TRƯỜNG ĐẠI HỌC HÀNG HẢI VIỆT NAM

KHOA CÔNG NGHỆ THÔNG TIN

THUYẾT MINH

ĐỀ TÀI NCKH CẤP TRƯỜNG

ĐỀ TÀI

Nghiên cứu chuẩn Dublin Core Metadata, ứng dụng xây dựng giải pháp thư viên số cung cấp tài liệu chuyên ngành cho Khoa Công nghệ thông tin – Trường Đại học Hàng hải Việt Nam.

Chủ nhiệm đề tài: KS. Lê Hoàng Dương

Hải Phòng, tháng 05 / 2015

MỤC LỤC

Mở đầu

Ngày nay, việc xây dựng các cổng thông tin điện tử là một nhu cầu cấp thiết đối với các trường đại học nhằm cung cấp công cụ truy cập đến các tài nguyên thông tin của Nhà trường cho người dùng, đặc biệt là đối tượng giảng viên và sinh viên. Tài liệu học tập, giáo trình, luận văn, tài liệu tham khảo là những tài nguyên vô cùng quan trọng nhằm phục vụ cho nhu cầu nghiên cứu và học tập của giảng viên và sinh viên của Nhà trường. Giải pháp xây dựng các thư viện tài liệu số để tích hợp vào trong cổng thông tin của Nhà trường đang được rất nhiều trường đại học quan tâm và phát triển. Tuy nhiên, vấn đề đặt ra hiện nay cho các thư viện tài liệu số là việc quản lý các tài nguyên khổng lồ của thư viện như thế nào để hỗ trợ việc tìm kiếm, truy hồi thông tin dễ dàng hơn, chính xác hơn, tìm kiếm theo ngữ cảnh của người sử dụng.

Để giải quyết các yêu cầu trên thì thư viện số phải sử dụng siêu dữ liệu chung để mô tả các bản ghi của danh mục và các từ vựng điều khiển chung cho phép gán định danh các tài liệu. Các thư viện tài liệu số thường sử dụng một chuẩn siêu dữ liệu nào đó để tổ chức các mô tả tài nguyên. Các chuẩn định dạng mô tả tài nguyên phổ biến như MARC, Dublin Core, BibTex,… Trong giới hạn của nghiên cứu này, tác giả tập trung vào việc tìm hiểu và xây dựng bộ siêu dữ liệu theo chuẩn Dublin Core. Tuy nhiên, trong quá trình thực hiện tìm hiểu và nghiên cứu, tác giả nhận thấy rằng việc sử dụng chuẩn siêu dữ liệu Dublin Core chỉ là tiền đề giúp tổ chức được các tệp thông tin phục vụ cho việc xây dựng các quan hệ ngữ cảnh của tài liệu, hướng tới việc triển khai hệ thống theo công nghệ Web 3.0 – Semantic Web. Vì vậy, hướng phát triển của đề tài trong thời gian tới để hoàn thiện được hệ thống tài liệu số chuyên ngành công nghệ thông tin là sẽ áp dụng công nghệ Semantic web vào trong hệ thống đang triển khai.

Mục đích của đề tài: tìm hiểu về chuẩn siêu dữ liệu Dublin Core, thực hiện phân tích hệ thống tài liệu số, triển khai xây dựng hệ thống tài liệu số chuyên ngành công nghệ thông tin có đính kèm thêm các siêu dữ liệu theo chuẩn Dublin Core.

Nội dung báo cáo: bao gồm phần mở đầu, 2 chương và phần kết luận. Chương I sẽ trình bày các kiến thức về siêu dữ liệu, chuẩn siêu dữ liệu Dublin Core, việc mã hóa siêu dữ liệu Dublin Core sử dụng XML và RDF. Chương II sẽ trình bày về việc phân tích chức năng bài toán thư viện tài liệu số, đặc tả dữ liệu của hệ thống và kết quả cài đặt.

CHƯƠNG 1: CƠ SỞ LÝ THUYẾT VỀ SIÊU DỮ LIỆU VÀ DUBLIN CORE

1.1 Metadata (siêu dữ liệu)

1.1.1 Metadata là gì

Metadata (siêu dữ liệu) là một thuật ngữ hiện đại cho các mục thông tin mà các thư viện truyền thống đưa vào các biên mục hoặc cơ sở dữ liệu của họ; hoặc là thông tin khai báo về những bộ sưu tập mà các bảo tàng đưa vào hệ thống của họ; Tuy nhiên thuật ngữ "siêu dữ liệu" thường được sử dụng để đề cập đến thông tin mô tả về những tài nguyên số. ***Metadata*** còn được định nghĩa là dữ liệu về các dữ liệu, là những thông tin chuyển tải ý nghĩa của các thông tin khác. Metadata bao gồm một tập hợp các phần tử thiết yếu để mô tả nguồn thông tin. Thuật ngữ "meta" xuất xứ là một từ Hy Lạp dùng để chỉ một cái gì đó có bản chất cơ bản hơn hoặc cao hơn. Vì vậy metadata là dữ liệu về dữ liệu.

Theo tiến sĩ Warwick Cathro (Thư viện Quốc gia Úc) thì "siêu dữ liệu là những thành phần mô tả tài nguyên thông tin hoặc hỗ trợ thông tin truy cập đến tài nguyên thông tin". Cụ thể trong tài liệu thì siêu dữ liệu được xác định là "dữ liệu mô tả các thuộc tính của đối tượng thông tin và trao cho các thuộc tính này ý nghĩa, khung cảnh và tổ chức. Siêu dữ liệu còn có thể được định nghĩa là dữ liệu có cấu trúc về dữ liệu".

Theo Gail Hodge siêu dữ liệu là "thông tin có cấu trúc mà nó mô tả, giải thích, định vị, hoặc làm cho nguồn tin trở nên dễ tìm kiếm, sử dụng và quản lý hơn. Siêu dữ liệu được hiểu là dữ liệu về dữ liệu hoặc thông tin về thông tin". Nói tóm lại thì siêu dữ liệu là thông tin mô tả tài nguyên thông tin.

1.1.2 Mục đích và yêu cầu

Mục đích và yêu cầu cốt lõi nhất của siêu dữ liệu (metadata) là góp phần mô tả và tìm lại các tài liệu điện tử trên mạng Internet. Sự phát triển mạnh mẽ của Internet đã tạo ra sự bùng nổ của các loại dữ liệu đa dạng ở dạng số, văn bản, âm thanh, hình ảnh, tài liệu đa phương tiện. Những tài liệu này có thể truy cập được trên mạng Internet song việc tìm kiếm chúng một cách hiệu quả và khoa học như với các hệ thống thông tin trực tuyến là hết sức khó khăn. Để góp phần tăng cường chất lượng tìm kiếm các tài liệu số trên mạng Internet, người ta đã đưa ra giải pháp sử dụng siêu dữ liệu.

Thực ra trong hoạt động thông tin – thư viện truyền thống, từ lâu đã có những khái niệm liên quan đến siêu dữ liệu. Các bản thư mục chứa các dữ liệu mô tả đối tượng như cho sách , cho tạp chí thì chúng cũng được coi như là một dạng siêu dữ liệu. Với việc tự động hóa công tác biên mục, phiếu thư mục được thay thế bằng biểu ghi thư mục. Như vậy thành phần siêu dữ liệu còn có thể được trình bày trong biểu ghi, vì vậy biểu ghi này

được coi là biểu ghi siêu dữ liệu (metadata record) của đối tượng được cơ sở dữ liệu quản lý. Với tài nguyên truyền thống trên giấy, thông tin mô tả được bố trí nằm ngoài đối tượng mà nó mô tả (Ví dụ, trên phiếu thư mục của mục lục thư viện, trong biểu ghi của CSDL). Nhờ những yếu tố mô tả như vậy, người ta có thể xác định và tìm kiếm lại được tài liệu một cách chính xác theo một vài yếu tố.

Ngày nay, nguồn tài liệu điện tử phát triển nhanh chóng và sự phân tán trên mạng nhiều đến mức không thể xử lý được một cách thủ công như đã và đang áp dụng đối với tài liệu xuất bản trên giấy. Để xử lý được hết tài liệu điện tử phân tán, người ta phải áp dụng các phương pháp tự động – sử dụng các chương trình đặc biệt (được gọi theo nhiều cách khác nhau như: robots, crawlers, spiders,...). Do tài liệu điện tử được tạo ra, thông thường không tuân thủ những quy định xuất bản truyền thống, không có những quy tắc nhất định giúp cho phép nhận dạng tự động được các yếu tố mô tả thông thường như tác giả, địa chỉ về xuất bản, thông tin về khối lượng... nên *cần thiết phải có những quy định thống nhất để các chương trình tự động nhận dạng và xử lý chúng theo các yêu cầu nghiệp vụ*. Những quy định như vậy được gọi là những quy định về siêu dữ liệu. Có thể thấy hiện nay, do nhiều chương trình máy tính chỉ định chỉ số dựa vào một số thành phần hạn chế như nhan đề hoặc toàn văn nên không hỗ trợ những tìm kiếm đặc thù (ví dụ theo tác giả, theo chủ đề, theo lĩnh vực...). Vì thế để tạo điều kiện cho các chương trình có thể đinh chỉ số tự động theo một số yếu tố xác định, người ta phải đưa thêm vào tài liệu điện tử những thuộc tính bổ sung để tăng cường mô tả tài nguyên thông tin. Các công cụ định chỉ số tự động sẽ được lập trình để nhận dạng các thuộc tính này và định chỉ số chúng, từ đó hỗ trợ tìm kiếm những thuộc tính đặc thù.

Như vậy một bản ghi metadata bao gồm một tập hợp những thuộc tính hoặc tập hợp những phần tử cần thiết để mô tả các tài nguuyên thông tin theo yêu cầu nghiệp vụ. Thông thường trong hoạt động nghiệp vụ thông tin – thư viện bao gồm các yếu tố như: Nhan đề tài liệu, tác giả, thông tin về xuất bản, nơi/vị trí lưu giữ, kiểu/dạng tài liệu...

1.1.3 Các loại Metadata

Việc tạo ra siêu dữ liệu cho các tài nguyên số là một phần quan trọng của các dự án số hóa và phải được kết hợp chặt chẽ vào các dòng công việc của dự án. Siêu dữ liệu nên được tạo ra và phù hợp với tài nguyên số để hỗ trợ cho việc khai thác, sử dụng, quản lý, tái sử dụng và xác minh các tài nguyên. Siêu dữ liệu thường được chia thành 3 loại:

Siêu dữ liệu mô tả (Descroptive metadata): sử dụng để đánh chỉ mục, khai thác và định danh tài nguyên số. Siêu dữ liệu dạng này cung cấp thông tin mà *cho phép phát hiện các bộ sưu tập hoặc đối tượng số thông qua sử dụng công cụ tìm kiếm,* và *cung cấp một ngữ cảnh nhằm giúp người dùng hiểu được thông tin gì*

đang tìm kiếm. Siêu dữ liệu cho mỗi đối tượng số cụ thể sẽ khác nhau tuỳ thuộc vào đối tượng số đó, nhưng thường bao gồm những phần tử thông tin như nhan đề hay tiêu đề - nó là cái gì, ai tạo ra nó, người cộng tác là ai (Contributors), ngôn ngữ, nó được tạo ra khi nào, vị trí của nó ở đâu, chủ đề, vv … Ở cấp độ của bộ sưu tập, người dùng thường có thể quyết định phạm vi, sự sở hữu, những hạn chế truy cập, và nhiều đặc tính quan trọng khác nhằm giúp người dùng hiểu được bộ sưu tập số đó. Một số chuẩn siêu dữ liệu mô tả có thể kể đến là MARC (**MA**chine-**R**eadable **C**atalog)và DC (Dublin Core).

Siêu dữ liệu cấu trúc (Structural metadata): mô tả các liên kết trong phạm vi hoặc giữa mỗi đối tượng thông tin liên quan. Một cuốn sách bao gồm các trang và chương sách là một trong những ví dụ rõ ràng nhất của siêu dữ liệu cấu trúc. Siêu dữ liệu cấu trúc thường sẽ giải thích các hình ảnh trang sách cấu thành lên mỗi chương sách như thế nào, và những chương sách đó cấu thành lên một cuốn sách như thế nào. Ngoài ra, cũng có những hình vẽ minh họa riêng rẽ, và siêu dữ liệu cấu trúc cũng có thể liên kết những hình này tới các chương sách, hoặc tới một danh mục bao gồm tất cả các hình ảnh minh họa trong một cuốn sách. Siêu dữ liệu cấu trúc trợ giúp người dùng di chuyển giữa mỗi đối tượng, bao gồm cả một đối tượng phức hợp.

Siêu dữ liệu quản trị (Administrative Metadata): Biểu diễn thông tin quản lý cho đối tượng số bao gồm: thông tin cần thiết để truy nhập và hiển thị tài nguyên và thông tin quản lý tài nguyên. Cụ thể Siêu dữ liệu quản trị có thể:

+ Mô tả một trình xem và duyệt thông tin, hoặc trình vận hành cần thiết để truy cập một đối tượng, tự động mở trình xem hoặc vận hành khi một người sử dụng chọn một nguồn tài nguyên số nào đó.

+ Mô tả các thuộc tính như độ phân giải của hình ảnh, kích cỡ tệp tin, hoặc tốc độ truyền tệp tin âm thanh.

+ Cung cấp một biểu ghi thông tin về một đối tượng đã được tạo ra khi nào và như thế nào, cũng như thông tin về quản lý quyền và lưu trữ.

Một chuẩn siêu dữ liệu quản trị có thể kể đến **METS -Tiêu chuẩn Truyền và Mã hóa Siêu dữ liệu (Metadata Encoding and Transmission Standard). METS** cung cấp một cấu trúc thống nhất để quản lý và truyền đi các đối tượng số. Dự án MOA2 (The Making of America II Project) đã phát triển thành công một định dạng mã hóa cho siêu dữ liệu mô tả, siêu dữ liệu cấu trúc và quản trị đối với các tài liệu dưới dạng hình ảnh, hoặc văn bản. Được Liên hiệp Thư viện số (Digital Library Federation) và Thư viện Quốc hội Mỹ (Library of Congress) ủng hộ, **METS** xây dựng dựa trên công việc nghiên

cứu của dự án MOA2. Tiêu chuẩn này cung cấp một định dạng cho mã hóa siêu dữ liệu cần thiết để quản lý đối tượng số của thư viện trong phạm vi một kho cơ sở dữ liệu, cũng như sự trao đổi các đối tượng số như vậy giữa nhiều kho cơ sở dữ liệu (hoặc giữa các kho cơ sở dữ liệu và người dùng). Những thư viện học thuật và nghiên cứu hàng đầu hiện nay đang trích dẫn **METS** như là một tiêu chuẩn quan trọng để vận hành gắn kết lẫn nhau trong một thư viện số, và dường như nó đang được hẫu thuẫn ngày càng đông trong cộng đồng thư viện trên thế giới.

1.1.4 Metadata được đặt ở đâu

Mối liên hệ giữa siêu dữ liệu và tài nguyên thông tin mà nó mô tả có thể được thể hiện ở một trong hai cách sau:

- Các phần tử metadata được chứa trong một biểu ghi tách biệt bên ngoài đối tượng mô tả.
- Các phần tử metadata có thể được nhúng (gắn) vào bên trong tài nguyên mà nó mô tả.

Trước đây với tài liệu truyền thống, các mô tả dữ liệu nằm ngoài đối tượng mô tả (được đưa vào phiếu thư viện hoặc biểu ghi CSDL), như vậy siêu dữ liệu được lưu trữ một cách tách biệt bên ngoài đối tương mô tả.

Với tài liệu điện tử, siêu dữ liệu của chúng được nhúng (gắn) trong bản thân tài nguyên hoặc liên kết với tài nguyên mà nó mô tả như trong trường hợp các thẻ meta của tài liệu HTML hoặc các tiêu đề TEI trong tài liệu điện tử.

Trong thực tế có nhiều chuẩn mô tả biên mục mang tính chất metadata khá thông dụng đang được áp dụng như: MARC21/UNIMARC, ISO-2709, Dublin Core Metadata... các dữ liệu metadata này thường được gắn vào phần đầu cho mỗi tài liệu điện tử được đưa vào máy chủ hoặc trên mạng internet nhằm hỗ trợ các công cụ tìm kiếm lọc ra các thông tin metadata để tổ chức thành các kho dữ liệu mà không cần dùng đến hệ quản trị cơ sở dữ liệu truyền thống. Thực tế thì ngay bản thân ngôn ngữ XML tự nó đã hỗ trợ việc hình thành một cơ sở dữ liệu toàn văn, phi cấu trúc và rất thuận lợi cho việc tìm kiếm và trao đổi thông tin.

1.2 Dublin Core Metadata

1.2.1 Dublin Core Metadata là gì

Dublin Core là một chuẩn siêu dữ liệu được quốc tế công nhận gồm 15 phần tử, được sử dụng để mô tả các loại tài nguyên số. Các phần tử này được thiết lập và thống nhất thông qua sự đồng thuận của quốc tế, nhóm liên ngành của các chuyên gia từ các thư viện, bảo tàng, nhà xuất bản và các lĩnh vực liên quan.

Bộ yếu tố này được hình thành lần đầu tiên vào năm 1995 bao gồm 15 yếu tố mô tả cốt lõi nhất (trong khi Marc21 có hơn 200 trường và rất nhiều trường con). Tháng 9/2001 bộ yếu tố siêu dữ liệu Dublin Core được ban hành thành tiêu chuẩn Mỹ, gọi là tiêu chuẩn "The Dublin Core Metadata Element Set" ANSI/NISO Z39.85-2001.

1.2.2 Đặc điểm của Dublin Core

(1) Tạo lập và sử dụng dễ dàng: cho phép những người không chuyên nghiệp có thể tạo các bản ghi mô tả đơn giản cho các tài nguyên thông tin và truy xuất chúng trên môi trường mạng một cách dễ dàng.

(2) Ngữ nghĩa dễ hiểu, sử dụng đơn giản: Việc khai thác thông tin trên mạng internet diện rộng thường gặp trở ngại bởi những sự khác nhau về thuật ngữ và sự mô tả thực tế. Dublin Core Metadata giúp những người dò tìm thông tin không chuyên có thể tìm thấy vấn đề mình quan tâm bằng cách hỗ trợ một tập hợp các phần tử thông dụng mà ngữ nghĩa của chúng được hiểu phổ biến. Vd.: yếu tố <tác giả> (Creator) được gán cho người tạo lập, nhà soạn nhạc, đạo diễn, trong vai trò là tác giả chính.

(3) Phạm vi quốc tế: Sự tham gia của hầu hết các đại diện từ các châu lục trong việc thiết lập các thông số kỹ thuật cho Dublin Core đảm bảo rằng Dublin Core có thể giải quyết được vấn đề đa văn hóa và đa ngôn ngữ của các tài liệu kỹ thuật số. Tháng 11 - 1999, đã có phiên bản của hơn 20 thứ tiếng: Phần Lan, Na Uy, Thái Lan, Nhật, Pháp, Đức, Hy Lạp, Indonesia, Tây Ban Nha. Tổ chức WWW phát triển Chuẩn Dublin Core trên nền tảng kết hợp đa ngôn ngữ, phục vụ cho môi trường tài nguyên thông tin điện tử mang tính chất đa văn hoá và đa ngôn ngữ. Hiện nay phiên bản 1.1 đã hỗ trợ 25 ngôn ngữ khác nhau.

(4) Khả năng mở rộng: Những nhà phát triển Dublin Core đã cung cấp một cơ chế cho việc mở rộng tập các phần tử Dublin Core, phục vụ nhu cầu khai thác các tài nguyên bổ sung. Các phần từ Metadata từ những tập các phần tử khác nhau có thể liên kết với metadata của Dublin Core. Điều này cho phép các tổ chức khác nhau với các chuyên ngành khác nhau có thể dùng các phần tử Dublin Core để mô tả thông tin thích hợp cho việc sử dụng tài nguyên trên Internet.

1.2.3 Ý nghĩa của Dublin Core trong Thư viện số

(1) Là một phương thức mô tả nguồn thông tin, đặc biệt là nguồn thông tin điện tử một cách có hiệu quả. Dublin Core càng đặc biệt phát huy tác dụng khi được sử dụng để mô tả tư liệu điện tử vốn khó xác định được loại hình và nội dung các yếu tố cần thể hiện.

(2) Thay thế cho các dạng thức trình bày thông tin trước đây như MARC do sự đơn giản trong cấu trúc mà người sử dụng có thể tự thiết kế theo yêu cầu của riêng mình.

(3) Cung cấp cho người sử dụng một phương án tiếp cận thông dụng thông qua các giao diện quen thuộc như Web.

(4) Tạo cho người cán bộ thư viện sự thuận tiện trong công tác khi không còn phải gò bó trong các trường, các yếu tố vốn dĩ đã rất đa dạng và phức tạp.

1.2.4 Các yếu tố của Dublin Core

a. Phân loại các yếu tố:

NỘI DUNG	**SỞ HỮU TRÍ TUỆ**	**THUYẾT MINH**
Nhan đề (Title)	Tác giả (Creator)	Ngày tháng (Date)
Đề mục (Subject)	Tác giả phụ (Contributor)	Mô tả vật lý (Format)
Mô tả (Description)	Xuất bản (Publisher)	Định danh (Indentifier)
Loại hình (Type)	Bản quyền (Rights)	Ngôn ngữ (Language)
Nguồn gốc (Source)		
Liên kết (Relation)		
Nơi chứa (Coverage)		

Bảng 1.1 Danh sách các yếu tố của Dublin Core

b. Các yếu tố cơ bản: Các yếu tố cơ bản của Dublin Core đều mang thuộc tính lựa chọn và có thể lặp lại. Mỗi yếu tố cũng có một giới hạn những hạn định, thuộc tính nhằm diễn giải chính xác ý nghĩa của các yếu tố.

1. ***Nhan đề (Title):*** Tên của nguồn thông tin thường do tác giả hoặc nhà xuất bản đặt cho tài liệu.
2. ***Tác giả (Creator):*** Người hoặc cơ quan chịu tránh nhiệm chính về nội dung trí tuệ của nguồn thông tin.
3. ***Đề mục (Subject):*** Chủ đề của nguồn thông tin và được thể hiện bằng từ vựng có kiểm soát gồm tiêu đề đề mục, số phân loại,...
4. ***Mô tả (Description):*** Phần thể hiện nội dung của nguồn thông tin bao gồm cả phần tóm tắt của tư liệu văn bản hoặc nội dung của tư liệu nghe nhìn
5. ***Xuất bản (Publisher):*** Cơ quan, tổ chức chịu trách nhiệm tạo lập, xuất bản nguồn thông tin trong định dạng thực.
6. ***Tác giả phụ (Contributor):*** Cá nhân hay tổ chức có những đóng góp về mặt trí tuệ cho tư liệu nhưng không phải là tác giả chính.

7. ***Ngày tháng (Date):*** ngày tháng có liên quan đến việc tạo lập, xuất bản hay công bố tư liệu. Có thể dùng chuẩn ISO 8601 (http://www.w3.org/TR/NOTE-datetime). Tham khảo chuẩn MIME tại:

   http://www.utoronto.ca/webdocs/HTMLdocs/Book/Book-3ed/appb/mimetype.html

8. ***Loại hình (Type):*** bản chất hay thể loại của tài nguyên được mô tả.
9. ***Mô tả vật lý (Format):*** Định dạng vật lý và kích thước của tư liệu như kích cỡ, thời lượng,.. Định dạng cũng còn được dùng để chỉ rõ phần mềm và phần cứng cần thiết để sử dụng tư liệu.
10. ***Định danh tư liệu (Identifier):*** Các thông tin về định danh tài liệu, các nguồn tham chiếu đến, hoặc chuỗi ký tự để định vị tài nguyên: URL (Uniform Resource Locators) (bắt đầu bằng http://), URN (Uniform Resource Name), ISBN (International Standard Book Number), ISSN (International Standard Serial Number), SICI (Serial Item & Contribution Identifier), ...
11. ***Nguồn gốc (Source):*** Các thông tin về xuất xứ của tài liệu, tham chiếu đến nguồn mà tài liệu hiện mô tả được trích ra/tạo ra, nguồn cũng có thể là: đường dẫn (URL), URN, ISBN, ISSN...
12. ***Ngôn ngữ (Language):*** Các thông tin về ngôn ngữ, mô tả ngôn ngữ chính của tài liệu: Có thể sử dụng chuẩn ISO 639 (tham khảo http://www.w3.org/WAI/ER/IG/ert/iso639.htm) để mô tả ngôn ngữ cho tài liệu.
13. ***Liên kết (Relation):*** Yếu tố này thể hiện những kết nối giữa những nguồn tư liệu có liên quan, mô tả các thông tin liên quan đến tài liệu khác. Có thể dùng đường dẫn (URL), URN, ISBN, ISSN...
14. ***Nơi chứa (Coverage):*** Những đặc tính về không gian và/hoặc thời gian của tư liệu. Không gian nơi chứa chỉ ra một vùng sử dụng địa danh hoặc toạ độ. Đặc tính thời gian trong yếu tố này chỉ ra khoảng thời gian mà tư liệu đề cập tới.
15. ***Bản quyền (Rights):*** Thông tin về tình trạng bản quyền, kết nối tới thông tin về tình trạng bản quyền hoặc dịch vụ cung cấp thông tin bản quyền cho tư liệu.

1.2.5 Các yếu tố mở rộng

Thực tế sử dụng Dublin Core cho thấy mỗi yếu tố cơ bản còn gộp chứa trong nó một vài thành tố phụ nhằm diễn đạt chi tiết hơn nội dung chính yếu tố đó. Các thành tố phụ được coi là các yếu tố mở rộng và được thể hiện thông qua những khung mã hoá cụ thể. Ví dụ khi thể hiện nội dung của một tài liệu, người ta cung cấp một vài cách tiếp cận khác nhau như qua ký hiệu phân loại, tiêu đề đề mục, từ khoá.

YẾU TỐ	YẾU TỐ MỞ RỘNG
Nhan đề (Title)	Nhan đề thay thế (isReplaceby)
Tác giả (Creator)	
Đề mục (Subject)	
Mô tả (Description)	Mục lục (Table of Contents) Tóm tắt (Abstract)
Xuất bản (Publisher)	
Tác giả phụ (Contributor)	
Ngày tháng (Date)	Tạo lập (Created) Có giá trị (Valid) Có hiệu lực (Available) Xuất bản (Issued) Hiệu đính (Modified)
Loại tài liệu (Type)	
Mô tả vật lý (Format)	Kích thước và thời lượng (Extent) Vật mang tin (Medium)
Định danh	
Nguồn gốc	
Ngôn ngữ	
Liên kết	
Nơi chứa	

Bản quyền	

Bảng 1.2 Danh sách yếu tố mở rộng của Dublin Core

1.2.6 So sánh đối chiếu với các yếu tố mô tả AACR2 và MARC

DC	AACR2	MARC
Nhan đề	Nhan đề chính	245$a
Tác giả	Tác giả chính	100, 245$c
Đề mục	Điểm truy cập khác	050, 082, 650
Mô tả	Phụ chú nội dung, yếu tố bổ sung nhan đề	245$b
Xuất bản	Nơi và nhà xuất bản	260$a, 260$b
Tác giả phụ	Tác giả liên quan	
Ngày	Năm xuất bản	260$c
Loại tài liệu	Phụ chú hình thức	
Mô tả vật lý	Mô tả vật lý	300
Định danh		
Nguồn gốc		
Ngôn ngữ		
Liên kết	Phụ chú	
Nơi chứa		
Bản quyền		

Bảng 1.3 So sánh các yếu tố của DC với AACR2 và MARC

1.3 Mã hóa Dublin Core trong XML

1.3.1 Một số kiến thức cơ bản về XML

XML (eXtensible Markup Language): là ngôn ngữ tạo cấu trúc dữ liệu văn bản được phát triển từ đầu năm 1996 dựa theo và tận dụng những điểm mạnh của chuẩn SGML (Standard Generalized Markup Language: được coi như là siêu ngôn ngữ có khả năng sinh ngôn ngữ khác), cùng những kinh nghiệm có được từ ngôn ngữ HTML (HyperText Markup Language). SGML phát triển cho việc định cấu trúc và nội dung tài liệu điện tử do tổ chức ISO (International Organization for Standardization) chuẩn hóa năm 1986.

SGML là do IBM đưa ra nhưng được phát triển bởi W3C (World Wide Web Consortium: tổ chức độc lập định ra tiêu chuẩn cho định dạng Web, máy chủ và ngôn ngữ), nhưng đặc tả XML lại do Netscape, Microsoft và các thành viên dự án Text Encoding Intiative (TEI) xây dựng. Tổ chức W3C XML Special Interest Group có đại diện từ hơn 100 công ty cùng nhiều chuyên gia được mời khác. W3C chính thức thông qua chuẩn XML vào tháng 2/1998.

XML là một hệ thống có luật dùng cho việc thiết kế các khổ mẫu (format) cho văn bản giúp tạo cấu trúc cho dữ liệu. Trong thực tế XML không phải là một ngôn ngữ lập trình, XML giúp máy tính dễ dàng tạo dữ liệu, đọc dữ liệu, trao đổi dữ liệu và làm cho cấu trúc dữ liệu trở nên rõ ràng và dễ hiểu hơn, ngoài ra XML còn có thể mở rộng, có nền tảng hoàn toàn độc lập và hỗ trợ tính quốc tế hóa, nội địa hóa. XML hỗ trợ hoàn toàn unicode.

XML và HTML?

Trong thực tế bản thân ngôn ngữ XML có nguồn gốc giống như ngôn ngữ định dạng siêu văn bản HTML (HyperText Markup Language) từ chuẩn ngôn ngữ định dạng văn bản tổng quát có cấu trúc SGML. Mỗi văn bản XML cũng sử các thẻ (tags), các từ được đặt trong ngoặc với ‘’ (mở và đóng) và dùng thuộc tính tên gọi của các phần tử (element) với mẫu name= “value”.

Trong khi HTML đặc biệt chú ý tới từng thẻ (tag) và thuộc tính (attribute) có ý nghĩa gì và phần văn bản giữa các thẻ đó hiển thị như thế nào trên trình duyệt thì XML sử dụng các thẻ chỉ để phân định ranh giới giữa các đoạn dữ liệu và coi việc đọc và xử lý dữ liệu hoàn toàn là nhiệm vụ của các ứng dụng. Nhưng khác với ngôn ngữ HTML, số lượng và tên gọi các phần tử trong XML là không hạn chế.

XML là một văn bản nhưng không giống với những loại văn bản thông thường mà ta có thể đọc được. Các chương trình dùng để tạo các dữ liệu được cấu trúc hóa thông thường được lưu dữ liệu trên đĩa cứng, sử dụng khuôn dạng text hay nhị phân. Một thuận lợi của khuôn dạng văn bản là cho phép người đọc có thể đọc nó với bất kỳ bộ soạn thảo

văn bản nào tùy thích. Các khuôn dạng văn bản cũng cho phép tìm lỗi dễ dàng hơn trong các ứng dụng. Giống như HTML các file XML là những file văn bản được tạo ra không phải với mục đích đề đọc, nhưng vẫn có thể đọc nếu thấy cần thiết. Tuy nhiên XML có điểm không bằng HTML, các luật dùng trong XML rất hạn chế, chỉ cần quên một thẻ, hay một thuộc tính không đi kèm với nội dung sẽ làm cho toàn bộ file XML đó ngừng hoạt động, trong khi đó lỗi này ở file HTML có thể được bỏ qua.

XML được xem như là ngôn ngữ mạnh hơn HTML do nó mang lại thông tin đầy đủ về dữ liệu. XML cung cấp "siêu dữ liệu" metadata hay còn được gọi là "dữ liệu về dữ liệu" (data about data). XML cho phép các nhà phát triển và quản trị công nghệ thông tin mô tả thông tin có liên quan tới các nguồn thông tin khác. Đây là phương pháp khai thác thông tin lý tưởng trong môi trường trao đổi thông tin từ các máy chủ ứng dụng cũng như từ các ứng dụng với nhau. Cấu trúc chặt chẽ của XML (nội dung được đặt giữa các thẻ metadata) cho phép các ứng dụng dễ dàng tìm kiếm và sử dụng nội dung đã tạo. Môi trường tài liệu XML trở thành một kho dữ liệu hỏi-đáp (query data repository) tương tự như cơ sở dữ liệu. Ngôn ngữ XML là giải pháp tích hợp cho vấn đề trao đổi dữ liệu tự động giữa các kho thông tin trên mạng Internet.

1.3.1.1 Chỉ thị xử lý và lời chú thích

Chúng ta thường thấy dòng lệnh <?xml version="1.0" encoding="utf-8" standalone="yes"?> nằm ở đầu file XML. Đây chính là chỉ thị xử lý, chỉ thị xử lý được đặt trong cặp Tag <? và ?>. Nó cho biết phiên bản đặc tả XML mà bộ phân tích cần làm theo, ngoài ra nó cho phép người lập trình cho biết dữ liệu trong XML dùng encoding nào, còn thuộc tính standalone sẽ cho biết tài liệu XML có cần đến một tài liệu khác không (có hai giá trị cho thuộc tính này đó là "yes" nếu không cần đến một tài liệu khác và "no" nếu cần).

```
<?xml version="1.0" encoding="utf-8"  standalone="yes"?>
      <Order>
        <OrderDate>2002-6-14</OrderDate>
        <Customer>Helen     Mooney</Customer>
        <Item>
          <ProductID>1</ProductID>
        <Quantity>2</Quantity>
         </Item>
        <Item>
          <ProductID>4</ProductID>
        <Quantity>1</Quantity>
         </Item>
</Order>
```

1.3.1.2 Không gian tên (Namespace)

Để khai báo một không gian tên ta chỉ cần đưa thêm thuộc tính xmlns:prefix vào bên trong phần tử gốc, prefix là tên của không gian tên, mỗi không gian tên cần mang một định danh duy nhất. Một không gian tên có thể là một địa chỉ internet hoặc một địa chỉ nào đó miễn là địa chỉ này phải duy nhất. Ví dụ sau đây sẽ tạo ra một không gian tên hs và áp dụng cho tất cả các phần tử

con:

```
<?xml version="1.0"?>
<BookOrder xmlns:hs="http://www.northwindtraders.com/customer">
<hs:Customer >
    <hs:Title>Mr.</Title>
    <hs:FirstName>Graeme</FirstName>
  <hs:LastName>Malcolm</LastName>
  </hs:Customer>
</BookOrder>
```

1.3.1.3 CDATA

Đoạn dữ liệu của CDATA là đoạn dữ liệu nằm giữa <![CDATA [và]]>. Những đoạn dữ liệu nằm trong CDATA khi đi qua trình phân tích sẽ được giữ nguyên như ban đầu, tức là khi gặp CDATA thì trình phân tích sẽ bỏ qua. Điều này rất cần thiết khi chúng ta viết những đoạn mã script trong tài liệu.

```
<script language="javascript">
<![CDATA[
function mag(){
   alert("This is CDATA! ");
}
]]
</script>
```

1.3.2 Data Type Define (DTD)

1.3.2.1 DTD là gì

Document Type Definition trong XML được viết tắt là DTD. Mục đich của DTD là để xác định cấu trúc và luật lệ của một dữ liệu XML. Mỗi XML có một DTD riêng tùy theo muc đích của người viết. DTD sử dụng một cú pháp ngắn gọn khai báo chính xác những yếu tố và tài liệu tham khảo có thể xuất hiện ở đâu trong tài liệu XML. DTD cũng khai báo các thực thể (Entity) có thể được sử dụng trong tài liệu XML.

Tại sao lại sử dụng DTD?

- Với một DTD, mỗi tập tin XML của bạn có thể thực hiện một mô tả của định dạng riêng của mình.
- Với một DTD, các nhóm độc lập của người dân có thể đồng ý sử dụng một DTD tiêu chuẩn cho việc trao đổi dữ liệu.
- Ứng dụng của bạn có thể sử dụng một tiêu chuẩn DTD để xác minh rằng các dữ liệu bạn nhận được từ bên ngoài là hợp lệ.
- Bạn cũng có thể sử dụng một DTD để xác minh dữ liệu của riêng bạn.

1.3.2.2 Cấu trúc DTD và các bước tạo

a. Cấu trúc DTD: bao gồm có 3 phần

Hình 1.1 Cấu trúc DTD

- Khai báo **Element**: khai báo Element gồm có tên của **Element** và nội dung của **Element**
- Khai báo **Attribute**: khai báo **Attribute** thuộc **Element** nào, tên **Attribute**, kiểu dữ liệu của**Attribute** và giá trị mặc định của **Attribute.**
- Khai báo **Entity**: khai báo tên của **Entity**, giá trị của **Entity** hay vị trí của giá trị **Entity**

b. Tạo DTD bao gồm 6 bước:

(1) Khai báo tất cả các element có trong XML

(2) Khai báo các element con cho từng element nếu có

(3) Xác định thứ tự xuất hiện của các element

(4) Khai báo tất cả thuộc tính của từng element nếu có

(5) Khai báo kiểu dữ liệu và giá trị mặc định cho thuộc tính

(6) Khai báo các Entity nếu có

1.3.2.3 Các dạng khai báo DTD

a) Phần tử <!DOCTYPE>

Để bắt đầu định nghĩa kiểu tư liệu DTD tham chiếu nội chúng ta dùng cú pháp sau: <!DOCTYPE root-element [DTD]>

Trong đó root-element là phần tử gốc của tài liệu XML, DTD là các định nghĩa cho các phần tử trong tài liệu XML.

```
<?xml version="1.0"?>
<!DOCTYPE note [
 <!ELEMENT note body>
 <!ELEMENT body    (#PCDATA)>
]>
```

```
<note>
  <body>Don't  forget  me  this  weekend</body>
</note>
```

Sử dụng định nghĩa DTD tham chiếu ngoại sẽ làm cho các ứng dụng XML của chúng ta trở nên dễ dàng chia sẽ và dùng chung với các ứng dụng khác. Có hai cách để chỉ định một DTD tham chiếu ngoại: Tham chiếu ngoại riêng và tham chiếu ngoại chung.

Những định nghĩa DTD tham chiếu ngoại riêng được sử dụng cho một nhóm người mang tính cá nhân, chúng không được dùng cho mục đích chung rộng lớn, mục đích phân phối. Còn những định nghĩa DTD tham chiếu ngoại chung sẽ mang tính cộng đồng hơn.

- Để định nghĩa một DTD tham chiếu ngoại riêng chúng ta dùng cú pháp sau: <!DOCTYPE root-element SYSTEM “filename”>

Trong đó root-element là tên của phần tử gốc trong tài liệu XML, filename là tên file định nghĩa kiểu tư liệu DTD, ví dụ:

```
<?xml version="1.0"?>
<!DOCTYPE  note  SYSTEM  "note.dtd">
<note>
    <to>Tove</to>
     <from>Jani</from>
     <heading>Reminder</heading>
   <body>Don't  forget  me  this  weekend!</body>
</note>
```

File note.dtd với nội dung như sau:

```
<!ELEMENT  note  (to,from,heading,body)>
<!ELEMENT to (#PCDATA)>
<!ELEMENT from (#PCDATA)>
<!ELEMENT  heading  (#PCDATA)>
<!ELEMENT body (#PCDATA)>
```

Địa chỉ chứa file DTD có thể một URL/URI.

```
<?xml version="1.0"?>
<!DOCTYPE note SYSTEM "http://www.w3schools.com/dtd/note.dtd">
<note>
  <to>Tove</to>
  <from>Jani</from>
  <heading>Reminder</heading>
  <body>Don't forget me this weekend!</body>
</note>
```

b) Khai báo Element:

- Element rỗng: **<!ELEMENT element-name EMPTY>**

- Element chứa text dạng parsed character: **<!ELEMENT element-name (#PCDATA)>**

- Element chứa text dạng bất kỳ: **<!ELEMENT element-name ANY>**

- Element với các thẻ con, tuần tự: **<!ELEMENT element-name (child1,child2,...)>**

Khai báo số lần xuất hiện của các thẻ con:

- Chỉ có tên thẻ con: xuất hiện duy nhất 1 lần

- **+**: phải xuất hiện tối thiểu 1 lần

- *****: xuất hiện 0 hay nhiều lần

- **?**: xuất hiện 0 hay 1 lần

- Child1|child2: hoặc child1 xuất hiện hoặc child2 xuất hiện

c) Khai báo Attribute:

<!ATTLIST element-name attribute-name attribute-type default-value>

- **Default value có thể là 1 trong các giá trị sau:**

- **value**: giá trị mặc định. Ví dụ:

  + DTD:

  <!ELEMENT square EMPTY>
  <!ATTLIST square width CDATA "0"> (“0” là giá trị mặc định)

  +Valid XML:

  <square width="100" />

- **#REQUIRED**: bắt buộc phải có giá trị. Ví dụ:

  + DTD:

  <!ATTLIST person number CDATA #REQUIRED>

+ *Valid XML:*

<person number="5677" />

+ *Invalid XML:*

<person />

- **#IMPLIED:** có thể có hoặc không attribute này. Ví dụ:

+ *DTD:*

<!ATTLIST contact fax CDATA #IMPLIED>

+ *Valid XML:*

<contact fax="555-667788" />

+ *Valid XML:*

<contact />

- **#FIXED**: giá trị attribute là hằng số, ví dụ:

+ *DTD:*

<!ATTLIST sender company CDATA #FIXED "Microsoft">

+ *Valid XML:*

<sender company="Microsoft" />

+ *Invalid XML:*

<sender company="W3Schools" />

- Khai báo tập giá trị cho attribute:

+ *DTD:*

<!ATTLIST payment type (check|cash) "cash">

+ *XML example:*

<payment type="check" />

Hay

<payment type="cash" />

Xét 1 ví dụ về việc khai báo 1 DTD:

Trong ví dụ này chúng ta xây dựng 1 DTD cho file XML lưu trữ thông tin tất cả sách trong 1 thư viện có các yêu cầu sau:

- Thẻ root: **thuVien**
- Trong thẻ root có ít nhất 1 thẻ <**sach**>

- Trong thẻ <sach> là các thẻ con theo thứ tự sau: **id, ten, tacGia, nhaXuatBan, gia**. Trong đó, **id, ten, tacGia** là những thẻ bắt buộc phải có, **nhaXuatBan, gia** xuất hiện tối đa **1 lần**
- Trong thẻ <**sach**> có 2 thuộc tính là **theLoai** và **ngonNgu**, trong đó giá trị của thuộc tính thể loại là một trong các giá trị sau: **Khoahọc**, **GiảiTrí**, **TinHọc**, thuộc tính **ngonNgu** có thể có hoặc không.
- Khai báo 2 **Entity** xuất hiện thường xuyên trong XML cho giá trị "**Nhà Xuất Bản Trẻ**" và "**Nhà Xuất Bản Giáo dục**"

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE thuVien [
    <!ELEMENT thuVien (sach+)>
    <!ELEMENT sach (id, ten, tacGia,nhaXuatBan?,gia?)>
    <!ELEMENT id (#PCDATA)>
    <!ELEMENT ten (#PCDATA)>
    <!ELEMENT tacGia (#PCDATA)>
    <!ELEMENT nhaXuatBan (#PCDATA)>
    <!ELEMENT gia (#PCDATA)>
    <!ATTLIST sach theLoai (Khoahọc|Giảitrí|Tinhọc) "Tinhọc">
    <!ATTLIST sach ngonNgu CDATA #IMPLIED>
    <!ENTITY nxbt "Nhà xuất bản Trẻ">
    <!ENTITY nxbgd "Nhà xuất bản Giáo dục">
]>
<thuVien>
    <sach theLoai="Khoahọc">
        <id>S00001</id>
        <ten>Lý thuyết đám mây</ten>
        <tacGia>O'Really</tacGia>
        <nhaXuatBan>&nxbt;</nhaXuatBan>
        <gia>100000</gia>
    </sach>
</thuVien>
```

1.3.2.4 DTD của Dublin Core

Cấu trúc file DTD của Dublin Core được định nghĩa:

(Liên kết URL: http://dublincore.org/documents/2002/07/31/dcmes-xml/dcmes-xml-dtd.dtd)

-dtd.dtd)

```
<!-- The namespaces for RDF and DCES 1.1 respectively -->
<!ENTITY rdfns 'http://www.w3.org/1999/02/22-rdf-syntax-ns#' >
<!ENTITY dcns 'http://purl.org/dc/elements/1.1/' >
<!-- Magic - do not look behind the curtain -->
<!ENTITY % rdfnsdecl 'xmlns:rdf CDATA #FIXED "&rdfns;"' >
```

```
<!ENTITY % dcnsdecl 'xmlns:dc CDATA #FIXED "&dcns;"' >
<!-- The wrapper element -->
<!ELEMENT rdf:RDF (rdf:Description)* >
<!ATTLIST rdf:RDF %rdfnsdecl; %dcnsdecl; >
<!ENTITY % dces "dc:title | dc:creator | dc:subject | dc:description |
dc:publisher | dc:contributor | dc:date | dc:type | dc:format |
dc:identifier | dc:source | dc:language | dc:relation | dc:coverage |
dc:rights" >
<!-- The resource description container element -->
<!ELEMENT rdf:Description (%dces;)* >
<!ATTLIST rdf:Description about CDATA #REQUIRED>
<!-- The name given to the resource. -->
<!ELEMENT dc:title (#PCDATA)>
<!-- An entity primarily responsible for making the content of the
resource. -->
<!ELEMENT dc:creator (#PCDATA)>
<!-- The topic of the content of the resource. -->
<!ELEMENT dc:subject (#PCDATA)>
<!-- An account of the content of the resource. -->
<!ELEMENT dc:description (#PCDATA)>
<!-- The entity responsible for making the resource available. -->
<!ELEMENT dc:publisher (#PCDATA)>
<!-- An entity responsible for making contributions to the content of
the resource. -->
<!ELEMENT dc:contributor (#PCDATA)>

<!-- A date associated with an event in the life cycle of the resource. -->
<!ELEMENT dc:date (#PCDATA)>
<!-- The nature or genre of the content of the resource. -->
```

```
<!ELEMENT dc:type (#PCDATA)>
<!-- The physical or digital manifestation of the resource. -->
<!ELEMENT dc:format (#PCDATA)>
<!-- An unambiguous reference to the resource within a given context. -->
<!ELEMENT dc:identifier (#PCDATA)>
<!-- A Reference to a resource from which the present resource is derived. -->
<!ELEMENT dc:source (#PCDATA)>
<!-- A language of the intellectual content of the resource. -->
<!ELEMENT dc:language (#PCDATA)>
<!-- A reference to a related resource. -->
<!ELEMENT dc:relation (#PCDATA)>
<!-- The extent or scope of the content of the resource. -->
<!ELEMENT dc:coverage (#PCDATA)>
<!-- Information about rights held in and over the resource. -->
<!ELEMENT dc:rights (#PCDATA)>
```

1.3.3 RDF

1.3.3.1 Khái niệm RDF

Resource Description Framework (RDF) là một Framework dùng để mô tả thông tin trên Web. RDF cung cấp một mô hình dữ liệu, và một cú pháp đơn giản sao cho các hệ thống độc lập có thể trao đổi và sử dụng nó. Đồng thời, nó được thiết kế sao cho hệ thống máy tính có thể hiểu được và có thể đọc được thông tin, chứ không phải chỉ để trình bày dữ liệu cho người dùng. Cú pháp của RDF dựa trên mô hình dữ liệu, và mô hình này ảnh hưởng đến cách thức mà những thuộc tính được mô tả và nó cũng làm cho cấu trúc của những mô tả đó trở nên rõ ràng. Điều này có nghĩa rằng RDF nó phù hợp cho việc mô tả tài nguyên web. Trong giới hạn của đề tài, RDF được sử dụng để mô tả quan hệ giữa tài nguyên và các thuộc tính.

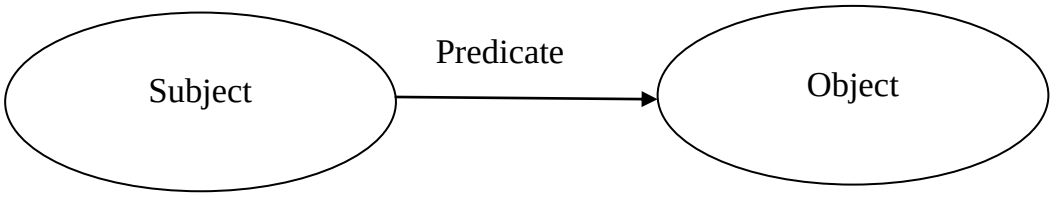
1.3.3.2 Cú pháp RDF

Để mô tả những thuộc tính của mọi thứ, thì cần phải có cách thức để đặt tên, hay xác định một số thứ như:

- Câu lệnh mô tả về cái gì (trong trường hợp này là trang web)
- Thuộc tính xác định (người tạo) của phát biểu này mô tả.
- Giá trị của thuộc tính của một phát biểu (người tạo là ai).

RDF sử dụng một thuật ngữ đặc biệt đó là bộ ba (tripple) để nói về những thành phần khác nhau của phát biểu. Trong đó một bộ ba có:

- Subject: thành phần này xác định cái gì mà phát biểu nói về (trong ví dụ là trang web), được gọi là chủ ngữ.
- Predicate: thành phần này xác định thuộc tính hay những đặc trưng của chủ ngữ của phát biểu xác định (người tạo, ngày tạo, hay ngôn ngữ trong ví dụ này), được gọi là vị từ.
- Object: thành phần này xác định giá trị của thuộc tính, được gọi là tân ngữ.



Hình 1.2 Bộ ba trong RDF

Xét một ví dụ: ta có một phát biểu

http://www.example.org/index.html has a creator whose value is John Smith.

Với phát biểu này thì những thuật ngữ RDF đối với những thành phần khác nhau của phát biểu là:

- Object là: URL *http://www.example.org/index.html.*
- Predicate là: *creator*.
- Object là: *John Smith*.

Tuy nhiên, phát biểu trên được viết ở ngôn ngữ tiếng Anh, nó có lợi trong việc giao tiếp giữa những người Anh, còn RDF thì tạo ra những phát biểu để máy có thể xử lý. Để làm những phát biểu này phù hợp cho việc xử lý bởi máy móc thì hai điều sau thực sự cần thiết:

- Một hệ thống với những định danh máy có khả năng xử lý để xác định chủ ngữ, vị từ và tân ngữ trong một phát biểu mà không có sự nhầm lẫn với những định danh trông có vẻ giống nhau mà được dùng bởi những người khác trên web.
- Một ngôn ngữ máy có thể xử lý để biểu diễn những phát biểu này và trao đổi giữa các máy với nhau.

URI:

Web cung cấp một ***mẫu định danh chung*** cho những mục đích này, được gọi là URI, URL là một loại đặc biệt của URI. Tất cả URI đều chia sẻ thuộc tính mà những người khác hay những tổ chức khác có thể tạo ra chúng một cách độc lập, và sử dụng chúng để xác định nhiều thứ. URI không bị hạn chế để xác định những thứ mà có những

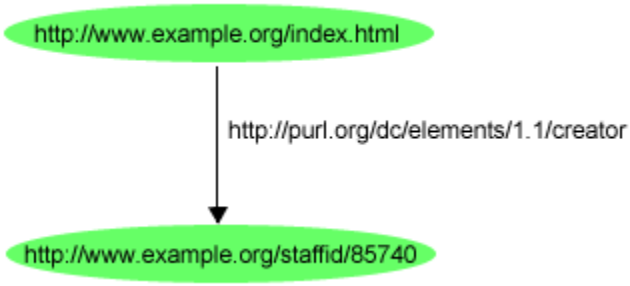
vị trí mạng hay sử dụng cơ chế truy cập máy tính khác. Thực ra, một URI có thể được tạo ra để tham chiếu tới bất cứ thứ gì mà cần thiết được đề cập tới trong một phát biểu, bao gồm:

- Những thứ có khả năng truy cập mạng, như một tài liệu điện tử, một hình ảnh, một dịch vụ (ví dụ dự báo thời thiết hôm nay cho Việt Nam), hay một nhóm những tài nguyên khác.
- Những thứ mà không có khả năng truy cập mạng, như những cuốn sách trong thư viện, những tập đoàn, con người.
- Những khái niệm trừu tượng mà không tồn tại một cách thực tế, như khái niệm của một "creator".

Bởi tính tổng quát này, RDF sử dụng những URI như nền tảng các cơ chế của nó để xác định những chủ ngữ, vị từ và tân ngữ trong những phát biểu. Như phần trước đã nói rằng RDF dựa trên ý tưởng của việc mô tả những phát biểu đơn giản về những tài nguyên, nơi mà mỗi phát biểu gồm có một chủ ngữ, vị từ và tân ngữ. Vì vậy trong RDF, phát biểu ở trên có dạng:

- Chủ ngữ: *http://www.example.org/index.html.*
- Vị từ: *http://purl.org/dc/elements/1.1/creator.*
- Tân ngữ: *http://www.example.org/staffid/85740.*

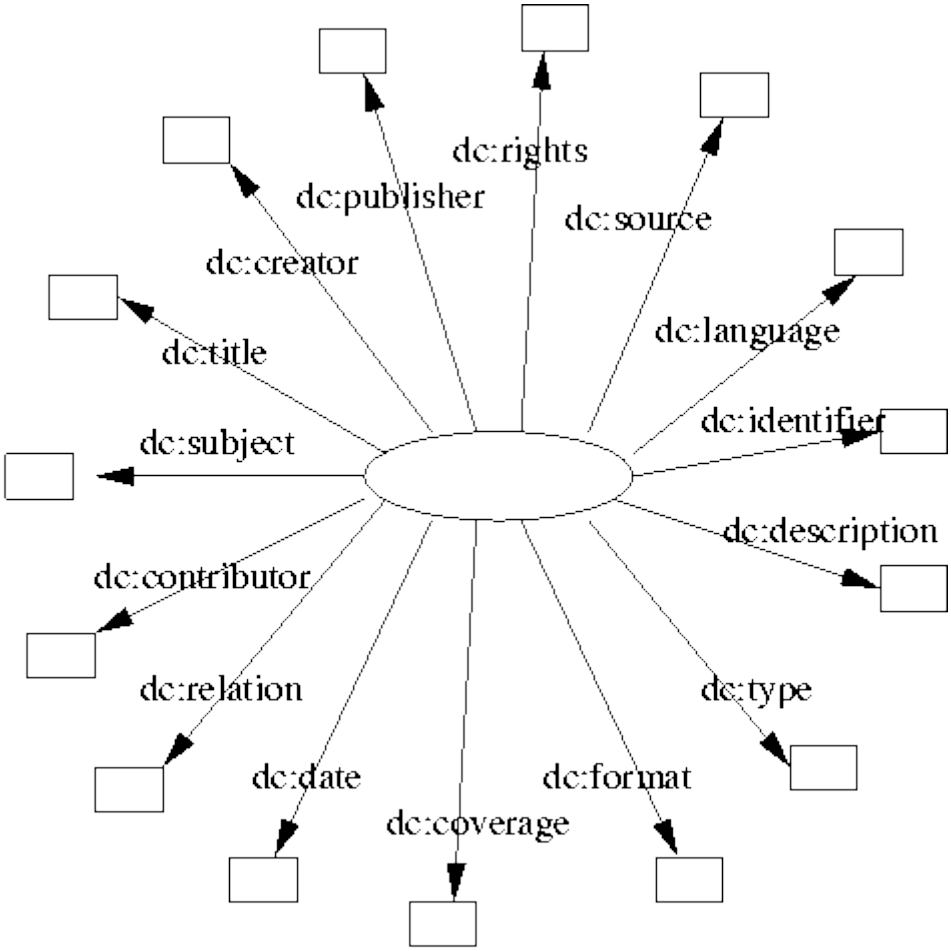
Chú ý URI được sử dụng như thế nào để xác định không chỉ chủ ngữ của phát biểu ban đầu, mà còn xác định vị từ và tân ngữ, thay vì sử dụng từ "*creator*" và "*John Smith*", một cách riêng biệt.



Hình 1.3 Một phát biểu RDF đơn giản

1.3.3.3 Mô hình RDF của Dublin Core

Mô hình RDF của Dublin Core:



Hình 1.4 Mô hình RDF của Dublin Core

Định nghĩa XML:

```
<?xml version="1.0"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
            xmlns:dc="http://purl.org/dc/elements/1.1/">
  <rdf:Description>
    <dc:creator>Karl Mustermann</dc:creator>
    <dc:title>Algebra</dc:title>
    <dc:subject>mathematics</dc:subject>
    <dc:date>2000-01-23</dc:date>
    <dc:language>EN</dc:language>
    <dc:description>An introduction to algebra</dc:description>
```

```
  </rdf:Description>
</rdf:RDF>
```

1.3.4 Các URI của chuẩn Dublin Core

DC	URI
Nhan đề (Title)	http://purl.org/dc/terms/title
Tác giả (Creator)	http://purl.org/dc/terms/creator
Đề mục (Subject)	http://purl.org/dc/terms/subject
Mô tả (Description)	http://purl.org/dc/terms/description
Xuất bản (Publisher)	http://purl.org/dc/terms/publisher
Tác giả phụ (Contributor)	http://purl.org/dc/terms/contributor
Ngày (Date)	http://purl.org/dc/terms/date
Loại tài liệu (Type)	http://purl.org/dc/terms/type
Mô tả vật lý (Format)	http://purl.org/dc/terms/format
Định danh (Indentifier)	http://purl.org/dc/terms/identifier
Nguồn gốc (Source)	http://purl.org/dc/terms/source
Ngôn ngữ (Language)	http://purl.org/dc/terms/language
Liên kết (Relation)	http://purl.org/dc/terms/relation
Nơi chứa (Coverage)	http://purl.org/dc/terms/coverage
Bản quyền (Rights)	http://purl.org/dc/terms/rights

Bảng 1.4 Các URI chuẩn của Dublin Core

1.3.5 Các bước tạo ra DCMES (Dublin Core Metadata Element Set) trong XML

(1) Khai báo phiên bản XML:

Hiện tại chỉ có một phiên bản XML được dùng là bản 1.0. Vì vậy khi thực hiện xây dựng một DCMES bằng XML cần khai báo <?xml version="1.0"?> ở dòng đầu tiền.

(2) Liên kết đến DTD của Dublin Core:

<!DOCTYPE rdf:RDF SYSTEM "http://purl.org/dc/schemas/dcmes-xml-20000714.dtd">

(3) Khai báo sử dụng RDF:

Việc khai báo sử dụng RDF là cần thiết vì sẽ giúp cho các chương trình có thể hiểu nghĩa của văn bản. Thêm đoạn sau vào dòng tiếp theo, sau phần khai báo liên kết DTD:

<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"

xmlns:dc= "http://purl.org/dc/elements/1.1/">

(4) Mô tả các tài nguyên:

Mỗi tài nguyên mô tả bởi các phần tử Dublin Core phải được đặt trong phần tử chứa – 1 cặp của thẻ rdf:Description. Mỗi phần tử sẽ chứa một tài nguyên được mô tả, các tài nguyên phải được định danh bởi các URI và mỗi URI sẽ được đặt trong thuộc tính about của phần tử rdf:Description:

<rdf:Description about="http://...">

……..

</rdf:Description>

Trong phần tử chứa rdf:Description đặt vào các phần tử Dublin Core với tiền tố dc ở đằng trước. Ví dụ phần tử Title sẽ được viết là dc:title:

<rdf:Description about="http://...">

<dc:title>Computer Network</dc:title>

</rdf:Description>

Ví dụ về

```
<?xml version="1.0" ?>
<!DOCTYPE rdf:RDF SYSTEM "http://purl.org/dc/schemas/dcmes-xml-20000714.dtd">
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:dc="http://purl.org/dc/elements/1.0/"
```

```
xmlns:dcq="http://purl.org/dc/qualifiers/1.0/">
<rdf:Description about="http://example.com/file/book1.pdf">
    <dc:type>text</dc:type>
    <dc:language>En<dc:language/>
    <dc:creator>Richard, John E.<dc:creator />
    <dc:title>Resource and environmental economics</dc:title>
    <dc:publisher>London : </dc:publisher>
    <dc:date>1995<dc:date/>
    <dc:format> <dc:format/>
    <dc:coverage> <dc:coverage/>
    <dc:relation> <dc:relation/>
    <dc:identifier> <dc:identifier/>
    <dc:rights <dc:rights/>
    <dc:description><dc:description/>
    <dc:subject>Environmental economics<dc:subject/>
</rdf:Description>
<rdf:RDF>
```

CHƯƠNG 2: ỨNG DỤNG CHUẨN DUBLIN CORE METADA TRONG TRIỂN KHAI THƯ VIỆN CUNG CẤP TÀI LIỆU CHUYÊN NGÀNH CÔNG NGHỆ THÔNG TIN

Việc sử dụng metadata trong việc triển khai các thư viện số hiện nay thực sự là vô cùng hữu ích và cần thiết. Sự hỗ trợ tìm kiếm cho người dùng khi sử dụng các thư viện số thực sự là một vấn đề được quan tâm. Dublin Core là chuẩn metadata đã được nghiên cứu và xây dựng phù hợp với nhu cầu triển khai các ứng dụng thư viện số, quản lý tài liệu,… Áp dụng metadata vào thư viện số sẽ hỗ trợ người dùng tìm kiếm tài liệu một cách hiệu quả, đồng thời sẽ là tiền đề để xây dựng hệ thống web thông minh, web ngữ nghĩa (semantic web). Dựa trên các phân tích bài toán thực tế, tác giả đưa ra các phân tích hệ thống và đặc tả dữ liệu như sau:

2.1 Các tác nhân của hệ thống

(1) Quản trị hệ thống: là người quản trị hệ thống, quản lý việc cập nhật thông tin cho các tài liệu. Các ca sử dụng của người quản trị: Đăng nhập, đăng xuất, quản lý mục tài liệu, quản lý tài liệu.

(2) Người dùng: là người dùng sử dụng trực tiếp các chức năng cung cấp của hệ thống thư viện. Các ca sử dụng của người dùng: xem tài liệu, tìm kiếm tài liệu, download tài liệu.

2.2 Biểu đồ ca sử dụng Usecase

Danh sách các usecase:

(1) Usecase đăng nhập: Quản trị cung cấp cho hệ thống thông tin mật khẩu và tên đăng nhập để hệ thống xem xét việc đăng nhập có thành công hay không.

(2) Usecase đăng xuất: Quản trị thoát khỏi hệ thống và hủy trạng thái hiện hành trong hệ thống.

(3) Usecase quản lý mục tài liệu: Quản trị cung cấp thông tin về các mục tài liệu như tên mục tài liệu, thông tin,… để hệ thống cập nhật các mục tài liệu hay loại bỏ, sửa đổi thông tin các mục tài liệu.

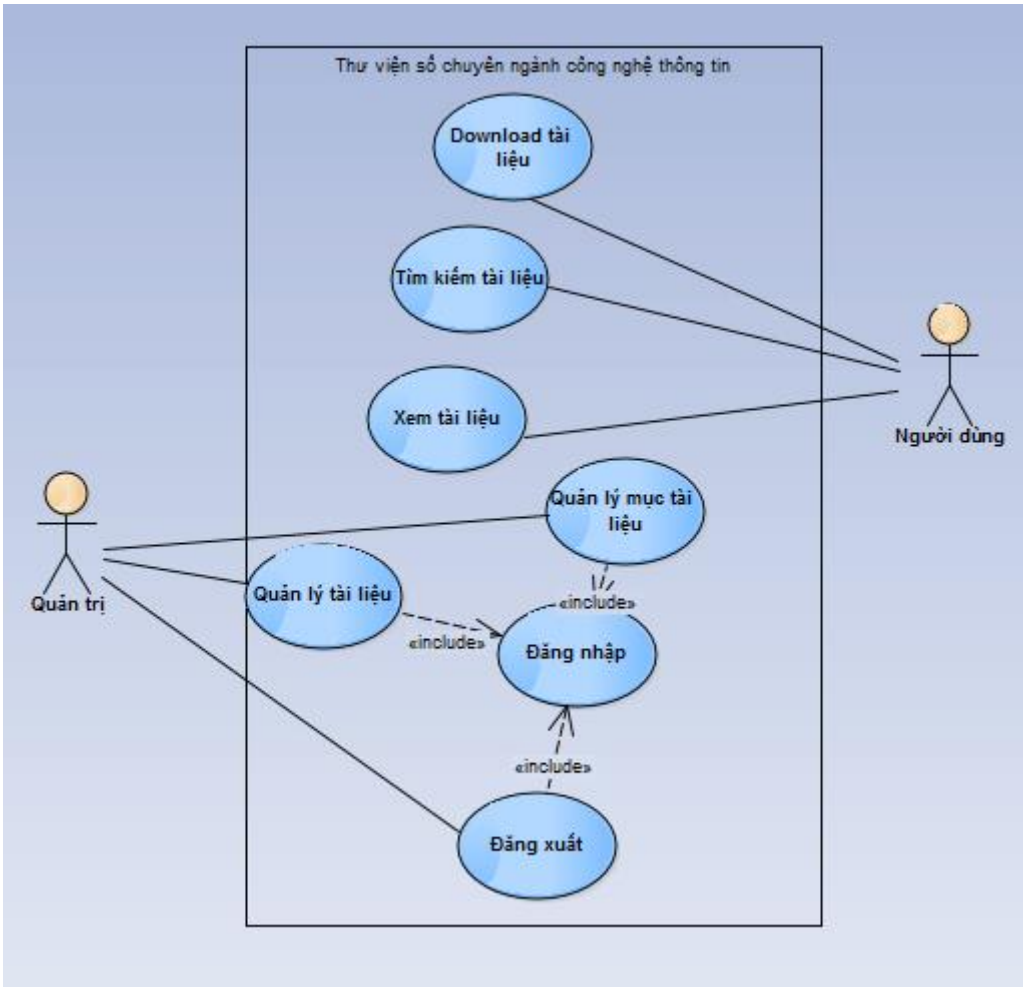
***(4) Usecase quản lý tài liệu*:** Quản trị cung cấp thông tin về các tài liệu như tên tài liệu, tác giả, mục tài liệu, ngày xuất bản,… để hệ thống cập nhật các tài liệu hay loại bỏ, sửa đổi thông tin các tài liệu.

(5) Usecase xem tài liệu: Người dùng sẽ đưa ra yêu cầu về tài liệu cần xem (click vào link tài liệu), hệ thống sẽ trả về kết quả hiển thị thông tin tài liệu.

(6) Usecase tìm kiếm tài liệu: Người dùng cung cấp các thông tin về tài liệu cần tra cứu theo các tiêu chí khác nhau để tìm ra thông tin tài liệu mong muốn.

(7) Usecase download tài liệu: Người dùng sẽ đưa ra yêu cầu download tài liệu, hệ thống sẽ trả về thông tin có download thành công hay không.

Biểu đồ ca sử dụng:



Hình 2.1 Biểu đồ ca sử dụng của hệ thống

2.3 Đặc tả dữ liệu hệ thống

Với mục đích xây dựng các ca sử dụng cho các tác nhân như phân tích ở trên, tác giả đưa ra đặc tả dữ liệu của hệ thống như sau:

(1) Phần siêu dữ liệu lưu thông tin tài liệu

Xây dựng trên file XML, sử dụng chuẩn Dublin Core gồm các phần:

a) File DTD của Dublin Core: với cấu trúc như sau

```
<!-- The namespaces for RDF and DCES 1.1 respectively -->
<!ENTITY rdfns 'http://www.w3.org/1999/02/22-rdf-syntax-ns#' >
<!ENTITY dcns 'http://purl.org/dc/elements/1.1/' >
<!-- Magic - do not look behind the curtain -->
```

```
<!ENTITY % rdfnsdecl 'xmlns:rdf CDATA #FIXED "&rdfns;"' >
<!ENTITY % dcnsdecl 'xmlns:dc CDATA #FIXED "&dcns;"' >
<!-- The wrapper element -->
<!ELEMENT rdf:RDF (rdf:Description)* >
<!ATTLIST rdf:RDF %rdfnsdecl; %dcnsdecl; >
<!ENTITY % dces "dc:title | dc:creator | dc:subject | dc:description |
dc:publisher | dc:contributor | dc:date | dc:type | dc:format |
dc:identifier | dc:source | dc:language | dc:relation | dc:coverage |
dc:rights" >
<!-- The resource description container element -->
<!ELEMENT rdf:Description (%dces;)* >
<!ATTLIST rdf:Description about CDATA #REQUIRED>
<!-- The name given to the resource. -->
<!ELEMENT dc:title (#PCDATA)>
<!-- An entity primarily responsible for making the content of the
resource. -->
<!ELEMENT dc:creator (#PCDATA)>
<!-- The topic of the content of the resource. -->
<!ELEMENT dc:subject (#PCDATA)>
<!-- An account of the content of the resource. -->
<!ELEMENT dc:description (#PCDATA)>
<!-- The entity responsible for making the resource available. -->
<!ELEMENT dc:publisher (#PCDATA)>
<!-- An entity responsible for making contributions to the content of
the resource. -->
<!ELEMENT dc:contributor (#PCDATA)>

<!-- A date associated with an event in the life cycle of the resource. -->
<!ELEMENT dc:date (#PCDATA)>
```

```
<!-- The nature or genre of the content of the resource. -->
<!ELEMENT dc:type (#PCDATA)>
<!-- The physical or digital manifestation of the resource. -->
<!ELEMENT dc:format (#PCDATA)>
<!-- An unambiguous reference to the resource within a given context. -->
<!ELEMENT dc:identifier (#PCDATA)>
<!-- A Reference to a resource from which the present resource is derived. -->
<!ELEMENT dc:source (#PCDATA)>
<!-- A language of the intellectual content of the resource. -->
<!ELEMENT dc:language (#PCDATA)>
<!-- A reference to a related resource. -->
<!ELEMENT dc:relation (#PCDATA)>
<!-- The extent or scope of the content of the resource. -->
<!ELEMENT dc:coverage (#PCDATA)>
<!-- Information about rights held in and over the resource. -->
                    <!ELEMENT dc:rights (#PCDATA)>
```

b) File XML lưu thông tin tài liệu tailieu.xml

```
<?xml version="1.0" ?>
<!DOCTYPE rdf:RDF SYSTEM "http://purl.org/dc/schemas/dcmes-xml-
20000714.dtd">
<rdf:RDF      xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:dc="http://purl.org/dc/elements/1.0/"
xmlns:dcq="http://purl.org/dc/qualifiers/1.0/">
<rdf:Description about="http://example.com/file/book1.pdf">
    <dc:type>text</dc:type>
    <dc:language>En<dc:language/>
    <dc:creator>Richard, John E.<dc:creator />
    <dc:title>Resource and environmental economics</dc:title>
    <dc:publisher>London : </dc:publisher>
    <dc:date>1995<dc:date/>
```

```
    <dc:format> <dc:format/>
    <dc:coverage> <dc:coverage/>
    <dc:relation> <dc:relation/>
    <dc:identifier> <dc:identifier/>
    <dc:rights <dc:rights/>
    <dc:description><dc:description/>
    <dc:subject>Environmental economics<dc:subject/>
</rdf:Description>
<rdf:RDF>
```

(2) Phần lưu thông tin các danh mục

Xây dựng trên cơ sở dữ liệu Mysql với các bảng dữ liệu:

a) Bảng danh mục tài liệu: *lưu thông tin các loại danh mục tài liệu trên hệ thống*

Tên trường	Kiểu dữ liệu	Mô tả
Id	Int	Mã danh mục tài liệu
Ten_danh_muc	Varchar	Tên danh mục tài liệu
Ghi_chu	Text	Thông tin ghi chú

b) Bảng tài khoản: *lưu thông tin tài khoản trên hệ thống*

Tên trường	Kiểu dữ liệu	Mô tả
Id	Int	Mã tài khoản
Tai_khoan	Varchar	Tài khoản
Mat_khau	Varchar	Mật khẩu
Ho_ten	Varchar	Họ tên
Trang_thai	Int	Trạng thái
Ma_quyen	Int	Mã quyền

c) ***Bảng mã quyền:*** *lưu thông tin quyền trên hệ thống*

Tên trường	Kiểu dữ liệu	Mô tả
Id	Int	Mã quyền
Ten_quyen	Varchar	Tên quyền

Trang_thai	Int	Trạng thái

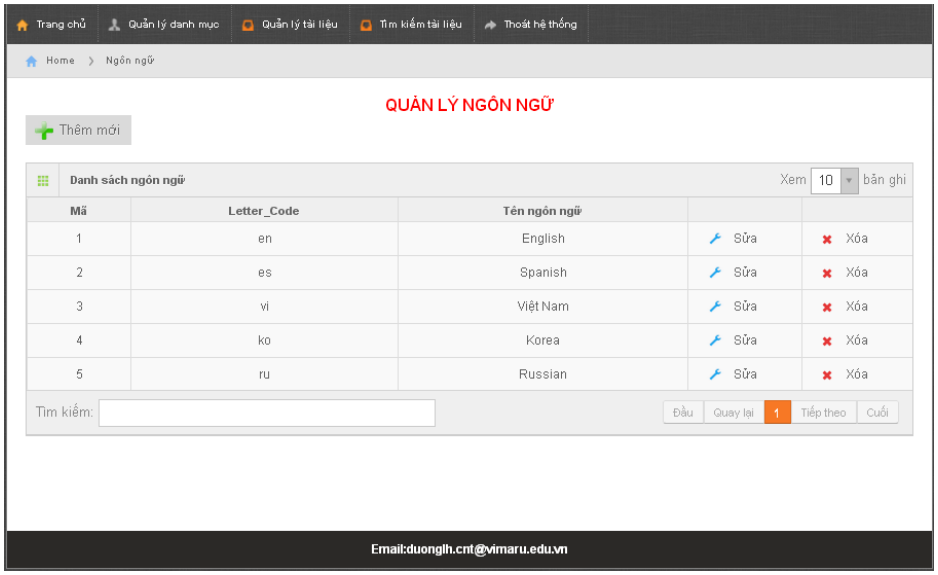
d) Bảng ngôn ngữ: *lưu thông tin bảng mã ngôn ngữ chuẩn sử dụng cho tài liệu*

Tên trường	Kiểu dữ liệu	Mô tả
Id	Int	Mã quyền
Letter_code	Varchar(2)	Mã viết tắt
Ten_ngon_ngu	Varchar	Tên ngôn ngữ

2.4 Kết quả cài đặt thử nghiệm:

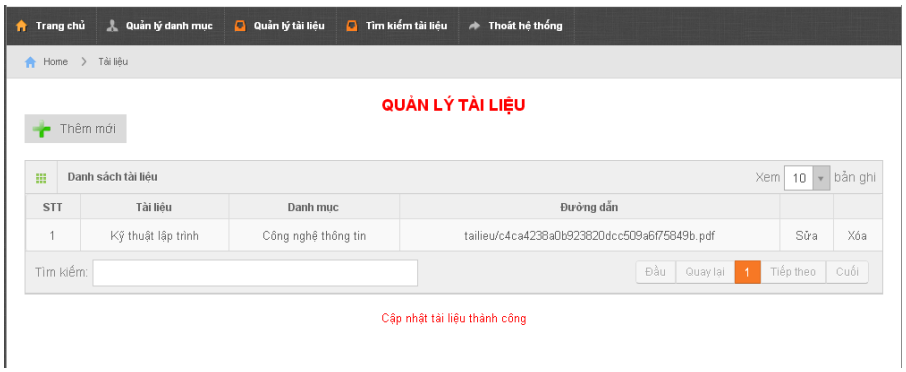
Một số giao diện của hệ thống được cài đặt thử nghiệm:

2.4.1 Giao diện trang quản lý



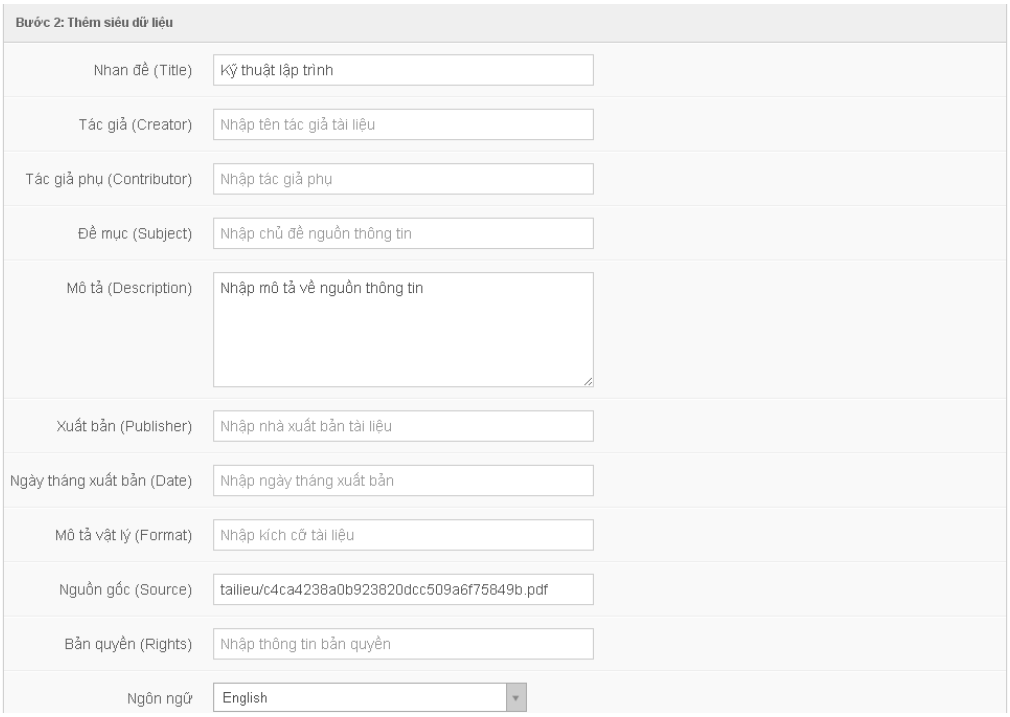
Hình 2.2 Giao diện trang quản lý hệ thống

2.4.2 Giao diện quản lý danh sách tài liệu



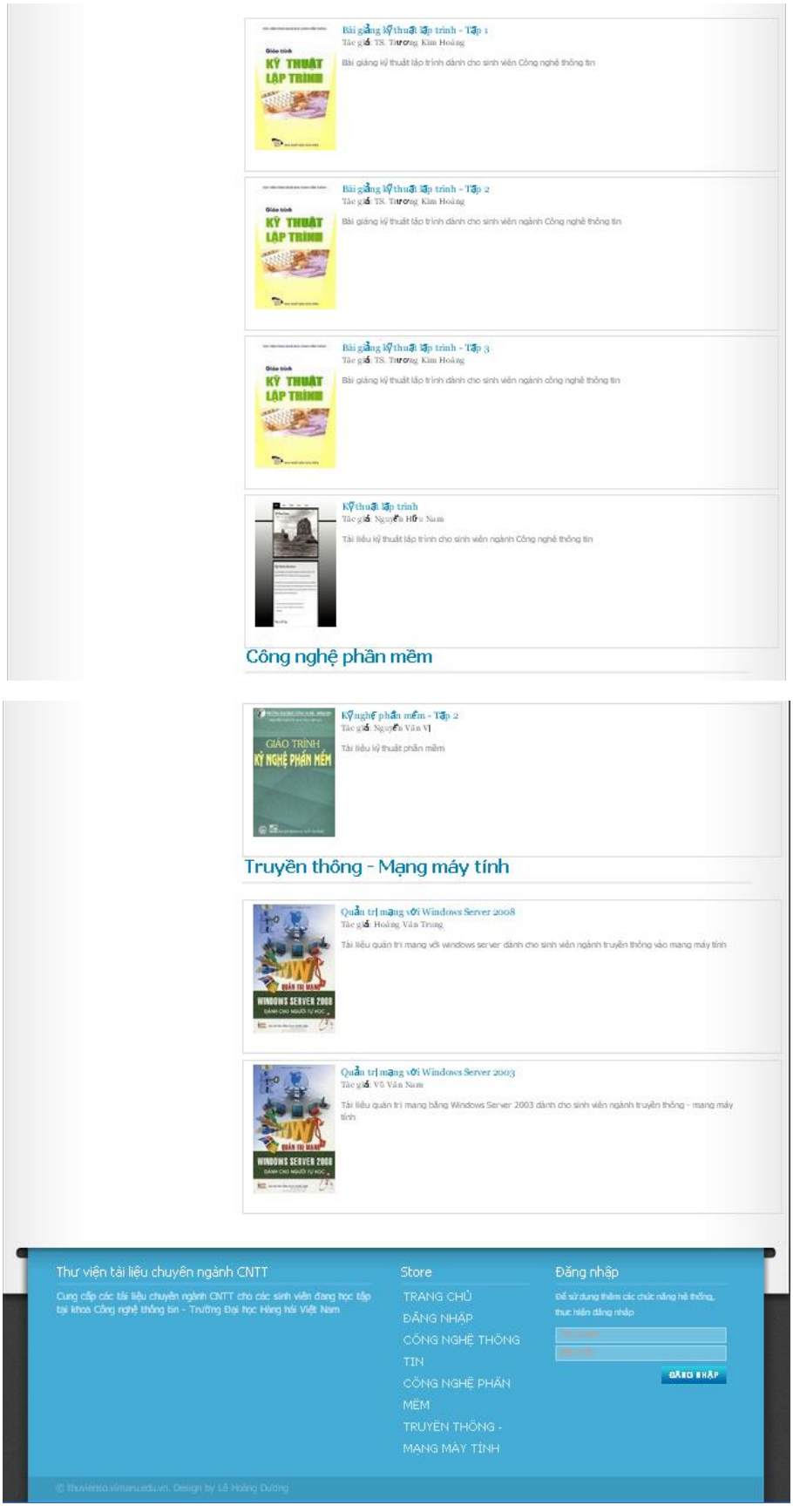
Hình 2.3 Giao diện quản lý danh sách tài liệu

2.4.3 Giao diện thêm siêu dữ liệu cho tài liệu



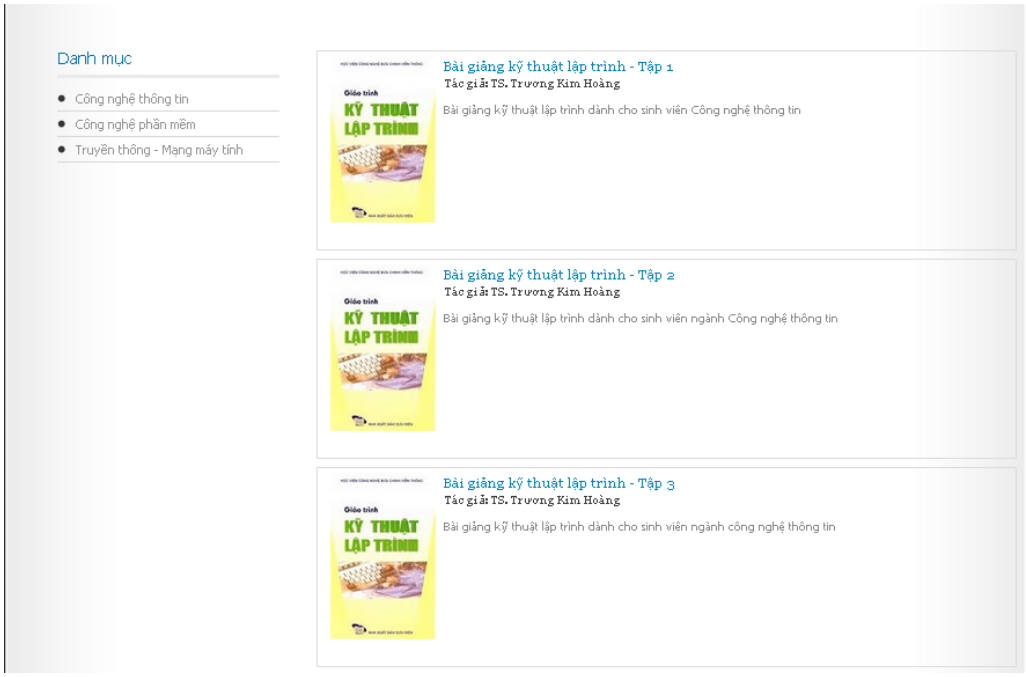
Hình 2.4 Giao diện thêm siêu dữ liệu cho tài liệu

2.4.4 Giao diện trang chủ hệ thống



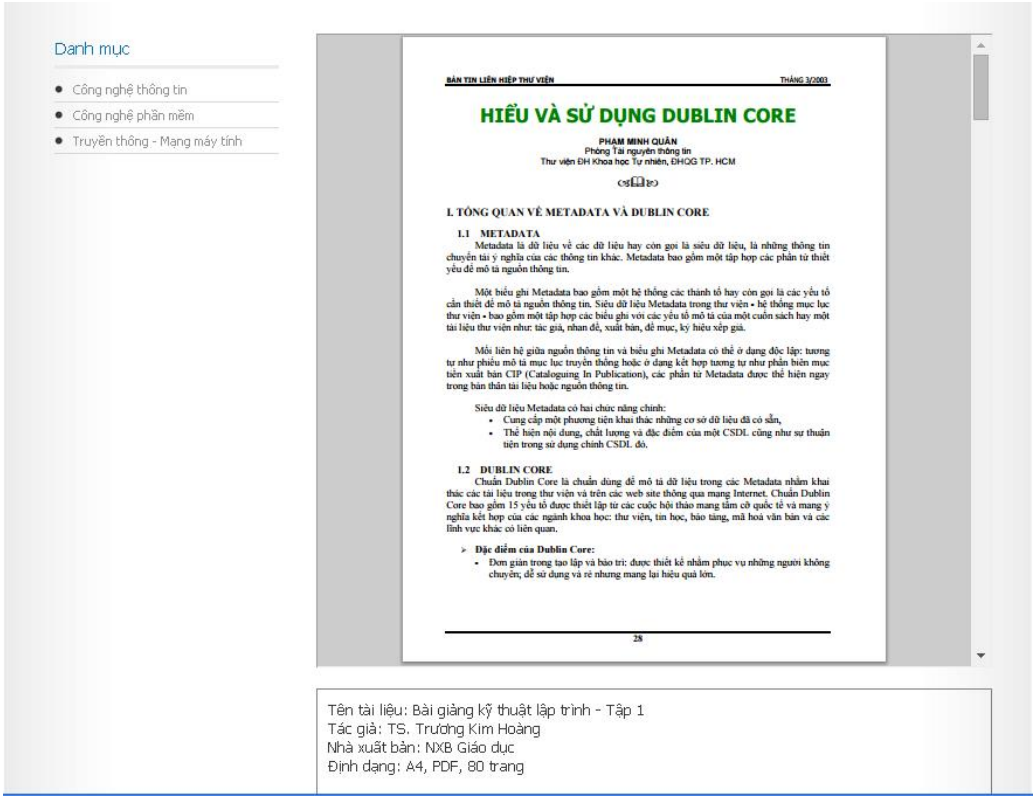
Hình 2.5 Giao diện trang chủ hệ thống

2.4.5 Giao diện danh sách tài liệu một số chuyên ngành



Hình 2.6 Giao diện danh sách tài liệu trong 1 chuyên ngành

2.4.6 Giao diện trang xem tài liệu



Hình 2.7 Giao diện trang xem tài liệu

KẾT LUẬN

Đề tài **Nghiên cứu chuẩn Dublin Core Metadata, ứng dụng xây dựng giải pháp thư viện số cung cấp tài liệu chuyên ngành cho Khoa Công nghệ thông tin – Trường Đại học Hàng hải Việt Nam** là đề tài có tính thực tế, sau thời gian nghiên cứu, tác giả đã tìm hiểu và nắm bắt được chuẩn Dublin Core và ứng dụng xây dựng thư viện số. Tuy nhiên do giới hạn của nội dung đề tài nên các chức năng của đề tài còn ở mức đơn giản, chủ yếu nhằm mục đích thể hiện cách tổ chức và ứng dụng chuẩn siêu dữ liệu Dublin Core trong hệ thống tài liệu số. Kết quả của nghiên cứu sẽ là cơ sở để tác giả tiếp tục phát triển hệ thống theo hướng Web ngữ nghĩa trong thời gian tới nhằm xây dựng hoàn thiện được hệ thống, phục vụ được cho nhu cầu thực tế đang đặt ra.