱經明，2001）。但研究卻也顯示，大多教師面對數學學習障礙學生的問題時，常反覆使用先前未成功的教學策略，而未能針對個別的學習問題，使用不同教學方式(Fuchs et. al., 1991) 。

如前述，在資訊不發達的昔日，教師須以一己之力，在有限的時間內，對不同學生、不同主題發展適性化的教學策略以挽救學習障礙學生，顯然是力不從心。拜資訊科技之賜，今日，不同教師間經由網路交換教學心得，已不再是夢想；倘能建構一數位學習(e-Learning)平台，讓所有面臨學習障礙學生的教師、家長，以及相關領域專家能在線上交換心得，進而能給予學習障礙學生適性的關懷與輔導，同時也提供教師即時的進修管道，以強化教師之學習障礙學生數學教學的專業知識與培養其適性化教學能力，不就有機會挽救這些學習障礙的學生了嗎？

是故，建構一知識管理導向之「輕度障礙學生數學教學之數位學習平台」，提供教師即時的進修管道，以強化教師之學習障礙學生數學教學的專業知識與培養其適性化教學能力，實有其價值與必要性。

數位學習的特色，在於受教者能打破時空限制，隨時隨地進行學習。經由數位學習，能使學習活動更有效率；加上教學內容可輕易大量複製，因能降低學習成本，達到「時時可學習（any time）」及「處處可學習（any where）」的學習目標。然而，目前的數位學習平台多以一般學生為教學對象，無法為前述學習障礙學生之教師、家長及相關領域專家，針對其面臨的個案，有效提供相關之診斷與教學知識及教學策略與教材內容。目前雖有相關研究，將學習者在教學平台上的學習行為，透過網站的紀錄檔(Log Files)資料加以探勘

(Mining)，期能辨識學習者之學習行為樣式(Pattern)，以利後續教學策略的調整(張智凱,2002， 蔡昌均,2001)；然其研究對象是數位學習平台的直接使用者；本研究的對象──輕度障礙學生卻非平台的使用者。如何經由他人敘述，獲得、分析、存取與管理大量且異質性之學生身心特質知識、教學知識、教材與教學案例；仍待進一步深入探討。

為此，本研究將研發「輕度障礙學生數學教學之數位學習平台相關實現技術」，核心模組與核心技術研發。

2.3 研究目的

本研究之總目標在建構一知識管理導向之輕度障礙學生數學教學之數位學習平台，本(子)計畫的主要目的在開發平台功能模組與其核心技術，並針對各模組之核心技術進行方法設計與元件開發，使核心技術能具共用性、彈性及可再用性，使本平台具備：(1)提供教師「輕度障礙學生數學教學」之適性化教學策略、方法與知識及教材之功能；(2)知識自動獲取、解析、儲存及自我學習甚至之功能；(3)教學個案知識分析、學習樣板(Pattern)建立及預測之功能；(4)提供輕度障礙學生線上個人化自我數學學習功能。

為達成上述研究之目的，本研究將進行下列主要研究項目：

(1) 設計功能模組細部架構:包含問題詢答 (User inquiry) 、知識分享 (Knowledge Sharing) 、個案診斷教學、知識擷取、專家庫與社群、學生線上個人化學習、學習評量、系統管理與維護。

(2) 發展一個具可適性化之教材內容標準及教材整合技術

(3) 研發下列核心技術：

- 知識存取控制(Access control) 技術
- 網路互動式問題與需求語意分析技術
- 知識擷取(Knowledge Mining) 技術

  (a)案例知識探勘與行為分析技術

  (b)個案學生學習歷程分析與個案知識探勘

(c)專家知識擷取(Expert Knowledge Capturing)

(d)網路知識探勘(Web Knowledge Mining)

- 學生線上個人化學習機制 (與子計劃一持續密切進行中)
- 適性化教材之建構及產生技術 (與子計劃一持續密切進行中)

本(子)計畫共依序執行三年

第一年之產出如下：

（1）平台之主要功能與功能模組之細部架構

（2）語意分析與知識轉換技術

（3）個案學習歷程庫模型及各類知識庫模型

（4）一個具可適性化之教材內容標準及教材整合技術

（5）適性化教材撥放機制

第二年產出如下：

第二年度之主要工作是核心技術開發，根據總計畫及各子計畫第一年之產出，本計劃在這個階段以分組同步的方式來進行主要之核心技術的開發，每一項核心技術的開發分成「核心技術設計」及「核心技術建置」兩個階段。核心技術建置主要工作依序是：(1)靜態模型設計、(2)動態模型設計、(3)資料模型設計、(4)知識模型設計、(5)案例庫之規劃與設計及(6)模組實際開發及資料庫建置。各核心模組的設計階段則依據不同的核心技術而有不同的設計程序，分別說明如下：

(1)知識存取控制技術開發：主要的技術開發的工作，包括「使用者及維護者之權限需求分析」、「平台知識分享之管理原則設計」、「知識分類的方法與技術發展」、「分散式角色為基之存取控制模型設計」、「知識分享之風險評估模式設計」與「知識存取控制模組織實際設計與建置」等六主要活動。

(2)個案學習歷程分析技術開發：從子計畫一「教學案例收集、模式設計與知識分析」的結果，進行「個案結構化分析」企圖找出一個適合進行探勘之個案資料結構，「選擇探勘的模式及預估之結果」進行「探勘技術的設計及測試」及「探勘演算法的設計」，之後即進入「核心技術建置」的階段，將上述之研究結果建置成系統元件。

(3)輕度障礙學生線上學習機制開發：參考子計畫一所收集之教學案例，首先進行「學生線上學習之程序設計」，接著進行「學生線上學學習之功能需求分析」，進行「學生線上學學習之核心元件需求分析」及「與其他核心機制之互動分析」，進行「界面設計」及進行「整合適性化教材播放機制設計」的工作，之後進入學生線上學習機制之「核心技術建置」階段。

(4) 問題與需求語意分析技術開發：根據本計畫第一年之「輸入介面與CTI整合研究」、「語音文字轉換機制設計」及「語意分析及知識轉技術研究」之成果進行「問題與需求語意分析技術開發」之「核心技術建置」。

(5) 專家知識擷取技術開發：進行以專家為中心之「角色互動分析」之後針對各種角色互動模式進行「互動內容分析」，根據「互動內容分析」之結果轉化成一專家知識之結構化資料結構模式，進行「專家知識探勘技術及演算法設計與測試」，最後進行專家知識探勘技術的「核心技術建置」階段。

(6) 分享知識與網路知識探勘技術開發：首先進行「網路知識種類分析」，之後進行「網路知識內容分析」及「網路知識內容價值分析」，根據上面兩個活動的產出進行「網路知識內容擷取界定」及「網路知識內容擷取技術設計」，之後進行各類「網路知識內容結構化設計」，針對不同的網路知識內容進行「網路知識探勘技術及演算法設計與測試」，最後進入網路知識探勘技術的「核心技術建置」階段。

第三年度之進度：

(1) 核心功能模組的驗證及測試：進行「核心功能模組測試」，之後進行平台「跨功能模組的測試」，執行「功能模組的改善」。

(2) 平台導入：利用子計畫(一)所收集建構之教學案例與模式與子計畫(二)所發展之知識管理引擎，實際建構一個完整之平台，提供教育工作者或輕度障礙生之家長進行系統驗證及實驗，首先進行「案例、資料及知識庫的建立」及「平台設定」，找來系統測試者包含專家、教師、家長及輕度障礙學生實際操作本系統，之後進行「初步的系統測試」並「評估平台之效能及進行改善計畫」。之後交由子計畫(一)之工作者進行「實驗設計」將被測試者分成實驗組及控制組，之後分析本平台在輕度障礙數學教學上所產生之效益。

(3) 追蹤及修正：針對實際驗證結果的缺失進行改善，以期達到預計的目標。

三、目前已完成研究成果

3.1 輕度障礙學生數學教學之數位學習平台架構與功能模組之開發

本數位學習平台乃針對輕度障礙學生之數學教學目標與欠缺之處，分析教育專家、教育工作者、家長及學生之需求及各成員與平台之間的互動(詳見總計畫)，設計一個能提供教育工作者適性化教學策略、方法及教材之平台，提供輕度數學障礙學生線上診斷與學習之環境，藉以提升教學效益並解決現階段學習過程中輕度障礙學生所遭遇之學習障礙。，茲分別說明「數位學習平台整體架構之設計」、「數位學習平台整體架構之主要功能模組」、「數位學習平台整體架構之使用者介面設計」如下：

(1) 數位學習平台整體架構之設計

本數位學習平台設計考量之依據：輕度障礙學生之特性、教育輕度障礙學生之教學支援需求、及目前之網路、語音與多媒體線上教學技術之應用，配合子計畫(一)「輕度障礙學生數學教學之數位知識內容建構」及子計畫(二)「輕度障礙學生數學教學之數位學習平台知識管理實現技術研發」之研究成果，結合核心模組與技術研發等相關工作而架構之輕度障礙學生數學教學之數位學習平台。

- 數位學習平台之系統配置

本系統配置將區分為使用者端、網路應用伺服器、應用程式伺服器、資料庫及知識庫、郵件伺服器。網路伺服器(詳見圖 1)：主要提供本系統主要網頁介面給使用者進行連結及使用相關系統功能；應用程式伺服器：主要提供本系統相關系統功能程式的儲存，將本系統所有的應用程式儲存於此，當使用者進行使用本系統時，系統將會連結至此伺服器進行應用程式的運算，再將最終結果回覆至網路頁面上；資料庫及知識庫：主要提供本系統相關資料及知識的儲存；郵件伺服器：主要提供本系統進行郵件的傳送。

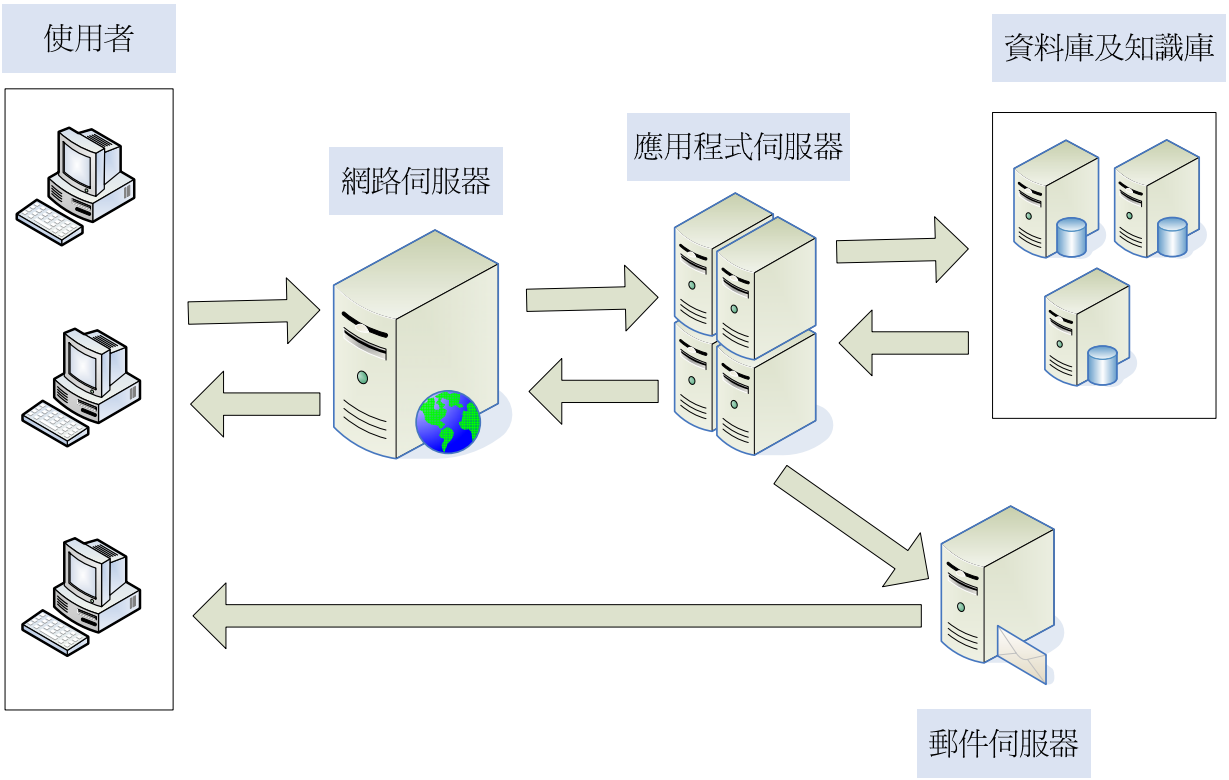

圖 1　數位學習平台之系統配置圖

- 數位學習平台之系統架構

本系統規劃採用多層次(Multi-Tier)&分散式物件(Distributed Objects) 資訊處理架構並以循環式動態性整合的系統發展模型建構（如圖 2）。運用類似網路服務(Web service)的概念將本系統各功能拆解成各個獨立的服務提供者，當應用程式或使用者進行系統使應用時，各應用程式便可提供與需求相對應的應用服務，最後再將結果傳回給要求服務的應用程式或使用者。此模式能方便系統的同步開發及系統效率的調整，當開發出效率更好的應

用程式時便能將原有的應用程式直接抽換成新的應用程式，省去更動程式內容時所要負擔的風險。而在系統開發時也可將各逐步完成的功能附掛置系統上，加快系統開發人員在進行系統開發時的開發效率。

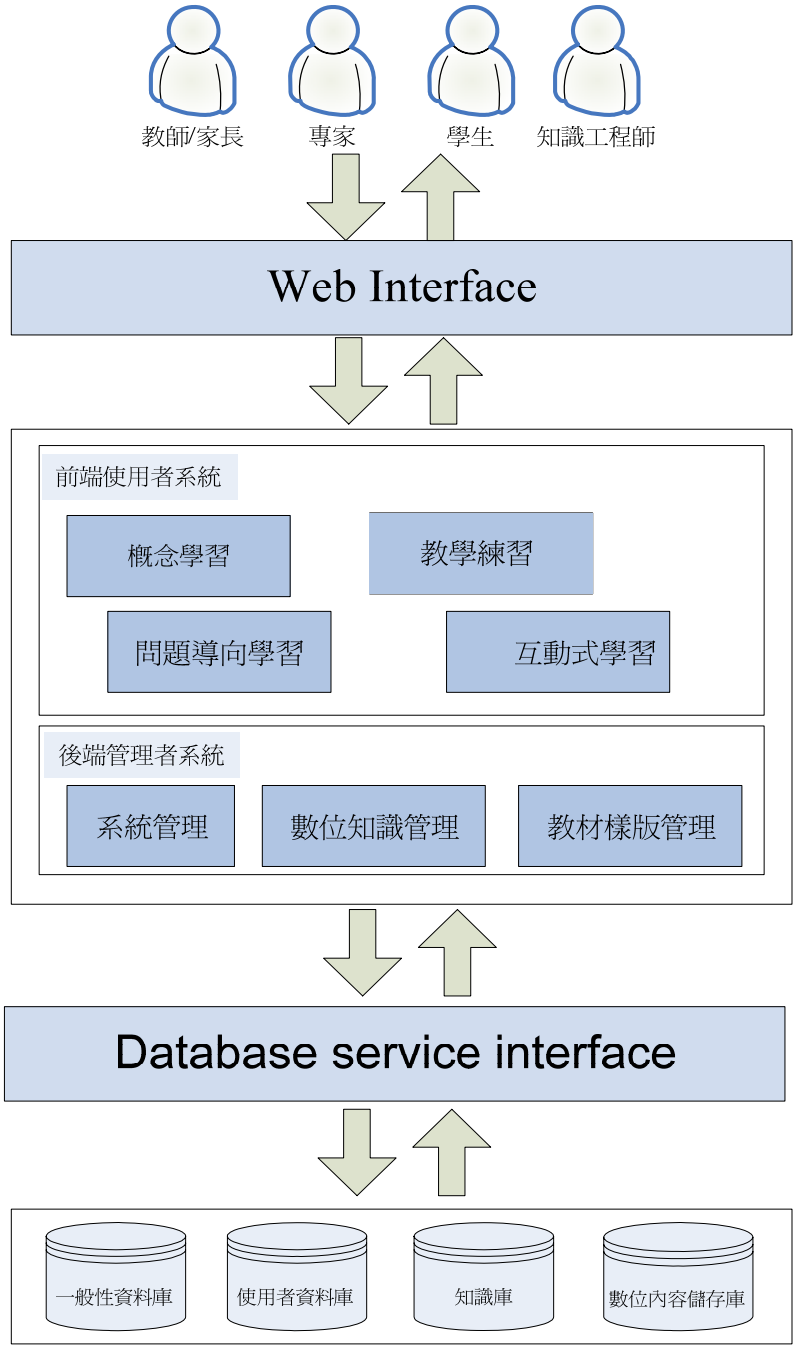

圖 2　數位學習平台之系統架構圖

(2) 數位學習平台功能架構之設計

本平台之主要功能模組如圖 3 所示，包括:功能介面、核心模組、知識管理引擎、數位知識內容儲存區四層；功能介面包括: 個案診斷教學、知識分享、問題詢答、受輔學生線上學習、專家庫與社群及系統管理；知識管理實現元件包括: 知識儲存、知識撿索、個案診斷教學推理、系統自我學習與內容維護；數位內容儲存區將儲存之數位內容包括: 數位知識、教學案例及個案診斷與學習歷程資料，平台之細部功能架構，如圖 4。

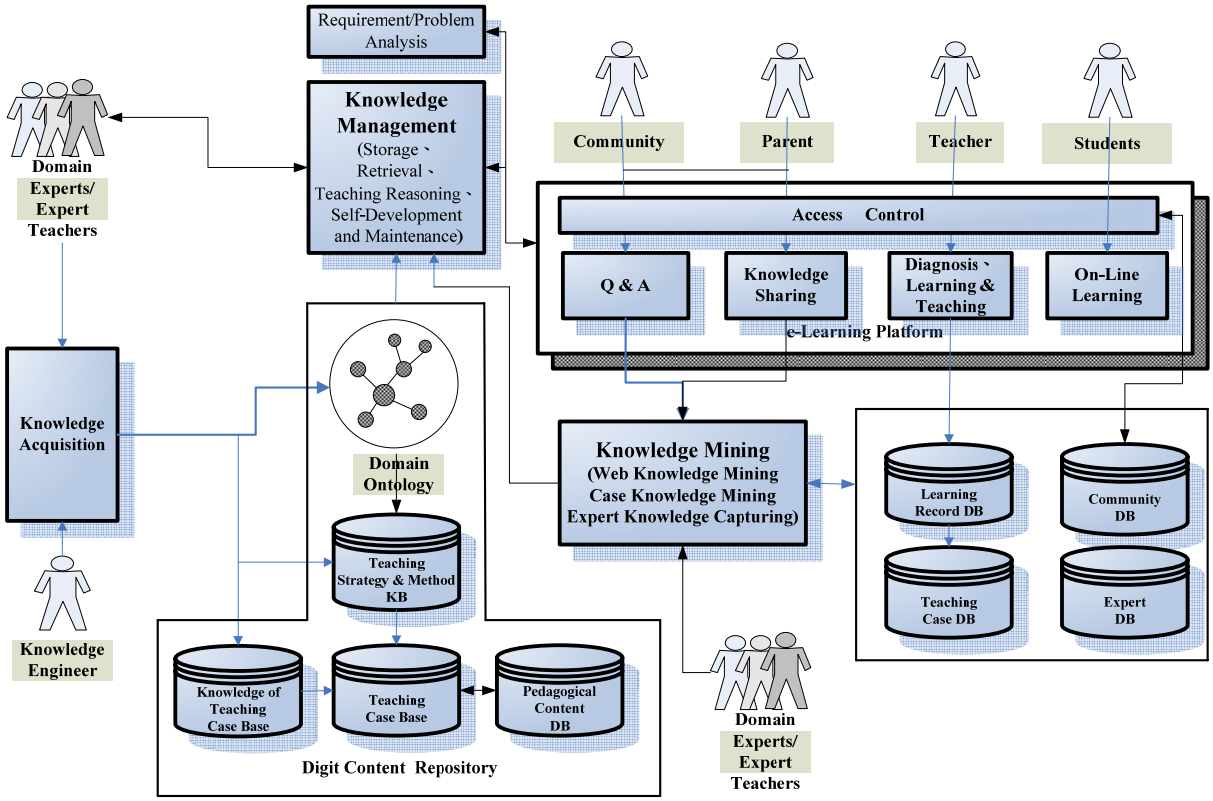

圖 3 數位學習平台系統之主要功能模組

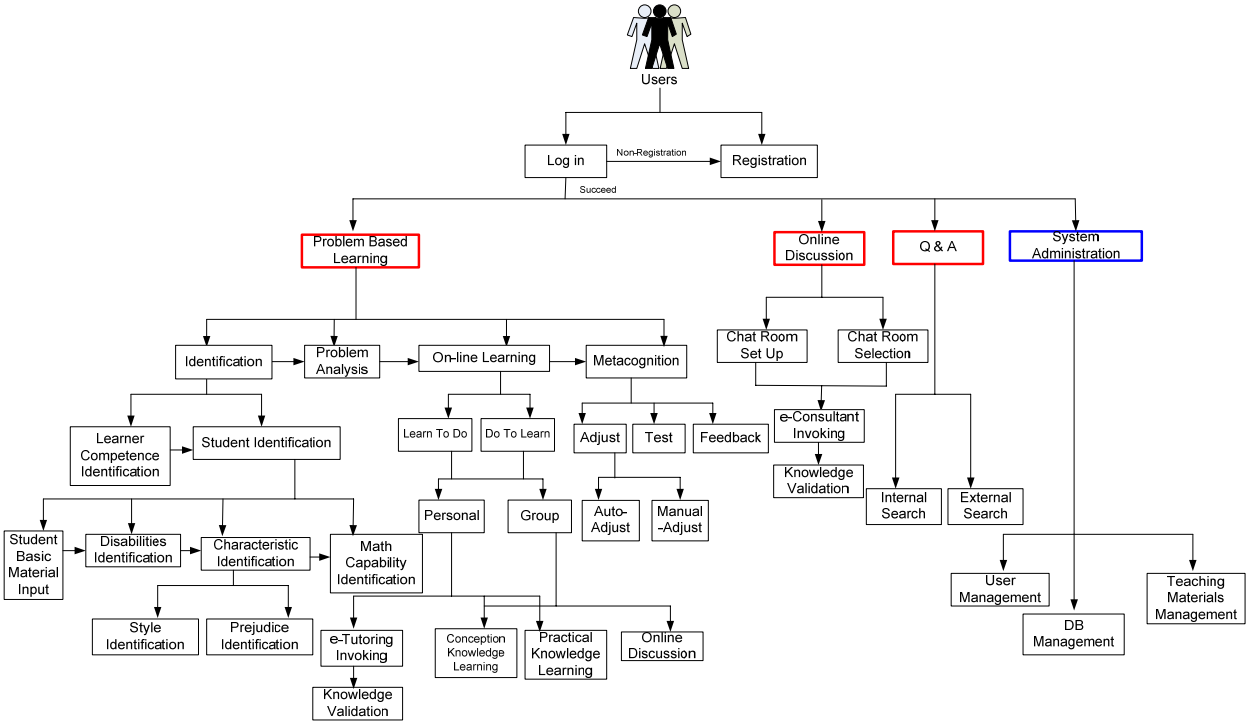

圖 4　數位學習平台系統之細部功能架構

(3) 數位學習平台功能模組之使用者介面設計 (請參閱輕度障礙學生數學教學之數位學習平台，網址：http://203.72.1.27/learn)

本數位學習平台使用者介面之設計圖，如圖5所示，使用者對象可分為系統管理者與前端使用者，平台提供給管理者「教材樣版管理」、「使用者管理」與「知識儲存管理」；另平台可以讓使用者進行「線上學習」，並提供「線上討論區」、「FAQ問題與回覆」、「相關資訊查詢」、「知識確認」之功能，詳細內容分述如下：

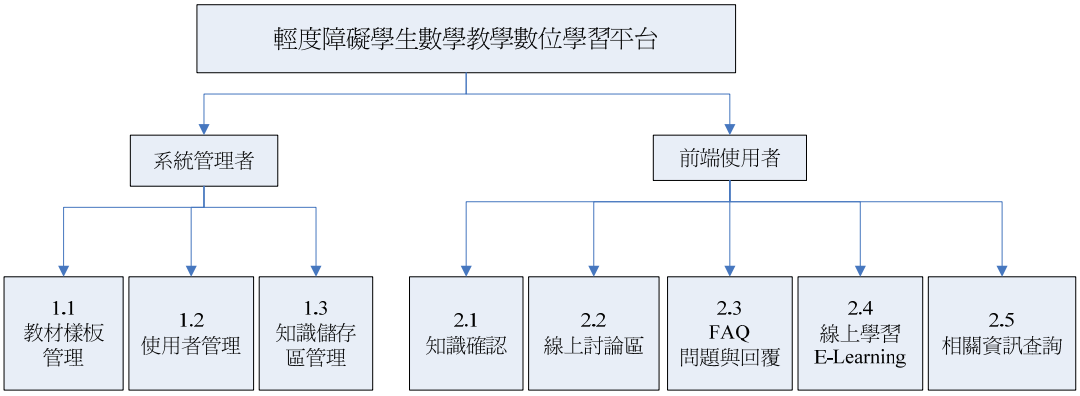

圖 5　**數位學習平台功能模組之使用者介面設計**

1.1 教材樣版管理

管理者登入系統後選擇教材樣板功能，選擇後系統會列出教材樣版資料清單，管理者可就既有的資料進行修正或刪除，也可新增教材樣版資料（圖 6）。

1.2 使用者管理

管理者登入系統後選擇使用者帳號管理功能，選擇後系統會列出使用者帳號資料清單，管理者可就原有的資料進行修正或刪除。

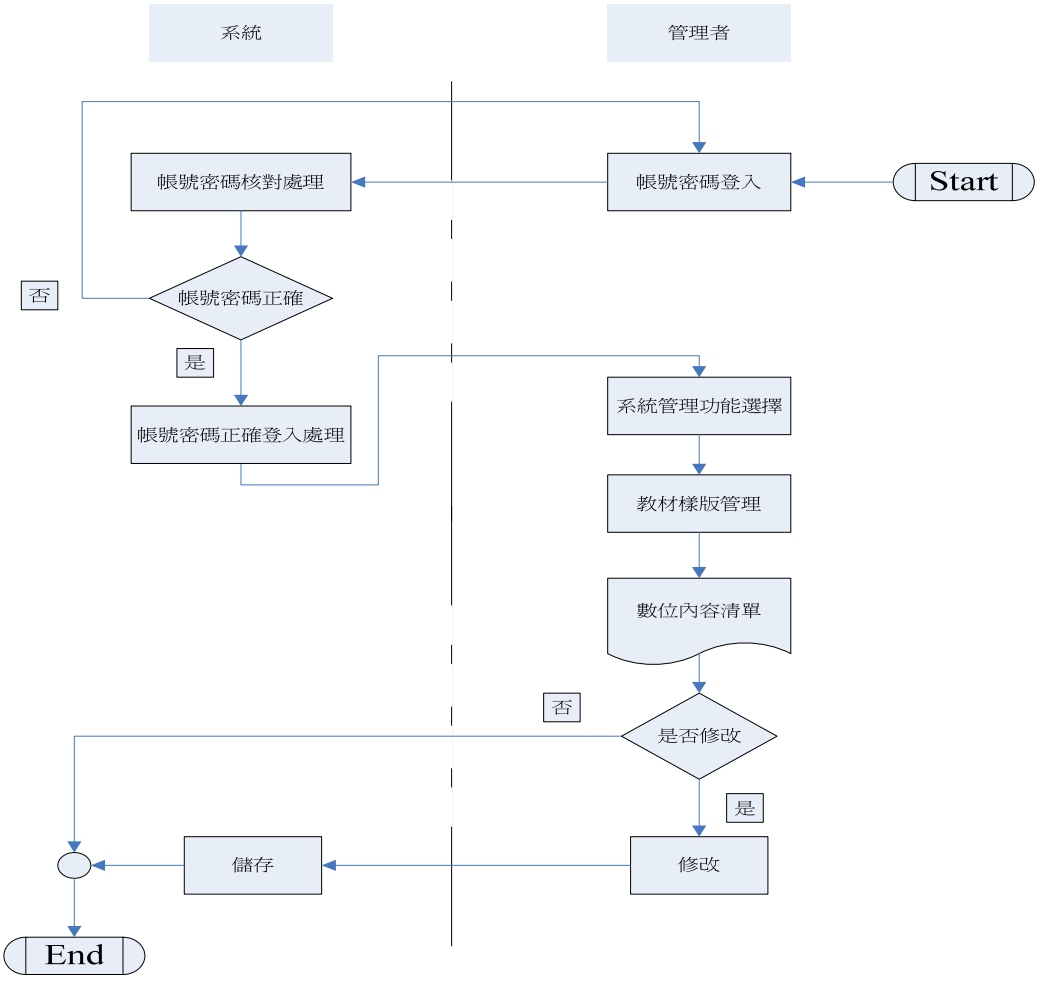

圖 6 教材樣版、使用者管理流程圖

1.3 知識儲存區管理

管理者登入系統後選擇知識儲存區管理功能（圖 7），選擇後系統便列出相關知識庫內容，管理者可就知識儲存區內資料進行修正。

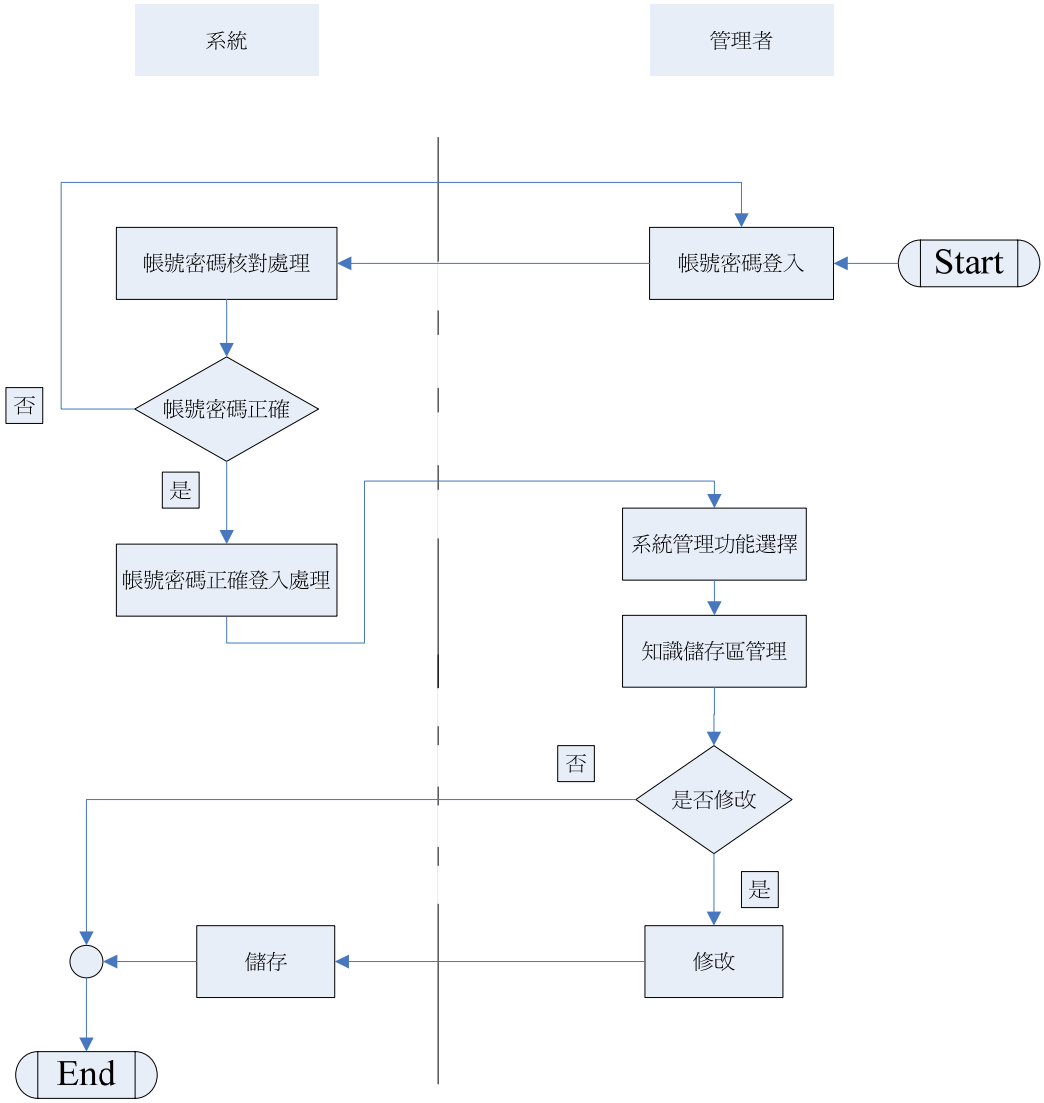

圖 7 知識儲存區管理流程圖

2.1 知識確認

專家使用者登入系統後選擇知識確認功能(圖 8)，選擇後系統便列出相關待審之知識清單內容，專家使用者可就待審知識資料進行確認審核及加入註解，金行知識的篩選。

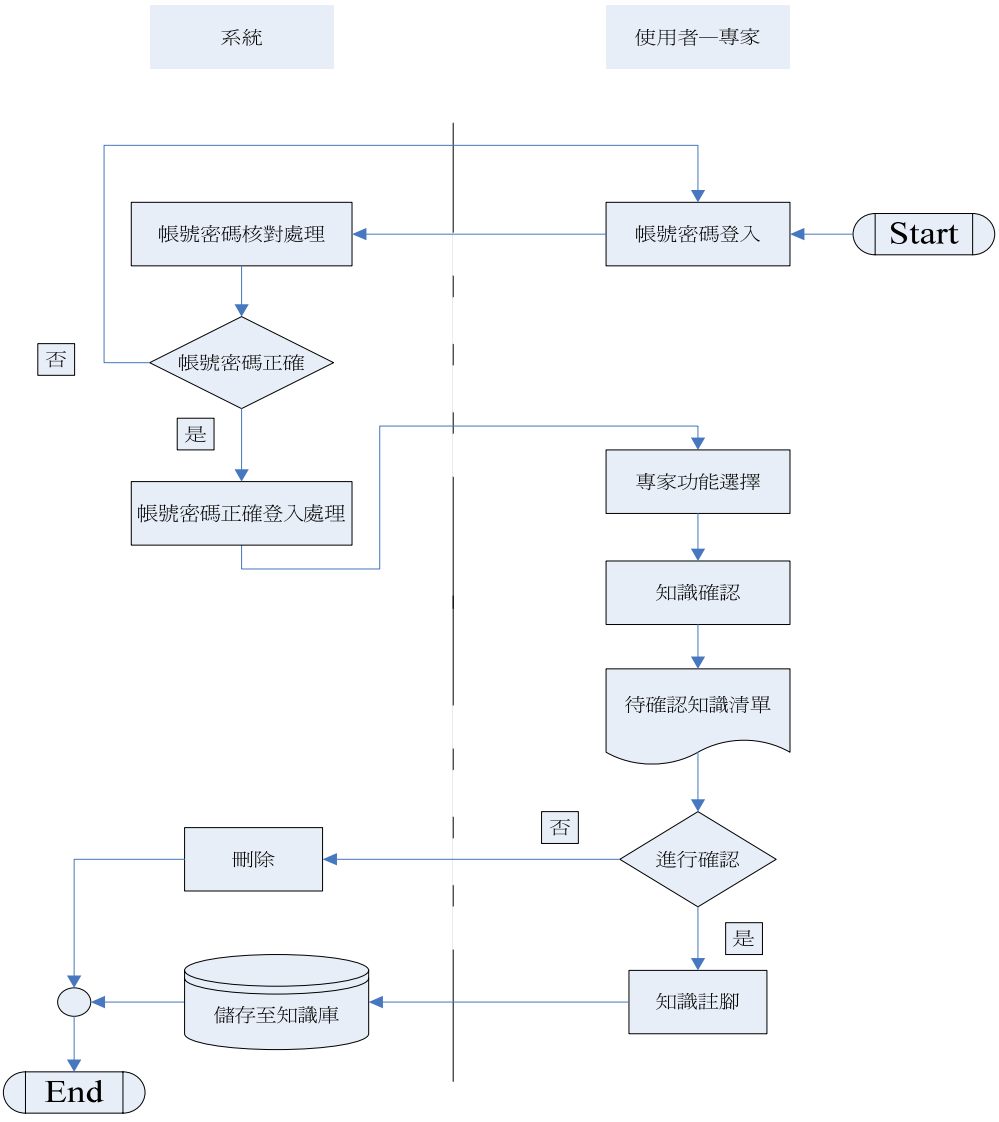

圖 8 知識確認流程圖

2.2 線上討論區

使用者欲進行線上討論時，使用者需將欲討論問題輸入至頁面透過系統對問題的拆解處理，將使用者分派至合適該問題的相關討論區中，如果使用者覺得不符合討論議題則可以另外預約一討論區，系統將會真對使用者的問題與領域專家進行配對，找出合適的領域專家，並預約雙方合適的時間再進行線上討論，系統並將討論結果進行節錄篩選，將這些知識經由領域專家的確認儲存至知識庫中（見圖 9）。

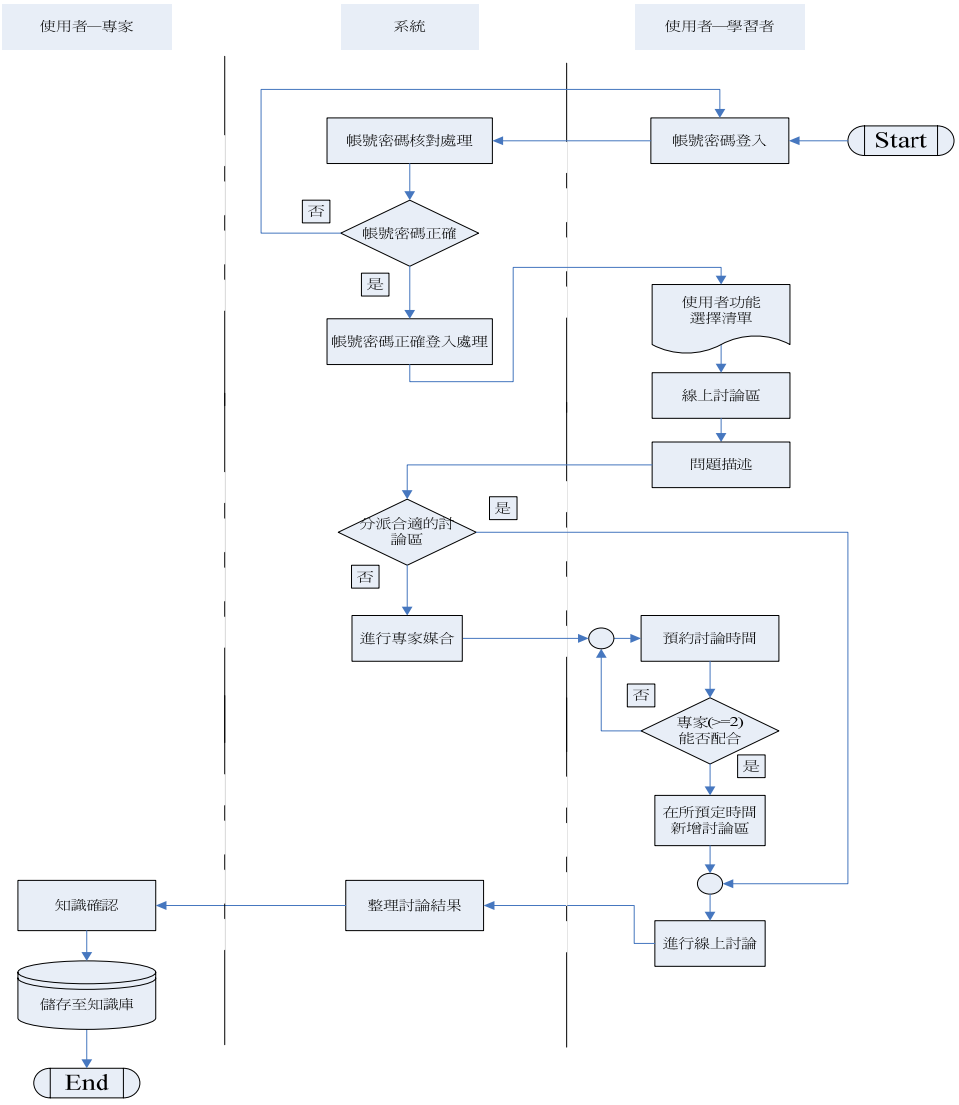

圖 9 線上討論區流程圖

2.3FAQ 問題與回覆

使用者至本系統進行相關問題的發問，系統會真對使用者的問題進行拆解並比對答案庫中是否有合適的資料，有合適的資料則直接回覆至頁面。如果搜尋不到資料則啟動網路搜尋機制，至網路上搜尋與問題相關的資訊加以整理過後再將答案回覆給使用者。如果回覆的資訊無法滿足使用者的需求，則系統會將問題儲存於暫存區中等待專家或是其他學習者的回覆（見圖 10）。

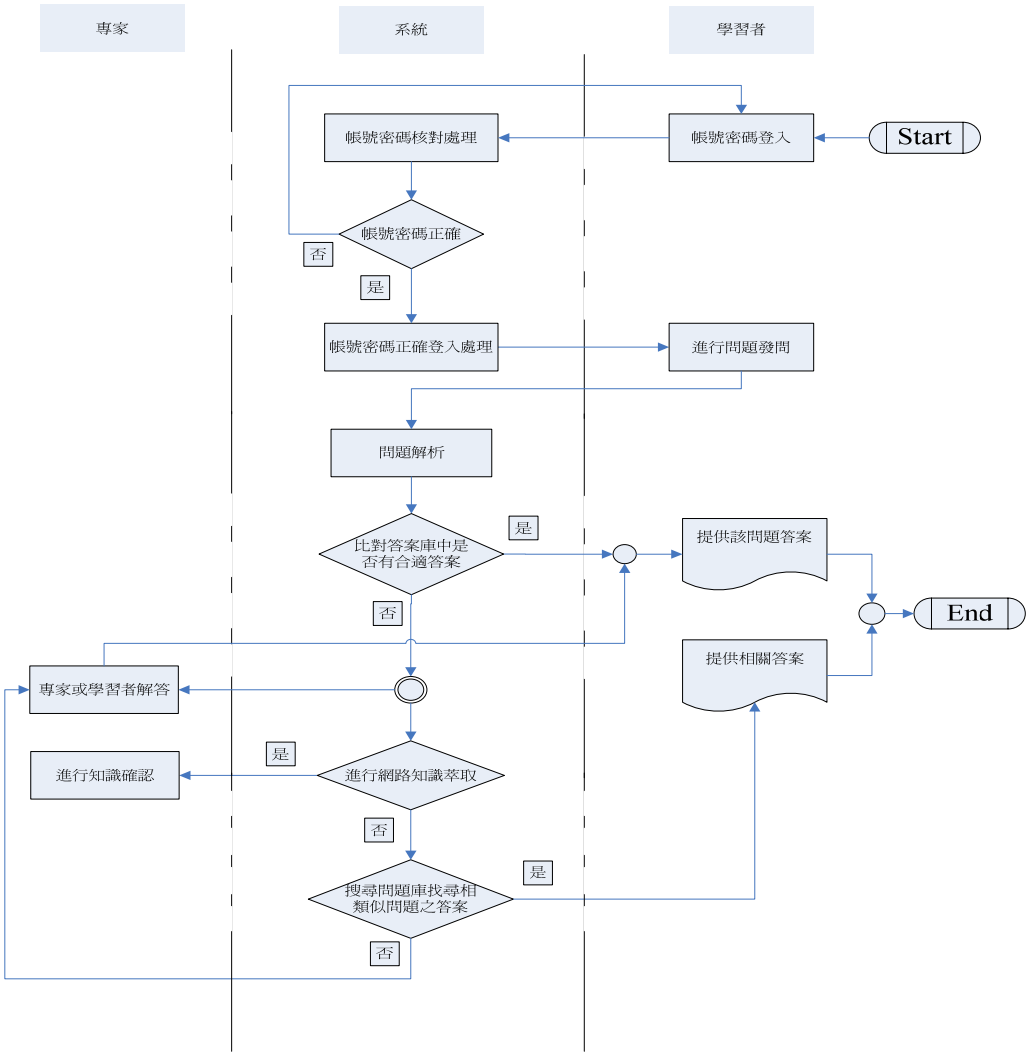

圖 10 FAQ 問題與回覆流程圖

2.4 線上學習

線上學習主要分成 1.先學後教、2.邊學邊教兩種學習模式。在進行線上學習之前系統會先進行教師職能的鑑定，鑑定出教師的教學職能後再進行學生能力的鑑定，鑑定學生的障礙程度與學習興趣，經由這些基本的能力鑑定過後系統會要求使用者將教學上的問題描述於系統上，之後系統會請使用者挑選合適自己的學習模式後再進行線上學習。

- 先學後教模式主要在針對這些問題設定學習目標，在針對學習目標建立學習物件及程序再讓使用者進行線上的學習（圖 11）。
- 邊學邊教模式主要差別在使用者可以針對學習的實際情況進行提出學習需求的變更，經由專家的確認後再進行學習物件的調整，調配出最符合使用者問題需求的學習模式（圖 12）。

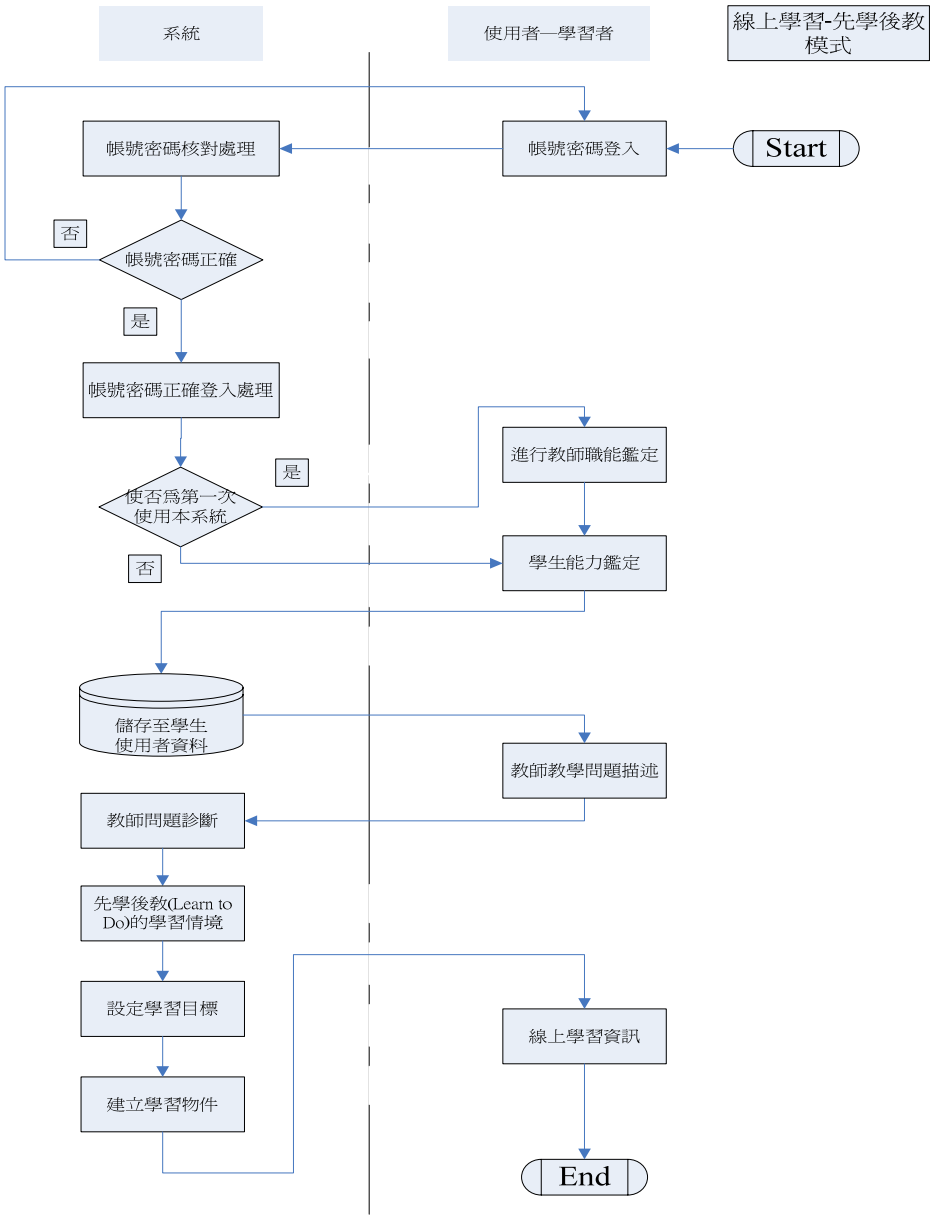

圖 11 線上學習(先學後教模式) 流程圖

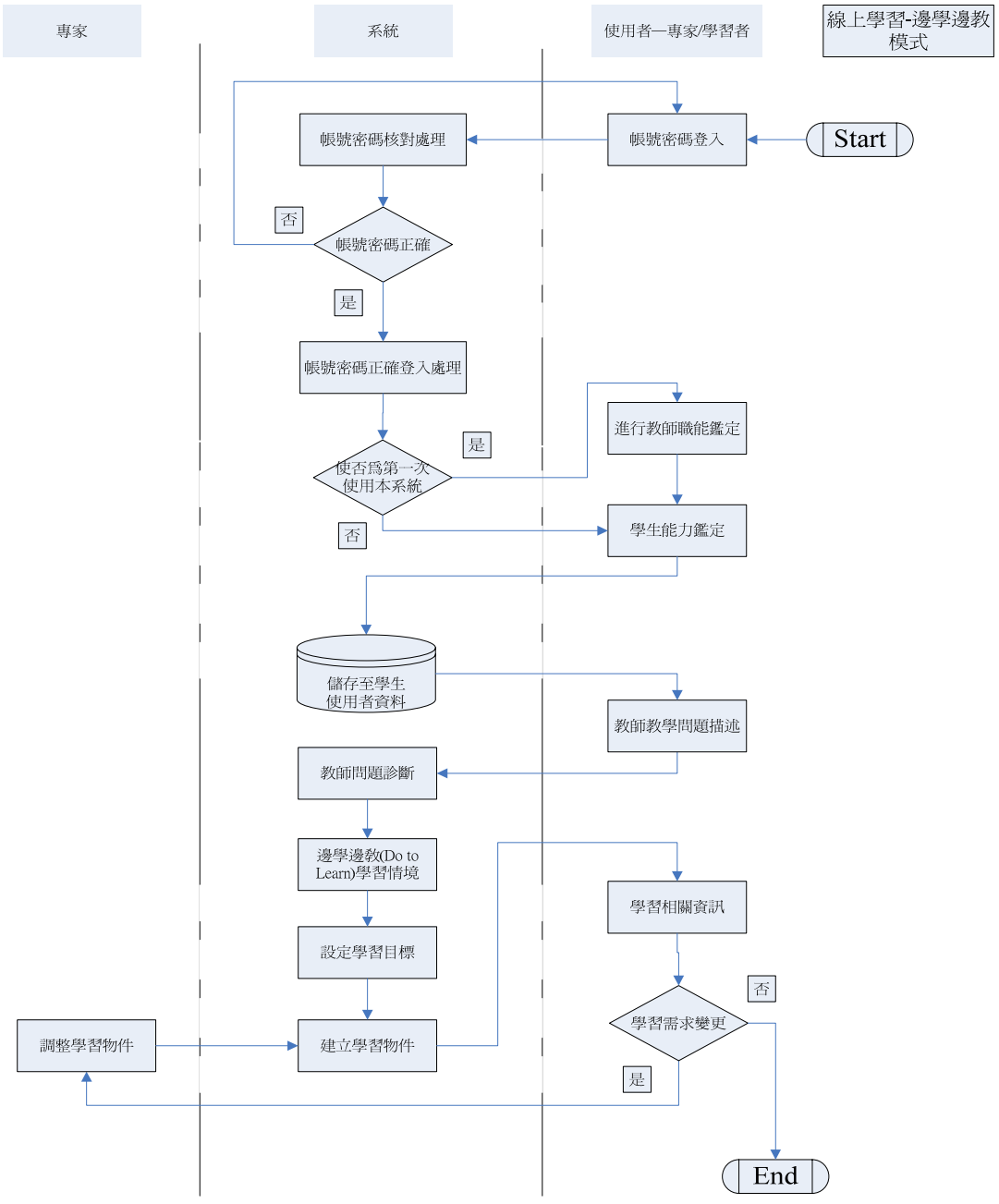

圖 12 線上學習(邊學邊教模式) 流程圖

2.5 相關資訊查詢

相關資訊的查詢，系統主要提供使用者可以進行領域知識、學習歷程、專家資訊等等的查詢，讓使用者可獲得相關所需的資訊內容（圖 13）。

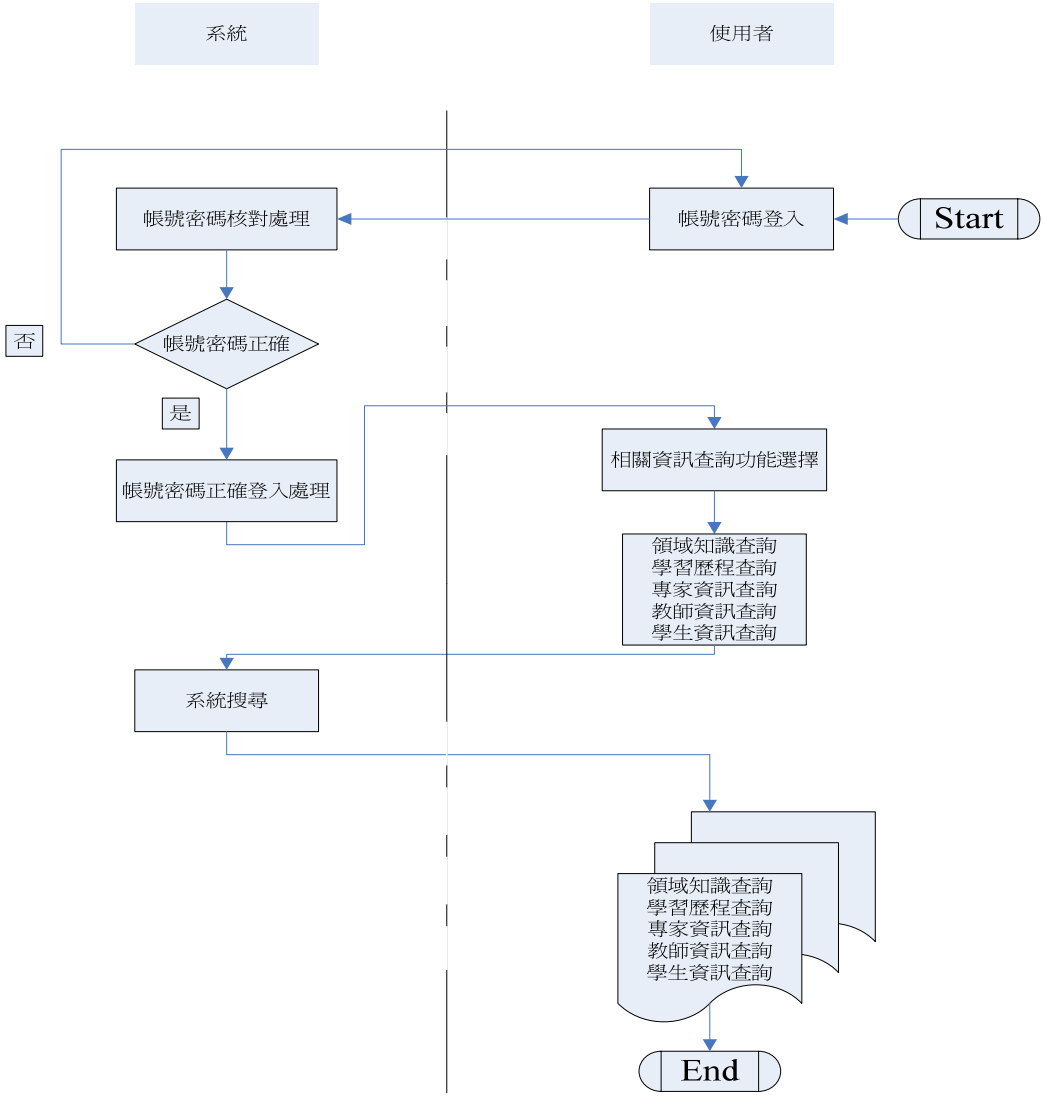

圖 13 相關資訊查詢流程圖

- 後台管理功能

左頁介面為後台管理主要選單.
目前階段將顯示所有的管理規則，未來在使用者登入之後，依據權能顯示可用之功能.

[新增伺服器頁面]
由於網頁檔案(web page)可能放置於不同的伺服器，因此必須記錄伺服器來源.
目前階段，所有的網頁檔案都放置於同一伺服器底下作業.

[新增用戶頁面]

這是現階段管理平台裡測試新增會員方式.

該新增會員的方法將會在一般的網頁裡以較視覺化的方式提供一般使用者註冊使用.

[新增角色頁面]

每個用戶將會賦予不同能力的角色.

該處是新增該平台中可能會使用到的角色清單

[以用戶加入到角色]

該處是以用戶為基準(User-Base to Role)，來設定不同的角色.

由下圖例中 User_ID: fish1984 同時擁有 SysAdmin、教師、學生…等角色能力.

用戶： 亞由宇

User to Roles

User_ID	Role_ID	Delete
fish1984	0000	刪除
fish1984	1001	刪除
fish1984	1002	刪除

Role List

Add	ID	Name	Description	AOrder
加入	0000	SysAdmin	系統管理員	
加入	1001	教師	教師群組	
加入	1002	學生	學生群組	

[以角色加入到用戶]

該處是以角色為基準(Role-Base to User)，來加入至不同的用戶裡.

角色： SysAdmin

Role to Users

Role_ID	User_ID	Delete
0000	fish1984	刪除
1001	fish1984	刪除
1002	fish1984	刪除

User List

Add	ID	Name	Description	EMail
加入	fish1984	亞由宇	測試帳戶	fish1984@pchome.com.tw
加入	Demo	Demo	Demo	Demo@Demo.com

[新增物件]

在這裡每個物件等於每一個網頁檔案(WEB Page).

新增物件裡面，可以將所有將受到控管的網頁，或是可能需要控管的網頁新增到該 Object List 裡.

Object

ID	Name	Description	Server	URL
S0001	新增伺服器	新增伺服器	localhost	/PageAdmin/faces/Server_Add.jsp
S0002	新增使用者	新增使用者	localhost	/PageAdmin/faces/User_Add.jsp
S0003	新增角色	新增角色	localhost	/PageAdmin/faces/Role_Add.jsp
S0004	使用者對角色管理	使用者對角色管理	localhost	/PageAdmin/faces/UtoR_Manager.jsp
S0005	角色對使用者管理	角色對使用者管理	localhost	/PageAdmin/faces/RtoU_Manager.jsp
S0006	增加物件	增加受管理的物件	localhost	/PageAdmin/faces/Object_Add.jsp
S0007	物件對角色管理	物件對角色管理	localhost	/PageAdmin/faces/OtoR_Manager.jsp
S0008	角色對物件管理	角色對物件管理	localhost	/PageAdmin/faces/RtoO_Manager.jsp

[以物件加入到角色]

該處是以物件為基準(Object-Base to Role)，來加入至不同的角色裡.

物件： 新增伺服器

Object to Role

Object_ID	Role_ID	Delete
S0001	0000	刪除
S0002	0000	刪除
S0003	0000	刪除
S0004	0000	刪除
S0005	0000	刪除
S0006	0000	刪除
S0007	0000	刪除
S0008	0000	刪除

Roles List

Add	ID	Name	Description	AOrder
加入	0000	SysAdmin	系統管理員	
加入	1001	教師	教師群組	
加入	1002	學生	學生群組	

[以角色加入到物件]

該處是以角色為基準(Role-Base to Object)，來加入至不同的物件裡.

角色： SysAdmin

Role to Object

Role_ID	Object_ID	Delete
0000	S0001	刪除
0000	S0002	刪除
0000	S0003	刪除
0000	S0004	刪除
0000	S0005	刪除
0000	S0006	刪除
0000	S0007	刪除
0000	S0008	刪除

Object List

Add	ID	Name	Description
加入	S0001	新增伺服器	新增伺服器
加入	S0002	新增使用者	新增使用者
加入	S0003	新增角色	新增角色
加入	S0004	使用者對角色管理	使用者對角色管理
加入	S0005	角色對使用者管理	角色對使用者管理
加入	S0006	增加物件	增加受管理的物件
加入	S0007	物件對角色管理	物件對角色管理
加入	S0008	角色對物件管理	角色對物件管理

一個用戶在登入時，將能取得該用戶所用有的角色陣列，藉由角色陣列，即可取得所賦予的物件權能.
所有被控管的網頁，至少引用一段系統的源碼(例如：檢測帳戶)，該物件控管方有作用.
JSP 引用範本如下：

宣告方式

```
<jsp:useBean id="PageAdmin" class="pageadmin.SessionBean1" scope="session"/>
```

引用後即可使用 PageAdmin 相關屬性，目前可用屬性列表如下.

```
PageAdmin.setUser_ID            //輸入帳號
PageAdmin.setUser_Password          //輸入密碼
PageAdmin.Login                 //執行登入
PageAdmin.Logout                //執行登出
PageAdmin.isLogin                   //檢查是否登入
PageAdmin.UtoRList              //執行使用者列出角色清單
PageAdmin.getUtoRList[]             //取得使用者列出角色清單
```

範例 EX1：
顯示 Hello Word

```
<jsp:useBean id="PageAdmin" class="pageadmin.SessionBean1" scope="session"/>
<jsp:getProperty name="PageAdmin" property="helloWord" /> //顯示 HelloWord 測試字串
```

範例 EX2：
顯示 Hello Word

```
<jsp:useBean id="PageAdmin" class="pageadmin.SessionBean1" scope="session"/>
<%
        out.println (PageAdmin.helloWord);
%>
```

範例 EX03：

傳入帳號密碼，並且檢查帳號密碼是否通過驗證.

```
<jsp:useBean id="PageAdmin" class="pageadmin.SessionBean1" scope="session"/>

<jsp:setProperty name="PageAdmin" property="user_ID" value="fish1984" /> //對 user_ID 屬性傳入一個值
<jsp:setProperty name="PageAdmin" property="user_Password" value="fish1984" /> //對 user_Password 屬性傳入一個值

<%
        PageAdmin.Login();   //嘗試登入
        if (PageAdmin.isLogin()==true){   //檢查是否登入成功
                Cookie CK = new Cookie ("User_Name",PageAdmin.getUser_ID());
//寫入到 Cookie
                response.addCookie (CK);
                out.println ("登入成功");
        }else{
                out.println ("登入失敗");
        }
%>
```

3.2 網路資料與知識萃取技術之開發 (請參閱附件一)

傳統教學受限於時間與空間，隨著科技及網際網路時代的來臨，數位學習平台可解決傳統教學所面臨的困難，而網路上充滿著豐富的知識，如何以自動化的方式有效的利用網路上的資料提供使用者所需的知識是一項很大的挑戰。針對此一問題，設計一個智慧型之網路知識擷取及建構機制的系統架構，如圖14所示，此架構主要包含三個部分：(1) 使用者語意分析機制(Semantic Analysis Mechanism，SAM)、(2) 整合式網頁搜尋引擎機制(Integrated Mechanism of Search Engines on Web)和(3)網頁知識擷取及建構機制(Knowledge Extraction and Construction Mechanism)。

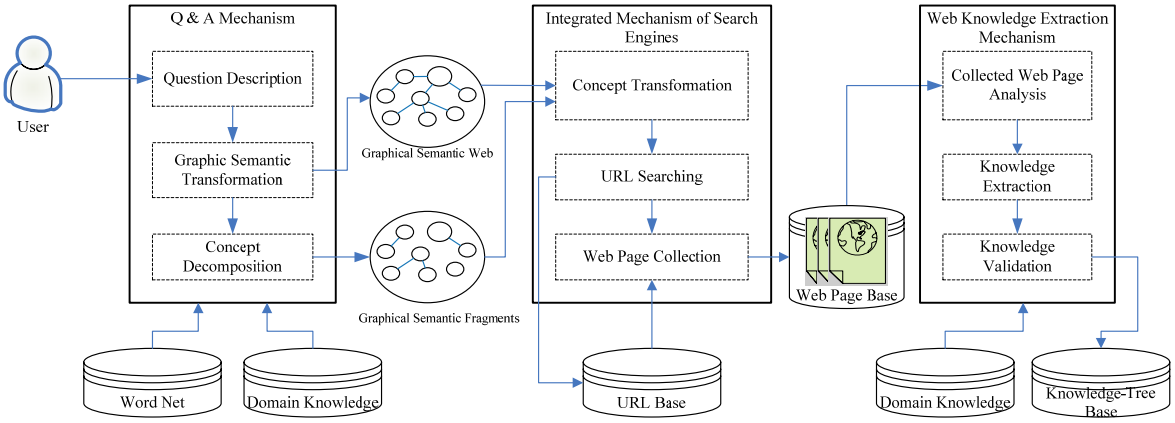

圖 14 網路資料與知識萃取之架構

所設計之架構能夠解析出使用者所提出之問題的涵義並找出以不同語法及同義字的表達方式，將原始的問題轉成一個圖形化的語意網路模型，以此模型來描述使用者所提出問題的概念；以此模型再配合整合式搜尋引擎機制來進行相關知識的搜尋，找出符合使用者問題之網頁；透過知識擷取機制擷取相關的知識，進一步建構成知識樹，最後將知識回覆給使用者。從學習者(使用者)的觀點分析網路資料與知識萃取架構的運作，步驟如下：

Step1：學習者以自然語言方式輸入待解的問題。

Step2：問題透過中文斷詞處理後，形成關鍵詞與圖形化語意概念圖。

Step3：在數個搜尋引擎中以關鍵詞搜尋相關網頁。

Step4：搜尋的網頁以標準化方式加以儲存。

Step5：將網頁內容與圖形化語意概念圖加以比對與篩選。

Step6：將網頁內容以圖形化語意概念圖擷取成知識並建構成知識樹加以儲存。

Step7：將知識交由專家評定與驗證。

Step8：將知識傳達給學習者

網路資料與知識萃取技術架構與相關技術之探討，已發表於 An Intelligent Web Knowledge Extraction Framework to Support E-Learning Content Collection, WorldConference on E-Learning in Corporate, Government, Heathcare, & Higher Education, Hawaii, USA, 2006.，請參閱附件一。以下將針對該架構所組成之「問題與需求語意分析」、「網路知識搜尋與過濾」與「網路知識編譯與擷取」分別說明之。

(1) 利用 5W1H 結合本體論 (請參閱附件二)

人們遇到問題時，往往會藉由搜尋引擎來尋找相關資訊，因此，透過網際網路獲取資源已成為數位學習有利的工具之一，藉由搜尋引擎之便找尋相關資訊，不僅迅速與方便，同時可以解決資訊匱乏的問題。然而，當使用者有問題時想獲得精確的答案時卻總是不得其門而入，可能輸入關鍵字後獲得許多雜亂的資訊，或是當使用者對此領域不了解時輸入關鍵字卻找不到想獲得的資訊，無論是在怎樣的平台上搜尋許多使用者都遇到這樣的一個問題，目前只能依靠使用者對於搜尋的資料逐步的輸入相關的關鍵字來搜尋所需問題之解答。

本研究提出一利用 5W1H 方法來解析出使用者所描述問題之語意，利用斷詞規則發現自然語言之關鍵字，以統一模型語言(Unified Modeling Language，UML)中之類別圖關係來建構出語意網概念模型，配合 ontology 技術加以延伸出更廣闊之概念，將建構的概念加入權重及過濾，最後以圖形化語意網表示，不僅可以讓使用者了解問題所衍生的語意，更可以讓使用者與電腦進行有效的溝通，進而轉成電腦可理解的語意，搜尋出使用者真正所需的資料。

(2) 應用領域本體論設計整合網路上搜尋引擎做網路知識搜尋與過濾 (請參閱附件三)

網路上的資訊是屬於一個超大型資料庫，能提供查詢服務的資訊軟體系統，稱為搜尋引擎(search engine)，透過網際網路獲取資源已成為數位學習有利的工具之一，藉由搜尋引

擎之便找尋相關資訊，不僅迅速與方便，同時可以解決資訊匱乏的問題，但是由於目前搜尋引擎所搜尋之知識量往往重複性太高，甚至有搜尋結果不符合需求的情況發生，既浪費頻寬且效能大幅降低。由於知識的蒐集、獲取、整合、儲存、管理、分享與運用之重要性與日驟增，如何正確的從使用者的觀點透過網路獲取正確的資訊且有效率地轉化成知識是學者長久來所追求目標之一。

利用前述之技術擷取出使用者自然語意問題之關鍵字，利用搜尋引擎(search engine)尋找相關資訊，並結合網頁內容探勘(web content mining)、資訊檢索(information retrieval)相關技術與導入領域實體(domain ontology)概念，針對搜尋後的摘要及標題進行資訊含量之計算，透過相關演算法過濾格式不完整、重覆性與廣告之內容，其後依資訊含量給予權重與排序，提供給使用者較貼切原意的網頁內容，進而提供相關性與重要性的參考，希望可有效避免使用者浪費精神與時間自行過濾檢索。

(3) 以領域本體為基礎之概念地圖自動化建構與文件分類機制 (請參閱附件四)

當學習平台之知識庫或教材庫無法滿足學習者知識的需求或解決問題時，則須由專家增加教材庫或知識庫的教學內容(content)，無法滿足立即回饋的需求。因此，本研究將設計網路知識擷取及建構之機制，此機制不僅能有效改善教學平台有限的知識庫或教材庫之不足，而且能使知識庫隨著使用者的使用不受限制的向外延伸及深入問題的核心。但如何以自動化的方式針對學習者想了解的問題，在網路上搜尋答案進而建構成一個有組織有系統的知識庫用於支援學習者進行學習的活動是一項大的挑戰。

利用前述問題與需求語意分析、網路知識搜尋與過濾之技術，提供給使用者較貼切原意的網頁內容，可以避免使用者浪費精神與時間自行過濾檢索，但是對於網路知識之儲存與最短時間內將不同的知識來源組合呈現給使用者仍有改進之空間，因此本機制乃藉由本體論與自然語言處理的結合把使用者所輸入的詢問句子分析，再將網頁內容與學習者的問題進行比對，不僅可依照其網頁內容分群讓使用者能更迅速的找到所需的文件，而且可以經過一連串的擷取程序找出各文章中符合的段落進而組成知識，甚至可以進行知識呈現與驗證的工作。透過上述之程序，期望能以最適性化的方式將隱含在網路中的知識提供給使用者，達到網路知識擷取與知識分享之目的。

3.4 知識存取權限與技術之研究 (請參閱附件五)

本階段利用本體論具概念的描述及概念與概念間關係表達的能力，來描述輕度障礙學生數學教學之數位學習平台的知識內容，進而透過管理概念層的知識本體來及管理本平台的知識。主要提出一符合本平台工作模式的本體論為基之知識存取控制的方法，主要包含(1)知識存取控制模型設計(2)知識權限擴展模型設計及(3)知識存取控策略架構設計，以滿足在本平台上所有工作者或學習者，使他們能根據工作中對知識的需求及知識存取的權限，使用分散在平台上的所有知識，促使知識能夠快速及安全的被分享至正確的人、時間及位置。本階段預期之具體產出包括：知識架構模型、知識存取控制模型、知識權限擴展模型、知識存取控制策略架構及知識存取控制機制。

本階段首先進行知識架構模型設計，主要任務及產出如下：

(1) 知識表達設計

本研究提出之三層式知識架構模型，依序為概念層、知識索引層及知識實體層。知識概念層(Conceptual layer)主要表達領域內的主要概念，包含組織、活動及產品，分別以本體論來表示；知識索引層(Knowledge Index Layer)主要描述概念層儲存於實體層之索引資訊，例如某一「案例」之索引資訊結構；知識實體層(Knowledge Physical Layer)為儲存不同類型之資料庫，例如法則知識庫、案例式知識庫或 XML 文件庫等。其知識表達的方式如下：

- 知識概念層：知識在資訊系統中有很多的呈現方式，例如以資料庫、XML、案例或其他，雖然本體論無法表達所有的知識，但是使用本體論可具有(i)知識分享：本體論的標準格式使知識分享更容易；(ii)知識再利用：相同的領域可以重複利用之前所制定的本體論，在不同系統中使用。且本體論在知識呈現能力上，對於“概念性的知識”可以具體的呈現。
- 知識索引層：透過知識索引層可以指引(Pointer)概念層知識對映到實體層知識的某些具體元素，例如：當一老師在進行數學教案的設計工作，所需的知識(Know-what、Know-why 及 Know-how) 可以結構化成許多的儲存模式，儲存於知識庫中，例如 Case、Rule、XML document 的型態。
- 知識實體層：儲存區裡知識的內容包含 RDB、XML、Doc、Graph、Rule-base 、Case-base 、Record 等，且又可依其結構分為結構化、半結構化與非結構化三大類，並可將實體知識分為三類：描述性的知識(Know-what)、程序性的知識(Know-how)、因果性的知識(Know-why)。

(2) 知識架構模型設計

本體論可視為一分類系統，當儲存區新增知識項目(文件、設計圖、圖表、記錄)時只要連結至主要分類，即連結至本體論裡一個至多個概念，因此本平台可以利用本體論來表達數學教學的知識，以協助管理使用者對於知識的使用權限，這一個工作者可以被允許存取什麼知識，依此建構一「知識架構模型」如圖15所示。

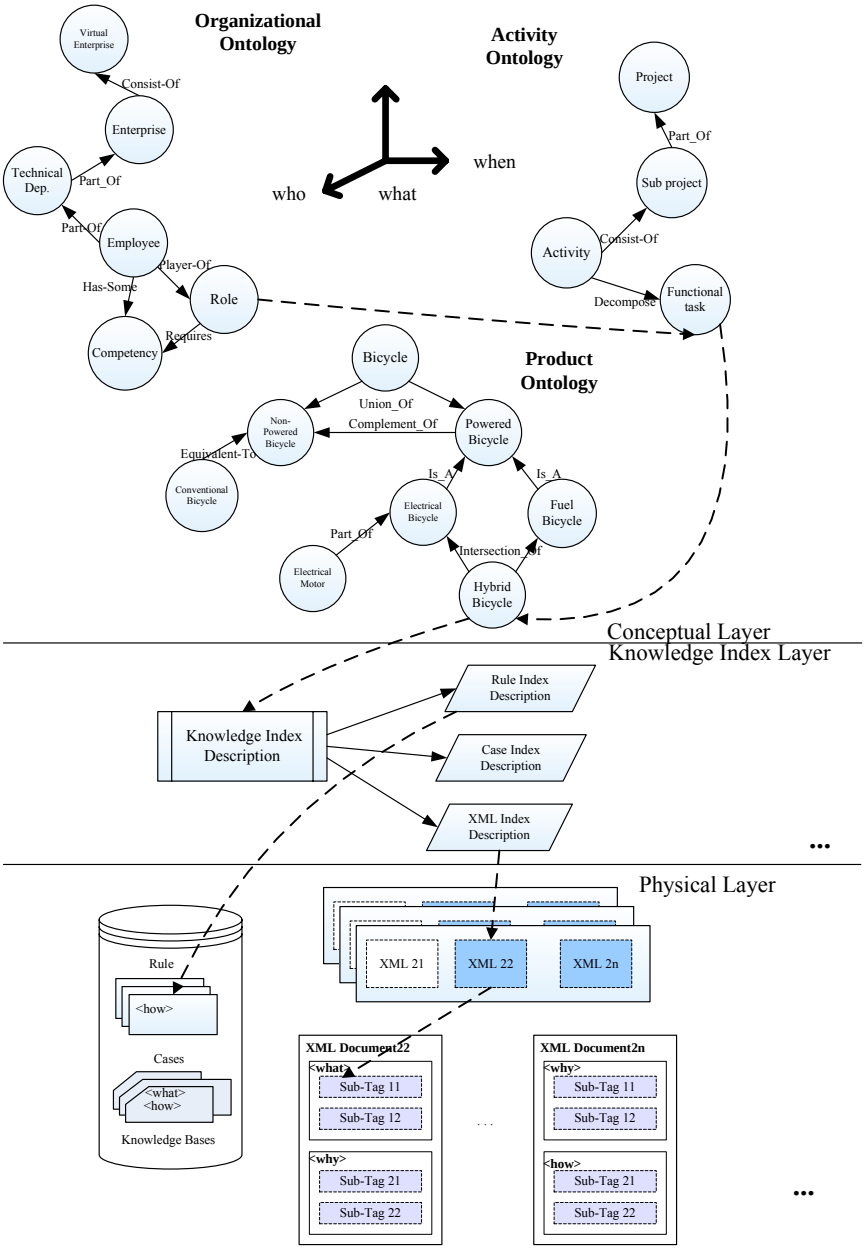

圖 15　Knowledge Framework Model

第三年主要任務為平台完成及測試，其完成工作為：

(1) 核心功能模組的驗證及測試：進行「核心功能模組測試」，之後進行平台「跨功能模組的測試」，執行「功能模組的改善」。

(2) 平台導入：利用子計畫(一)所收集建構之教學案例與模式與子計畫(二)所發展之知識管理引擎，實際建構一個完整之平台，提供教育工作者或輕度障礙生之家長進行系統驗證及實驗，首先進行「案例、資料及知識庫的建立」及「平台設定」，找來系統測試者包含專家、教師、家長及輕度障礙學生實際操作本系統，之後進行「初步的系統測試」並「評估平台之效能及進行改善計畫」。之後交由子計畫(一)之工作者進行「實驗設計」將被測試者分成實驗組及控制組，之後分析本平台在輕度障礙數學教學上所產生之效益。

(3) 追蹤及修正：針對實際驗證結果的缺失進行改善，以期達到預計的目標。

四、結論與討論

系統平台功能完成，知識庫內容仍持續建置與修改，平台請參考 http://elearn.ime.ncku.edu.tw/eLearn/index.jsp 。

參考文獻

附件 1

An Intelligent Web Knowledge Extraction Framework to Support E-Learning Content Collection

Chin-Bin Wang[1], Huimei Yang[3], Yuh-Min Chen[2], Hui-Chuan Chu[4], *Tsung-Yi Chen[1,2], Derchian Tsaih[1]

[1]Department of Electronic Commerce Management
Nan Hua University
Chia-Yi, Taiwan, ROC

[2]Institute of Manufacturing Engineering
National Cheng Kung University
Tainan, Taiwan, ROC

[3]Dept. of Business Administration
Tatung Institute of Commerce and Technology
Chia-Yi, Taiwan, ROC

[4]National University of Tainan
Tainan, Taiwan, ROC

*Corresponding author's email: tychen@cacerl.ime.ncku.edu.tw

Abstract: Traditional teaching methods are limited by time and space. With the arrival of increasingly advanced technologies and the internet era, e-learning platforms can solve many of the problems faced by traditional teaching. The internet is filled with knowledge; how to automatically and effectively use such information in fulfilling the needs of users is a great challenge. In focusing on this problem, this study has designed a system framework for an intelligent internet knowledge extraction and construction mechanism. The system framework designed by this study can analyze the problems raised by the user, subsequently finding and using graphical semantic network models of different language and synonyms; the system then uses this model to describe the concepts of the user problem. The model or a concept model derived from breaking down the model is used to extract keywords and search for relevant information using an integrated mechanism of search engines. A knowledge extraction mechanism then extracts the relevant knowledge from the resulting web pages, further constructing a knowledge tree, finally giving a response to the user.

Introduction

E-learning is one means of spreading and expanding knowledge; its primary difference with traditional teaching methods lies in that e-learning combines information technology and the internet in order to compensate for the flaw of time and space limitations in traditional education. After undergoing the conversion process of digitalization, the content of traditional teaching becomes easier to edit, compile, link, and organize; reusability and sharability are greatly improved as well. However, although e-learning platforms can solve many of the problems faced by traditional teaching, the creation of digital curriculum is costly in terms of both time and resources; such curriculum is also dwarfed by the incredible amounts of information available on the internet. When people face a problem, they generally use search engines to seek out relevant information. As a result, the use of the internet in obtaining resources has already become a powerful tool in e-learning; not only is using a search engine to find information quick and convenient, it can also resolve the problem of information deficiency. However, when users seek precise answers to a question, search engines often return excessive amounts of useless information after keywords are entered; users often do not find their desired information when they are unfamiliar with the relevant field of knowledge even after entering keywords. Regardless of which platform is used, many users face this sort of problem. Currently, users must be depended upon to continually enter related keywords in order to search for answers to their questions. Also, there is too much overlapping in the scattered information of the internet. This situation creates another problem – information overload. How to help users correctly obtain correct information and convert it to knowledge is one of the long-term goals of the research.

This study uses the education theory of problem-based learning (PBL) [1] as the core of the learning platform framework. After the students use natural language to describe the problem, they can instantly and dynamically acquire the learning content from the information or curriculum bases of the learning platform. When the bases of the learning platform do not satisfy the needs or solve the problems of the learner, experts are then needed to expand the content of the curriculum base or the knowledge base; the need for an instantaneous response cannot be satisfied. As a result, this study will design a knowledge extraction and construction mechanism on web; this mechanism can not only improve the problem of deficiencies in limited knowledge or curriculum bases. Also, as the user does not face limitations, the knowledge base can expand outwards and delve into the cores of problems. However, the questions of how to automatically address the problems of users and how to search for answers online and subsequently construct an organized, systematic knowledge base to support e-learning activities of users present a significant challenge.

Functional Framework Design

This study offers an intelligent internet knowledge extraction and construction framework to support the automatic expansion of knowledge bases in e-learning systems. Under this framework, focusing on analyzing the process of learners getting answers for their problems, we found many unsolved problems: (1) how to decipher the meaning of user problems, (2) how to integrate multiple search engines to search relevant web pages and subsequently filter, arrange, and store, (3) how to properly extract knowledge from web pages, and (4) how to evaluate the effectiveness and relevance of extracted knowledge. This section will first use an operation scenario to describe the usage process of learners for this framework, and then construct a functional framework based on this process.

Operation Scenario

Based on social construction theory [1,2], there should be group discussion in the environment of e-learning. In any stage of learning, learners can activate the group discussion area at any time and begin Q&A, online discussion, or web extraction. In order to clearly understand the relationship between learners and this study, we analyze the operation scenario of the entire system from the perspective of the user and then proceed with modal construction. An initial functional framework will be constructed based on this in the next section. An explanation of the steps of the operation scenario is as follows:

Step 1: Learners input unresolved questions using natural Chinese language.

Step 2: After the problem goes through Chinese semantic processing, key phrases and a graphical semantic concept figure are created.

Step 3: Search the database and give a response to the user's problem; if no related knowledge is found, proceed to Step 4.

Step 4: Using the key phrases, search for relevant web pages using multiple search engines.

Step 5: Compare and select web page contents and graphical semantic concept figure.

Step 6: Extract web page contents as graphical semantic concept figures and construct a knowledge tree.

Step 7: Offer the knowledge to an expert for judgment and evaluation.

Step 8: Communicate the knowledge to the learner; the learner can further browse the knowledge in the knowledge tree.

Functional Framework

Based on the operation scenario described above, we will begin the task of designing the system framework. The system framework is composed of three parts: (1) a semantic analysis mechanism (SAM) for user questions, (2) integrated mechanism of search engines on web, and (3) a knowledge extraction mechanism, as shown in Figure 1.

Besides providing a graphical user interface and allowing the user to enter questions and browsing search responses, the question and answer mechanism has a graphical semantic transformation mechanism and a concept decomposition mechanism at its core. The graphical semantic transformation mechanism also contains two sub-modules: a user semantic analysis module and a graphical semantic net transformation module. Through these two sub-modules, natural language processing and segmentation can be used to convert user-described questions to network structure graphs of semantic concepts [3]. Through the concept decomposition mechanism of the framework, the complete concept is analyzed and taken apart to be used in the integrated multi-search engine mechanism for URL search; the goal is to find web addresses appropriate for the question. The integrated multi-search engine mechanism uses the concept transformation mechanism: the concept graphical figures having been taken apart, the sub-concept graphical figures are processed by the key word extraction mechanism to extract key words; the specialized keyword composition mechanism, based on the qualities of the four search engines chosen by this study, reconstructs the keywords; the four search engines are then used to search for URLs, after which processing and arrangement techniques are used to perform analysis of URL relevance and appropriateness [4,5,6,7]. The web knowledge extraction and construction mechanism can assign an appropriate weight value based on the importance of web pages, then further categorize and arrange them, finally performing knowledge extraction; the knowledge is then transferred to experts for testing and evaluation. If it is appropriate for the described problem, then it will be stored in the database.

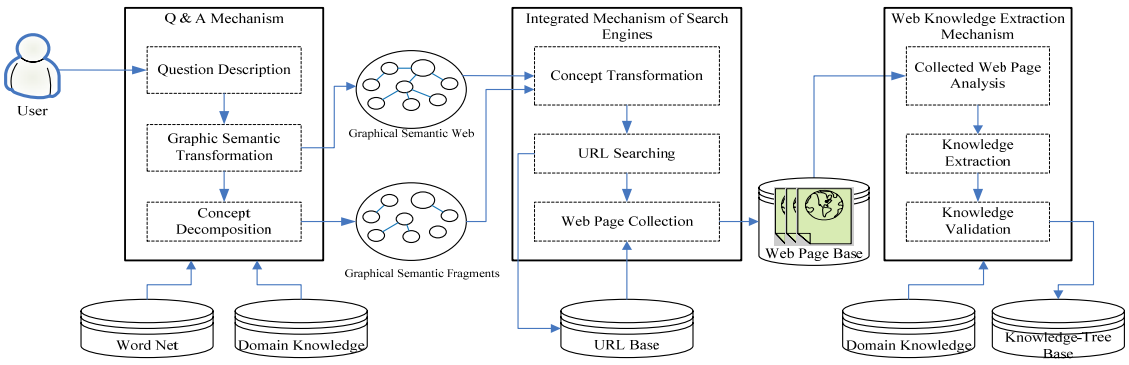

Fig.1 Functional Framework of Web Knowledge Extraction and Construction System

Graphical Semantic Net Transformation Mechanism

The graphical semantic net transformation mechanism (as in Figure 2) includes: a semantic analysis module, a graphical semantic net transformation module to help us produce graphical semantic nets from described problems, and a concept decomposition mechanism. This mechanism uses a word building rule base, a Chinese lexicon, a Domain lexicon, and a segmented words rule base to perform deconstruction on the problem described by the user; the mechanism then uses a part-of-speech base, a part-of-speech rule base, and another type of part-of-speech base to perform characteristic marking processing phrases. Semantic network conversion uses the results of the marking described above, a part-of-speech base [3], a synonym lexicon base [8], and a stop word base to filter redundant phrases and form key concepts. Domain ontology and knowledge base is used as the basis for constructing concept models; key concepts are used as centers for extending semantics, forming extended conceptual semantic nets. The term frequency inverse document frequency (TF-IDF) method is used; TF calculates the frequency of appearances of a key word in a given document, while IDF calculates the frequency of appearances by key words in a category of documents [9, 10, 11].

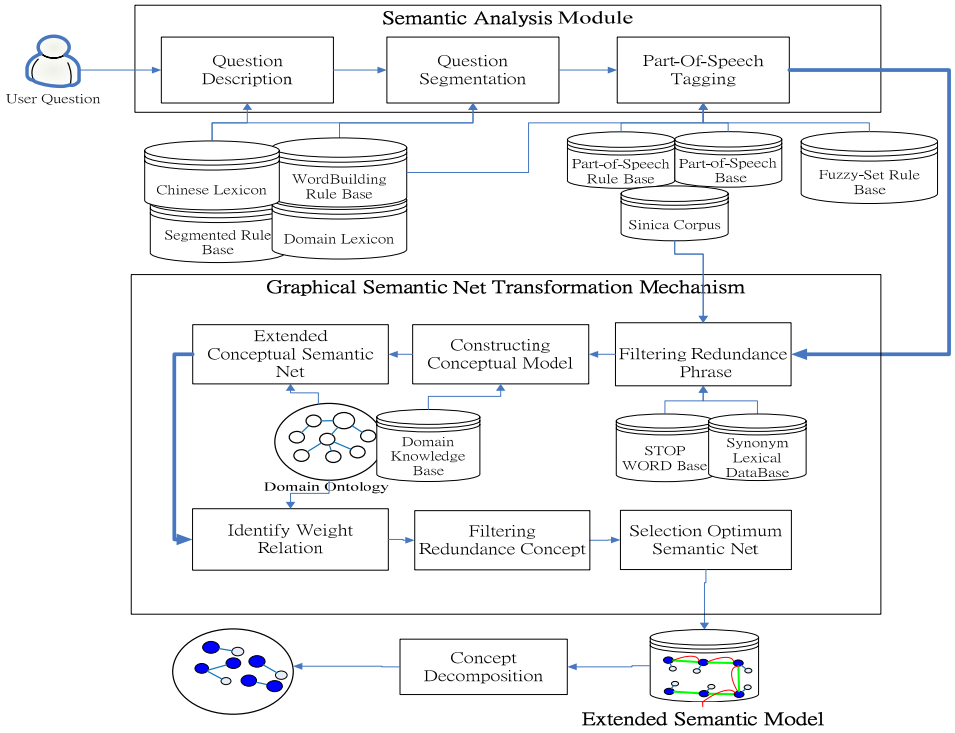

Fig. 2 Graphical Semantic Net Transformation Mechanism

Semantic Analysis Module

Using multiple kinds of databases to help in semantic analysis and semantic transformation includes problem segmentation, part-of-speech tagging, semantic extension, filtering, construction of concept models and concept decomposition; the descriptions of the primary modules are as follows:

(1) Question Description Module: provides a user interface for the user to perform problem description; sends the problem to the Question Segmentation Module.

(2) Question Segmentation Module: Primary functions include question segmentation, word merging, defining of candidate phrases, phrase selection, and phrase merging. They are described separately

below:

- Question Segmentation: based on problems described by the user, using the Chinese lexicon and segmentation rules, as well as the rule of prioritizing for the words and phrases appearing with greatest frequency, the question is deconstructed into individual words.
- Word Merging: using determinative measure compounds and reduplications in the word building rule base to perform comparison with the Chinese lexicon, smaller words and phrases are combined into (longer) phrases.
- Identifying Candidate Phrases: Using phrase construction rules and all possible phrases listed by the Chinese lexicon, phrases resulting from word merging become candidate phrases.
- Phrase Selection: Using the priorities of longer phrases, small standard deviation, minimal attached language elements, smallest word count in compound words, greatest frequency of one-word phrases, greatest total frequency of phrases, candidate phrases are selected.
- Phrase Merging: Selected phrases are (when possible) combined into longer phrases using phrase construction rules.

(3) Part-of-Speech Tagging Module: Phrases resulting from merging are used to construct Markov language models (using the Sinica Corpus) [2]; a part-of-speech base and part-of-speech rule base are used in calculating the frequency of appearance by phrase linkages.

Semantic Net Transformation Mechanism

The function of this mechanism is to perform filtering for useless phrases resulting from semantic analysis model tagging, construction of semantic net models, defining the weight of phrases, extension of concept semantic nets, and selection of optimum semantic nets; the purpose of such is to transform semantics into graphics. The explanations are as follows:

(1) Redundant Phrase Filtering: this method uses a synonym lexicon base and a stop word base to help filtering, using a part-of-speech base and part-of-speech rule base to filter non-nouns and non-verbs; this action can filter out the majority of meaningless phrases to increase system efficiency.
(2) Conceptual Model Construction: the use of formalization concept calculation means using statistical methods to perform analysis on information quantity, subsequently discovering concept structure from information combination; graphical visualization is produced [8]. These relationships and domain ontology help in the construction of concept models.
(3) Extension of Conceptual Semantic Net: using domain ontology and a knowledge base to extend related concepts, a semantic net is constructed from the various constructed concepts.
(4) Defining Weight Relations: using normalizing concept analysis theory and calculating the relationship between each extension and problem concept, extended semantic nets are formed from semantic nets.
(5) Filtering Redundant Concepts: Using a knowledge base and concepts to perform TF-IDF calculation, a parameter benchmark is formed for identified concepts; those with parameters lower than the benchmark are filtered out.
(6) Selection of Optimum Semantic Net: Defined concepts are given weights for arrangement and selection; those with higher weights are selected.

Concept Decomposition Mechanism

Selected semantic nets undergo concept decomposition, and then are combined with the various identified concept. After being combined, weight calculation for concepts and the knowledge base are performed using the TF-IDF method. Those with high weight are made keywords; those with low weight must be recombined with other concepts, until each concept is finished being combined. This process allows for search processing by the integrated mechanism of multi-search engines.

Integrated Mechanism of Multi-Search Engines

The framework of this subsystem is as shown in Figure 3. Using the results from the sub-framework described in the above section, this mechanism performs searches for internet information. This phase combines exploration of the Web, search engine, as well as filtering and arrangement techniques to find web pages suitable for the user's question [9]; the URLs are stored in the Web Pages Base. The primary activities are as described below:

(1) Concept Transformation: using the graphical semantic transformation mechanism, the semantic nets are entered into the concept decomposition mechanism; the primary function is to perform division of the complete graphical semantic net, so that it forms multiple meaningful sub-concepts. This task is performed to avoid the inability to find results for the entire concept on the internet. As a result, the task of searching includes two separate parts: first to transfer the entire graphical semantic net and the complete concept to the four search engines chosen by this study (AltaVista, GAIS, Google, Yahoo) to perform web searches; second is to perform searches for the "graphical semantic fragments" resulting from the concept decomposition mechanism, and then to convert the concepts to be searched for to sentences.
(2) Keyword extraction: extraction of keywords from concept models or sub-concept models.
(3) Specialized keyword composition: custom searches made for keywords based on the special characteristics of each search engine.
(4) URL searching: each search engine performs searches; search results are entered into the URL base. Stored fields include: search engine name, web address, keywords, title, original page size, pure text size, and summary.
(5) URL format standardizing: the format of most information on the internet is presented in Hypertext Markup Language (HTML) and Portable Document Format (PDF), which are either non-structured or semi-structured; XML can be used to attain the goal of structuring. XML is a language for data description, primarily used in designing web pages for information capable of being structured; it allows users to freely define labels related to their documents, while at the same time using custom labels, properties, XML schemas, and Document Type Definitions (DTD) [12] to define URLs into needed formats. This is referred to as URL Format Standardizing, after which the results are stored in the standardizing URL base.
(6) URL filtering and ranking: format-standardized URLs undergo comparison, primarily using the knowledge offered by graphical semantic nets and domain knowledge as a basis; those unsuitable for use are automatically removed. Those initial URLs remaining are collected and then even more carefully filtered using occurrence hit algorithms and filter hyperlink algorithms. The development of these two algorithms is the mission for the next phase. The concept is: the former calculates occurrence and hit values to remove multiple appearances of the same URL, while the latter uses calculation of occurrence and hit parameters to remove ads and gain the needed URLs. Afterwards, the importances of URL summaries are given appropriate weight values; ranking is then performed.
(7) Web page collection: collection of actual web page content after standardizing, filtering, etc.
(8) Web page format standardizing: formats are converted to the XML standard needed for this study, and are finally stored in the web pages base for knowledge extraction performed by the knowledge extraction mechanism.

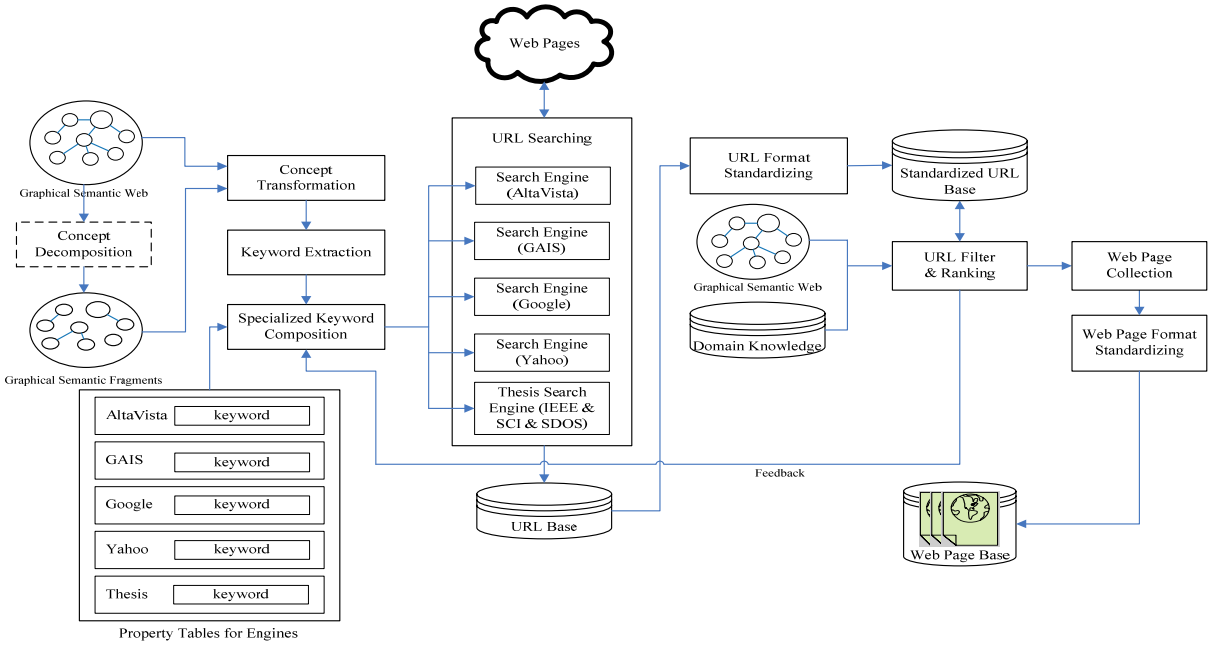

Fig. 3 Integrated Multi-Search Engines Sub-system

Knowledge Extraction and Construction Mechanism

With regard to one of the core technologies of this study – "knowledge extraction and construction module" – a functional framework for normalized web knowledge extraction mechanism and construction subsystems is as shown in Figure 4. The core ability of this sub-framework is to extract user-needed knowledge based on domain knowledge. The detailed functions of the three primary mechanisms included in this sub-framework are described in detail in the following subsections.

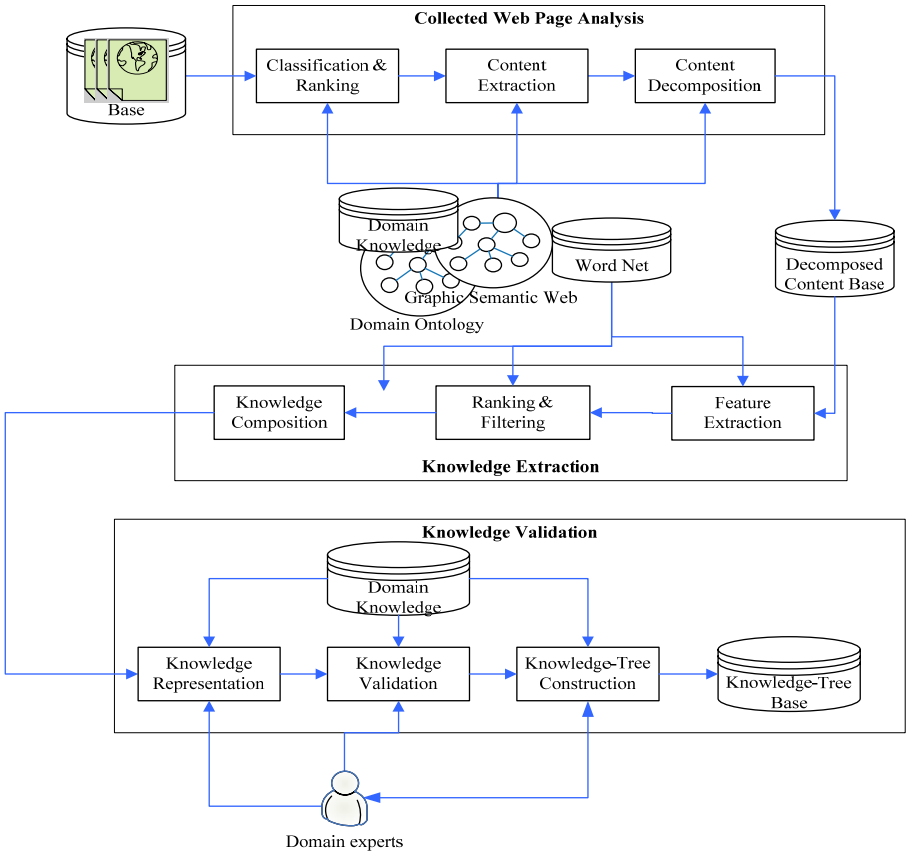

Fig. 4 Knowledge Extraction and Construction Sub-system

Web Page Analysis Mechanism

This mechanism uses the results from the integrated multi-search engine mechanism (referred to in Section 4) to perform the task of initial analysis on web page contents. There are three primary activities in this phase, as described below:

(1) Classification and ranking: standardized web pages saved in XML form in the web page base are first classified, and then arranged with web pages of the same category.
(2) Content extraction: with regard to documents that have been standardized and stored as XML, the XML format is composed of many different labels; each label has a specific meaning in XML, and can mark which search engine a given page came from, document size, document summary, etc. Using the custom label formats in the search engines, the main bodies of the web documents are extracted.
(3) Content decomposition: web documents are composed of individual paragraphs, and each paragraph is in turn composed of individual sentences. In this step, the main bodies of the web documents must first be extracted using custom labels; content parts are deconstructed. Based on question marks, periods …etc, the content is deconstructed into separate paragraphs and sentences. Then, based on the two methods of document frequency and information gain, the importance of keywords in each document and sentence is calculated; this facilitates the "knowledge extraction mechanism" of the next step.

Knowledge Extraction Mechanism

Most web documents are quite long, but users generally seek information quickly; as a result, this study's "knowledge extraction mechanism" concentrates overly long documents into simple summaries,

saving the time needed for users to browse through information; quickly presenting concentrated knowledge to users is the purpose of this mechanism. Web document contents are deconstructed into paragraphs using the process described in section 5.1; analyzing which paragraphs are directly related to the information the user seeks is the primary goal of the knowledge extraction mechanism. In this mechanism, the segmentation of words and phrases in basic document analysis is the basic function of this mechanism; these functions have already been given a basic introduction before in this paper, so this section focuses on the explanation of primary functions below.

(1) Feature extraction: the purpose of feature extraction lies in reducing the amount of information. Unimportant phrases are removed from feature space; the number of features is reduced in this manner. After web page contents are decomposed into paragraphs, they are further decomposed into sentences using commas, periods, question marks, etc; the results are then saved into the decomposed content base. The weight of keywords in the sentences is calculated; those with particularly high weight are chosen, and then undergo ranking and filtering.
(2) Ranking and filtering: after a benchmark value is given, those sentences with weight too low are removed; domain phrase construction rules and domain knowledge are combined to rank sentences according to their weight.
(3) Knowledge construction: after using domain phrase construction rules, the knowledge of the entire web document can be produced and compiled. Those sentences with high weight value are ranked; using this information along with the domain phrase base phrase segmentation and Global Bushy Path (GBP) [13] to produce a dynamic summary for each document.

Knowledge Validation

After the knowledge contained in a web page undergoes a string of extraction processes, forms summaries, and is combined into knowledge, knowledge presentation and testing must be performed. The framework of knowledge testing subsystems is as shown in the lower half of Figure 1; such includes a number of primary elements, including: exhibition of knowledge, confirmation of knowledge, and construction of a knowledge tree; a detailed explanation is as follows:

(1) Knowledge Representation: after the knowledge of the web pages undergoes the core processes previously described in section 5.2, the knowledge hidden in the web pages can be extracted; after it has been formed into summaries and combined into knowledge; domain knowledge is applied to remove less meaningful or relatively unrelated words and phrases, allowing concise knowledge summaries to be presented to the user.
(2) Knowledge Validation: besides extracting and summarizing the knowledge contained in the web pages, the accuracy of the knowledge must be guaranteed; as such, domain experts perform knowledge testing on the summaries of each document. Those documents that are ruled accurate by the experts are outputted, forming the primary input for construction of knowledge trees.
(3) Knowledge Tree Construction: each document summary forms pitch points; the file headers of the pitch points head the titles of the web pages, while the headers in the back are web document summaries. Calculate the similarity between each new pitch point and root pitch point, and then decide the order of visitation on the knowledge tree. With each time a user-entered key word/phrase passes through the web integrated multi-search engine mechanism and web knowledge extraction mechanism, the number of pitch points increases. At this time, code the pitch point structure of the knowledge tree into a Huffman Tree, and then store it into the knowledge tree database. At this point, coding the Huffman Tree is for the purpose of facilitating clear understanding the numbering

of the summaries after pitch points increase in number. In the future, if users are not satisfied with summaries resulting from document extraction, the Huffman Tree form can be used to calculate the similarity between pitch points at hand and the pitch points of the knowledge tree database; this allows for the user to compare the summaries in the pitch points at hand with the summaries of similar past documents. If similar documents still fail to satisfy the user, then the Huffman Tree can be further used to find a second similar document. Proceeding further in this manner, it is hoped that the user's needs can be satisfied.

Discussions and Conclusions

This study proposes an intelligent web knowledge extraction and construction mechanism framework design. The first portion performs phrase segmentation and marking according to the user's question and based on a number of rule bases and databases. It then uses the concept of standardization to analyze and construct semantic nets; ontology then forms conceptual level relationships and extension. Because of the overabundance of extended concepts, weight relationships between concepts are used to establish a benchmark value to filter redundant or useless concepts and produce a graphical semantic net. The second portion, using an integrated multi-search engine mechanism, analyzes, evaluates, filters, and ranks web pages to find URLs and web pages that fit user problems, and then stores them in a web page base. Because the summaries are found online, they are generally presented surrounding keywords, perhaps the first ten sentences of a document. Situations like these prevent one from finding the needed knowledge hidden in web pages; whether or not net structure graphs can replace keyword-based searches is a concept that has not yet been evaluated. Format standardization may aid in inserting more meaningful content and headings, helping in knowledge extraction. The third portion extracts hidden knowledge and then saves it into a knowledge base following expert evaluation. Unfortunately, in the past, sentences are often awkward or the meaning of the document is twisted and misunderstood; such leads to unsatisfactory document summaries. How to use domain knowledge and domain ontology to smooth out sentences and fit the original meaning is an interesting point for future study.

The contributions of this study are as follows:

(1) Allows the user to more easily understand whether or not a semantic description is correct.
(2) Graphical semantic nets can be applied to other related domains, such as knowledge maps, discussion areas, etc. Plentiful knowledge can be dug out, helping us to understand the contents.
(3) Reduced search time for users, redundancy in web pages, and raised accuracy and effectiveness.
(4) Provides web page knowledge closer to the original meaning for user reference.
(5) Information documents use XML technology to attain the goals of consensus and structuring, aiding in the later additions, revisions, or editing. Bandwidth problems do not cause damage or loss.

Acknowledgement

This research is financially supported by National Science Council of the Republic of China under Contract Nos: NSC94-2524-S-024-002, NSC94-2524-S-006-005 and NSC94-2524-S-006-006.

References

[1] Hui-Chuan Chu, Hon-Yan Lu, Yuh-Min Chen, Chia-Jou Lin, Chih-Ming Lin, Problem-Based

e-Learning to Support Mathematics Teaching for Students with Mild Disabilities: Model and System Framework, World Conference on Educational Multimedia, HyperMedia and Telcommunication, 2006.

[2] Solomon, J. Social influences on the construction of pupils' understanding of science. Studies in Science Education, p. 63-82, 1987.

[3] CKIP (Chinese Knowledge Information Processing), http://godel.iis.sinica.edu.tw/.

[4] The AltaVista Search Engine, http://www.altavista.com/.

[5] The GAIS Search Engine, http://gais.cs.ccu.edu.tw/.

[6] The Google Search Engine, http://www.google.com/.

[7] The Yahoo Search Engine, http://search.yahoo.com/.

[8] Ganter B., and Wille R., "Formal Concept Analysis: Mathematical Foundations," 1999.

[9] Web Mining Books - Morgan Kaufmann, "Mining The Web-Discovering Knowledge From Hypertext Data," 2003.

[10] Tatsunori Mori., "Information Gain Ratio as Term Weight-The case of Summarization of IR Results," In Proceedings of the 19th International Conference on Computational Linguistics, p. 688-694, 2002.

[11] Tatsunori Mori., Miwa Kikuchi., and Kazufumi, Yoshida Term "Weighting Method based on Information Gain Ratio for Summarizing Documents retrieved by IR systems," In Proceedings of NTCIR Workshop 2 Meeting, 2001.

[12] Norman Walsh, "A Technical Introduction to XML," World Wide Web Journal (http://www.nwalsh.com/docs/articles/xml/), 1998.

[13] Salton G., Singhal A., Mitra M., and Buckley C., "Automatic text structuring and summarization information processing & management," p. 193-207, 1997.

附件 2

利用 5W1H 結合本體論做網路資料探勘

緒論

近年來資訊技術蓬勃發展，由於網際網路盛行拉近了人與人之間的距離，所有的訊息都可以傳遞到世界上其它任何角落，也由於資訊數位化的因素，造成大量的資訊充斥在隨手可得的網路世界中，許多電子文件的服務也與日俱增，要如何尋找、收集資料，然後整理、探勘為有用的資訊便顯得一門重要的學問，要有效的管理這些資料也因此顯得格外重要。

全球資訊網(World Wide Web, WWW)目前儼然成為網際網路資訊的重要來源。它所提供的資訊包羅萬象，資訊量增加的速度也越來越快。資訊公開普及化有正面的意義，但數量過多且來源分散的資訊，卻未必是好事。缺乏一致的管理，特定主題的相關網頁，散佈在各處，不知道有多少，也不知道如何去尋找相關資訊；此外，人類記憶與處理能力上亦有限制，不能無限量的瀏覽或儲存、分析資訊。因此急需一個搜尋引擎工具，以協助使用者瀏覽資訊，避免使用者迷失在網際網路的空間中。

在現今資訊爆炸的時代，每天都有新的資訊產生。為了從這些大量的資訊中，準確的獲取有用的資訊，文章的自動摘要處理變的越來越重要。通過閱讀文章摘要而不是全文能極大的加速資訊過濾速度，幫助人們了解概況或確定是否應該詳讀原文。這一技術是快速準確獲取資訊的一個有用工具，在現代人們求於快速簡潔的獲取知識，它的市場需求相當廣泛。

研究目的

在本篇論文，我們將藉由分析使用者所輸入的中文問句，以自然語言方式，為本研究做中文資訊擷取的方法與步驟，建立具快速分析、可攜性佳、準確度高等特性之中文資訊擷取系統。在短期方面，我們將觀察各資料領域的特性來建立跨領域之中文資訊擷取系統；在長期方面，我們也希望將擷取出來的資訊做進一步加值的應用。本篇論文之研究目的有下列幾項：

一、分析文字文件中中文語句的結構、順序及組合方式，研究中文資訊擷取技術。

二、整理相關文獻，探討資訊擷取技術之研究進展，並做分析比較。

三、建立高準確度之中文資料擷取系統來快速分析、擷取中文文件。

四、觀察不同資料領域之特性，研究跨領域資訊擷取之可行性，發展高可攜性之中文資訊擷取系統。

五、分析此中文資訊擷取系統之實驗結果，並做分析與討論。

研究預期貢獻

本研究之具體貢獻如下：

一、可讓使用者更容易了解其所描述之問句的真正涵意。

二、圖形化語意網可應用於其他相關之領域，例如知識地圖、討論區等，能挖掘出豐沛的知識，幫助我們容易了解內容。

三、本研究歸納出 5W1H 之問句類型，能更快速及提高準確性。
四、透過語意網的擴展，能讓使用者找到更多所需知識。
五、轉換後的字串，可大大提高搜尋的準確率。
六、有限自動機(Finite Automata)的建立，可改善比對的效率及準確率。
七、在5W1H對應庫中，意圖類型的對應字眾多，可透過權重庫來設定權重值，可經過使用者的每一筆處理過後的資料來做訓練，形成一回饋機制，希望能在未來做深入探討，達到本研究最終目的。

1.圖形化語意網轉換機制

本研究是依使用者問題為導向之架構，如圖 2 所示，建構設計分為三大區塊，而本研究以第一區塊 **Q&A Mechanism** 為重點，其包括語意分析機制及語意轉換機制及語意網轉換機制三部分，在這部分可幫助我們產出意圖及圖形化語意網(Graphical Semantic Net)。第二部分為網頁整合型搜尋引擎機制(Integrated Mechanism of Search Engines on Web)，預計將產出將以 XML 型態呈現 URL 與 web pages。而在第三部分為知識萃取機制(Knowledge Extraction Mechanism)，以網頁特徵進行分類及排序之萃取，經專家驗證確認後，儲存於知識庫後，回覆給使用者。

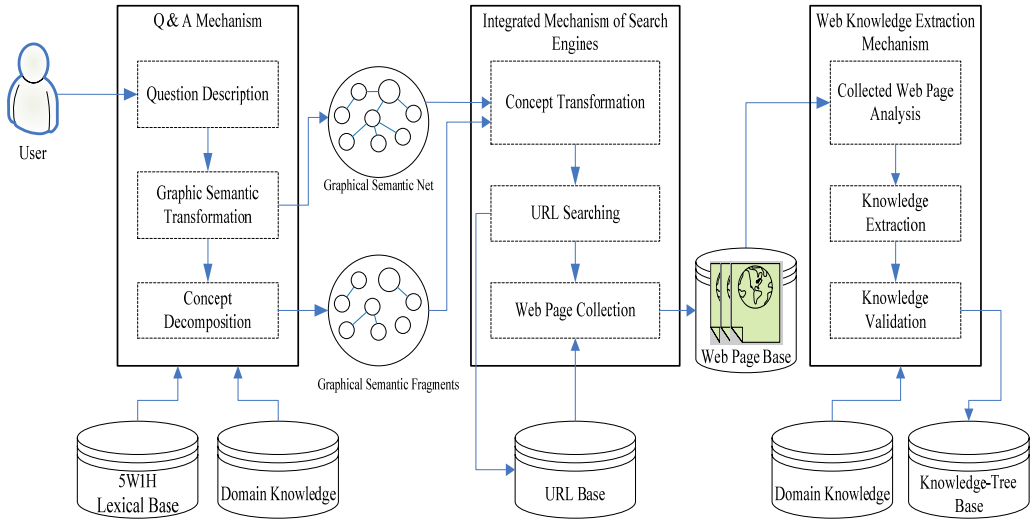

圖 1：系統架構圖
資料來源：本研究

本研究架構如圖 1 所示，主要包括語意分析機制(Semantic Analysis Mechanism)、語意網轉換機制(Semantic Net Transformation Mechanism)及意圖轉換機制(Intention Transformation Mechanism)，三機制主要可幫助我們產出問題意圖(Intention)、關鍵字、圖形化語意網(Graphical Semantic Net)及問題轉換。

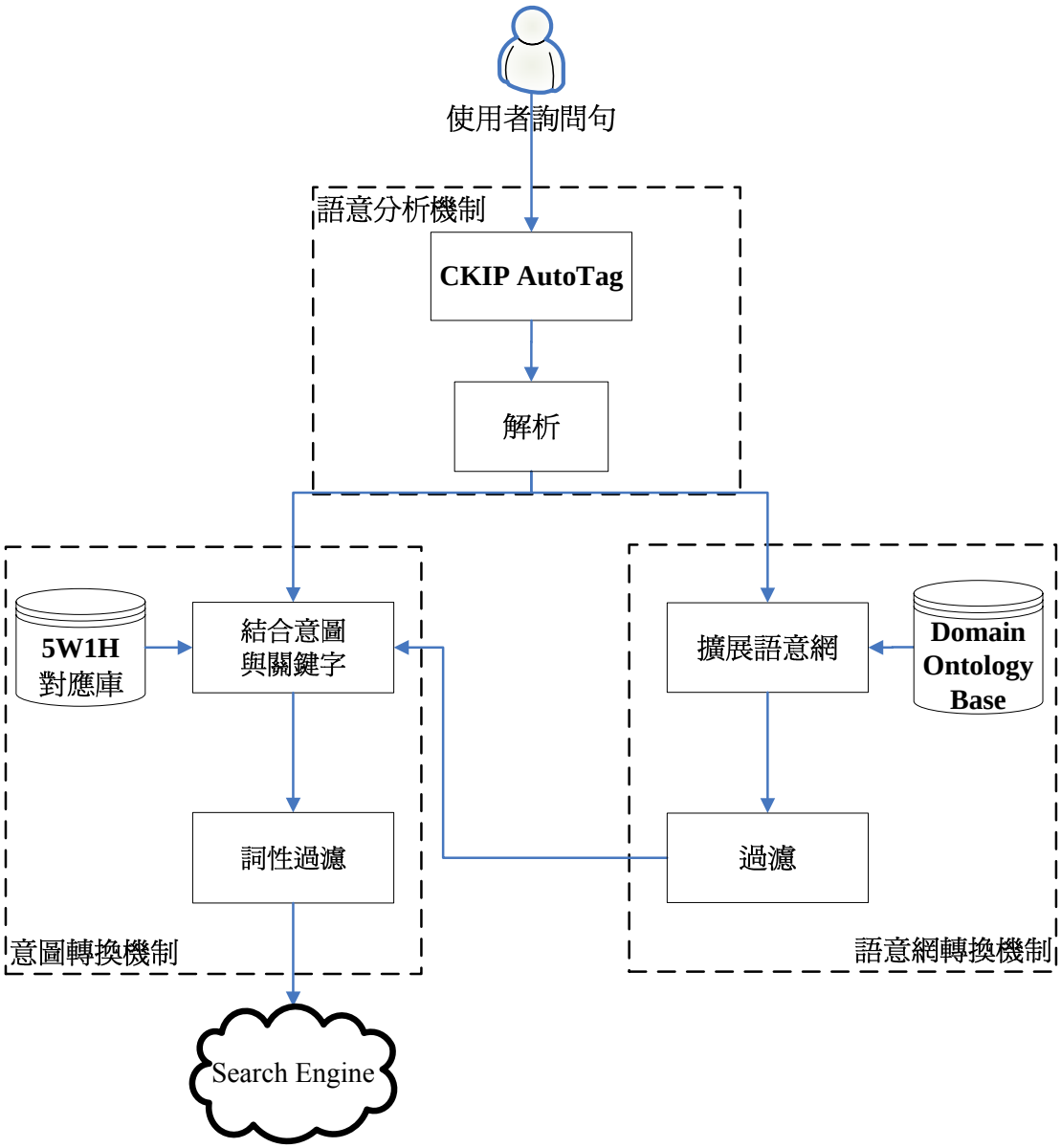

圖 2：圖形化語意網轉換機制架構圖

資料來源：本研究

本研究提出一以 5W1H 結合領域本體論做資料探勘，以圖 2 來表示，透過意圖及語意網的結合，提高搜尋的準確性，以解決使用者之問題。本機制除了提供使用者一個圖形化的使用者介面及供使用者輸入問題及查閱搜尋結果的回應外，它是以語意分析機制(Semantic Analysis Mechanism)、語意網轉換機制(Semantic Net Transformation Mechanism)及意圖轉換機制(Intention Transformation Mechanism)三部分為核心。語意分析機制中將使用者描述的問題，利用自然語言處理與斷詞(Natural Language Processing and Segmented)及詞性標記技術處理，以前處理(Pre-Processing)後再透過解析將意圖及關鍵字萃取出來。語意網轉換機制主要是將拆解後關鍵字利用 Domain Ontology 來進行擴展成語意網。意圖轉換機制主要將意圖與關鍵字結合後，於資料庫中找出所對應資料，經轉換後，於網路搜尋引擎上做查詢。

1.1 語意分析機制(Semantic Analysis Mechanism)

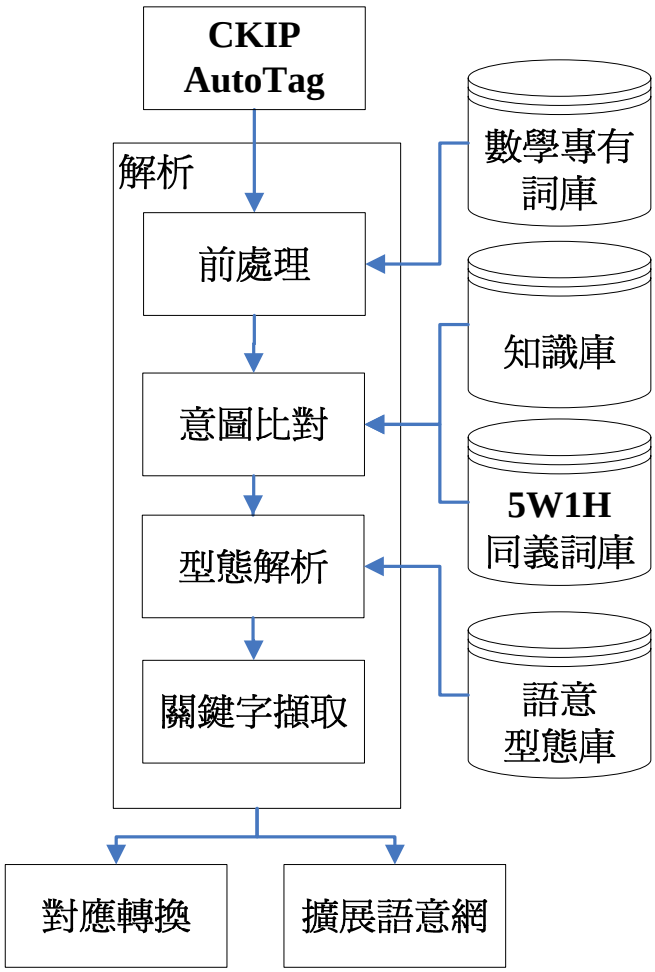

圖3：語意分析機制
資料來源：本研究

本機制如圖 3 所示，主要包含五步驟，將使用者的問句利用中研院 CKIP 做自動斷詞及標記詞性、意圖類型比對及關鍵字擷取，其步驟主要功能如下：

（1）CKIP AutoTag

主要將使用者的問題斷詞及詞性標記，把我們需要的關鍵字標記出來，我們將以一例子「如何學習除法」來說明， 以 CKIP 斷詞及標記[4]後分為三部分「如何(D)學習(VC) 除法(Na)」。以下圖步驟表示：

（2）前處理(Pre-Processing)

將上步驟如未能斷詞完成的詞做結合，如「四則運算」等專有名詞，會斷成「四(Neu)則(Nf)運算(VC)」三部分，透過此步驟與數學專有詞庫對比對，可將「四則運算」合併為單一的專有名詞「Na」、「Nb」「……」等詞彙。

（3）意圖比對(Intention Matching)

將問句字串配合 5W1H 同義詞庫比對出所屬問句的意圖類型，其詞庫如表 1 所

示。

表 1：5W1H 同義詞資料表

what	how	why	who	when	where
什麼	如何是	是為什麼	誰	是何時	在哪
何謂	是怎樣	怎麼會這樣	何人	在什麼時候	在何地
什麼是	是什麼樣	是何故	何等人	在什麼時間	在哪裡
什麼為	是怎麼樣	為何會	什麼人	於幾時	什麼地方
什麼叫做	是何事	為什麼會	是誰	在什麼階段	什麼地點
是什麼	為何事	為什麼是	為誰	在何時	在何處
是叫做什麼	是如何	是為何	誰是	在幾時	在哪邊
何謂為	為如何	怎麼會	是何人	於何時	在何方
何謂是	為怎樣	為什麼要	何人是	何時是	哪裡是
			是什麼人	何時為	是哪裡
			誰會	幾點是	是何方
			有誰	是幾點	是何處
			誰要	是什麼時候	是什麼地方
			是有誰	是什麼時間	是什麼地點
				是幾時	是哪裡
				是何時	是在哪

資料來源：本研究整理

（4）型態解析(Parsing)

主要目的來驗證型態，以建立有限自動機(Finite Automata)方式，與語意型態庫做快速掃瞄出屬何種類型，可驗證與先前之意圖屬同一類型，如圖 4 所示。

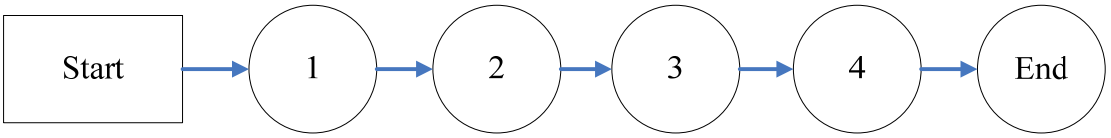

圖4：有限自動機流程圖

（5）關鍵字擷取(Keyword Extract)

經上步驟後，問句字串中包含名詞「N」的字串擷取出來，如「Na」、「Nb」、「VA」、「VB」「…」等來作為關鍵字，以提供下一步驟做擴展語意網。

1.2語意網轉換機制(Semantic Net Transformation Mechanism)

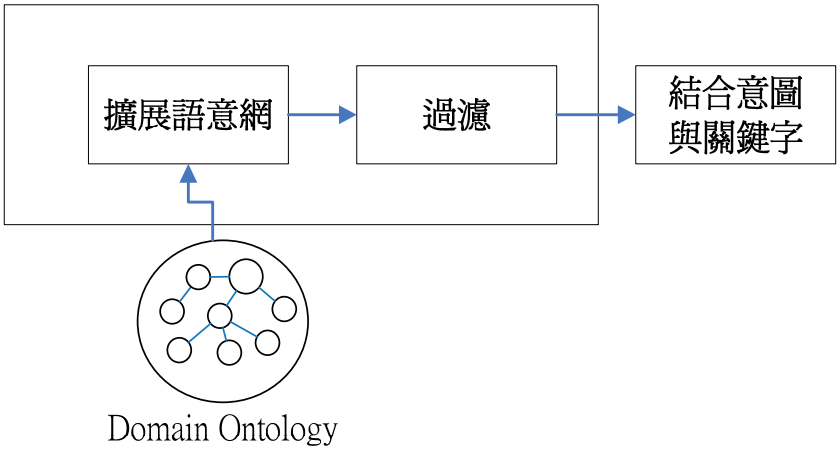

圖5：語意網轉換機制

資料來源：本研究

本機制包括三步驟，如圖 5 所示，將包含名詞「Na」或「VA」等的關鍵字，透過 Ontology 技術擴展成語意網，目的為將語意轉換成圖形化。其說明如下：

（1）擴展語意網(Extended Semantic Net)

被延伸後的關鍵字，利用領域本體論(Domain Ontology)去延伸相關概念出來，形成語意網。以下圖 6 為例：

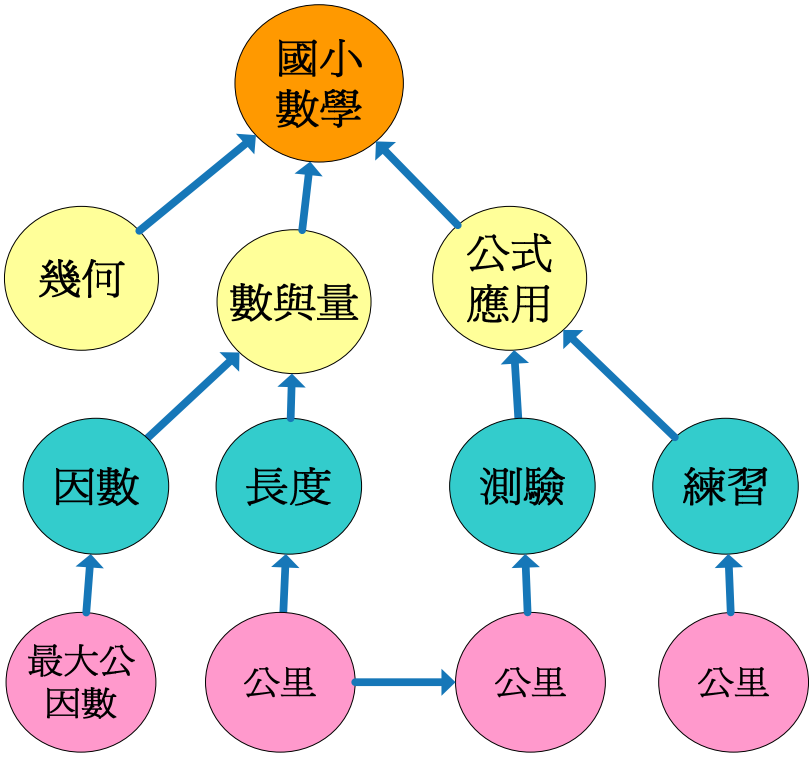

圖6：國小數學領域之ontology架構

資料來源：本研究整理

（2）過濾(Filtering)

根據本體論技術，在系統上設立一參數門檻值，可依使用者所需作增加或減少。以下圖 7 為例：

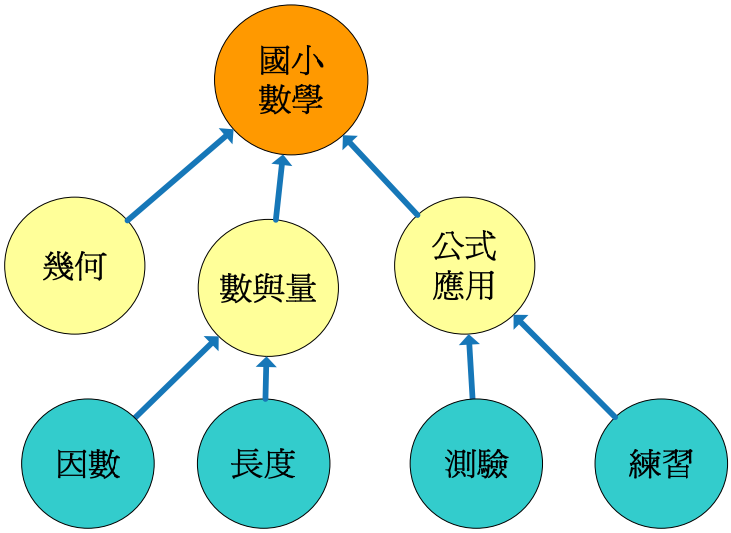

圖7：國小數學領域之ontology架構

資料來源：本研究整理

1.3 意圖轉換機制(Intention Transformation Mechanism)

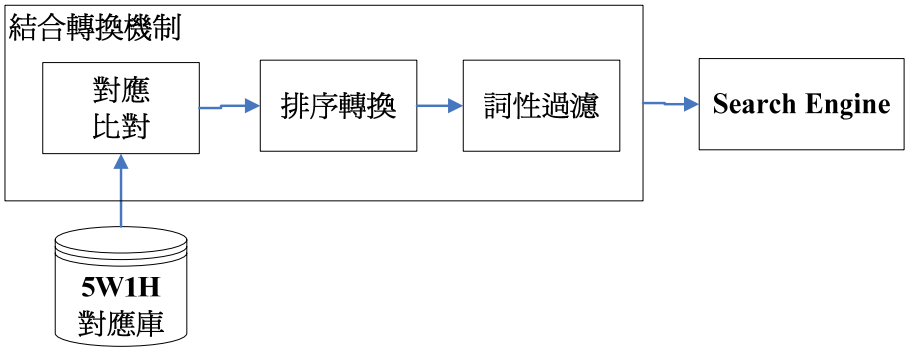

圖8：意圖轉換機制

資料來源：本研究整理

本機制目的是將意圖與關鍵字結合後，以 5W1H 對應庫的「動名詞」對應出來，再轉換成一般句型及過濾詞性後於搜尋引擎做資料搜尋。

（1）對應比對(Corresponsively)

透過5W1H對應庫如表2的比對，將所對應的動名詞找出來。

表 2：5W1H 對應資料表

意圖類型	類型	Domian Keyword	對應**名詞**	對應**動詞**
why	數學類	數學知識	來源、源由、理由、原因、起源	發生、發現
		等腰三角形面積		
		四則運算		
		數學的教學模式		
		平行四邊形		
		智能障礙		

when	數學類	數學知識	日期、起源、時間	發生、發現、發源、發跡
		等腰三角形面積		
		四則運算		
		數學的教學模式		
		平行四邊形		
		智能障礙		
what	數學類	數學知識		
		等腰三角形面積		
		四則運算		
		數學的教學模式		
		平行四邊形		
		智能障礙		
how	數學類	數學知識	解法、答案、公式、算法、教法、方法、步驟、題庫、題材、類型、題目、方式、用途、公式	教、算、輔導、輔助、運算、計算、練習、求得、求、解、教導、解釋
		等腰三角形面積		
		四則運算		
		數學的教學模式		
		平行四邊形		
		乘法運算		
		智能障礙(Na)	情況、狀況、問題	認識、學習
where	數學類	數學知識	發源地、原點、原處、發生地、原生地、出處	
		等腰三角形面積		
		四則運算		
		數學的教學模式		
		平行四邊形		
		智能障礙		
who	數學類	數學知識	發明人、原著、作者、發明者	
		等腰三角形面積		
		四則運算		

		數學的教學模式		
		平行四邊形		
		智能障礙		

資料來源：本研究整理

（2）5W1H對應字擷取

主要目的是以TF公式1計算對應字與關鍵字在文件中共同出現的頻率，以此方式做推論假設兩者之關係重要程度。

TF(Term Frequency)：TF模式認為在某一篇文章當中出現的次數若較多，則代表此字詞對於這篇文章而言，有一定的重要度。因此次數頻率就是直接把字詞出現的次數，做為此字對於特定物件的重要程序評比。

公式1：

則此n個字詞對於此文章的重要程度可表示如下：

$$tf_{ij} = \frac{n_j}{n_{all}}$$

關鍵詞 j 代表在文件i出現的頻率，其中

n_j：j 文件i 出現的次數

n_{all}:表示文件 i 所有具意義的總字數

（3）排序轉換(Ranking Transformation)

經上一步驟後，從資料表中對應出來的「動名詞」配合關鍵字來做排序轉換，本研究經過文件訓練後「四則運算」＋「方法」＝「名詞」＋「對應字」，透過此方式可做直接結合轉換。

以下圖9例子表示：

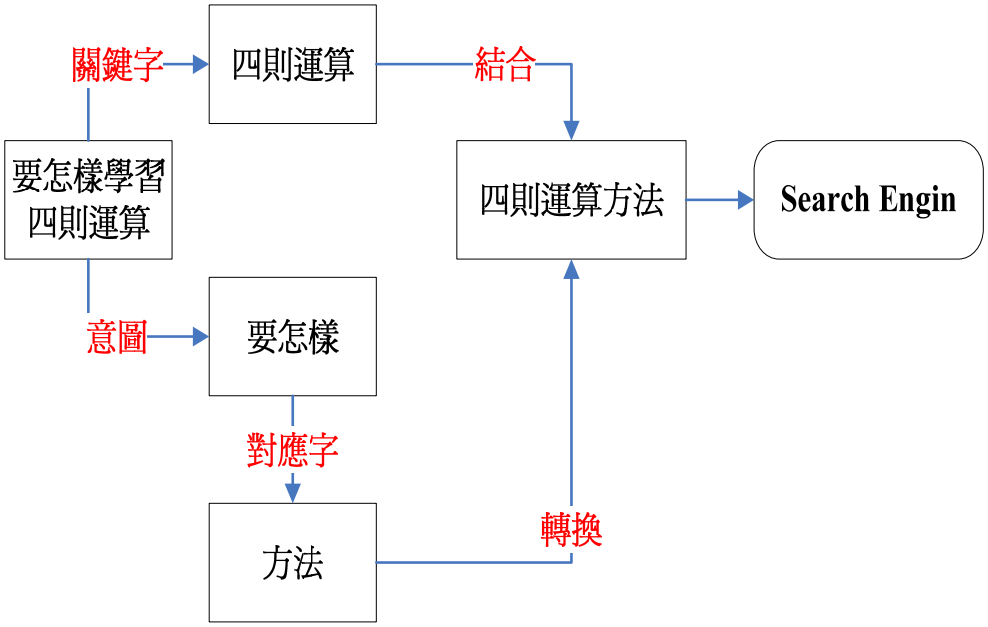

圖9：轉換示意圖
資料來源：本研究

（4）詞性過濾(Filtering)

將轉換過後的字串做詞性過濾，讓下一步驟整合型網頁搜尋引擎機制做搜尋處理，以解決使用者之問題。

結論與未來展望

本研究主要是在探討：自然語言的處理技術、意圖與關鍵字的擷取、擴展語意網、型態解析及問題轉換。而本研究主要針對幾項問題來進行分類，從使用者問句中去擷取出意圖類型及關鍵字來進行語意擴展及結合轉換，最後透過搜尋引擎來找到更好的資料回覆給使用者，達到本研究之目的。

本研究之具體貢獻如下：

(1)可讓使用者更容易了解其所描述之問句的真正涵意。

(2)圖形化語意網可應用於其他相關之領域，例如知識地圖、討論區等，能挖掘出豐沛的知識，幫助我們容易了解內容。

(3)本研究歸納出 5W1H 之問句類型，能更快速及提高準確性。

(4)透過語意網的擴展，能讓使用者找到更多所需知識。

(5)轉換後的字串，可大大提高搜尋的準確率。

(6)有限自動機(Finite Automata)的建立，可改善比對的效率及準確率。

(7)在5W1H對應庫中，意圖類型的對應字眾多，可透過權重庫來設定權重值，可經過使用者的每一筆處理過後的資料來做訓練，形成一回饋機制，希望能在未來做深入探討，達到本研究最終目的。

附件3

應用領域本體論設計整合網路上搜尋引擎機制

摘要

近來由於產業及科技的競爭，以致於相關知識的蒐集、獲取、整合、儲存、管理、分享與運用之重要性相對提升。隨著網際網路發展，如何以自動化的方式有效獲取網路上的資訊提供使用者所需的知識是一項很大的挑戰。

本研究結合利用資料探勘發掘網頁內容知識並檢視其相似性且導入領域實體概念，發展強化搜尋引擎的過濾及排序機制，透過演算法去除格式不完整、有重覆性網站且針對搜尋後的摘要及標題進行資訊含量之運算，其值若介於本研究所設立之可接受範圍，便進一步計算摘要權重值；若遇到描述不同但意思相仿的摘要，會應用領域實體所建立的法則計算詞彙相似程度，其後給予適當權重值，系統則將前述每篇摘要之權重排列順序並檢視符合原意與否，再取回其網頁內容，經由擷取就變成可利用知識，此知識可提供給使用者解決問題之參考，本研究著重於國小數學學習領域方面。希冀能節省使用者自行過濾檢索時間與減少頻寬資訊量。

關鍵詞：搜尋引擎、網頁內容探勘、資訊檢索、實體論

緒論

隨著資訊科技日新月異，網路資源越來越豐沛且複雜，上網檢索資料變成解決問題的方法之一。目前各家搜尋引擎功能相當強大，導致檢索的知識重複性太高且產生不符合所需，浪費頻寬效能降低，因此搜尋引擎出現三種問題[7]：

一、網路資訊成長迅速，單一搜尋引擎難以處理龐大資訊。

二、多具引擎搜尋結果其資訊量重複性太高，使用者需花費很多的時間與精神，從中檢索所需資訊。

三、無意義廣告伴隨著檢索而呈現。

所以相關知識的蒐集、獲取、整合、儲存、管理、分享與運用之重要性相對提升。如何將上述問題轉成一個圖形化語意網路模型，此模型可描述問題的概念，搭配 5W1H 規則及原意產生關鍵字，其後至網路上檢索，最後藉由本研究設計的強化搜尋引擎過濾與排序機制，檢索貼切原意的摘要及網頁，經由擷取找出真正的知識，提供給使用者參考。

本研究流程架構分為三個部分：前處理機制(concept transformation、keyword extraction、specialized keyword composition)、整合式搜尋機制(integration searching)與後處理機制(format standardizing、filter & ranking、web page collection)，如圖1所示。

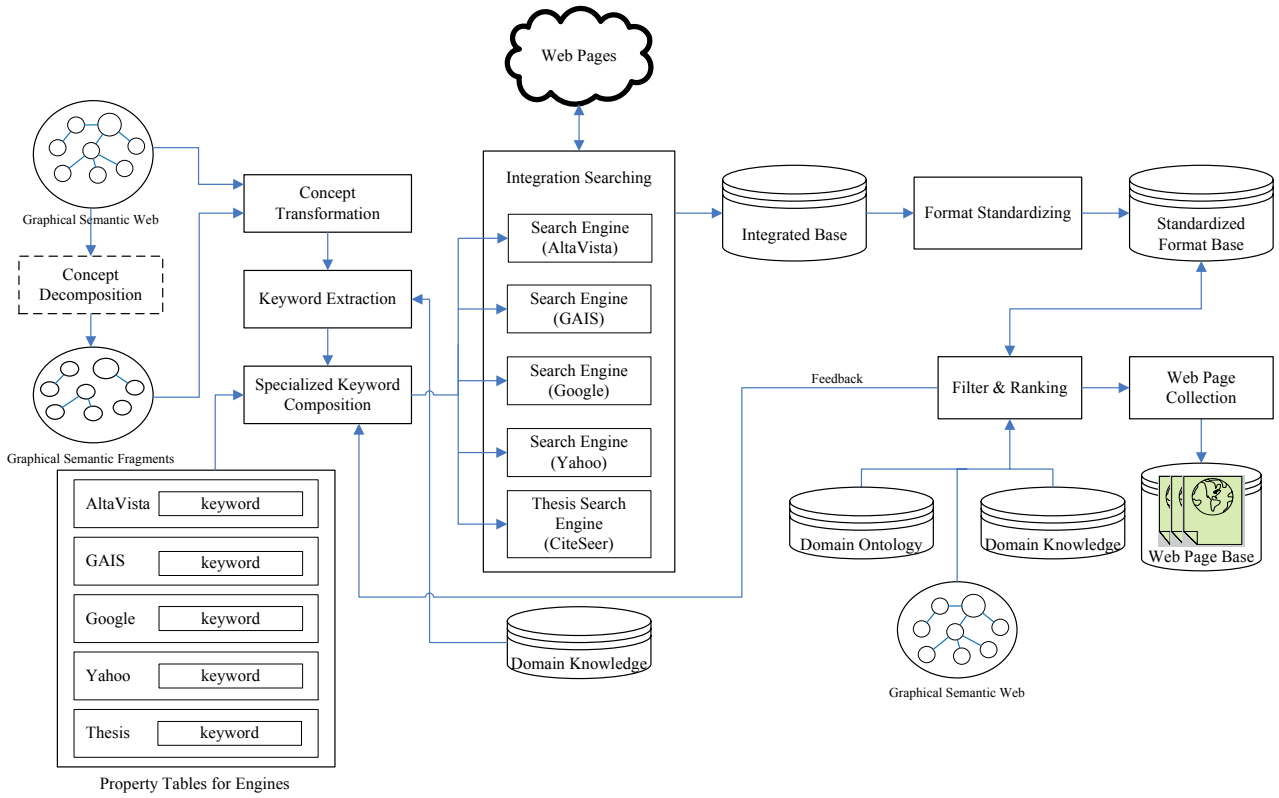

圖 1：網路整合式搜尋引擎機制架構

前處理機制

前處理機制會依循第一階段產出的圖形化語意網，針對關鍵字進行處理將其組合成有意義的字組，而且置換成符合每具搜尋器的語法，其步驟如下：

一、concept transformation：將類似網狀結構的圖形化語意網輸入至 concept decomposition mechanism，主要功能是把完整圖形化語意網進行分割的動作，分成多個有意義的子概念，此作用是為避免完整的概念於網路上無法找到符合的結果。簡單來說，圖形化語意網是根據使用者的問題解析並過濾無意義部份，搭配 5W1H(who、why、what、when、where、how)規則及意圖推算出最佳語意網，彙整出有關此問題的關鍵字集；此外透過實體論找出另一組關鍵字，其優勢可讓不同特性的概念與概念間，依某種關係尋找所需的語意內容，建立起真正可表示使用者意圖的關鍵字。因此搜尋的項目分別包含兩類，一是根據 5W1H 規則與意圖所推算出的語意網，二是本體論的語意網，稍作彙整後將傳遞至本研究所選定的四具著名搜尋引擎(AltaVista、GAIS、Google、Yahoo)進行資訊搜尋。

此機制包含 keyword processing 與 specialized keyword composition 兩個部份，前者是針對語意網中的關鍵字進行處理，經由領域知識(domain knowledge)與 5W1H 規則判斷形成關鍵字集(keyword set)，其判斷方法為公式 3；後者是依據每具搜尋引擎之特性或屬性特徵，透過程式客製化成符合搜尋語法，形成領域關鍵字集(domain keyword set)。關鍵字配合布林函數 AND 符號(&)作排列組合，但排除重複情況及零排列，計算出總共有幾種，若檢索無解則採用關鍵字遞減方式逐一搜尋，如有三組關鍵字分別為國小、四則運算及乘法，先利用三組字組合出關鍵字，從中擇其一較貼切原意的字組優先搜尋；無解則利用兩組關鍵字或單組關鍵字進行排列組合，再

置入搜尋器搜尋；若檢索依然呈現無效結果便啟動後處理機制中的回饋機制(feedback)，其內容將於第四章詳細說明，其他依此類推繼續執行前述步驟，如圖 2 所示。

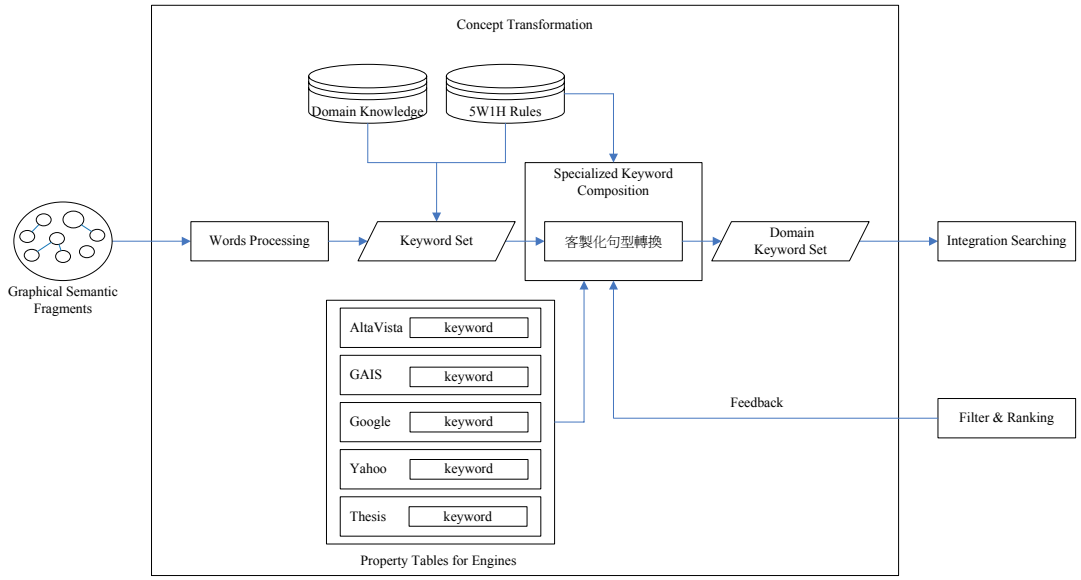

圖 2：concept transformation 機制

整合式搜尋機制

一、integration searching：每具搜尋引擎至全球資訊網檢索，其搜尋結果經由彙整將被置入integrated base，由於網路上搜尋的資訊大多以非結構為主，容易造成系統負荷與讀取誤判，為求資訊一致性，所以每則資訊經處理且儲存於資料庫，其資料表欄位包含搜尋引擎名稱、網址、關鍵字、標題、標籤、詞彙數目、摘要位置、摘要與原始文件等訊息。表1則為本研究所整理的四個著名搜尋引擎比較表。

表1：搜尋引擎比較表

	AltaVista	GAIS	Google	Yahoo!	
搜尋方式	以關鍵字查詢為主	以關鍵字查詢為主	以關鍵字查詢為主	以分類目錄瀏覽為主	
搜尋字數上限	中文 800 字 英文 800 字	中文無 英文無	中文無 英文 2048 字	中文無 中文 100 字	英文無 英文 100 字
搜尋格式	html、pdf、ppt、doc、	htm、html、xml、txt	rtf、ps、pdf、xls、ppt、	htm、html、pdf、xls、ppt、doc、xml、txt	

	xml、txt		doc、txt	
類似查詢	有	有	有	有
自然語言查詢	無	有	有	無
多國語言查詢	有	中文、英文	有	有
欄位查詢	url	url	link、related	title、url
刪除重複網址	無	無	無	無
刪除無效連結	無	無	無	無
刪除廣告	無	無	無	無
搜尋引擎排名	全球第 1	台灣第 6	全球第 6	全球第 2

後處理機制

針對後處理機制檢索後的資訊格式重新整理，使其有一致性，接續藉由本研究所設計比對演算法與排序演算法計算每則摘要之資訊含量及權重值，同時訂定門檻值作排序，其後則應用領域實體論(domain ontology)概念處理摘要中涉及同義詞部份，這是本研究重點核心之一，最後經由判斷並獲取符合原意的網頁，以便於知識萃取，其步驟如下：

一、format standardizing：主要透過程式執行任務；網路上的資料格式大多以HTML(Hyper Text Markup Language)與 PDF(Portable Document Format)呈現且內容都是非結構與半結構性，利用 XML(Extensible Markup Language)可達成結構化目的；XML 是一套資料的描述語言，主要是用來設計網頁中可攜帶結構化的資訊，並且允許使用者可以自行定義和它們文件相關的標籤，同時可透過自訂標籤、屬性、XML schema 與 DTD(Document Type Definition)[3][5]，對於每則摘要的相關資訊進行定義成為標準格式，可謂 format standardizing，隨後儲存至 standardized format base。

二、filter & ranking：如圖 3 所示，將每則資訊已格式標準化的摘要與標題進行一連串處理，主要依據圖形化語意網和領域知識所提供的知識為基準，搭配領域實體論

與原意，透過演算法運算資訊含量、權重值與相似度；其中相似度是利用本研究所建立的同義詞規則，計算句子中詞彙與資料庫中詞彙的相似程度，可以解決意思同描述不同之摘要問題；最後給予適當門檻值並排列順序且逐一與原意比對找出真正符合的資訊。

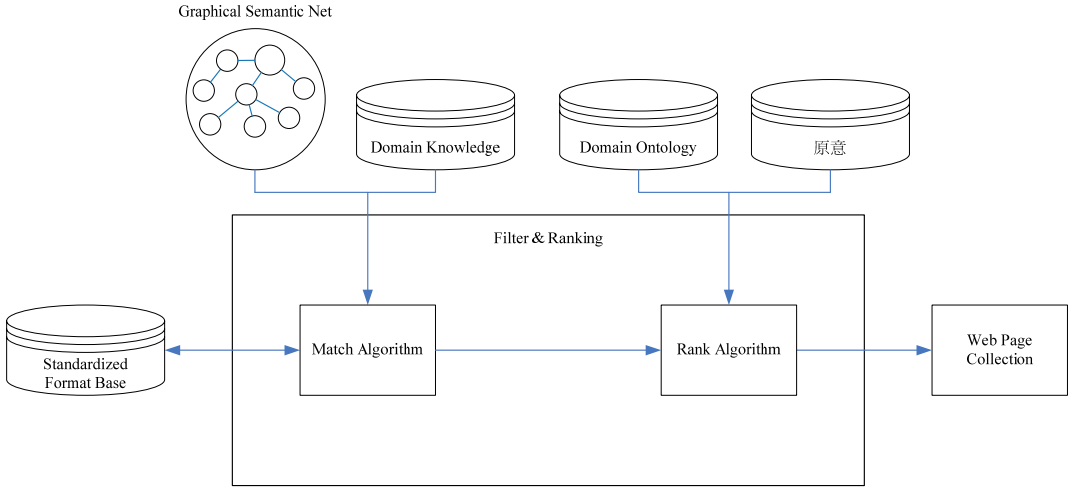

圖 3：filter 與 ranking 機制

首先系統過濾網址，藉由程式來判斷 URL 是否有重複，重複則直接刪除資料庫中的相關資料列，反之針對摘要與標題處理成可比較單位與 N 元詞(N-gram)，由 N 個字所組成則稱為 N 元詞，利用 N 元詞特性及句子結構建立句子中所有的可比較單位，前述資訊含量其相關計算公式如公式 1，公式 2 所示，因此標題分別與圖形化語意網和領域知識運算數值並相加產生資訊含量值，而且每篇摘要的資訊含量會分成高、中與低三類，資訊含量低的摘要立即剔除，保留含量高及中等之資訊，本研究所提的資訊含量等級會藉由系統實作後設立適當值域，計算每篇摘要的資訊含量，其計算公式如下：

$$R(i, j) = \frac{(U_i \cap U_j) * 2}{U_i \cup U_j} \quad \text{(公式 1)}$$

$$S(i) = \sum_j R(i, j) \quad \text{(公式 2)}$$

：中文詞彙

U_i ：所有句子 I 與句子 J 中相同的可比較單位

$U_i \cup U_j$

其中有四個比較法則：

- N 元詞只能與 N 元詞比對
- 標注詞只能與標注詞比對
- 每個詞只能比對成功一次
- 詞的比對不考慮順序性

上述步驟是將不符合的項目自動地刪除，剩下符合項目，換句話說不符合項目是指資訊含量低與重複網址的摘要，反之符合項目包含資訊含量高與中等的摘要，統稱為比對演算法(match algorithm)，演算流程如圖4所示。

本研究採取Gruber的定義：Ontology 是一種對某一個概念的詳細描述，包括對

於概念、關聯、實體的描述[17]。其後再針對資訊含量中等與高等摘要進行判斷，部份摘要是描述不同但意思相同可稱同義詞，所以可給予相等權重值，至於計算權重的法則有下列五點[8][9]：

(一)、頻率關鍵詞法：動詞與名詞是句子的核心部份，文件中每一個動詞與名詞皆視為重要詞彙，而詞彙的重要程度，則視該詞彙在文件中所發生次數多寡。

(二)、標題關鍵詞法：一篇文章的標題往往選取與主題相關的字詞所組合而成，因此出現在標題的字詞要給予較高的權重值。

(三)、位置法：一篇文章最重要的部分大部分位於文章的首句與末句；學者曾指出簡單的摘錄文件中的前 60、150 或 250 個詞彙，便達到了 90%以上的可接受度。

(四)、標籤線索法：超文件提供某些特殊標籤，如：斜體字、粗體字、底線與大小寫字體，都可以呈現相關重要的訊息。

(五)、領域實體論法：句子是由詞彙所組成，但詞彙之間會存在著某些特定關係；同義詞包含廣義上的相關詞與狹義上的同義詞，前者是指某篇摘要描敘不同，但意義與原意類似的詞，在此本研究根據主觀的判斷，蒐集詞彙拆解後的意思，進一步架構相關詞規則；後者是指某篇摘要述敘不同，但意義與原意相同的詞[，本研究透過蒐集並觀察詞彙，建構同義詞規則，如：幾何與代數同屬於數學；先乘除後加減是意指四則運算等諸如此類關係。在此會善用已建立的領域實體概念架構，裡面包含同義詞與延伸詞，設法從中條列出詞彙的規則。目前研究彙整出各四條規則分別為縮寫相關詞(同義詞)、單字相關詞(同義詞)、多字相關詞(同義詞)及中英文相關詞(同義詞)，其餘規則尚在研討分析中，日後會逐一建入。至於關鍵字與相關詞或詞彙與同義詞的相似度計算，其公式如下：

$$Sim\left(W_1, W_2\right) = \frac{2 \times \left|S\left(W_1\right) \cap S\left(W_2\right)\right|}{\left|S\left(W_1\right)\right| + \left|S\left(W_2\right)\right|} \quad \text{(公式 3)}$$

W_i：中文詞彙

$S(W_i)$：將中文詞彙W_i拆解成詞素，所得的詞素集合

$|S(W_i)|$：詞素集合 $S(W_i)$長度

其效能將於第五章系統實作詳述，但五條法則乃需要依實際情況去做調整，才能進行權重之運算。簡言之領域實體論法有五大步驟：

1、從Domain Ontology中蒐集整理相關詞與同義詞並建立其規則。

2、依摘要中的關鍵字選定適合之規則進行運算或比對，關鍵字可能是N元詞或標注詞。

3、適用規則一則利用詞彙之詞素與關鍵字計算相似程度，判斷此值位居何種值域中，給予相對等的權重。

4、適用規則二與三則利用自行建立的關係表直接判斷，其後給予已訂立於資料庫中的權重值。

5、完成一至四步驟後，即可得知同義詞權重(Synonym Weight)。

前述逐一完成後將運算總體權重值，其公式如下：

$$SCORE = \sum_{k=1}^{n} TP_k + PW + \sum_{l=1}^{m} T_l W_t + SW \quad \text{(公式4)}$$

：摘要中第k個詞彙的權重　n：重要詞彙總數

TP_k：位置權重　　：詞彙的標題與標籤權重　m：加權詞彙總數

PW：同義詞權重(Synonym Weight)　　TW　　：總得分

SW　　$SCORE$

摘要經計算後此處會設定適當門檻值，此值將於第五章中說明，取出門檻值以上的摘要並由高至低排序，接著依順序取每筆摘要與原意進行相似度運算判斷是否符合，不符合時則擷取第n筆資料來檢驗，若檢索結果無較佳解立刻啟動回饋機制返回concept transformation，此時除了依前述的關鍵字集重新採用詞彙遞減方式來組合之外，更透過5W1H規則中的映對(mapping)關係，如本欲查詢有關「輕度障礙生對四則運算認知程度」方面的問題，但在5W1H規則裡卻映對到「數學閱讀障礙程度」，兩者之間看起來存在某種程度的關聯性，因此可利用這對映關係進行搜尋，希冀能檢索出符合的答案。此後依然藉由specialized keyword composition客製化成符合的搜尋語法繼續於網路上檢索，其他則依此類推繼續執行每個步驟，統稱為排序演算法(rank algorithm)，演算流程如圖5所示。

三、web page collection：取回經由格式標準化與過濾排序的實際網頁內容，並儲存於 web page base。

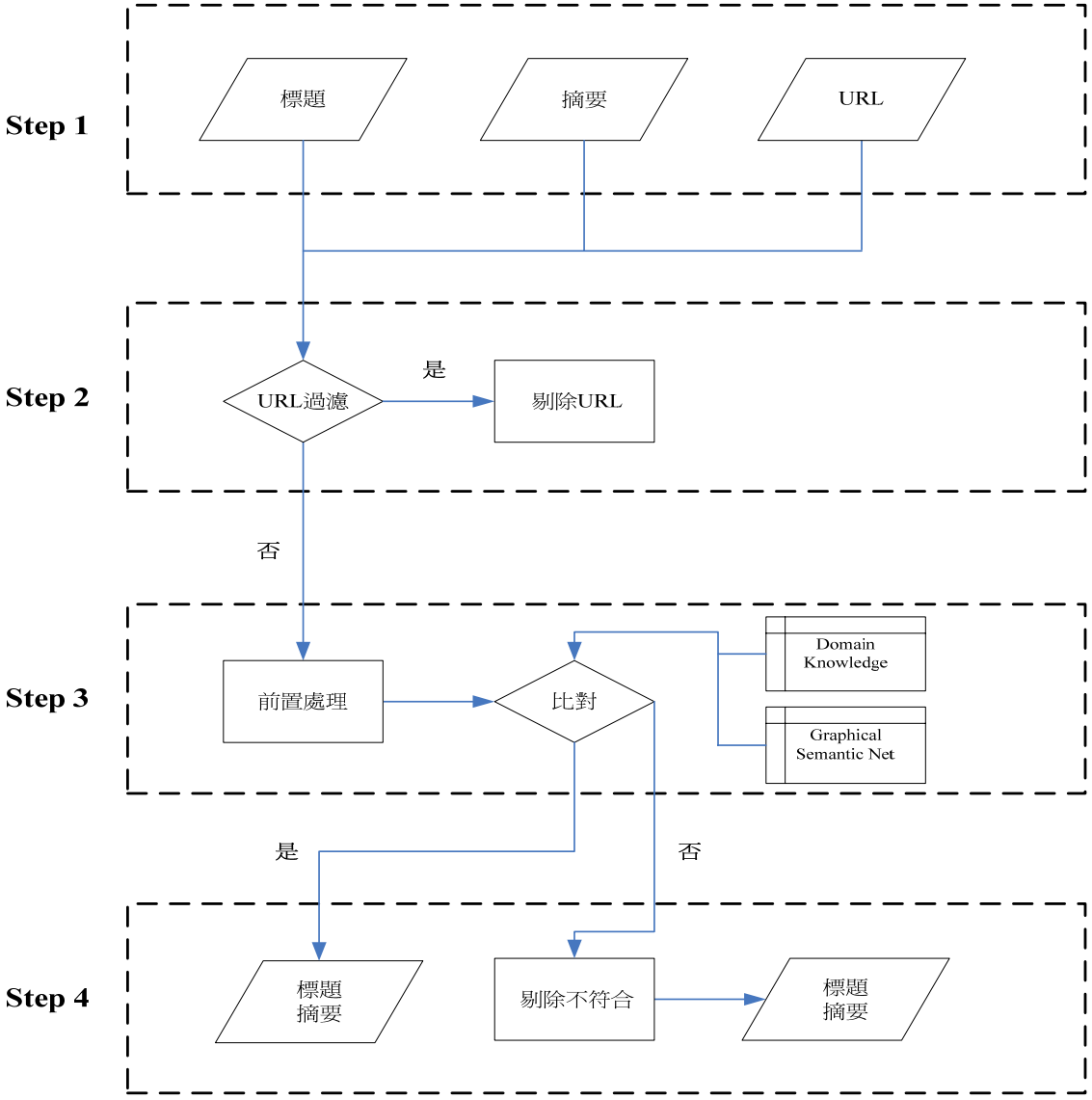

圖 4：比對演算法流程

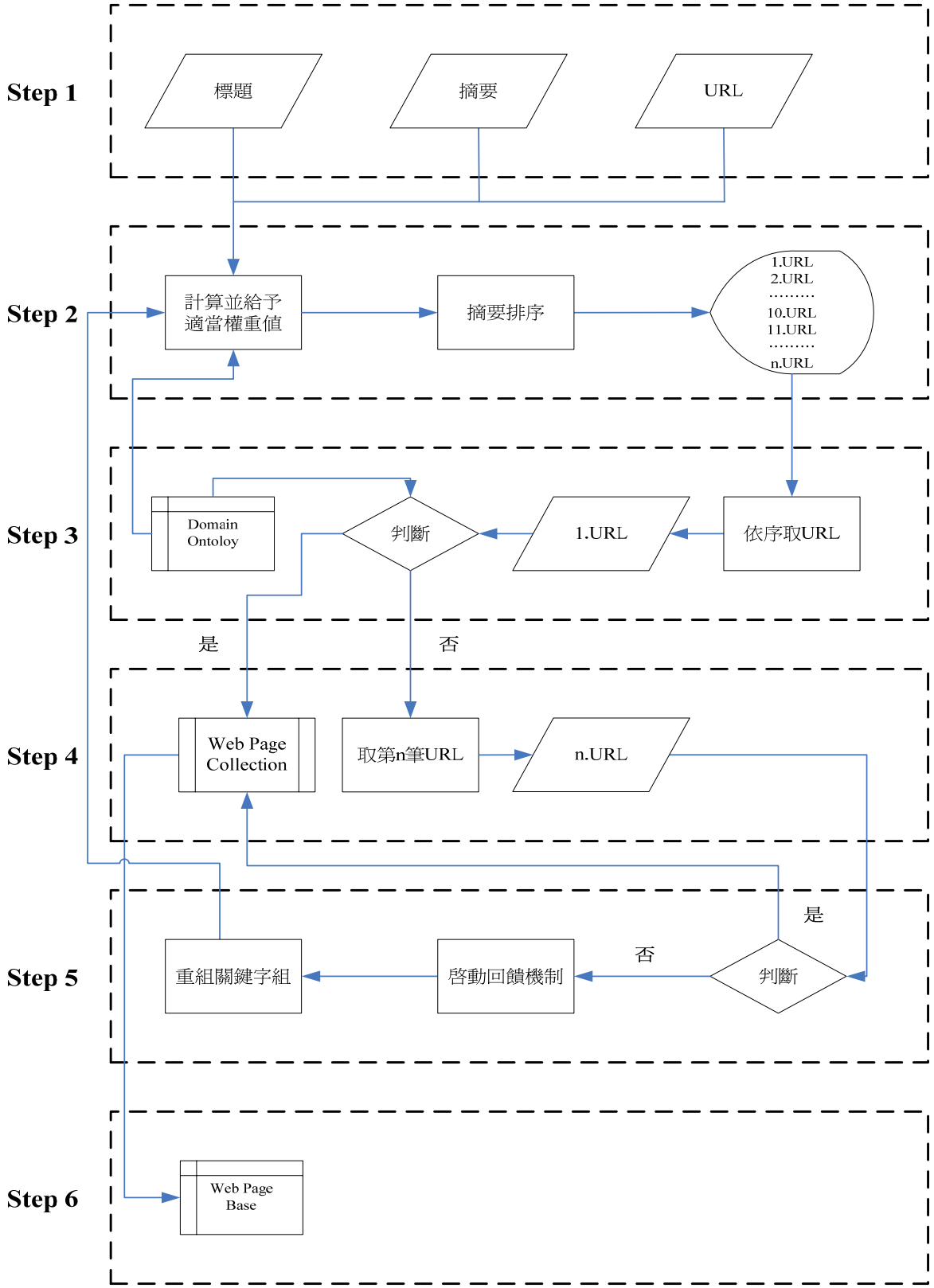

圖 5：排序演算法流程

結論

經研究發現各家搜尋引擎功能都相當齊全，檢索結果差異性不大，重點能減少頻寬負載量，避免使用者浪費精神與時間自行過濾結果，因此本研究設計一套強化

的搜尋引擎過濾與排序機制，透過比對演算法，去除重複性網址，並求得每則摘要的資訊含量；排序演算法則可計算得知每則摘要的權重值；導入領域實體論法，一來可解決描述不同但意義卻相同的摘要，二來可影響摘要權重值排序及抓取真正網頁的優先權。最後本研究預期產出及貢獻如下：

一、設計建構關鍵字集組合、格式標準化與過濾排序機制的整合式搜尋引擎。

二、提出領域實體論法以解決摘要中的相關詞及同義詞問題。

三、減少使用者檢索時間讀取重複性太高的網頁且提高準確性與效能。

四、提供較貼切原意的網頁給予使用者作為參考。

五、資料文件利用 XML 技術達成一致性與結構化目的，有利於日後增減修改或傳遞，不會因為頻寬問題而損毀遺失。

現今網路上搜尋的摘要，都以關鍵字為主，經研究發現呈現結果都以摘要特定內容段落為主，導致無法確實檢索網頁裡所潛藏的知識，未來可改用網狀結構圖形取代關鍵字為主之搜尋，其效能雖尚待評估，但是對於格式標準化過程能嵌入更有意義的內容與標題，有助於往後知識萃取或知識樹建構。由於前述各機制與演算法尚在積極進行中，因此期盼將來能建構出更完善的概念、機制或模型，如：領域本體論概念應用於摘要中關鍵字的歧義詞部份，若某關鍵字無法找出所需答案，可利用反義詞再次檢索。

參考文獻

[1] Cooley, R., Mobasher, B. and Srivastava, J., “Web mining：information and pattern discovery on the World Wide Web,” *9th IEEE International Conference on Tools with Artificial Intelligence (ICTAI’97)*, 1997, pp. 558-567.

[2] Jenkins, C., Kackson, M., Burden, P. and Wallis, J., “Searching the world wide web：an evaluation of available tools and methodologies,” *ELSEVIER Journal on Information and software technology*, 1998, pp. 985-994.

[3] Norman, Walsh, “A Technical Introduction to XML,” *World Wide Web Journal*, 1998 (http://www.nwalsh.com/docs/articles/xml/).

[4] Spertusm, E., “ParaSite：Mining Structural Information on the Web,” *The Sixth International World Wide Web Conference (WWW6)*, 1997, pp. 1205-1215.

[5] 王常威，『以內容為基礎之 XML 文件分類方法之研究』，2004，成功大學資訊管理研究所碩士論文。

[6] 陳麴合，『超連結與關鍵字頻分析之搜尋引擎研究』，2001，屏東科技大學資訊管理研究所碩士論文。

[7] 許志新，『分散式搜尋引擎之設計與實作』，1996，中正大學資訊工程研究所碩士論文。

[8] 邱立豐，『互動式概念查詢應用於網路文件自動摘要之效益』，2002，雲林科技大學資訊管理研究所碩士論文。

[9] 黃純敏、吳郁瑩，『網路中文文件自動摘要』，台灣區網際網路研討會TANET，1999，國立中山大學承辦。

[10] 柯淑津，『從詞網出發的中文複名詞的語意表達』，International Journal of

Computational Linguistics and Chinese Language Processing，2003，pp. 93-108。
[11] 謝文泰、陳鈺文、張覆平，『以句子資訊量來產生文件摘要之模式』，財團法人資訊工業策進會。
[12] The AltaVista Search Engine, http://www.altavista.com/.
[13] The GAIS Search Engine, http://gais.cs.ccu.edu.tw/.
[14] The Google Search Engine, http://www.google.com/.
[15] The Yahoo Search Engine, http://search.yahoo.com/.
[16] The CiteSeer Engine, http://citeseer.ist.psu.edu/.
[17] Tom Gruber, Ontology Definition,
http://www-ksl.Stanford.edu/kst/what-is-an-ontology.html.

附件4

以領域本體論為基之概念地圖自動化建構與文件分類機制

摘要

近年來隨著網際網路的普及與知識經濟的快速發展，網路上充斥著大量的訊息資料。但伴隨而來的便是造成使用者難以藉由網際網路找尋到符合自己需求的訊息內容並且耗費大量的時間在過濾無效的資訊徒增檢索資訊時的困擾。如何有效運用網路上所提供的訊息內容，並提供使用者有效的知識為現今的一大挑戰。

本研究以本體論為基礎結合文字探勘技術，藉由以使用者輸入內容所進行網頁文件資料探勘，進行相關領域概念內容擷取，並運用關聯法則進行自動化建構領域概念圖。藉由自動化所產生的領域概念圖來支援文件分類，提升使用者進行文件知識檢索時的效率，並提供以查詢內容為核心之領域概念擴展圖，強化使用者對整體概念內容之了解。本研究主要著重於國小之輕度障礙學生數學教學領域，期望藉由本研究幫助輕度障礙學生之教師或家長能快速且有系統的進行知識獲取。

關鍵詞：文字探勘、領域概念圖、資料分類

一、 緒論

近年來，隨著資訊科技的快速發展與網際網路使用的普及化，使得人們許多的活動都轉移至網際網路上進行，從早期的網頁至現今的 Web 2.0，促使人們便於進行資訊分享活動，但這也使得人們在遭遇問題時便會即刻反應至網路上搜尋相關問題的解答。目前人們藉由搜尋引擎所搜尋回來的資料量都非常的龐大且雜亂無章，徒增使用者進行資訊檢索時的負擔。加上這些內容所提供的資訊內容都是片片段段並且零散的資料，使得使用者所接受的訊息量超過其所能負載的程度，並無法有系統的了解知識內容的整體概念。

輕度障礙生進行教學輔導時必須因材施教，假若遭遇特殊案例學生，一般傳統教師會根據過去經驗或查閱書籍文章來解決輕度障礙生有關學習方面之問題，但年輕教師常受限於過去之經驗不足或相關書籍文章缺乏，難以找尋合適之教學案例來教導特殊障礙學生。在搜尋相關資料期間，輕度障礙生的學習狀況常因而停頓下來，無法一連貫進行系統化教學。因此本研究期望藉由教師或家長所提問之問題至網路搜尋回與問題相關之文件內容，經由整理、過濾、分類，建構出該領域之領域概念圖。協助教師或家長對於輕度障礙領域內容之整體了解，並提供與領域概念相對應之文件，降低使用者檢索時之訊息負載量也加快找尋相關文件之速度。

二、 研究架構

為了有效率的藉由網際網路所搜尋回的文件進行概念地圖建構，並運用所建構出之知識地圖來對文件進行分類，本研究有下列三個主要部份，如圖一所示：

1. 概念擷取(Concept Extraction)
   - 中文詞彙斷詞處理
   - 詞彙過濾

- 特徵詞擷取
- 同義詞擴展

2. 領域概念建構(Domain Concept Construct)
   - 概念地圖建構
   - 定義概念關聯程度
3. 文件分類(Document classification)
   - 相似度計算
   - 文件過濾與排序
   - 資料分類

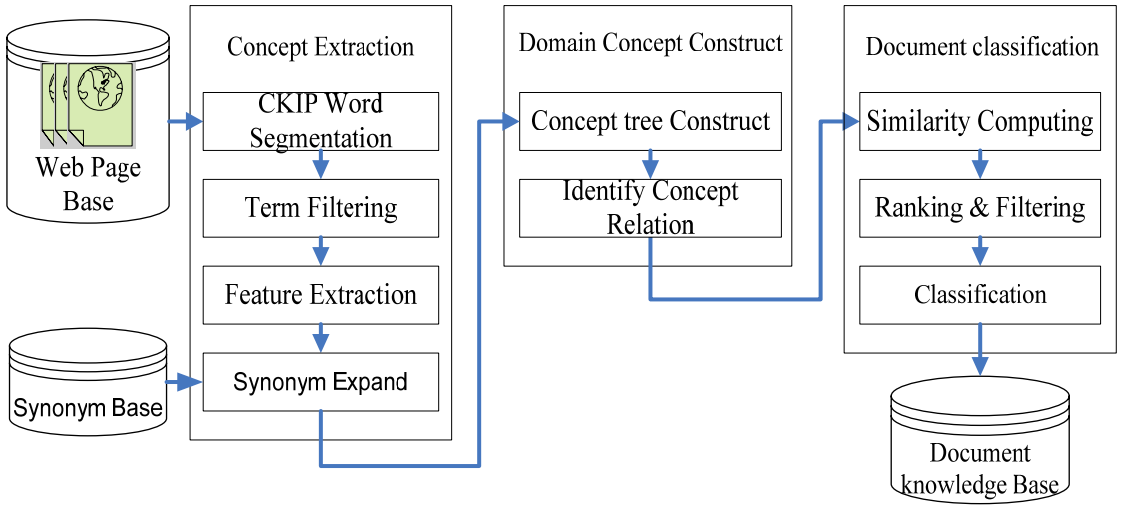

圖 1.概念地圖自動化建構與文件分類機制架構圖

三、 概念地圖自動化建構與文件分類機制

1. 概念擷取

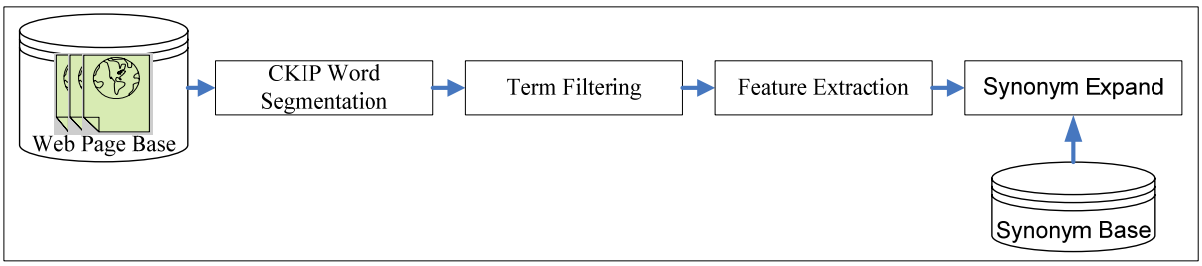

圖 2.概念擷取機制

本機制(圖 2)主要包含五步驟，將依據使用者的問句所搜尋回來之文件，利用中研院 CKIP 進行斷詞及標記詞性處理、過濾有效詞彙、特徵詞擷取及同義詞擴展，其步驟主要功能如下：

1.1. 中文詞彙斷詞處理

對於任何語言來說，詞是自然語言的基本單位。由於中文文字表達方式與歐美語系表達方式不同，以英文為例：其詞彙之間是以空白符號做為區隔；中文文字卻沒有適當的判斷方式。所以，欲了解中文文件所表要達之意義，則必須對中文文件進行詞性標記的前處理，以找出使用者的語意內容。我們以中央研究院詞庫小組所研發的 CKIP 中文斷詞系統，來處理本研究的中文文件資料。以「如何學習除法」

為例子來說明，經由 CKIP 斷詞及標記後分為三部分「如何(D)學習(VC)除法(Na)」。

1.2. 過濾有效詞彙

將經由中研院 CKIP 斷詞處理後之文件進行詞性過濾，過濾掉停用字、冠詞、介係詞及連接詞等與文件概念無關的字詞以降低無異議之特徵詞彙。再運用下列歸納出之詞彙合併規則進行相關詞彙的合併，以顯示出詞彙所要表示的真正意義。

1. VJ(狀態及物動詞)+VH(狀態不及物動詞)
   Ex:抗(VJ)+憂鬱(VH)
2. Na(普通名詞)+VB(動作類及物動詞)
   Ex:穴道(Na)+指壓(VB)
3. Na(普通名詞)+Nc(地方詞)
   Ex:心理(Na)+科(Nc)
4. Na(普通名詞)+VH(狀態不及物動詞)
   Ex:營養(Na)+不良(VH)
5. Na(專有名詞)+Na(普通名詞)+Na(普通名詞)
   Ex:身心(Na)　障礙(Na)　學生(Na)
6. VA(動作不及物動詞)+VH(狀態不及物動詞)
   Ex:營養(Na)+不良(VH)
7. A(非謂形容詞)+Na(普通名詞)
   Ex:膠原(Na)+蛋白(VH)

1.3. 特徵辭擷取

根據 Wu et al.(2002)[9]的研究指出， 主題字及關鍵字通常是由名詞-動詞(Noun-verb)及名詞-名詞(Noun-noun)的配對所組成，因此藉由詞性標記及篩選便可挑選出有意義之詞類。再經由領域相關度、領域一致性與領域權重值計算[3]，挑選出該領域之專有名詞。

- 領域相關度計算

主要計算該名詞的領域關連程度 DR：該候選專有名詞在領域文集出現的機率。

假設有 n 個領域{D1,D2,…,Dn}，則一候選專有名詞 t 對特定領域 Dk 的領域相關度(DRt,k)定義如下：

$$DR_{t,k} = \frac{p\left(t\middle|D_k\right)}{\sum_{j=1}^{n} p\left(t\middle|D_j\right)}$$

where $\mathrm{P}\left(t\middle|D_k\right)$ is probabilities of

term $\mathrm{t} \in \mathrm{D_k}$

$$\mathrm{P}\left(t\middle|D_k\right) = \frac{f_{t,k}}{\sum_{t' \in D_k} f_{t',k}}, \textit{where} \quad f_{t,k} = \sum_{d \in D_k} f_{t,d}$$

- 領域一致性

用來評斷該候選專有名詞是某可以成為該特定領域的專有名詞，評量跟名詞是否經成出現在該領域的文件中。

假這一候選專有名詞 t 對特定領域 Dk 的領域一致性(DCt,k)，定義如下：

$$DC_{t,k} = \sum_{d \in D_k} \left(P_t\left(d\right) \log \frac{1}{P_t\left(d\right)} \right)$$

where $P_t\left(d\right)$ is the probability that document d includes term t

$$E\left(P_t\left(d\right)\right) = \frac{f_{t,d}}{\sum_{d' \in D_k} f_{t,d'}}$$

where $f_{t,d}$ is the term frequency of term t in document d

- 領域權重值

以一個線性組合結合候選專有名詞的領域一致性(DC)及領域關聯度(DR)做為該名詞對該領域的權重值。候選專有名詞 T 對領域 Dk 的領域權重值 DWt,k，定義如下：

$$DW_{t,k} = \alpha DR_{t,k} + (1-\alpha) DC_{t,k}^{norm}$$

$$\text{where } \alpha \in (0,1)$$

Normalized function：

$$DC_{t,k}^{norm} = \frac{DC_{t,k}}{\underset{t' \in D_k}{Max}\left(DC_{t,k}\right)}$$

1.4. 同義詞擴展

經由上述動作處理過後便可擷取出該領域的相關概念詞彙，再運用同義詞擴展進行相關詞彙的概念化統一。藉由同義詞庫的比對便可將慈惠進行概念化的統一。

2. 領域概念建構

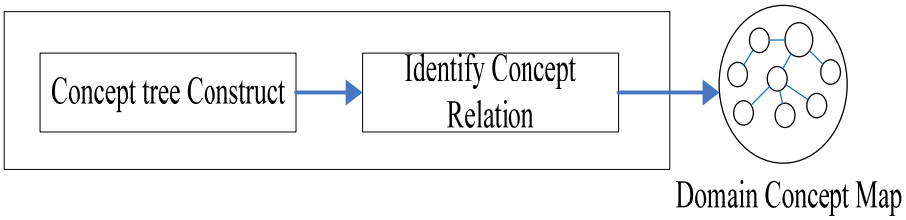

圖 3.領域概念建構機制

在本研究中此階段須假設：”一份文件中若多個關鍵特徵詞同時出現，則它們必定存在其關聯性。”，因此在本研究主要利用由 Apriori 演算法所衍生出的 DHP 演算法[1]進行特徵詞間關聯性的搜尋，運用事先訂定之知識本體為核心去搜尋出與該知識本體相關的領域關鍵詞，以便進行之事本體之擴展，主要步驟如下：

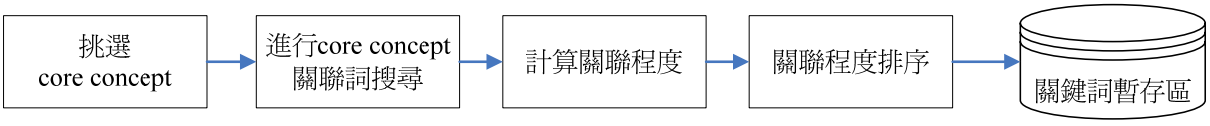

圖 4.概念地圖建構流程

(1)挑選 Core Concept：

將文件中之關鍵詞與已訂定之知識本體進行關鍵詞比對，篩選出核心概念關鍵詞。

(2)進行 Core Concept 關聯詞搜尋：

運用 DHP 演算法進行與核心關鍵詞高度共同出現頻率之詞彙之篩選。將每篇文件之特徵詞進行集合，再將每一特徵詞逐一配對，篩選出經常共同出現之詞組。

(3)計算關連程度：

藉由設定關聯詞之門檻值及計算特徵詞組共同出現之頻率，篩選出高於門檻值之詞組。

(4)關連程度排序：

依照詞組共同出現之門檻值進行關連程度由高至低進行排序。

3. 文件分類

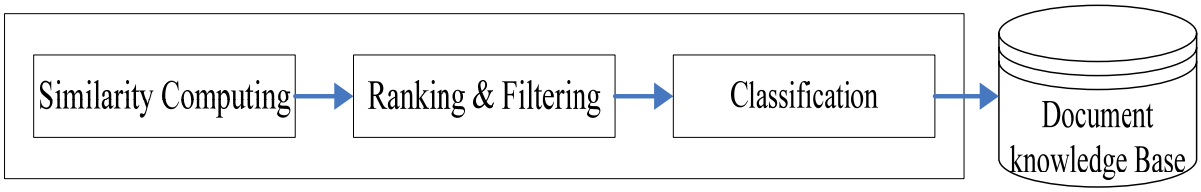

圖 5.領域概念建構機制

3.1.1. 相似度計算

將文件與自動化所建構之領域概念本體進行相似度計算，便可依概念結構進行文件分類，將文件分類至相關概念內容下。主要運用 cosine coefficient 公式來計算[4]，公式如下：

$$C(X,Y)=\frac{f_{X\cap Y}}{\sqrt{f_X}\times\sqrt{f_Y}}$$

3.1.2. 排序與過濾

將經由相似度計算過後的文件內容進行排序與過濾，過濾低於門檻值之文件內容。

3.1.3. 資料分類

經由排序與過濾相似程度不高之文件後，文件內容便依照與概念之相近程度進行分類，便於使用者之查詢使用。

4. 結論

本研究提出一套以使用者問題為導向，藉由使用者查詢之關鍵字內容至網際網路搜尋相關文件，再將文件內容自動化建構成領域概念圖，藉由領域概念圖支援文件分類。期望藉由概念圖的建立，讓使用者進行問題查詢時能加快系統回覆速度，並藉由提供領域概念圖使其了解整體知識概念內容。

本研究之具體貢獻如下：

(1)藉由圖型化呈現領域概念圖，可讓使用者系統化的了解與其問題概念相關之概念內容。

(2)藉由領域概念之文件分類，增進使用者進行文件搜尋時之效率與準確率。

參考文獻

[1] J. S. Park, M. S. Chen, and P. S. Yu, “An effective hash based algorithm for mining association rules,” Proceedings of the ACM SIGMOD International Conference on Management of Data, San Jose, USA, pp. 175-186 (1995).

[2]陳永德(1997)，中文斷詞中長詞優先、詞頻對比及前詞優先規則之使用，國立臺灣大學心理學研究所博士論文。

[3]廖崇倫(2004)，基於概念模型自動化建構技術之智慧型資訊擷取，國立成功大學資訊工程系研究所碩士論文。

[4]魏玲玉、曾守正，以文件倉儲概念實現動態群聚與多重文件摘要之研究-以中文電子新聞為例，資訊管理學報，2006/07，pp.153-176。

[5]潘雅真(2004)，企業式知識地圖，中華大學資訊管理系研究所碩士論文。

[6]Uschold, M. & Gruninger, M. (1996). Ontologies: Principles, Methods and Application.Knowledge Engineering Review, Vol. 11, No. 2., 39-73.

[7]Gruber, T.R. (1992). A Translation Approach to Portable Ontology Specifications (Technical Report KSL 92-71). Stanford University, Knowledge Systems Laboratory.

[8]Uta Priss, http://www.upriss.org.uk/fca/fca.html, 2003.

[9] Wu, S. H., Day, M.Y., Tsai, T.H. and Hsu, W.L., FAQ-centeredOrganizational Memory, in Matta, N. and Dieng-Kuntz, R. (ed.),Knowledge Management and Organizational Memories, Kluwer Academic Publishers, 2002.

附件5

ARTICLE IN PRESS

Available online at www.sciencedirect.com

SCIENCE DIRECT

ELSEVIER

Robotics and Computer-Integrated Manufacturing ▮ (▮▮▮) ▮▮–▮▮▮

Robotics and Computer-Integrated Manufacturing

www.elsevier.com/locate/rcim

Secure resource sharing on cross-organization collaboration using a novel trust method

Tsung-Yi Chen[a,b], Yuh-Min Chen[a,*], Chin-Bin Wang[b], Hui-Chuan Chu[c], Huimei Yang[d]

[a]Institute of Manufacturing Engineering, National Cheng Kung University, Tainan, Taiwan
[b]Electronic Comerence Management Department, Nan Hua University, Chia-Yi, Taiwan
[c]National University of Tainan, Tainan, Taiwan
[d]Department of Business Administration, Tatung Institute of Commerce and Technology, Chia-Yi, Taiwan, ROC

Received 11 August 2005; received in revised form 24 April 2006; accepted 28 April 2006

Abstract

A virtual enterprise (VE) consists of a network of independent, geographically dispersed administrative business domains that collaborate with each other by sharing business processes and resources across enterprises to provide a value-added service to customers. Therefore, the success of a VE relies on full information transparency and appropriate resource sharing, making security and trust among subjects significant issues. Trust evaluation to ensure information security is most complicated in a VE involving cross-organization collaboration. This study presents a virtual enterprise access control (VEAC) model to enable resource sharing for collaborative operations in the VE. A scenario for authentication and authorization in the life cycle of a VE is then described to identify the main activities for controlling access. Also developed herein is a trust evaluation method based on the VEAC model to improve its security while safeguarding sensitive resources to support collaborative activities. The trust evaluation method involves two trust evaluation sub-models, one to evaluate the level of trust between two virtual enterprise roles, and another to measure the level of trust between two projects. The two sub-models support each other to make resource-sharing decisions, and are developed based on the concepts of direct, indirect, and negative trust factors. Finally, an example of measuring the trust between two subjects is demonstrated after introducing the two sub-models. The VEAC-based trust evaluation method enables the following: (1) secure resource sharing across projects and enterprises, (2) collaborative operation among participating workers, (3) increased information transparency and (4) lowered information delay in VEs.

Keywords: Virtual enterprise; Resource sharing; RBAC; Trust; Access control; Collaboration

1. Introduction

Most enterprises adopt a virtual enterprise (VE) business model for activities related to products and services required by customers. VEs evoke notions of cooperation, cohesiveness and trust among coworkers from different organizations to accomplish common goals. Hence, VEs have to respond quickly to customer expectations by integrating processes, activities and resources from different enterprises through enterprise alliances [1]. In practice, a VE is implemented with a distributed and collaborative business process, in which individuals from different enterprises cooperate on business-related activities or processes by remote coordination, communication and control [2–4].

Effective virtual enterprising requires fully transparent and effective sharing of resources, including information, application systems and knowledge, throughout the business cycle [1]. Information sharing, including real-time capability, enables operational improvements and reduces the overall cost [5]. Information resources to support the practical operations in VE can be classified into three categories: (1) information brought by participating enterprises, (2) information generated by activities in a

*Corresponding author. Tel.: +886 62757575x63922; fax: +886 62085334.
E-mail address: ymchen@mail.ncku.edu.tw (Y.-M. Chen).

0736-5845/$ - see front matter © 2006 Elsevier Ltd. All rights reserved.
doi:10.1016/j.rcim.2006.04.005

VE and (3) the information assets of a VE The three categories of information should be securely managed and shared with an appropriate mechanism. Charles et al. [6] explored a dynamic coalition problem by emphasizing information sharing and security risks among groups. Zha and Ding [7] analyzed the necessity and impact of sharing information among supply chain partners in several sharing modes. However, resource sharing introduces trust and authority management issues, and shows the significance of resource access comtrol.

Access control and sharing determines whether a subject can access resources controlled by another subject, and protects the confidentiality, integrity and availability of resources [8]. The subject, which is a member of the VE, can be an employee, role, agent or software application. Access control for VEs is difficult to accomplish because (1) members of the VE frequently change, (2) VEs have many members with often complex inter-relationships, (3) VEs may be integrated or distributed and (4) VEs are Internet-based and heterogeneous [9]. Because of the decentralized and dynamic characteristics in VE environments, access control for VE is impossible with traditional access control approaches [10,11].

Trust management in an organization refers to complex relationships among individuals, systems and organizational information management policies, and becomes particularly cumbersome in a VE, which involves cross-organizational activities [12]. Trust evaluation in a VE concerns safety and availability among individuals when delegating to partially trusted coworkers performing tasks concerning the aim of the VE. Therefore, the current trust model is not well suited to VEs due to its dynamic cooperative and collaborative properties. Trust management has been supported in part by some recent literature. Shand et al. [13] presented a trust and risk framework to enable secure collaboration in ubiquitous and pervasive computer systems. Tran et al. [14] developed a trust-based peer-to-peer access control framework with a scoring system to assess the access value by combining direct and indirect trusts with direct and indirect contributions. Dimmock et al. [15] applied the OASIS access control system, and extended role-based policy language to make decisions based on trust and risk analysis. Barrett and Konsynski [16] proposed a method for classifying inter-organization information sharing systems. Zuo and Panda [17] developed a labeling scheme after analyzing the issue of trust from two perspectives, the 'subject' and 'object'. Although access control across multi-enterprises has rarely been studied, a trust evaluation method should be developed for a VE for four reasons: (1) a VE differs from a peer-to-peer environment, (2) no model enables control of resource sharing across organization boundaries to support collaborative and cooperative business activities, (3) no model considers the trust evaluation among coworkers and projects [18] and (4) the resources accessed by users in the VE cannot be predicted.

This study adopts the virtual enterprise access control (VEAC) model to improve resource sharing and information transparency among enterprise members, and the VEAC-based trust evaluation method to increase the security, flexibility and scalability of resource sharing. One difficulty in measuring the trust of a subject among VE is the lack of a method to examine the degree to which a subject should be trusted [17]. This study first introduces a VEAC model for collaborative operation among each participating enterprise [19]. Second, a scenario for authentication and authorization in VE is presented to find the main authentication and authorization activities, and to indicate the interactive relationships among core access control mechanisms. Finally, a trust evaluation method based on the proposed VEAC model is developed by analyzing security problems, role rights, qualifications and responsibilities, project relations, cooperative relations and role hierarchical relations, which are the core components of VEAC. The VEAC-based trust method allows: (1) resource sharing across projects and enterprise boundaries, (2) secure collaborative operation among participating coworkers, (3) increased information transparency and (4) reduced information delays in a VE.

2. VEAC model

Although Wang et al. presented a VEAC model in [19], that study did not describe it in detail. Therefore, this section introduces the VEAC model and its basic components as depicted in Fig. 1. The model is derived from the resource management requirements and the characteristics of a VE, and includes two sub-models, a project-based access control (PBAC) model for managing public resources stored in a VE, and a role-based access control (RBAC) model for handling private resources held on individual enterprise members.

2.1. RBAC model

RBAC involves three fundamental components, the base model, role hierarchy and constraints. The bottom of Fig. 1 shows the RBAC model [20–23]. Elements and relationships in RBAC are described simply as follows:

- User (U), also called Subject, denotes a human, web service, application or agent in an enterprise.
- Role (R) denotes a set of functional jobs or responsibilities, and is expressed as a set of permissions.
- Private Object (PrivateO) is a sub-class of Object class, and denotes resources in an enterprise associated with private permissions.
- Private Permission (PrivateP) is the approval of a particular mode of access to one or more private objects.
- Session (S) represents each session, through which users map to one or more roles.

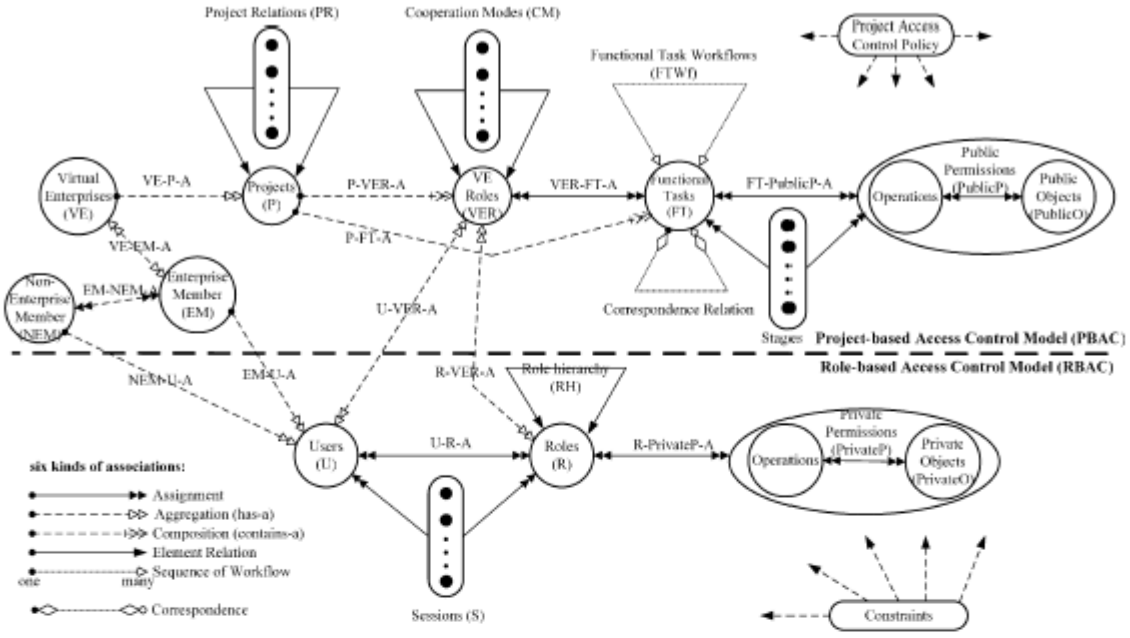

Fig. 1. VEAC model.

The RBAC model assigns each user to play roles associated with private permissions given to perform operations on a private object. A user only plays a role at a session where he can activate a subset of roles assigned to it. The following three relations among roles denote the privilege assignment of role: *Role Hierarchy*, *Static Separation of Duty* (SSD) and *Dynamic Separation of Duty* (DSD). The RBAC model utilizes two relationships to represent the aggregation relationships between two elements: EM–R-A between the Enterprise Member and Role elements, representing the Role elements in each Enterprise element, and EM–U-A between the Enterprise Member and User elements, representing the User elements belonging to the Enterprise Member element.

2.2. PBAC model

The PBAC model is shown in the upper layer of Fig. 1. The core concept of model development, elements and relations in the PBAC model are introduced and defined in the following sub-sections.

2.2.1. Fundamental elements

This sub-section introduces the fundamental elements of the PBAC model in the Set theorem:

- VE = {*ve*: *ve* is a dynamic Internet organization which consists of enterprise members (EM) performing a project to achieve one common business goal}.
- EM = {*em*: *em* can be a substantive enterprise organization, VE or individual, and is a VE member, with at least one worker participating directly in the VE activities}.
- *Non-Enterprise Member* (NEM) = {*nem*: *nem* can be a substantive enterprise organization, VE or individual, but not a VE member. A *nem* has at least one worker participating directly in activities of enterprise members, and the activities have direct relations with functional tasks of VE}.
- *Project* (P) = {*p*: *p* is the set of functional tasks, projects and sub-projects, which is performed by a VE}.
- *Functional Task* (FT) = {*ft*: *ft* is a set of VE activities, which has a common objective and is performed by several virtual enterprise roles (VER)}. A functional task involves five attributes:
  (1) *FT-state* records the state of the functional task being performed;
  (2) *FT-stage* records current timestamp of a functional task for appropriate resource sharing according to its states;
  (3) *Allowed-reference* is a Boolean data type to decide whether the functional task can be referred by relative functional task in a post-version project;
  (4) *Allowed-sub-project* decides whether the functional task can be referred by its sub-projects; and
  (5) *Allowed-main-project* decides whether the functional task can be referred by its super-project.
- VER = {*ver*: *ver* is a virtual role created to enable professional division within VE, which is assigned to perform more than one FT}.
- *Object* (O) = {*o*: *o* is an information resource including public and private resources which can be database, entity, attribute, tuple, document, XML document, application, software component or knowledge}.

- *Public Object* (PublicO) = {*public-o*: *public-o* is a subset of objects, which is owned by a VE and stored in a VE's common repository}.
- *Operation* = {*op*: *op* is a set of access authorities, such as "write", "read" and "execute"}.
- *Public Permission* (PublicP) = {*public-p*: *public-p* is a permitted mode of access to a public object}.
- *Permission* = {*x*: $x \in PublicP \cup PrivateP$}.
- *Project Access Control Policy* (PACP): PACP identifies which project resources are protected and shared according to the relations among projects and the sharing rules, and what activities are forbidden in the VE scope.

2.2.2. Foundational assignments

The various assignment relations among elements are defined as follows:

- *FT-S-PublicP-A*: a triple assignment among three elements: *Functional Task*, *Stage* and *Pubic Permission*. It is represented by $R_{ft\text{-}s\text{-}publicp\text{-}a} = \{(ft, st, public\text{-}p)$: $ft \in FT$, $st \in Stage$, and $public\text{-}p \in PublicP\}$ means that public permission *public-p* is assigned to functional task *ft* in stage *s*.
- *P-VER-A*: a one-to-many binary assignment is represented by $R_{p\text{-}ver\text{-}a} = \{(p, ver)$: $p \in P$, $ver \in VER$ and *p* "involves" *ver*}.
- *VER-FT-A*: a many-to-many binary assignment is represented by $R_{ver\text{-}ft\text{-}a} = \{(ver, ft)$: $ver \in VER$, $ft \in FT$ and *ver* "performs" *ft*}.
- *VE-EM-A*: a many-to-many binary assignment is represented by $R_{ve\text{-}em\text{-}a} = \{(ve, em)$: $ve \in VE$, $em \in EM$ and *em* "is a member of" *ve*}.
- *VE-P-A*: a one-to-many binary assignment is represented by $R_{ve\text{-}p\text{-}a} = \{(ve, p)$: $ve \in VE$, $p \in P$ and *ve* "performs" *p*}.
- *EM-NEM-A*: a many-to-many binary assignment is represented by $R_{em\text{-}nem\text{-}a} = \{(em, nem)$: $em \in EM$, $nem \in NEM$ and *nem* "supports" *em* "to perform some tasks of the" *VE-EM-A_virtual_enterprise* (*em*)}.
- *Functional Task Workflow* (FTWf): a many-to-many binary assignment is represented by $R_{\mathrm{FTWf}} = \{(ft_i, ft_j)$: $ft_i, ft_j \in FT$, $p_i p_j \in P$, $ft_i \subset p_i$, $ft_j \subset p_j$, $i \neq j$, ft_i is an event-functional task of the action-functional task ft_j} means ft_j is authorized to use the public permissions of ft_i while ft_i is accomplished.
- *Correspondence*: a one-to-one binary relation on *FT* is represented by $R_{\mathrm{correspondence}} = \{(ft_i, ft_j)$: $ft_i, ft_j \in FT$, $p_i p_j \in P$, $ft_i \subset p_i$, $ft_j \subset p_j$, $i \neq j$, ft_i "is the pre-version of" ft_j while ft_j is the post-version of" ft_i}.
- *EM-U-A*: a one-to-many binary assignment is represented by $R_{em\text{-}u\text{-}a} = \{(em, u)$: $em \in EM$, $u \in U$ and *em* "has an employee" *u*}.
- *NEM-U-A*: a one-to-many binary assignment is represented by $R_{nem\text{-}u\text{-}a} = \{(nem, u)$: $nem \in NEM$, $u \in U$ and *nem* "has an employee" *u*}.
- *R-VER-A*: a many-to-many binary assignment is represented by $R_{r\text{-}ver\text{-}a} = \{(r, ver)$: $r \in R$, $ver \in VER$ and *r* "is assigned to play" *ver*}.
- *U-VER-A*: a many-to-many binary assignment is represented by $R_{u\text{-}ver\text{-}a} = \{(u, ver)$: $u \in U$, $ver \in VER$ and *u* "is assigned to play" *ver*}.

2.2.3. Project relations

A Project Relation (R_{p}) describes the interactions, cooperation modes and priority between two projects, and determines the level of resource sharing among them. Different project relations may exist between two projects, and project relations may change with time based on project management and sharing requirements. To introduce the project relations, given a set Project (*P*) and *x*, $y \in P$, a binary relation Project Relation (R_{p}) on *P* is a subset of $P \times P$. The project relation is split into five sub-relations:

- *Subset* Relation (R_{ps}) describes a project "main-project", which is decomposed into several projects "sub-projects" to be executed by different VEs. A main-project is permitted to access the resources of its sub-project, but an administrator may set or disable this capability. The subset relation is denoted by $R_{\mathrm{ps}} = \{(x, y)$: $x, y \in P$, $x \neq y$ and *x* "is a subset of" *y*}.
- *Version* Relation (R_{pv}) describes a project *y* called the "post-version project", which is extended from a project *x* called "pre-version project", and which is planned with reference to the pre-version project. Hence, the pre- and post-version projects have similar targets, functional tasks and participants. The version relation may result in correspondences between functional tasks of the two projects. The relation is represented by $R_{\mathrm{pv}} = \{(x, y)$: $x, y \in P$, $x \neq y$ and *x* "is the pre-version of" *y*}.
- *Reference* Relation (R_{pr}) describes a project *x*, called the "referring project", referring to the resources in another project *y*, called the "referred project". If the reference relation exists between two projects, then users in the referring project can refer to the resources of the referred project. The functional task involved in the referred project is allowed to be referred as long as the value of its attribute "allowed-reference" is "true". The relation is given by $R_{\mathrm{pr}} = \{(x, y)$: $x, y \in P$, $x \neq y$, *x* "refers to resources in" *y* and $(\neg \exists x R_{\mathrm{pe}} y) \wedge (\neg \exists y R_{\mathrm{pe}} x)\}$.
- *Process* Relation (R_{pp}) indicates the execution sequence of two sub-projects, and determines the time for sharing project resources. When a project is split into several sub-projects, the process relation can be adopted to indicate the executive sequence of all sub-projects. While the relation is constructed on two projects, the administrator must specify the sequences of related functional tasks across project boundaries. The relation is represented by $R_{\mathrm{pp}} = \{(x, y)$: $x, y, z \in P$, $x \neq y \neq z$, $(\exists x R_{\mathrm{ps}} z) \wedge (\exists y R_{\mathrm{ps}} z)$, and *x* "must be achieved, then start" *y*}.

- *Exclusive* Relation (R_{pe}) denotes that two projects are mutually conflicting, indicating that the resources of the two projects cannot be referred to by each other. The relation is represented by $R_{pe} = \{(x, y)$: $x, y \in P$, $x \neq y$, x "conflicts with" y, and $(\neg\exists x R_{pr} y) \wedge (\neg\exists y R_{pr} x)\}$.

2.2.4. Cooperation modes between two VERs

This sub-section presents three cooperation modes among VERS according to the resource sharing requirements for collaborative operations in the VE.

Cooperation Mode (R_c) describes interactions among VERs based on the dependent level of their duties. Given a set VER, x and $y \in VER$, a binary relation Cooperation Relation (xR_cy) on VER is a subset of $VER \times VER$, which is differentiated into three cooperation relations. For convenience in the following discussion, two items are first defined in terms of authority inheritance. According to the cooperative mode, a VER may inherit strongly or weakly the privileges from the other VERs. The *strong inheritance* indicates that the privileges of a VER can be completely inherited by the other VERs, while the *weak inheritance* means that the privileges can only be partially inherited, i.e. only some privileges of a VER are inherited.

- *Dependent Single-task Mode* ($xR_{cds}y$): The dependent single-task mode is a binary relation and represented by $R_{cds} = \{(x, y)$: $x, y \in VER$, $x \neq y$, $\exists(x, ft_1), (y, ft_1) \in VER\text{-}FT\text{-}A \rightarrow FT\text{-}PublicP\text{-}A\_public\_permission$ $(\{VER\text{-}FT\text{-}A\_functional\_task(x)$: $(x, ft) \in VER\text{-}FT\text{-}A\})$ are inherited strongly by virtual enterprise role y, and $FT\text{-}PublicP\text{-}A\_public\_permission(\{VER\text{-}FT\text{-}A\_functional\_task(y)$: $(y, ft) \in VER\text{-}FT\text{-}A\})$ are inherited strongly by virtual enterprise role x, and $(\neg\exists x R_{cdm} y) \wedge (\neg\exists y R_{cdm} x) \wedge (\neg\exists x R_{ci} y) \wedge (\neg\exists y R_{ci} x)\}$ means that VER x and y cooperate to perform a functional task ft_1, and they have the same access privilege to all its resources.
- *Dependent Multi-task Mode* ($xR_{cdm}y$): The dependent multi-task mode is a binary relation and represented by $R_{cdm} = \{(x, y)$: $x, y \in VER$, $x \neq y$, $\forall(x, ft_x), (y, ft_y) \in VER\text{-}FT\text{-}A \rightarrow FT\text{-}PublicP\text{-}A\_public\_permission(\{VER\text{-}FT\text{-}A\_functional\_task(x)$: $(x, ft_x) \in VER\text{-}FT\text{-}A\})$ are inherited weakly by virtual enterprise role y, and $FT\text{-}PublicP\text{-}A\_public\_permission(\{VER\text{-}FT\text{-}A\_functional\_task(y)$: $(y, ft_y) \in VER\text{-}FT\text{-}A\})$ are inherited weakly by virtual enterprise role x, and $(\neg\exists x R_{cds} y) \wedge (\neg\exists y R_{cds} x) \wedge (\neg\exists x R_{ci} y) \wedge (\neg\exists y R_{ci} x)\}$ means that VER x and y perform related functional tasks separately and outputs of the functional tasks are referred to each other.
- *Independent Mode* ($xR_{ci}y$): The independent mode is a binary relation and represented by $R_{ci} = \{(x, y)$: $x, y \in VER$, $x \neq y$, $FT\text{-}PublicP\text{-}A\_public\_permission(\{VER\text{-}FT\text{-}A\_functional\_task(x)$: $(x, ft_x) \in VER\text{-}FT\text{-}A\})$ are not inherited by virtual enterprise role y, and $FT\text{-}PublicP\text{-}A\_public\_permission(\{VER\text{-}FT\text{-}A\_functional\_task(y)$: $(y, ft_y) \in VER\text{-}FT\text{-}A\})$ are not inherited by virtual enterprise role x, and $(\neg\exists x R_{cds} y) \wedge (\neg\exists y R_{cds} x) \wedge (\neg\exists x R_{cdm} y) \wedge (\neg\exists y R_{cdm} x)\}$ means that VER x and y perform independent functional tasks separately, disregarding their outputs. If two virtual enterprise roles work in an independent mode, they may not have each other's access privileges for functional tasks performed by them.

2.2.5. Properties of relations

To avoid security problems caused by privilege expansion resulting from element relations, and to strengthen private and public resource security, three binary relation properties—reflexive, symmetric and transitive—are applied to the above relations. In a project formation stage, enterprise members in a VE determine whether each cooperation mode and project relation complies with these three properties. Each enterprise member can then identify these three properties based on its own resource sharing rules. The enterprise can also set the *depth* of the transitive property, and require symmetric and transitive properties to be valid only in the same *department*.

2.3. Role relation net (RRN)

Fig. 2 shows an RRN, which is an applied example of the VEAC model. An RRN comprises the basic elements and relations defined in Section 2, which identify the interactive relations among projects, cooperation modes, roles and hierarchical relations in enterprise members, assignment relations between users and roles, and relations between roles and VE roles. In the RRN, through project relations to facilitate the resource sharing across projects, cooperation modes among VERs to enhance the information transparency of a VE, and roles and hierarchical relations to simplify assignment of privileges, users can be assigned proper privileges within a time frame based on roles played by users and VERs used by roles. Section 4.3 illustrates the proposed trust evaluation method using RRN as an example.

3. Scenario for authentication and authorization in VE

The IT environments of large, distributed VEs generally consist of various platforms and applications. Subjects can access various resources deployed on different platforms. Two fundamental access control functions, authentication and authorization, and other related access control activities are shown in Fig. 3 and introduced below:

- Constructing the VEAC model: when a VE is organized, all enterprise members in the VE need to plan the VE objectives, processes, schedules and resources collaboratively. Administrators in this stage must construct the VEAC model, including the design of all elements and the assignments among elements, to enable resource sharing and reuse. Consequently, a VEAC specification is produced from the constructed VEAC model.

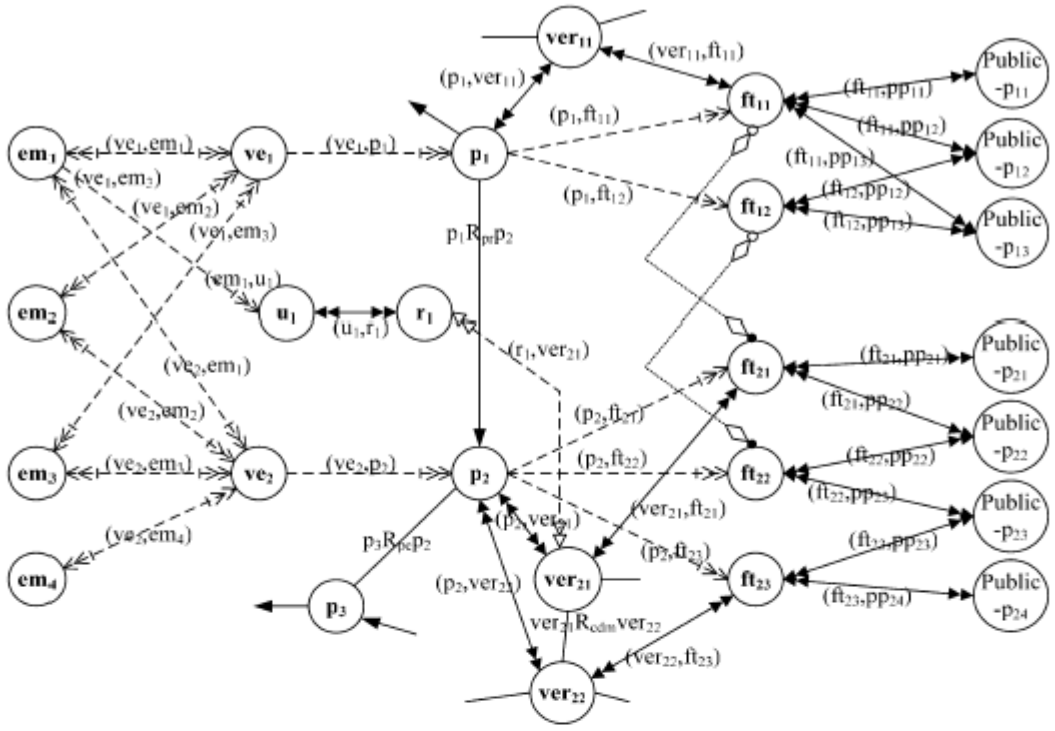

Fig. 2. Part of an RRN.

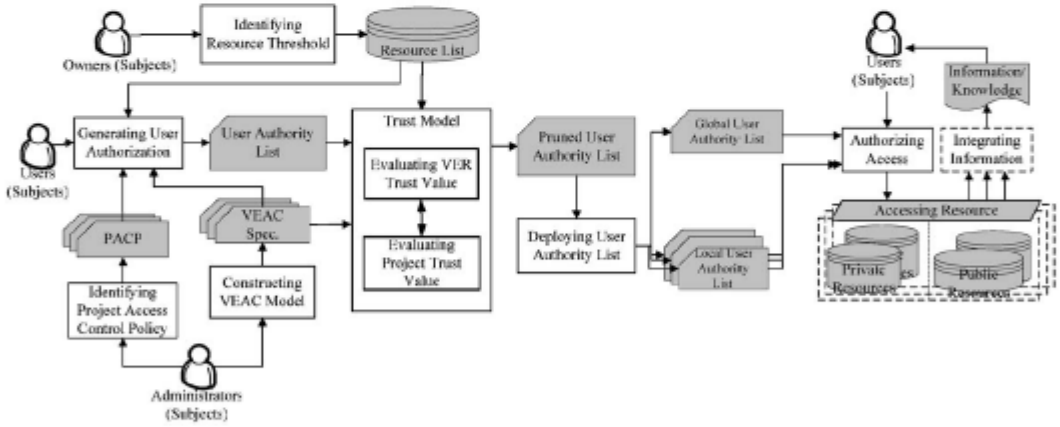

Fig. 3. Access control framework in a VE.

- Identifying the project access control policy: when a VE is formed, a PACP should be identified based on the regulations of the VE for resource usage and sharing.
- Determining the resource threshold: the owner of each resource can set or change the resource threshold according to the secure requirement of dynamic business environment. Each resource involves both the VER and project thresholds, which are recorded in the resource list.
- Generating user authorization: when a user logs into the VEAC system, the user authorization list is generated from private and public authorization algorithms which analyzes the PACP, VEAC specification and resource list. The trust evaluation method is then applied to assess the trust values for the VER and project. Based on these trust values, the system then prunes the user authorization list of trust values that are lower than the threshold of a resource. Finally, the pruned user authorization list is split into local user authorization lists, which are deployed on each enterprise member's access control mechanism.
- Controlling access: when a user successfully logs in, and the user authorization list is generated and deployed, the user can request access to the private resources stored in

all enterprise members and the public resources stored in the VE based on the user authorization list.

4. Trust evaluation method

This section refines and redefines the concept of direct and indirect trusts presented in some other studies, and proposes the concept of a negative trust to improve the level of trust, thus enhancing the match among the requirements of practical VE environments. The direct and indirect trust values are defined as the *positive interrelated coefficient*, which intensifies the level of trust between two VERs, while the negative trust value is defined as the *negative interrelated coefficient*, which enables the sub-models to decrease the level of trust between two VERs. This section develops a trust evaluation method from the subject interaction perspective, which is based on the VEAC model, and which expands the concept of direct and indirect trusts. The trust evaluation method is used to measure the level of belief or disbelief among two subjects (VER and project) for resolving the trust issues resulting from unclear assignment among elements and secure resource sharing across enterprise and project boundaries. This section describes various trust functions based on, (1) cooperation modes between two VERs or project relations between two projects, (2) dependence on responsibilities between two subjects, (3) the intersectional ratio of resources used in performing two functional tasks, and (4) the intersectional ratio of enterprise members participating in two projects. Fig. 4 illustrates the structure and significant features of the trust method containing two sub-models. The details are introduced as follows:

(1) *Trust evaluation sub-model for VER* is adopted to assess the trust level from one VER to another. The trust evaluation sub-model for a VER comprises a direct trust function, indirect trust functions at different depths and a negative trust function, as follows: (a) the *direct trust* function is calculated from the intersection ratio of the functional task assignments based on the cooperative mode between two VERs; (b) the *indirect trust* functions are determined from the direct trust function from one VER to another via the others (third-VERs); and (c) the *negative trust* function is obtained by considering the mutual relationships among the trustee, trusted and their third-VERs, based on the modes of cooperation among them.

(2) *Trust evaluation sub-model for projects* is employed to determine the trust level from the perspective of a particular project to another. Its value is obtained from various project relations and the resource assignment. The trust evaluation sub-model for a project also uses direct, indirect and negative trust functions to determine the trust value between two projects. The direct trust function of a project is calculated by combining the version, subset, reference and process direct trust values with an exclusive direct trust value. The concepts of development of the indirect and negative trust function for projects resemble the indirect and negative trust functions of the trust sub-model of a VER.

4.1. Trust evaluation sub-model for VER

This sub-section describes the trust evaluation sub-model for a VER, including a direct trust function, indirect trust functions at different depth, a negative trust function and a trust function.

4.1.1. Trust evaluation functions for VER

The part of a RRN displayed in Fig. 5 includes several VERs and cooperative modes linking VERs, denoting the direct and indirect trusts for a VER. As demonstrated in Fig. 5, the solid line between two VERs is the direct trust, and the dashed line between two VERs represents the indirect trust. The rules of cooperation between VERs allow only one direct trust between two VERs. However, the indirect trust value can exceed 1 when the transitive depth of the cooperation mode exceeds 1. The three trust classes are defined as follows:

- *Direct Trust* from ver_i to ver_j, $DT_{ver}(ver_i, ver_j)$, is defined as the level of trustworthiness of ver_j for ver_i, i.e., the

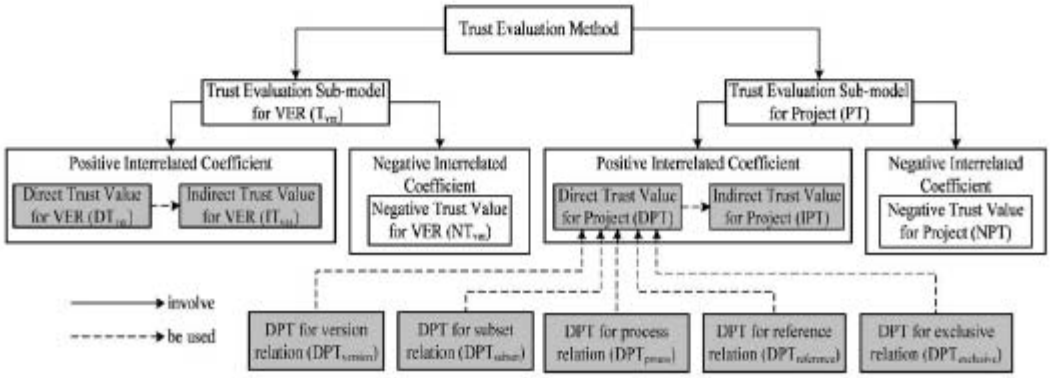

Fig. 4. Structure of the trust evaluation method.

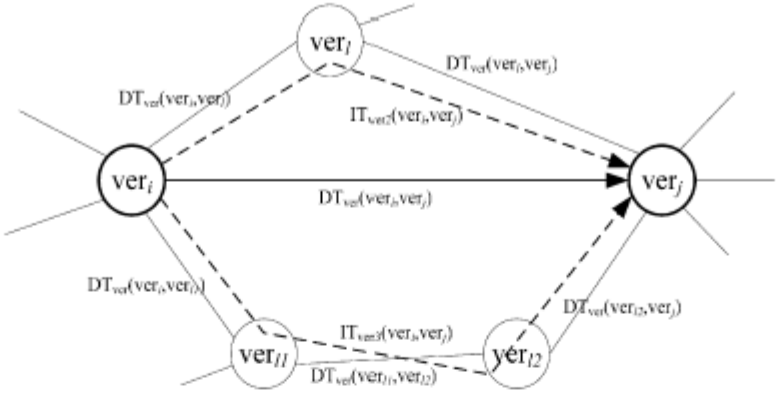

Fig. 5. Part of RRN denoting the direct and indirect trusts for VER.

level to which the trusted subject (ver_j) is believed by the trustee subject (ver_i). The two subjects (VER) are regarded as nodes, and the cooperative mode linking a trusted subject with a trustee subject is treated as an edge with a trust degree. The risk of accessing an unauthorized resource via different cooperative modes between the trusted and trustee VERs might depend on the level of dependence upon the responsibilities assigned to the two VERs and their cooperation mode. Function (1) shows the direct trust function. Therefore, one of the three cooperative modes can be adopted to lead the trust value in the range [0, 1].

$$DT_{ver}(ver_i, ver_j) = \begin{cases} 1 & \text{if } R_c = R_{cds}, \\ |FT_i \cap FT_j|/\text{Min}\{|FT_i|, |FT_j|\} & \text{if } R_c = R_{cdm}, \\ 0 & \text{if } R_c = R_{ci}, \end{cases} \quad (1)$$

where $DT_{ver}(ver_i,ver_j)$ is the direct trust of ver_j for ver_i; ver_i the trustee VER; ver_j the trusted VER; R_c the cooperative mode including R_{cds}, R_{cdm} and R_{ci}; R_{cds} the dependent single-task cooperative mode; R_{cdm} the dependent multi-task cooperative mode; R_{ci} the independent cooperative mode; FT_i and FT_j the functional tasks performed by virtual enterprise role ver_i and ver_j, respectively; $|FT_i|$ and $|FT_j|$ the numbers of functional tasks assigned to ver_i and ver_j, respectively; and $|FT_i \cap FT_j|$ the number of functional tasks assigned simultaneously to both ver_i and ver_j.

- *Indirect Trust* from ver_i to ver_j, $IT_{ver}(ver_i, ver_j)$, is expressed as the level of trustworthiness of ver_j for ver_i via third-virtual enterprise roles (third-VERs) that interact with ver_i, ver_j or both, such as ver_l, ver_{I1} and ver_{I2} in Fig. 5. The indirect trust can be considered as a path composed of edges connecting ver_i with ver_j via different third-VERs. Hence, the indirect trust can involve zero or more paths from a trustee subject to a trusted subject, where the number of the paths is determined from the number of the third-VERs that can cooperate directly with atleast one of the two subjects. The indirect trust function is derived from the direct trust function by considering all edges of a path from the trustee subject to the trusted subject. When the transitive property of the cooperation mode is available and its depth equals 2, the indirect trust function at depth 2 is defined as function (2), which utilizes the product of two direct trust functions. The total number of multiple indirect trusts at depth 2 is then averaged to keep IT_{ver} in the range [0, 1].

$$IT_{ver_2}(ver_i, ver_j) = \frac{\sum_{l=1}^{k_2}[DT_{ver}(ver_i, verI_l) \times DT_{ver}(verI_l, ver_j)]}{k_2}, \quad (2)$$

where $IT_{ver_2}(ver_i, ver_j)$ is the indirect trust of ver_j for ver_i at depth 2; $DT_{ver}(ver_i,ver_j)$ the direct trust of ver_j for ver_i; $verI_l$ the third-virtual enterprise role (third-VER) that cooperates with ver_i and ver_j simultaneously; and k_2 the number of third-VERs that cooperate with ver_i and ver_j simultaneously, i.e., the number of paths from ver_i to ver_j via $verI_l$ while depth equals 2, $1 \leqslant l \leqslant k_2$.

The indirect trust functions at depth 3 and beyond can be obtained from function (2). The indirect trust function at depth 3 is represented in function (3).

$$IT_{ver_3}(ver_i, ver_j) = \frac{\sum_{d=1}^{k_3}[DT_{ver}(ver_i, verI_{l1}) \times DT_{ver}(verI_{l1}, verI_{l2}) \times DT_{ver}(verI_{l2}, ver_j)]}{k_3}, \quad (3)$$

where $IT_{ver_3}(ver_i, ver_j)$ is the indirect trust of ver_j for ver_i at depth 3; $verI_{l1}$ the third VERs that directly cooperate with ver_i and $verI_{l2}$; $verI_{l2}$ the third-VERs that directly cooperate with ver_j and $verI_{l1}$, and k_3 the number of paths from ver_i to ver_j at depth 3.

Finally, the indirect trust function is denoted in function (4), which must be limited by Eq. (5) in which the weighted factors for indirect trust at various depths

are determined by the administrator, and the sum of all the weighted factors must equal 1.

$$IT_{ver}(ver_i, ver_j) = \sum_{w=2}^{\text{max-depth}} \alpha_w \times IT_{ver_w}, \tag{4}$$

$$\sum_{w=2}^{\text{max-depth}} \alpha_w = 1, \tag{5}$$

where $IT_{ver}(ver_i,ver_j)$ is the total indirect trust value of ver_j for ver_i; α_w the trust weighted factor for indirect trust value at depth w, $2 \leqslant w \leqslant$ max-depth; and max-depth the maximal depth of available transitive property.

- *Negative Trust* from ver_i to ver_j, $NT_{ver}(ver_i, ver_j)$, is defined as the level of untrustworthiness of ver_j for ver_i, and is adopted to decrease the level of trust between ver_i and ver_j. Fig. 6 shows the part of RRN denoting the negative trust for the VER. The negative trust function defined in function (6) rises when the trusted and trustee subjects cooperate with third-VERs using different cooperation modes. All third-VERs may be categorized into three groups. The numbers of the three third-VERs are represented by variables k, n and p, which are defined in function (6). Consequently, the negative trust is in the range [0, 1].

$$NT_{ver}(ver_i, ver_j) = \frac{\sum_{m=1}^{n} DT_{ver}(ver_j, ver_{lm})}{(k+n-p)}, \tag{6}$$

where $NT_{ver}(ver_i,ver_j)$ is the negative trust of ver_j for ver_i; k the number of third-VERs cooperating with ver_i and ver_j simultaneously, i.e., the number of indirect trust values from ver_i to ver_j; n the number of third-VERs that cooperate with ver_j with either cooperation modes R_{cds} or R_{cdm} and without ver_i; p the number of third-VERs that cooperate with ver_j via the cooperation mode R_{ci} and without ver_i; and ver_{lm} the third-VERs that cooperate with ver_j with either cooperation modes R_{cds} or R_{cdm} and without ver_i.

In contrast to variable p in function (6), variables k and n enable the negative trust value to raise the trust level for the VER.

The trust function for VER as displayed in function (7) is obtained by combining direct trust (DT_{ver}), indirect trust (IDT_{ver}) and negative trust (NT_{ver}), in which Eq. (8) should suffice irrespective of how the weighted factors (C_{D1}, C_{I1} and C_{N1}) are set. The three weighted factors are determined by project administrators based on the influences of the direct, indirect and negative trusts on the trust evaluation sub-model for VER. Intuitively, if a trust value contributes more in terms of data value, it should be weighted more in the trust value calculation. Each resource in a VE involves both a VE role threshold and a project threshold (refer to Section 2), which can be frequently adjusted by the resource owner to adapt to the requirement of the virtual enterprise environment for resources sharing. When these three coefficients are altered, function (7) can provide an adequate secure information sharing method. The trust value for VER (T_{ver}) is in the range [−1, 2] under the limitations of Eq. (8). The secure threshold of each resource is high when T_{ver} approaches 2, and is low when T_{ver} approaches −1.

$$T_{ver}(ver_i, ver_j) = C_{D1} DT_{ver}(ver_i, ver_j) + C_{I1} IT_{ver}(ver_i, ver_j) - C_{N1} NT_{ver}(ver_i, ver_j), \tag{7}$$

$$C_{D1}, C_{I1} \text{ and } C_{N1} = [0, 1], \tag{8}$$

where C_{D1} is the trust weighted factor for the direct trust; C_{I1} the trust weighted factor for the indirect trust, and C_{N1} the trust weighted factor for the negative trust.

4.1.2. Example of assessing trust value for VER

Fig. 7 shows the VERs and relations as an example of the trust evaluation sub-model for the VER. The example includes nine VERs ver_1, ver_2,…, ver_9, and specifically indicates some direct trusts, which can be used to assess the indirect and negative trusts, and thus obtain the trust value for the VER of ver_2 for ver_1 ($T_{ver}(ver_1, ver_2)$).

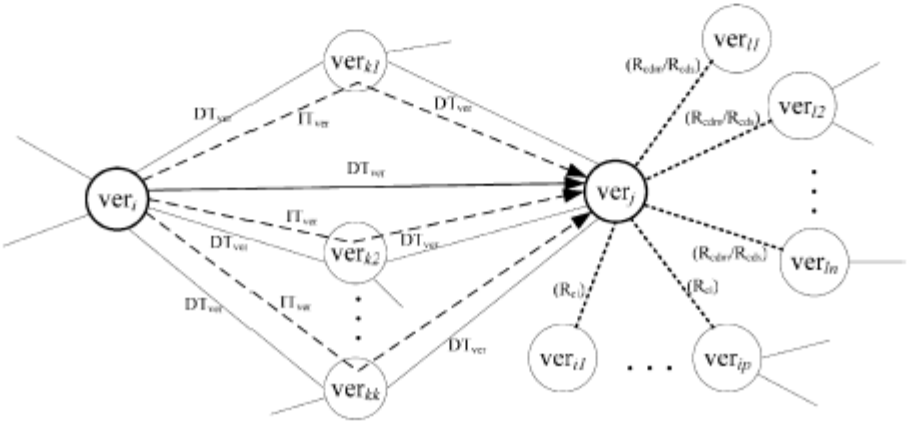

Fig. 6. Part of RRN representing the negative trust for VER.

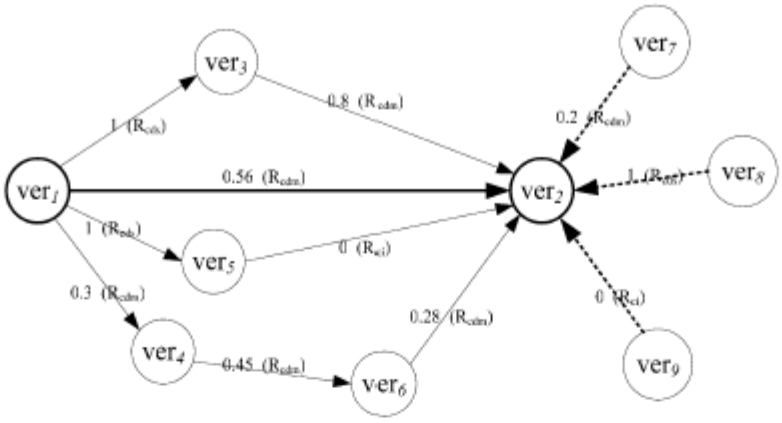

Fig. 7. Example of assessing trust value for VER.

From Fig. 7, the following is obtained:

$DT_{ver}(ver_1, ver_2) = 0.56.$

Using Functions 2 and 3 yields

$$IT_{ver_2}(ver_1, ver_2) = \frac{1 \times 0.8 + 1 \times 0}{2} = 0.4,$$

$$IT_{ver_3}(ver_1, ver_2) = \frac{0.3 \times 0.45 \times 0.28}{1} = 0.0378.$$

Assume that $\alpha_2 = 0.75$ and $\alpha_3 = 0.25$. Using function (4) yields

$IT_{ver}(ver_1, ver_2) = 0.75 \times 0.4 + 0.25 \times 0.0378 = 0.30945.$

From Fig. 7 and these relations among the VERs, we can infer that $k = 3$ (including ver_3, ver_5 and ver_{4-6}), $n = 2$ (including ver_7 and ver_8) and $p = 1$ (including ver_9).

Substituting k, n and p into function (6) yields

$$NT_{ver}(ver_1, ver_2) = \frac{0.2 + 1}{3 + 2\text{-}1} = 0.3.$$

Based on the secure threshold of resource, set $C_{D1} = 0.7$, $C_{I1} = 0.3$ and $C_{N1} = 0.5$.

Function (7) yields

$$T_{ver}(ver_1, ver_2) = 0.7 \times 0.56 + 0.3 \times 0.30945 - 0.5 \times 0.3 = 0.334835.$$

The above mathematical manipulations yield $T_{ver}(ver_1, ver_2) = 0.334835$. Considering the resource sharing among VERs in a project, ver_1 is authorized to access the resource owned by ver_2, while the resource threshold is equal to or below the calculated trust value for the VER.

4.2. Trust evaluation sub-model for project

The project relations defined in Section 2.2.3 can specifically indicate the operation mode of interaction among projects, enabling project resources to be shared or reused during the project lifecycle. Consequently, security for project resources is vital to project success. This sub-section describes a trust evaluation sub-model, resolving the difficulty of indefinite assignments across project boundaries.

4.2.1. Trust evaluation functions for project

This sub-section initially defines terms concerning the trust evaluation sub-model for projects, and then presents some functions for assessing the level of trust of each project relation from one project to the others. As with the VER trust evaluation sub-model, three trust values are considered, defined as follows.

(1) *Direct Trust* from project p_i to p_j, $DPT(p_i,p_j)$, is defined as the level of trustworthiness of p_j for p_i (see Fig. 8) and is calculated from the project relations between p_i and p_j. The DPT is a positive correlation coefficient increasing the trust intensity with its increased value. The solid line between two projects in Fig. 8 denotes the direct trust for a project. Since various project relations enable different levels of resource sharing, the DPT is written as function (9), which comprises five direct trust values for version, subset, process, reference and exclusive project relations, where the direct trust for exclusive project relation acts as a key gate for determining whether the DPT is 0 or greater than 0. Hence, function (9) and the five direct trust functions defined in this sub-section clearly indicate that the direct trust function for project is in the range [0, 1].

$$DPT(p_i,p_j) = [(DPT_{version}(p_i,p_j) + DPT_{subset}(p_i,p_j) + DPT_{reference}(p_i,p_j) + DPT_{process}(p_i,p_j))/4] \times DPT_{exclusive}(p_i,p_j), \quad (9)$$

where $DPT(p_i,p_j)$ is the direct trust of p_j for p_i; and $DPT_{version}(p_i,p_j)$, $DPT_{subset}(p_i,p_j)$, $DPT_{reference}(p_i,p_j)$, $DPT_{process}(p_i,p_j)$ and $DPT_{exclusive}(p_i,p_j)$ separately denote the direct trust of p_j for p_i at version, subset, reference, process and exclusive relations.

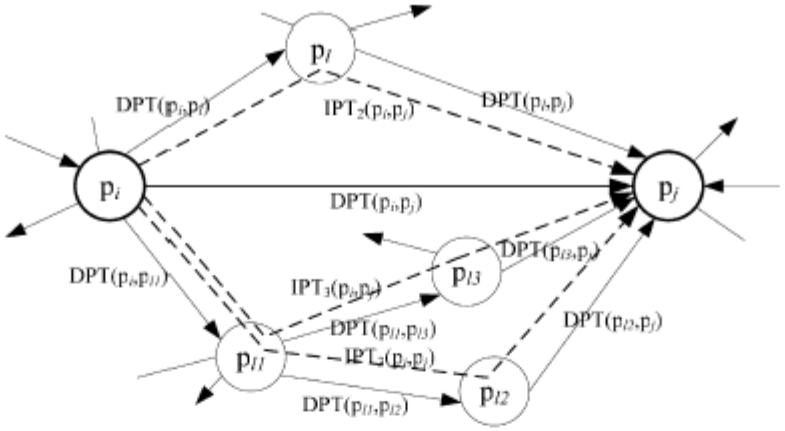

Fig. 8. Part of RRN presenting the direct and indirect trusts for project.

The five direct trust functions with various project relations are described in order, as follows:

- *Direct trust* function *for version project* relation from project p_i to p_j, $DPT_{\text{version}}(p_i,p_j)$, measures the trustworthiness intensity of project p_j for project p_i when considering the version project relation. The risk of accessing an unauthorized resource via the version relation might depend on the intersection of enterprise members from the two projects. Based on the above principle, the direct trust function for the version project relation is derived as function (10), which is in the range [0, 1].

$$DPT_{\text{version}}(p_i,p_j)=\begin{cases}\dfrac{|FT_{jk}\text{ corresponding to }FT_{jk}|}{\text{Min}\{|FT_i|,|FT_j|\}} & \text{if } \exists(p_iR_{\text{pv}}p_j) \text{ and } \exists(EM_{im}\neq EM_{jn}),\\ 1 & \text{if } \exists(p_iR_{\text{pv}}p_j) \text{ and } \forall(EM_{im}=EM_{jn}),\\ 0 & \text{otherwise,}\end{cases} \tag{10}$$

where p_i is the trustee project; p_j the trusted project; $DPT_{\text{version}}(p_i,p_j)$ the direct trust of p_j for p_i at version project relation; FT_i the function tasks involved in p_i; FT_j the function tasks involved in p_j; $|FT_i|$ the number of function tasks involved in p_i; $|FT_j|$ the number of function tasks involved in p_j; $|FT_{jk}$ corresponding to $FT_{jk}|$ the number of functional tasks assigned to p_i and linked to the functional tasks assigned to p_j via correspondence relations; EM_{im} the enterprise members participating in project p_i; EM_{jn} the enterprise members participating in project p_j; and $p_iR_{\text{pv}}p_j$ the version project relation between p_i and p_j.

- *Direct trust* function *for subset project* relation from project p_i to p_j, $DPT_{\text{subset}}(p_i,p_j)$, measures the trustworthiness of project p_j for p_i when considering a subset project relation. The risk of accessing an unauthorized resource through a subset relation might depend on the amount of resources used by projects p_i and p_j. Therefore, the direct trust function for subset project relation can be obtained as function (11), which is in the range [0, 1].

$$DPT_{\text{subset}}(p_i,p_j)=\begin{cases}\dfrac{|PublicP_j\cap PublicP_j|}{\text{Min}\{|PublicP_j|,|PublicP_j|\}} & \text{if } \exists p_iR_{\text{ps}}p_j,\\ 0 & \text{otherwise,}\end{cases} \tag{11}$$

where $DPT_{\text{subset}}(p_i,p_j)$ is the direct trust of p_j for p_i at subset project relation; $PublicP_i$ the public resources assigned to projects p_i; $PublicP_j$ the public resources assigned to projects p_j; $|PublicP_i|$ the number of public resources assigned to projects p_i; $|PublicP_j|$ the number of public resources assigned to projects p_j; $|PublicP_i\cap PublicP_j|$ the number of public resources assigned simultaneously to projects p_i and p_j; and $p_iR_{\text{ps}}p_j$ the subset project relation between p_i and p_j.

- *Direct trust* function *for reference project* relation from project p_i to p_j, $DPT_{\text{reference}}(p_i,p_j)$, measures the trust intensity of project p_j for p_i when addressing the reference project relation. The risk of accessing an unauthorized resource using a reference relation is based on the enterprise members participating in the two projects or in other projects. Consequently, the direct trust function for the reference project relation can be determined as function (12), which is in the range [0, 1].

$$DPT_{\text{reference}}(p_i,p_j)=\begin{cases}\dfrac{|EM_j\cap EM_j|}{\text{Min}\{|EM_j|,|EM_j|\}} & \text{if } \exists p_iR_{\text{pr}}p_j,\\ 0 & \text{otherwise,}\end{cases} \tag{12}$$

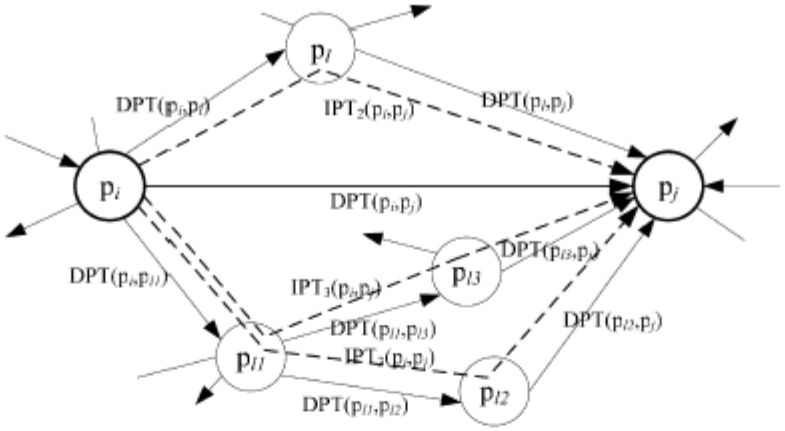

Fig. 8. Part of RRN presenting the direct and indirect trusts for project.

The five direct trust functions with various project relations are described in order, as follows:

- *Direct trust* function *for version project* relation from project p_i to p_j, $DPT_{\text{version}}(p_i,p_j)$, measures the trustworthiness intensity of project p_j for project p_i when considering the version project relation. The risk of accessing an unauthorized resource via the version relation might depend on the intersection of enterprise members from the two projects. Based on the above principle, the direct trust function for the version project relation is derived as function (10), which is in the range [0, 1].

$$DPT_{\text{version}}(p_i,p_j)=\begin{cases}\dfrac{|FT_{jk}\text{ corresponding to }FT_{jk}|}{\text{Min}\{|FT_i|,|FT_j|\}} & \text{if } \exists(p_iR_{\text{pv}}p_j) \text{ and } \exists(EM_{im}\neq EM_{jn}),\\ 1 & \text{if } \exists(p_iR_{\text{pv}}p_j) \text{ and } \forall(EM_{im}=EM_{jn}),\\ 0 & \text{otherwise,}\end{cases} \tag{10}$$

where p_i is the trustee project; p_j the trusted project; $DPT_{\text{version}}(p_i,p_j)$ the direct trust of p_j for p_i at version project relation; FT_i the function tasks involved in p_i; FT_j the function tasks involved in p_j; $|FT_i|$ the number of function tasks involved in p_i; $|FT_j|$ the number of function tasks involved in p_j; $|FT_{jk}$ corresponding to $FT_{jk}|$ the number of functional tasks assigned to p_i and linked to the functional tasks assigned to p_j via correspondence relations; EM_{im} the enterprise members participating in project p_i; EM_{jn} the enterprise members participating in project p_j; and $p_iR_{\text{pv}}p_j$ the version project relation between p_i and p_j.

- *Direct trust* function *for subset project* relation from project p_i to p_j, $DPT_{\text{subset}}(p_i,p_j)$, measures the trustworthiness of project p_j for p_i when considering a subset project relation. The risk of accessing an unauthorized resource through a subset relation might depend on the amount of resources used by projects p_i and p_j. Therefore, the direct trust function for subset project relation can be obtained as function (11), which is in the range [0, 1].

$$DPT_{\text{subset}}(p_i,p_j)=\begin{cases}\dfrac{|PublicP_j\cap PublicP_j|}{\text{Min}\{|PublicP_j|,|PublicP_j|\}} & \text{if } \exists p_iR_{\text{ps}}p_j,\\ 0 & \text{otherwise,}\end{cases} \tag{11}$$

where $DPT_{\text{subset}}(p_i,p_j)$ is the direct trust of p_j for p_i at subset project relation; $PublicP_i$ the public resources assigned to projects p_i; $PublicP_j$ the public resources assigned to projects p_j; $|PublicP_i|$ the number of public resources assigned to projects p_i; $|PublicP_j|$ the number of public resources assigned to projects p_j; $|PublicP_i\cap PublicP_j|$ the number of public resources assigned simultaneously to projects p_i and p_j; and $p_iR_{\text{ps}}p_j$ the subset project relation between p_i and p_j.

- *Direct trust* function *for reference project* relation from project p_i to p_j, $DPT_{\text{reference}}(p_i,p_j)$, measures the trust intensity of project p_j for p_i when addressing the reference project relation. The risk of accessing an unauthorized resource using a reference relation is based on the enterprise members participating in the two projects or in other projects. Consequently, the direct trust function for the reference project relation can be determined as function (12), which is in the range [0, 1].

$$DPT_{\text{reference}}(p_i,p_j)=\begin{cases}\dfrac{|EM_j\cap EM_j|}{\text{Min}\{|EM_j|,|EM_j|\}} & \text{if } \exists p_iR_{\text{pr}}p_j,\\ 0 & \text{otherwise,}\end{cases} \tag{12}$$

(3) *Negative Project Trust* Function from project p_i to p_j, $NPT(p_i,p_j)$, is a negative correction coefficient in the range 0 and 1, and is applied to reduce the project trust intensity. Function (18) shows the negative trust function for project.

$$NPT(p_i,p_j)=\frac{\sum_{m=1}^{n} DPT(pI_m,p_j)}{k+n-\mathrm{p}}, \tag{18}$$

where k is the number of third-projects with project relations with projects p_i and p_j, simultaneously, and the number of indirect trust values from p_i to p_j (such as projects p_{k1}, $p_{k2},\ldots,$ p_{kk}); n the number of third-projects which have subset, version, reference or process project relations with project p_j (such as projects $p_{m1}, p_{m2},\ldots, p_{mn}$), and p the number of third-projects which have an exclusive project relation with project p_j (such as projects $p_{n1}, p_{n2},\ldots,$ p_{np}).

Considering DPT, IPT and NPT, this study presents the trust function for projects as shown in function (19), where C_{D2}, C_{I2} and C_{N2} denote three real coefficients used as weighted factors that can be restricted with Eq. (20). Different trust values for project are obtained by altering the three coefficients based on the project security policy.

$$PT(p_i,p_j)=C_{D2}\times DPT(p_i,p_j)+C_{I2}\times IPT(p_i,p_j) - C_{N2}\times NPT(p_i,p_j), \tag{19}$$

$$C_{D2},C_{I2} \text{ and } C_{N2}=[0,1]. \tag{20}$$

4.2.2. Example of assessing trust value for project

Fig. 10 shows an example of the project trust model, which considers 10 projects (p_1, $p_2,\ldots,$ p_{10}) and various project relations. This example aims to assess the project trust value of p_2 from the perspective of p_1 ($PT(p_1, p_2)$). To simplify the illustration of the example, some direct trusts for project are assumed as displayed in Fig. 10.

From Fig. 10, function (9) is applied to yield

$$DPT(p_1,p_2)=\frac{0+0.72+0+0.4}{4}\times 1=0.28.$$

Using function (15) yields

$$IPT_2(p_1,p_2)=\frac{0.6\times 0+0.33\times 0.2+0\times 0.4}{3}=0.022.$$

In the example, only indirect trust for project at depth 2 is available, so that the total indirect trust equals the indirect trust for project at depth 2.

Referring to Fig. 10 and calculating all project relations among projects, $k=3$ (including projects p_3, p_4 and p_5), $n=3$ (including projects p_6, p_7 and p_8) and $p=2$ (including projects p_9 and p_{10}).

Substituting these integers into function (18) yields

$$NPT(p_1,p_2)=\frac{0.6+0.45+0.3}{3+3-2}=0.3375.$$

Based on the threshold of project resource security and the restriction on Eq. (20), set $C_{D2}=0.7$, $C_{I2}=0.3$ and $C_{N2}=0.5$.

Substituting these coefficients into function (19) yields

$$PT(p_1,p_2)=0.7\times 0.28+0.3\times 0.022-0.5\times 0.3375 =0.03385.$$

The above mathematical manipulation yields the project trust value of p_2 from the perspective of p_1, $PT(p_1,p_2)=0.03385$. Considering the resource sharing across projects, project p_1 is authorized to access the resources owned by project p_2, while the secure threshold for the resources is equal or less than the project trust value.

4.3. An example of trust evaluations for virtual enterprise role and project

This sub-section uses Fig. 2 as an example to introduce the application of the proposed trust method. Table 1 lists three of all attributes of each functional task in Fig. 2.

Table 2 lists the project and VER thresholds of all public permission (resource).

Table 3 lists the assignments between VER and public permission.

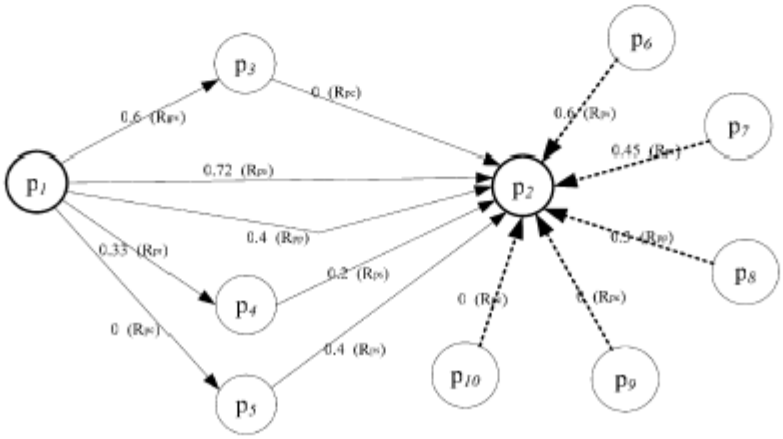

Fig. 10. Example of calculating trust value for project.

Table 1
Attribute list of FTs

FT	Attributes		
	Allowed reference	Allowed sub-project	Allowed main project
ft_{11}	F	F	F
ft_{12}	T	T	F
ft_{21}	T	T	T
ft_{22}	T	T	T
ft_{23}	T	F	F

Table 2
Threshold list of public permission

Public permission (resource)	Threshold of project	Threshold of VER
Public-p_{11}	0.7	0.8
Public-p_{12}	0.62	1
Public-p_{13}	0.2	1
Public-p_{21}	0.1	0.7
Public-p_{22}	0.22	0
Public-p_{23}	0.35	0
Public-p_{24}	0.4	0.6

Table 3
VER public permission assignment list

VER	Public permission (resource)
ver_{11}	*Public*-p_{11}, *Public*-p_{12}, *Public*-p_{13}
ver_{21}	*Public*-p_{21}, *Public*-p_{22}
ver_{22}	*Public*-p_{23}, *Public*-p_{24}

Table 4
VER authorization list after considering sharing and trusts

VER	Public permission (resource)
ver_{11}	*Public*-p_{11}, *Public*-p_{12}, *Public*-p_{13}
ver_{21}	*Public*-p_{21}, *Public*-p_{22}, *Public*-p_{23}
ver_{22}	*Public*-p_{23}, *Public*-p_{24}, *Public*-p_{22}

In the example, some states are set, including attributes, assignments, thresholds and trust for project and virtual enterprise role. Finally, we can decide each subject's authorizations based on trust values. While $PT(p_1,p_2) = 0.35$, $PT(p_2,p_3) = 0.2$, $PT(p_1,p_3) = -1$, $T_{ver}(ver_{21},ver_{22}) = 0.5$ and $T_{ver}(ver_{22},ver_{21}) = 0.6$, the authorizations of each virtual enterprise role are listed in Table 4.

5. Discussion and conclusions

Resource management and sharing in collaborative VE environment will in the future become increasingly complicated because of the need for information transparency. Based on the results of the requirements of resource sharing in VE, this study proposed a VEAC-based trust evaluation method to resolve the issue of trust evaluation for sharing resources across enterprise and project boundaries.

5.1. Results and contributions

The VEAC model can significantly simplify the explicit specifications and administration of access control in VE by specifying the various relations among various elements, while the trust evaluation method provides a secure mechanism for supporting VEAC's need for security and flexibility. The detailed results and contributions of this study are:

(1) The proposed trust evaluation sub-model for VER and the trust evaluation sub-model for project can measure the trust value among various VERs to facilitate the secure resource sharing across organization.
(2) The VEAC-based trust method can solve the drawback from the VEAC model and facilitate more security and flexibility for resource sharing to support cross-organizational collaborative activities in VE.
(3) With the change of each resource threshold, each resource's owner can frequently adjust the security level to adapt to various secure threats.
(4) This study may provide a suitable foundation for building a high-assurance trusted cooperative platform in dynamic virtual teams.

5.2. Further research

To develop a VEAC mechanism for managing and facilitating resource sharing, some investigations need to be performed, and the following factors should be considered in future:

(1) This study only considered two elements of the VEAC model to develop the trust evaluation method; the other elements should be considered in the future.
(2) As well as direct, indirect and negative trust factors, other factors, such as the user's historical data, should be addressed to determine the level of trust and access resources and amount of referral from other trusted entities.
(3) This study does not consider that the user might share a resource with unauthorized users after legally acquiring it.
(4) Methods for the access control server to call and use resources in the heterogeneous platform were not addressed.
(5) An enterprise may participate in several competing VEs. Leaking of professional key technology or data should be prevented.
(6) Future studies may apply the eXtensible Access Control Markup Language (XACML) presented by OASIS to develop project access control policy frameworks to integrate access strategies among enterprises.

(7) VEAC model-based algorithms for generating user authorization are highly promising for use in supporting the VEAC system.

Acknowledgment

This research is financially supported by National Science Council of the Republic of China under Contract nos: NSC94-2524-S-024-002, NSC94-2524-S-006-005 and NSC94-2524-S-006-006.

References

[1] Chen YM, Liang MW. Design and implementation of a collaborative engineering information system for allied concurrent engineering. Int J Comput Integr Manuf 1999;13(1):11–30.

[2] Ouzounis EK. An agent-based platform for the management of dynamic virtual enterprises. Ph.D thesis, 2001.

[3] Park JS, Hwang J. RBAC for collaborative environments: role-based access control for collaborative enterprise in peer-to-peer computing environments. In: Proceedings of the eighth ACM symposium on access control models and technologies, 2003. p. 93–9.

[4] Kanet JJ, Faisst W, Mertens P. Application of information technology to a virtual enterprise broker: the case of Bill Epstein. Int J Prod Econ 1999:23–32.

[5] Stephens B. Security architecture for system wide information management. In: Digital avionics systems conference, 2005.

[6] Charles E, Phillips TC, Ting SAD. Information sharing and security in dynamic coalitions. In: Proceedings of the seventh ACM symposium on access control models and technologies, 2002. p. 87–96.

[7] Zha X, Ding N. Study on information sharing in supply chain. In: Proceedings of the seventh international conference on electronic commerce; 2005. p. 787–9.

[8] Li N, MitChell JC, Winsborough WH. Beyond proof-of-compliance: security analysis in trust management. J ACM 2005;52(3):474–514.

[9] Biba KJ. Integrity considerations for secure computer systems. Bedford, MA: The MITRE Corporation; 1977.

[10] Frenkel A, Afsarmanesh H, Garita C. Hertzberger LO. Supporting information access rights and visibility levels in virtual enterprise. In: IFIP TC5/WG5.3 second IFIP working conference on infrastructures for virtual organizations: managing cooperation in virtual organizations and electronic business towards smart organizations, 2000.

[11] Kern A, Schaad A, Moffett J. Enterprise role administration: an administration concept for the enterprise role-based access control model. In: Proceedings of the eighth ACM symposium on access control models and technologies, 2003. p. 3–11.

[12] Au R, Looi M, Ashley P. Automated cross-organizational trust establishment on extranets. In: Proceedings of workshop on information technology for virtual enterprises, 2001. p. 3–11.

[13] Shand B, Dimmock N, Bacon J. Trust for ubiquitous, transparent collaboration. In: Proceedings of the first IEEE international conference on pervasive computing and communications, 2003. p. 153–60.

[14] Tran H, Hitchens M, Varadharajan V, Watters P. A trust based access control framework for P2P file-sharing systems. In: Proceedings of the 38th Hawaii international conference on system sciences, 2005. p. 302c.

[15] Dimmock N, Belokosztolszki A, Eyers D. Using trust and risk in role-based access control policies. SACMAT 2004:156–62.

[16] Barrett S, Konsynski B. Inter-organization information sharing systems. MIS Quart 1982; 83–105.

[17] Zuo Y, Panda B. Component based trust management in the context of a virtual organization. In:ACM symposium on applied computing, 2005. p. 1582–8.

[18] Ahn, G.J. Specification and classification of role-based authorization policies. In: Twelfth IEEE international workshops, 2003. p. 202–7.

[19] Wang CB, Chen TY, Chen YM, Chu HC, Yang H. Access control requirements and model for resource management and sharing in virtual enterprise. In: Automation conference, 2005.

[20] Al-Kahtani MA, Sandhu R. A model for attribute-based user-role assignment. In: 18th Annual computer security applications conference 2002. p. 353–62.

[21] Botha RA, Eloff JHP. Designing role hierarchies for access control in workflow systems. In: Computer software and applications conference, 2001. p. 117–22.

[22] Dridi F, Muschall B, Pernul G. Administration of an RBAC system. In: Proceedings of the 37th annual Hawaii international conference, 2004. p. 187–92.

[23] Kern A, Schaad A, Moffett J. Enterprise role administration: an administration concept for the enterprise role-based access control model. In: Proceedings of the eighth ACM symposium on access control models and technologies; 2003. p. 3–11.

附件 6

+ Models

ARTICLE IN PRESS

Available online at www.sciencedirect.com

SCIENCE DIRECT

ELSEVIER

Computers in Industry xxx (2006) xxx–xxx

COMPUTERS IN INDUSTRY

www.elsevier.com/locate/compind

Development of an access control model, system architecture and approaches for resource sharing in virtual enterprise

Tsung-Yi Chen [a,b], Yuh-Min Chen [a,*], Hui-Chuan Chu [c], Chin-Bin Wang [d]

[a] Institute of Manufacturing Engineering, National Cheng Kung University, Tainan, Taiwan, ROC
[b] Department of Electronic Commerce Management, Nan Hua University, Chia-Yi, Taiwan, ROC
[c] National University of Tainan, Tainan, Taiwan, ROC
[d] Department of Information Management, Nan Hua University, Chia-Yi, Taiwan, ROC

Received 8 September 2005; accepted 21 April 2006

Abstract

Secure information sharing is one of key factors for success of virtual enterprise (VE). The study identifies the characteristics of a VE and analyzes the requirements of a VE access control. A Virtual Enterprise Access Control (VEAC) model is proposed to handle resource management and sharing across each participating enterprise, which consists of a Project-based Access Control (PBAC) sub-model to manage public resources and a Role-based Access Control (RBAC) sub-model to manage private resources. The architecture of a VEAC model-based system is developed and consists of three core mechanisms including the Virtual Enterprise Access Control Center (VEACC), Security Gatekeeper (SG) and Global Certificate Authority Center (GCAC). Based on the system architecture, the study proposes certificate authentication, user authority and access control approaches to identify user's identity on-line, update and search user authority lists, and access private and public resources. The results of this study will facilitate more secure resource sharing, and overcome cooperation barrier from trust among participating enterprises in VE.

Keywords: Virtual enterprise; Information sharing; RBAC; Access control; Certificate authority

1. Introduction

Virtual enterprise (VE) is a network of independent, geographically dispersed administrative business domains that cooperate by sharing business processes and resources across enterprises to provide a value-added service to customers. VE is treated as one of the most promising business strategies for enterprises to meet global competition [1,2]. VEs integrate the processes, activities and resources from different enterprises through enterprise alliances to rapidly respond to customer expectations. In practice, a VE is implemented with a distributed and collaborative business process, in which individuals from different enterprises cooperate on business-related activities or processes through remote coordination, communication and control [3,4].

Real-time information sharing and resource management within a manufacturing-based company or across companies are essential in the era of internet. For instance, a new automobile model is developed by a virtual enterprise that involves approximately 20,000 designers and engineers from hundreds of divisions and departments, some of which are in different enterprises in different countries. A virtual enterprise can be comprised of several sub-VEs. In the above example, one of sub-VEs in the VE to perform product design involves four sub-projects: engine design, cool system design, transmission case design and framework design. The engineers of engine design sub-project design an engine for the new automobile model collaboratively. Information related to the engine design must be shared real-time to related engineers in the sub-project or other projects. Owing to the decentralized and dynamic characteristics in virtual enterprise environments, the success of a virtual enterprise heavily relies on full information transparency and correct resource sharing, including information, application systems and knowledge throughout the business cycle [4]. Even though the resource sharing leads to security and authority management problems, the issues of information delay and promote information transparency are still required to solve among business

* Corresponding author. Tel.: +886 6 2757575x63922; fax: +886 6 2085334.
E-mail address: ymchen@mail.ncku.edu.tw (Y.-M. Chen).

0166-3615/$ – see front matter © 2006 Elsevier B.V. All rights reserved.
doi:10.1016/j.compind.2006.04.003

partners. The levels of resource sharing depend on characteristics of the VE, such as cooperative relationships with partners, depth of trust, functional tasks and contractual agreements. Access control and sharing for resource is most complicated in a virtual enterprise involving cross-organizational activities. There must be security and audit measures to ensure that resource is legally used for the purpose intended by virtual enterprise.

The earliest access control models for resource sharing include Access Control Lists (ACLs) and Access Control Matrix (ACMs). These schemes are simple and intuitive, but are only useful for small organizations [5]. Most current access control policies, such as Mandatory Access Control (MAC), Discretionary Access Control (DAC), Role-based Access Control (RBAC) [6–9], Task-based Access Control (TBAC) and Task–Role-based Access Control (T-RBAC) [10–12], consider merely the authorization management within a single organization. Some researchers have studied distributed role-based access control to delegate administration to individual departments in an enterprise [13]. Team-based Access Control 2004 (TMAC04) was built on the RBAC, which allows users to join team roles in an organization [14]. Park et al. proposed a composite role-based access control approach that separates organizational and system level role structures to support scalable and reusable RBAC models [13,15]. Cohen presented the family of Coalition-based Access Control (CBAC) models and policies to share specific data and functionality with coalition partners [16].

Although role-based methods have been successfully used in resource management within an enterprise, there are still many issues on management of resource sharing across organization boundaries to support collaborative and cooperative business activities. Access control for virtual enterprising is complicated because (1) members of the VE may change frequently; (2) VEs have members with complicated relationships; (3) VEs may be integrated or distributed; (4) VEs are Internet-based and heterogeneous [17–23]. The goal of this study is to provide a solution for information sharing across enterprises to facilitate cross-enterprise collaboration and concurrency, and thus enable the above-mentioned difficulties to ease.

This study proposes a Virtual Enterprise Access Control (VEAC) model to solve the problem of authorization management and security control among organizations within a VE. The proposed model consists of a Project-based Access Control (PBAC) model for managing *public resources* within VE and an RBAC model for managing the sharing of an individual enterprise's *private resources* with VE members. The architecture of a VEAC model-based system is developed and consists of three core mechanisms. Based on the system architecture, the study proposes certificate authentication, user authority and access control approaches to update and search user authority lists. Besides resolving the issues of resource sharing across organizations, the following properties of the proposed access control model make flexible, adaptable, extensible and instantaneous at a minimum administrative cost: (1) the model enables resource managing and sharing collaboratively; (2) the model enables change of access rights dynamically; (3) the study prevents to disclose business secret in VE; (4) the access authorization may be extended to the partners of the VE members.

2. Requirement analysis for access control in VE

The characteristics of a VE are identified by analyzing its life cycle and member interactions.

(1) AVE may consist of several distributed VEs or enterprises.
(2) A VE's participating members and business processes in a change during its life cycle.
(3) A VE emphasizes professional division and dynamic cooperation among a highly heterogeneous membership.
(4) A VE conducts business processes across enterprises divided into different stages, in which each stage has its own participants, resources and aims.
(5) In a VE, various resources are shared and distributed over all participating enterprises and used by their employees (users).
(6) A VE globally specifies members' obligations, responsibilities and roles.
(7) A change in a member's role in a process should not affect the obligations and responsibilities in its other assigned roles.
(8) Regulations do not constrain the selection of members in participating enterprises' partners.
(9) Each member may own its enterprise resource management policy and access control model.
(10) Shared VE resources include private resources owned by a participating enterprise and stored in its own repositories, and public resources belonging to the VE and stored in a public repository.

Based on the general requirements in access control in [10,11,16,17,20], this study identified the following requirements for access control model design: (1) only the security administrator is allowed to change security attributes; (2) roles may inherit authority either fully or partially); (3) the model supports active and passive access control, as well as the principle of strict least privilege; (4) the fine-granted authority requirements are fulfilled; (5) access authority may vary with tasks or roles; (6) the model can manage all users and resource objects in the enterprise [17,24,25].

Besides the above requirements, based on the characteristics of VE, additional requirements must be considered when developing a VEAC model, as follows:

(1) Since the organization structure of a VE is dynamic, access rights and resource objects can be changed in real time.
(2) The model considers all users' access rights, because resource administrators cannot predict who will access which resources in a VE.
(3) As a VE is formed to achieve a certain goal in a limited time frame, each VE has different goal and business processes. A VE is always conducted as a project. Therefore, project is an essential unit of access control.

(4) Since each enterprise has a legacy access control system, the VEAC model is easily integrated with various access control models or policies.
(5) The VE manages and shares resources collaboratively.
(6) To facilitate trust among enterprises, the access policy in VE is planned and managed together by administrators of all participating enterprises.
(7) The VE can maintain the consistency of policies and manage the conflicts between VE access policy and members' own access policies.

Because the VE emphasizes applications of Information Technology and Network across enterprise boundaries, the following system-related factors are considered when developing a VEAC-based system:

(1) System must offer a gateway to access resources on distributed heterogeneous platforms must be offered.
(2) For high runtime efficiency, the access control system must be able to interact directly with other applications or agents.
(3) Users' identity must be authenticated via a third party called a Certificate Authority (CA) Center due to the issues of authentication and non-repudiation.
(4) To support integrity and confidentiality for information exchange, a Public Key Infrastructure (PKI) is needed.
(5) A flexible security system needs a Plug-and-Play key component to mediate between the VEAC model and other RBAC-based models.

3. Virtual enterprise access control model

Each participating enterprise may already have adopted an access control model before joining a VE. Therefore, the VEAC model must be able to integrate with other access control models. As RBAC is the most popular access control model [19,26], the proposed VEAC model consists of a PBAC sub-model which can integrate into various role-based access control sub-models. This section presents and describes the two sub-models.

3.1. Overview of the concept

A VE's activities may use its own *public resources* of VE and the private shared resources of participating enterprises. Fig. 1 illustrates the conceptual framework of the VEAC model in which the PBAC sub-model is designed to manage public VE resources, while the role-based access control model manages the private resources of participating enterprises. The VEAC framework primarily emphasizes on the following capabilities to resolve the problems of access control across enterprises: (1) the access control models of participating enterprises can be plugged-in or plugged-out at any time without affecting the performance of access control models in other participating enterprises; (2) the model can simultaneously manage public and private resources; (3) the basic information of models can be updated with changes in the environment to authorize new users; (4) the user authorities can be generated according to role hierarchy and relations; (5) the stratified management method is used to increase the security of public and private resources.

3.2. Role-based Access Control model

This study slightly adjusted the RBAC model to seamlessly integrate it with the PBAC model. In the adjusted Role-based Access Control model, elements and assignments are simply described as follows:

- Users (U) represent a human or agent in an organization, which include direct users, indirect users, and non-member users.
- Roles (R) represent functional jobs or responsibilities.

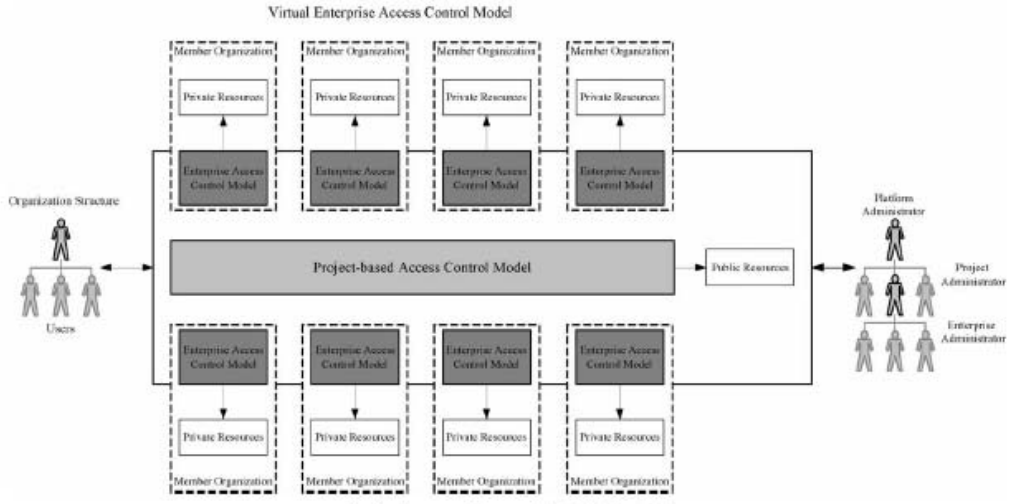

Fig. 1. Conceptual framework of VEAC model.

- Private Objects (PrivateO) represent resources in an enterprise associated with private privileges. Private Objects are usually classified into three levels including public, proprietary, and protection. The public classification can be provided to the partners in a VE.
- Private Permissions (PrivateP) are approvals of a particular mode of access to one or more private objects.
- Sessions (S) represent each session, via which users are mapping to one or more roles;
- U–R–A $\subseteq$ U $\times$ R is a many to many user to role assignment relation.
- R–PrivateP–A $\subseteq$ R $\times$ PrivateP is a many to many role to private permission assignment relation.
- PrivateP–PrivateO–A $\subseteq$ PrivateP $\times$ PrivateO is a many to many private privilege to private object assignment relation.

3.3. Project-based Access Control model

This section elucidates the PBAC model and defines all its elements, assignments among elements and assignments among models.

3.3.1. Core concept of the PBAC model

A "*virtual enterprise*" (VE) can perform several "*projects*" (P), but a project can only be performed by one VE. Different "project relations" (PR), such as subset, exclusion and reference, exist among projects. Activities within a project can be divided into several "*functional tasks*" (FT), each of which has access to certain public resources, which is their "*public permission*" (PublicP). A project involves some "*virtual enterprise roles*" (VER) to perform functional tasks.

A VE is composed of several real "enterprise members" (EM), each of which can participate in more than one VE. "*Non-enterprise members*" (NEM) are real enterprises that do not participate directly in the activities of VE but participate in the activities of an enterprise member which performs directly the activities of the VE. All VE participants, including three user types, are called "*users*" (U) which may play a different "*role*" (R) in a different "session". Each role has access to private resources, called a "*private permission*" (PrivateP). A superior role can inherit the privileges of inferior roles through "*role hierarchy*" (RH). The enterprise member plays a VE role through a user or role to obtain the privilege of sharing public resources in the VE and carry out practically the obligations a given VE role, and to achieve the VE goals.

"*Project access control policy*" is designed to identify the resource sharing rules in a project. Through constructing relations among projects and a project access control policy, users can share resources among projects. The rules of sharing can be modified at any time (Fig. 2).

3.3.2. Fundamental elements

This section defines all elements of the PBAC model:

- *Virtual Enterprise* (VE): The VE is a dynamic Internet organization, consisting of enterprise members, to achieve a business goal.
- *Enterprise Member* (EM): An EM can be a substantive enterprise organization, VE or individual, and is a VE member, with at least one worker participating directly the VE activities, and responsible for playing at least one virtual enterprise role.

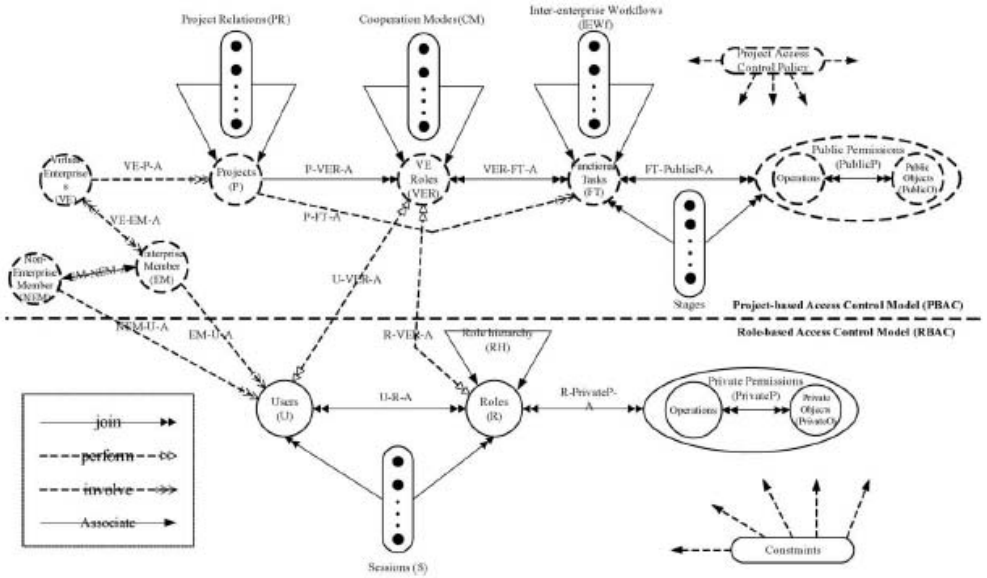

Fig. 2. Virtual enterprise access control (VEAC) model.

- *Non-enterprise Member* (NEM): An NEM is a substantive enterprise organization, VE or individual, but not a VE member, and participates indirectly in VE activities. An NEM has at least one worker participating directly in activities of enterprise members, and the activities have direct relations with functional task of VE.
- *Project* (P): A Project is the basic unit of VE activity. One project can have participants which are enterprises, departments or individuals, known as enterprise members. A project can be further divided into several sub-projects with various project relations. A project is composed of orderly functional tasks performed by enterprise members.
- *Functional Task* (FT): A FT is a set of VE activities which have a common objective to achieve a part of VE's responsibilities.
- *Virtual Enterprise Role* (VER): VERs, virtual roles created to enable professional divisions within VE, are the divisions of duties or activities in a VE, which are assigned to enterprise members to perform. Functional tasks can be assigned to one to more VERs.
- *Object* (O): Objects are the public and private resources held by VE and enterprise members. This study focuses on information objects, which can be databases, entities, attributes, tuples, documents, XML documents, applications, software components or knowledge.
- *Public Object* (PublicO): Public objects are objects used by enterprise members and stored in a VE's common repositories. Public Objects are provided for performing functional tasks or are created when functional tasks are completed.
- *Private Object* (PrivateO): Private objects are a subset of objects owned by a VE's member and stored in a private repository.
- *Public Permission* (PublicP): Public permissions indicate permitted modes of access to public objects.
- *Private Permission* (PrivateP): Private permissions indicate a permitted mode of access to a private object.
- *Permission*: Permission = {PublicP ∪ PrivateP}.
- *Project Access Control Policy* (PACP): PACP identifies which project resource are protected and shared according to the relations among projects and the shared rules, and what activities are forbidden in the virtual enterprise scope. Each project involves a PACP which can be performed automatically by the VEAC system. The PACP can be dynamically created, enforced and adjusted when the VE environment changed.

3.3.3. Project relations

A Project Relation (PR) describes the interaction, cooperation modes and priority between two projects. Different project relations may exist between two projects and change with time according to project management requirements. In the VEAC platform, the administrators construct a relative project resource access strategy in project access control policy (PACP) to indicate the level of resource sharing of each type of project relations. In the project life cycle, the project relations and the PACP can be changed at any time to respond to demands of resource sharing.

(1) *Subset Relation* (PR_{subset}): Describes the relation between a "main-project" and its "sub-project". The subset relation is a binary relation. Several constraints are applied to use of subset relation: (a) a main-project may have more than one sub-project; (b) a sub-project may be involved in only one main-project; (c) an enterprise member may participate in the main- and sub-projects; (d) a public permission may be merely assigned to different projects with subset relations. A main-project is allowed to access the resources of its sub-projects, but an administrator may set or disable the capability.

(2) *Version Relation* ($PR_{version}$): Describes a project "post-version project" which is extended from a project "pre-version project" and planned with reference to the pre-version project. Therefore, the pre- and post-version projects have similar targets, functional tasks and participants. The version relation between two projects may cause the correspondences between functional tasks of the two projects. The version relation is a binary relation.

(3) *Reference Relation* ($PR_{reference}$): Describes that a project "referring project" refers to the resources in other project "referred project". If the reference relation exists between two projects, the resources of referred project can be referred by users in referring project. The following constraints are applied when using the reference relation: (1) a project may set up more than one reference relation with other projects for resource sharing; (2) a project may refer to various projects simultaneously.

(4) *Process Relation* ($PR_{process}$): Describes the executive sequence of two sub-projects from time perspective. It determines the time to sharing project resources. Expression $PR_{process}$ (*event-project 1, ..., event-project m; condition 1, ...condition n; action-project k*) means that if event-project p_i for $1 \le i \le m$ is accomplished and condition c_j for $1 \le j \le n$ is valid, then action-project p_k can be triggered. When a project is decomposed into several sub-projects, Process Relation can be used to determine the executive sequence of all sub-projects. The process relation between two projects is a binary relation. While the relation is built on two projects, the administrator must specify the sequences of related functional tasks across project boundary. At the stages of executing an action-functional task which can use the resources of the event-functional tasks in event-project. The following constraints must be obeyed while using the process relation: (a) a process relation exists between two projects which must have the subset relation; (b) an event-project may trigger more than one action-project simultaneously; (c) an event-functional task may trigger more than one action-functional task simultaneously; (d) an action-project may be triggered if all of its event-projects are accomplished.

(5) *Exclusive Relation* ($PR_{exclusive}$): Identifies mutual conflict between projects, so that the resources of the two projects cannot be referred to each other. The exclusive relation is default. That is, if no other relation exists between two projects, then two projects are pre-set as Exclusive Relation. The exclusive relation is a binary relation. Supposing two

projects are exclusive, then all functional tasks in a project are exclusive with the other project. The following constraints must be obeyed while using the process relation: (a) a project may conflict with more than one project simultaneously; (b) a public permission may not be assigned to two exclusive projects; (c) an enterprise member is not allowed to be assigned to two mutual exclusive projects.

Fig. 3 shows an air force bomber project as an example. Project 1.1, "aircraft structure" is, decomposed into four Sub-projects: Project 1.1.1 "fuselage", Project 1.1.2 "wings", Project 1.1.3 "tail", and Project 1.1.4 "landing". The schedule of Project 1 "air force bomber" in order is: Project 1.1 "aircraft structure", Project 1.2 "propulsion systems", Project 1.3 "aircraft control systems" and Project 1.4 "armament systems". The relation between Projects 1 and 2 is an exclusive relation. Therefore, any resources of the two projects will be not shared during their life cycles. Partial works of Project 1.1.2 "wings" and Project 1.1.4 "landing" must refer to design diagrams of Project 1.1.1 "fuselage" while a stage of structure design of the Project 1.1.2 and the Project 1.1.4 "landing" is performed by workers of the two projects.

3.3.4. Cooperation modes

This section presents three cooperation modes among virtual enterprise roles according to the resource sharing requirements of collaborative operations in the VE:

Cooperation Mode (CM) describes interactive method among virtual enterprise roles according to the dependence of their duties. The use of cooperation modes is constrained by the following rules:

(1) A virtual enterprise role is permitted to have different cooperation modes with other VE roles.
(2) Only one cooperation mode is permitted between two VE roles.
(3) The use of cooperation modes among virtual enterprise roles should consider the authority conflict problems caused by the reflexive, symmetric and transitive properties, as well as security problems caused by unlimited extension of permissions. The three above-mentioned properties of relations are discussed in detail in Section 3.3.5.

According to the VE coordination requirements, three cooperation modes exist:

(1) *Dependent Single-task Mode*: When several virtual enterprise roles cooperate to perform a functional task, they all have the same access privilege to all its resources.
(2) *Dependent Multi-task Mode*: Virtual enterprise roles perform related functional tasks separately. Outputs of the functional tasks are referred to each other.
(3) *Independent Mode*: Virtual enterprise roles perform independent functional tasks separately, disregarding their outputs. If two virtual enterprise roles work in an independent mode, then they may not have each other's access privileges for functional tasks performed by them.

3.3.5. Property of relations

This section presents a Role Relation Net (RRN) to identify the interactive relations among projects, cooperation modes, roles and hierarchical relations in enterprise members, assignment relations between users and roles and relations between roles and VE roles. Through the RRN, users can be authorized proper privileges in proper time according to roles played by users and VERs performed by roles.

Fig. 4 shows an example of an RRN. The RRN includes two projects, Projects P1 and P2, performed by virtual enterprise

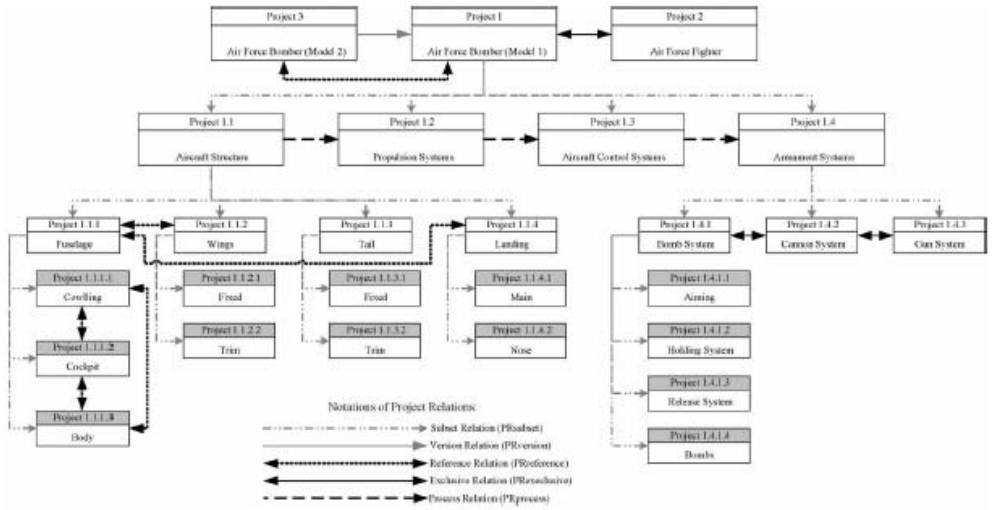

Fig. 3. Example of project relations.

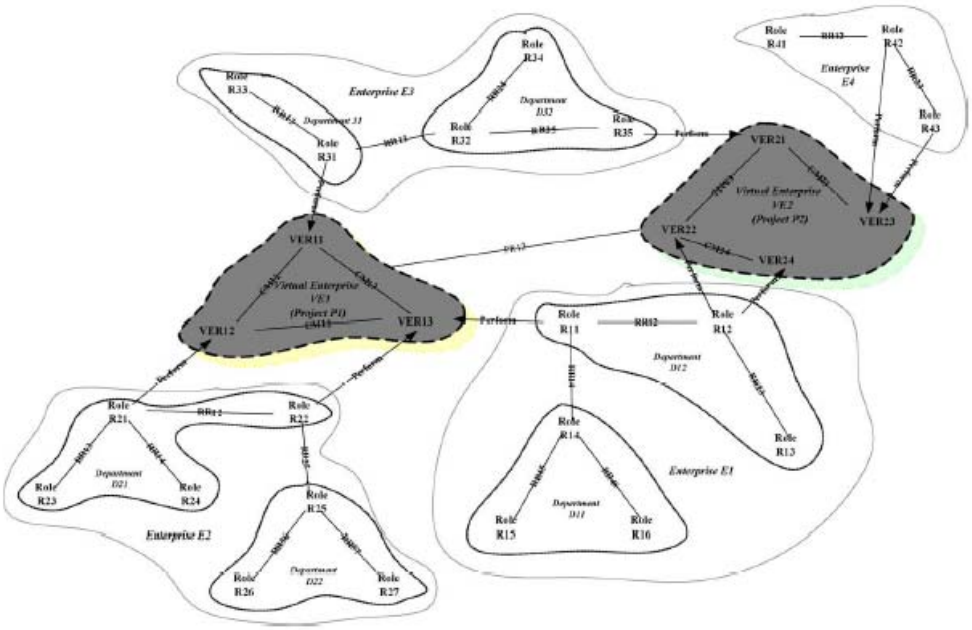

Fig. 4. Role Relation Net (RRN).

VE1 and VE2, respectively. The members of VE1 involve enterprises E1, E2 and E3, while the members of VE2 involve enterprises E1, E3, and E4. Role R21 in enterprise E2 is responsible for playing VE role VER12 in VE1. Through the cooperation mode CM12 between VER12 and VER11, Role R21 can be authorized to use part of VER11's resources. Through the role relation RR12 between role R21 and R22, R22 is allowed to access part of VER12's resources of in VE1.

To avoid security problems caused by privilege expansion due to element relations, and to strengthen private and public resource security, three binary relation properties — reflexive, symmetric, and transitive — are applied to the above-mentioned relations.

In the project formation stage, members in a VE determine whether each cooperation mode and project relation satisfies these three properties. The enterprise itself can identify these three properties according to its own resource sharing rules. The enterprise can also set the depth of the transitive property and require symmetric and transitive properties to be valid only in the same department.

Table 1
List of project relation properties

Project relation (PR)	Reflexive	Symmetric	Transitive
(1) Subset relation	X	X	O (Degree)/X
(2) Version relation	X	X	O (Degree)/X
(3) Reference relation	X	O/X	O (Degree)/X
(4) Process relation	O/X	O/X	O (Degree)/X
(5) Exclusive relation	X	O/X	O (Degree)/X

Tables 1–3 list properties in project relations, cooperation modes and role relations, respectively. Table 1 shows all possible combinations of the five project relations concerning reflexive, symmetric and transitive properties, which are introduced as follows:

(1) Subset Relation: The subset relation does not satisfy reflexive and symmetric properties. However, a project manager can determine whether the transitive property is satisfied. Meanwhile, the project manager can determine the continuability of the transitive property for resource sharing.
(2) Version Relation: The version relation does not satisfy the reflexive and symmetric properties, while the project manager can determine the transitive property. Meanwhile,

Table 2
List of cooperation mode properties

Cooperation mode (CM)	Reflexive	Symmetric	Transitive
(1) Dependent single-task mode	X	O	O (Degree)/X
(2) Dependent multi-task mode	X	O	O (Degree)/X
(3) Independent mode	X	O	O (Degree)/X

Table 3
List of role hierarchy properties

Role hierarchy (RH)	Reflexive	Symmetric	Transitive
Role relation name	X	X	O (Degree/Department)/X

the project manager can determine the continuability of the transitive property according to demands.

(3) Reference Relation: The reference relation does not satisfy the reflexive property because a project does not need to refer to itself. However the project manager can determine whether the symmetric and transitive properties are satisfied according to demands. Meanwhile, a project manager can determine the continuability of the transitive property according to demands. If the symmetric property exists between projects p_1 and p_2, project p_1 refers to project p_2, and vice versa. If the transitive property of reference relations exists among projects, and project p_1 refers to project p_2 and project p_2 refers to project p_3, then project p_1 can refer to project p_3.

(4) Process Relation: The project manager may determine whether the reflexive, symmetric and transitive properties are satisfied.

(5) Exclusive Relation: The exclusive relation does not satisfy the reflexive property, while the project manager can determine whether the symmetric and transitive properties are satisfied.

Table 2 lists the three cooperation modes showing all possible combinations of their reflexive, symmetric and transitive properties. Since these properties have the same value, only the Dependent Single-Task Mode is explained. The Dependent Single-Task mode does not satisfy the reflexive property, but certainly satisfies the symmetric property because VE role ver_1 has cooperation relations of Dependent Single-Task Model with ver_2. Conversely, ver_2 has cooperation relations of Dependent Single-Task Mode with ver_1. Additionally, the project manager may determine whether transitive property and depth of transitability are satisfied.

Table 3 lists role hierarchical relations, showing all possible hierarchical relations among roles, concerning their reflexive, symmetric and transitive properties. The properties of role hierarchy should be determined by the resource sharing strategy of an enterprise or department. Therefore, the Role hierarchy does not satisfy the reflexive and symmetric properties, but it is permitted to have different transitive properties among departments in the same enterprise. The depth of a role hierarchy's transitive property can also be determined, and the validity of transitive property may be established only within a department.

The properties of listed relations primarily have three effects: (1) to enhance resource sharing flexibility among projects and the availability of resource sharing in a VE; (2) to analyze whether project relations violate listed rules and to discover conflicts; (3) to analyze RRN to generate a user's privilege according to listed contents.

3.3.6. Foundational assignments

This sub-section defines various assignment relations among elements as follows:

- *Functional Task-Stage-Public Permission-Assignment* (FT-S-Public-A): A triple assignment relation among three elements: Functional Task, Stage, and Pubic Permission. Public permissions are assigned to functional tasks in stages. The relation among them is: FT × Stage × Public Permission.
- *Project-Virtual Enterprise Role-Assignment* (P-VER-A): This relation records the assignment relation between projects and virtual enterprise roles, and describes which virtual enterprise roles are included in a project.
- *Virtual Enterprise Role-Functional Task-Assignment* (VER-FT-A): This relation records the assignment relation between virtual enterprise role and functional task, and describes which functional tasks are performed by which virtual enterprise roles.
- *Virtual Enterprise-Enterprise Member-Assignment* (VE-EM-A): This relation records assignment relations between a VE and its enterprise members.
- *Virtual Enterprise-Project-Assignment* (VE-P-A): This relation records the assignment relations between a virtual enterprise and its projects, and describes which project is performed by a virtual enterprise.

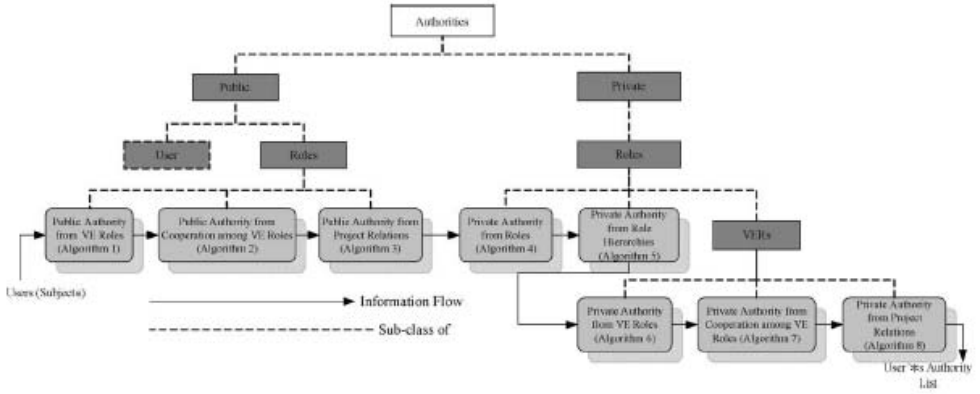

Fig. 5. Authority classifications.

3.3.7. Assignments across models

This section defines the relation assignments across models to establish the combination relations of relevant elements among two access control models. They are:

- *Enterprise Member-User-Assignment* (EM-U-A): This relation records the assignment relations between users and enterprise members.
- *Non-Enterprise Member -User-Assignment* (NEM-U-A): This relation records the non-enterprise members for which a user works.
- *Role-Virtual Enterprise Role-Assignment* (R-VER-A): This relation records the assignment relations between roles and virtual enterprise roles.
- *User-Virtual Enterprise Role-Assignment* (U-VER-A): This relation records what VE roles a user may play.

4. Classification of user authorities

Initially, according to the sources of user's authorities, a user's authorities can be classified into two categories as shown in Fig. 5:

(1) *Public authority*: The authority of public resources, which is obtained from VE roles performed by user roles. The authority of public resources can be subdivided into authority held by user and authority held by role. Because the algorithms for generating authority held by user is included in the algorithms for generating authority held by role, this study explores only the authority held by role. Its sources can be subdivided into three types:
   (a) *Public Authority from VER*: The access authority derives from virtual enterprise roles played by user's roles in an enterprise member. Since user's roles can play different virtual enterprise roles, these authority of virtual enterprise roles may derive from different projects.
   (b) *Public Authority from Cooperation among VER*: The access authority derives from virtual enterprise roles that cannot be played by user's roles. These authorities are obtained through cooperation modes among VE roles played by the user's roles and other VE roles leading to resource sharing.
   (c) *Public Authority from PRs*: The access authority derives from resource sharing among projects.

(2) *Private authority*: Authority of private resources existing in enterprise members and obtained through user roles. This authority can be subdivided into five types:
   (a) *Private Authority from Roles*: The access authority derives from user's roles.
   (b) *Private Authority from RHs*: The access authority derives from hierarchical relations between the entering user's roles played and roles not played by him. These roles inherit partial authority of other roles with which they have hierarchical relations.
   (c) *Private Authority from VERs*: A VE role can be collaboratively played by many roles. To reach the common goal for performing VE roles, roles may share part of their authorities to other collaborative roles.

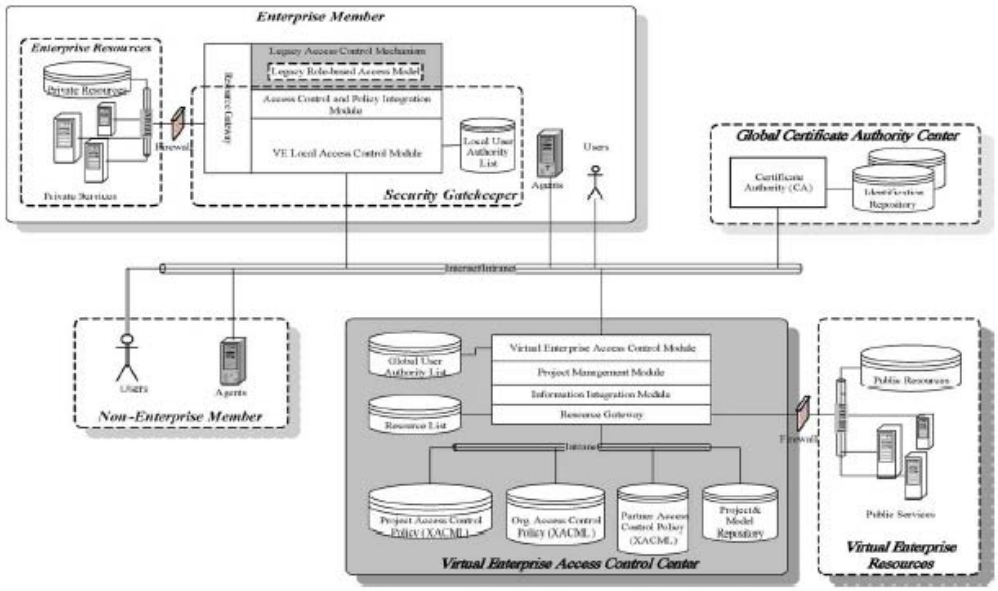

Fig. 6. A VEAC system architecture.

Therefore, the access authority derives partially from the authorities of other roles with which the role cooperates collaboratively to perform the VE role.

(d) *Private Authority from Cooperation among VERs*: This authority uses authorities owned by other roles, exists in private resource of enterprise and is obtained through the cooperation model of playable virtual enterprise role and other virtual enterprise roles.

(e) *Private Authority from PRs*: This authority uses authorities existing in private resources in other enterprises and obtained through project relations.

5. System architecture and approaches design

To support resource management and security control in VE, this study developed a VEAC system based on the proposed VEAC model.

5.1. System architecture

This section designs the VEAC system architecture according to resource management requirements and characteristics in VE.

Fig. 6 shows the VEAC system architecture, in which the primary mechanism includes a *Virtual Enterprise Access Control Center* (VEACC) responsible for authority management security control, and deployable in a leader enterprise. Every enterprise member joining the VE has to install a *Security Gatekeeper* (SG) to protect its own resources. To authenticate the user's identity on the Internet, the VEACC sends the user's login to the *Global Certificate Authority Center* (GCAC).

The main mechanisms in the VEAC system architecture are introduced as follows:

- *Virtual Enterprise Access Control Center* (VEACC): The aims of the VEACC include: (1) to enable the administrator to construct and maintain systems; (2) to provide an interactive interface with other mechanisms, and encryption and decryption for secure communication; (3) to generate user authority lists according to the VEAC model; (4) to authenticate the user; (5) to request resource services in the VE. Based on the aims and function requirements of virtual enterprise access control, a functional framework of the VEACC is designed as Fig. 7, which displays the main functional modules or components and their repositories. The functional framework of VEACC consists of the following modules:

(1) GUI user and administrator interface includes user requirement interface, team administrator interface, organization administrator interface and platform administrator interface.

(2) Model and policy integration module includes both model integration unit and policy translation unit.

(3) Authentication and access control module includes identification and authentication unit, policy handler unit, audit unit, session management unit, and access control and authorization unit.

(4) Information integration mechanism is able to transform various information into a understandable information format for users.

(5) Resource gateway is an interactive interface to connect public resources and each member enterprise's security

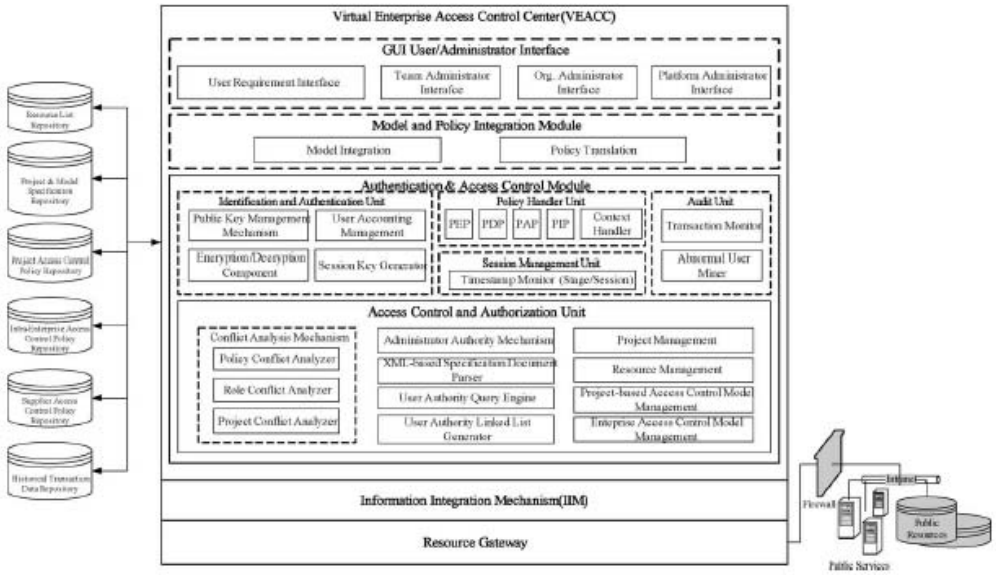

Fig. 7. Functional framework of VEACC.

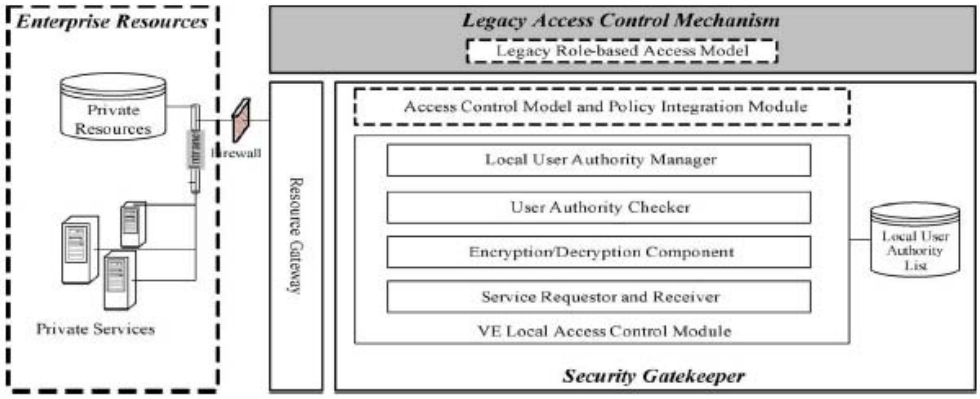

Fig. 8. Functional framework of security gatekeeper.

gatekeeper for accessing private resources through firewall.

These repositories in the framework are designed to store (1) resource list, (2) project and model specification, (3) project access control policy, (4) intra-enterprise access control policy, (5) supplier access control policy, and (6) historical transaction data to support access control activities, including user's authentication and authorization, and examining disallowed accesses.

- *Security Gatekeeper* (SG): Every enterprise that joins VE has to install this component to (1) protect its own internal resources; (2) act as an interface for communicating with VEACC; (3) request resource services in enterprises. The functional framework of security gatekeeper designed as Fig. 8 which consists of three main function modules: (1) virtual enterprise local access control module comprising local user authority manager, user authority checker, encryption and decryption component, and service requestor and receiver; (2) access

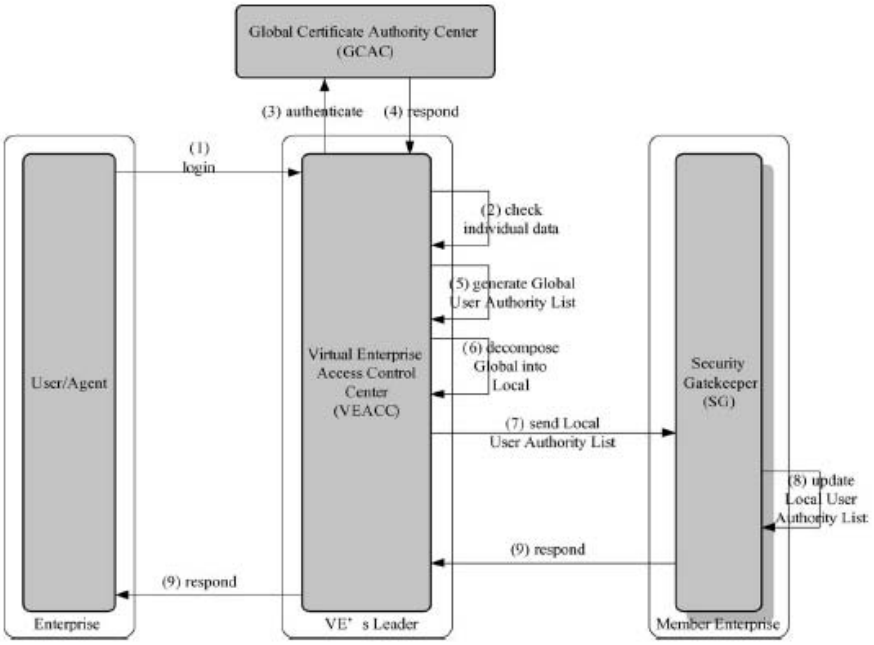

Fig. 9. Approach for updating user authority list.

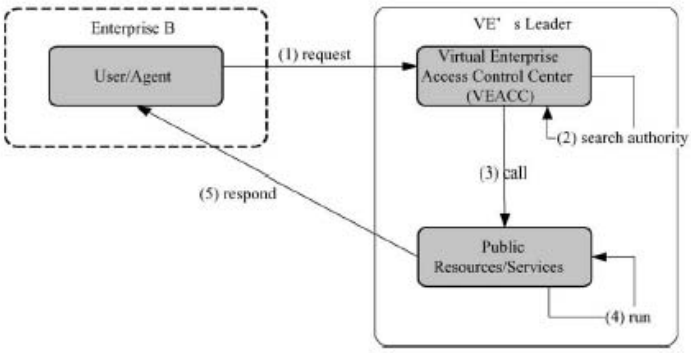

Fig. 10. Approach for public resource access without information integration.

control model and policy integration module; (3) resource gateway.

- *Global Certificate Authority Center* (GCAC): Every user or enterprise must have a digital certificate to authenticate them within the Network. The GCAC, a third party, is responsible for certificate authentication and notifying VEACC of the results.

5.2. Certificate authentication, authority and access control approaches

In virtual enterprise access control, authenticating a user is an essential step before authorizing the user for any protected operation. Since VE members often change, the VE user authority has to be frequently updated to protect its resources. Therefore, the certificate, authorization and access control management are important in a VE. This section shows the operations related to this job. Analyzing the resource access requirement in virtual enterprise, regardless of public or private resources, in which they include two access modes to need integration and not integration. In addition to the access modes, peer-to-peer private resource access mode is often used, too. The following sub-sections will illustrate the approaches in order.

5.2.1. Approach for updating user authority list

When a user enters the VEAC system, the system must generate a user authority list, and update each SG's local user authority list and the VEACC's global user authority list. This approach is shown in Fig. 9 and explained as follows:

(1) User logs onto the VEACC and enters his personal data including name, validity period, public key information and a signed hash of the certificate data.
(2) VEACC authenticates the user's personal basic data; if the user data are incorrect, then the VEACC rejects the user.
(3) If the user data are correct, then the VEACC sends the user personal data to the GCAC to authenticate the digital certificate.

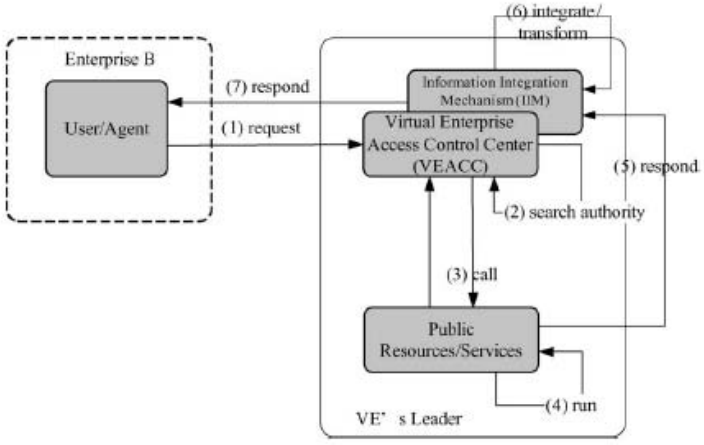

Fig. 11. Approach for public resource access with information integration.

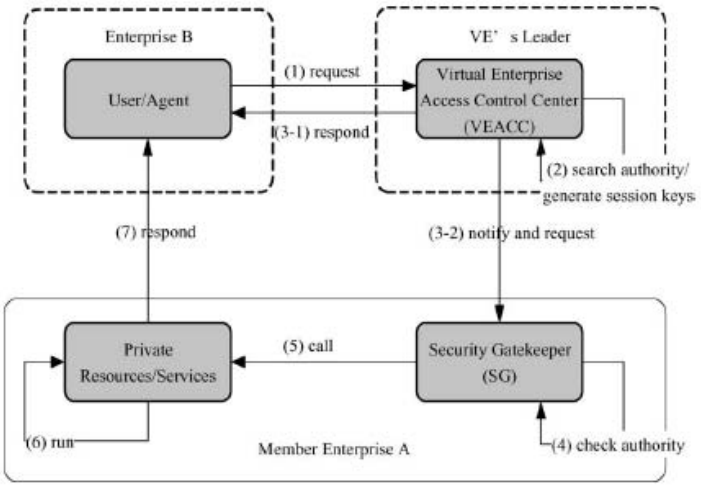

Fig. 12. Approach for private resource access without information integration.

(4) GCAC sends the verification results to VEACC.
(5) If the user's digital certificate is correct, then the VEACC generates the user authorities and adds them to the global user authority list.
(6) VEACC decomposes the user authorities according to the enterprise owning each resource, and generates a local user authority list for each enterprise.
(7) VEACC sends a local user authority list of each enterprise to their SG.
(8) Each SG updates its local user authority list.
(9) SG informs VEACC that SG has completed the updating procedure.
(10) VEACC informs the user that he may access VE resources.

5.2.2. Approaches for accessing public resources

A variety of public resources in virtual enterprise is shared, some of which could need to be integrated. VEACC provides two approaches for accessing public resources with and without information integration, shown in Figs. 10 and 11, respectively. The information integration mechanism (IIM) supports the information format transformation among enterprises.

- Approach for accessing public resources without information integration

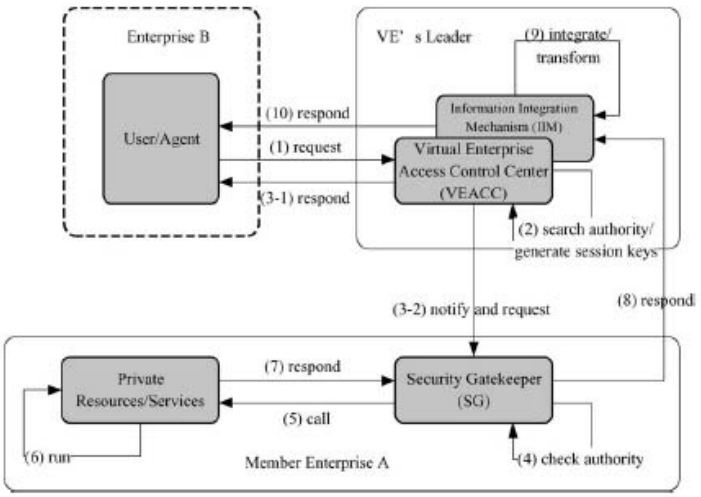

Fig. 13. Approach for private resource access with information integration.

The approach for public resource access without information integration is shown in Fig. 10 and explained as follows:

(1) User/Agent in enterprise B requests access to Public Resources.
(2) VEACC receives the request, and searches the user authority from the Global User Authority List.
(3) If User's requested operation is allowed, then VEACC sends a call statement to Public Resources in a virtual enterprise platform.
(4) The Public Resources perform the service requested by User.
(5) The Public Resources directly respond with the results using an appropriate format to represent the User information.

- Approach for accessing public resources with information integration

The approach for public resource access with information integration is shown in Fig. 11 and explained as follows:

(1) User/Agent in enterprise B requests access to Public Resources which need to be transformed into another format.
(2) VEACC receives the request, and searches the user authority from the Global User Authority List.
(3) If User's requested operation is allowed, then VEACC sends a call statement to Public Resources in a virtual enterprise platform.
(4) The Public Resources perform the service requested by User.
(5) The Public Resources directly respond with the results to Information Integration Mechanism (IIM).
(6) The IIM proceeds with information integration and transformation according to the information requirement of enterprise B.
(7) The IIM respond with the results using enterprise B's format to represent the responded information.

5.2.3. Approaches for accessing private resources

A variety of private resources in enterprise is shared, which could need integration or not. VEACC provides two approaches for accessing private resources with and without information integration, shown in Figs. 12 and 13, respectively.

- Approach for accessing private resources without information integration

The approach for accessing private resource without information integration is shown in Fig. 12 and explained as follows:

(1) User/Agent in enterprise B requests access to Private Resources.
(2) VEACC receives the request, and searches the user authority in the Global User Authority List. If the User is allowed to access the Private Resource, then the VEACC generates a pair of session keys.
(3.1) VEACC responds with one session key.
(3.2) Simultaneously, the VEACC sends to the SG the other session key and the User/Agent's request.
(4) SG verifies again the authority for the request.
(5) If the request is valid, then SG with gateway calls the requested Private Resource.
(6) The Private Resource in member enterprise A performs the service requested by User in enterprise B.
(7) The Resource/Service directly responds with the results using an appropriate information format and encrypting it using the session key.

- Approach for accessing private resources with information integration

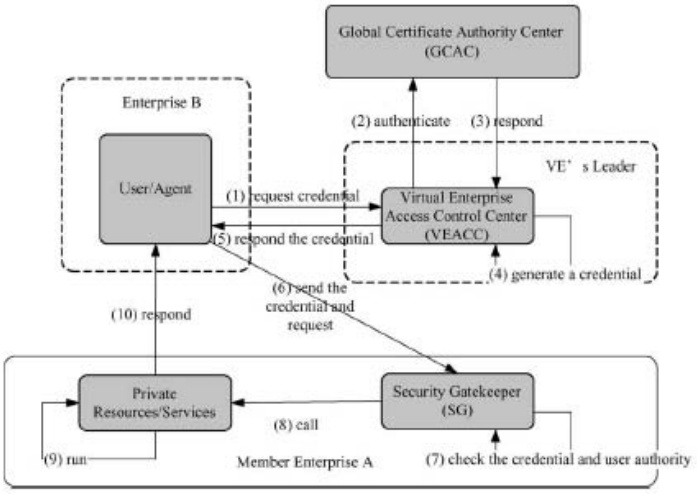

Fig. 14. Approach for accessing peer-to-peer private resources without information integration.

The approach for accessing private resource with information integration is shown in Fig. 13 and explained as follows:

(1) User/Agent in enterprise B requests access to Private Resources which need information integration and transformation.
(2) VEACC receives the request, and searches the user authority in the Global User Authority List. If the User is allowed to access the Private Resource, then the VEACC generates a pair of session keys.
(3.1) VEACC responds with one session key.
(3.2) Simultaneously, the VEACC sends to the SG the other session key and the User/Agent's request.
(4) SG verifies again the authority for the request.
(5) If the request is valid, then SG with gateway calls the requested Private Resource.
(6) The Private Resource in member enterprise A performs the service requested by User in enterprise B.
(7) The Resource/Service directly responds with the results using a standard information format to SG.
(8) The SG encrypts the results by using the session key and sends it to IIM.
(9) The IIM proceeds with information integration and transformation according to the information requirement of enterprise B.
(10) The IIM responds the results of request to User in enterprise B.

5.2.4. Approaches for accessing peer-to-peer private resources without information integration

In virtual enterprise environment, in order to speed up the efficiency of information access, the approach for accessing peer-to-peer private resources which need not to integrate is shown in Fig. 14, in which each step is introduced as follows.

(1) User logs onto the VEACC for requesting a credential and further enters his personal data including name, validity period, public key information and a signed hash of the certificate data.
(2) VEACC authenticates the user's personal basic data; if the user data are incorrect, then the VEACC rejects the user. If the user data are correct, then the VEACC sends the user personal data to the GCAC to authenticate the digital certificate.
(3) GCAC sends the verification results to VEACC.
(4) VEACC generates a credential for the user if the digital certificate is correct.
(5) VEACC responds the credential to the user.
(6) User sends the credential and request to SG.
(7) SG checks the credential and user authority.
(8) If the credential and user authority are legal, SG calls the private resource to supply service for the request.
(9) The private resource runs the request.
(10) The private resource responds the result to user.

In the approach, the steps (7)–(10) can be repeated to request other services during a timestamp.

6. Discussion and conclusions

The aims of this study may help VEs to successfully solve the challenges of resource management and sharing among enterprises. The study has already accomplished the phase objective to propose a VEAC model, design the architecture of a prototype system and the functional frameworks of its core mechanisms, and develop the approaches for authentication and authorization in VE. However, the study has some deficiencies. For example, the non-RBAC model and integration of its access policies was not investigated. If an enterprise adopts non-RBAC models and other access policies, it must perform additional model-transferring process to transform them to RBAC in order to integrate them with VEAC project-based access control model.

6.1. Results and contributions

The results and contributions of this study were as follows:

- The model may: (1) enables resources management and sharing in VE; (2) facilitate dynamic change of access right based on the organization structure of a VE; (3) preserve the access rights of users who are not affected under the change the organization of a VE or its members; (4) prevents to disclose business confidential information in virtual enterprising; (5) ban all users working in an enterprise and its partners from accessing resources in the VE, when the enterprise drops out from the VE.
- The system architecture and approaches enable: (1) users from anywhere can take up to date information; (2) single authentication can entry multi-domains to access resources; (3) authorization considers not only individual privilege but also privilege from other workers that work together with him; (4) the extent of resources sharing among workers depends on the cooperation relations among them and task requirements.

6.2. Further research

In the electronic commercial environment, resource management and sharing will become more complicated in the future. The proposed VEAC model solves access control and VE resource sharing problems. The implementation of the VEAC model-based access control system prototype is a great software engineering. In the future we will make up a distributed software engineering team to develop the system prototype using object-oriented software development methodology. However, some problems still need to be resolved.

(1) This study did not consider that the user might share a resource with unauthorized users, for example by copying it, after legally acquiring the resource.
(2) Algorithms based on the VEAC model should be developed to generate user authority.
(3) Methods for the access control server to call and use resources in the heterogeneous platform were not considered.

(4) An enterprise may adopt a non-RBAC-based scheme. Therefore, integrating different access control schemes or policies should be a focus points for future studies.
(5) Ideally, an enterprise should keep its original access control model when joining a VE. Therefore, a 'plug-and-play' access control integrating mechanism with a ability should be developed.
(6) An enterprise may participate in several competing VEs. Preventing the leaking of key technology or data should be considered.
(7) Future studies should adopt the eXtensible Access Control Markup Language (XACML) proposed by OASIS to develop access control policy frameworks enabling access strategies to be integrated among enterprises.

Acknowledgment

This research is financially supported by National Science Council of the Republic of China under Contract Nos.: NSC94-2524-S-024-002, NSC94-2524-S-006-005 and NSC94-2524-S-006-006.

References

[1] A. Frenkel, H. Afsarmanesh, C. Garita, L.O. Hertzberger, Supporting information access rights and visibility levels in virtual enterprise, in: Proceedings of the Second IFIP working Conference on Infrastructures for Virtual Organizations: Managing Cooperation in Virtual Organizations and Electronic Business towards Smart Organizations, 2000.
[2] N. Mezzetti, Towards a model for trust relationships in virtual enterprises, database and expert systems applications, in: Proceedings of the 14th International Workshop, 2003, pp. 420–424.
[3] T.J. Smith, L. Ramakrishnan, Joint Policy Management and Auditing in Virtual Organizations, Grid Computing, in: Proceedings of the Fourth International Workshop, 2003, pp. 117–124.
[4] Y.M. Chen, M.W. Liang, Design and implementation of a collaborative engineering information system for allied concurrent engineering, International Journal of Computer Integrated Manufacturing 13 (1) (1999) 11–30.
[5] S. Oh, S. Park, Task-role-based Access Control model, Information System (2003) 533–562.
[6] G.J. Ahn, Specification and classification of role-based authorization policies, in: Proceedings of the 12th IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises, 2003, pp. 202–207.
[7] R.A. Botha, J.H.P. Eloff, Designing Role Hierarchies for Access Control in Workflow Systems, in: Proceedings of the 25th Annual International Computer Software and Applications Conference, 2001, pp. 117–122.
[8] C. Yang, C.N. Zhang, Designing secure E-commerce with Role-based Access Control, in: Proceedings of the IEEE International Conference on E-Commerce, 2003, pp. 313–319.
[9] F. Dridi, B. Muschall, G. Pernul, Administration of an RBAC System, in: Proceedings of the 37th Annual Hawaii International Conference on System Sciences, 2004, pp. 187–192.
[10] A. Kern, A. Schaad, J. Moffett, Enterprise role administration: an administration concept for the Enterprise Role-based Access Control model, in: Proceedings of the Eighth ACM Symposium on Access Control Models and Technologies, 2003, pp. 33–40.
[11] C.J. Moon, D.H. Park, S.J. Park, D.K. Baik, Symmetric RBAC model that takes the separation of duty and role hierarchies into consideration, Computers & Security (2004) 126–136.
[12] D. Shin, G.J. Ahn, J.S. Park, An Application of Directory Service Markup Language (DSML) for Role-based Access Control (RBAC), in: Proceedings of the 26th Annual International Computer Software and Applications Conference, 2002, pp. 934–939.
[13] E.C. Cheng, An Object-oriented Organizational Model to Support Dynamic Role-based Access Control in Electronic Commerce Applications, in: Proceedings of the 32nd Annual Hawaii International Conference on System Sciences, Vol.: Track8, 1999, p. 9.
[14] F.T. Alotaiby, J.X. Chen, A Model for Team-based Access Control (TMAC 2004), in: Proceedings of the International Conference on Information Technology: Coding and Computing, vol. 1, 2004, pp. 450-454.
[15] J.S. Park, K.P. Costello, T.M. Neven, J.A. Diosomito, Access Management for Distributed Systems: a Composite RBAC approach for Large, Complex Organizations, in: Proceedings of the Ninth ACM Symposium on Access Control Models and Technologies, 2004.
[16] E. Cohen, T.K. Roshan, W. Winsborough, D. Shands, Models for Coalition-based Access Control (CBAC), in: Proceedings of the Symposium on Access Control Models and Technologies, 2002, pp. 97–106.
[17] L. Zhang, G.J. Ahn, B.T. Chu, A Rule-based Framework for Role-based Delegation and Revocation, ACM Transactions on Information and System Security (TISSEC) 6 (3) (2003).
[18] M.H. Kang, J.S. Park, J.N. Froscher, Access Control Mechanisms for Inter-organizational Workflow, in: Proceedings of the Sixth ACM Symposium on Access Control Models and Technologies, 2001.
[19] R. Nabhen, E. Jamhour, C. Maziero, RBPIM: a PCIM-based Framework for RBAC, Local Computer Networks, in: Proceedings of the 28th Annual IEEE International Conference, 2003, pp. 52–61.
[20] G. Denker, J. Millen, Y. Miyake, Cross-domain Access Control via PKI, in: Proceedings of the Third International Workshop on Policies for Distributed Systems and Networks, 2002, pp. 202–205.
[21] S. Osborn, Integrating role graphs: a tool for security integration, Data & Knowledge Engineering (2002) 317–333.
[22] W. Yamazaki, H. Nishiyama, F. Mizoguchi, Design of collaborative agent system with Access Control for smart-office environment, in: Proceedings of the 10th IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises, 2001, pp. 205–210.
[23] C. Yang, C.N. Zhang, Secure Web-based Applications with XML and RBAC, Information Assurance Workshop, IEEE Systems, Man and Cybernetics Society, 2003, pp. 276–281.
[24] J. Luo, D. He, Research on Object-oriented Role-based Access Control Model, in: Proceedings of the Fourth International Conference on Parallel and Distributed Computing, Applications and Technologies, 2003, pp. 132–135.
[25] J.D. Moffett, Control principles and role hierarchies, in: Proceedings of the Third ACM Workshop on Role-based Access Control, 1998, pp. 63–69.
[26] K. Furst, T. Schmidt, G. Wippel, Managing Access in Extended Enterprise Networks, Internet Computing, IEEE 6 (5) (2002) 67–74.

Mr. Tsung-Yi Chen is a PhD candidate of Institute of Manufacturing Engineering, National Cheng Kung University, Taiwan, ROC and a Lecturer of Electronic Commerce Management Department, Nan Hua University in Taiwan, ROC. He gained his MS degree from Institute of Manufacturing Engineering, National Cheng Kung University, Taiwan, ROC in 2001 and the BS degree from Department of Applied Mathematics, Providence University, Taiwan, ROC in 1996. His research interests include Virtual Enterprise, E-Commerce and E-Business, Enterprise and Information Integration, and Access Control.

Dr. Yuh-Min Chen is currently a Professor and the Director of Institute of Manufacturing Engineering, National Cheng Kung University, Taiwan, ROC. He graduated from the Ohio State University with a PhD degree in Industrial and Systems Engineering in 1991 and received his MS and BS degrees from National Tsing Hua University, Taiwan, ROC in 1981 and 1983, respectively. Before joining the faculty of Institute of Manufacturing Engineering in 1994, he worked as a research engineer in Structural Dynamics Research Corporation, USA for 3 years. His current research

interests include Enterprise Integration, Engineering Data and Knowledge Management, Computer-Aided Concurrent Engineering and Manufacturing Information Systems.

Dr. Hui-Chuan Chu is an Associate Professor of Department of Special Education, National University of Tainan in Taiwan, ROC. She received her PhD degree from Columbia University in 1998. Her research interests are Knowledge Management, Teacher Knowledge, and Integration of Information Technology in Teacher Education.

Dr. Chin-Bin Wang is currently a Professor and the Chairman of Electronic Commerce Management Department, Nan Hua University in Taiwan, ROC. He received his PhD degree in Computer Science from the City University of New York in 1995, and gained his MS degree from University of Southern California and BS degrees from National Tsing Hua University, Taiwan, ROC in 1985 and 1981, respectively. His research interests include Data Mining, Network Management, Engineering Data and Knowledge Management, System Integration.

附件 7

A Formal Virtual Enterprise Access Control Model

Tsung-Yi Chen, Yuh-Min Chen, and Chin-Bin Wang

***Abstract*—A virtual enterprise (VE) refers to a cooperative alliance of legally independent enterprises, institutions, or single persons that collaborate with each other by sharing business processes and resources across enterprises in order to raise enterprise competitiveness and reduce production costs. Successful VEs require complete information transparency and suitable resource sharing among coworkers across enterprises. Hence, this investigation proposes a formal flexible integration solution, named the formal VE access control (VEAC) model, based on the role-based AC model, to integrate and share distributed resources owned by VE members. The formal VEAC model comprises a fundamental VEAC model, a project AC policy (PACP) language model, and a model construction methodology. The fundamental VEAC model manages VE resources and the resources of participating enterprises, in which various project relationships are presented to facilitate different degrees of resource sharing across projects and enterprise boundaries, and cooperative modes among VE roles are presented to enable collaboration among coworkers in a VE. This PACP language model features object–subject–action–condition AC policies that jointly determine user access authorizations. In addition, the methodology supplies a systematic method to identify fundamental elements of the VEAC model and to establish assignments between elements and relations.**

***Index Terms*—Access control (AC), resource sharing, role-based access control (RBAC), virtual enterprise (VE).**

I. Introduction

VIRTUAL enterprise (VE) is regarded as one of the most promising business strategies to enhance the global competitiveness of enterprises [1]. VEs integrate the processes, activities, and resources from different enterprises through enterprise alliances to respond quickly to customer expectations. Frenkel *et al.* [2] defined a VE as a collaborative group of existing autonomous enterprises, which selectively share their expertise, skills, and resources to accomplish a common product or service. In practice, a VE is generally implemented with a distributed and collaborative business process, in which individuals from different enterprises cooperate on business-related activities or processes by remote coordination, communication, and control [1]. To attain VE goals and support each other's functionalities, enterprises in a VE must share and exchange information, knowledge, and resources. The features that determine the access level to the local information of every enterprise, when considering the competitive and cooperative relationships among enterprises, include the degree of trust between two enterprises, the function of the enterprises in the VE, and contractual agreements [2], [3]. Lu *et al.* [4] proposed a trust-based privacy preservation method for P2P data sharing.

A collaborative engineering environment allows multiple engineers to work simultaneously with individual assembly parts. Some manufacturing industries, e.g., the automotive sector, use VEs to maintain business relationships with their suppliers and corporate customers, enabling manufacturers to collaborate on the design, production, assembly, and marketing of new products. For instance, designing and developing a new car is a complex and lengthy process; during product R&D, engineering and design drawings can be shared over secure network among the contracting firm, testing facility, marketing firm, and downstream manufacturing and service companies [5]. Information concerning the design for a new product at various segments of the VE has to be visible to all members of the VE at any time. Consequently, the information must be managed properly, with appropriate access control (AC) models, strict policies, discipline, and daily monitoring. Development of a new car model by a VE might involve approximately 20 000 designers and engineers from hundreds of divisions and departments, some of which belong to different enterprises in different countries. One sub-VE in the car-manufacturing VE performs car design, which contains four subprojects, namely, engine design, cool system design, transmission case design, and framework design. Engine designers in the engine design subproject collaboratively develop an engine for the new car model. Information related to the engine, such as drawing and engineering data, is generated and shared in real time to workers in the subproject and other subprojects. Therefore, the success of a VE depends wholly on transparent and effective sharing of information resources, including information, application systems, and knowledge, throughout the business cycle [1]. Not all business partners are equally trusted in today's complex business environment. Today's partners could become tomorrow's competitors. Hence, enterprises do not generally like sharing information. Consequently, a VE or related business strategy, such as allied concurrent engineering or virtual team, is likely to fail. Therefore, VE urgently needs secure and trustworthy AC model, approach, and mechanism that can manage distributed resources across enterprises and share them with collaborative workers. To secure information sharing, competitive and cooperative relationships among enterprises should be considered when using the proposed model to evaluate a user's authorization to access resources.

Secure resource management and sharing across organizational boundaries have seldom been addressed. AC for VEs is difficult for the following reasons: 1) enterprise members in a VE may change frequently; 2) each enterprise member

Manuscript received August 6, 2006; revised May 18, 2007. This work was supported in part by the National Science Council of the Republic of China, Taiwan, under Contract NSC96-2221-E-343-002. This paper was recommended by Associate Editor J. Miller.

T.-Y. Chen and C.-B. Wang are with the Department of Electronic Commerce Management, Nanhua University, Chia-Yi 62248, Taiwan, R.O.C. (e-mail: tsungyi@mail.nhu.edu.tw; cbwang@mail.nhu.edu.tw).

Y.-M. Chen is with the Institute of Manufacturing Engineering, National Cheng Kung University, Tainan 70101, Taiwan, R.O.C. (e-mail: ymchen@mail.ncku.edu.tw).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSMCA.2008.923090

(EM) typically has many roles and users; 3) a VE often has many EMs with complicated interrelationships—for example, members may cooperate and also compete with one another; 4) organizations within a VE may be dynamic and perform unpredictable activities; and 5) VE resources may be Internet based, distributed, and heterogeneous. Few studies have explored control of access to knowledge, which is one of the most important assets for an enterprise. Therefore, developing an AC mechanism for knowledge protection has been recognized as a vital research topic in knowledge management [6]–[8]. Although role-based models have been adopted successfully for resource management within an enterprise, collaborative VE systems using role authorization management approaches have not been widely investigated. In contrast to conventional AC models, AC for a VE does not specifically assign rights to each role or user in advance because of the dynamic characteristics of VE organizations, such as flexibility and mobile resource sharing. To our knowledge, no studies have developed models for resource sharing management that support collaborative and cooperative business activities across organizational boundaries. Before achieving secure resource sharing in a VE that increases corporate global competitiveness, several requirements for trust management, such as scalability, flexibility, dynamic security, decentralization, and mutual trust, must be addressed [9]. Hence, VEs require an appropriate AC model.

Based on the conceptual AC model in VEs [10], [11], this investigation develops a formal VEAC model to solve the problem of authorization management and to secure AC among organizations within a VE. The formal VEAC model comprises a fundamental VEAC model, a project AC policy (PACP) language model, and a model construction methodology. The proposed fundamental VEAC model comprises a [project-based access control (PBAC)] model for managing public resources within VE and a role-based AC (RBAC) model for managing the sharing of an individual enterprise's private resources with VE members. Public resources are generated, used, modified, and owned by VE activities and are stored or implemented in a VE or its partners. And, private resources are owned by partners and shared with other workers who could be from different partners. This PACP language model features object–subject–action–condition AC policies that jointly determine user access authorizations. Moreover, the methodology supplies a systematical method to identify fundamental elements of the VEAC model and establish assignments between elements and relations. The proposed formal VEAC model provides VE workers with efficient management and easy access to relevant resources and up-to-date information, thus eliminating information delay and enhancing information transparency.

II. Related Works

AC systems and technologies are required to protect such resources and information from illegal access. This section surveys a number of studies related to the aims of this paper, including AC, VE, and AC policy.

A. AC

AC protects the computing system against unauthorized access or modification of information resources [12]. AC determines whether a user has rights to use a given resource; an AC system governs when and how resources can be used by whom. So far, many AC methods had been presented.

Early AC methods for resource management include AC lists (ACLs) and the AC matrix (ACM). A simple ACM is an array containing one row per subject in the system and one column per object. Entries in the matrix specify the operation or access each subject has to each object [13]. These methods are straightforward, intuitive, and only useful for small organizations [14]. ACLs implement the ACM by representing the columns as lists of users attached to a protected object. Each object is associated with an ACL that stores all subjects and the subject's approved operations for a given object. Most AC models, including mandatory AC, discretionary AC, RBAC, task-based AC, and task RBAC [15]–[17], only consider authorization management within a single organization. Furst *et al.* [18] investigated distributed RBAC to delegate administration of resources to individual departments within an enterprise. In RBAC, users are assigned roles that are associated with approved permissions for performing an operation on an enterprise resource (object) [19]. Team-based AC 2004, derived from RBAC, enables users to join team roles within an organization [20].

B. VEs

A VE is defined as a cooperative alliance in which a group of legally independent enterprises, institutions, and individuals cooperate for a particular goal [21]. Ouzounis [22] defined VE as a network of different administrative business domains that cooperate by sharing business processes and resources to provide a value-added service to customers. VE environments (Fig. 1) contain users (subjects/workers) from various enterprises, such as EMs, partners, suppliers, customers, and other VEs. VE-related activities are undertaken by users from different enterprises using collaboration and concurrence. Such a business environment results in complex AC problems. In particular, all VE resources that may be stored on and owned by different enterprises should be managed fully and should be shared as much as possible.

1) Characteristics of VEs: Kanet *et al.* [21] decomposed the life cycle of a VE into five phases, namely, identification, formation, design, operation, and dissolution. Ouzounis [22] found that the life cycle of a VE should include two major phases: establishment and management. Based on the analysis of life cycle and interactions, a VE has the following characteristics [23]–[25].

1) A VE may consist of several distributed VEs or enterprises.
2) A VE's participating members and business processes may be changed during its life cycle.
3) A VE emphasizes professional division and dynamic cooperation among a highly heterogeneous membership.
4) A VE conducts business processes of different stages across enterprises, in which each stage has its own participants, resources, and aims.
5) Various resources in a VE are shared and distributed over all participating enterprises and used by their employees (users).
6) A VE globally specifies members' obligations, responsibilities, and roles.

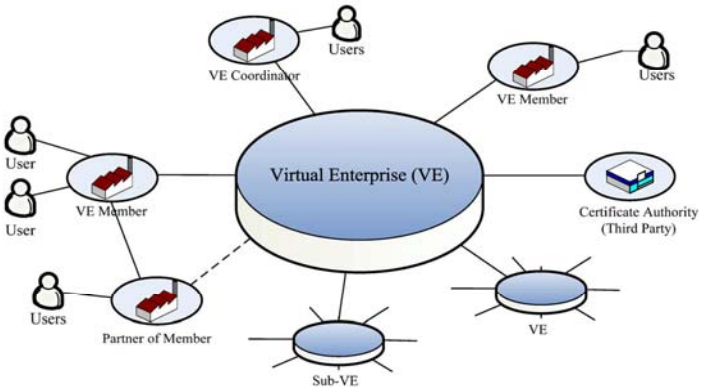

Fig. 1. VE environment.

7) A change in a member's role in a process should not affect the obligations and responsibilities in its other assigned roles.
8) Regulations do not constrain the selection of members in participating enterprises' partners.
9) Each member may own its enterprise resource management policy and AC model.
10) Shared VE resources include private resources owned by a participating enterprise and stored in its own repositories and also public resources belonging to the VE and stored in a public repository.

The levels of resource sharing among partners depend on VE characteristics, including levels of cooperation with partners, degree of trust, distributed tasks, and contractual agreements. When each participant in a VE brings information to the VE, the participant will not want to share more proprietary information than necessary with VE members because of information security issues. Information in VEs can be divided into three areas: 1) information of an individual partner brought to the VE; 2) information generated by the VE; and 3) information assets of the VE [26]. The information must be protected and distributed in a secure manner among all participants.

2) Requirement Analysis for AC in VE: Based on the general requirements of AC expressed in [27] and [28], this paper identifies the following requirements for AC model design: 1) Only the security administrator should be permitted to modify security attributes; 2) roles should be able to inherit authority either fully or partially; 3) positive authorizations and negative authorizations, as well as the principle of strict least privilege, should be supported; 4) the fine-grained authority requirements should be fulfilled; 5) access authority may change with tasks or roles; and 6) the model should be able to manage all users and resource objects in the enterprise [29]–[31].

Aside from the aforementioned requirements, according to the characteristics of VE, additional requirements must be considered when developing a VEAC model, as follows.

1) Since the organization structure of a VE is dynamic, access rights and resource objects can be changed in real time.
2) The model considers all users' access rights because resource administrators cannot predict who will access which resources in a VE.
3) As a VE is formed to achieve a certain goal in a limited time frame, each VE has different goal and business processes. A VE is always conducted as a project. Therefore, project is an essential element of AC in VE.
4) Since each enterprise has a legacy AC system, the VEAC model should be easily integrated with various AC models or policies.
5) The VE manages and shares resources collaboratively.
6) To facilitate trust among enterprises, the access policy in VE is planned and managed together by administrators of all participating enterprises.
7) The VE can maintain the consistency of policies and manage the conflicts between VE access policy and members' own access policies.

C. AC Policy

A significant shortcoming of existing AC systems is that they were developed by using a specific AC policy, which was defined by Lorch *et al.* [32], regarding how services can be utilized. AC policies are typically represented as follows: 1) constrained logic programs that support specific policy options; 2) constrained checks; and 3) administrator queries [33]. AC policy can restrict the use of services to suitably qualified principals and specify constraints that must hold when a service is invoked [19].

Recent development of AC policy framework includes languages and graphical approaches that specify different AC policies in a single framework [34]. A graph transformation-based security policy framework was proposed by Koch *et al.* [12] that included negative and positive constraints. The negative constraints specify graphs not contained in any system graph, and positive constraints specify graphs explicitly constructed in a system graph. By combining a formal framework and a logic-based language, Jajodia *et al.* [35] developed the authentication specification language that can be used to identify different AC policies that can coexist within the same system and be enforced by the same security server. Moreover, security assertion markup language is an XML framework identified by OASIS security services to exchange authentication and authorization information. For AC across enterprises, Belokosztolszki and Moody [36] proposed metapolicies. Hada and Kudo [37] proposed XML AC Language, an XML-based language for

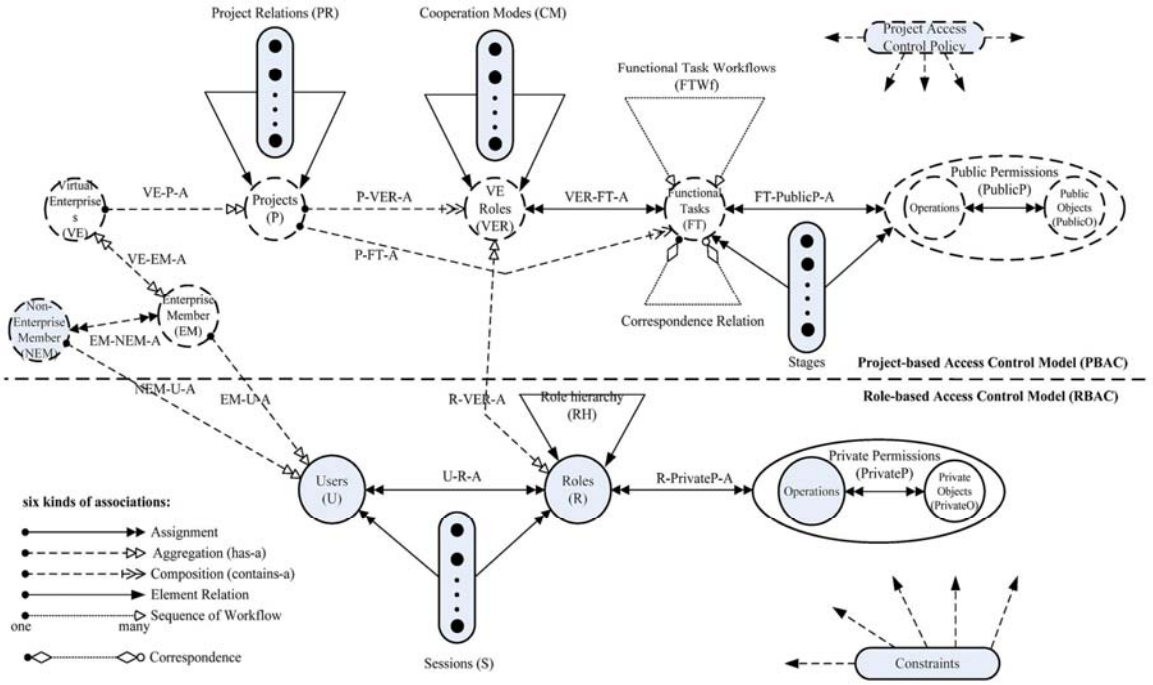

Fig. 2. Fundamental VEAC model.

provisional authorization, which articulates the security policies to be enforced for specific access to XML documents and provides XML with a sophisticated AC mechanism that enables an initiator to securely browse XML documents and securely update each document. Boella and van der Torre [38] studied normative multiagent systems for secure knowledge management based on AC policies.

III. Fundamental VEAC Model

This section introduces the proposed fundamental VEAC model, and its basic elements (Fig. 2), which has been derived from the requirements of AC for VE and characteristics of VE. It includes two submodels: one PBAC model for managing public resources stored on VE and one RBAC model for managing private resources stored on individual EMs [10]. In the model, the solid-line and the dashed-line circles are used to represent the elements in the RBAC and PBAC models, respectively. The six kinds of associations are proposed to indicate the various relationships among elements. Assignment is a well-known relationship in RBAC to continue using in the model. Aggregation is a grouping of other elements, which is also called a has-a association. For example, a VE is a group of EMs. Composition is an inclusion of other elements, which is also called a contains-a association. If the containing element is destroyed, the elements that it contains are also destroyed. Element relation is an interacting mode of other independent elements, which is further decomposed into various relations to facilitate resource sharing. Sequence of workflow is the order in which elements follow one another. Correspondence is a version mapping relation of a functional task (FT) in other project.

A. RBAC Model

This paper slightly adjusts the basic RBAC model [39]–[41] and seamlessly integrates it with the PBAC model. In the adjusted RBAC model, as shown in the bottom layer of Fig. 2, each element is described straightforwardly as follows.

1) User (U) represents a human or agent in an enterprise, which includes direct users, indirect users, and nonmember users.
2) Role (R) represents a functional job or responsibility.
3) Private object (PrivateO) denotes a resource in an enterprise associated with private privileges. Private objects are generally classified into three levels, which are public, proprietary, and protection. The public classification can be provided to the partners in a VE.
4) Private permission (PrivateP) is an approval of a particular mode of access to one or more private objects.
5) Session (S) maps a user to one or more roles.
6) $U\text{–}R\text{–}A \subseteq U \times R$ represents a many-to-many user to role assignment relation.
7) $R\text{–PrivateP}\text{–}A \subseteq R \times \text{PrivateP}$ represents a many-to-many role to PrivateP assignment relation.
8) $R_{\text{re}} = \{(x, y) : x, y \in R, x \neq y,$ and x conflicts with $y\}$ signifies that role x conflicts with role y, and x and y cannot be both assigned to the same user.
9) $R_{\text{rh}} = \{(x, y) : x, y \in R, x \neq y,$ and x is a superior of $y\}$ indicates that role x is a senior to role y, and x inherits the PrivatePs of y.
10) $U\text{–}R\text{–}A_u(r) : R \rightarrow 2^U$, a function mapping a role r to a set of users that can play this role.
11) $U\text{–}R\text{–}A_r(u)$: $U \rightarrow 2^R$, a function mapping a user u to a set of roles that can be played by this user.

12) $R-\text{PrivateP}-A_r(\text{private\_p}) : \text{PrivateP} \rightarrow 2^R$, a function mapping a PrivateP, private_p, to a set of roles that is authorized to access this PrivateP.
13) $R-\text{PrivateP}-A_{\text{private\_p}}(r) : R \rightarrow 2^{\text{PrivateP}}$, a function mapping a role r to a set of PrivatePs that allows to be accessed by this role.

B. PBAC Model

The top portion of Fig. 2 shows the PBAC model. The core concept of model development, elements, and relations in the PBAC model are introduced and defined in the following sections in order.

1) Core Concept of the PBAC Model: A VE can perform several projects (P) simultaneously, but a project can only be performed by one VE. A project includes management-level and operational-level tasks. The management-level tasks control and manage the project's progress and output according to the project timestamp, whereas the operational level comprises FTs supervised and controlled by the project schedule. Different project relations (PRs), such as subset, exclusion, and reference, exist among projects to facilitate resource sharing (refer to Section IV). Activities within a project can be divided into several FTs, each of which has access to certain public objects (PublicOs), which is public permission (PublicP) of the FT. FTs involved in a project are constructed for performing VE activities in the VE formation stage. The FTs are assigned to VE roles (VERs) that are virtual roles created based on division of labors. It is required to meet certain conditions to start or end an FT. According to the goal and task requirements, an FT can be divided into different stages by timestamp or FT. Users are given different privileges depending on the project stage and FTs. A VE is composed of several real EMs, each of which can participate in more than one VE. Non-EMs (NEMs) are enterprises that do not participate directly in the activities of VE but participate in the activities of an EM which performs directly the activities of the VE. All VE participants, including three user types (direct, indirect, and nonmember users), are generally called users (U) which may play a different role (R) in a different session. Each role has access to private resources, called a PrivateP. A superior role can inherit the privileges of inferior roles through role hierarchy (RH). The EM plays a VER through a user or role to obtain the privilege of sharing public resources in the VE and carry out practically the obligations of a given VER and to achieve the VE goals. PACP is designed to identify the resource sharing rules in a project. Through constructing relations among projects and a PACP, users can share resources among projects. The rules of sharing can be modified at any time.

To simplify the complex assignment and facilitate resource sharing across domains, some relations are gained by exploring the three viewpoints of project, VE, and enterprise. From the project viewpoint, PRs including subset, version, reference, process, and exclusive relations (defined in Section IV) are found out depending on the features of project, facilitating sharing among projects. From the VE viewpoint, cooperative relations including dependent single-task, dependent multitask, and independence (defined in Section V) are found out depending on the information requirements of interaction and cooperation among workers in VE, facilitating sharing among enterprises involved in a VE. From the enterprise viewpoint, relations proposed by RBAC [39], [40], including role hierarchy, static separation of duty, and dynamic separation of duty, are used herein to facilitate sharing among roles in an enterprise.

2) Fundamental Elements: This section concisely introduces the fundamental elements of the PBAC model, each of which is represented as follows.

1) VE = {ve: ve represents a dynamic Internet organization consisting of EMs executing a project to achieve one common business goal}.
2) EM = {em: em can be a substantive enterprise organization, a VE, or an individual, and it is a VE member with at least one worker participating directly in the VE activities}.
3) NEM = {nem: nem can be a substantive enterprise organization, a VE, or an individual, but it is not a VE member; a nem has at least one worker participating directly in the activities of EMs, and the activities have direct relations with the FT of the VE}.
4) Project (P) = {p: p denotes the set of FTs, projects, and subprojects performed by a VE}.
5) FT = {ft: ft is a set of VE activities, which have a common objective and are undertaken by several VERs}.
6) VER = {ver: ver represents a virtual role formed to enable professional division within VE, which is assigned to perform more than one FT}.
7) Object (O) = {o: o denotes an information resource including public and private resources which can be a database, entity, attribute, tuple, document, XML document, application, software component, or knowledge}.
8) PublicO = {public-o: public-o represents a subset of objects owned by a VE, stored in a VE's repository, and implemented in a VE's platform}.
9) Operation = {op: op is a set of access authorities, such as write, read, and execute}.
10) PublicP = {public-p: public-p represents a permitted mode of access to a PublicO}.
11) Permission = $\{x: x \in \text{PublicP} \cup \text{PrivateP}\}$.
12) PACP: PACP identifies which project resources are protected and shared according to the relations among projects and the shared rules and which activities are forbidden in the VE scope. Each project involves a PACP, which can be performed automatically by the VEAC system. The PACP can be dynamically created, enforced, and modified by administrators when the VE environment changes. The main rules described in PACP include the following: 1) rules of resource sharing among projects, describing the resource sharing strategy and relations among projects; 2) rules of resource usage in a project, including constraints on VERs, FTs, PublicPs, and assignments between elements; 3) rules of resource sharing of various cooperation modes, identifying the level of resource sharing according to the cooperation mode between VERs; and 4) rules of exception handling, which can be classified into rules of permitted exception handling and rules of forbidden exception handling. A PACP language model used to construct the PACP is shown in detail in Section VI.

3) Assignments and Relations: The following sections define the concept of assignments and relations between two

elements involved in the model based on the concept of a product set (refer to Definitions 1 and 2). Some functions relating to all elements in the VEAC model are defined and then applied to the following sections. These functions are shown in Appendix I.

Definition 1: Given two sets A and B, the product set or Cartesian product of A and B, called the assignment of A and B in AC domain, is $A \times B = \{(a, b) : a \in A, \text{and } b \in B\}$.

Definition 2: Given sets A and B, a binary relation R from A to B is a subset of $A \times B$, i.e., $R \subseteq A \times B$.

4) Foundational Assignments: According to Definitions 1 and 2, the various assignment relations among elements are defined as follows.

1) FT$-S-$Public$-A \subseteq$ FT $\times S \times$ Public$-A$, triple assignment among three elements: FT, S, and PubicP, FT$-S-$Public$-A$ represents the set $R_{\text{ft}-\text{s}-\text{publicp}-\text{a}} =$ {(ft, st, public-p): ft $\in$ FT, st$\in$ Stage, public-p$\in$ PublicP, the public-p is assigned to ft in stage s}.
2) $P-$VER$-A \subseteq P \times$ VER, one-to-many P to VER assignment, is denoted by $R_{\text{p}-\text{ver}-\text{a}} = \{(p,$ ver):$p \in P$, ver $\in$ VER, and p involves ver$\}$. The relation describes which VERs are included in project p.
3) VER$-$FT$-A \subseteq$ VER $\times$ FT, a many-to-many VER to FT assignment, is represented by $R_{\text{ver}-\text{ft}-\text{a}} =$ {(ver, ft): ver $\in$ VER, ft $\in$ FT, and ver performs ft}. This relation describes which FTs are undertaken by which VERs.
4) VE$-$EM$-A \subseteq VE \times$ EM, a many-to-many VE to EM assignment, is denoted by $R_{\text{ve}-\text{em}-\text{a}} =$ {(ve, em): ve $\in$ VE, em $\in$ EM, and em is a member of ve}.
5) VE$-P-A \subseteq$ VE $\times P$, one-to-many binary assignment from a VE to P, is represented by $R_{\text{ve}-\text{p}-\text{a}} =$ {(ve, p): ve $\in$ VE, $p \in P$, and ve performs p}. This relation records which project is performed by a VE.
6) EM$-$NEM$-A \subseteq$ EM $\times$ NEM, many-to-many EM to NEM assignment, is represented by $R_{\text{em}-\text{nem}-\text{a}} =$ {(em, nem): em $\in$ EM, nem $\in$ NEM, and nem supports em to perform some tasks of the VE$-$EM$-A_{\text{ve}}$(em)}. This relation holds the assignments between EMs and its partners (NEMs) to support the tasks of a VE.
7) FT workflow (FTWf) $\subseteq$ FT $\times$ FT, many-to-many binary relation on FT, is denoted by $R_{\text{FTWf}} = \{(\text{ft}_i, \text{ft}_j) : \text{ft}_i,$ $\text{ft}_j \in$ FT, p_i, $p_j \in P$, $\text{ft}_i \subset p_i$, $\text{ft}_j \subset p_j$, $i \neq j$, ft_i is an event FT of the action FT $\text{ft}_j\}$ that indicates that ft_j is authorized to use the PublicPs of ft_i when ft_i is accomplished.
8) Correspondence $\subseteq$ FT $\times$ FT, one-to-one binary relation on FT, is represented by $R_{\text{correspondence}} = \{(\text{ft}_i, \text{ft}_j) : \text{ft}_i,$ $\text{ft}_j \in$ FT, p_i, $p_j \in P$, $\text{ft}_i \subset p_i$, $\text{ft}_j \subset p_j$, $i \neq j$, ft_i is the preversion offt_j, whereas ft_j is the postversion of$\text{ft}_i\}$.

5) Assignments Across Models: This section defines the assignment relations across models in order to establish the combination relations of relevant elements among two AC models. These relations are as follows.

1) EM$-U-A \subseteq$ EM $\times U$, one-to-many EM to U assignment, is represented by $R_{\text{em}-\text{u}-\text{a}} = \{(\text{em}, u)$: em $\in$ EM, $u \in U$, and em have an employee $u\}$. If $\exists$ $\text{em}_1 R_{\text{em}-\text{u}-\text{a}} u_1$, $\text{em}_2 R_{\text{em}-\text{u}-\text{a}} u_2$, $\text{em}_1, \text{em}_2 \in$ EM, and $u_1, u_2 \in U$, then $\neg\exists$ $\text{em}_2 R_{\text{em}-\text{u}-\text{a}} u_1$.
2) NEM$-U-A \subseteq$ NEM$\times U$, one-to-many NEM to U assignment, is denoted by $R_{\text{nem}-\text{u}-\text{a}} =$ {(nem, u): nem $\in$ NEM, $u \in U$, and nem have an employee $u\}$. If $\exists$ $\text{nem}_1 R_{\text{nem}-\text{u}-\text{a}} u_1$, $\text{nem}_2 R_{\text{nem}-\text{u}-\text{a}} u_2$, nem_1, $\text{nem}_2 \in$ NEM, and u_1, $u_2 \in U$, then $\neg\exists$ $\text{nem}_2 R_{\text{nem}-\text{u}-\text{a}} u_1$.
3) $R-$VER$-A \subseteq R \times$ VER, many-to-many R to VER assignment, is represented by $R_{\text{r}-\text{ver}-\text{a}} = \{(r,$ ver): $r \in R$, ver $\in$ VER, and r is assigned to play ver$\}$, then VER ver can be assigned to different roles, whereas one role can play different VERs at the same time.

IV. PRs

A PR (R_{p}) indicates the level of information exchange and reuse and also the situation of cooperation between two projects. Various PRs describing the relation between two projects can propagate the authorizations of an FT to other FTs. Different PRs may occur between two projects and may alter with time based on project management and share requirements. While a VEAC-based AC platform is implemented, administrators construct the project resource access strategy in a PACP to indicate the level of resource sharing of each type of PRs. In the project life cycle, the PRs and the PACP can be modified at any time to respond to the demands of resource sharing. Resource sharing or reusing is determined based on five attributes of each FT: 1) FT state (A_{state}) holds the status of the FT being performed; 2) FT stage (A_{stage}) records the current timestamp of an FT for appropriate resource sharing according to its states; 3) allowed reference (A_{ref}) decides whether the FT can be referred by relative FT in a postversion project; 4) allowed subproject ($A_{\text{sub}-\text{p}}$) determines whether the FT can be referred by its subprojects; and 5) allowed main project ($A_{\text{main}-\text{p}}$) decides whether the FT can be referred by its main project.

To introduce the PRs, given a set Project (P) and x, $y \in P$, a binary relation PR (R_{p}) on P is a subset of $P \times P$, which is distinguished into five subrelations presented in the following sections. For convenience in the following discussion, two inherited functions applied in the following sections are defined to indicate varying degrees of privilege inheritance.

1) Strong-inherited function $\text{Inher}_{\text{strong}}$(ft) is defined as all permissions assigned to the ft are inherited, including read (to retrieve data), update (to modify data), insert (to write new data), and create (to create an object).
2) Weak-inherited function $\text{Inher}_{\text{weak}}$(ft) is defined as only read permission from the ft is inherited.

A. Subset Relation

Subset relation (R_{ps}) describes the relation between a main project and its subproject. The relation simplifies a large number of assignments. For instance, an FT called announcement shows information about the progress of a project. Through the subset relation, all workers in the main project and subprojects of the project are permitted to look up the progress of the project. The set of pairs of projects between which have subset relation is represented by $xR_{\text{ps}}y = \{(x, y) : x, y \in P, x \neq y,$ and x "is a subset of" $y\}$. A main project is permitted to access the resources of its subproject, but an administrator may set or disable the capability by changing the status of the allowed main-project attribute of its each FT. Fig. 3 shows an example of the subset relation to demonstrate these constraints,

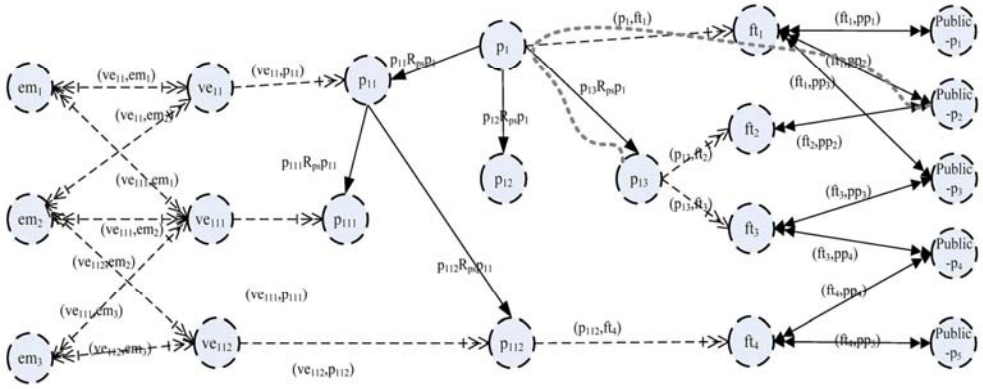

Fig. 3. Example of subset relation.

where Project p_1 involves three subprojects p_{11}, p_{12}, and p_{13}, and project p_{11} is further decomposed into subprojects p_{111} and p_{112}. Owing to $p_{13}R_{\rm ps}p_1$, PublicPs, such as public-p_2 and public-p_3, are assigned to p_1 and p_{13} via FTs, whereas EMs, such as $\rm em_1$ and $\rm em_2$, are permitted to participate in $\rm ve_{11}$ and $\rm ve_{111}$. Two functions $\rm privilege_{main-p}(ft)$ and $\rm privilege_{sub-p}(ft)$ are defined, respectively, in (1) and (2), shown at the bottom of the page, for propagating user's privilege from the main project and subproject, respectively, where variables are introduced as follows. Function (1) indicates that the privileges of $\rm ft_{1\mathit{i}}$ involve the PublicP and PrivateP assigned to the $\rm ft_{1\mathit{i}}$ and $\rm ft_{2\mathit{j}}(1 \le \mathit{j} \le \mathit{n})$ when the conditions shown in the equation hold; otherwise, the privileges of $\rm ft_{1\mathit{i}}$ only have the PublicP and PrivateP from $\rm ft_{1\mathit{i}}$. Due to the limited space, function (2) shows the propagation of user privileges from subproject, which is similar to function (1) and is not further introduced in detail. p_1 is the main project of p_2 that is the subproject of p_1, $\rm ft_{1\mathit{i}}$'s are the FTs involved in p_1, $1 \le i \le m$, and $\rm ft_{2\mathit{j}}$'s are the FTs involved in p_2, $1 \le j \le n$. Several constraints are applied to use a subset relation: 1) A main project may have more than one subproject; 2) a subproject is only involved in one main project; 3) an EM may participate in the main and subprojects; and 4) a PublicP is only permitted to be assigned to different projects with subset relations.

B. Version Relation

Version relation ($R_{\rm pv}$) describes a project y called a postversion project that is extended from a project x called preversion project and planned with reference to the preversion project. Therefore, the pre- and postversion projects have similar targets, FTs, and participants. The relation helps support version-dependent authorizations by enabling the reuse of resources for a new product, thus reducing its time to market. Because the pre- and postversion projects have similar targets, activities, and participants, the postversion FT in the postversion project corresponds to the preversion FT in the preversion project. While the postversion FT is performed, the privileges owned by the preversion FT are inherited by the postversion FT using the weak inheritance. The set of pairs of projects between which have version relation is represented by $xR_{\rm pv}y = \{(x, y) : x, y \in P, x \neq y,$ and x "is the preversion of" $y\}$. Fig. 4 shows an example of the version relation, which demonstrates that project p_1 is the preversion of project p_2. Project p_1 for developing a car engine consists of FTs $\rm ft_{11}$ and $\rm ft_{12}$, whereas p_2 for developing a new engine based on the engine developed by p_1 comprises $\rm ft_{21}$, $\rm ft_{22}$, and $\rm ft_{23}$. FTs $\rm ft_{11}$ (requirement analysis) and $\rm ft_{12}$ (conceptual design) correspond to $\rm ft_{21}$ (requirement analysis) and $\rm ft_{22}$ (conceptual design), respectively, whereas $\rm ft_{23}$ (primary design) is created for another task, which is not extended from p_1. Therefore, while the $\rm ft_{21}$ performed, workers must refer significantly to information owned by $\rm ft_{11}$. Due to $p_1 R_{\rm pv} p_2$, each FT in project p_2 is performed by VERs, which are allowed to refer to PublicPs of corresponding FTs in p_1 if the attribute allowed reference of corresponding FT is true. As shown in Fig. 4, a user u_1 is assigned to perform the $\rm ft_{21}$ through (u_1, r_1), $(r_1, \rm ver_{21})$, and $(\rm ver_{21}, ft_{21})$; in addition to the public-p_{21} and public-p_{22}, u_1 may refer to the public-p_{11}, public-p_{12}, and public-p_{13}. Function (3) shown at the bottom of the next page is presented to indicate that the privileges of $\rm ft_{2\mathit{j}}$ involve the PublicP and PrivateP assigned to the $\rm ft_{2\mathit{j}}$, and partial PublicPs of the corresponded FT $\rm ft_{1\mathit{i}}$ of $\rm ft_{2\mathit{j}}$ through the use of weak inheritance function when the conditions shown in the function hold; otherwise, the privileges of $\rm ft_{2\mathit{j}}$ only have the PublicP and PrivateP from $\rm ft_{2\mathit{j}}$. p_1 is the preversion project of p_2 that is the postversion project of p_1, $\rm ft_{1\mathit{i}}$'s are the FTs involved in p_1, and $\rm ft_{2\mathit{j}}$'s corresponding to $\rm ft_{1\mathit{i}}$'s are the FTs involved in p_2. Several constraints are applied when using the version relation to support resource sharing between two projects: 1) A postversion project has less than one preversion

$$\text{privilege}_{\text{main-p}}(\text{ft}_{1i}) = \begin{cases} \text{FT-Permission-}A(\text{ft}_{1i}) \cup \text{FT-Permission-}A(\text{ft}_{2j}) & \text{if } \exists\, p_1 R_{\text{ps}} p_2 \wedge A_{\text{main-p}} \text{ of } \text{ft}_{2j} = \text{“true”} \\ \text{FT-Permission-}A(\text{ft}_{1i}) & \text{otherwise} \end{cases} \quad (1)$$

$$\text{privilege}_{\text{sub-p}}(\text{ft}_{2j}) = \begin{cases} \text{FT-Permission-}A(\text{ft}_{2j}) \cup \text{FT-Permission-}A(\text{ft}_{1i}) & \text{if } \exists\, p_1 R_{\text{ps}} p_2 \wedge A_{\text{sub-p}} \text{ of } \text{ft}_{1i} = \text{“true”} \\ \text{FT-Permission-}A(\text{ft}_{2j}) & \text{otherwise} \end{cases} \quad (2)$$

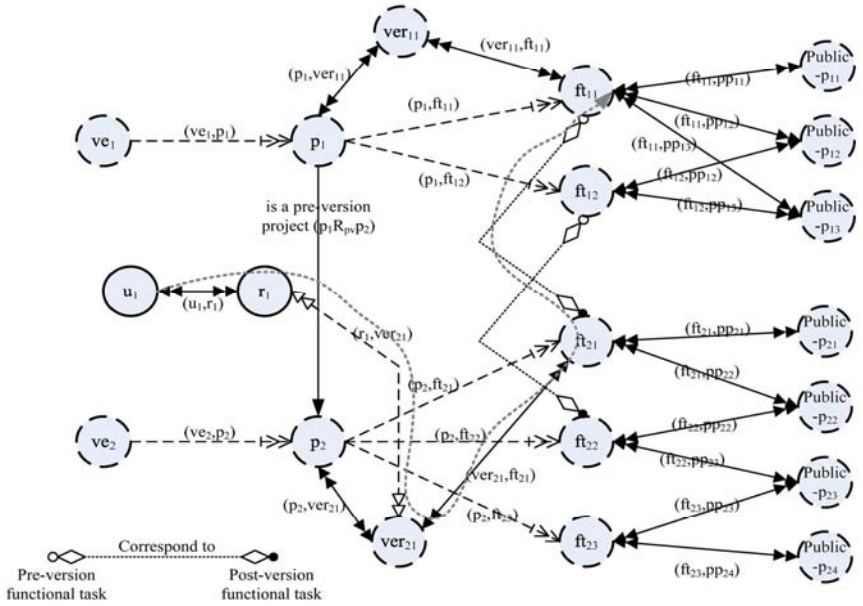

Fig. 4. Example of version relation.

project contrariwise; 2) an FT has less than one corresponding FT; and 3) an EM may participate in pre- and postversion projects simultaneously.

C. Reference Relation

Reference relation ($R_{\rm pr}$) describes a project x called referring project referring to resources in other project y that is called the referred project. The reference relation indicates that the same users and enterprises can participate in both the referring and referred projects. If two projects have a reference relation, then users in the referring project can refer to the resources of the referred project. While the value of attribute allowed reference of an FT equals true, then the FT can be referred. The set of pairs of projects between which are referred by each other is represented by $xR_{\rm pr}y = \{(x,y) : x,y \in P, x \neq y,\ x$ refers to resources in y, and $(\neg\exists\, xR_{\rm pe}y) \wedge (\neg\exists\, yR_{\rm pe}x)\}$. Project x may refer to y if and only if the following conditions hold: $R_{xi} \cap R_{yj}$, $\text{EM}_{xm} \cap \text{EM}_{yn}$, $\text{FT}_{xk} \cap \text{FT}_{yh}$, $\text{PublicP}_{xv} \cap \text{PublicP}_{yw}$, and $\text{PrivateP}_{xe} \cap \text{PrivateP}_{yf}$ permit unequal ϕ, where R_{xi}, EM_{xm}, FT_{xk}, PublicP_{xv}, and PrivateP_{xe} are associated with project x, and R_{yj}, EM_{yn}, FT_{yh}, Public-P_{yw}, and Private-P_{yf} are associated with project y. That is, roles, EMs, FTs, PublicP, and PrivateP may be assigned to p_1 and p_2. Fig. 5 shows an example of the reference relation, which indicates that project p_1 can refer to project p_2 through the reference relation $p_1R_{\rm pr}p_2$. Role r_{31} is assigned to perform VERs ver_{11} and ver_{21}, ver_{11} performs FTs ft_{11} and ft_{12} in project p_1, and ver_{21} performs ft_{21} in project p_2. Therefore, user u_{31} may utilize the public-p_{11}, public-p_{12}, public-p_{13}, public-p_{21}, and public-p_{22} through (u_{31}, r_{31}). The following constraints are applied when using the reference relation: 1) A project may be assigned to more than one project for resource sharing, and 2) a project may refer to more projects simultaneously.

D. Process Relation

Process relation ($R_{\rm pp}$) describes the executive sequence of two subprojects from the time view and can determine the time for sharing project resources. A process relation can be applied to determine the executive sequence of all subprojects of a project. The set of pairs of projects between which have process relation is represented by using $xR_{\rm pp}y = \{(x,y) : x,y,z \in P, x \neq y \neq z, (\exists\ xR_{\rm ps}z) \wedge (\exists\ yR_{\rm ps}z)$, and x "must be achieved, then start" $y\}$. While the relation is built on two projects, the administrator must specify the sequences of related FTs across the project boundary. This relation can support process-dependent authorization propagation when executing an action FT that can use the resources of the event FTs in event project. Fig. 6 shows an example of a process relation, in which project p_1 denotes the event project of action project p_2; p_1 performs ft_{11} and ft_{12}, and p_2 performs ft_{21}, ft_{22}, and ft_{23}; and ft_{11} denotes an event FT that triggers the ft_{21} and ft_{22} (called action FTs). When ft_{21} is

$$\text{privilege}_{\rm version}(\text{ft}_{2j}) = \begin{cases} \text{FT}-\text{Permission}-A(\text{ft}_{2j}) \cup \text{Inher}_{\rm weak}\,(\text{FT}-\text{PublicP}-A_{\text{public\_p}}(\text{ft}_{1i})) & \text{if } \exists\, p_1R_{\rm pv}p_2 \wedge A_{\rm ref} \text{ of } \text{ft}_{1i} = \text{“true”} \\ \text{FT}-\text{Permission}-A(\text{ft}_{2j}) \text{ otherwise} \end{cases} \quad (3)$$

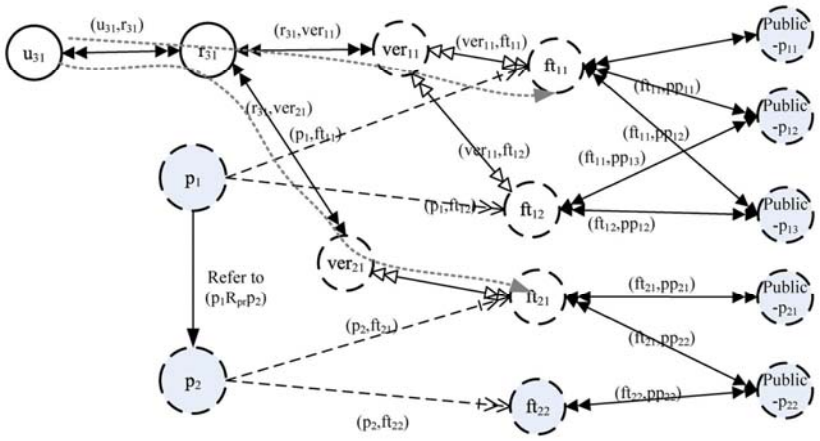

Fig. 5. Example of reference relation.

triggered and performed, user u_1 obtains authorizations public-p_{11}, public-p_{12}, and public-p_{13} from ft_{11}, and authorizations public-p_{21} and public-p_{22} from ft_{21}.

Function (4) shown at the bottom of the page, showing the propagation of user privilege by using process relation, is presented to indicate that the privileges of $\text{ft}_{\text{action}}$ involve the PublicP and PrivateP assigned to the $\text{ft}_{\text{action}}$ and also partial permissions of the event FT ft_{event} of the $\text{ft}_{\text{action}}$ through the use of the weak inheritance function when the conditions shown in the function hold; otherwise, the privileges of $\text{ft}_{\text{action}}$ only have the PublicP and PrivateP from $\text{ft}_{\text{action}}$. p_1 is the action project of p_2 that is the event project of p_1, $\text{ft}_{\text{action}}$ is the action FT included in p_1, and ft_{event} is the event FT included in p_2. Using the process relation must obey the following constraints: 1) A process relation exists between two projects which must have the subset relation; 2) an event project may trigger more than one action project simultaneously; 3) an event FT may trigger more than one action FT simultaneously; and 4) an action project may be triggered if all of its event projects are accomplished.

E. Exclusive Relation

Exclusive relation (R_{pe}) identifies mutual conflict between two projects, signifying that the resources of the two projects cannot refer to each other. The exclusive relation is default. That is, two projects are preset as exclusive relation if no other relation exists between them. The set of pairs of projects that conflict with each other is represented by $xR_{\text{pe}}y = \{(x, y) : x, y \in P, x \neq y, x$ "conflicts with" y, and $(\neg\exists\ xR_{\text{pr}}y) \wedge (\neg\exists\, yR_{\text{pr}}x)\}$. If two projects are exclusive, then all users, EMs, FTs, and permissions in a project are exclusive with the other project. That is, an enterprise is disallowed from participating simultaneously in two projects with exclusive relation; attempts by users of the exclusive projects to use the same resources are rejected. Using the process relation must obey the following constraints: 1) A project may conflict with more than one simultaneously; 2) a PublicP may not be assigned to two exclusive projects; and 3) an EM is not allowed to be assigned to two mutual exclusive projects.

V. Cooperation Modes Among Two VERs

This section introduces three cooperation modes among VERs based on the resource sharing requirements of collaborative operations in the VE.

Cooperation mode (R_{c}) describes interactions among VERs according to the dependent level of their duties. Given a set VER, x and $y \in$ VER, a binary relation cooperation relation (R_{c}) on VER is a subset of VER $\times$ VER, which is distinguished into three cooperation relations. For convenience in the following discussion, two items are first defined in terms of authority inheritance. A VER in cooperative mode can inherit strongly or weakly the privileges from the other VER. Strong inheritance means that the privilege of a VER can be fully inherited by the other VER, whereas weak inheritance means that the privilege can only be partially inherited, such as only inheriting read privilege.

1) Dependent single-task mode (R_{cds}) is the most seamless cooperative relationship between two VERs, working together to perform FTs, that have dependencies and share resources with each other. The two VERs' permissions are inherited from each other via strong inheritance (defined in Section IV). When two VERs collaboratively perform different FTs, the users playing the two VERs obtain the same permissions from the FTs. The set of pairs of VERs with R_{cds} is represented by using $xR_{\text{cds}}y = \{(x, y) : x, y \in \text{VER}, x \neq y, \exists(x, \text{ft}_1), (y, \text{ft}_1) \in$

$$\text{privilege}_{\text{process}}(\text{ft}_{\text{action}}) = \begin{cases} \text{FT}-\text{Permission}-A(\text{ft}_{\text{action}}) \cup \text{Inher}_{\text{weak}}\,(\text{FT}-\text{Permission}-A(\text{ft}_{\text{event}})) \\ \text{if } \exists(p_1 R_{\text{pr}} p_2) \wedge (A_{\text{state}} \text{ of } \text{ft}_{\text{event}} = \text{“achieved”}) \wedge (A_{\text{ref}} \text{ of } \text{ft}_{\text{event}} = \text{“true”}) \\ \text{FT}-\text{Permission}-A(\text{ft}_{\text{action}}) \text{ otherwise} \end{cases} \tag{4}$$

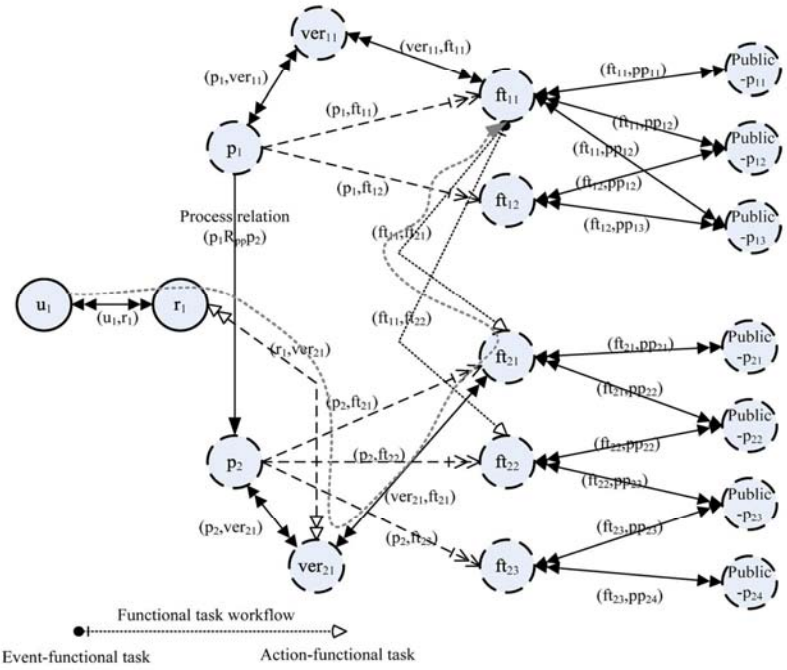

Fig. 6. Example of process relation.

VER−FT−$A \rightarrow$ FT−PublicP−$A_{\text{public\_p}}$({VER − FT − A_ft$(x) : (x, \text{ft}) \in$ VER−FT−A}) are inherited strongly by VER y, and FT−PublicP−$A_{\text{public\_p}}$({VER−FT−A_ft$(y) : (y, \text{ft}) \in$ VER−FT−A}) are inherited strongly by VER x, and $(\neg\exists\ xR_{\text{cdm}}y) \wedge (\neg\exists\ yR_{\text{cdm}}x) \wedge (\neg\exists\ xR_{\text{ci}}y) \wedge (\neg\exists\ yR_{\text{ci}}x)\}$ means that VERs x and y cooperate to perform an FT ft_1 and have the same access privilege to all its resources.

2) Dependent multitask mode (R_{cdm}) indicates that two VERs interact when performing different FTs. For instance, the results of an FT performed by a VER affect those of an FT performed by another VER. The two VERs inherit each other's permissions via weak inheritance. The set of pairs of VERs with R_{cdm} is represented by using $xR_{\text{cdm}}y = \{(x, y) : x, y \in \text{VER}, x \neq y, \forall(x, \text{ft}_x),$ $(y, \text{ft}_y) \in$VER−FT−$A \rightarrow$FT−PublicP−$A_{\text{public\_p}}$ ({VER−FT−$A_{\text{ft}}(x) : (x, \text{ft}_x) \in$ VER−FT−A}) are inherited weakly by VER y, and FT−PublicP−$A_{\text{public\_p}}$ ({VER−FT−$A_{\text{ft}}(y) : (y, \text{ft}_y) \in$ VER−FT−A}) are inherited weakly by VER x, and $(\neg\exists\ xR_{\text{cds}}y) \wedge (\neg\exists\ yR_{\text{cds}}x) \wedge (\neg\exists\ xR_{\text{ci}}y) \wedge (\neg\exists\ yR_{\text{ci}}x)\}$. Hence, VERs x and y perform related FTs separately, and that outputs of the FTs are referred to each other.

3) Independent mode (R_{ci}) indicates that two VERs independently perform their FTs, disregarding the outputs generated by other FTs. The relation is applied to protect business secrets when companies that compete with each other perform VERs. If the two VERs work independently, then they are not permitted to perform the same FTs and have each other's access privileges for FTs performed by them. The set of pairs of VERs between which have R_{ci} is represented by $xR_{\text{ci}}y = \{(x, y) : x, y \in$ VER, $x \neq y, T$−PublicP−$A_{\text{public\_p}}$ ({VER−FT−$A_{\text{ft}}(x) : (x, \text{ft}_x) \in$ VER−FT−A}) are not inherited by VER y, and FT−PublicP−$A_{\text{public\_p}}$({VER−FT−$A_{\text{ft}}(y) : (y, \text{ft}_y) \in$ VER−FT−A}) are not inherited by VER x, and $(\neg\exists\ xR_{\text{cds}}y) \wedge (\neg\exists\ yR_{\text{cds}}x) \wedge (\neg\exists\ xR_{\text{cdm}}y) \wedge (\neg\exists\ yR_{\text{cdm}}x)\}$.

The use of cooperative relations is constrained by the following rules.

1) $\#(\{y : (x_1, y) \in R_c, x_1, y \in \text{VER}\}) \geq 0$ means that a VER x_1 is permitted to have different cooperation modes with other VERs.
2) $\#(\{(x_1, y_1) : (x_1, y_1) \in R_c, x_1, y_1 \in \text{VER}\}) \leq 1$ signifies that only one cooperation mode is permitted between two VERs.

VI. PACP Language Model

Based on the VEAC model, the PACP language model for VEs designed in this paper, as Fig. 7 shows, is represented in class model of Unified Modeling Language (UML) and mainly targets contents of information text. This model features an object–subject–action–condition AC policy consisting of multiple sets of authorization rules that jointly determine user access permissions. Therefore, regarding specific resource (object), authorization (action) to execute certain resource is granted to certain users (subject) under certain restrictions (conditions).

The PACP language model for VEs has been proposed in this section for the following reasons: 1) to provide a method that effectively describes resource AC policy for VEs; 2) to reduce costs and complexity in resource AC; 3) to improve flexibility in managing access permission; and 4) to make the management of resource access permission adaptive to changing needs in a

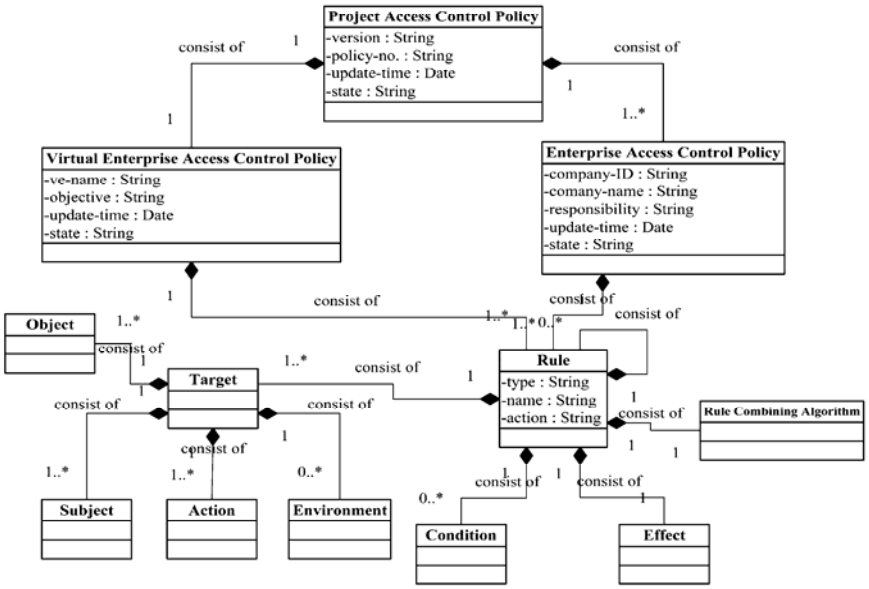

Fig. 7. PACP language model.

business environment in a timely manner. The PACP language model has the following main components.

1) PACP. A PACP consists of one VEAC policy (VEACP) and many enterprise AC policies (EACPs), which are sets of rules.
2) VEACP, a set of rules, describes the regulation and constraint on resource AC and sharing in a VE to manage the VE's resource.
3) EACP, consisting of a series of rules, describes rules and conditions for enterprise resource AC for each EM. Its rules shall not be in conflict with the VEACP it belongs to and must comply with the sharing rules agreed upon by VE so to make available resource in need of sharing.
4) Rule element is the most basic unit of policy and corresponds to the conventional concept of authorization. The principal components of rule have a target, effect, condition, and rule combining algorithm. Each rule permits or denies one or more subjects to performing actions on one or more objects under some conditions.
5) A target element involved in a rule defines the set of objects, subjects, and actions to which the rule or policy applies.
6) Object may be data, information, and knowledge owned by the VE or one of its EMs.
7) A subject is an actor whose attributes may be referenced by a predicate. Actor may be a user, role, enterprise, or VER.
8) An action is an operation on resource.
9) A condition element represents additional constraints that further refine rule applicability.
10) Rule combining algorithm compresses the output from the embraced rules. The PACP language model has four rule combining algorithms: deny overrides, permit overrides, first applicable, and only-one-applicable. Based on the selected combining algorithm, an authorization decision can be permit, deny, not applicable, or indeterminate.
11) Effect is the intended consequence of a satisfied rule— either Permit or Deny.

VII. VEAC Model Construction Methodology

The proposed formal VEAC model can efficiently manage and share information resources in the VE life cycle. To assist the administrators of VEs and their EMs to successfully implement the proposed fundamental VEAC model and to use the PACP language model appropriately for VE information resource security and sharing, this section develops a VEAC model construction methodology based on the five phases of VE life cycle, namely, identification, formation, design, operation, and dissolution phases. The methodology provides the security administrators of the leader and partners of VEs with a systematic method for the following reasons: 1) to identify the fundamental elements of VEAC model, such as P, VER, FT, U, R, PublicP, and PrivateP; and 2) to establish assignments between elements, PRs between projects, and cooperation modes between VERs. The VEAC model applied for certain VE is initially planned at the formation phase, all elements and assignments of the VEAC model are designed at the design phase, and the constructed VEAC model is implemented at the operation phase. Thus, information resources are managed at the operation and dissolution phases. The goal, procedure, inputs, outputs, and related method and technologies of each phase of the methodology are separately introduced in the following sections.

A. Identification and Formation Phases

Fig. 8 shows the first two phases in the proposed methodology, namely, identification and formation phases, which are

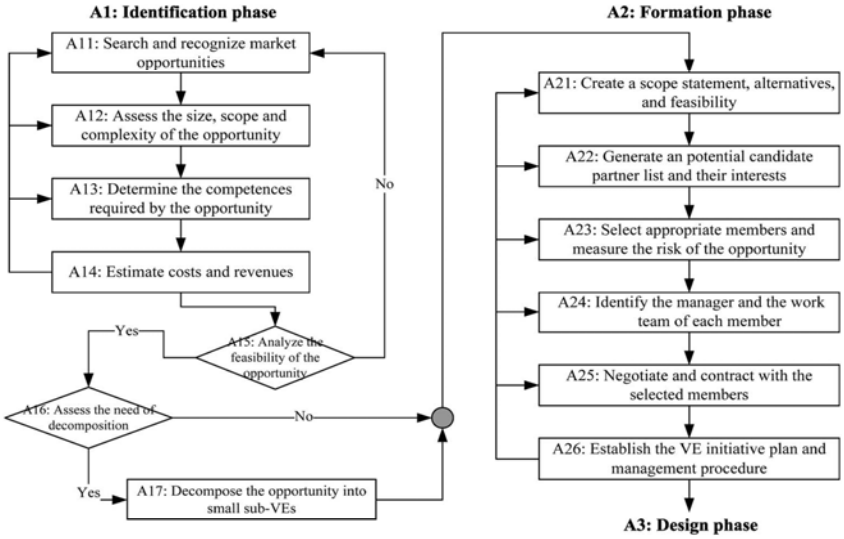

Fig. 8. Identification and formation procedures of a VE.

introduced simply as follows.

1) Identification phase numbered A1 defines the boundaries of a VE to analyze whether the goals, technologies, and cost of the VE are acceptable; to evaluate the complexity of the VE; and to establish procedures for supporting later VE activities. The leader of a VE generally analyzes historical transaction data or carries out market research to find out a valuable and feasible market opportunity and then to form a VE. To achieve the aims of the identification phase, the following seven numbered actions (the left of Fig. 8) should be undertaken in order or repetitively: (A11) searching and recognizing market opportunities; (A12) assessing the size, scope, and complexity of the opportunity; (A13) determining the competences required by the opportunity; (A14) estimating costs and revenues; (A15) analyzing the feasibility of the opportunity; (A16) assessing the need of opportunity decomposition; and (A17) decomposing the opportunity into small sub-VEs to perform the decomposed opportunities, thus establishing $R_{\rm ps}$ between the main and subprojects. The final output of the phase is a practical and valuable opportunity.
2) Formation phase numbered A2 selects suitable partners against alignment factors for their skills, experiences, and capabilities; identifies each member's responsibilities explicitly; ensures that every member of the VE understands his own individual roles and responsibilities; and allocates project resources, including people, service, facilities and equipment, supplies and materials, and money. To accomplish this process at the formation phase, the following six numbered actions (the right of Fig. 8) should be executed in order or repetitively: (A21) creating the scope statement, alternatives, and feasibility of a VE; (A22) generating a potential candidate partner list and their interests; (A23) selecting appropriate partners for the VE and its sub-VEs and measuring the possible risk from the partners; (A24) identifying the VE manager and work team of each partner; (A25) negotiating and contracting with the selected partners for sharing and using resources; and (A26) establishing the initiative plan and management procedure of a VE and its sub-VEs. The final outputs of achieving the six actions include a certain VE organizational structure model and contracts for cooperation among all EMs. The design phase is then executed based on this model.

B. Design Phase

The design phase in the proposed methodology is a significant phase for constructing a real VE based on the proposed VEAC model, since it is relative mostly to the plan and design, and resource use and assignment of VEs (Fig. 9). The actions involved at the phase are achieved collaboratively by the security administrators of the VE leader and all partners for managing public and private resources and VE user authorizations. The design phase numbered A3, which includes three subprocedures A31, A32, and A33, is described as follows.

1) Subprocedure A31—Plan and Design VE: The subprocedure models a VE in terms of organization, business, process, and activity perspectives. The detailed organizational structure model, business and resource sharing regulations, VE process model, and activity models of each EM are produced at the end of subprocedure A31. The subprocedure involving six actions is numbered and described below.

(A311) Identify all participators of each partner. Each partner in the VE is assigned certain tasks or responsibilities at the formation phase. The subprocedure starts with action A311 from the organizational view, in which each partner has to choose suitable employees or teams to perform enterprise-assigned tasks, according to employees' skills, experiences, and capabilities. At the time, partners must offer a list of employees who participate directly or indirectly in the VE and are permitted to access the VE resources. The employees involved in the list become user elements in the VEAC model.

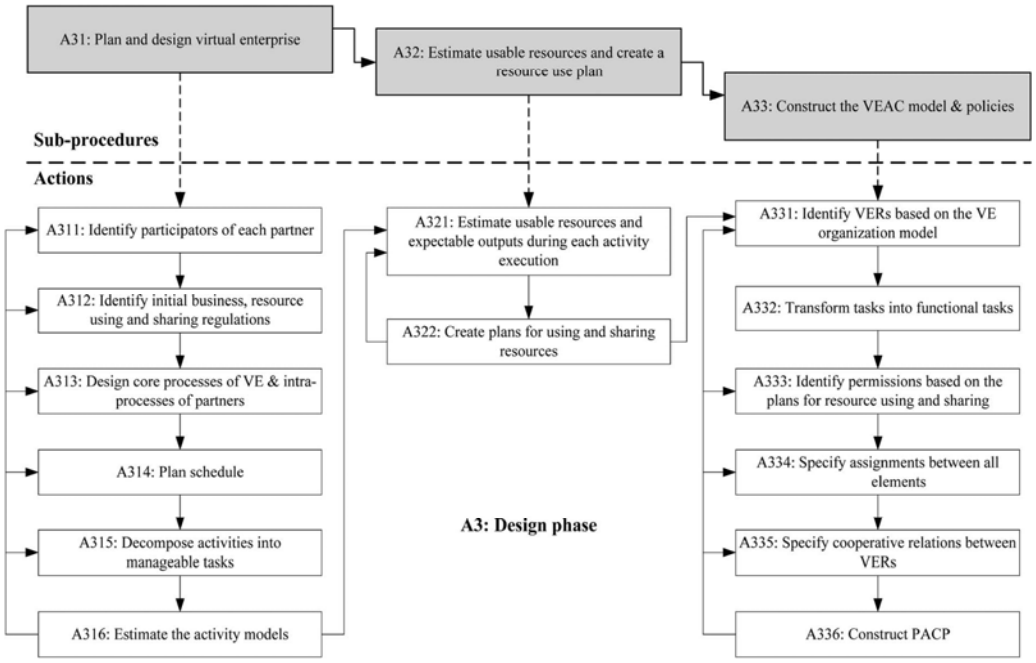

Fig. 9. Design procedure of a VE based on the VEAC model.

(A312) Identify initial business and resource using and sharing regulations. From a business perspective, action A312 identifies regulations regarding usage and sharing of resources to restrict the behavior of partners and specify each participator's responsibility and obligations. In the VE organizational structure model, every participator in a VE is assigned certain tasks, which are performed and restricted by the regulations. The regulations are then converted into VEACP and EACPs by A336.

(A313) Design the core processes of VE and the intraenterprise processes of partners. Based on the planned VE process at A2, a VE leader at the phase designs the core processes of the VE project represented by a project evaluation and review technique chart. The core processes are composed of many activities to accomplish VE's goal. Each activity in the core process is assigned to certain partners to perform. Each partner must then spread up and perform its assigned activities and integrate them into its intraenterprise processes. Finally, PRs R_{pv}, R_{pr}, R_{pp}, and R_{pe} can be established at action A313 if they are needed.

(A314) Plan schedule. According to the core VE processes designed by A313, the VE leader at the action negotiates and communicates with partners to plan the start and end times of each activity in the core VE and intraenterprise processes, and the activity prerequisites.

(A315) Decompose activities into manageable tasks. The activities involved in the core VE and intraenterprise processes are further decomposed into tasks until every task can represent a manageable amount of work that can be planned, scheduled, and assigned. A work breakdown structure, comprising a hierarchical decomposition of project, activities, and tasks, is planned at this point. The decomposed tasks are then further decomposed or combined into manageable tasks in terms of resource AC. The priority of every manageable task is determined from the start and end times of the original tasks, the information flow between tasks and task outputs.

(A316) Estimate the activity models. An activity model is composed of some partially ordered tasks that are conducted to achieve the actions to be performed within a VE. Action A316 estimates the duration of every task and changes the baseline based on reasonable estimations. The following factors should be addressed: 1) the resources that should be used; 2) the amount of time required; 3) how many people are needed; 4) the skills that are necessary; and 5) the tasks that need to be completed before other tasks are started. Subprocedure 2 is executed after all tasks are estimated.

2) Subprocedure A32—Estimate Usable Resources and Create a Resource Use Plan: Subprocedure A32 estimates the usable VE resources and builds a resource use plan for the entire life cycle of a VE. The plan is adopted to restrict assignments between elements and to build the PACP.

(A321) Estimate usable resources and expectable outputs. The first action of this subprocedure estimates usable VE resources according to the activity models outputted by A316. These resources include public and private resources, which are supplied or shared with partners to facilitate the execution of VE tasks. In addition, the administrator has to expect possible outputs during the execution of each task and know whom the outputs will be shared with. Some shared outputs should be specified by specific data containers, which are then converted into permissions and assigned operations permitted on them at A333.

(A322) Create plans for using and sharing resources. Based on the regulations created by A312, the result of A321, and

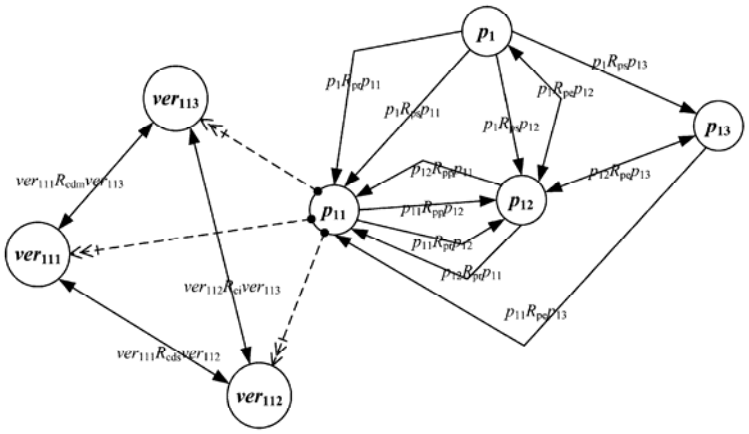

Fig. 10. Engine R&D project.

the activity models of A316, the action forms a resource using and sharing plan, which describes which users, activities, and partners can use or share which resource.

3) Subprocedure A33—Construct the VEAC Model and Policies: Subprocedure A33 identifies the elements, assignments, relations, and policies involved in the VEAC model according to the outputs at subprocedures A31 and A32.

(A331) Identify VERs based on the VE organizational model. VERs can be identified by using two different methods: decomposing the VE's goal or decomposing the VE's organizational structure. In the first method, the aim of a VE is decomposed into little goals that can be completed by a single individual or team. Each of these little goals is then transformed into a VER. In the second method, the organizational structure of a VE is decomposed hierarchically into different levels of element, namely, EM, department, team, role, and user. Several elements in the same level are then chosen to form a VER if they can be assigned to different workers; they have the same authorizations, and they do not have resource security problems resulting from sharing or collaboration. Finally, those single elements that cannot be assigned to different workers are converted into VERs.

(A332) Transform tasks into FTs. Action A332 transforms the manageable tasks in activity models into FTs, whose properties must be filled in. If the resources of a manageable task simultaneously allow and disallow sharing, then the manageable task must be decomposed into two or more FTs.

(A333) Identify permissions based on the plans for resource using and sharing. Action A333 combines resources and assigns operations on the resources to form PublicP and PrivateP. The PublicPs are identified by a VE leader administrator, and the PrivatePs are identified by every partner's administrator.

(A334) Specify assignments between all elements. All elements involved in the VEAC model have been identified at previous actions. This action specifies all assignments between two elements, such as VE$-P-A$, $P-$VER$-A$, VER$-$FT$-A$, FT$-$PublicP$-A$, and $R-$VER$-A$.

(A335) Specify cooperative modes between VERs. The cooperative modes between two VERs are specified here according to the resource usage and sharing plan.

(A336) Construct PACP. Based on the proposed PACP language model, the action utilizes the business regulations and the resource using and sharing plan to build the PACP of the VE, including a VEACP and several EACPs. The VEACP is built by the administrator of the VE leader, and the EACPs of partners are built by their administrators.

C. Operation and Dissolution Phases

1) Operation phase first sets up the real VEAC model modeled in the third phase. The VEAC system can then manage VE information resources and generate user authorizations, and monitor, control, and report progress against goals, schedule, and milestone of the VE.
2) Dissolution phase assesses successes or failure at the conclusion of the VE, and its results pave the experience for the next new VE. The PACP established at the design phase of a real VE must be modified to comply with the resource sharing rules after the VE dissolution.

VIII. Example of Practical VE Applying the VEAC Model

This section utilizes the automobile industry as an example to verify the feasibility of the proposed fundamental VEAC model and the PACP language model. Fig. 10 shows a new car engine R&D project (p_1) performed by VE ve_1. In Fig. 10, only parts of the projects are shown; some elements and assignments regarding the project and its three subprojects are shown in detail in the following tables. The engine R&D project has three subprojects: cylinder head design (p_{11}), cylinder block design (p_{12}), and crankshaft design (p_{13}). p_{11} is associated with p_{12} by using process and reference relations $(p_{11}R_{\mathrm{pp}}p_{12}$ and $p_{11}R_{\mathrm{pr}}p_{12})$; p_{12} is associated with p_{11} via process and reference relations $(p_{12}R_{\mathrm{pp}}p_{11}$ and $p_{12}R_{\mathrm{pr}}p_{11})$; p_{13} is exclusive to p_{11} and p_{12} via $p_{11}R_{\mathrm{pe}}p_{13}$ and $p_{12}R_{\mathrm{pe}}p_{13}$, and p_1 is exclusive to p_{12} via $p_1R_{\mathrm{pe}}p_{12}$. According to these definitions, p_{11} and p_{12} have stronger requirement for resource sharing, whereas p_{12} and p_{13} are independent. This example focuses on trust evaluation between the four projects and trust evaluation between three VERs (ve_{11}, ve_{12}, and ve_{13}) involved

TABLE I
VE−P−A AND P−VER−A LISTS

VE Name	Performed Project	Involved VERs	Objectives of the Project
ve_1: engine R&D	p_1	ver_{11}, ver_{12}	Designing a car engine (displacement: 2000cc., and horsepower >140 Hp)
ve_{11}: cylinder head design	p_{11}	ver_{111}, ver_{112}, ver_{113}	Designing the cylinder head of the engine
ve_{12}: cylinder block design	p_{12}	ver_{121}, ver_{122}, ver_{123}, ver_{124}	Designing the cylinder block and the timing gear cover of the engine
ve_{13}: crankshaft design	p_{13}	ver_{131}, ver_{132}, ver_{133}, ver_{134}, ver_{135}, ver_{136}	Designing the crankshaft and connecting rod of the engine

TABLE II
COMPANY LIST

Company No.	Company Name	Number of Employees	Company Address	Core Capacities
em_1	Company-A	100	Tainan Taiwan	Block, Internal Combustion Engine
em_2	Company-B	20	Taipei Taiwan	Cooling System
em_3	Company-C	200	Beijing China	Cylinder
em_4	Company-D	1200	Detroit USA	Cylinder
em_5	Company-E	5	TaichungTaiwan	Internal and External Combustion Engines
em_6	Company-F	13	Tokyo Japan	Main Bearing, Vibration Damper
em_7	Company-G	100	Shangha China	Flywheel, Crankshaft, Cam

in p_{11}; hence, some elements or assignments are ignored in the following tables.

Table I, the VE−P−A and P−VER−A lists, shows the VE name, project performed by the VE, the VERs involved in the VE, and the project objectives. For example, ve_1 involves two VERs, ver_{11} and ver_{12}, and performs project p_1 whose aim is to develop a 2000 cc car engine with at least 140 hp.

Table II lists the detailed information for each company participating in the four VEs.

Table III, the VE−EM−A list, shows all EMs in each VE; for instance, the companies participating in ve_{11} are em_1, em_2, and em_3.

Table IV lists the attributes of FTs that are associated with the four projects, including the number, name, allowed reference, allowed subproject, and allowed main-project attributes.

Table V lists the P−FT−A with project names, the number of FTs assigned to the projects, and the FTs involved in the projects.

Table VI lists the executed sequence of FTs involved in the two projects (p_{11} and p_{12}) between which a process relation is held. Consequently, when the event FT ft_{111} is achieved, the action FT ft_{121} is triggered. According to the process relation definition, ft_{121} will hierarchy all or part of the privileges assigned to ft_{111} when ft_{121} is executed.

Table VII, FT−PublicP−A, lists each FT and PublicPs assigned to each FT.

Table VIII shows the VER−FT−A list, in which only VERs involved in ve_{11} are considered and listed.

In the aforementioned example, ve_{11} (cylinder head design) is used as an example to construct PACP for managing resources that belong to ve_{11}, as shown in the Appendix II. With the objective of cylinder head design of a new car engine, this VE consists of three EMs, i.e., Company-A (em_1), Company-B (em_2), and Company-C (em_3), responsible for oil filler cap design, cylinder head design, and stopper design, respectively.

TABLE III
VE−EM−A LIST

VE Name	Enterprise Members
ve_1	$em_1, em_2, em_3, em_4, em_5, em_6, em_7$
ve_{11}	em_1, em_2, em_3
ve_{12}	em_3, em_4, em_5
ve_{13}	em_6, em_7

In this PACP (see Appendix II), only part of the rules in the VEACP and part of the rules in the EACP of Company-A are listed. According to VEACP rule-ve_{11}-001, when two tasks ft_{111} and ft_{112} are being executed from May 20, 2007 to October 20, 2008, all Company-A, Company-B, and Company-C personnel may read knowledge of know-what about cylinder head design, car engine, and cylinder. The EACP rule-em_1-001 for Company-A dictates that, from November 20, 2007 to October 20, 2008, all Company-B and Company-C personnel may read R&D knowledge related to oil filler cap design.

IX. CONCLUSION AND FUTURE WORK

The results and contributions of this paper are as follows.

1) The formal VEAC model, including the fundamental VEAC model, PACP language model, and construction

TABLE IV
ATTRIBUTE LIST OF FTs

FT No.	FT Name	Attributes		
		Allowed-reference	Allowed-sub-project	Allowed-main-project
ft_{11}	Sub-project progress management	T	T	F
ft_{12}	Sub-project progress management	T	T	F
ft_{13}	Sub-project progress management	T	T	F
ft_{14}	Bulletin	T	T	T
ft_{111}	Oil filler cap design	T	T	F
ft_{112}	Cylinder head design	T	T	F
ft_{113}	Stopper design	T	T	F
ft_{121}	Cylinder liner design	T	T	F
ft_{122}	Cylinder head knock pin design	T	T	F
ft_{123}	Clutch housing design	T	T	F
ft_{124}	Engine rear bracket design	T	T	F
ft_{131}	Crankshaft design	F	F	F
ft_{132}	Crankshaft bearing upper metal design	F	F	F
ft_{133}	Lower oil ring design	F	F	F

TABLE V
$P-$FT$-A$ LIST

Project Name	Number of FTs	Functional Tasks
p_1	4	$ft_{11}, ft_{12}, ft_{13}, ft_{14}$
p_{11}	4	$ft_{11}, ft_{111}, ft_{112}, ft_{113}$
p_{12}	5	$ft_{12}, ft_{121}, ft_{122}, ft_{123}, ft_{124}$
p_{13}	4	$ft_{13}, ft_{131}, ft_{132}, ft_{133}$

TABLE VI
SEQUENCE LIST

Event-Functional Task	Action-Functional Task
ft_{111}	ft_{121}
ft_{121}	ft_{122}
ft_{122}	ft_{112}
ft_{112}	ft_{123}

TABLE VII
FT$-$PublicP$-A$ LIST

FT	Public Permissions
ft_{11}	*public*-p_1
ft_{12}	*public*-p_2
ft_{13}	*public*-p_3
ft_{14}	*public*-p_4
ft_{111}	*public*-p_5
ft_{112}	*public*-p_6
ft_{113}	*public*-p_7
ft_{121}	*public*-p_7, *public*-p_8
ft_{122}	*public*-p_6, *public*-p_8
ft_{123}	*public*-p_9
ft_{124}	*public*-p_{10}
ft_{131}	*public*-p_{11}, *public*-p_{13}
ft_{132}	*public*-p_{14}
ft_{133}	*public*-p_{15}

TABLE VIII
VER$-$FT$-A$ LIST

VER	Performed Functional Tasks
ver_{111}	$ft_{11}, ft_{111}, ft_{112}$
ver_{112}	$ft_{11}, ft_{112}, ft_{113}$
ver_{113}	ft_{12}, ft_{112}

methodology, is proposed to facilitate VE resource management and sharing across organizations.

2) The fundamental VEAC model is designed to adapt to changes in VE members, both individuals and organizations, without affecting authorities of VERs, and eliminates the need to reset users' access authorities due to changes in cooperation targets.
3) Participation or withdrawal of an enterprise does not change the existing management model of resource access, thus significantly reducing administrative cost and complexity.

The results of this paper may help VEs solve the challenges of resource management and sharing among enterprises. Resource management and sharing will become increasingly complicated in the future owing to the requirement of strong information transparency. The proposed formal VEAC model solves AC and VE resource sharing challenges.

However, this paper has some deficiencies. For instance, the non-RBAC model, and integration of its access policies, has not been explored. An enterprise that adopts non-RBAC models and other access policies must perform additional model-transferring process to transform the models to RBAC to integrate them into the proposed PBAC model. This paper does not consider the possibility that the user might share a resource with unauthorized users, for example, by copying it, after legally acquiring the resource. The works in future are listed as follows.

1) An enterprise might adopt a non-RBAC-based scheme. Therefore, integrating different AC schemes or policies should be a focus for future works.
2) An enterprise should ideally retain its original AC model when joining a VE. Hence, a "plug-and-play" AC integration mechanism should be developed.
3) Because an enterprise might participate in several competing VEs, preventing the leaking of key technology or data should be considered.
4) Distributed security infrastructure including distributed heterogonous security architecture and collaborative VE policy management approaches should be completely designed for implementing the VEAC system.

APPENDIX I

TABLE IX
LIST OF FUNCTIONS RELATED TO THE VEAC MODEL

Function	Domain	Co-domain	Description
$VE\text{-}EM\text{-}A_{em}(ve)$	VE	2^{EM}	a *ve* to a set of *EM*s that participate in this *ve*
$VE\text{-}EM\text{-}A_{ve}(em)$	EM	2^{VE}	an *em* to a set of *VE*s that involve this *em*
$VE\text{-}P\text{-}A_{vp}(p)$	P	VE	a project *p* to a *VE* that performs this *p*
$VE\text{-}P\text{-}A_{p}(ve)$	VE	2^{P}	a VE *ve* to a set of *Project*s that are performed by this *ve*
$P\text{-}VER\text{-}A_{p}(ver)$	VER	P	a *ver* to a project *p* that involves this *ver*
$P\text{-}VER\text{-}A_{ver}(p)$	P	2^{VER}	a project *p* to a set of *VER*s that are assigned to this *p*
$P\text{-}FT\text{-}A_{ft}(p)$	P	2^{FT}	a project *p* to a set of *FT*s that are involved in this *p*
$P\text{-}FT\text{-}A_{p}(ft)$	FT	P	a *ft* to a project that involves this *ft*
$VER\text{-}FT\text{-}A_{ver}(ft)$	FT	2^{VER}	a *ft* to a set of *VER*s that perform this *ft*
$VER\text{-}FT\text{-}A_{ft}(ver)$	VER	2^{FT}	a *ver* to a set of *FT*s that are performed by this *ver*
$EM\text{-}U\text{-}A_{u}(em)$	EM	2^{U}	an *em* to a set of *U*s that are employees of this *em*
$EM\text{-}U\text{-}A_{em}(u)$	U	EM	a user *u* to an *EM* that involves this *u*
$NEM\text{-}U\text{-}A_{u}(nem)$	NEM	2^{U}	a *nem* to a set of *U*s that are employees of this *nem*
$NEM\text{-}U\text{-}A_{nem}(u)$	U	NEM	a user *u* to a *NEM* that involves this *u*
$EM\text{-}NEM\text{-}A_{em}(nem)$	NEM	2^{EM}	a *nem* to a set of *EM*s with tasks are supported by this *nem*
$EM\text{-}NEM\text{-}A_{nem}(em)$	EM	2^{NEM}	an *em* to a set of *NEM*s that support some tasks of this *em*
$R\text{-}VER\text{-}A_{r}(ver)$	VER	2^{R}	a *ver* to a set of *R*s that play this *ver*
$R\text{-}VER\text{-}A_{ver}(r)$	R	2^{VER}	a role *r* to a set of *VER*s that this *r* plays
$FT\text{-}PublicP\text{-}A_{public\_p}(ft)$	FT	$2^{PublicP}$	a *ft* to a set of *PublicP*s over all stages
$FT\text{-}PublicP\text{-}A_{ft}(public\_p)$	$PublicP$	2^{FT}	a *public_p* to a set of *FT*s over all stages
$FT\text{-}PrivateP\text{-}A_{private\_p}(ft)$	FT	$2^{PrivateP}$	a *ft* to a set of *PrivateP*s over all stages
$FT\text{-}PrivateP\text{-}A_{ft}(private\_p)$	$PrivateP$	2^{FT}	a *private_p* to a set of *FT*s over all stages
$FT\text{-}Permission\text{-}A(ft)$	FT	$2^{PublicP \quad PrivateP}$	a *ft* to a set of *Permission*s (including private and public permissions) over all stages
$FT\text{-}PublicP\text{-}A_{ft}(st)$	$Stage$	2^{FT}	a stage *st* to a set of *FT*s
$Stage_{public\_p}(st)$	$Stage$	$2^{PublicP}$	a stage *st* to a set of *Public-P*s, $Public\_Permission(st) \subseteq \{public\_p: (FT\text{-}PublicP\text{-}A_{ft}(st), public\_p) \in FT\text{-}PublicP\text{-}A\}$, which can change with *st*
$Stage_{ft}(st)$	$Stage$	2^{FT}	a stage *st* to a set of *FT*s, $Functional\_Task(st) \subseteq \{ft: (FT\text{-}PublicP\text{-}A_{ft}(st), public\_p) \in FT\text{-}PublicP\text{-}A\}$, which can alter with *st*
$Correspondence_{post}(ft)$	FT	FT	a pre-version FT *ft* to its post-version FT
$Correspondence_{pre}(ft)$	FT	FT	a post-version FT *ft* to its pre-version FT
$FTWf_{event}(ft)$	FT	2^{FT}	an action FT *ft* to a set of its event FTs
$FTWf_{action}(ft)$	FT	2^{FT}	an event FT *ft* to a set of its action FTs
$RH_{senior}(r)$	R	2^{R}	a role *r* to a set of *R*s, which are the senior roles of the *r*
$RH_{junior}(r)$	R	2^{R}	a role *r* to a set of *R*s, which are the junior roles of the *r*
$PR_{subset}(p)$	P	2^{P}	a project *p* to a set of *P*s with which the *p* has a subset relation
$PR_{version}(p)$	P	2^{P}	a project *p* to a set of *P*s with which the *p* has a version relation
$PR_{reference}(p)$	P	2^{P}	a project *p* to a set of *P*s with which the *p* has a reference relation
$PR_{process}(p)$	P	2^{P}	a project *p* to a set of *P*s with which the *p* has a process relation
$PR_{exclusive}(p)$	P	2^{P}	a project *p* to a set of *P*s with which the *p* has a exclusive relation
$CM_{cds}(ver)$	VER	2^{VER}	a *ver* to a set of *VER*s that cooperate with the *ver* by using dependent single-task mode
$CM_{cdm}(ver)$	VER	2^{VER}	a *ver* to a set of *VER*s that cooperate with the *ver* by using multi-task mode

APPENDIX II

TABLE X

EXAMPLE OF PACP FOR THE DEVELOPMENT OF CYLINDER HEAD OF A CAR ENGINE

```
<PACP Version= "version 1.1.1" Policy-no.= "N00233" Update-time= "5/15/2007" State= "active">
    <VEACPVE-name= "ve11" objective= "cylinder head design" Update-time= "5/15/2007" State= "active">
        <RuleSet>
            <RuleCombiningAlgorithm>permit-overrides</RuleCombiningAlgorithm>
            <Rule Type= "rule-kind" Name= "rule-ve11-001" Action= "active">
                <Target>
                    <SubjectSet>
                        <Subject>Company-A</Subject>
                        <Subject>Company-B</Subject>
                        <Subject>Company-C</Subject>
                    </SubjectSet>
                    <ActionSet>
                        <Action>read</Action>
                    </ActionSet>
                    <ObjectSet>
                        <Object>know-what to cylinder head design</Object>
                        <Object>know-what to car engine</Object>
                        <Object>know-what to cylinder</Object>
                    </ObjectSet>
                    <Environment> date>=5/20/2007 and date<=10/20/2008 </Environment>
                </Target>
                <Condition>ft111(oil filler cap design) and ft112 (cylinder head design) are being
                    executed</Condition>
                <Effect>permit</Effect>
            </Rule>
            <Rule Type= "rule-kind" Name= "rule-ve11-002" Action= "active">
                <Target>
                    <SubjectSet>
                        <Subject>Company-A</Subject>
                        <Subject>Company-B</Subject>
                        <Subject>Company-C</Subject>
                    </SubjectSet>
                    <ActionSet>
                        <Action>write</Action>
                        <Action>read</Action>
                    </ActionSet>
                    <ObjectSet>
                        <Object>all resources assigned to ft11</Object>
                    </ObjectSet>
                    <Environment> date>=5/20/2007 and date<=10/20/2008</Environment>
                </Target>
                <Condition> anyone of ft111(oil filler cap design), ft112 (cylinder head design) and ft113 (stopper
                    design) are being executed</Condition>
                <Effect>permit</Effect>
            </Rule>
        </RuleSet>
    </VEACP>

    <EACP Company-ID= "em1" Company-name= "Company-A" Responsibility= "oil filler cap design"
        Update-time= "5/16/2007" State= "active">
        <RuleSet>
            <RuleCombiningAlgorithm>permit-overrides</RuleCombiningAlgorithm>
            <Rule Type= "rule-kind" Name= "rule-em1-001" Action= "active">
                <Target>
                    <SubjectSet>
                        <Subject>Company-B</Subject>
                        <Subject>Company-C</Subject>
                    </SubjectSet>
                    <ActionSet>
                        <Action>read</Action>
                    </ActionSet>
                    <ObjectSet>
                        <Object>R&D knowledge related to oil filler cap design</Object>
                    </ObjectSet>
                    <Environment> date>=11/20/2007 and date<=10/20/2008</Environment>
                </Target>
                <Condition>f111 is completed</Condition>
                <Effect>permit</Effect>
            </Rule>
        </RuleSet>
    </EACP>
    ...

</PACP>
```

REFERENCES

[1] Y.-M. Chen and M.-W. Liang, "Design and implementation of a collaborative engineering information system for allied concurrent engineering," *Int. J. Comput. Integr. Manuf.*, vol. 13, no. 1, pp. 11–30, Jan. 2000.

[2] A. Frenkel, H. Afsarmanesh, C. Garita, and L. O. Hertzberger, "Supporting information access rights and visibility levels in virtual enterprises," in *Proc. 2nd IFIP Work. Conf. Infrastructure Virtual Enterprise*, 2000, pp. 177–192.

[3] J. Ma and M. A. Orgun, "Trust management and trust theory revision," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 36, no. 3, pp. 451–460, May 2006.

[4] Y. Lu, W. Wang, B. Bhargava, and D. Xu, "Trust-based privacy preservation for peer-to-peer data sharing," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 36, no. 3, pp. 498–502, May 2006.

[5] E. Turban, D. King, D. Viehland, and J. Lee, *Electronic Commerce: A Managerial Perspective*. Upper Saddle River, NJ: Pearson Educ. Int., 2006.

[6] H. R. Rao and S. J. Upadhyaya, "Special issue on secure knowledge management," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 35, no. 1, p. 185, Jan. 2005.

[7] E. Bertino, L. R. Khan, R. Sandhu, and B. Thuraisingham, "Secure knowledge management: Confidentiality, trust, and privacy," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 36, no. 3, pp. 429–438, May 2006.

[8] R. Singh and A. F. Salam, "Semantic information assurance for secure distributed knowledge management: A business process perspective," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 36, no. 3, pp. 472–486, May 2006.

[9] R. Au, M. Looi, and P. Ashley, "Automated cross-organizational trust establishment on extranets," in *Proc. Workshop Inf. Technol. Virtual Enterprises*, 2001, pp. 3–11.

[10] T.-Y. Chen, Y.-M. Chen, C.-B. Wang, and H.-C. Chu, "Development of an access control model, system architecture and approaches for information sharing in virtual enterprise," *Comput. Ind.*, vol. 58, no. 1, pp. 57–73, Jan. 2007.

[11] T.-Y. Chen, Y.-M. Chen, H.-C. Chu, C.-B. Wang, and H. Yang, "Secure resource sharing on cross-organization collaboration using a novel trust method," *Robot. Comput.-Integr. Manuf.*, vol. 23, no. 4, pp. 421–435, Aug. 2007.

[12] M. Koch, L. V. Mancini, and F. Parisi-Presicce, *Graph Transformations for the Specification of Access Control Policies*. Amsterdam, The Netherlands: Elsevier science B. V, 2002.

[13] D. F. Ferraiolo, D. R. Kuhn, and R. Chandramouli, *Role-Based Access Control*. Norwood, MA: Artech House, 2003.

[14] S. Oh and S. Park, "Task-role-based access control model," *Inf. Syst.*, vol. 28, no. 6, pp. 533–562, Sep. 2003.

[15] A. Kern, "Advanced features for enterprise-wide role-based access control," in *Proc. Comput. Security Appl. Conf.*, 2002, pp. 333–342.

[16] C. J. Moon, D. H. Park, S. J. Park, and D. K. Baik, "Symmetric RBAC model that takes the separation of duty and role hierarchies into consideration," *Comput. Security*, vol. 23, no. 2, pp. 126–136, Mar. 2004.

[17] D. Shin, G. J. Ahn, and J. S. Park, "An application of directory service markup language (DSML) for role-based access control (RBAC)," in *Proc. Comput. Softw. Appl. Conf.*, 2002, pp. 934–939.

[18] K. Furst, T. Schmidt, and G. Wippel, "Managing access in extended enterprise networks," *IEEE Internet Comput.*, vol. 6, no. 5, pp. 67–74, Sep./Oct. 2002.

[19] J. Bacon, K. Moody, and W. Yao, "A model of OASIS role-based access control and its support for active security," *ACM Trans. Inf. Syst. Security*, vol. 5, no. 4, pp. 492–540, Nov. 2002.

[20] F. T. Alotaiby and J. X. Chen, "A model for team-based access control (TMAC)," in *Proc. Inf. Technol.: Coding Comput.*, 2004, vol. 1, pp. 450–454.

[21] J. J. Kanet, W. Faisst, and P. Mertens, "Application of information technology to a virtual enterprise broker: The case of Bill Epstein," *Int. J. Prod. Econ.*, vol. 62, no. 1, pp. 23–32, May 1999.

[22] E. K. Ouzounis, "An agent-based platform for the management of dynamic virtual enterprises," Ph.D. dissertation, Tech. Univ. Berlin, Berlin, Germany, 2001.

[23] J. S. Park and J. Hwang, "RBAC for collaborative environments: Role-based access control for collaborative enterprise in peer-to-peer computing environments," in *Proc. 8th ACM Symp. Access Control Models Technol.*, 2003, pp. 93–99.

[24] N. Mezzetti, "Towards a model for trust relationships in virtual enterprises," in *Proc. 14th Int. Workshop Database Expert Syst. Appl.*, 2003, pp. 420–424.

[25] T. J. Smith and L. Ramakrishnan, "Joint policy management and auditing in virtual organizations," in *Proc. 4th Int. Workshop Grid Comput.*, 2003, pp. 117–124.

[26] G. Steinke and R. Leamon, "Information security issues facing virtual enterprises," in *Proc. Int. Conf. Eng. Technol. Manage.*, 1996, pp. 641–644.

[27] H. Zhu, "Some issues of role-based collaboration," in *Proc. Can. Conf. Elect. Comput. Eng.*, 2003, vol. 2, pp. 687–690.

[28] G. Kolaczek, "Specification and verification of constraints in role based access control," in *Proc. 12th IEEE Int. Workshops Enabling Technol.: Infrastructure Collaborative Enterprise*, 2003, pp. 190–195.

[29] J. Luo and D. He, "Research on object-oriented role-based access control model," in *Proc. 4th Int. Conf. Parallel Distrib. Comput., Appl. Technol.*, 2003, pp. 132–135.

[30] J. D. Moffett, "Control principles and role hierarchies," in *Proc. 3rd ACM Workshop Role-Based Access Control*, 1998, pp. 63–69.

[31] S. Osborn, "Integrating role graphs: A tool for security integration," *Data Knowl. Eng.*, vol. 43, no. 3, pp. 317–333, Dec. 2002.

[32] M. Lorch, S. Proctor, R. Lepro, D. Kafura, and S. Shah, "First experiences using XACML for access control in distributed systems," in *Proc. ACM Workshop XML Security*, 2003, pp. 25–37.

[33] S. Barker and P. J. Stuckey, "Flexible access control policy specification with constraint logic programming," *ACM Trans. Inf. Syst. Security*, vol. 6, no. 4, pp. 501–546, Nov. 2003.

[34] M. Coetzee and J. H. P. Eloff, "Virtual enterprise access control requirements," in *Proc. Annu. Res. Conf. South Afr. Inst. Comput. Scientists Inf. Technologists Enablement Through Technol.*, 2003, pp. 285–294.

[35] S. Jajodia, P. Samarati, M. L. Sapino, and V. S. Subrahmanian, "Flexible support for multiple access control policies," *ACM Trans. Database Syst.*, vol. 26, no. 2, pp. 214–260, Jun. 2001.

[36] A. Belokosztolszki and K. Moody, "Meta-policies for distributed role-based access control systems," in *Proc. 3rd Int. Workshop Policies Distrib. Syst. Netw.*, 2002, pp. 106–115.

[37] S. Hada and M. Kudo, "XML document security based on provisional authorization," in *Proc. 7th ACM Conf. Comput. Commun. Security*, 2000, pp. 87–96.

[38] G. Boella and L. van der Torre, "Security policies for sharing knowledge in virtual communities," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 36, no. 3, pp. 439–450, May 2006.

[39] A. Kern, A. Schaad, and J. Moffett, "Enterprise role administration: An administration concept for the enterprise role-based access control model," in *Proc. 8th ACM Symp. Access Control Models Technol.*, 2003, pp. 3–11.

[40] R. A. Botha and J. H. P. Eloff, "Designing role hierarchies for access control in workflow systems," in *Proc. 25th Annu. Int. Comput. Softw. Appl. Conf.*, 2001, pp. 117–122.

[41] F. Dridi, B. Muschall, and G. Pernul, "Administration of an RBAC system," in *Proc. 37th Annu. Hawaii Int. Conf. Syst. Sci.*, 2004, pp. 187–192.

Tsung-Yi Chen received the B.S. degree from Providence University, Taichung, Taiwan, R.O.C., in 1996, and the M.S. and Ph.D. degrees from the Institute of Manufacturing Engineering, National Cheng Kung University, Tainan, Taiwan, in 2001 and 2006, respectively.

He is currently an Assistant Professor with the Department of Electronic Commerce Management, Nanhua University, Chia-Yi, Taiwan. His research interests include virtual enterprise, e-commerce and knowledge commerce, enterprise and information integration, access control, and knowledge sharing.

Yuh-Min Chen received the B.S. and M.S. degrees from National Tsing Hua University, Hsinchu, Taiwan, R.O.C., in 1981 and 1983, respectively, and the Ph.D. degree in industrial and systems engineering from Ohio State University, Columbus, in 1991.

He is currently a Professor and the Director of the Institute of Manufacturing Engineering, National Cheng Kung University, Tainan, Taiwan. Before joining the faculty of the Institute of Manufacturing Engineering in 1994, he was a Research Engineer with the Structural Dynamics Research Corporation, Plano, TX, for three years. His current research interests include enterprise integration, engineering data and knowledge management, computer-aided concurrent engineering, and manufacturing information systems.

Chin-Bin Wang received the B.S. degree from National Tsing Hua University, Hsinchu, Taiwan, R.O.C., in 1981, the M.S. degree from the University of Southern California, Los Angeles, in 1985, and the Ph.D. degree in computer science from the City University of New York, New York, in 1995.

He is currently a Professor and the Chairman of the Department of Electronic Commerce Management, Nanhua University, Chia-Yi, Taiwan. His research interests include data mining, network management, engineering data and knowledge management, and system integration.

附件 8

以角色為基支援跨網域之網頁應用程式授權方法設計

摘要

隨著資訊應用複雜性，及開發成本考量，虛擬團隊合作開發，已成為趨勢。現今許多網頁應用程式(Web Application)的設計，都是由虛擬團隊分工設計而成；而這些開發的網頁應用程式，都會建構會員管理機制，用以區分管理者(Admin)、一般使用者(User)、閱覽者(Guest)，使之控管部分的服務，不被非法的存取。在權限控管的部分，較多採用以角色為基礎的存取控制(Role-Based Access Control)，作為存取控制的基礎。

這個機制在實作上受到諸多的限制，使用者對網頁應用程式上的工作階段(Session)，無法直接地分享資訊給其它的網頁應用程式作存取。雖然可用 Cookie 將部分的資訊轉存於客戶端，但分享給其他網頁應用程式存取僅限定於相同網域之下(例如：*.yahoo.com)使用，另外 Cookie 本身也有儲存量的限制。為求簡單達到將資訊分享於多個網頁應用程式上，即將所有的網頁應用程式整合在同一台伺服器上，以方便實作整個系統對於所設計的網頁應用程式施行存取控制。但將所有的服務全部集中在同一台伺服器上，將會造成伺服器嚴重的負擔；且開發者在本地端的建置環境，不一定與遠端伺服器的環境相符。

為解決上述問題，本研究以使用者對網頁應用程式上的工作階段(Session)，結合 XML 達成網頁應用程式權限存取跨網域的實作，使各個網頁應用程式可以在自己的伺服器中運作，不再需要統一彙整到同一伺服器中運行，且不再受限整個系統中網頁應用程式需以同一程式語言撰寫。

關鍵字：RBAC Access Control、Web Application、XML、跨網域

目錄

第一章 緒論

如何讓 Web Application 與使用者之間的工作執行階段(Session)分享給其它的 Web Application，一直是許多開發人員心中的問題。在網際網路上許多人提出不同的見解，包含使用 ASP.net 的單一簽入方法、利用 SQL 來製作大型的 Catch 機制、架設 Memcatched 伺服器來存放使用者與 Web Application 之間的工作執行階段(Session)以分享給其它的 Web Application、利用 PHP 中 Session 的特性...等諸多方法。

ASP.net 是目前最容易實作單一簽入的開發語言，但是很顯然的必須使用 ASP.net 來製作。利用 SQL 來建置一個暫存的資料表放置資訊，這是不錯的方法，但是若是要授權能取的資訊的 key 值，就必須存放於客戶端上的 Cookie，使用 Cookie 就意味著必須在相同網域下才能運行。為了改善 SQL 殺雞用牛刀來存放臨時資訊，以及無法真正達到跨越網域，而有了 Memcatched 伺服器的發展，它的好處就是使用伺服器的記憶體來當作一個大型的 Session Catch，只要有該伺服器的 IP 位址與端口位置(Port)就能輕易撈取資訊；但這就意味著 Memcatched 的危險性，必須使用防火牆對 Memcatched 進行保護，並且要預防有心人士直接從端口中撈取資料。關於 PHP 的 Session 特性，即是利用 PHP 的運行環境作為一個更大的工作執行階段，讓每一個 Web Application 在這個工作階段之下能有權限取得共享的資訊，這表示著每一個 Web Application 必須是在同一伺服器底下，並且是屬於同一個 PHP 的執行工作階段裡。

本研究的目的，使程式設計人員可以不用改變開發習慣，也無須增資其它因應成本，即能達到位於不同網域與伺服器中的 Web Application 能共享資訊，並透過 RBAC 的授權機制，能更進一步的控制用戶對於每一個 Web Application 中各項元件的詳細設定。

第二章 文獻探討

第一節 Role-Base Access Control[4,10]

以角色為基礎存取控制機制(RBAC)在 1992 年被正式提出。以角色為基礎之存取控制主要的觀念是在使用者及資訊資源權限之間加入角色(職務)的概念，將所需的權限指派該角色，再將使用者指派至所屬的角色上，以角色來決定使用者是否執行某個物件的權限，是一種非隨意性的存取控制(non-discretionary access control)。Sandhu 等人提出 RBAC 模型，主要包括了四個元件，分別為使用者(User, U)、角色(Roles, R)、權限(Permissions, P)、工作階段(Session ,S)，各元件關係圖如 2.1 所示，主要的特色包含了權責區分(Separation of Duty)、職務階層(Role Hierarchy)、限制(Constraint)、最小權限(Least Privilege)。

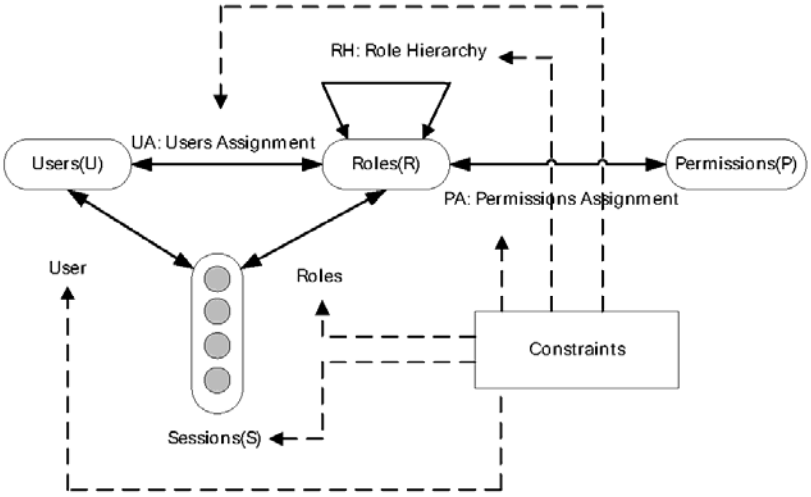

圖 2.1 以角色為基礎之存取控制

第二節 Web Application

Web Application[1]是一種架設在 Web 伺服器(Server)上，接受使用者資訊、處理並回傳網頁資訊的應用程式，基本的架構是 Client/Server，Server 只負責處理資料，並且和資料儲存區間溝通與交換資料，在處理完成後產生網頁指令，並輸出到用戶端。用戶端(Client)是一種實作網路通訊的本機應用程式(Local Application)通常是瀏覽器(Internet Explorer/ Firefox)，或是實作網路通訊用戶端的伺服器程式，負責處理/解譯由伺服器傳輸而來的網頁標記資料(tagged data)，然後繪製(rendering)成一個完整的網頁畫面，並且處理存在標記資料中的指令程式碼(scripting language)，讓用戶端具有快速處理與回應的能力。Web Application 使用的標記資料是一種以標記(tag)定義各項資料的格式，藉以讓用戶端處理的資料流(stream)，網頁使用的標記資料格式稱為 HTML，或者是使用更嚴謹的 XML 格式，裡面包含了文字資料，超連結或是加密的二進位資料(編碼為 Base64 格式的字串)。常見的 Web Application 包含 Webmail(電子郵件收發)、Online Retail Sales(線上零售)、Discussion Boards(討論區)、Weblogs(部落格)等。

第三節 Session

網頁用程式執行工作階段狀態[7]，讓使用者在使用者巡覽不同的網頁時，儲存和擷取使用者的數值。HTTP 是沒有狀態 (Stateless) 的通訊協定，表示 Web 伺服器會將頁面的每個 HTTP 要求視為獨立要求；伺服器不會保留先前要求所使用的變數值。網頁應用程式工作階段狀態會在限制時間間隔內，將來自相同瀏覽器的要求識別為一個工作階段，並且提供保存這個工作階段期間內之變數值的功能。工作階段是由唯一的工作階段識別項所識別，當啟用網頁應用程式的工作階段狀態時，應用程式中對頁面的每個要求，都會檢查瀏覽器送出的 Session ID 值。

如果並未提供任何 Session ID 值，網頁應用程式會啟動新的工作階段，然後搭配回應將該工作階段的 Session ID 傳送給瀏覽器。

根據預設，Session ID 值會存放在 Cookie 中，但是也可以設定應用程式將 Session ID 值存放在工作階段的 URL 中。只要使用相同的 Session ID 值持續產生要求，工作階段就會視為使用中。如果特定工作階段的要求間隔超過指定的逾時值(通常為 30 分鐘)，則工作階段會視為已過期。以過期的 Session ID 值產生要求，會導致啟動新的工作階段。

第四節 Cookie

Cookie[3,5,8]是網際網路上最常見的一種儲存少量資訊的機制，網站伺服器透過 Cookie 將少量的資訊儲存至使用者的瀏覽器內，主要是因為網頁傳輸協定(Hypertext Transfer Protocol)是一種無狀態的通訊協定(Stateless Protocol)，因此大多數的網站都會利用 Cookie 來維持與使用者之間的部分資訊，或儲存網頁應用程式所需要的資訊。基本上 Cookie 可分為持續性 Cookie(Persistent Cookie)與暫時性 Cookie(Transient Cookie or Session Cookie)兩種。

1. 持續性 Cookie

   持續性的 Cookie 限制，會因不同的瀏覽器而略有不同，大致上包含下列項目：

   (1) Cookie 總數不可超過 300 個

   (2) 單一網域不可超過 20 個

   (3) 每一個 Cookie 內容不可大於 4096 個位元組

2. 暫時性 Cookie

   暫時性 Cookie(Transient Cookie)又稱為 Session Cookie，係指在瀏覽器工作階段，將暫時需要的資訊存放於客戶端的記憶體中，當使用者關閉瀏覽器後，該 Cookie 隨之消失。

第五節 XML

XML[2]是由 W3C(Word Wide Web Consortium)[11]所制定，延伸自 SGML 的標記語言，由於具有開放的架構，加上不被技術平台所限制的特性，在 XML 推出後隨即廣受業界所接受。XML 的應用範圍相當廣泛，只要透過標準化的標籤制訂方法，系統開發者即可利用它來開發企業間通訊的資料格式，甚至是通訊協定。XML 為 Web Services 其它技術的基礎，也由於 XML 本身跨平台的特性，方使 Web Services 得以突破技術平台的限制。

第六節 Memcached

Memcached[6]是一套分散式的快取系統，最初是 Danga Interactive 為 LiveJournal 所發展的。Memcached 缺乏認證及安全機制，在使用上僅需得知位址(IP)及端口(Port)即可取出資料，由於企業在建置 Web Services 時已加入具高安全層級之防火牆，Memcached 於防火牆之內受到保護，因此該技術在企業界仍被廣泛使用。

第三章 架構流程

第一節 系統架構

本研究機制包括兩個主要架構：(1)RBAC 授權機制，(2)跨網域處理機制。本章節以「輕度障礙生數位學習平台」建置為例，在這個平台上其中有兩個團隊分別負責製作「概念學習」及「問題導向學習」兩個區塊。製作「概念學習」的團隊慣用 ASP.net 作為開發語言，製作「問題導向學習」的團隊慣用 JSP 作為開發語言。在以往的慣例都會協調以什麼樣的語言開發為主軸，非必要時不使用其它的程式語言做為輔佐；假設位於遠端的伺服器是 Linux 的平台，ASP.net 目前無法完全的相容於 Apache 中，因此「概念學習」的團隊就必須放棄慣用的 ASP.net 改由 JSP 或是其他 Apache 可以運行的開發語言；假設「問題導向學習」的團隊改用 JSP 以符合在遠端伺服器中 Apache 的要求，但遠端伺服器不一定能完全的配合「問題導向學習」團隊在建置上的所有需求，當中必須來來回回不斷的協調與確認，才能在最後上線時所碰到的問題降至最低。透過本系統的架構，無論是「概念學習」或「問題導向學習」的設計團隊，不再有這種困擾，他們可以將服務建置在自己的伺服器中，不需依賴遠端伺服器上的需求及標準，如此般「概念學習」的開發團隊，即可使用慣用的開發語言建置服務。

本系統架構是將設計團隊的每一個網頁，都視為每一個物件(Object)，將這些物件的位址全部導入 RBAC 授權中心，並運用 RBAC 的運作原理配置角色，最後再將角色指向給用戶。用戶在登入後取得對應角色即可取得一張可用的物件表，當用戶在操作 Web Application 時，便是驗證這張物件表來判斷使用者是否有操作 Web Application 中部分網頁的權限。當用戶移動到不同的 Web Application 時，系統會優先判斷用戶是否曾經登入過，曾經登入過的用戶都會取得一個由 RBAC 授權中心所頒予的 Session Client；當系統得知用戶有 Session Client 時，將會透過這個進入 RBAC 授權中心要求取得完整的授權資料，RBAC 授權中心在確定用戶身分後，將會以 XML 的格式將用戶的相關資訊傳遞給 Web Application，由 Web Application 重新建立使用者工作執行階段，讓用戶的權限如同 Web Application 轉移之前一樣。本研究機制基礎架構圖如 3.1 所示。

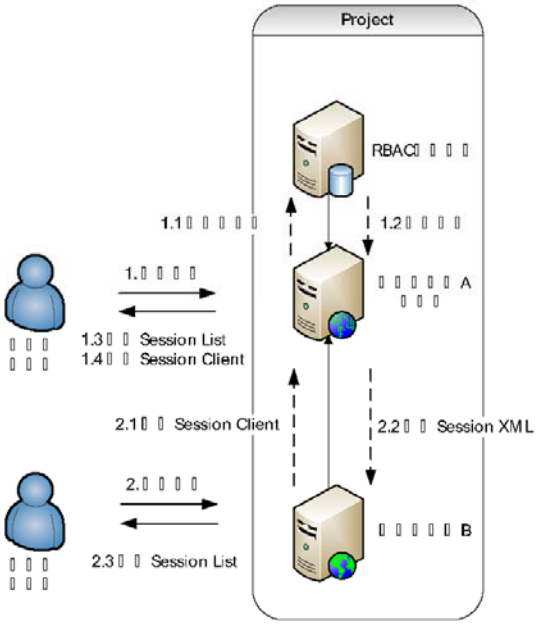

圖 3.1 系統基礎架構

第二節 RBAC 授權中心/機制

一個合法的使用者，在登入之後，必須經過 RBAC 授權中心取的自己的授權關係表(Session List)，這個關係表將會儲存於使用者與 Web Application 的工作執行階段(Session)中。RBAC 授權中心是利用 SQL 資料表與預存程序來完成 RBAC 的授權機制。

在使用者首次進入系統時，會將用戶 ID 傳送至 SQL 中，透過預存程序取得關聯的角色 ID，由於一個使用者可能會有多個角色，因此建立一組角色陣列以儲放這些可用的角色 ID；經由預存程序的控制，可以優先過濾出使用者是否在核定的時間內、核定的地點內擁有存取該角色之能力，若正確無誤則將此角色 ID 記錄下來，反之則拒絕存取該角色 ID，直到過濾完全部角色陣列成為一個可用的角色陣列清單，該部分詳細流程請參閱圖 3.2。經由預存程序過濾出的可用角色 ID 後，便進入另一個預存程序，將可用的角色 ID 取得關聯的物件 ID，由於一個角色可能會包含多個可用物件，因此必須建立一組物件陣列以存放這些可用的物件 ID；經由預存程序的控制，將不在核定時間內、核定地點內的物件予以剃除。最後將產生一張完整的用戶授權關係表(Session List)，並將之存放於使用者與 Web Application 的工作執行階段(Session)中。該部分的詳細流程圖請參閱圖 3.3。最後產生的關係授權表(Session List)存放於使用者與 Web Application 工作執行階段(Session)中的資訊請參閱表 3.1。

方法。大部分開發者優先想到的便是利用 Cookies 來儲存 Session List。但多數的瀏覽器(如：IE 或 Firefox)對於 Cookies 有幾個條件上的限制：

(1)只允許每一個網站(Web Application)儲存 20 個 Cookies。

(2)部分瀏覽器對 Cookies 的數量加上絕對限制，總數不超過 300 個。

(3)Cookie 內容不可超過 4096 個位元組。

經由 RBAC 授權中心所產生的 Session List，無法預估一個用戶會有多少個角色以及相依可用元件。單一個網站只能使用 20 個 Cookies 顯然是不敷使用，雖然部分瀏覽器提供用戶可自行修改部分參數，但這就必須改變使用者經驗來達到系統需求。另外 Cookie 的安全性原則，僅能存取自身網站或是允許的相依網域之下(例如：*.yahoo.com)，當使用者跨出域名之後，Cookie 就無權存取。因此使用 Cookie 作為跨網域的傳遞媒介不在此考慮，取而代之的是由伺服器建立一組唯一的雜湊碼(Session Client)，在 RBAC 授權中心建立起 Session List 時一併將 Session Client 遞送給使用者。

當使用者進入服務時，將檢驗使用者與 Web Application 工作執行階段(Session)是否擁有 Session List，並依據 Session List 來控制使用者是否有權限操作 Web Application 上的部分服務；當使用者與 Web Application 工作執行階段(Session)並不存在由 RBAC 授權中心提供的 Session List 時，將會檢查使用者是否存在著 Session Client，若使用者不存在 Session Client，表示使用者未曾登入過系統，或閒置時間過長以至於 Session Client 失效，此時將會引導使用者回到登入畫面。反之使用者不存在 Session List，卻擁有 RBAC 授權中心頒予的 Session Client，表示使用者已經跨離原本的 Web Application 工作執行階段，進入了另一個新的 Web Application 工作執行階段，因此 Web Application 將自行透過 Session Client 向 RBAC 授權中心重新取得使用者的授權關係表，RBAC 授權中心在取得 Web Application 的請求後，將會驗證 Session Client，並透過 XML 格式將完整的使用者關係授權表(Session XML)回傳給發出請求的 Web Application，讓 Web Application 重新建立起與使用者的工作執行階段(Session)。跨網域請求使用者授權關係表流程圖請參閱圖 3.4，由 RBAC 授權中心回傳的 Session XML 請參閱表 3.2。

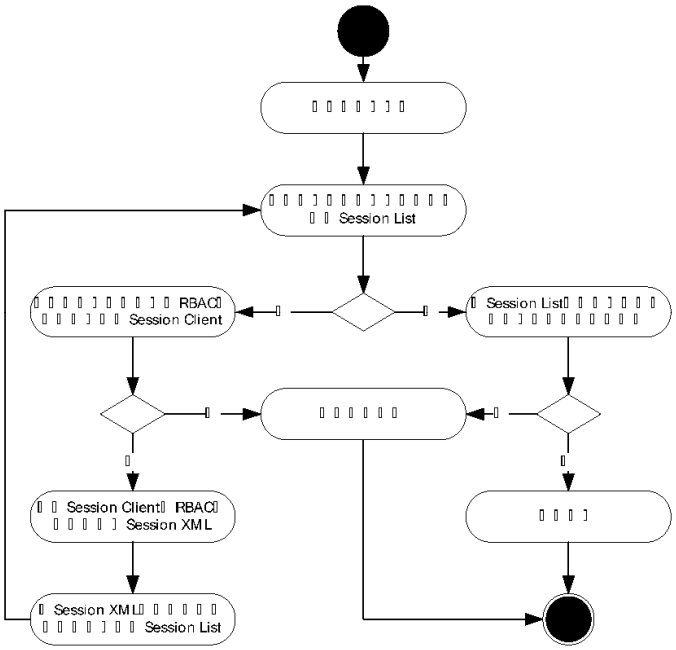

圖 3.4 跨網域請求使用者授權關係表流程圖

表 3.2 Session XML

```
<?xml version="1.0" encoding="UTF-8" ?>
<PageAdmin>
    <User ID="ayu">
        <Name>亞由</Name>
        <Email>sfw.sakana@gmail.com</Email>
    </User>
    <Role Count="2">
        <RID>browser01</RID>
        <RID>sysadmin</RID>
    </Role>
    <Object Count="14">
        <OID>Admin_O2R</OID>
        <OID>Admin_Objects</OID>
        <OID>Admin_R2O</OID>
        <OID>Admin_R2U</OID>
        <OID>Admin_Roles</OID>
        <OID>Admin_U2R</OID>
        <OID>Admin_Users</OID>
        <OID>Logout</OID>
```

```
        <OID>O_List</OID>
        <OID>Radmin_EX01</OID>
        <OID>Session_List</OID>
        <OID>Session_XML</OID>
        <OID>Session_XML_Show</OID>
        <OID>Index</OID>
    </Object>
</PageAdmin>
```

第四節 跨網域機制比較

我們的目的是將 Web Application 中的資訊傳遞給其它的 Web Application 使用。除了本研究所提之跨網域處理機制外，目前亦有其它的跨網域機制處理辦法，其方法有三：

(1) Cookie：是儲存少量資訊於客戶端的機制，但由於 Cookie 本身的限制使得，欲遞送大量資訊顯得困難；若將資訊以批次的方式遞送，將使傳遞效率變差；直接修改客戶端瀏覽器軟體，將必須改變使用者經驗，且增加部分轉製成本。另外使用 Cookie 機制在跨網域部分僅限定於相同域名之下。

(2) Memcached：是分散式的快取系統，在實作的部分類似於 Session 的存取方式。使用 Memched 技術須架設 Memcached 伺服器以及改變部分系統原始碼，在建構上必須付出一定成本，且 Memcached 機制在安全上仍有顧忌及疑慮，由於企業界在建構 Web Services 已建置相關高安全性防火牆，因此該方法在企業界仍被廣泛使用。

(3) Other Agent：是基於兩種不同系統進行異質傳輸，而設計的代理程序。在運用上是將相關資訊傳遞給代理人程式，經由代理人程式與其它代理人溝通將資訊順利輸出。在建置上必須付出代理人程序製作之成本。

本研究機制，資訊的傳遞位於伺服器之間，無須改變使用者的使用經驗；在系統架構的部分，資訊的存取位於 Web Application 中的 Session，不需另外指向於其它存取位置上(Memcached 機制須將存取位置指向於 Memcached 伺服器)，由於資訊在傳遞時是使用 HTTP 通道，因此不需要透過第三方軟體代理人來傳輸，在傳輸的期間可透過 SSL 加密通道進一步的增強安全性，亦可對傳輸的內文進行加密。本研究機制與其它三種機制比較請參閱表 3.3。

表 3.3 相關機制比較表

	Cookie 方法	Memcached 技術	Other Agent	本研究機制
使用者經驗是否改變	是	否	否	否
是否需改變資訊存取方式	否	是	是	否
是否需另外開發第三方軟體	是	否	是	否
資料傳遞安全性	視遞送方式而定	須藉由其它方式加強	視 Agent 設計架構而定	視遞送方式而定
跨網域能力	僅於相同域名之下	可跨網域	可跨網域	可跨網域

第四章 實作設計

在這個章節裡，將透過SQL實作一個簡單的RBAC授權中心，以及基本的RBAC授權控制服務，另外也將實作跨網域存取物件的部分。

第一節 資料庫格式

從 RBAC 的基本模型中，可以知道它是由基本的三個部件所組成的：使用者(User)、角色(Role)、物件(Object)，因此在資料庫裡優先建立起這三張資料表，建立 RBAC 最低需求資料表可參閱表 4.1、4.2、4.3，實際運用可視實際需求調整資料表內的資訊規格，如表 4.1 的使用者資料表，可視需求增加信箱資訊(Email)、地址資訊(Address)、電話資訊(TEL)...等。

表 4.1 使用者資料表

名稱	說明
ID	唯一值索引鍵，作為用戶帳號使用
Password	密碼，作為用戶登入驗證之用
Name	易於辨識的使用者名稱

表 4.2 角色資料表

名稱	說明
ID	唯一值索引鍵，作為角色 ID 使用
Name	易於辨識的角色名稱

表 4.3 物件資料表

名稱	說明
ID	唯一值索引鍵，作為物件 ID 使用
Name	易於辨識的物件名稱

建立好 RBAC 的基本元素後，接著建立兩兩相依的關係表，分別是使用者與角色之間的關係表，角色與物件之間的關係表，這兩張關聯資料表請參閱表 4.4 與 4.5。

表 4.4 使用者與角色關係資料表

名稱	說明
U_ID	用戶 ID。與 R_ID 成為雙索引
R_ID	角色 ID。與 U_ID 成為雙索引

表 4.5 角色與物件關係資料表

名稱	說明
R_ID	角色 ID。與 O_ID 成為雙索引
O_ID	物件 ID。與 R_ID 成為雙索引

使用者帳戶 ID 與「使用者與角色關係資料表」中的 U_ID 互相關聯，角色資料表中的 ID 與「使用者與角色關係資料表」的 R_ID 互相關聯。角色資料表的 ID 與「角色與物件關係資料表」中的 R_ID 互相關聯，物件資料表中的 ID 與「角色與物件關係資料表」的 O_ID 互相關聯；如此般使用者便能透過帳戶 ID 透過「使用者與角色關係資料表」與「角色與物件關係資料表」取得相應的物件 ID，詳細的資料表關連圖請參閱圖 4.1 所示。

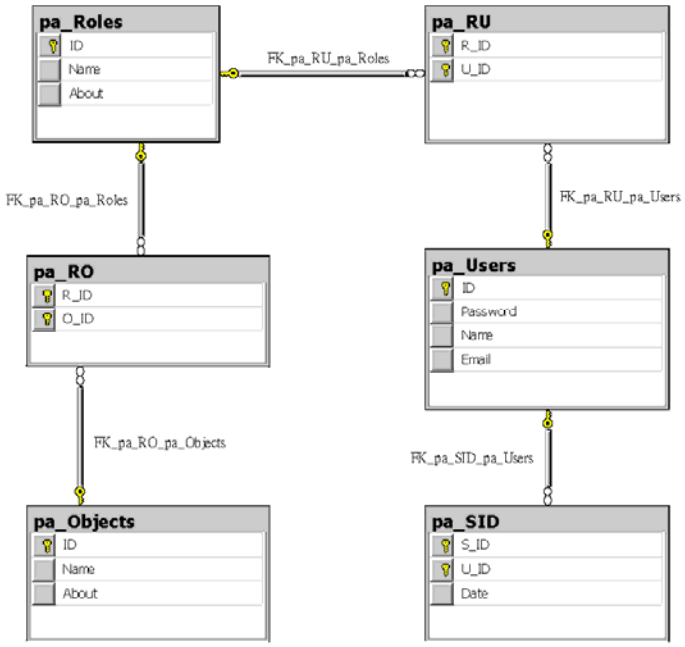

圖 4.1 資料庫關聯圖表

第二節 預存程序

預存程序是 MS-SQL 特有的，其程式語言又稱為 T-SQL，它的好處是將複雜的資料搜尋、處理、過濾，可以優先的從 SQL 中處理好，再展示給前端的開發語言；並非由前端的開發語言逐步得處理資料庫內的檢索、過濾、搜尋等步驟。屆時系統發完成後，若因為有特殊需求改變資料庫內容，僅需修正部分的預存程序即可完成欲更變的內容，而無需一併修改前端的開發程式。在此舉出一個簡單的情境，當非使用預存程序所設計的系統在完成之後，想要增加一個具有紀錄使用者所有操作紀錄的資料表；雖然資料庫製表容易，但是前端所有的應用程序必須重新改寫增加一個具有 Log 能力的函式。使用者有多少的操作能力，就必須補充多少的相關程式碼。若使用預存程序，只需要預存程序中加入對資料表 Log 的能力就能解決此狀況，而無須再修正前端的應用程式。

在本系統添入幾個基本的預存程序，包含：使用者登入(User_Login)、角色列表(User_RoleList)、物件列表(User_ObjectList)、Session Client 增加(SID_Add)、Session Client 驗證(SID_Check)、Session Client 刪除(SID_Del)。其它如，增刪修使用者、增刪修角色、增刪修物件…皆可撰寫於預存程序中，由前端程式直接呼叫使用。

第三節 前端後台介面

前端的後台介面功能包含：新增使用者、刪除使用者、新增角色、刪除角色、新增物件、刪除物件、使用者與角色關連、角色與物件關連。在物件的部分，是將

Web Application 裡的每一個網頁都是為物件，透過角色來區分，哪些網頁(物件)可以被哪些角色所存取。使用者登入介面，請參閱圖 4.2，使用者登入介面除了可以由此頁進行登入外，也可以從其它設計較為美觀的介面傳遞參數進入。增刪使用者介面請參閱圖 4.3，增加使用者除了可以由此後台操作，亦可由較為美觀的介面傳遞參數操作。增刪角色介面請參閱圖 4.4。增刪物件介面請參閱圖 4.5；增刪角色與物件關係的方式可由兩個面向進行，一是由角色為基準加入到物件，該部分請參閱圖 4.6，另一是由物件為基準加入到角色，該部分請參閱圖 4.7。增刪用戶與角色關係的方式也由兩個面向進行，一是由角色為基準配置到指定的用戶上，該部分請參閱圖 4.8，另一是由用戶為基準指向到特定的角色上，該部分請參閱圖 4.9。

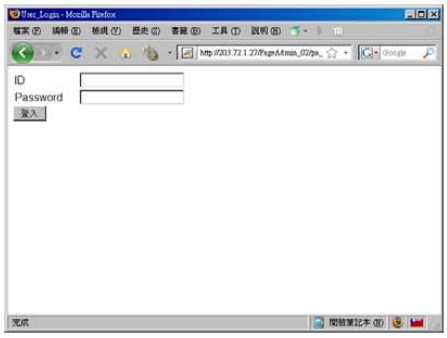

圖 4.2 使用者後台登入介面

圖 4.4 增刪角色後台介面

圖 4.3 增刪使用者後台介面

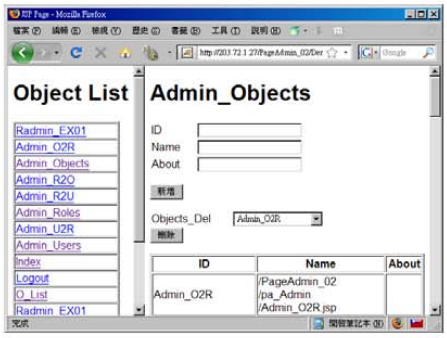

圖 4.5 增刪物件後台介面

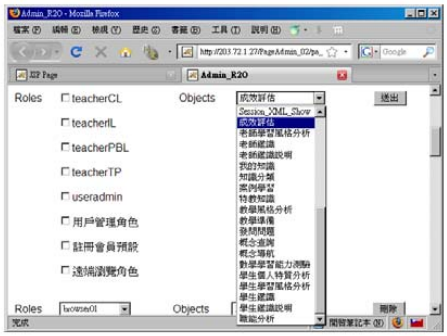

圖 4.6 由角色為基準加入到物件

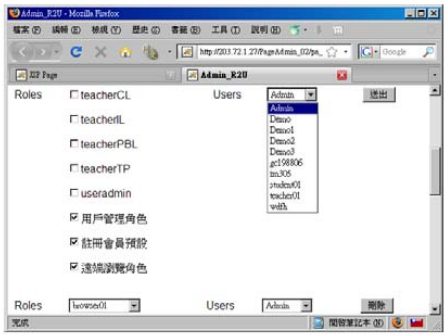

圖 4.8 由角色為基準配置到用戶

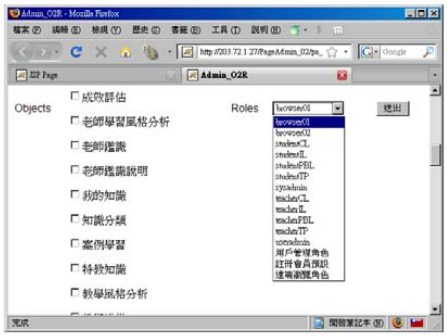

圖 4.7 由物件為基準加入到角色

圖 4.9 由用戶為基準指向到角色

第四節 演算流程

本研究機制演算流程可區分為兩大部分，(1)使用者在成功登入後，由 RBAC 授權中心取得因應的角色和權限，以及(2)使用者進入網頁時判別使用者是否具有權能操作，這個部分涵蓋至跨越網域識別使用者權限之能力。

壹、由 RBAC 授權中心資訊取得演算流程

這個部分的演算流程，係指使用者經過身分確認完成後，由 RBAC 授權中心傳遞因應的角色和權限給使用者。其中包含的副程式有取得角色、取得物件、建立 Session List。

1. 副程式：由 RBAC 授權中心取得可用的角色資訊

由伺服器向 RBAC 授權中心取得使用者可用角色，透過使用者帳戶進入預存程序可取得一角色陣列。

2. 副程式：由 RBAC 授權中心取得可用的物件資訊

由伺服器向 RBAC 授權中心取得使用者可用物件，透過使用者帳戶進入預存程序，將會取得使用者角色陣列，再由使用者角色陣列取得使用者可用物件。

3. 副程式：使用者登入後建立 Session List

使用者在通過身分確認後，將透過 RBAC 授權中心取得可用角色及物件，將這些資訊放入 Web Application 中的執行工作階段中，以方便進行可用權限的比對。

4. 主程式：使用者登入

使用者登入進行身分確認，係將使用者帳號與密碼傳遞至 RBAC 授權中心，由資料中的預存程序進行身分確認；驗證成功回傳 True，驗證失敗回傳 False。在通過身分確認完畢後，即開始建立 Session List，其過中包含向 RBAC 授權中心取得該使用者之角色與物件資訊。

由 RBAC 授權中心取的使用者角色與物件相關資訊，主程式與副程式之間互動過程，完整演算流程請參閱圖 4.10。

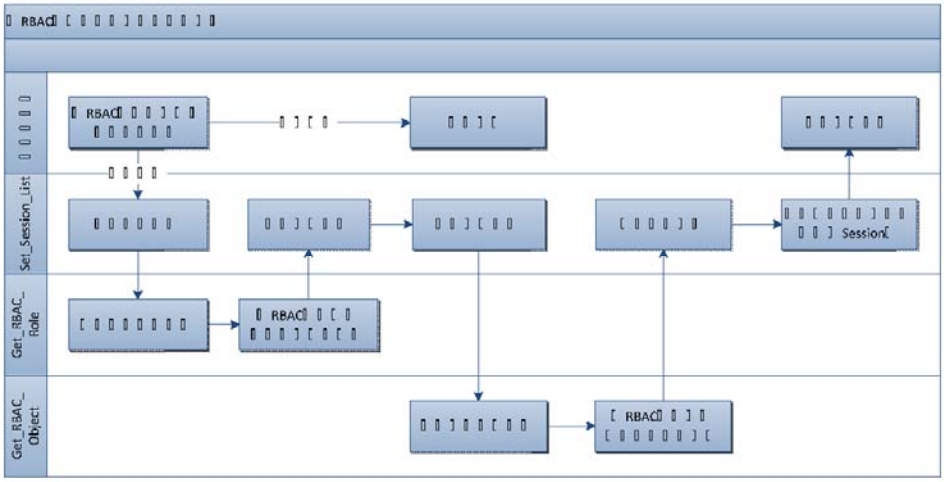

圖 4.10 由 RBAC 授權中心資訊取得演算流程

貳、權限判斷能力演算流程

使用者在進入一個網頁必須經過權限的判斷，若使用者的 Session List 帶有該網頁的 Title 名稱，即表示使用者擁有讀取這個網頁的能力。其中所包含的副程式有檢查 Session List 是否存在、檢查 Session Client 是否存在、以及借由 Session Client 取得 Session XML 轉換為 Session List。

1. 副程式：檢查 Session List 是否存在

Session List 是在使用者登入系統後，由 RBAC 授權中心回傳之角色與物件資訊儲存於使用者與伺服器執行工作階段的清單。我們可以藉由讀取特定的 Session 以判定使用者是否曾經順利的完成整個登入作業過程。由於無法確認每個使用者會取得的特定角色與物件，因此藉由登入作業時將使用者的部分資訊一併寫入

Session List 中，除了可以作為 Session List 是否存在外，也能方便其它的 Web Application 進行更人性化的介面撰寫 (例如：Session List 中存在使用者名稱 (yuuchilyann)，在使用者操作某一服務，該服務可以由 Session List 取得使用者名稱並於畫面中顯示：歡迎 yuuchilyann 使用者操作本服務。)

2. 副程式：檢查 Session Client 是否存在

Session Client 是在使用者登入系統後，在 RBAC 授權中心回傳角色與物件資訊的同時，透過預存程序調配一組雜湊值一併地回傳給使用者，並記錄於使用者的客戶端軟體上。當 Web Application 讀取 Session List 失敗時，將會確認 Session Client 是否存在，若存在將可使用 Session Client 來取得 Session XML。

3. 副程式：由 Session XML 轉換為 Session List

當使用者進入某一服務時，Web Application 無法檢測到 Session List 的存在，意味者使用者可能尚未進行登入作業過程，或使用者跨越至別的伺服器，以至於伺服器與使用者執行工作階段內並無 Session List 的存在。因此必須透過 Session Client 的存在來識別使用者是否曾經進行過登入作業過程；若使用者擁有 Session Client 即可透過該雜湊值傳遞給 RBAC 授權中心，以取得完整的 Session XML，該 Session XML 內容包含完整的使用者角色、物件、相關資訊等。在順利取得 Session XML 之後即能重新的轉換為 Session List

4. 主程式：判斷使用者是擁有操作權限

當使用者進入一項服務介面時，主要的目標任務即是判斷使用者，是否具備該服務的操作權限。使用者所有的角色資訊與物件資訊皆存放於 Session List 中，若 Session List 中並無該服務的名稱，即視為使用者無此服務的操作能力。當中延伸出判別使用者是否曾進行過登入作業過程，以及使用者是否跨越至其它伺服器操作服務。

使用者進入服務時進行權限判斷能力，主程式與副程式之間互動演算流程請參閱圖 4.11。

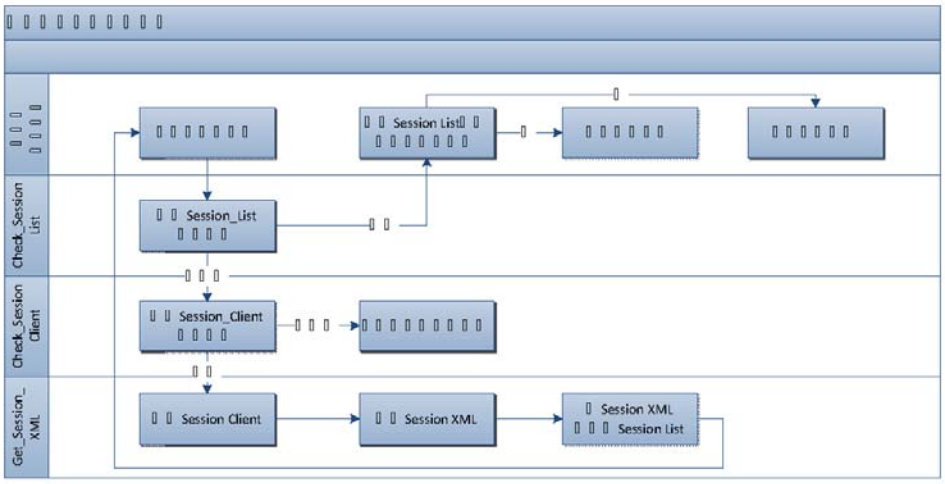

圖 4.11 權限判斷能力演算流程

第五章 結論

支援跨網域之網頁應用程式，在以往的設計可以透過下面幾種方法達成：

(1) 利用 Cookie 來攜帶資訊，但它只能攜帶少量的資訊，雖然可以藉由修改客戶端瀏覽器來達到更大量的資訊攜帶，但這必須改變使用者經驗。除此之外，Cookie 的存取被限制只能在某一設定的域名內。

(2) 利用 Memcached 與 Other Agent 的方式，藉由改變網頁應用程式存取資訊的位置，指向到一特定位置上進行存取。這個方法必須修改原本程序對於資訊存取的方式，如果原先開發某個物件的團隊已不存在，或某物件經過封裝後，無法再進行程式碼的修訂，則開發團隊群必須面臨重新開發該物件的窘境。

(3) 以使用 ASP.net 所提供的單一簽入技術為例，這裡先忽略當前的單一簽入技術，還未能對每一個物件進行低階控管的問題。採用這個方法達到支援跨網的網頁應用程式的先決條件，是所有的開發團隊必須統一使用 ASP.net。假設部分的開發團隊並不熟悉 ASP.net 的開發語言，則必須花費許多成本來解決這個問題，亦可能降低開發效率。

為避免因上述所造成的困擾，本研究機制提出一個有效的解決辦法。多數的網頁應用程式在撰寫「判讀使用者權限」程序時，皆是把使用者權限存放在 Session 內；本研究機制在客戶端與伺服器間的權限判讀，仍舊使用 Session 作為儲存和擷取使用者權限的方法，不需重新指向到某一特地位置上進行存取。當使用者存取其它伺服器時，將會由跨網域機制透過 RBAC 授權中心，取得以 XML 為格式的使用者權限授權表，重新為使用者與伺服器執行工作階段，建置完整的使用者權限授權關係；由鑑於此，伺服器之間的網頁應用程式，不再需要由同一種開發語言撰寫，無論是何種語言的網頁應用程式，僅需將 XML 格式的授權內容轉換於

Session 內，即可使網頁應用程式進行「判讀使用者權限」之能力。

本研究並未設計安全性議題，首先是由 RBAC 授權中心所產生的 Session List，它僅存在使用者與網頁應用程式間的執行工作階段內，排除伺服器與使用者系統已被植入後門，Session List 是無法用其它方式讀出，即使讀出也無法由客戶端發動將可用的 Session List 轉植入於執行工作階段中，其次是 RBAC 授權中心頒予的 Session Client 是由 128 位元的雜湊碼任意組成，並於閒置後 30 分鐘後自動失效，因此要由此來猜測可用的 Session Client 並不容易，即使得知有效的 Session Client 也僅能讀出完整的 Session List，並不能由客戶端發動轉植入使用者與網頁應用程式之間的執行工作階段中。由於網頁應用程式自身的諸多安全性限制，使本研究方法無須刻意針對安全性問題加以解決，若要加強本研究的安全性依然有因應對策。首先本研究的資訊都是經由標準 HTTP 通道進行，可由伺服器調整改為 SSL 安全通道傳輸，所有的服務、文本、參數、包含 Session XML 都能經由 SSL 安全通道傳輸。至於 Session XML 的資訊傳遞，僅在於伺服器與伺服器之間，並未經過客戶端，因此 Session XML 的傳遞可由伺服器限定僅允許核定的 IP 或域名進行溝通。若擔憂有效的 Session Client 被獲取，且是在核定允許的 IP 內，並破解 SSL 傳輸安全通道取得了 Session XML，那麼還可以針對 XML 文件內的文本進行額外加密，進一步的提高安全層級。

本研究所提出的機制僅解決透過 RBAC 授權機制，可以對所有的物件(網頁)進行詳細的角色配置，並跨越網域將完整的授權資訊傳達給所有的網頁應用程式；但並未對傳達於網頁應用程式之間的授權資訊進行最佳化。假設有五個分佈於不同網域上的網頁應用程式，每一個網頁應用程式恰好有 40 個註冊會員預設可用物件(網頁)，藉由 Session XML 發佈到每一個網頁應用程式產生使用者與網頁應用程式執行工作階段中的 Session List 將會有 200 筆，但對於單一網頁應用程式有效資料僅有 40 筆，其於 160 筆是其它網頁應用程式所需要的。未來針對本研究的持續進行，可對於 Session XML 進行最佳化，將對應於正確的網頁應用程式傳遞所需的資訊即可，避免傳遞過於完整的 Session XML，使網頁應用程式在建置 Session List 時造成額外的伺服器負擔。

參考文獻

1. 朱明中，「Web Application 首部曲」，http://www.microsoft.com/taiwan/msdn/columns/jhu_ming_jhong/A-ASP.NET_Architecture.htm
2. 李長庚，「一個開放的 Web-Based Single Sign-On 服務架構」，交通大學資訊管理學研究所碩士論文，2002 年。
3. 廖英彥，「網際網路單一簽入系統應用」，世新大學資訊管理學系碩士論文，2005 年。
4. 蘇彩妤，「以本體論為基之虛擬企業知識存取控制研究」，成功大學製造工程研究所碩士論文，2006 年。
5. Kristol D., "Http Cookie: Standards, Privacy, and Politics", ACM Transactions on Internet Technology, Vol. 1, No. 2, pp. 151-198, 2001.
6. Memcached, http://www.danga.com/memcached
7. MSDN，「工作階段狀態概觀」，http://msdn2.microsoft.com/zh-tw/library/ms178581(VS.80).aspx
8. PERSISTENT CLIENT STATE HTTP COOKIES, http://wp.netscape.com/newsref/std/cookie_spec.html
9. Sandhu R. S., "Lattice-Based Access Control Models", IEEE Computer, Vol. 26, pp. 9-19, 1993.
10. Chandramouli R., Ferraiolo D., Gavrila S., Kuhn R., Sandhu R., "Proposed NIST standard for role-based access control", ACM Transactions on Information and Systems Security, Vol. 4, No. 3, pp. 224-274, August 2001.
11. W3C, http://www.w3c.org