

應用知識本體於網路個人化學習系統之架構設計

Using Ontology to Design the Framework of Personalized Web Learning Systems

石旭原(Shiu-Yuan Shr) 楊東麟(Don-Lin Yang) 趙銘(Ming Chao)

逢甲大學資訊工程研究所

407 臺中市郵政信箱 25-119 號

stone23@ms10.hinet.net dlyang@fcu.edu.tw mingchao@fcu.edu.tw

摘要

本研究主要是將知識本體(ontology)的觀念應用於線上學習系統。除了對教材和題庫引入知識本體的概念，以達到再使用(reuse)的目的外；更透過 Java 監控程式來記錄我們欲獲得之學習者學習歷程，並將這些歷程使用結構化的知識本體語言(ontology language)(如：XML 或 RDF)來加以表示。以此知識表示法，當作語意網頁(semantic web)的基礎，以便未來在利用網頁使用探勘(web usage mining)技術探討學習者行為模式時能夠輕易的搜尋使用，進而提供個人化的(personalized)線上學習環境。

關鍵字：知識本體，再使用，知識本體語言，語意網頁，個人化。

Abstract

Ontology is applied to web learning systems in our work. Besides using ontology to describe our teaching materials and examinations for reuse, we also used Java monitor program to record the student learning profile and stored these profiles in structured ontology language. Based on this knowledge representation, we can employ as a foundation to build semantic web. In the future, we can determine the user's pattern by using web usage mining technique that based on this architecture, and we also can provide personalized web environment.

keywords : ontology, reuse, ontology language, semantic web, personalization.

1. 前言

網際網路的蓬勃發展確實帶來了許多革命性的應用，例如：炒得火熱的電子商務，不知吸引多少資金、人力投入其中，也造就了不少名就一時的網路新貴；只不過，其最終的獲益結果卻令人相當的不滿意。正當大家懷疑網路是否真的開始走下坡時，另外一項熱門的應用似乎正快速的影響我們的生活－網路學習。[22]

目前的網路學習大部分指的是透過一般瀏覽器(如：MSIE)來學習線上教材。透過開放性和一致性的介面環境，使用者可以輕鬆的點選畫面上其所欲學習的單元／教材來進行學習。網路學習最大的優點在於它提供學習者充分的自主性。學習者可以透過超本文(hypertext)的技術輕易的、立即的鏈結至其所要到達的網頁。也可以根據個人的喜好、能力、和學習的習慣來自我調整學習速度以達到部分個人化的功能。

然而，目前大部分的網路學習系統普遍還存在著許多問題[2][6]：

- (1) 過度的自主性和開放性衍生出了認知超載(cognitive overhead)和迷失方向(disorientation)的問題[4]。目前大部分的網路學習系統充其量只能稱得上是資訊提供站。缺乏適當的導引機制，容易讓學習者在超媒體空間(hyperspace)中產生認知超載和迷失方向等問題，降低學習成效。
- (2) 大部分的系統無法觀察、記錄、分析使用者的特質。如此一來，系統便無法對於個別的使用者提供差異性的、個人化的服務。
- (3) 系統缺乏學習、即時修正的能力。由於無法適度維護線上使用者的相關特性，因此無法對不適當的教學和導引來進行修正。我們冀望藉由知識本體(ontology)的觀念來提供老師和學生個人化的環境。一方面對教材作

完整描述，例如：此教材為何種教學法、檔案格式、教材和教材間的相依性、適合哪一種類的學生等相關資訊。另一方面提供適度的導引機制，更可以藉由此階層性架構清楚的觀察到使用者的瀏覽路徑，在考試機制中我們亦加入此概念。此外，也記錄老師瀏覽的習慣來判斷老師的偏好，例如：老師習慣以何種資訊來觀察學生(學生的考試資訊或學習歷程)。系統利用網頁探勘(web mining)技術記錄使用者的相關資訊(如：學習者停留在某頁教材的時間、學習者目前的學習進度、老師較偏好瀏覽哪些教材、老師習慣以哪些資訊來觀察學生等)，並針對所記錄的資料作前處理的動作使之成為較有結構的資料，進而分析使用者的特性，以期能針對不同的使用者特性(如：基本資料、學習進度、測驗情況等)來作一個別化的導覽。期望透過這些技術的整合來克服網路學習的種種缺點，使老師能有效的指導學生，增加學習者的學習效果。

本文的架構為：第二節就相關研究領域進行文獻探討；第三節則是敘述所提出的系統架構和其目的；第四節說明系統的建構方法，最後第五節則對本文做一簡單的結論並說明未來研究之方向。

2. 相關文獻

2.1 知識本體語言(Ontology Language)引導網路學習

本節將簡單的敘述傳統 HTML 的缺點，然後進一步的說明為何使用 XML 來取代 HTML。最後舉實際的例子來說明如何使用知識本體語言來改進 XML 語意不足的缺點。

2.1.1 HTML

傳統的 HTML，其簡單易學的標籤語言對於促進網際網路的蓬勃發展確實功不可沒。然而，儘管 HTML 在人機介面的建構上提供了相當強大的功能，可是相對的，卻非常不利於機器之間的溝通，對於未來下一代網路的語意網頁(semantic web)，著實也產生了許多困難。所謂的語意網頁和傳統網頁的最大不同點在於，傳統網頁中對於資料本身的描述性和資料間的關係並未加描述；然而，語意網頁卻有著豐富的註釋(包含同義詞之間的映對)，使得不管使用者本身或軟體代理人(software agent)都可以清楚地了解文件中的語意。只是目前最普及的網頁標籤語言 HTML 不足以用來建構語意網頁。一般而言，HTML 存在著以下的限制[4][7][8][11]：

(1) 缺乏結構性(structure)：簡單的 HTML 缺乏嚴謹的結構性。HTML 除了規範簡單的 <head>和 <body>的架構外，大部分的標籤(tag)間並無主從關係。沒有清楚的定義標

籤出現的順序、次數；也沒有嚴謹的規定 open tag 和 close tag。

(2) 缺乏延伸性(extensibility)：HTML 的標籤並沒有可延伸性。HTML 是一種固定用途的標記語言。無法根據不同領域的應用需求，讓文件使用者訂定自己需要的標籤，此種無法延伸的缺點也侷限了其發展。

(3) 缺乏語意性(semantic)：HTML 的標籤並無法真實的反映出夾在標籤間的文件內容。這一項限制對於邁向語意網頁是一種嚴重的障礙。雖然使用者可以很容易的透過瀏覽器來閱讀 HTML 所編輯文件的內容，但是對於軟體代理人而言，卻無法從 HTML 內容得知文件的相關資訊，所以就無法對文件作有系統的分類整理或自動化的處理使用者的相關要求。

由於 HTML 的種種限制，使得 HTML 只能稱得上是一種資料的"展示"語言，無法對於網路教材之本身和教材間的相依性作一清楚的描述，也因此無法對學習者作出適度的引導機制。

2.1.2 XML

HTML 的限制，引發許多研究改以使用 XML 來取代 HTML[4][7][11][15][23]。這樣的改變確實也達到了相當程度的成功。對於 XML 而言，上述 HTML 的各項限制都可以迎刃而解。不僅結構嚴謹，對於標籤的定義一點也不馬虎。最重要的是，XML 是一種詮釋語言(meta-language)，其目的主要是用來描述文件的內容，設計者可以使用自訂的標籤來定義文件資料所代表的意義，例如：教材中使用到的課程名稱就可以定為<course>、使用何種教學法<teach-type>、教材和教材間的前後相依性<dependency>。其標籤也就成為了名副其實的詮釋資料(metadata)。

2.1.3 知識本體語言

要實作語意網頁，須以知識本體為基礎。而知識本體需以知識本體語言來協助建構。所謂的知識本體，簡單的說就是一種分類法。其定義如下："一個知識本體是一個正式的(formal)與明確的(explicit)規格，說明大家都能共同接受的概念(shared conceptualisation)"[17][19]。其中所謂的"正式的(formal)"，指的是其具有標準的格式，使得軟體代理人能夠輕易讀取其內容；"明確的(explicit)"指的是被描述的概念所應有的限制，如：款項，這一概念所輸入的屬性值就必須為一數值(numeric)，而非字串(string)；而"共同接受的(shared)"，其所代表的意思是為大家所共同認同的一個標準，而非限制於某一個人或單位；最後，就是"概念(conceptualisation)"，其意義指的是一般真實世界中某些現象的抽象模型，如：學生、老師、

鉛筆、黑板等較高層次的抽象模型。有了一致的知識本體之後，要如何將之表現出來，進而和別人交換資訊？可以利用知識本體語言來將使用者心中所要呈現的抽象概念表現出來。

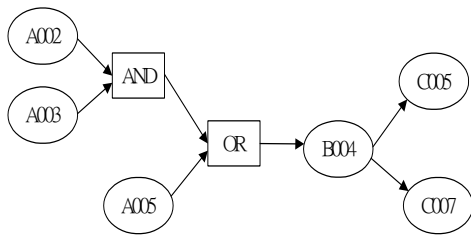
目前網路上有許多的知識本體語言被使用。如：XOL(XML-based ontology-exchange language)、RDF(Resource Description Framework) / RDFS(Resource Description Framework Schema)、或者是 DAML+OIL(DARPA Agent Markup Language + Ontology Inference Language)[17][18]等。廣泛的說，XML也可以算是一個知識本體語言，只不過，其對於標籤間的關係描述算是薄弱了些。

這些語言各有各的特性或專屬領域。如XOL專門應用在描述生物資訊，並簡單的定義一些相關的基本元素(primitive)，如：class hierarchy、slot等。RDF本身並沒有定義標籤的基本元素，而是提供一個標準模式(model)和相關語法來描述標籤間的關係(relationship)，所構成的基本元素則是透過RDFS來定義。

至於DAML+OIL則是較新的知識本體語言。其定義自己的語法，並且提供非常豐富的文法來供使用者使用。其具備的優點如下：

- (1) 對於使用者而言，具有高度的直觀性。也就是說當一般使用者檢視其語法建構成的知識本體時，不會有看不懂或模稜兩可的情形。
- (2) 語法本身具有類似框架的架構(frame-like)和物件導向的概念。
- (3) 可以和現存的知識本體語言作相當程度的結合。不僅可以使用XML基本的語法來表現，也可以使用RDF來呈現資料(在使用RDF語法來表現時，存在一表格專門來映對兩語言間的標籤)。不僅如此，其更可以使用RDFS來定義知識本體中相關的基本元素。

以下實際舉一例子來說明如何使用知識本體語言來改進一般傳統XML的缺失，此例是使用DAML+OIL語言。假設我們有一課程，其中單元和單元間的關係如圖一[4]，此圖類似於概念圖：



圖一 教材間相依性之概念圖

圖一是說明在我們學習B004單元之前，系統中建議必須最好先學完A002與A003，或

者是先學完A005；而當學完B004之後，我們可以選擇C005或C007來繼續學習。程式(一)是以傳統的XML來描述，標籤<prerequisite>出現了兩次，而且沒有提供額外的訊息，導致系統不知是要學習完A002、A003與A005之後才繼續學習B004；還是學完A002與A003後、或者是先學完A005，即可學習B004。顯現出XML的不足之處。

然而，若使用DAML+OIL來改良此一缺點，改良過的語法如程式(二)。其中我們可以看出程式(二)能夠清楚的表現出教材的資訊，且使得單元和單元間的相依性不再變得模糊不清。

```

<dependency>
<current>
  <courseunit id="B004" url="B004.asp"/>
</current>
<prerequisite>
  <courseunit id="A002" url="A002.asp"/>
  <courseunit id="A003" url="A003.asp"/>
</prerequisite>
<prerequisite>
  <courseunit id="A005" url="A005.asp"/>
</prerequisite>
<followup id="C005" url="C005.asp"/>
<followup id="C007" url="C007.asp"/>
</dependency>
  
```

程式(一) 描述圖一之XML格式

```

ontology-definitions
slot-def "prerequisite"
slot-def "followup"
slot-def "is"
class-def chapter
class-def currentchapter
  subclass-of chapter
  slot-constraint is
    value-type B004
class-def A002
  subclass-of chapter
class-def A003
  subclass-of chapter
class-def B004
  subclass-of chapter
  slot-constraint followup
    value-type C005 OR C007
  slot-constraint prerequisite
    value-type (A002 AND A003) OR A005
class-def A005
  subclass-of chapter
class-def C005
  subclass-of chapter
class-def C007
  subclass-of chapter
  
```

程式(二) 描述圖一之DAML+OIL語法

2.2 學習理論

2.2.1 概念圖理論

概念圖理論為 Novak 於 1972 年所提出的，主要的基本元素為概念(concept)、概念間的連結線(link)與連結語。此理論主要的目的是要將學習者所學習過的概念和將要學習的概念間之關係表示出來，強調學習者在學習的過程中，必須以既有的概念為基礎，將新的概念建構於舊有知識上，以形成一新的知識結構，達到有意義的學習[6]。

這樣的描述方法恰好和我們所欲達到的語意網頁相類似。透過連結線和連結語，我們可以清楚的表示出整個教材的階層架構和教材間的相依關係。再者，老師藉由概念圖方法安排單元進度時，便可思考將其教學策略一併描繪其中，如針對同一教材安排不同的學習歷程。而學生藉由概念圖，不僅可以對整份教材作一宏觀的檢視，更可以藉由概念圖引導的作用，避免陷入方向迷失、認知超載的迷思中。

2.2.2 凱勒(Keller)的教學模式

由於教學網頁具備超本文(hypertext)的特質，提供使用者開放性、自主性、整合性的資訊環境，在網路的環境下，充份給予使用者自主的資訊搜尋、機動性的資訊鏈結及探索性的學習模式。這樣的一個環境恰好提供了個別化教學的契機。凱勒所提倡的個別化教學系統，做法是將教材分為許多個小單元，而且每個小單元都設有一定的精熟標準，當學習者達到此一單元的精熟標準後，才能繼續往下一單元學習，否則必須重新學習該單元[4]。這樣的學習模式和布魯姆的精熟理論有著異曲同工之妙。兩者皆強調單元和單元間存在相依性，而且學習下一單元前，對前單元必須有相當程度的了解才行。

以上的教學理論，都指出一個良好線上學習系統的設計，測驗的內容和教材的設計安排(導引機制)皆必須是有順序、有條理的。

2.3 網頁探勘(Web Mining)和個人化(Personalization)

隨著網際網路的蓬勃發展，上線的網站也如雨後春筍般的冒出，網站間的競爭亦趨白熱化。每個網站無不絞盡腦汁爭取更多的使用者，增加網站更多的商機。然而這樣的做法其實只是在表面上工夫，網站所提供的服務仍是無法滿足使用者的需求。於是，個人化的研究和應用便應運而生了。

個人化，亦有人稱之為一對一，幾乎現存的網站都提及這一熱門的名詞，所謂的個人化，顧名思義就是要提供符合個別使用者需求的服務。然而，網站的管理者無法直接觀察使用者的表情、語氣的變化來了解使用者的喜好，而只能透過觀察使用者在網頁上的行為，進而記錄、分析其行為，來了解使用者。

想要在網頁上提供個人化的服務，現存的資訊技術中，網頁探勘是一相當不錯的選擇。目前，網頁探勘技術分為三種：網頁內容探勘(web content mining)、網頁結構探勘(web structure mining)、以及網頁使用探勘(web usage mining)。

網頁內容探勘，就是對於網頁上的文件內容(如：圖形、影像、文字或網頁內部的超連結結構)作資料探勘。而網頁結構探勘則是對於網頁上不同的連結作資料探勘，其和網頁內容探勘最大的相異處在於：網頁內容探勘強調的是對文件本身內部的超連結作資料探勘；而網頁結構探勘則是對於網頁上不同文件的超連結結構作資料探勘的動作。至於網頁使用探勘技術則是在探討如何正確的記錄使用者在網頁上的瀏覽路徑，來作行為分析的資料探勘[3][9][10]。因我們較關心網頁使用的狀況，故以下就網頁使用探勘之步驟分別細述之。經由歸納，其主要分為三個步驟：

- (1) 正確的記錄網路上的使用者瀏覽行為，進而蒐集這些原始資料(raw data)。而資料蒐集方面有分為伺服器端、客戶端、和代理伺服器端三種資料的蒐集來源。
- (2) 當有了一些原始資料後，接下來的動作便是將這些原始資料轉換成較有意義的交易(transaction)記錄。由於資料來源是使用許多不同技術自網頁上擷取下來的，因此在做資料探勘之前，必須對這些資料來作處理，如此才可使用資料探勘的方法來加以分析。目前常用於資料前處理的方法較著名的有 Maximum Forward Reference 以及瀏覽者傾向為導向的方法。
- (3) 最後再對這一些較有規則性的記錄作資料探勘。用以探勘(區分)出使用者所代表(屬於)的模式(pattern)。較常使用的方法為關聯式法則來作資料探勘的動作。

在虛擬的網路學習環境中提供老師和學生個人化服務是主要的關心議題，我們希望藉由了解使用者的基本資料並且觀察其線上所作的一連串動作，進而利用網頁使用探勘對資料分析的技術來分析其人格特質，增進對使用者的了解，以提供個人化的服務，達到有效率的管理和學習。

3. 系統目的和架構

不久的將來，在網路上的大部分資訊交流都將透過軟體代理人來處理。試想，當我們需要某一個商品的資訊時，便派遣一個軟體代理人到網路上幫我們帶回所要的相關資訊，如此便利的網路世界，豈不便哉！只可惜，當人們沉浸此美夢時，卻遇到一個非常嚴重的問題—軟體代理人如何在不同的網頁上蒐集我們所

要的資訊，更何況這些資訊是缺乏結構性的。目前人工智慧的發展程度尚不足以解決這一大堆網頁上自然語言的問題，更何況，若網頁上又使用某些腳本(script)語言的話，那就更顯得複雜了。於是，許多組織、研究者就此問題探討出許多不同的解決方法[16][19]，其中一種解決方法就是建構知識本體。將網頁中所有的資訊加以分門別類，並且在每一筆資料前都加上描述資料內容的標籤，如：“石旭原”，我們就在此資料加上<逢甲大學學生>標籤，如此就可以很清楚的知道石旭原是一位逢甲大學學生。以此形式，我們當然可以任意加上許多描述“石旭原”的屬性，如：<年齡>、<出生地>、<學習歷程>、<成績表現>等相關資料。

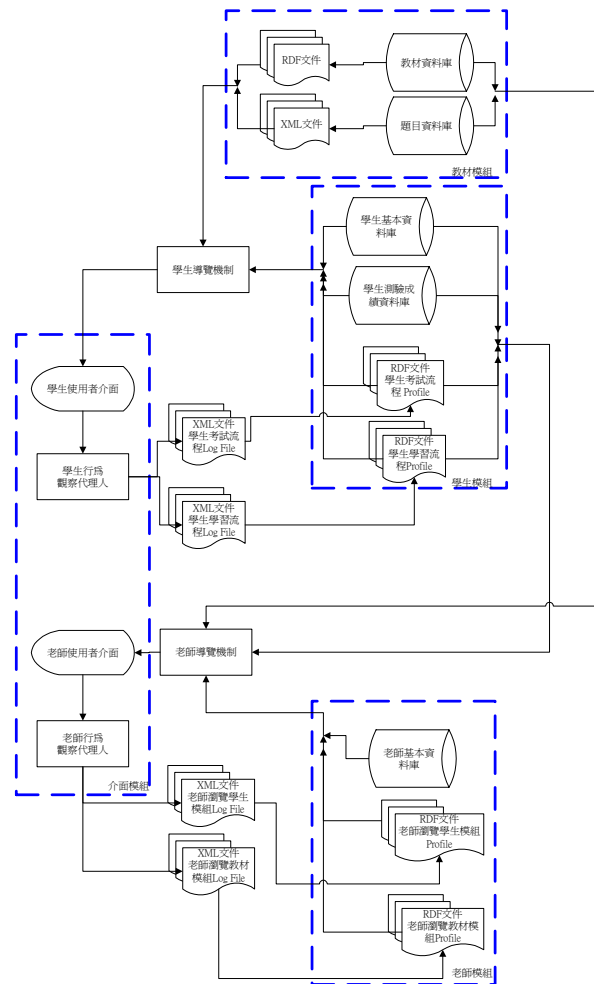
我們的系統就是將上述概念沿用過來，將所有的教材資料、系統資料、學生資料、老師資料等，全部都使用這樣子的表示法。如此一來，我們的資料不僅可以被我們使用，也可以很輕易的被其他使用者所尋獲，達到再使用(reuse)的目的。

若更深一層的探討，我們可以清楚的了解到，目前網路上大部分的資料都是建構在資料庫中的，當我們要去存取資料時，所碰到的第一件事即是資料庫異質性的問題，這使得我們無法輕易的存取到別人的資料；相對地，別人來存取我們的資料時也會遇到相同的問題。為了將降低此問題的影響，我們在處理後端的資料時，採用 XML 和 RDF 一起來管理(有些基本資料是採用傳統的資料庫) [13][20]。會使用 XML 或 RDF 來實作整個系統，主要考量到的原因歸納如下[1][5][13][14][15][20]：

- (1) XML 和 RDF 是 W3C(World Wide Web Consortium)所建立出來的標準，也就是說，全世界大部分的組織和單位都會依此標準來撰寫文件內容，若加上使用相同的知識本體，則彼此間的文件就可透過一致的介面來進行交換。
- (2) 資料間的關係(relationship)。傳統的資料庫並無法精確的描述出資料和資料間的相依性，而純粹使用 XML 來管理資料又缺乏豐富的語法來描述資料間的關係，所以我們利用 RDF 對資料的描述性來彌補這一方面的不足。
- (3) 資料的分享性(data sharing)。網路上現今充斥著許多線上的教學教材，只可惜這些寶貴的教材間彼此卻缺乏交換的標準。原因無他，只在於這些教材和教材間，並無一致描述教材的知識本體存在。舉一個例子來說，當有一門線上資料結構的課存在時，大家即使用共通的標準(XML 或 RDF)來描述其相關的屬性，如：教材所使用的教學法為何、屬於哪一章節、教材內容的關鍵字為何、教材所儲存的檔案格式為何、其所在的 URI 為何等相關資訊。如此一來，當我們派出軟體代理人到網路上幫我們尋找時，因為是使用標準語言來描述教材，所以軟體代理人可以很輕易的幫我們找到所需要的教材，而相對的，我們的教材也可以很輕易的被別人的軟體代理人所找到，達到資料分享的目的。

何、其所在的 URI 為何等相關資訊。如此一來，當我們派出軟體代理人到網路上幫我們尋找時，因為是使用標準語言來描述教材，所以軟體代理人可以很輕易的幫我們找到所需要的教材，而相對的，我們的教材也可以很輕易的被別人的軟體代理人所找到，達到資料分享的目的。

本研究的重點是以知識本體為基礎來提供老師和學生個人化的網路環境。有鑒於在網路上提供個人化的重要性，我們試圖將個人化引進我們的教學系統。系統的架構如圖二，主要包括四個模組：學生模組、老師模組、教材模組以及介面模組。由於系統中最主要的個人化對象為老師和學生，故我們針對這兩個不同的角色提供不一樣的使用者介面和導覽機制。以下就以這兩種不同的角色來敘述我們的系統架構。



圖二 教學系統之系統架構

由學生的角度來看，可以從系統架構圖中看出“學生使用者介面”所欲呈現的畫面，是由“學生導覽機制”所提供的，也就是說，每一

位學生所瀏覽的教材順序、測驗题目的難易度等畫面皆不相同。而系統中的”學生導覽機制”，其背後的知識庫來源主要為”教材模組”和”學生模組”。其中”教材模組”中所保有的資料為”教材資料庫”(如：Flash、Powerpoint、Word 等套裝軟體所開發的教材)和”題目資料庫”，對於這兩大資料庫，我們分別以 RDF 和 XML 文件格式來描繪其中所包含的各項資訊；而”學生模組”當中，我們記錄了”學生基本資料庫”、”學生測驗成績資料庫”、”RDF 文件學生考試流程的輪廓(profile)”和”RDF 文件學生學習流程輪廓”等相關資料。其中”RDF 文件學生考試流程輪廓”和”RDF 文件學生學習流程輪廓”資料來源是根據觀察代理人傳回的”XML 文件學生考試流程日誌檔(Log File)”、”XML 文件學生學習流程日誌檔”所擷取整理出來的。而”學生導覽機制”就是根據”學生模組”中所記錄的學生資料，再結合”教材模組”，來分析每個學生的學習情況和相關的學習特性，提供學生較適性的學習。

另外，從老師的角度來切入的話，我們提供老師來瀏覽(修改)其教材和測驗題是否編排合適的機制。對於老師所關心的學生學習狀況的問題，老師們除了可藉由瀏覽單筆學生資料來判斷某一位學生的學習情況，做出適當的回饋，如：修改某一位學生的導覽機制或透過其他機制來和學生互動(mail、chatroom)；亦可自行下簡單的指令來瀏覽屬性類似的學生群。而我們的系統中也提供老師個人化的服務。一樣是透過”觀察代理人”幫我們在客戶端(Client)記錄老師在系統中的種種行為。系統將根據”RDF 文件老師瀏覽學生模組輪廓”來提供老師在作學生資料瀏覽時個人化的服務；另外，也依照”RDF 文件老師瀏覽教材模組輪廓”來判斷某一位老師較偏好瀏覽哪一份教材和題庫，以減少老師在網路上漫遊的時間。

4. 系統的建構方法

本節我們將依架構圖中的模組分別說明。在教材模組部分，是將所擁有的教材資料以 RDF 文件格式來做一個索引檔，也可說是教材的說明檔；而題目部分，我們則是直接將題目以 XML 文件建置(題型分為是非、單選、複選、填充題)，程式(三)為我們系統的題目範例(是非題)。而在教材部分，最重要的是確保教材有統一的知識本體，並且以階層式的架構來呈現。這裡我們以張系國教授其所彙整出的 GrowingBook(<http://www.cs.pitt.edu/~jung/GrowingBook/HTML/main.html>)為例，其有一份以資料結構課程為主的教材，其中每一章節中，都有定義好的階層架構(每一章節的內容結構都為負責各章節的老師自行定義的)，如圖三

我們只列出其中一章的教材結構。而其所對應的 RDF 格式則描述在程式(四)，此範例只單純的描繪圖三的階層架構，實際上，我們可以加入我們所需要的資訊。

我們系統中的資料儲存不完全使用 XML 和 RDF，原因是為了配合本校的遠距教學計畫，因為本校目前正著手建置遠距教學系統，為了相配合，使得我們有些部分的資料是直接採用原有的資料庫(避免重複的資料產生)。

```
<?xml version="1.0" encoding="Big5"?>
<Question>
  <CourseName>DataStructure</CourseName>
  <Chapter>Tree</Chapter>
  <Description>葉節點(root)為最上層的節點。</Description>
  <Level>易</Level>
  <Type>是非題</Type>
  <Multimedia Question/>
  <Solution>根節點為最上層的節點</Solution>
  <Memo/>
  <Item>
    <ItemDescription>對</ItemDescription>
  </MultimediaItem/>
  </Item>
  <Item>
    <ItemDescription>錯</ItemDescription>
  </MultimediaItem/>
  <ISanswer/>
  </Item>
</Question>
```

程式(三) 使用 XML 編寫的考題範例

```
1 Chapter5
  1.1 Basic Sorting
    1.1.1 Selection Sort
    1.1.2 Insertion Sort
    1.1.3 Bubble Sort
  1.2 Quick Sort
  1.3 Heap Sort
  1.4 Merge Sort
  1.5 Other Sorts
```

圖三 描述教材階層架構的範例

```
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#">
  <rdf:Description ID="Chapter5">
    <rdf:type
      resource="http://www.w3.org/2000/01/rdf-sch
```

```

ema#Class"/>
<rdfs:subClassOf
rdf:resource="http://www.w3.org/2000/01/rdf-
schema#Resource"/>
</rdf:Description>
<rdf:Description ID="Basic Sorting">
<rdf:type
resource="http://www.w3.org/2000/01/rdf-sch
ema#Class"/>
<rdfs:subClassOf rdf:resource="#Chapter5"/>
</rdf:Description>
<rdf:Description ID="Selection Sort">
<rdf:type
resource="http://www.w3.org/2000/01/rdf-sch
ema#Class"/>
<rdfs:subClassOf rdf:resource="#Basic
Sorting"/>
</rdf:Description>
<rdf:Description ID="Insertion Sort">
<rdf:type
resource="http://www.w3.org/2000/01/rdf-sch
ema#Class"/>
<rdfs:subClassOf rdf:resource="#Basic
Sorting"/>
</rdf:Description>
<rdf:Description ID="Bubble Sort">
<rdf:type
resource="http://www.w3.org/2000/01/rdf-sch
ema#Class"/>
<rdfs:subClassOf rdf:resource="#Basic
Sorting"/>
</rdf:Description>
<rdf:Description ID="Quick Sort">
<rdf:type
resource="http://www.w3.org/2000/01/rdf-sch
ema#Class"/>
<rdfs:subClassOf rdf:resource="#Chapter5"/>
</rdf:Description>
<rdf:Description ID="Heap Sort">
<rdf:type
resource="http://www.w3.org/2000/01/rdf-sch
ema#Class"/>
<rdfs:subClassOf rdf:resource="#Chapter5"/>
</rdf:Description>
<rdf:Description ID="Merge Sort">
<rdf:type
resource="http://www.w3.org/2000/01/rdf-sch
ema#Class"/>
<rdfs:subClassOf rdf:resource="#Chapter5"/>
</rdf:Description>
<rdf:Description ID="Other Sorts">
<rdf:type
resource="http://www.w3.org/2000/01/rdf-sch
ema#Class"/>
<rdfs:subClassOf rdf:resource="#Chapter5"/>
</rdf:Description>
</rdf:RDF>

```

程式(四) 使用 RDF 編寫圖三教材階層架構

在學生模組方面，也是為了和本校遠距教學計畫可以做相當程度的融合，所以在學生模組部分，學生的基本資料庫和學生測驗成績資料庫都直接採用本校遠距教學系統中的資料庫；而新建構”RDF 文件學生考試流程輪廓”和”RDF 文件學生學習流程輪廓”。由於這兩個輪廓所建構的方式都是由觀察代理人記錄產生的，建構方法大同小異，所以本文只舉出其中的”RDF 文件學生學習流程輪廓”來作說明。整個輪廓的建構流程如下：

- (1) 當學習者登錄到本系統時，其瀏覽的畫面是”學生導覽機制”所提供的。此時學習者所瀏覽的網頁，其背後都有一個監控程式來記錄我們所想要擷取的學習者學習歷程資料。其做法可以使用 Java Applet 和 Java Script 來達成。
- (2) 當學習者關閉瀏覽器或進入其他的學習畫面時，監控程式會將其所記錄的資料傳回，並將之存入一個 XML 格式的日誌檔 (Log File)，程式(五)為一個學生瀏覽課程時之日誌檔範例，而程式(六)為其 DTD(Document Type Definition)。
- (3) 最後，我們從日誌檔中擷取出我們所要的資訊，並將之儲存為文法結構更豐富的 RDF 文件，我們將之呈現在程式(七)，而整個流程如圖四。

經由以上的三個步驟，我們可以記錄到使用者的偏好資訊，以供後續推理機制使用。

```

<?xml version="1.0"?>
<!--Log File-->
<Log>
  <Logentry>
    <Date><Day>01</Day><Month>08</Month><Year>2001</Year><Time>15:32:20</Time></Date>
    <Material>140.134.26.204/material/datastructure/SelectionSort</Material>
    <Duration>3</Duration>
  </Logentry>
  <Logentry>
    <Date><Day>01</Day><Month>08</Month><Year>2001</Year><Time>15:32:50</Time></Date>
    <Material>140.134.26.204/material/datastructure/InsertSort</Material>
    <Duration>3</Duration>
  </Logentry>
  <Logentry>
    <Date><Day>01</Day><Month>08</Month><Year>2001</Year><Time>15:33:01</Time></Date>
    <Material>140.134.26.204/material/datastructure/BubbleSort</Material>
    <Duration>3</Duration>
  </Logentry>
</Log>

```

```

<Date><Day>01</Day><Month>08</Month><Year>2001</Year><Time>15:33:20</Time></Date>
<Material>140.134.26.204/material/datastructure/QuickSort/</Material>
<Duration>120</Duration>
</Logentry>
</Log>

```

程式(五) 學生瀏覽教材時的日誌檔

```

<?xml version="1.0" encoding="UTF-8"?>
<!ELEMENT Log (Logentry+)>
<!ELEMENT Logentry (Date, Material, Duration)>
<!ELEMENT Date (Day, Month, Year, Time)>
<!ELEMENT Material (#PCDATA)>
<!ELEMENT Duration (#PCDATA)>
<!ELEMENT Day (#PCDATA)>
<!ELEMENT Month (#PCDATA)>
<!ELEMENT Year (#PCDATA)>
<!ELEMENT Time (#PCDATA)>

```

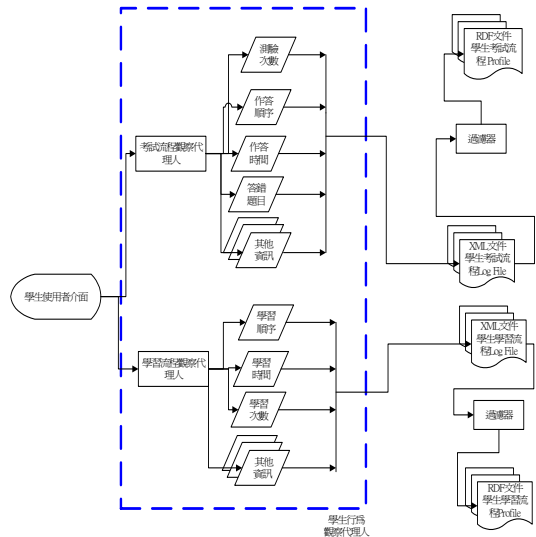
程式(六) 學生日誌檔對應之 DTD

```

<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#">
  <rdf:Description ID="Stone">
    <totaltime>129</totaltime>
    <readcourse>4</readcourse>
    <finalmaterial>QuickSort</finalmaterial>
    <detail>
      <rdf:bag>
        <rdf:li><rdf:value>SelectionSort</rdf:value>
          <visit>1</visit><visitpercent>25</visitpercent>
          <time>3</time><timepercent>2</timepercent></rdf:li>
        <rdf:li><rdf:value>InsertSort</rdf:value>
          <visit>1</visit><visitpercent>25</visitpercent>
          <time>3</time><timepercent>2</timepercent></rdf:li>
        <rdf:li><rdf:value>BubbleSort</rdf:value>
          <visit>1</visit><visitpercent>25</visitpercent>
          <time>3</time><timepercent>2</timepercent></rdf:li>
        <rdf:li><rdf:value>QuickSort</rdf:value>
          <visit>1</visit><visitpercent>25</visitpercent>
          <time>120</time><timepercent>93</timepercent></rdf:li>
      </rdf:bag>
    </detail>
  </rdf:Description>
</rdf:RDF>

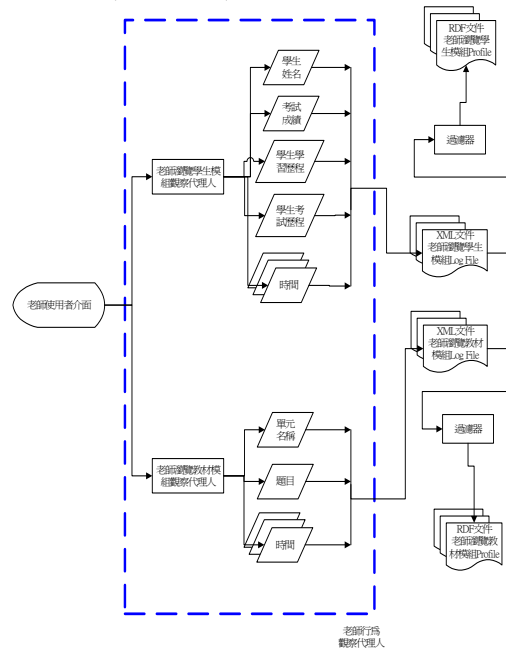
```

程式(七) 學生瀏覽教材時的輪廓(profile)



圖四 學生行為觀察代理人建構學生輪廓流程圖

老師模組的做法和學生模組大同小異，只是其所記錄的資料和學生模組所記錄的資料有些微的不同。除了原本就擁有的基本資料庫外，又新建構了老師線上行為的輪廓。其所記錄的資訊如：老師在瀏覽學生資料時，較喜好檢視何種資訊(例如：由個人或群組的角度來觀察學生、偏好使用哪些資料來觀察學生行為)、使用哪些機制來和學生作互動等。而瀏覽教材模組時，常使用哪一些資訊來取得其所欲瀏覽的教材和題庫。我們對於詳細的資訊都加以蒐集，然後再從中擷取出我們想要的資訊。圖五為老師行為觀察代理人建構老師輪廓流程圖。而表一、表二則分別表示系統已記錄或將記錄之學生、老師線上行為。



圖五 老師行為觀察代理人建構老師輪廓流程圖

表一 系統中記錄的學生線上行為資料

編號	行為	相關說明
1	學習過的課程。	記錄此位學生的背景知識。
2	學習開始的時間。	記錄時間點。
3	學習結束的時間。	記錄時間點。
4	瀏覽某一課程(網頁)的時間。	累計時間量。
5	瀏覽某一課程(網頁)的次數。	累計次數。
6	瀏覽每一課程(網頁)的順序。	記錄學習歷程。
7	進入系統的總時間。	累計時間量。
8	進入系統的總次數。	累計次數。
9	在討論室中發表文章的記錄。	學生回答哪些問題，或提出哪些問題。
10	在聊天室的發言情況。	學生的發言內容。
11	互動的對象。	被呼叫的使用者為何?
12	測驗的情況。	作答的題目為何?對錯?
13	學生的作業、測驗成績。	記錄學習評量。
14	搜尋的情況。	搜尋的關鍵字為何?

表二 系統中記錄的老師線上行為資料

編號	行為	相關說明
1	進入系統的時間。	記錄時間點。
2	離開系統的時間。	記錄時間點。
3	進入系統的總時間。	累計時間量。
4	進入系統的總次數。	累計次數。
5	透過何種資訊瀏覽學生狀態。	觀察學生測驗成績、瀏覽教材的時間或作業的分數。
6	使用何種角度來瀏覽學生的行為。	針對單一學生來進行觀察。或針對某一種類的學生來進行觀察。
7	在討論室中發表文章的記錄。	記錄老師回答哪些問題，或提出哪些問題。
8	在聊天室的發言情況。	老師的發言內容。

9	互動的對象。	被呼叫的使用者為何?
10	瀏覽、修改哪一教材、題目。	記錄老師瀏覽、修改哪一頁教材或哪一題目。
11	學生對於老師的評估回饋。	讓學生填寫問卷調查表。

至於使用者介面，其所呈現給使用者的畫面則是根據導覽機制所給與的量化值來決定資訊排列的先後順序，或者是隱藏某些資訊不顯示在使用者面前等。導覽機制是根據我們前面所整理的輪廓來作建議的，其根據相類似的輪廓(如：對某一教材的總瀏覽時間或瀏覽次數)來判定某一類使用者是否看法相同(like-minded)同一類型。或者直接比對使用者的日誌檔來作分類。在這裡要強調的是，系統中是採用XML和RDF資料格式來記錄相關的資訊，這些檔案格式對於系統而言都是可以輕易了解的。一旦系統可以輕易的了解資訊，便可以使用資料探勘的技術從輪廓中擷取出所要的資訊，提供使用者個人化的服務了[12][13][20][21]。

由於學生和老師所欲瀏覽的畫面不全相同，所以系統用來判斷使用者喜好的先決條件也不盡相同。圖六和圖七分別為學生和老師的導覽機制流程圖。經由簡單的流程圖，便可以清楚的看見這兩個導覽機制的差異。

本系統依照實際的教學和功能區分為四大模組，過去幾年，本實驗室對於這些個別模組已經做了些許的研究，也獲得了初步的經驗和實驗結果[2][6]，因此本系統著重在應用知識本體於整個系統的規劃和設計上。並將這些收集到的資料加以彙整，以作為實作網頁使用探勘的基礎。不同以往的只是在於我們使用了XML和RDF的檔案方式來記錄我們所需要的個別使用者資料，就如前面章節所敘述的，一個共通的標準，可以讓我們更輕易地設計軟體程式。做個人化之前，要先讓電腦了解我們的需求，也就是要讓系統可以處理(了解)我們所需要的資訊。知識本體只是提供個人化服務一個穩定的基礎，透過一個標準化格式的建立，電腦在處理資訊時，不再像以往一樣盲目的搜尋，而是可以根據知識本體所定義的階層式架構來處理，以達到我們所要的目標。

5. 結論

網路學習，勢必成為未來的趨勢。許多優秀的網路教學平台已經如雨後春筍般的出現，如：中山大學遠距教學網、鼎新電腦 EasyLearning 及肯心 CanLearn。現今網路學習所應注重的應該是從平台的建置回歸到基本的教材編寫了。教材的編寫並非如所想像中的簡單，以為只要作一些動畫和互動式按鈕即可，其牽涉到許多高深的教育理論、計算機科學、心理學。

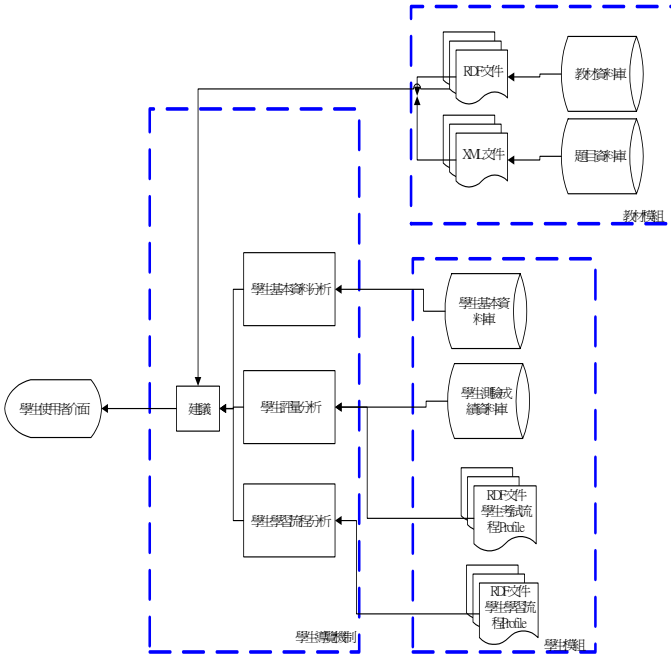
在此，我們將知識本體的觀念引入網路學習。因此所提出的系統架構在此概念的引入後，能提供以下的功能：

- (1) 提高資料的描述性：使用詮釋資料 (metadata) 來建立系統的資料。不管是使用者資料、或者是教材和題目的相關資料，都可以擁有充分的描述。
- (2) 提高資料再使用性：透過一致的知識本體，線上的資料都可以達到再使用的目的。
- (3) 作為個人化的基礎：經由知識本體的應用，電腦便可以了解其中系統資訊，如此就可以使用資料探勘的技術從輪廓中擷取出所要的資訊，提供使用者個人化的服務了。

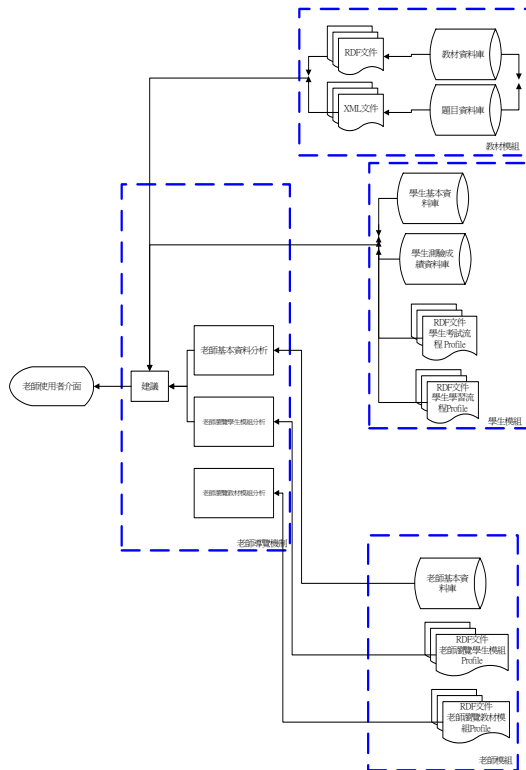
本文提出一系統架構，說明如何使用知識本體語言來建構線上學習系統。實際上，我們的系統目前還在持續建構中，要努力、改進的地方還相當多。未來研究的重點將擺在知識本體的一致性。也就是每一學科教材、測驗題目甚至使用者輪廓等的階層架構都必須符合某一標準的知識本體規格(也可視為樣版(template))，如此加上知識本體語言之輔助建構，以達到語意網頁和再使用的目的。

6. 參考資料

1. 何東隆、李美真，精通 XML 與網頁設計，文魁資訊股份有限公司。
2. 吳怡慧，互動式環境之介面設計—以國中物理力學電腦輔助學習系統為例，逢甲大學資訊工程研究所碩士論文，民 86。
3. 吳凱雯、王孝熙、林敏慧、王逸民，”利用資料探勘技術提供網際網路使用者人化服務”，廿一世紀數位生活與網際網路科技研討會 Session3D-8 論文，民 90。
4. 吳錫修、蔡新民、楊博清、孫霞繡，”以 XML 設計資訊網教學課程之瀏覽控制機制”，第八屆國際電腦輔助教學研討會 (ICCAI'1999) 論文，pp.332-334，民 88。
5. 林信成，”XML 相關技術與下一代 Web 出版趨勢之研究”，



圖六 學生導覽機制



圖七 老師導覽機制

- www.tku.edu.tw/sclin/research/pub/XMLWeb.htm.
6. 陳俊隆，一個以多代理人為基礎之智慧型教學代理人，逢甲大學資訊工程研究所碩士論文，民 89。
 7. 陳桓佑、陳景宜、朱光宇、陳百薰、洪正欣，”以 XML 實作 Metadata 觀念，並應用於遠距教學教材”，第八屆國際電腦輔助教學研討會 (ICCAI'1999) 論文，pp.195-202，民 88。
 8. 陳桓佑、陳景宜、賈言從、洪正欣，”SMIL 在遠距教學之應用研究”，第八屆國際電腦輔助教學研討會 (ICCAI'1999) 論文，pp.213-215，民 88。
 9. 陳煜儒、楊東麟，”利用網路探勘技術提供個人化搜尋服務”，廿一世紀數位生活與網際網路科技研討會 Session3D-5 論文，民 90。
 10. 陳榮靜、沉慧宇，”一個新的網站上瀏覽路徑資料探勘研究”，第六屆資訊管理研究暨實務研討會－資料探勘與應用論文，民 89。
 11. 黃錦法，”XML 應用於多家商店產品搜尋支援”，第六屆資訊管理研究暨實務研討會－網際網路技術與應用論文，民 89。
 12. Gediminas Adomavicius、Alexander Tuzbilin，”Using Data Mining Methods to Build Methods to Build Customer Profiles”，IEEE Computer，vol. 34，no. 2，pp.74-82，Feb. 2001.
 13. Ibrahim Cingil，Asuman Dogac and Ayca Azgin，”A Broader Approach to Personalization”，Communication ACM，vol. 43，no. 8，pp.136-141，Aug. 2000.
 14. Stefan Decker、Prasenjit Mitra and Sergey Melnik，”Framework for the Semantic Web：An RDF Tutorial”，IEEE Internet Computing，vol. 4，no. 6，pp.68-73，Nov/Dec. 2000.
 15. Stefan Decker、Sergey Melnik、Frank Van Harmelen、Dieter Fensel、Michel Klein、Jeen Broekstra、Michael Erdmann and Ian Horrocks，”The Semantic Web：The Roles of XML and RDF”，IEEE Internet Computing，vol. 4，no. 5，pp.63-74，Sept/Oct 2000.
 16. Dieter Fensel，”Ontologies and Electronic Commerce”，IEEE Intelligent System，vol. 16，no. 1，Jan/Feb 2001.
 17. Dieter Fensel、Frank van Harmelen、Ian Horrocks、Deborah L. McGuinness、Peter F. Patel-Schneider，”OIL：An Ontology Infrastructure for the Semantic Web”，vol. 16，no. 2，pp.38-45，March/April 2001.
 18. Horrocks、D.Fensel、J.broekstra、S. Decker、M.Erdmann、C. Grble、F.van Harmelen，”The Ontology Inference Layer OIL”，Aug 2000，<http://www.ontoknowledge.org>.
 19. James Hendler，”Agents and the Semantic Web”，IEEE Intelligent System，vol. 16，no. 2，pp.30-37，March/April 2001.
 20. Cingil, I., Dogac, A. and Azgin, A. “A Broader Approach to Personalization Technical Report”，SRDC，Middle East Technical University，July 1999；www.srdc.metu.edu.tr/publications.html.
 21. Bamshad Mobasher，Robert Cooley，and Jaideep Srivastava，”Automatic Personalization Based on Web Usage Mining”，Communication ACM，vol. 43，no. 8，pp.142-151，Aug. 2000.
 22. Mahbubur Rahman syed，Diminishing the Distance in Distance Education”，IEEE Multimedia，vol. 8，no. 3，pp.18-20，July-Sept，2001.
 23. Len Seligman、Arnon Rosentbal，”XML's Impact on Databases and Data Sharing”，IEEE Computer，vol. 34，no. 6，pp.59-67，June 2001.