

應用於適性化數位學習系統之存取控制機制研究

薛夙珍

朝陽科技大學資訊管理系

呂瑞麟*

國立中興大學資訊管理學系

白琿瑜

朝陽科技大學資訊管理系

摘要

適性化學習(Adaptive Learning)近年來成為一項熱門的研究，許多專家學者針對學習者不同的學習需求，提出多種適性化學習路徑規劃機制，給予學習者個人化的學習路徑與教材導覽，以提升學習者的學習成效與學習效率。而隨著數位學習系統被廣泛的使用於各產官學界，系統的存取控制議題即成為一項重要的研究。過去有許多專家學者為資訊系統中使用者的存取權限管理提出不同的存取控制機制，可惜的是無法完全適用於適性化數位學習系統。因此本研究在綜觀目前數位學習系統的存取控制需求與適性化學習最新研究後，以適性化數位學習系統中學習資源的存取控制問題著手，結合主體角色、教學規則與授體角色的設計方式，改善數位學習系統導入適性化學習路徑導覽後，產生的過度授權及授權規則數量過多的問題，並且針對課程編輯權限委任授權之需求，提出一套適性化數位學習系統所需之存取控制機制，不僅能有效的控管學習資源的存取、降低授權規則數量及增加資源分享性，更提升了課程活動樹設計上的彈性。

關鍵字：適性化學習、角色為基礎的存取控制、教學規則

* 通訊作者

A Study of Access Control Mechanism in Adaptive e-Learning System

Sue-Chen Hsueh

Department of Information Management,
Chaoyang University of Technology

Eric Jui-Lin Lu*

Department of Management Information Systems,
National Chung Hsing University

Chen-Yui Pai

Department of Information Management,
Chaoyang University of Technology

Abstract

In recent years, the adaptive learning becomes a hot spot topic, many various planning mechanisms of adaptive learning according to different learning styles, requirements and progress of learners are proposed to promote the learning effectiveness and efficiency of learners. Due to e-learning system is widely used in the manufacturing, official and academia, an access control in e-learning system becomes another important issue. During the past decade, researchers proposed different access control mechanisms for e-learning system to manage the users' authorizations, however, they can not be completely suited for the adaptive e-learning system. As the results, the proposed research focuses on the problems of access control in learning resources and delegations of e-learning system, not only to reduce the costs of system management but also to increase the uses of resource sharing.

Key words: Adaptive learning, role-based access control, pedagogical rule

* Corresponding Author

壹、緒論

一、研究背景與目的

隨著網路通訊科技與多媒體技術的蓬勃發展，數位學習(e-Learning)已成為知識傳遞與經驗分享的重要應用。為了解決傳統“one-size-fits-all”的教學方式，提升學習者的學習成效與學習效率，適性化(Adaptive)的數位學習系統已成為一項熱門的研究(Brusilovsky & Peylo 2003; Hatzilygeroudis et. al. 2005; Chen et. al. 2006)。所謂適性化即是藉由學習評估(Learning Evaluation)的方式，取得學習者個人的學習成果、進度或預期目標等特性，結合教學規則(Pedagogical Rules)，呈現學習者在學習過程中所需存取的學習資源；授課教師亦可視學習者的學習狀況調整其教學策略與學習資源，充分發揮有如一對一個別指導的教學效果，因此數位學習系統朝向學習者需求的適性化研究儼然已成為不可擋之趨勢。

在一般的數位學習系統中，皆具有基本的存取控制機制(Access Control)，其主要的目的在於控管系統中使用者對資源的存取權限，其中以角色為基礎的存取控制機制(Role-Based Access Control，簡稱RBAC) (Sandhu & Bhamidipati 1998)使用最為廣泛。RBAC主要是以角色(Role)做為使用者權限控管的基礎，因此可以依照學習課程或學習主題定義出角色，再將有修課的學習者指派(Assign)到角色上，並且授權角色課程中所有學習資源的閱讀權限。如此一來，可有效的控管有修課的學生才允許閱讀課程中的學習內容，而未修課的學生或系統訪客(Guest)無法閱讀課程中的學習內容，並且可以使得授權規則(Authorization Rules)數量大幅的減少而降低管理成本。然而，此存取控制架構卻無法完全適用於適性化學習系統。

在適性化數位學習系統中，課程編輯者(Course Author)可依其專業及教學經驗來訂定課程活動樹(Activity Tree)，在每個章節閱讀過後會有學習評估來測試學生在此章節的學習成效，而修課學生即可依照課程活動樹的順序學習。由於適性化課程中每位學生的學習評估結果各有不同，因此所需補充閱讀的學習資源亦不相同，而以RBAC架構中將學習課程定義成一個角色，再將課程中所有學習資源的存取權限授予此角色，如此一來即會造成不必要的授權情況發生。

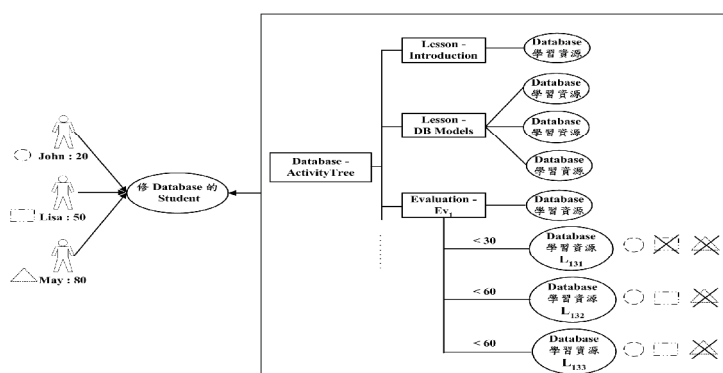
我們以一個範例詳細說明上述的問題：

今有一門課程，名稱為“Database”，如圖1所示。範例中學習評估Ev₁成績低於60分的學生需閱讀補充學習資源L₁₃₂與L₁₃₃，而評估成績低於30分的學生需閱讀補充學習資源L₁₃₁，然因評估成績低於30分的學生亦低於成績60分，因此總共需閱讀L₁₃₁、L₁₃₂與L₁₃₃三份補充學習資源。學生John在Ev₁中學習評估成績為20分，需補充學習資源L₁₃₁、L₁₃₂與L₁₃₃，學生Lisa學習評估成績為50分，需補充學習資源L₁₃₂與L₁₃₃，而學生May學習評估為80分，則不需補充任何學習資源¹。

¹ 圓圈表示John對學習資源L₁₃₁、L₁₃₂、L₁₃₃的存取需求，方型表示Lisa對學習資源L₁₃₁、L₁₃₂、L₁₃₃的存取需求，三角型表示May對學習資源L₁₃₁、L₁₃₂、L₁₃₃的存取需求。

而以RBAC架構中，我們設定角色“修Database的Student”擁有存取課程中所有學習資源的權限，然而對學生Lisa而言，不需學習資源L₁₃₁的存取權限，而對學生May而言亦不需學習資源L₁₃₁、L₁₃₂與L₁₃₃的存取權限，如此一來即造成不必要的授權。

再者，RBAC存取控制機制在資源存取權限的訂定上亦具有改進空間。由於RBAC是針對課程中的學習資源逐一授權的方式，如圖1中John的範例而言，需產生8項授權規則，以Lisa而言，產生了7項授權規則，以May而言，則產生了5項授權規則；而隨著課程中的教學規劃愈複雜，學習資源數量愈來愈多的情況下，授權規則的數量即會變得十分龐大且難以管理；而在學習資源變更的彈性上更受到了侷限。當課程中有一份學習資源L₁₂₃不符合教學需求時，課程編輯者要以學習資源L₁₂₄取代時，以傳統RBAC的存取控制方式要將L₁₂₃的存取權限移除，並且新增L₁₂₄的存取權限，而此方式會造成存取權限的變動，增加學習資源的變更成本。



因此，為了解決適性化授權的問題，我們巧妙的將教學規則與Rule-based存取控制策略結合，當課程中學習者需要補充學習資源時，才動態授權予學習者，此方式即可避免不必要的授權情況發生。

而在授權數量與變更彈性的問題上，本研究延伸Moyer與Ahamad(2001)所提出授體角色(Object Role)的概念，並且進一步定義授體角色階層(Object Role Hierarchy)，有效的強化授體角色之應用。我們將相同存取權限的學習資源指派至授體角色中，如此一來能夠有效的減少授權數量的產生，增加學習資源變更上的彈性；並且以角色階層的特性，更進一步簡化重複性的存取權限指派。

最後，為了提供教學上的彈性，使教學內容更符合學習者需求，課程編輯者可授權授課教師修改課程活動樹之權限，在存取權限上即產生委任授權(Delegation)之需求。然而在一門課程中，可能包含多個班級(Section)，當某個班級的授課教師修改了課程活動樹，為了不影響其它班級的存取權限，我們提出了權限取代(Replace)的方式，取代修改班級中主體角色原本的存取權限，並不是直接對原本的存取權限做更改，如此一來，其他班級的存取權限才不會受到影響。

我們利用系統實作驗證所提出之方法的可行性，經由實際系統雛型的驗證顯示，本

研究所提出的模型架構相較於一般以RBAC為基礎的數位學習系統中，在授權上更符合適性化的需求，並且在課程架構的彈性與存取權限數量上皆有明顯的改善。

本文在綜觀適性化學習最新研究後，以適性化數位學習系統中學習資源的存取控制與委任授權角度著手，著重於理論與實務之結合。在第二節文獻探討中，將探討目前應用於Web-based的存取控制機制相關研究與發展，所包含的內容：RBAC、RuBAC、GRBAC與XML相關研究。第三節中我們改善傳統RBAC架構，結合適性化學習中的教學規則與Rule-based存取控制策略及授體角色的應用，產生具有適性化的SALeS(SCORM-based Adaptive e-Learning System)存取控制模式，並對傳統Credential模式、RBAC模式與SALeS模式進行分析比較。第四節為系統架構，此章節中詳細介紹SALeS模式中各個檔案之功能與資料儲存結構。第五節為系統雛型，我們將SALeS模式實作出適性化學習系統，並以使用者的角度，說明系統中各個功能。最後在第六節中，將對於我們的研究做一個結論與建議。

貳、文獻探討

一、存取控制模式

存取控制目的在於避免系統中的資源被惡意使用者故意破壞、刪除、或更改，針對系統中的使用者擁有何種物件的存取權限與存取的一項保護措施，因此存取控制在網路通訊與資訊安全領域中扮演著極為重要的角色。

(一) 以角色為基礎的存取控制機制(Role-based Access Control, RBAC)

在系統存取控制領域中主要以Sandhu等學者所提出的RBAC使用的最為廣泛，而在2001年由美國國家標準與技術研究院(National Institute of Standards and technology，簡稱NIST)加以收編、整理後訂定標準，稱為NIST RBAC(Ferraiolo et. al. 2001)。

在NIST RBAC基本模型架構中具有下列定義：

- USERS、ROLES、OPERATIONS與OBJECTS：分別表示為系統中的使用者、主體角色、對授體的操作權限與授體。
- $UA \subseteq \text{USERS} \times \text{ROLES}$ ：表示使用者與主體角色間有多對多的指派關係，亦即使用者可以擁有多個主體角色，而主體角色亦可被分派給多位使用者(User Assignment)。
- $\text{PERMISSIONS} = 2^{(\text{OPERATIONS} \times \text{OBJECTS})}$ ：權限，表示授體可擁有的操作權限。
- $PA \subseteq \text{PERMISSIONS} \times \text{ROLES}$ ：主體角色和權限間有多對多的指派關係，即主體角色可以擁有多個權限，而權限則可同時被多個主體角色所擁有(Permission Assignment)。

在RBAC控制模式中，可將主體角色視為組織中某一職務，再依據主體角色給予不同的權責(PERMISSIONS)，最後將一群使用者指派至主體角色。當使用者要改變其權責時，只需把使用者指派到另一個新的主體角色上，取消原主體角色的連結，且原主體角

色與新主體角色中其它的使用者並不會有任何的影響；如此一來可以使得授權規則大幅度的減少，並且能有效的降低授權上的管理成本。RBAC存取架構雖然可以有效的管理系統中使用使用者對系統資源的存取權限，但是隨著系統應用愈趨於複雜，使用者需求愈趨於多變的情況下，RBAC已無法滿足需求，因此另有專家學者提出以“角色”為基礎並且加以延伸應用的存取控制模式。

（二）以規則為基礎的RBAC存取控制模式 (Rule-based RBAC, RB-RBAC)

以規則為基礎的存取控制(Al-Kahtain & Sandhu 2004)，是個以規則為主的資訊安全政策(Policy)，實施的方式是根據系統管理者事先定義系統內資源的管制規則(Rule)，利用規則的設定，協助使用者在符合某一特定條件時能動態指派至角色中，以存取角色相關權限，因此能更有效的減少管理成本。規則的描述一般都是由條件(Condition)與活動(Action)所結合，並且會以使用者屬性、資源屬性或環境屬性等加以設定，當政策內容同時擁有多個規則時，可依其重要性排列規則執行的優先順序。

而在適性化數位學習系統中，藉由授課教師在課程活動樹中設定教學規則，利用教學規則來控管學習者的學習路徑，調整學習者在學習的過程中所需存取的學習內容。例如，若學習者U所上的課是“資料庫管理系統”的“SQL語法”，如果U學習“SQL語法”的效果不好，系統就可以讓U存取“集合的簡介”，我們就可以設計出如 $can_access(U, \text{“集合的簡介”}) \leftarrow (Role(U) = \text{“資料庫管理系統”} \cap Grade(\text{“SQL語法”}) < 60)$ 的規則，因此我們即可控制當學習者對學習資源產生需求時，才動態的配置“集合的簡介”給U，如此一來即可有效的解決學習資源適性化授權的問題。

然而，將RBAC的存取架構結合教學規則，應用於適性化數位學習系統會產生如在緒論中討論過的授權規則數量過多問題，因此在本文中將結合RBAC架構與授體角色的概念，改善授權規則數量的問題。

（三）GRBAC存取控制模式 (Generalized Role-based Access Control, GRBAC)

在Moyer與Ahamad(2001)的研究中，認為角色的概念不僅止以使用者(Subject)的觀點，亦可以系統資源(Object)或系統環境(Environment)來劃分，分別為主體角色(Subject Role)、授體角色(Object Role)與環境角色(Environment Role)三種。其中主體角色即是以使用者的觀點所建立的角色(例如：授課教師或學生等)；而授體角色即是依據系統資源的型態(例如：信件或書籍等)或檔案類型(例如：.mp3或.bmp等)，對眾多的授體做角色的分類，角色的存取授權是以授體所屬的類型為基礎，而不是傳統上直接指定授體存取的方式。例如：在使用者Tom被限定，僅能在下午3:00～5:00間，存取.mp3類型的檔案，因此在此存取政策中，“.mp3類型”即為授體角色，而不是針對每一個.mp3的檔案各別逐一的給予授權，因此應用授體角色可有效的控管同一類型的授體存取權限，使授權管理更為簡易迅速。

因此在我們的所提出的存取控制架構中將延伸授體角色的概念，並且進一步定義授體角色階層，利用授體角色，以減少授權規則數量的產生，並且以角色階層的特性，更進一步簡化重複性的存取權限指派。

二、XML-based Access Control

在近年來，XML「可擴展標示語言」(eXtensible Markup Language)成為資訊傳播的重要格式。由於XML 具備可擴展性、資料與樣式分離等特色，各個系統可根據自身的需求對資料進行其他加值處理，因此能有效的表達網路上的各種知識，為資料的交換與處理提供新的機制；例如在Murata等學者(2006)的研究中，利用XML來描述RBAC，將XML具有豐富結構的特性及RBAC階層的特性加以結合，能有效且精細地控管角色的存取權限。

在Fundulaki與Marx(2004)的研究中，便將各專家學者對存取控制的策略定義加以歸納整理，XML存取控制的規則定義應包含授權存取者(Requester)：其認證資訊可為使用者資訊或系統中的角色，例如role_type=“Instructor”，利用這些資訊來驗證該使用者或角色，並賦予適當的執行權限與資源系統資源；系統資源(Resource)：即為此政策描述中，允許存取的系統資源，由於系統資源架構是以XML方式表達，因此可利用XPath的方式走訪系統資源的節點；執行的動作(Action)：指存取資源的方式，例如新增、修改、刪除及查詢功能；影響(Effect)：表示此政策授權與否，例如effect=” Grant”表示正授權，而effect=” Deny”表示負授權；存取範圍(Propagation)：表示此政策的執行範圍，其範圍有僅限於節點本身 “The node only”、本節點與屬性 “The node and its attributes”、本節點與其子文字節點 “the node and its text node children” 及節點與其所有子節點 “the node, its attribute, all its descendents and their attributes”，由上述五個元件組成的一個集合。

由於目前的XML剖析器(XML Parser)皆支援XPath 1.0，因此本研究即採用XPath 1.0的節點描述方式來表達系統內各角色的存取政策與委任授權，並且利用XML檔案來記錄系統中的各項資訊，以達到細微的存取控制。

參、模式設計與分析

一、模式設計

為了有效的達到適性化系統中學習資源的存取控制，動態的授權學習資源的閱讀權限，本研究將教學規則與Rule-based存取控制策略結合，藉由學習者的評估成績，動態給予學習資源的閱讀權限。在進一步說明存取控制規範的相關用法之前，先將常用的邏輯符號定義做詳細的說明，其定義如下：

- $u \in USERS$ (1)
- $a \in ACTIONS$ (2)
- $l \in L$ (3)
- $PERM = 2^{(ACTIONS \times L)}$ (4)
- $p \in PERM$ (5)

在定義(1)中，USERS表示適性化系統中使用使用者集合；定義(2)中，ACTIONS表示操作的集合，其值可以是閱讀、新增、刪除或修改；定義(3)中，L表示系統中的學習資源集合；而在定義(4)中，PERM為存取權限的集合，為操作與學習資源間的指派關係；定義(5)中，p即表示單一的存取權限。

我們採用Role-based存取控制概念，將學習課程定義成一個主體角色(Subject Role，簡稱SR)，再將選修課程的學習者指派至主體角色中，如此方式可有效控管有修課的學習者才允許閱讀課程中的學習內容。而主體角色存取控制定義如下：

$$sr \in SR \quad \dots\dots\dots(6)$$

$$UA \equiv USERS \times SR \quad \dots\dots\dots(7)$$

$$assigned_users(u, sr) = \{u \mid (u, sr) \in UA\} \quad \dots\dots\dots(8)$$

$$currentSR(u) = \{sr \mid (u, sr) \in UA\} \quad \dots\dots\dots(9)$$

$$SP \equiv SR \times PERM \quad \dots\dots\dots(10)$$

在定義(6)中，SR表示主體角色的集合；定義(7)中，UA表示使用者與主體角色間的指派關係；定義(8)中以assigned_users()函數表示將主體角色“sr”指派予使用者“u”，而每個使用者可擁有一至多個主體角色；定義(9)中，currentSR()函數表示使用者在此一session期間所擔任的主體角色；定義(10)中，SP表示主體角色與存取權限的指派關係。

在適性化數位學習系統中，主要是利用學習評估與教學規則的方式，針對課程中不同學生的學習評估結果，依教學規則中指定的學習資源，產生學生適性化的學習路徑。以圖1為例，課程“Database”的課程編輯者在建立課程活動樹時，即設定學習評估Ev1中評估成績低於30分的學習者，需補充學習資源L131及評估成績低於60分的學習者，需補充學習資源L132與L133，因此在教學規則的設計上分別以Rule_{A-1}、Rule_{A-2}及Rule_{A-3}函式來表示：

$$Rule_{A-1} : IF DB_{sr}.score(DB.Ev1) < 30 THEN read(DB.L_{131})$$

$$Rule_{A-2} : IF DB_{sr}.score(DB.Ev1) < 60 THEN read(DB.L_{132})$$

$$Rule_{A-3} : IF DB_{sr}.score(DB.Ev1) < 60 THEN read(DB.L_{133})$$

在Rule_{A-1}函式中，DB_{sr}.score(DB.Ev1)函數表示修Database的學習者“DB_{sr}”在課程Database“DB”的學習評估“Ev1”中得到的成績；而read(DB.L₁₃₁)函數表示擁有權限閱讀課程“DB”中的學習資源“L₁₃₁”，因此Rule_{A-1}函式即表示若修Database的學習者在學習評估Ev1中成績低於30分者，需閱讀學習資源L₁₃₁；當學習者進行學習時，隨著不同學習者在學習評估中得到不同的成績，對學習資源的存取需求即不相同。特別一提的是，在教學規則的設計中，亦可設定當學習評估成績高於80分的學習者，也可以補充某些進階的學習資源；換句話說，符合教學規則的學習者才需補充學習資源，而未符合教學規則的學習者則不需補充學習資源。

Rule _{B-1} : allow(u,p _i) ←	currentSR(u) = DB _{sr} && condition(DB _{sr} .score(DB.Ev ₁)<30). p _i =read × DB.L131
Rule _{B-2} : allow(u,p _j) ←	currentSR(u) = DB _{sr} && condition(DB _{sr} .score(DB.Ev ₁)<60). p _j =read × DB.L132
Rule _{B-3} : allow(u,p _k) ←	currentSR(u) = DB _{sr} && condition(DB _{sr} .score(DB.Ev ₁)<60). p _k =read × DB.L133

然而在存取權限的設計中，針對課程中的學習資源逐一授權的方式，授權規則的數量仍然十分龐大，並且侷限了學習資源變更上的彈性。因此我們應用授體角色(Object Role，簡稱OR)的概念，利用授體角色做為存取權限與學習資源間的媒介，將相同存取權限的學習資源指派至授體角色中，不直接授權學習資源存取權限予主體角色，如圖2所示，當授課教師想要在某個章節中補充學習資源時，僅需將學習資源指派至授體角色中即可；而授體角色中的學習資源形態亦可以是課程練習或系統實作等非順序性的學習內容。如此一來，不僅能有效減少存取權限數量，更大幅提升課程活動樹設計上的彈性。授體角色存取控制定義如下：

$$\begin{aligned} \text{or} &\in \text{OR} && \dots\dots\dots(11) \\ \text{OL} &\equiv \text{OR} \times \text{L} && \dots\dots\dots(12) \\ \text{PERM} &= 2^{(\text{ACTIONS} \times \text{OR})} && \dots\dots\dots\cancel{(13)} \rightarrow (13) \\ \text{assigned_objects(l,or)} &= \{ l \mid (l, \text{or}) \in \text{OL} \} && \dots\dots\dots(14) \end{aligned}$$

Rule_{C-1} : allow(u, p_i) $\leftarrow$ currentSR(u) = DBsr && condition(DBsr.score(DB.Ev₁) < 30).
p_i = read $\times$ or₁ , or₁ = {DB.L₁₃₁}

Rule_{C-2} : allow(u, p_j) $\leftarrow$ currentSR(u) = DB_{sr} && condition(DB_{sr}.score(DB.EV₁) < 60).
 p_j = read $\times$ or_j, or_j = {DB.L₁₃₂、DB.L₁₃₁}

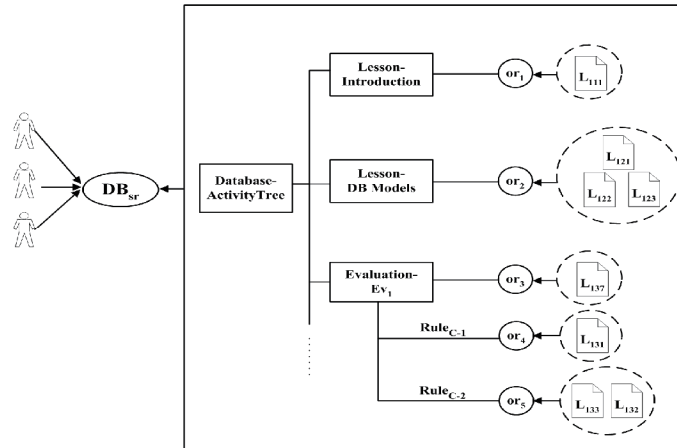


圖2：存取權限、授體角色與學習資源間的關係

授體角色除了解決上述存取權限數量過多與教材授權的彈性問題外，亦可形成樹狀架構。在教學的設計上，教學規則間通常可具有存取關連性，舉例來說：在學習評估DB.EV₁中，評估成績低於30分的學習者，除了需閱讀成績低於30分的補充學習資源，亦需閱讀成績低於60的補充學習資源，因此本研究在授體角色的設計上採用“階層”的特性，上階層的授體角色允許擁有下階層授體角色對學習資源的存取權限，如此一來可有效減重複性的存取權限指派，因此低於30分的學習者即需閱讀學習資源L₁₃₁、L₁₃₂與L₁₃₃，如圖3所示。

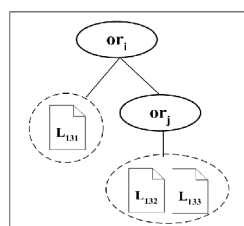


圖3：授體角色階層

為了提供教學上的彈性，使教學內容更符合學習者需求，課程編輯者可授權授課教師修改課程活動樹之權限，因此即產生委任授權之需求；而此委任授權方式除了提供教學的彈性外，當授課教師修改課程活動樹後，其活動樹順序或課程內容等相關的教學資訊，亦可透過回饋機制，提供課程編輯者做為參考。

在課程的設計上，一門課程經常會有多個班級(Section)，而每個班級有各自的授課教師、助教與學習者，當課程編輯者授權授課教師A修改課程活動樹時，授課教師A可視

教學之需求，對課程活動樹做修改；然而，其他班級的學習者所閱讀的課程活動樹仍需為課程編輯者所編輯的版本，因此當授課教師A更改課程活動樹內容時，需產生新的存取權限，利用取代(Replace)的方式，取代A班中學習者原本的存取權限，並不是直接對原本的存取權限做更改，如此一來，其他班級的存取權限才不會受到影響。因此委任授權之定義如下：

$$\text{can_delegate_perm}(\text{srtor}) = \{p \mid (p, \text{srtor}) \in \text{SP}\} \quad \dots\dots\dots(15)$$

我們將課程編輯者稱為“委任者”(Delegator)，而授課教師稱為“授任者”(Delegatee)，委任者只允許授權屬於自己的權限給授任者。在定義(15)中，can_delegate_perm()函數表示主體角色僅允許授權屬於自己的權限集合，其中srtor表示委任者主體角色。

二、SALeS模式分析

我們以傳統Credential模式、RBAC模式與SALeS模式在學習資源的授權數量上分析比較。假設課程“DB”中某學習評估設定評估成績區間為 $0 \leq \text{成績} < 30$ 分的學習者需補充學習資源 $L_{1 \sim 9}$ ，成績區間 $30 \leq \text{成績} < 50$ 分需補充學習資源 $L_{4 \sim 9}$ ，而成績區間 $50 \leq \text{成績} < 70$ 分需補充學習資源 $L_{7 \sim 9}$ ，修課學生分別為John、Lisa與May，三位學習者的成績皆介於成績區間 $0 \leq \text{成績} < 30$ 之間。

由於Credential的方式必需對每個使用者與每個學習資源間的存取權限做逐一的授權，在三位學習者皆需補充學習資源 $L_{1 \sim 9}$ 的情況下，則需產生27項授權，即 3×9 項授權。在RBAC模式中，我們可以將John、Lisa與May指派至“DB_{sr}”角色中，並且授權DB_{sr}學習資源 $L_{1 \sim 9}$ 的存取權限，則需產生9項授權。其實，不論是Credential模式或者是RBAC模式，它們的授權方式都是靜態的；也就是說，不論學生學習的成績為何，系統都必須在事前先把這些權限授權出去，也因此造成了不必要的授權數量增加。

在SALeS模式中，我們應用了授體角色，並以Rule-based的方式動態授權，因此授權數量可以大量降低，而且有效的解決不必要的授權情況發生；以相同的範例，課程編輯者或者授課教師可以設定三個“階層式”的受體角色，分別為 $\text{or}_1 = \{L_{7 \sim 9}\}$ 、 $\text{or}_2 = \{L_{4 \sim 6}\}$ 、以及 $\text{or}_3 = \{L_{1 \sim 3}\}$ ，且 or_3 是最高階層，然後是 or_2 ，最低階層的是 or_1 ；因此僅需產生3項授權規則（即成績區間為 $0 \leq \text{成績} < 30$ 分的學習者可存取 or_3 ，成績區間 $30 \leq \text{成績} < 50$ 分的學習者可存取 or_2 ，成績區間 $50 \leq \text{成績} < 70$ 分的學習者可存取 or_1 ）。SALeS模式除了可以降低授權數量以降低存取管理的成本之外，它也允許在不變動授權規則的情形下，讓授課教師或者教材編輯者調整補充學習資源；例如，假設我們需要將補充學習資源 L_4 改成X，我們只需將 L_4 從 or_2 移除，並將X加入 or_2 即可。

若以C代表課程數量，假設每一課程的學生數量都是N，每一課程的評估成績區間（或者授權角色的數量）為I，而每一個成績區間的補充學習資源數量為L。在最差的情形下，所有學生學習成果都和範例中的John、Lisa與May一樣，則以傳統Credential模

式，最多需要增加CxNxIxL項授權；若將每一課程指派一個角色並將該課程的學生都指定為該角色，則以RBAC模式，最多需要增加CxIxL項授權；若採用SALeS模式，最多只需要增加CxI項授權規則。

肆、系統架構

為了實踐設計論理，本文提出SALeS模式，其架構如圖4所示，包含學習資源儲存庫(Learning Resource Repository)、主體角色(Subject Roles)、使用者資訊(User Information)、主體角色權限(Subject Policies)、委任授權記錄(Delegation)、系統課程(Courses)、授體角色(Object Roles)與學習者學習記錄(Learning Record)，以下將說明各個檔案的功能與資料儲存結構。

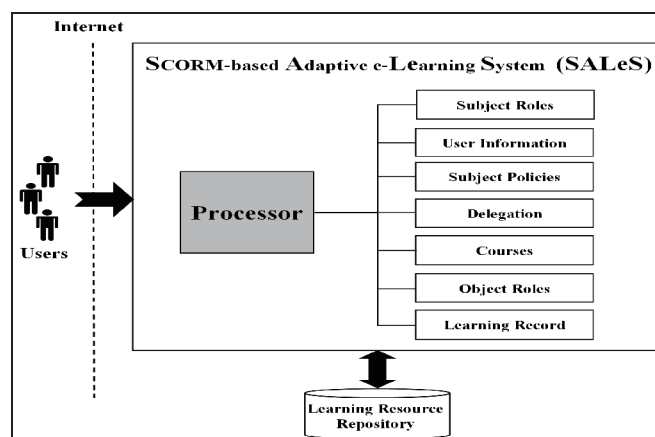


圖4：SALeS model

一、學習資源儲存庫

為了針對適性化學習資源存取控制之研究，我們假設此模式應用範圍屬於學習管理系統(Learning Management System, 簡稱LMS)，並且應用學習資源檔(LearningResources.xml)來模擬系統中SCO-based學習資源儲存庫(Repository)，系統所需之學習資源已編輯完成，並且儲存學習資源的實體檔案路徑於學習資源儲存庫中。學習資源檔的文件定義格式(Schema；以W3C XML Schema定義)如圖5所示，根元素<LearningResources>為學習資源的集合，表示所模擬的學習資源儲存庫，其子元素<SCO>則代表LMS中可管理與追蹤的最小單位學習資源，每個SCO節點表示一份學習資源，其屬性id表示此份學習資源識別，其值具有唯一性；在實務上識別值可採用通用唯一識別碼(Universally Unique Identifier, 簡稱UUID)的產生方式來避免識別值重複的可能性，由於UUID在顯示上過於冗長，為了便於說明，本研究中識別值的產生方式採用時間序列×亂碼再轉換為長整數的10位數表示方式。屬性label則表示此份學習資源的描述，為了能夠有效的達到學習資源

的重複使用性，我們提供學習資源的metadata，以利課程編輯者與授課教師對學習資源的搜尋，屬性keyword表示此份學習資源的關鍵字，屬性owner表示上傳此份學習資源的系統使用者，屬性path為學習資源於系統中的實體檔案存取路徑。

```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <xsd:schema elementFormDefault="qualified" xmlns:xsd="http://www.w3.org/2001/XMLSchema">
3   <xsd:element name="LearningResources" type="LerningResourcesType" />
4   <xsd:complexType name="LearningResourceType">
5     <xsd:sequence>
6       <xsd:element minOccurs="0" maxOccurs="unbounded" name="SCO" type="SCOType" />
7     </xsd:sequence>
8   </xsd:complexType>
9   <xsd:complexType name="SCOType">
10    <xsd:attribute name="id" type="xsd:string" />
11    <xsd:attribute name="label" type="xsd:string" />
12    <xsd:attribute name="keyword" type="xsd:string" />
13    <xsd:attribute name="learningResourceType" type="xsd:string" />
14    <xsd:attribute name="owner" type="xsd:string" />
15    <xsd:attribute name="path" type="xsd:string" />
16  </xsd:complexType>
17 </xsd:schema>

```

圖5：LearningResources.xsd

學習資源儲存庫XML檔如圖6所示，當課程編輯者欲搜尋與“Database”相關的學習資源時，可利用label、keyword或owner等屬性查詢的方式，找出屬性值包含“Database”字串的學習資源；例如以字串“Database”搜尋屬性“Keyword”，則會找出6份學習資源，分別位於行3-行8中。

```

1 <?xml version="1.0" encoding="UTF-8" ?>
2 <LearningResources xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
3   xsi:noNamespaceSchemaLocation="LearningResources.xsd">
4   <SCO id="cRt8X2MCJc" label="Introduction to Database" keyword="Database,DB History" owner="Tom"
5     learningResourceType="NarrativeText" path="/CourseImports/Course-1/Database/AA.htm"/>
6   <SCO id="chXpNj0mxu" label="Database Models" keyword="Database,Hierarchical model,Relation model" owner="Tom"
7     learningResourceType="NarrativeText" path="/CourseImports/Course-1/Database/AB.htm"/>
8   <SCO id="f3A5y1NCBj" label="DBMS" keyword="Database,Relational DBMS" owner="Cathy"
9     learningResourceType="NarrativeText" path="/CourseImports/Course-1/Database/BA.htm"/>
10  <SCO id="nou8Qe5mEV" label="DBMS Internals" keyword="Database,Database Design" owner="Cathy"
11    learningResourceType="NarrativeText" path="/CourseImports/Course-1/Database/BB.htm"/>
12  <SCO id="kCRt3vk6y9" label="Apps. of DB" keyword="Database,Database Applications" owner="Cathy"
13    learningResourceType="NarrativeText" path="/CourseImports/Course-1/Database/CA.htm"/>
14  <SCO id="K95m3P_CHz" label="DBMS Products" keyword="Database,MySQL,SQL Server" owner="Cathy"
15    learningResourceType="NarrativeText" path="/CourseImports/Course-1/Database/CB.htm"/>
16  <SCO id="cRt8X2MCDb" label="File-based Systems" keyword="File-based System" owner="Frank"
17    learningResourceType="NarrativeText" path="/CourseImports/Course-3/Supplement/AEA.htm" />
18  ...
258 </LearningResources>

```

圖6：LearningResources.xml

二、主體角色

使用者於登入後必須選擇該次登入所將扮演的主體角色。為了表達使用者對數位學習系統中學習資源的存取控制，本研究在主體角色種類的定義上將結合Wang (2006)與Khan (2004)的研究，歸納LMS中課程規劃、課程設計、教學階段所需包含的主體角色種類與權責，並假設數位學習平台系統環境已開發完成，因此系統建置與行銷階段所需的主體角色不在此討論範圍；所定義的主體角色種類與角色職責共為六種，分別為：(1)系統管理員(Administrator)，負責建立使用者註冊資料，管理使用者帳戶，而因

學習資源存取控制之需求，系統管理員亦可將學生指派至課程的學習者角色；(2)課務主任(Director)，負責制定系統中所需之課程，並且可指派使用者擔任課程編輯者(Course Author)或授課教師(Instructor)，若是學校系所中，課務主任即類似系主任的職位；(3)課程編輯者之職責即是依據所被指派的課程，利用學習資源儲存庫中的學習資源，建置課程活動樹及訂定教學規則。若課程編輯者欲編寫學習資源內容，即需利用LCMS新增學習資源，再將學習資源導入課程活動樹中，而課程編輯者亦可經由授權的方式，授予授課教師更改課程中學習評估的內容及教學規則，使課程更符合授課教師的教學規劃；(4)授課教師在經由課程編輯者授權後，可更改學習評估與教學規則，並且可指派使用者擔任課程中的助教(Tutor)；(5)助教之職責為協助授課教師監督學習者的學習進度；(6)學習者(Student)即負責課程內容的學習。

在SALeS模式中，為了有效的管理不同主體角色對課程中學習資源的存取權限，除了系統管理員與課務主任之外，課程編輯者、授課教師、助教與學習者四種主體角色，會隨著課程的建立由系統自動產生相對應的主體角色。主體角色檔Schema如圖7所示，根元素<Subject_Roles>表示系統內所有主體角色的集合，其子元素為<Subject_Role>及<Course_Roles>；由於“系統管理員”與“課務主任”不屬於任何特定課程的角色，因此我們將此二種主體角色獨立成為元素<Subject_Roles>之子元素，以元素<Subject_Role>表示，其屬性id表示主體角色的識別，屬性label表示主體角色的描述，而屬性role_type表示此主體角色的種類，其值可為Administrator或Director；而元素<Course_Roles>表示與課程相關的角色，此元素會隨著課程建立的同時而產生，其目的在於簡化主體角色中對課程重複描述的資訊，其子元素<Subject_Role>所用的課程資訊皆與此元素相關；元素<Course_Roles>擁有屬性course_id與xmldoc，其中屬性course_id表示所索引的課程識別，屬性xmldoc為屬性course_id所索引的XML檔。一般而言，一門課程中僅有一個課程編輯者角色，而為了控管修課人數，會區分出多個班級(Section)，而課程編輯者所編輯的課程內容為不同Section所共同使用，因此元素<Course_Roles>即擁有子元素<Subject_Role>及<Section>，其中<Subject_Role>的角色種類為CourseAuthor；元素<Section>表示主體角色所屬的班級，目的亦是用於簡化其子元素<Subject_Role>對Section的描述，其屬性id表示班級代號，屬性label表示班級名稱，子元素<Subject_Role>則表示班級中的主體角色，其角色種類值僅可為Instructor、Tutor或Student。

```

1  <?xml version="1.0" encoding="UTF-8"?>
2  <xsd:schema elementFormDefault="qualified"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
3    <xsd:element name="Subject_Roles" type="Subject_RolesType" />
4    <xsd:complexType name="Subject_RolesType">
5      <xsd:sequence>
6        <xsd:element minOccurs="1" maxOccurs="unbounded"
  name="Subject_Role" type="Subject_RoleType" />
7        <xsd:element minOccurs="0" maxOccurs="unbounded"
  name="Course_Roles" type="Course_RolesType" />
8      </xsd:sequence>
9    </xsd:complexType>
10   <xsd:complexType name="Course_RolesType">
11     <xsd:sequence>
12       <xsd:element minOccurs="1" maxOccurs="unbounded"
  name="Subject_Role" type="Subject_RoleType" />
13       <xsd:element minOccurs="1" maxOccurs="unbounded"
  name="Section" type="SectionType" />
14     </xsd:sequence>
15     <xsd:attribute name="course_id" type="xsd:string" />
16     <xsd:attribute name="xmlDoc" type="xsd:string" />
17   </xsd:complexType>
18   <xsd:complexType name="SectionType">
19     <xsd:sequence>
20       <xsd:element minOccurs="1" maxOccurs="unbounded"
  name="Subject_Role" type="Subject_RoleType" />
21     </xsd:sequence>
22     <xsd:attribute name="id" type="xsd:int" />
23     <xsd:attribute name="label" type="xsd:string" />
24   </xsd:complexType>
25   <xsd:complexType name="Subject_RoleType">
26     <xsd:attribute name="id" type="xsd:string" />
27     <xsd:attribute name="label" type="xsd:string" />
28     <xsd:attribute name="role_type" type="eLearning_RoleType" />
29   </xsd:complexType>
30   <xsd:simpleType name="eLearning_RoleType">
31     <xsd:restriction base="xsd:string">
32       <xsd:enumeration value="Administrator"/>
33       <xsd:enumeration value="Director"/>
34       <xsd:enumeration value="CourseAuthor"/>
35       <xsd:enumeration value="Instructor"/>
36       <xsd:enumeration value="Tutor"/>
37       <xsd:enumeration value="Student"/>
38     </xsd:restriction>
39   </xsd:simpleType>
40 </xsd:schema>

```

圖 7：SubjectRoles.xsd

主體角色XML檔如圖8所示，檔案中行5-行17為課程“資料庫管理系統”相關的主體角色，由於課程“資料庫管理系統”擁有二個班，分為A班與B班，因此課程編輯者所編輯的課程內容為A與B班共同使用，而班級中則會有各自的授課教師、助教與學習者，因此行6為資料庫管理系統課程編輯者角色，而行8-行10與行13-行15為A與B班各自的授課教師、助教與學習者角色。

```

1  <?xml version="1.0" encoding="UTF-8" ?>
2  <Subject_Roles xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="SubjectRoles.xsd">
3    <Subject_Role id="OBjekIwhir" label="Administrator" role_type="Administrator" />
4    <Subject_Role id="gK9i2bx1ul" label="Director" role_type="Director" />
5    <Course_Roles course_id="MIS-970001" xmlDoc="Courses.xml">
6      <Subject_Role id="tvjLOqOLj7" label="Database(id=MIS-970001):CourseAuthor" role_type="CourseAuthor" />
7      <Section id="1" label="A">
8        <Subject_Role id="7iehIAELuy" label="Database-A(id=MIS-970001-1):Instructor" role_type="Instructor" />
9        <Subject_Role id="QPsIbMtlF" label="Database-A(id=MIS-970001-1):Tutor" role_type="Tutor" />
10       <Subject_Role id="PIpdUgYLoB" label="Database-A(id=MIS-970001-1):Student" role_type="Student" />
11     </Section>
12     <Section id="2" label="B">
13       <Subject_Role id="hhJys0dfuU" label="Database-B(id=MIS-970001-2):Instructor" role_type="Instructor" />
14       <Subject_Role id="NuihhhXLk8" label="Database-B(id=MIS-970001-2):Tutor" role_type="Tutor" />
15       <Subject_Role id="z2B7jUlfse" label="Database-B(id=MIS-970001-2):Student" role_type="Student" />
16     </Section>
17   </Course_Roles>
18   <Course_Roles course_id="MIS-970002" xmlDoc="Courses.xml">
19     ...
20     ...
21     ...
22   </Course_Roles>
23   ...
24   ...
25 </Subject_Roles>

```

圖 8：SubjectRoles.xml

三、使用者資訊

使用者資訊檔所記錄的資料為使用者的帳號資訊與授予的主體角色，使用者初次使用系統時，需由系統管理者先行建立使用者帳號與密碼，經由指派主體角色的程序後，每個使用者可擁有0至多個主體角色，而在每個session期間只允許擔任一個主體角色，使用者資訊檔Schema如圖9所示，根元素<Users>表示系統中使用者集合，屬性xmlDoc為屬性subject_role_id所索引的XML檔，其子元素<User>表示每位使用者的相關資訊，其屬性id為使用者於系統中登入的帳號，屬性值具有唯一性，可為學號或教職員編號，屬性passwd為登入系統的驗證密碼，在實務上，passwd值也可以依照XML Encryption的方式進行加密，屬性username為使用者的姓名；由於每位使用者可被授予0至多個主體角色，因此元素<User>可擁有多個子元素<Subject_Role>，表示使用者被授予的主體角色，其屬性subject_role_id表示主體角色的識別。

```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <xsd:schema elementFormDefault="qualified"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
3   <xsd:element name="Users" type="UsersType" />
4   <xsd:complexType name="UsersType">
5     <xsd:sequence>
6       <xsd:element minOccurs="1" maxOccurs="unbounded"
7         name="User" type="UserType" />
8     </xsd:sequence>
9     <xsd:attribute name="xmlDoc" type="xsd:string" />
10  </xsd:complexType>
11  <xsd:complexType name="UserType">
12    <xsd:sequence>
13      <xsd:element minOccurs="0" maxOccurs="unbounded"
14        name="Subject_Role" type="Subject_RoleType" />
15    </xsd:sequence>
16  </xsd:complexType>
17  <xsd:complexType name="Subject_RoleType">
18    <xsd:attribute name="subject_role_id" type="xsd:string" />
19  </xsd:complexType>
20 </xsd:schema>
21

```

圖9：Credential.xsd

使用者資訊XML檔案如圖10所示，行6-行8表示使用者“Tom Ford”的帳號資訊與被授予的主體角色，其主體角色id=“tvjLOqOLj7”所索引到SubjectRoles.xml檔中為“資料庫管理系統的課程編輯者”。

```

1 <?xml version="1.0" encoding="UTF-8" ?>
2 <Users xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
3   xsi:noNamespaceSchemaLocation="Credential.xsd" xmlDoc="SubjectRoles.xml">
4   <User id="Joy" passwd="3iPvs" username="Joy Brel">
5     <Subject_Role subject_role_id="gK9i2bx1ul" />
6   </User>
7   <User id="Tom" passwd="Ti2ix" username="Tom Ford">
8     <Subject_Role subject_role_id="tvjLOqOLj7" />
9   </User>
10  <User id="Bob" passwd="12763" username="Bob Levy">
11    <Subject_Role subject_role_id="7iehIAELuy" />
12  </User>
13  <User id="May" passwd="1xRet" username="May Tumes">
14    <Subject_Role subject_role_id="QPsIbMtfIF" />
15    <Subject_Role subject_role_id="NuihhhXLk8" />
16  </User>
17  <User id="John" passwd="TTvw8" username="John Willen">
18    <Subject_Role subject_role_id="PIpdUgYLoB" />
19    :
20    :
21  </User>
22    :
23    :
24  </Users>
25

```

圖10：Credential.xml

四、主體角色權限

當課務主任制定課程時，系統除了自動產生相對應的主體角色外，亦會產生主體角色相關的存取權限；在預設的情況下，課程編輯者即可依據所被指派的課程，對課程活動樹做閱讀(Read)、新增(Write)、修改(Modify)與刪除>Delete)的權限，而課程中的授課教師、助教與學生均擁有對課程活動樹閱讀的權限。由於每個主體角色對系統中的學習資源擁有不同的存取權限，因此在本文中對主體角色的存取控制策略定義如下：

Policy_Spec:={subject role, action, effect, resource, propagation}

每個Policy_Spec集合表示一項存取控制策略，其中subject role表示此權限為某一特定主體角色才允許執行，action表示執行權限，effect表示對權限的狀態，resource表示存取路徑，propagation表示授權的存取路徑中是否允許往下遞延；由於XML具有階層的特性，上階元素可經由遞迴的方式存取下階元素的資料，因此將resource與propagation結合，直接授權父節點的存取權限，經由遞迴的方式，其效用等同於一一授權子節點的存取權限，如此一來不僅可達到授權，亦可有效的減少存取控制策略數量。

主體角色權限Schema如圖11所示，根元素<Subject_Policies>為主體角色權限集合，子元素<Policy>表示對某一特定檔案的存取策略，其屬性xmldoc為屬性resource所索引的XML檔，元素<Policy>可擁有0至多個子節點<Policy_Spec>，其屬性subject_role_id表示授權的主體角色，屬性action則表示存取方式，其值可為Read、Write、Modify或Delete，屬性effect則表示此權限的狀態，其值可為Grant、Deny或Replace，其中Grant表示允許存取授權路徑，Deny則表示不允許，屬性resource則表示授權的存取路徑，而Replace則表示以新的授權路徑“new_resource”取代授權路徑“resource”，因此當屬性effect值為Replace時，則會合併出現屬性new_resource，屬性propagation則表示此存取權限的授權範圍，其範圍為Local或Recursive，其中Local表示存取範圍僅限於節點本身，包含屬性及文字節點，而Recursive則表示目前節點及其子孫節點，並且包含節點屬性及文字節點。

1	<?xml version="1.0" encoding="UTF-8"?>	23	<xsd:simpleType name="ActionType">
2	<xsd:schema elementFormDefault="qualified"	24	<xsd:restriction base="xsd:string">
3	xmlns:xsd="http://www.w3.org/2001/XMLSchema">	25	<xsd:enumeration value="Read"/>
4	<xsd:element name="Subject_Policies" type="Subject_PoliciesType"/>	26	<xsd:enumeration value="Write"/>
5	<xsd:complexType name="Subject_PoliciesType">	27	<xsd:enumeration value="Modify"/>
6	<xsd:sequence>	28	<xsd:enumeration value="Delete"/>
7	<xsd:element minOccurs="1" maxOccurs="1"	29	</xsd:restriction>
8	name="Policy" type="PolicyType"/>	30	</xsd:simpleType>
9	</xsd:sequence>	31	<xsd:simpleType name="EffectType">
10	</xsd:complexType>	32	<xsd:restriction base="xsd:string">
11	<xsd:complexType name="PolicyType">	33	<xsd:enumeration value="Grant"/>
12	<xsd:sequence>	34	<xsd:enumeration value="Deny"/>
13	<xsd:element minOccurs="0" maxOccurs="unbounded"	35	<xsd:enumeration value="Replace"/>
14	name="Policy_Spec" type="Policy_SpecType"/>	36	</xsd:restriction>
15	</xsd:sequence>	37	</xsd:simpleType>
16	<xsd:attribute name="xmldoc" type="xsd:string"/>	38	<xsd:simpleType name="PropagationType">
17	</xsd:complexType>	39	<xsd:restriction base="xsd:string">
18	<xsd:complexType name="Policy_SpecType">	40	<xsd:enumeration value="Local"/>
19	<xsd:attribute name="subject_role_id" type="xsd:string"/>	41	<xsd:enumeration value="Recursive"/>
20	<xsd:attribute name="action" type="ActionType"/>	42	</xsd:restriction>
21	<xsd:attribute name="effect" type="EffectType"/>	43	</xsd:simpleType>
22	<xsd:attribute name="resource" type="xsd:string"/>	44	</xsd:schema>
	<xsd:attribute name="propagation" type="PropagationType"/>		
	<xsd:attribute name="new_resource" type="xsd:string" use="optional"/>		
	</xsd:complexType>		

圖11：SubjectPolicies.xsd

主體角色權限XML檔如圖12所示，行9中表示主體角色“id=tvjLOqOLj7”即資料庫管理系統的課程編輯者，執行新增權限“action=Write”，授權允許“effect=Grant”，存取檔案路徑“resource=/Courses/Course[@id='MIS-970001']/*”，並且可以遞迴此一節點及其子孫節點“propagation=Recursive”；換句話說，此項存取控制策略即表示“主體角色資料庫管理系統的課程編輯者”“允許”“新增”“課程id=‘MIS-970001’的課程活動樹及訂定教學規則”。

```

1 <?xml version="1.0" encoding="UTF-8" ?>
2 <Subject_Policies xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="Policies.xsd">
3   <Policy xmldoc="Courses.xml">
4     <Policy_Spec subject_role_id="gK9i2bx1ul" action="Read" effect="Grant"
      resource="/Courses/Course" propagation="Local"/>
5     <Policy_Spec subject_role_id="gK9i2bx1ul" action="Write" effect="Grant"
      resource="/Courses/Course" propagation="Local"/>
6     <Policy_Spec subject_role_id="gK9i2bx1ul" action="Modify" effect="Grant"
      resource="/Courses/Course" propagation="Local"/>
7     <Policy_Spec subject_role_id="gK9i2bx1ul" action="Delete" effect="Grant"
      resource="/Courses/Course" propagation="Local"/>
8     <Policy_Spec subject_role_id="tvjLOqOLj7" action="Read" effect="Grant"
      resource="/Courses/Course[@id='MIS-970001']" propagation="Recursive"/>
9     <Policy_Spec subject_role_id="tvjLOqOLj7" action="Write" effect="Grant"
      resource="/Courses/Course[@id='MIS-970001']" propagation="Recursive"/>
10    <Policy_Spec subject_role_id="tvjLOqOLj7" action="Modify" effect="Grant"
      resource="/Courses/Course[@id='MIS-970001']" propagation="Recursive"/>
11    <Policy_Spec subject_role_id="tvjLOqOLj7" action="Delete" effect="Grant"
      resource="/Courses/Course[@id='MIS-970001']" propagation="Recursive"/>
12    <Policy_Spec subject_role_id="7ieh1AELuy" action="Read" effect="Grant"
      resource="/Courses/Course[@id='MIS-970001']" propagation="Recursive"/>
13    <Policy_Spec subject_role_id="QPslbMtlfl" action="Read" effect="Grant"
      resource="/Courses/Course[@id='MIS-970001']" propagation="Recursive"/>
14    <Policy_Spec subject_role_id="PlpdUgYLoB" action="Read" effect="Grant"
      resource="/Courses/Course[@id='MIS-970001']" propagation="Recursive"/>
15    <Policy_Spec subject_role_id="hhJysdFuU" action="Read" effect="Grant"
      resource="/Courses/Course[@id='MIS-970001']" propagation="Recursive"/>
16    <Policy_Spec subject_role_id="NuihhXlk8" action="Read" effect="Grant"
      resource="/Courses/Course[@id='MIS-970001']" propagation="Recursive"/>
17    <Policy_Spec subject_role_id="z2B7jUlfse" action="Read" effect="Grant"
      resource="/Courses/Course[@id='MIS-970001']" propagation="Recursive"/>
18    <Policy_Spec subject_role_id="7ieh1AELuy" action="Read" effect="Replace"
      resource="//Evaluation[@id='gaG49IN4GL']" new_resource="//Evaluation[@id='sXOj8-G9fv']"
      propagation="Recursive" />
19    <Policy_Spec subject_role_id="QPslbMtlfl" action="Read" effect="Replace"
      resource="//Evaluation[@id='gaG49IN4GL']" new_resource="//Evaluation[@id='sXOj8-G9fv']"
      propagation="Recursive" />
20    <Policy_Spec subject_role_id="PlpdUgYLoB" action="Read" effect="Replace"
      resource="//Evaluation[@id='gaG49IN4GL']" new_resource="//Evaluation[@id='sXOj8-G9fv']"
      propagation="Recursive" />
21    :
22    :
23    :
24    :
25    :
26    :
27    :
28    :
29    :
30    :
31    :
32    :
33    :
34    :
35    :
36    :
37    :
38    :
39    :
40    :
41    :
42    :
43    :
44    :
45    :
46    :
47    :
48    :
49    :
50    :
51    :
52    :
53    :
54    :
55    :
56    :
57    :
58    :
59    :
60    :
61    :
62    :
63    :
64    :
65    :
66    :
67    :
68    :
69    :
70    :
71    :
72    :
73    :
74    :
75    :
76    :
77    :
78    :
79    :
80    :
81    :
82    :
83    :
84    :
85    :
86    :
87    :
88    :
89    :
90    :
91    :
92    :
93    :
94    :
95    :
96    :
97  </Policy>
98 </Subject_Policies>

```

圖 12：SubjectPolicies.xml

五、委任授權

為了提供教學上的彈性，使教學內容更符合學習者的需求，因此我們允許課程編輯者授權授課教師以“Replace”的方式對課程活動樹中為Evaluation的元素及其教學規則做修改。當課程編輯者委任授權時，系統即會將此份委任授權資訊記錄於委任授權檔中，其委任授權Schema如圖13所示，根元素<Delegations>為委任授權的集合，擁有屬性xmldoc為屬性subject_role_id所索引的XML檔，元素<Delegations>可擁有0至多個子元素<Delegator>，每個<Delegator>元素表示一項委任授權清單中的委任者，其屬性subject_role_id表示委任者的識別，而屬性xmldoc表示元素<Policy_Spec>的屬性path所索引的XML檔；元素<Delegator>擁有1至多個子元素<Policy_Spec>，表示當委任授權發生時，則至少會有一個委任授權的權限產生；而元素<Policy_Spec>架構與主體角色權限<Policy_Spec>相同，特別一提的是，元素<Policy_Spec>中屬性subject_role_id亦表示授任者，而我們允許以Replace的方式授權，因此在屬性action則限定Replace值，屬性effect則限定允許的狀態。而課程編輯者可授予授課教師權限，亦可將委任之權限撤消(Revocation)，當課程編輯者撤消委任權限時，授任者的<Policy_Spec>即會被刪除，若元素<Delegator>無任何子元素，則元素<Delegator>亦會同時被刪除，即表示此項委任授權已被撤消。

當主體角色登入系統時，除了主體角色權限外，系統亦會判斷主體角色是否擁有被委任授權的記錄，因此主體角色可執行的權限則為主體角色權限加上委任授權的權限。

```

1  <?xml version="1.0" encoding="UTF-8"?>
2  <xsd:schema elementFormDefault="qualified"
3    xmlns:xsd="http://www.w3.org/2001/XMLSchema">
4    <xsd:element name="Delegations" type="DelegationsType" />
5    <xsd:complexType name="DelegationsType">
6      <xsd:sequence>
7        <xsd:element minOccurs="0" maxOccurs="unbounded"
8          name="Delegator" type="DelegatorType" />
9      </xsd:sequence>
10     <xsd:attribute name="xmlidoc" type="xsd:string" />
11   </xsd:complexType>
12   <xsd:complexType name="DelegatorType">
13     <xsd:sequence>
14       <xsd:element minOccurs="1" maxOccurs="unbounded"
15         name="Policy_Spec" type="Policy_SpecType" />
16     </xsd:sequence>
17   </xsd:complexType>
18   <xsd:complexType name="Policy_SpecType">
19     <xsd:sequence>
20       <xsd:attribute name="subject_role_id" type="xsd:string" />
21       <xsd:attribute name="action" type="xsd:string" fixed="Replace"/>
22       <xsd:attribute name="effect" type="xsd:string" fixed="Grant"/>
23       <xsd:attribute name="resource" type="xsd:string" />
24       <xsd:attribute name="propagation" type="xsd:string" />
25     </xsd:sequence>
26   </xsd:complexType>
27 </xsd:schema>

```

圖 13：Delegations.xsd

委任授權XML檔如圖14所示，行3-行6為一項委任授權關係，表示資料庫管理系統的課程編輯者授予A班及B班的二位授課教師，允許以Replace的方式，對課程資料庫管理系統活動樹中為Evaluation的章節及其子節點做修改。

```

1  <?xml version="1.0" encoding="UTF-8"?>
2  <Delegations xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
3    xsi:noNamespaceSchemaLocation="Delegations.xsd" xmlidoc="Courses.xml">
4    <Delegator subject_role_id="tvjLOqOLj7">
5      <Policy_Spec subject_role_id="7ieh1AELuy" action="Replace" effect="Grant"
6        resource="/Courses/Course[@id='MIS-970001']/ActivityTree/Evaluation" propagation="Recursive" />
7      <Policy_Spec subject_role_id="hhJys0dfuU" action="Replace" effect="Grant"
8        resource="/Courses/Course[@id='MIS-970001']/ActivityTree/Evaluation" propagation="Recursive" />
9    </Delegator>
10    ...
51 </Delegations>

```

圖 14：Delegations.xml

六、系統課程

課程編輯者需於學習者選修課程前，先將課程活動樹架構建置完成。當學習者進行學習時，系統即會依據活動樹中每個子節點，依序呈現其授體角色所擁有的學習資源，系統課程Schema如圖15所示。在系統課程Schema中，根元素<Courses>為系統課程的集合，具有屬性xmlidoc表示屬性object_role_id所索引的XML檔，其子元素<Course>表示一門課程，當系統內擁有n門課程，即會有n個<Course>元素，具有屬性id為課程的識別，屬性label表示課程名稱；當課程編輯者開始建立課程活動樹時，系統即會在元素<Course>下自動產生子元素<ActivityTree>與<Pedagogical_Rules>。

元素<ActivityTree>表示課程活動樹，其子節點具有順序性，元素<ActivityTree>可擁有0至多個子元素<Lesson>或<Evaluation>，其中元素<Lesson>表示此章節為一般的學習內容，而元素<Evaluation>則表示此章節為學習評估，其屬性id表示此章節的識別，屬性label表示此章節的名稱，而屬性object_role_id則表示此章節所對應的授體角色，因此在此學習者個人化的學習路徑中，即會將授體角色中的學習資源逐一呈現，而在元素

<Evaluation>中可擁有一至多項子元素<PRule>，表示此學習評估所需參照的教學規則，其屬性idref即表示所參照的教學規則識別。元素<Pedagogical_Rules>為此門課程教學規則的集合，可擁有多個子元素<Ped_Rule>，每個元素<Ped_Rule>即表示一項教學規則，其屬性id為規則識別，由於教學規則中，學習者的學習成績需以成績評量區間來判斷學習者是否符合教學規則的條件，因此我們以IMS簡單順序規範(IMS Simple Sequence Specification)中對教學規則成績評量區間設計為例，以屬性greaterorequal與lessthan表示學習評估的成績區間，屬性greaterorequal表示評估成績需大於或等於，而屬性lessthan則表示評估成績需小於，因此必需將此二項屬性合併判斷，即學習者的評估成績需介於此成績區間內，才符合教學規則的條件，允許存取授體角色，屬性object_role_id即表示此教學規則所屬存取的授體角色。

1	<?xml version="1.0" encoding="UTF-8"?>	24	<xsd:complexType name="LessonType">
2	<xsd:schema elementFormDefault="qualified"	25	<xsd:attribute name="id" type="xsd:string" />
3	xmlns:xsd="http://www.w3.org/2001/XMLSchema">	26	<xsd:attribute name="label" type="xsd:string" />
4	<xsd:element name="Courses" type="CoursesType" />	27	<xsd:attribute name="object_role_id" type="xsd:string" />
5	<xsd:complexType name="CoursesType">	28	</xsd:complexType>
6	<xsd:sequence>	29	<xsd:complexType name="EvaluationType">
7	<xsd:element minOccurs="0" maxOccurs="unbounded"	30	<xsd:sequence>
8	name="Course" type="CourseType" />	31	<xsd:element minOccurs="0" maxOccurs="unbounded"
9	</xsd:sequence>	32	name="PRule" type="PRuleType" />
10	<xsd:attribute name="xml:doc" type="xsd:string" />	33	</xsd:sequence>
11	</xsd:complexType>	34	<xsd:attribute name="id" type="xsd:string" />
12	<xsd:complexType name="CourseType">	35	<xsd:attribute name="label" type="xsd:string" />
13	<xsd:sequence>	36	<xsd:attribute name="object_role_id" type="xsd:string" />
14	<xsd:element minOccurs="1" maxOccurs="1"	37	</xsd:complexType>
15	name="ActivityTree" type="ActivityTreeType" />	38	<xsd:complexType name="PRuleType">
16	<xsd:element minOccurs="1" maxOccurs="1"	39	<xsd:attribute name="idref" type="xsd:string" />
17	name="Pedagogical_Rules" type="Pedagogical_RulesType" />	40	</xsd:complexType>
18	</xsd:sequence>	41	<xsd:complexType name="Pedagogical_RulesType">
19	<xsd:attribute name="id" type="xsd:string" />	42	<xsd:sequence>
20	<xsd:attribute name="label" type="xsd:string" />	43	<xsd:element minOccurs="0" maxOccurs="unbounded"
21	</xsd:complexType>	44	name="Ped_Rule" type="Ped_RuleType" />
22	<xsd:complexType name="ActivityTreeType">	45	</xsd:sequence>
23	<xsd:sequence>	46	</xsd:complexType>
	<xsd:element minOccurs="0" maxOccurs="unbounded"	47	<xsd:complexType name="Ped_RuleType">
	name="Lesson" type="LessonType" />	48	<xsd:attribute name="id" type="xsd:string" />
	<xsd:element minOccurs="0" maxOccurs="unbounded"	49	<xsd:attribute name="greaterorequal" type="xsd:int" />
	name="Evaluation" type="EvaluationType" />	50	<xsd:attribute name="lessthan" type="xsd:int" />
	</xsd:sequence>	51	<xsd:attribute name="object_role_id" type="xsd:string" />
	</xsd:complexType>		</xsd:complexType>
			</xsd:schema>

圖15：Courses.xsd

系統課程XML檔如圖16所示，行7-行10中學習者經由學習評估“id=gaG491N4GL”後會得到學習評估成績，系統即會藉由學習評估成績來判定學習者是否符合教學規則“id=KIFBxpla9n”或id=“4nIKMosBM”所定訂的成績區間，當學習者成績介於0-30分時，則自動存取授體角色id=“EBSuSc2q0U”，而當成績介於30-60分時，則會將授體角色id=“3Z3SO-Gc2A”中的學習者源新增至學習者的學習路徑中；而值得一提的是，課程中的課程編輯者或學習者所需管理的角色僅為課程的主體角色，角色的管理並不會受到授體角色的產生或授權而有所影響。

```

1  <?xml version="1.0" encoding="UTF-8" ?>
2  <Courses xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
   xsi:noNamespaceSchemaLocation="Courses.xsd" xml:doc="ObjectRoles.xml">
3    <Course id="MIS-970001" label="Database">
4      <ActivityTree>
5        <Lesson id="2iuFIDK80G" label="Introduction" object_role_id="xn_VTpY7_W" />
6        <Lesson id="z2B7kHpCB2" label="DB model" object_role_id="wQuaJIoCzr" />
7        <Evaluation id="gaG49IN4GL" label="Evaluation-1" object_role_id="soghvMmAzh">
8          <PRule idref="KIFBxpla9n" />
9          <PRule idref="4nIKMosBM" />
10       </Evaluation>
11       ...
51     </ActivityTree>
52     <Pedagogical_Rules>
53       <Ped_Rule id="KIFBxpla9n" greaterorequal="30" lessthan="60" object_role_id="EBSuSc2q0U" />
54       <Ped_Rule id="4nIKMosBM" greaterorequal="0" lessthan="30" object_role_id="3Z3SO-GczA" />
55       ...
68     </Pedagogical_Rules>
69   </Course>
70   <Course id="MIS-970002" label="Artificial Intelligence">
71     ...
125  </Course>
126  ...
389 </Courses>

```

圖 16：Courses.xml

七、授體角色

由於我們將課程中所需閱讀的學習資源指派予授體角色中，因此課程活動樹中的每個子節點與教學規則皆會擁有一個授體角色，每個授體角色可被指派1至多個學習資源，而授體角色中的學習資源並無閱讀的順序性，當學習者在閱讀授體角色中的學習資源時，可自由選擇學習資源來閱讀；舉例來說，當課程編輯者希望學習者由五題課堂練習題中挑選其中三題練習，學習者在閱讀時，不需侷限題目順序，只要做滿三題即可，此時課程編輯者僅需將此五題練習題指派至一個授體角色中，即可達到此目的。

授體角色Schema如圖17所示，根元素<Object_Roles>為系統中授體角色的集合，擁有子元素<Object_Role_Authorization>與<Object_Role_Hierarchy>，其中子元素<Object_Role_Authorization>表示授體角色與被指派的學習資源集合；由於在教學規則中授體角色具有階層的關係，因此我們以子元素<Object_Role_Hierarchy>表示授體角色間階層關係的集合；元素<Object_Role_Authorization>擁有屬性xml:doc表示屬性sco_id所索引的XML檔，其擁有0至多個子元素<Object_Role>，每個<Object_Role>元素表示一個授體角色，其屬性id表示此授體角色的識別；元素<Object_Role>可擁有一至多個子元素<LearningResource>，表示所被指派的學習資源，其屬性label表示此學習資源的描述，屬性sco_id表示此學習資源所參照的學習資源儲存庫中學習元件的識別；元素<Object_Role_Hierarchy>可擁有0至多個授體角色的階層關係，我們以元素<OR_Hierarchy>表示每個授體角色的階層，上階層的授體角色亦會繼承下階層授體角色的學習資源，其屬性idref表示所對應的授體角色的識別。

```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <xsd:schema elementFormDefault="qualified"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
3   <xsd:element name="Object_Roles" type="Object_RolesType" />
4   <xsd:complexType name="Object_RolesType">
5     <xsd:sequence>
6       <xsd:element minOccurs="0" maxOccurs="unbounded"
7         name="Object_Role_Authorizations"
8         type="Object_Role_AuthorizationsType" />
9       <xsd:element minOccurs="0" maxOccurs="unbounded"
10        name="Object_Role_Hierarchy"
11        type="Object_Role_HierarchyType" />
12     </xsd:sequence>
13   </xsd:complexType>
14   <xsd:complexType name="Object_Role_AuthorizationsType">
15     <xsd:sequence>
16       <xsd:element minOccurs="0" maxOccurs="unbounded"
17         name="Object_Role" type="Object_RoleType" />
18     </xsd:sequence>
19   </xsd:complexType>
20   <xsd:complexType name="Object_RoleType">
21     <xsd:sequence>
22       <xsd:element minOccurs="1" maxOccurs="unbounded"
23         name="LearningResource" type="LearningResourceType" />
24     </xsd:sequence>
25   </xsd:complexType>
26   <xsd:complexType name="LearningResourceType">
27     <xsd:sequence>
28       <xsd:element minOccurs="1" maxOccurs="unbounded"
29         name="OR_Hierarchy" type="OR_HierarchyType" />
30     </xsd:sequence>
31   </xsd:complexType>
32   <xsd:complexType name="OR_HierarchyType">
33     <xsd:sequence>
34       <xsd:element minOccurs="1" maxOccurs="unbounded"
35         name="OR_Hierarchy" type="OR_HierarchyType" />
36     </xsd:sequence>
37   </xsd:complexType>
38   <xsd:complexType name="OR_HierarchyType">
39     <xsd:sequence>
40       <xsd:element minOccurs="1" maxOccurs="unbounded"
41         name="OR_Hierarchy" type="OR_HierarchyType" />
42     </xsd:sequence>
43   </xsd:complexType>
44   <xsd:complexType name="OR_HierarchyType">
45     <xsd:sequence>
46       <xsd:element minOccurs="1" maxOccurs="unbounded"
47         name="OR_Hierarchy" type="OR_HierarchyType" />
48     </xsd:sequence>
49   </xsd:complexType>
50   <xsd:complexType name="OR_HierarchyType">
51     <xsd:sequence>
52       <xsd:element minOccurs="1" maxOccurs="unbounded"
53         name="OR_Hierarchy" type="OR_HierarchyType" />
54     </xsd:sequence>
55   </xsd:complexType>
56   <xsd:complexType name="OR_HierarchyType">
57     <xsd:sequence>
58       <xsd:element minOccurs="1" maxOccurs="unbounded"
59         name="OR_Hierarchy" type="OR_HierarchyType" />
60     </xsd:sequence>
61   </xsd:complexType>
62   <xsd:complexType name="OR_HierarchyType">
63     <xsd:sequence>
64       <xsd:element minOccurs="1" maxOccurs="unbounded"
65         name="OR_Hierarchy" type="OR_HierarchyType" />
66     </xsd:sequence>
67   </xsd:complexType>
68   <xsd:complexType name="OR_HierarchyType">
69     <xsd:sequence>
70       <xsd:element minOccurs="1" maxOccurs="unbounded"
71         name="OR_Hierarchy" type="OR_HierarchyType" />
72     </xsd:sequence>
73   </xsd:complexType>
74   <xsd:complexType name="OR_HierarchyType">
75     <xsd:sequence>
76       <xsd:element minOccurs="1" maxOccurs="unbounded"
77         name="OR_Hierarchy" type="OR_HierarchyType" />
78     </xsd:sequence>
79   </xsd:complexType>
80   <xsd:complexType name="OR_HierarchyType">
81     <xsd:sequence>
82       <xsd:element minOccurs="1" maxOccurs="unbounded"
83         name="OR_Hierarchy" type="OR_HierarchyType" />
84     </xsd:sequence>
85   </xsd:complexType>
86   <xsd:complexType name="OR_HierarchyType">
87     <xsd:sequence>
88       <xsd:element minOccurs="1" maxOccurs="unbounded"
89         name="OR_Hierarchy" type="OR_HierarchyType" />
90     </xsd:sequence>
91   </xsd:complexType>
92   <xsd:complexType name="OR_HierarchyType">
93     <xsd:sequence>
94       <xsd:element minOccurs="1" maxOccurs="unbounded"
95         name="OR_Hierarchy" type="OR_HierarchyType" />
96     </xsd:sequence>
97   </xsd:complexType>
98   <xsd:complexType name="OR_HierarchyType">
99     <xsd:sequence>
100      <xsd:element minOccurs="1" maxOccurs="unbounded"
101        name="OR_Hierarchy" type="OR_HierarchyType" />
102    </xsd:sequence>
103  </xsd:complexType>
104 </xsd:schema>

```

圖17：ObjectRoles.xsd

授體角色XML檔如圖18所示，行15-行21為資料庫管理系統中，教學規則“id=KIFBxpla9n”與教學規則“id=4nIKMosBM”所需存取的授體角色，由於此二項授體角色具有階層關係，如行50-行53所示，因此當學習者可閱讀授體角色“id=3Z3SO-GczA”中的學習資源時，亦可閱讀授體角色“id=EBSuSc2q0U”中的學習資源。

```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <Object_Roles xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="ObjectRoles.xsd">
3   <Object_Role id="xn_VTpY7_W">
4     <LearningResource label="DB Introduction" sco_id="cRt8X2MCJc"/>
5   </Object_Role>
6   <Object_Role id="wQuaJIoCzr">
7     <LearningResource label="DB Models" sco_id="chXpNj0mxu"/>
8     <LearningResource label="Exercise-1" sco_id="12x5d780xf"/>
9     <LearningResource label="Exercise-2" sco_id="34V6RH78TJ"/>
10  </Object_Role>
11  <Object_Role id="soghvMmAzh">
12    <LearningResource label="AEv-1" sco_id="7RJwzk-mAt"/>
13  </Object_Role>
14  ...
59 </Object_Role_Authorizations>
60 <Object_Role_Hierarchy>
61   <OR_Hierarchy idref="3Z3SO-GczA">
62     <OR_Hierarchy idref="EBSuSc2q0U" />
63   </OR_Hierarchy>
64   <OR_Hierarchy idref="Rm-JSail9I">
65     <OR_Hierarchy idref="EBSuSc2q0U" />
66   </OR_Hierarchy>
67   ...
117 </Object_Role_Hierarchy>
118 </Object_Roles>

```

圖18：ObjectRoles.xml

八、學習者學習記錄

在SALeS系統中，課務主任制定課程時，系統會依據課程的班級數，自動產生課程學習者學習記錄檔，當課程擁有二個班級時，則會產生二份學習者學習記錄檔，而在同一個班級中，所有學習者的學習記錄即會儲存於檔案中；學習者學習記錄Schema如圖19所示，根元素<LearningRecord>為課程中所有學習者的學習記錄集合，其屬性course_id表示所對應的課程識別，屬性xmldoc表示屬性course_id所索引的XML檔，元素<LearningRecord>擁有0至多個子元素<Student>，當修課人數為n時，則會擁有n個子元素<Student>；元素<Student>即表示學習者的學習記錄，其屬性user_id表示學習者的帳號，元素<Student>可擁有多個子元素<Lesson>或<Evaluation>，當學習者閱讀完畢後，系統即會自動將此節點記錄於學習者的個人學習記錄中，表示已閱讀過的課活動樹節點，其中

<Lesson>表示此節點為一般的學習內容，而<Evaluation>則表示此節點為學習評估，皆擁有屬性id表示課程活動樹節點的識別，當元素為<Evaluation>時，即表示此節點為學習評估，因此即會增加屬性score來記錄學習者的評估成績。

```

1  <?xml version="1.0" encoding="UTF-8"?>
2  <xsd:schema elementFormDefault="qualified"
3    xmlns:xsd="http://www.w3.org/2001/XMLSchema">
4    <xsd:element name="LearningRecord" type="LearningRecordType" />
5    <xsd:complexType name="LearningRecordType">
6      <xsd:sequence>
7        <xsd:element minOccurs="0" maxOccurs="unbounded"
8          name="Student" type="StudentType" />
9        <xsd:sequence>
10         <xsd:attribute name="course_id" type="xsd:string" />
11         <xsd:attribute name="xmlid" type="xsd:string" />
12         <xsd:complexType name="StudentType">
13           <xsd:sequence>
14             <xsd:element minOccurs="0" maxOccurs="unbounded"
15               name="Evaluation" type="EvaluationType" />
16             </xsd:sequence>
17             <xsd:attribute name="user_id" type="xsd:string" />
18             </xsd:complexType>
19             <xsd:complexType name="EvaluationType">
20               <xsd:sequence>
21                 <xsd:attribute name="id" type="xsd:string" />
22                 <xsd:attribute name="score" type="xsd:int" />
23                 </xsd:sequence>
24                 <xsd:complexType name="LessonType">
25                   <xsd:sequence>
26                     <xsd:attribute name="id" type="xsd:string" />
27                     </xsd:sequence>
28                   </xsd:complexType>
29                 </xsd:sequence>
30               </xsd:complexType>
31             </xsd:sequence>
32           </xsd:complexType>
33         </xsd:sequence>
34       </xsd:sequence>
35     </xsd:complexType>
36   </xsd:schema>

```

圖19：LearningRecord.xsd

學習者學習記錄XML檔如圖20所示，行4-行15為學習者John與May於課程“course_id=MIS-970001”的學習記錄，此班級的學生學習記錄則會記錄於檔案名稱為LearningRecord-MIS-970001-1.xml檔案中，系統會由課程活動樹中的第一個子節“id=2iuFIDK80G”點逐一導覽，當John閱讀過後，則此節點即會記錄於John的學習記錄中，如行5所示，當節點為學習評估時，在John完成此評估測驗後，系統會新增屬性score=20來記錄John評估成績，並且藉由評估成績來判斷課程中的教學規則，經由教學規則判斷後，John需補充三份學習資源，因此系統即會呈現學習資源“Hierarch. Data Model”、“Relation Model”及“File-based System”至John的學習路徑中。

```

1  <?xml version="1.0" encoding="UTF-8"?>
2  <LearningRecord xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
3    xsi:noNamespaceSchemaLocation="LearningRecord.xsd" course_id="MIS-97001" xmlid="Courses.xml">
4    <Student user_id="John">
5      <Lesson id="2iuFIDK80G">
6        <Lesson id="z2B7kHpCB2">
7          <Evaluation id="gaG49IN4GL" score="20"/>
8        </Lesson>
9      </Student>
10     <Student user_id="Lisa">
11       <Lesson id="2iuFIDK80G">
12         <Lesson id="z2B7kHpCB2">
13           <Evaluation id="gaG49IN4GL" score="50"/>
14         </Lesson>
15       </Student>
16     <Student user_id="May">
17       <Lesson id="2iuFIDK80G">
18         <Lesson id="z2B7kHpCB2">
19           <Evaluation id="gaG49IN4GL" score="80"/>
20         </Lesson>
21       </Student>
22     <Student user_id="pCDIYIoAhE">
23       <Lesson id="5DFbxbf2kO">
24         <Lesson id="z2B7kHpCB2">
25           <Evaluation id="gaG49IN4GL" score="100"/>
26         </Lesson>
27       </Student>
28     </LearningRecord>

```

圖20：LearningRecord-MIS-970001-1.xml

伍、系統雛型與實驗

本研究結合了Sun's Java Standard Edition 1.5 及Apache's Tomcat 技術與數位學習通用標準SCORM，以適性化數位學習系統中學習資源的存取控制問題與委任授權需求著手，建置一套適性化數位學習系統所需之存取控制機制，我們稱為SALeS (Scorm-based Adaptive e-Learning System)，以下將詳細說明我們的系統實作。

為了達到適性化系統中學習資源的存取控制，使用者在使用系統時，需經由登入之程序，選取所需擔任的主體角色，系統即會依據不同的主體角色呈現不同的權責。範例中，使用者Joy擁有課務主任的角色，因此具有制定系統課程的權限，圖21即為課務主任建置課程的畫面，課程資料庫管理系統擁有二個班級，當課程建置完成時，系統即會自動產生課程相關的主體角色，如圖8中行5-行17所示，並且建立主體角色權限，如圖12中行8-行17所示，且在系統課程檔中新增元素<Course>，如圖16中行3所示，最後建立課程班級的學習者學習記錄檔，分別為LearningRecord-MIS-970001-1.xml與LearningRecord-MIS-970001-2.xml二個XML檔案。

圖21：Director - Create Course

而在此範例中使用Tom擔任課程資料庫管理系統的課程編輯者，當使用者Tom以“資料庫管理系統課程編輯者”的身份登入後，即可開始編輯課程活動樹，圖22為課程編輯者的課程活動樹編輯畫面，每個章節後即呈現授體角色所擁有的學習資源。

圖22：Manage course activity tree

當課程編輯者建立學習評估與教學規則時，需先指定學習評估內容，當學習評估建立完成後，即需建置教學規則並設定學習評估成績區間，而系統會同時會產生授體角色，並將學習資源指派至授體色中。

然而，由於在適性化教學中，學習評估低於30分的學習者還需閱讀成績區間介於30~60的補充教材，因此在教學規則的設計上，授體角色擁有階層關係，如圖23中，評估成績介於0-30分的學習者，亦需補充30-60分授體角色中的學習資源。

圖23：Pedagogic rule implement

此外，課程編輯者亦可授權課程的授課教師，以Replace的方式，修改活動樹中節點為Evaluation的內容，當授權完畢後，其委任授權記錄即如圖14中行3-行6所示。

當授課教師Bob登入後，即可對課程活動樹中<Evaluation>元素進行修改，如圖24所示，範例中Bob修改“Evaluatuion-1”中(1)將學習評估內容由Database Ev-1取代原本的AEv-1；(2)符合成績區間0~30的學習者再補充閱讀學習資源“Database Definition Language”。

圖24：Instructor - Replace Evaluation

系統即會將有修改的<Evaluation>元素及其教學規則<Ped_Rule>進行複製，並且建立為新的<Evaluatuion>元素，其課程XML檔案變化如圖25中行11-行14與行55所示，由於授課教師並沒有修改規則“id=KIFBxpla9n”，因此元素<PRule>會被完整複製下來；特別一提的是，若原本教學規則中的授體角色擁有階層的關係，則系統亦將此階層關係複製給新的授體角色，其授體角色XML檔案變化如圖26中，行22-行28為新增的授體角色，而原本授體角色“id=3Z3SO-GczA”擁有階層關係，因此新產生的授體角色“id=RM-

JSail9I”亦擁有階層關係，如圖26中行64-行66所示；而在主體角色權限XML檔中，亦會產生存取權限為Replace的存取策略，如圖12中行18-行20所示。

```

1  <?xml version="1.0" encoding="UTF-8" ?>
2  <Courses xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
   xsi:noNamespaceSchemaLocation="Courses.xsd" xmdoc="ObjectRoles.xml">
3    <Course id="MIS-970001" label="Database">
4      <ActivityTree>
5        <Lesson id="2iuFIDK80G" label="Introduction" object_role_id="xn_VTpY7_W" />
6        <Lesson id="z2B7kHpCB2" label="DB model" object_role_id="wQuaJIoCzr" />
7        <Evaluation id="gaG49IN4GL" label="Evaluation-1" object_role_id="soghvMmAzh">
8          <PRule idref="KIFBxpIa9n" />
9          <PRule idref="4nIKMosBM" />
10         </Evaluation>
11         <Evaluation id="sXOj8-G9fv" label="Database Ev-1" object_role_id="mhttkfkMZY">
12           <PRule idref="KIFBxpIa9n" />
13           <PRule idref="2S0G_gHBjx" />
14         </Evaluation>
15         .
16         .
17         .
18         .
19         .
20         .
21         .
22         .
23         .
24         .
25         .
26         .
27         .
28         .
29         .
30         .
31         .
32         .
33         .
34         .
35         .
36         .
37         .
38         .
39         .
40         .
41         .
42         .
43         .
44         .
45         .
46         .
47         .
48         .
49         .
50         .
51       </ActivityTree>
52       <Pedagogical_Rules>
53         <Ped_Rule id="KIFBxpIa9n" greaterorequal="30" lessthan="60" object_role_id="EBSuSc2q0U" />
54         <Ped_Rule id="4nIKMosBM" greaterorequal="0" lessthan="30" object_role_id="3Z3SO-GezA" />
55         <Ped_Rule id="2S0G_gHBjx" greaterorequal="0" lessthan="30" object_role_id="Rm-JSail9I" />
56         .
57         .
58         .
59       </Pedagogical_Rules>
60     </Course>
61     .
62     .
63     .
64     .
65     .
66     .
67     .
68     .
69     .
70     .
71     .
72     .
73     .
74     .
75     .
76     .
77     .
78     .
79     .
80     .
81     .
82     .
83     .
84     .
85     .
86     .
87     .
88     .
89     .
90     .
91     .
92     .
93     .
94     .
95     .
96     .
97     .
98     .
99     .
100    </Courses>

```

圖25 : Courses.xml - Replace Evaluation

```

1  <?xml version="1.0" encoding="UTF-8" ?>
2  <Object_Roles xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
   xsi:noNamespaceSchemaLocation="ObjectRoles.xsd">
3    <Object_Role_Authorization xmdoc="LearningResources.xml">
4      <Object_Role id="xn_VTpY7_W">
5        <LearningResource label="DB Introduction" sco_id="cRt8X2MCJc"/>
6      </Object_Role>
7      <Object_Role id="wQuaJIoCzr">
8        <LearningResource label="DB Models" sco_id="chXpNj0mxu"/>
9        <LearningResource label="Exercise-1" sco_id="12x5d780xf"/>
10       <LearningResource label="Exercise-2" sco_id="34V6RH78TJ"/>
11     </Object_Role>
12     <Object_Role id="soghvMmAzh">
13       <LearningResource label="AEv-1" sco_id="7RJwzk-mAt"/>
14     </Object_Role>
15     <Object_Role id="EBSuSc2q0U">
16       <LearningResource label="Hierarch. Data Model" sco_id="DuMweSYCuY"/>
17       <LearningResource label="Relation Model" sco_id="PImJBhy6BD"/>
18     </Object_Role>
19     <Object_Role id="3Z3SO-GezA">
20       <LearningResource label="File-based Systems" sco_id="cRt8X2MCDb"/>
21     </Object_Role>
22     <Object_Role id="Rm-JSail9I">
23       <LearningResource label="File-based Systems" sco_id="cRt8X2MCDb"/>
24       <LearningResource label="Database Definition Language" sco_id="cRt8X2MCDb"/>
25     </Object_Role>
26     <Object_Role id="mhttkfkMZY">
27       <LearningResource label="Database Ev-1" sco_id="HW-psQ-CCo"/>
28     </Object_Role>
29     .
30     .
31     .
32     .
33     .
34     .
35     .
36     .
37     .
38     .
39     .
40     .
41     .
42     .
43     .
44     .
45     .
46     .
47     .
48     .
49     .
50     .
51     .
52     .
53     .
54     .
55     .
56     .
57     .
58     .
59   </Object_Role_Authorization>
60   <Object_Role_Hierarchy>
61     <OR_Hierarchy idref="3Z3SO-GezA">
62       <OR_Hierarchy idref="EBSuSc2q0U" />
63     </OR_Hierarchy>
64     <OR_Hierarchy idref="Rm-JSail9I">
65       <OR_Hierarchy idref="EBSuSc2q0U" />
66   </OR_Hierarchy>
67   .
68   .
69   .
70   .
71   .
72   .
73   .
74   .
75   .
76   .
77   .
78   .
79   .
80   .
81   .
82   .
83   .
84   .
85   .
86   .
87   .
88   .
89   .
90   .
91   .
92   .
93   .
94   .
95   .
96   .
97   .
98   .
99   .
100  </Object_Role_Hierarchy>
101  </Object_Roles>

```

圖26 : ObjectRoles.xml - Replace ObjectRole

當學習者John登入系統後，即可開始閱讀課程內容，系統即會結合主體角色的存取權限與課程活動樹中的子節點，呈現適性化的學習路徑，圖27為學習者John的課程閱讀畫面，因學習者John在學習評估Evaluation-1中得到成績20分，因此需閱讀四份補充的學習資源，而John即需依照學習路徑完成課程內容的學習，直至課程結束。

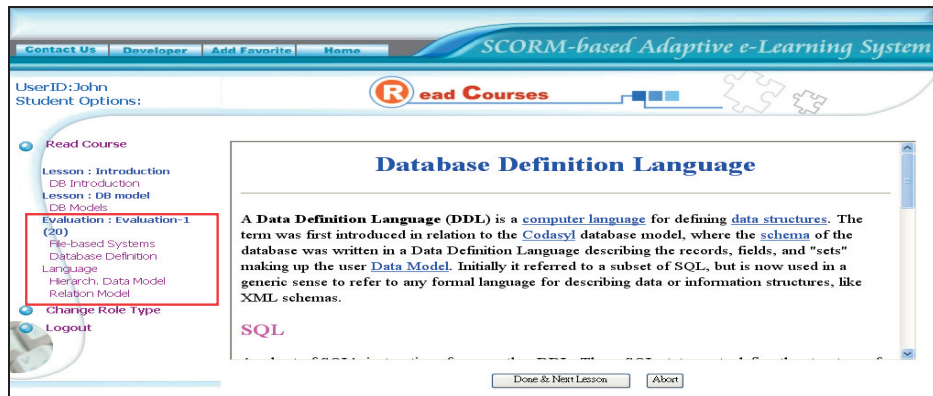


圖27：Student John - Learning Sequence

實驗結果：我們以國立中興大學資訊管理學系的程式設計課程的第一次小考為模擬對象。該課程總共有75位學生，其中大四學生三位、大三學生一位、大二學生22位、其餘為大一學生。該課程的第一次小考範圍包含物件導向的概念以及基本資料型態，其中基本資料型態的介紹可以經由另一門課程--計算機概念--的“整數表示法”、“浮點數表示法”、以及“ASCII碼與萬國碼”的學習得到幫助，因此在系統中，我們定義出兩個授權角色 $or_1 = \{ \text{“整數表示法”, “浮點數表示法”} \}$ 以及 $or_2 = \{ \text{“ASCII碼與萬國碼”} \}$ ， or_2 的階層高於 or_1 ；另外，我們只需要定義兩項授權規則：若學習者的成績區間為 $0 \leq \text{成績} < 20$ 分，則允許其存取 or_2 ；以及若學習者的成績區間為 $21 \leq \text{成績} < 40$ 分，則允許其存取 or_1 。該小考總分為70分，低於20分的學生共有21人，低於40分（不包含低於20分）的學生共有36人。若以傳統Credential模式來開發系統，總需要225 (75×3)項授權；若以RBAC模式進行，則需3項授權。

陸、結論與建議

綜合上述設計概念，本研究具有下列特色：第一，藉由教學規則與存取控制結合，形成Rule-based存取關係，動態授權學習者學習資源的存取權限；第二，利用授體角色做為教學規則與補充學習資源間的媒介，不僅能有效的達到學習資源的存取控制，在規則的彈性與數量上更大幅的改善；第三，以XML檔案架構呈現系統資訊，在學習資源的存取授權上以XPath的方式描述，因此可精細(Fine-grained)的存取系統中管理與追蹤的最小元件-SCO。

本研究提出SALeS模式來改善目前數位學習系統中學習者適性化學習路徑導覽對於學習資源的存取控制機制不足之處，提供適性化數位學習系統一個有效的存取控制機制，利用存取控制模式中，使用者和主體角色配合教學規則中授體角色和學習資源間的存取架構，提供一個符合適性化需求的存取控制機制與彈性的教學規則設計方式，讓教材編輯者與授課教師在課程活動樹的管理更有效率，有效的解決課程架構與學習資源間一對一的鏈結關係，減少教學規則設定的複雜度，而學生亦能擁有適性化的學習路徑與動態的學習資源存取權限。在未來工作上，對於學習資源的搜尋我們希望能提供更有效率的搜尋方式，例如以語意(Semantic)的方式搜尋學習資源；由於我們目前在教學規則的制定上，採用較簡單的規則描述方式，因此在未來希望也能夠利用非量化資料做為學習評估橫量的依據，並且在規則的階層繼承上，希望能夠達到部份規則中，特定部份的學習資源繼承方式，使得系統更趨於完整。

參考文獻

1. Al-Kahtani, M. A., and Sandhu, R. "Rule-based RBAC with Negative Authorization," *Proceedings of Computer Security Applications*, 2004, pp. 405-415.
2. Brusilovsky, P., and Peylo, C. "Adaptive and Intelligent Web-based Educational Systems," *International Journal of Artificial Intelligence in Education*, 2003, pp. 156-169.
3. Chen, C. M., Liu, C. Y., and Chang, M. H. "Personalized Curriculum Sequencing Utilizing Modified Item Response Theory for Web-based Instruction," *Expert Systems with Applications* (30:2), 2006, pp. 378-396.
4. Ferraiolo, D. F., Sandhu, R., Gavrila, S., Kuhn, R. D., and Chandramouli, R. "Proposed NIST Standard for Role-Based Access Control," *ACM Transactions on Information and System Security* (4:3), 2001, pp. 224-274.
5. Fundulaki, I., and Marx, M. "Specifying Access Control Policies for XML Documents with XPath," *Proceedings of the ninth ACM Symposium on Access Control Models and Technologies*, 2004, pp. 61-69.
6. Hatzilygeroudis, I., Giannoulis, C., and Koutsojannis, C. "Combining Expert Systems and Adaptive Hypermedia Technologies in a Web Based Educational System," *Proceedings of Advanced Learning Technologies*, 2005, pp. 249-253.
7. Khan, B. H. "People, Process and Product Continuum in e-Learning: The e-Learning P3 Model," *Educational Technology* (44:5), 2004, pp. 33-40.
8. Moyer, M. J., and Ahamad, M. "Generalized Role-Based Access Control," *International Conference on Distributed Computing Systems*, 2001, pp. 391-398.
9. Murata, M., Tozawa, A., Kudo, M., and Hada, S. "XML Access Control Using Static Analysis," *ACM TISSEC* (3), 2006, pp. 292-324.

10. Sandhu, R. and Bhamidipati, V. “An Oracle Implementation of the PAR97 Model for Permission-Role Assignment,” *Proceedings of the 3rd ACM Workshop on Role-Based Access Control*, 1998, pp. 13-21.