

디지털 특수자료를 위한 XML 스키마 기반의 메타데이터 표현 체계*

A Metadata Representation Scheme based on XML Schema
for Special Digital Collections

오 삼 군(Sam-Gyun Oh)*

채 진 석(Jin-Seok Chae)**

초 록

정보자원의 전달 매체와 형태가 다양화됨에 따라서 이에 대한 관리방법 또한 다양화되어 왔다. 도서관 환경에서는 정보자원을 위한 관리방법으로서 AACR, KCR 등의 목록규칙이 정립되었으며 이러한 목록규칙에 근거한 정보자원관리를 자동화하고자 하는 노력의 결과로서 MARC가 개발되었다. 하지만, MARC 레코드는 서지 레코드가 지니고 있는 의미적 관계의 표현을 지원하지 못하는 구조적 경직성으로 인해 다양하고 상이한 기술적 특성을 지니는 정보자원들을 적절히 기술하는데 제약이 따른다. 즉, MARC의 기본 설계 목적이 몇몇 정보유형에는 비교적 적합하더라도 새로운 형태의 정보유형의 다양성을 지원하는데 어려움이 있다. 또한 MARC를 활용한 정보자원 관리 방식에서는 정보자원 간 연결 관계의 표현을 지원하지 못한다. 즉, MARC의 데이터 모델은 자원기술의 대상을 단일의 객체로 파악하는 단층 데이터 모델이기 때문에 여러 객체들 간의 연결 관계를 설정할 수 있는 다층 데이터 모델을 이용한 정보자원 기술이 필요한 경우는 적절치 못하다. 본 연구에서는 다층 데이터 모델을 지원하는 IFLA FRBR 기본 모델을 기초로 하여 전자도서관에서 사용되는 고서, 고문서, 음악 자료, 학술회의 및 세미나 자료의 관리에 있어서 이용자의 정보요구를 최대한 수용할 수 있는 최적의 메타데이터 모델과 이에 대한 XML 스키마 기반의 표현 체계를 제시하고자 한다.

ABSTRACT

As there are diverse delivery media and forms of information resources, their management schemes are diverse as well. In library community, cataloguing rules for describing information resources such as AACR and KCR have been developed. The efforts to automate management of information resources based on these rules resulted in the development of MARC. However, MARC records are restricted in describing the information resources and MARC has various and distinct characteristics of the structural rigidity, which does not support the representation of extended semantic structures that exist among bibliographic entities. Therefore, since the data model for MARC is single-layer data model, it is not appropriate for describing information resources represented by multi-layer data model which can be used to set up the relationships among various objects in digital libraries. In this paper, we propose an a metadata model for digital libraries based on the IFLA FRBR basic model which supports multi-layer data model and a representation scheme based on XML Schema to manage the metadata about old books, old documents, resource related to music, conferences and seminars.

키워드: 다층 데이터 모델, 전자도서관, IFLA FRBR 기본 모델, XML 스키마
multi-layer data model, digital library, IFLA FRBR model, XML Schema

* 본 연구는 서울대학교 디지털도서관 프로젝트의 지원으로 추진되었음.

** 성균관대학교 문헌정보학과 부교수(samoh@skku.ac.kr)

*** 인천대학교 컴퓨터공학과 부교수(jschae@incheon.ac.kr)

■ 논문접수일자 : 2004년 11월 13일

■ 게재확정일자 : 2004년 12월 19일

1. 서론

정보자원의 전달 매체와 형태가 다양화 됨에 따라서 이에 대한 관리방법 또한 다양화되어 왔다. 도서관 환경에서는 정보자원을 위한 관리방법으로서 AACR2(Anglo-American Cataloguing Rules 2), KCR(Korea Cataloguing Rules) 등의 목록규칙이 정립되었으며 이러한 목록규칙에 근거한 정보자원관리를 자동화하고자 하는 노력의 결과로서 MARC(Machine Readable Cataloging)가 개발되었다. 현재 대다수의 도서관에서의 정보자원 관리시스템은 그 기반 기술 체계로서 MARC를 활용하고 있다. MARC가 그 뿌리로 삼고 있는 목록규칙 체계는 전통적인 도서관에서 관리하던 서적형태의 정보자원에 대한 상세한 기술을 그 기본 목적으로 한다.

하지만, MARC 레코드는 서지 레코드가 지니고 있는 의미관계의 표현을 지원하지 못하는 구조적 경직성으로 인해 다양하고 상이한 기술적 특성을 지니는 정보자원들을 적절히 기술하는데 제약이 따른다. 즉, MARC의 기본 설계 목적이 몇몇 정보유형에는 비교적 적합하더라도 새로운 형태의 정보유형의 다양성을 지원하는데 어려움이 있다. 또한 MARC를 활용한 정보자원 관리 방식에서는 정보자원 간 연결 관계의 표현을 지원하지 못한다. 즉, MARC의 데이터 모델은 자원기술의 대상을 단일 객체로 파악하는 단층 데이터 모델이기 때문에 여러 객체들 간의 연결 관계를 설정할 수 있는 다층 데이터 모델을 이용한 정보자원 기술이 필요한 경우는 적절치 못하다(조재인 2003).

이러한 MARC 데이터 모델의 기본적인 한계점을 극복하고자 이용자 서비스 지향적인 21세기형 서지관리 모델을 제시하기 위해서 IFLA(International Federation of Librarians Associations and Institutions)에서는 FRBR(Functional Requirements for Bibliographic Records) 모델을 제시하였다(IFLA, 1998). 목록의 첫째 목적은 찾고자 하는 자원을 제대로 검색할 수 있도록 하는 것이고, 둘째 목적은 검색된 자원 중에서 서로 연관성이 있는 정보를 그룹해서 보여 줄 수 있어야 한다는 것이다. 현존하는 목록 시스템 중에서 목록의 둘째 목적을 제대로 구현한 사례는 찾아보기 힘들며 IFLA FRBR 모델은 특히 이 둘째 목적을 충실히 달성하려는 노력의 일환으로 이루어진 것이다(Hickery 2002, Bearman 1999).

본 연구의 목적은 이러한 IFLA FRBR 기본 모델을 기초로 하여 고서, 고문서, 음악 자료, 학술회의/세미나 자료의 관리에 있어서 이용자의 정보요구를 최대한 수용할 수 있는 최적의 메타데이터 모델과 이에 대한 XML 표현체계를 제시하고자 한다. IFLA의 FRBR 모델을 기반으로 메타데이터를 설계함으로써 목록의 제2목적의 성취하는데 한 걸음 더 전진할 수 있을 것으로 기대하고 있으며, 또한 모든 메타데이터 요소와 어휘에 XML 네임스페이스를 활용한 URI 체제를 확립함으로써 시맨틱 웹 기반 디지털 도서관의 기반기술을 견고히 다지는 한 예제를 제공하게 될 것이다.

2. 디지털 특수자료를 위한 메타데이터 모델

서울대학교 전자도서관 구축을 위한 메타데이터 연구는 2단계로 구성되어 있다. 첫 번째 단계는 메타데이터 핵심 요소를 추출하는 과정으로, 이 단계에서는 서울대학교에서 보유하고 있는 정보자원을 조사하여, 메타데이터 개발에 적용할 적절한 분류과정을 거쳐 대상 자료유형을 선정한 후, 해당 정보원의 특성을 잘 반영할 수 있는 요소를 추출하는 작업이 포함되었다.

이와 같은 과정을 통하여 추출된 메타데이터 요소는 메타데이터를 저장하고 교환하기 위해서 데이터 전송의 표준 포맷으로 인정을 받은 XML로서 표현 및 저장되며, 메타데이터 교환시 데이터 무결성 문제의 해결을 위해서 XML 스키마를 기반으로 코딩한다. XML 스키마에서는 이름공간(namespace)을 지원하므로, 서울대학교에서 개발한 메타데이터 요소를 이름공간으로 관리할 수 있다(Powell 2001, Beckett 2002). 메타데이터 연구의 2 단계에서는 1단계에서 획득된 경험을 바탕으로

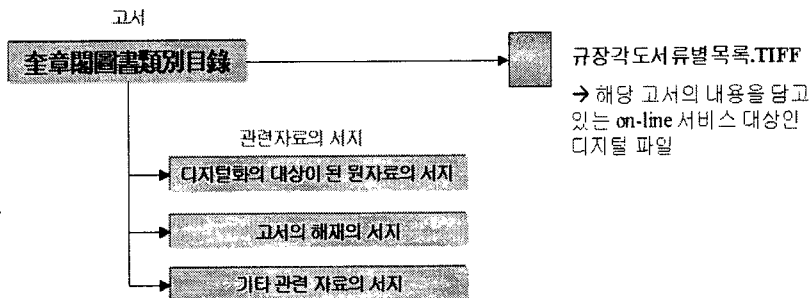
로 메타데이터 개발이 필요한 모든 정보자원 에 대한 메타데이터 핵심 및 확장 요소를 추출한다.

이외에도, 새롭게 조합되고 구성된 서울대학교 메타데이터 표준안내에 포함된 메타데이터 요소들의 의미가 상호 충돌하지 않고 표현될 수 있도록 하기 위하여, 디지털 도서관 소프트웨어 개발 시에 RDF/XML 이용하여 메타데이터를 표현한다면 데이터의 의미적 상호운용성은 더욱 증진될 것이다.

2.1 단층 데이터 모델 유형

2.1.1 고서

고서를 위한 메타데이터의 설계 방향은 KOMARC으로는 적절히 기술하기 어려운 기술 항목을 수용하기 위하여 고서 특유의 서명 관련 정보의 기술 요구사항을 수용하며, 고서 특성을 반영한 스킴 체계를 구축하고, 관련된 다른 정보 자원(해제, 관련자료)과의 연계정보를 구축할 수 있는 토대를 마련하는 것이다. <그림 1>은 고서 이용자 서비스 흐름도를 보여주고 있다.



<그림 1> 고서 이용자 서비스 흐름도

2. 1. 2 고문서

고문서 메타데이터 설계의 방향은 고서와 마찬가지로 KOMARC으로는 적절히 기술하기 어려운 기술 항목을 수용하기 위하여, 고문서 특유의 저자 관련 정보의 기술 요구사항을 수용하고, 고문서의 특성을 반영한 스킴 체계를 구축하며, 관련된 다른 정보 자원(해제, 성책고문, 딸림자료 등)과의 연계정보를 구축할 수 있는 토대를 마련하는 것이다. <그림 2>는 고문서 이용자 서비스 흐름도를 보여주고 있다.

2. 2 다층 데이터 모델 유형

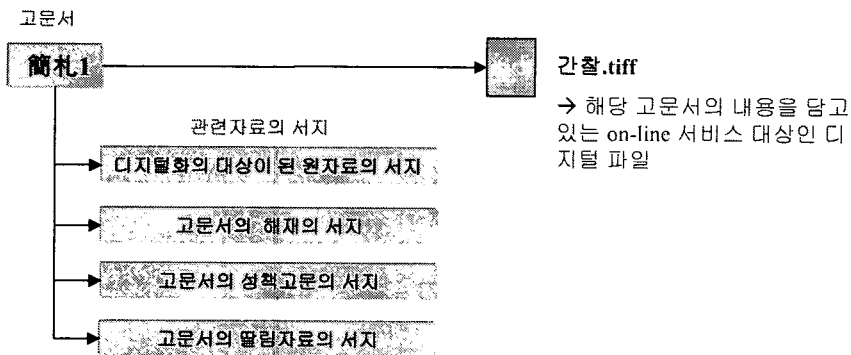
2. 2. 1 학술회의, 세미나, 강연회 및 전시회 관련 자료

학술회의, 세미나, 강연회 및 전시회 등의 자료에 대한 메타데이터 설계의 방향은 KOMARC와 같은 단층 데이터 모델의 한계성을 극복하기 위하여, 학술회의나 세미나에서 발생하는 정보자원(Article)들의 특성에 적합한

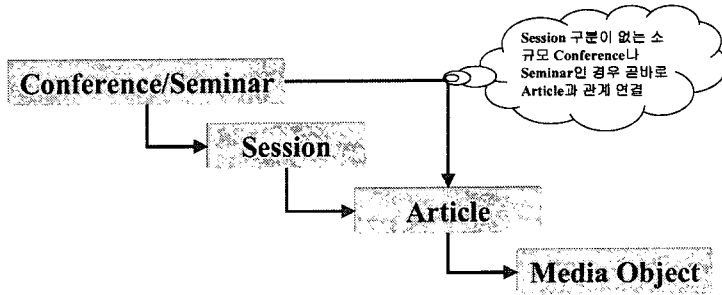
메타데이터 모델 개발과 기술 요소의 선정하여 체계적인 자료 관리 및 이용자 정보서비스의 토대를 마련하는 것이다. 또한, 학술회의/세미나의 세션(Session), 발표 자료(Article), 컴퓨터 파일(Media Object)을 별도의 기술 대상으로 관리하고 이들 기술대상간의 연결 관계를 기술하는 것을 가능하게 한다. 즉, 다양한 학술관련 정보자원에 대하여 “학술/세미나”를 기준으로 하여 “세션” > “발표자료” > “컴퓨터 파일”의 순서로 하향식으로 설계하여, 실제 이용자 정보 서비스의 대상이 되는 컴퓨터 파일에 대한 서지 관리에 있어서 목록의 집중 기능을 제고하고자 한다.

<그림 3>은 학술회의, 세미나, 강연회 및 전시회 등의 자료에 대한 메타데이터 모델을 보여주고 있고, <그림 4>는 학술회의, 세미나, 강연회 및 전시회 등의 자료에 대한 이용자 서비스 흐름도를 보여주고 있다. <그림 3>에서 사용된 용어의 의미는 다음과 같다.

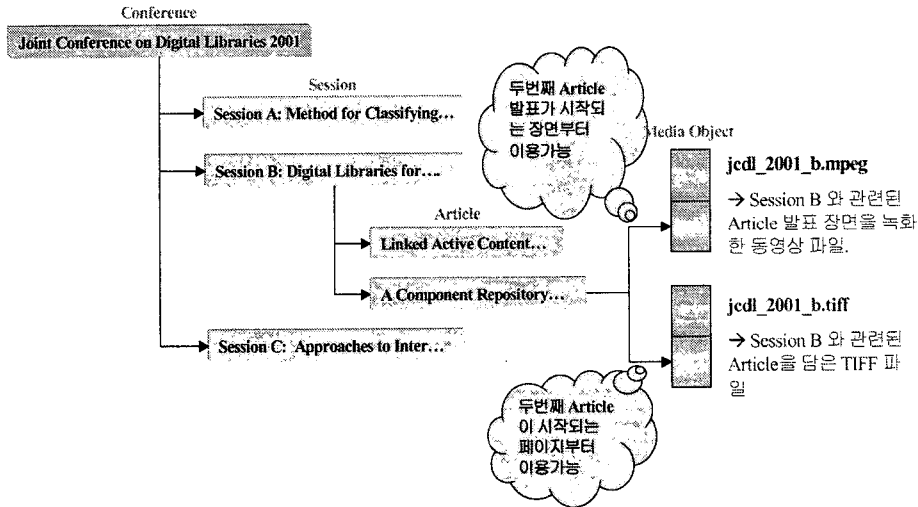
- Conference/Seminar: 발표 자료가 발생할 수 있는 각종 학술회의
- Session: 규모가 큰 학술회의인 경우 관



<그림 2> 고문서 이용자 서비스 흐름도



〈그림 3〉 학술회의, 세미나, 강연회 및 전시회 등의 메타데이터의 모델



〈그림 4〉 학술회의, 세미나, 강연회 및 전시회 등의 자료 이용자 서비스 흐름도

런 주제별로 나누어진 세부 분류

- Article: 학술회의/세미나에서 발표되는 논문이나 발표문
- Media Object: 발표 자료를 디지털화한 파일

〈그림 3〉의 모델의 설명하자면, 하나의 컨퍼런스나 세미나는 여러 세션(session)으로 나눌 수 있으나 각 세션은 반드시 한 컨퍼런스나 세미나에 속해야 한다. 각 세션에서 다수의 논문이 발표될 수 있으나 한 논문발표는 반드시

한 세션에서만 가능하다. 마지막으로 한 논문은 다양한 컴퓨터 파일 형태로 저장될 수 있도록 설계되었다.

〈그림 4〉의 서비스 흐름도는 이용자가 먼저 관심있는 컨퍼런스나 세미나를 선택하면, 세션 정보를 보여주고, 특정 세션을 선정하면, 그 세션에서 발표된 논문제목들이 열거되고, 이용자가 관심있는 한 논문을 선택하면, 그 논문에 대한 접근 가능한 파일을 유형별로 사용할 수 있다는 점을 보여 주고 있다.

2. 2. 2 음악 자료

음악 자료 메타데이터 설계의 방향은 학술 회의, 세미나, 강연회, 전시회 등의 자료와 마찬가지로 MARC와 같은 단층 데이터 모델의 한계성을 극복하는 것이다. 이를 위하여 IFLA FRBR의 기본 데이터 모델을 기본적으로 수용하고 필요한 부분을 수정하여 특정 작품과 관련된 다양한 정보자원에 대한 목록의 집중 기능을 제고하고, 음악과 관련된 다양한 형태의 정보자원과 자원 간의 관계성을 규정하고 이러한 관계성을 포함하는 메타데이터 모델을 구현하여 음악분야 학술 연구자가 간편하고 신속하게 연구수행에 필요한 자료를 수집할 수 있도록 음악분야 디지털 도서관서비스 구축을 위한 새로운 토대를 마련하고자 한다. 이러한 음악 자료 메타데이터 모델에서의 기술 대상은 다음과 같다.

(1) 저작(Work)

- 기술 대상은 추상적인 개념의 음악 작품 자체이다. 저작은 추상적인 내용 그 자체를 지칭하는 것이기 때문에 어떤 물리적인 형태를 취하고 있는 것은 '저작'이 아닌 점을 유의해야 한다.
- '저작'을 작품에 대한 연주, 녹음, 악보 혹은 물리적인 표현물 자체와 혼동하면 안 된다.
- 베토벤 5번 교향곡에 관한 음악작품은 여러 가지 표현방법(Text, Score, Audio, Visual)으로 전달될 수 있다.
- 이러한 여러 형태의 표현물의 공통적인 특성들은 '저작' 단계에서 기술하고 이러한 여러 종류의 표현물(Instantiation)과

'저작'과의 연결 관계를 설정함으로써 동일 콘텐츠에 대한 여러 표현물들에 대한 개별적이 아닌 그룹화된 형태의 이용자 서비스가 가능하게 된다. 예) "Phantom 오페라" 작품 자체

(2) 저작구조의 선언(Work Structure Declaration)

- '저작'은 내부적인 하부구조를 가질 수 있으며 이러한 하부구조를 '섹션(Section)'이라 칭한다. '섹션'은 또한 여러 유형[악장(음악), 장면(오페라)]으로 구분될 수 있다
- 이러한 '저작'의 하부구조는 단 한번만 정의되며, '표현물'은 이렇게 선언된 '저작구조'를 전체나 일부분을 공유한다고 할 수 있다.
- 특정 '표현물'이 '저작'내에서 어떠한 부분에 해당하는지 연결 관계를 생성하기 위해서 이 단계에서 선언된 '저작'의 하부구조인 '섹션'을 참조하게 된다.

예) 베토벤 5번 교향곡 1악장: Allegro
 베토벤 5번 교향곡 2악장: Andante
 베토벤 5번 교향곡 3악장: Allegro
 베토벤 5번 교향곡 4악장: Allegro

(3) 표현물 (Instantiation)

- 이 단계에서 기술의 대상은 하나의 '저작'과 관련된 여러 다른 형태의 '표현물'들이다.
- 이러한 '표현물'은 '컨테이너(Container)'라는 물리적인 저장물에 담겨서 이용할 수 있는 형태로 제공된다.
- 표현형태(representation type)는 가사

(text), 악보(score), 오디오(audio), 비디오(video) 등을 예로 들 수 있다. '표현물' 단계에서는 동일한 '저작'과 관련된 여러 '표현물'들의 형태를 구별할 수 있도록 기술한다.

예) Phantom 오페라 악보(Score)

Phantom 오페라를 연주한 것에 대한
Sound Recording(Audio)

Phantom 오페라를 연주한 것에 대한
Video Recording(Visual)

Phantom 오페라의 가사(Text)

(4) 저작구조 바인딩(Work Structural Bindings)

- 이 단계에서는 <그림 5>와 같이 '저작구조의 선언' 단계에서 선언된 '저작'의 하부구조를 참조하여 '표현물'의 각각의 하부 구조와 실질적인 서비스 대상인 '컴퓨터 파일(Media Object)'과의 연결 관계를 규정한다.
- 베토벤 5번 교향곡 1 악장('저작'의 '섹션')

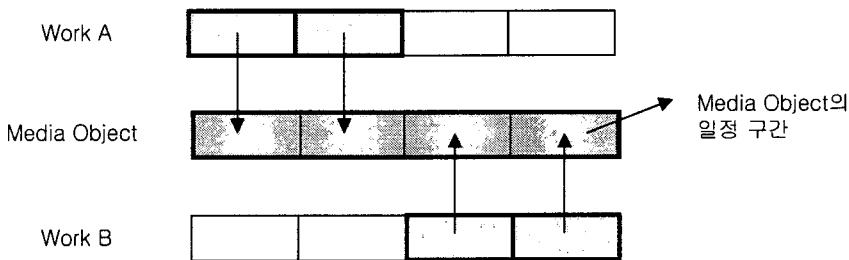
의 연주를 녹음한 것(표현물)이 베토벤 5번 교향곡 전체를 연주한 내용을 저장하고 있는 컴퓨터 파일의 어떤 구간에 해당하는지 연관지을 수 있다.

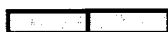
- 베토벤 5번 교향곡 1악장(저작 섹션)의 악보(표현물)가 베토벤 5번 교향곡 내용 전체를 담고 있는 악보를 디지털화한 컴퓨터 파일의 어떠한 부분에 해당하는지 연관지을 수 있다.

- 이러한 구조적 바인딩(Structural Binding)을 통해서 시스템 이용자는 하나의 '저작'과 관련된 다양한 형태의 디지털 자원들을 이용하는데 있어서 '저작'의 특정 부분과 관련된 '컴퓨터 파일'의 정확한 이용위치를 지시받음으로써 이용자가 원하는 정보에 직접 접근하여 시간을 절약할 수 있게 된다.

(5) 컨테이너(Container)

- 동일 '저작'과 관련된 여러 '표현물'은 '컨테이너'라는 물리적인 저장물을 통해서 이



 : Work 의 내용 중 일정 부분을 표현하는 Instantiation 을 나타낸 것

<그림 5> '저작', '표현물', '컴퓨터 파일' 간의 구조적 바인딩

용할 수 있는 상태로 만들어 진다.

- 하나의 '컨테이너'에는 1 개의 '저작'과 관련된 '표현물'들만이 포함될 수도 있고, 여러 개의 '저작'과 관련된 '표현물'들이 포함될 수도 있다. 즉, 하나의 '컨테이너'에는 여러 '저작'과 관련된 내용이 포함 될 수 있다.
- 이러한 '컨테이너'에는 오페라와 같이 규모가 큰 '저작'의 일부만 담겨 있을 수 있고, 또한 하나의 '컨테이너'에 여러 개의 '저작'이 담겨 있을 수도 있다. '표현물'에 대한 예로는 CD, LP, 책자 형태의 악보집(anthology of scores) 등이 있다.

(6) 컨테이너 구조선언과 구조 바인딩
(Container Structure Declaration & Structural Bindings)

- '컨테이너'도 '저작'과 마찬가지로 내부적인 하부구조를 지닐 수 있다.
- 이 단계에서는 <그림 6>과 같이 '컨테이너'의 하부구조에 대하여 선언하며 하부

구조의 각 단계와 실질적인 서비스 대상인 '컴퓨터 파일'과의 '구조적 바인딩'을 규정하게 된다.

○ 아이템(Item)

- Side A, Side B 혹은 Disc 1, Disc 2 등과 같이 하나의 '컨테이너'에 속한 항목이다.

- '아이템' 하단에는 Div가 올 수도 있고 아니면 곧바로 Chunk가 올 수도 있다.

예: '베토벤 5번 교향곡' 연주곡이 수록된 더블 CD의 Disc 1, Disc 2.

'베토벤 5번 교향곡' 연주곡이 수록된 카세트테이프의 Side A, Side B.

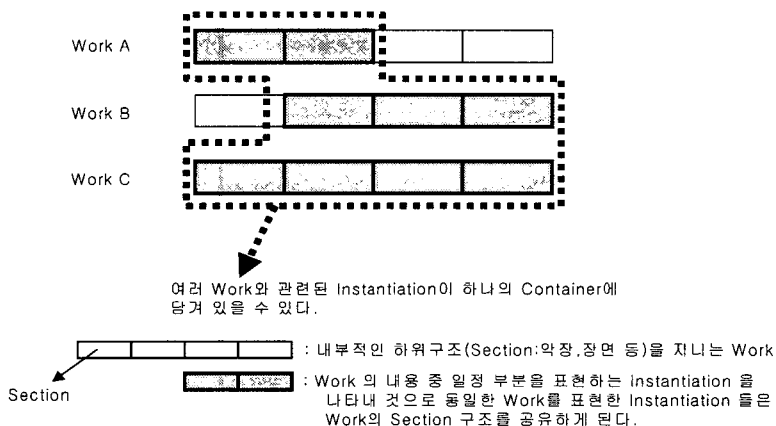
○ Div

- '아이템' 하단에서의 구분의 단위

○ Chunk

- '컨테이너'의 내부 구조의 마지막 단계로 '아이템'의 하단이나 Div의 하단에 위치한다.

- '컴퓨터 파일'과 실질적인 바인딩이 이루어지는 부분이다.



<그림 6> '저작', '표현물', '컨테이너' 간의 구조적 관계

예: '베토벤 5번 교향곡'을 담고 있는 Tape 인 경우 A면(Item)의 1악장, 2악장 등으로 Chunk를 구분할 수 있다.

- '저작구조의 선언'과 마찬가지로 '컨테이너'의 일정 부분과 '컴퓨터 파일'의 특정 이용 구간을 정확하게 지시함으로써 이용자 정보 접근에 소요되는 시간을 절약할 수 있도록 한다.

(7) 컴퓨터 파일(Media Object)

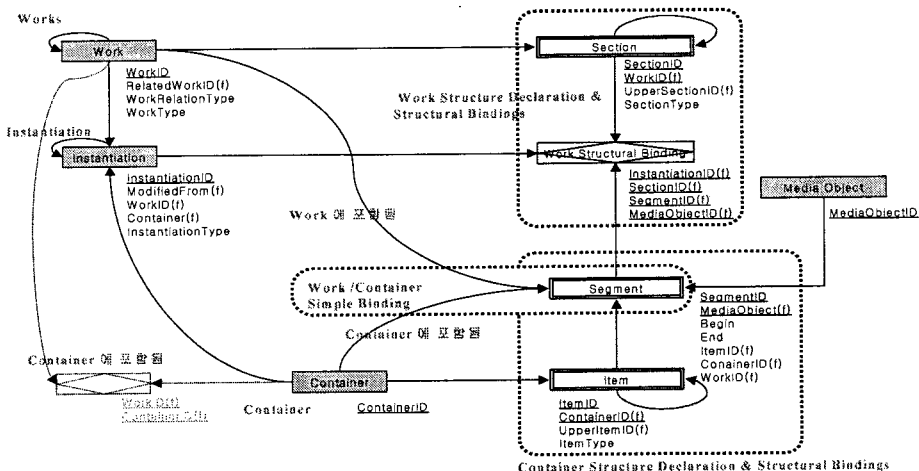
- '컴퓨터 파일'은 작품의 연주, 녹음, 악보, 가사 등과 같은 '표현물'을 디지털화한 디지털 자원을 의미한다.
- 하나의 '컴퓨터 파일' 내에는 여러 '저작'과 관련된 '표현물'들이 포함되어 있을 수 있다.
- 여러 개의 '컴퓨터 파일'이 특정 '저작'과 관련된 하나의 '표현물' 내용을 이루고 있을 수 있다.

- 하나의 '컴퓨터 파일'내에는 여러 '컨테이너'에 담겨 있는 내용이 포함되어 있을 수 있다.
- 여러 개의 '컴퓨터 파일'이 하나의 '컨테이너'에 담겨 있는 내용을 이루고 있을 수 있다.

예) Beethoven_sym_5_1.wav (교향곡 중 1 악장만 담고 있음)
 Beethoven_sym_5_2.wav (교향곡 중 2 악장만 담고 있음)
 Beethoven_sym_5.mp3 (교향곡 전체를 담고 있음)

아래의 <그림 7>은 음악자료 메타데이터 모델의 ER-Diagram을 보여주고 있고, 그 모델의 내용을 상세히 설명하면 다음과 같다.

- '저작'은 다른 '저작'과 연관될 수 있고 '저작'의 내용이 여러 섹션(section)으로 나뉠 수 있다.
- 하나의 '저작'은 다양한 '표현물'에 담겨질 수 있으나, 하나의 '표현물'은 오직 한 '저작'에 속한다.

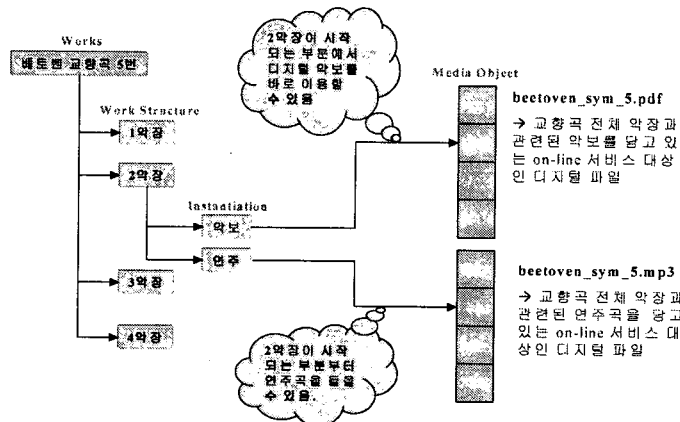


<그림 7> 음악 자료 메타데이터 모델의 ER-Diagram 표현

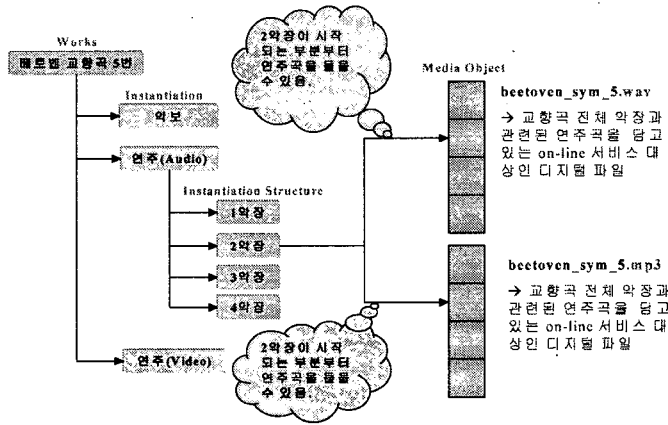
- 작에 관한 것이어야 한다.
- ‘저작’의 내용이 여러 ‘세그먼트(segment)’로 나뉘질 수 있으나, 하나의 ‘세그먼트’는 한 ‘저작’에 관한 것이어야 한다.
- 하나의 ‘세그먼트’는 다양한 ‘저작구조 바인딩’의 형태를 취할 수 있으나, 한 ‘저작구조 바인딩’은 한 ‘세그먼트’에 관한 것이어야 한다.
- 한 ‘아이템’이 여러 ‘세그먼트’에 속할 수 있으나, 한 ‘세그먼트’는 한 ‘아이템’에 관한 것이어야 한다.
- 한 ‘세그먼트’는 오직 한 ‘컴퓨터 파일’ 형태를 취해야 한다.
- ‘저작’과 마찬가지로 하나의 ‘표현물’은 다른 ‘표현물’들과 연관관계를 형성할 수 있고, 다양한 ‘저작구조바인딩’으로 표현될 수 있다.
- 한 ‘섹션’에 있는 내용이 여러 ‘저작구조 바인딩’에 포함될 수 있으나, 그러나 하나의 ‘저작구조 바인딩’은 한 ‘섹션’에만 속해야 한다.

- 한 ‘컨테이너’는 여러 ‘표현물’, ‘세그먼트’, ‘아이템’을 담을 수 있으나, 각 ‘표현물’, ‘세그먼트’, ‘아이템(item)’은 한 ‘컨테이너’에만 속해야 한다.
- 점선으로 표현된 ‘저작’과 ‘컨테이너’ 관계는 직접적인 연결관계는 아니지만, ‘컨테이너’에 포함된 내용이 ‘저작’과 간접적으로 연결되어 있음을 보여 주고 있다.

본 연구가 IFLA의 FRBR 모델에 근거하여 음악정보를 상세하게 모델링한 관계로 이용자의 요구에 더 부합하는 형태로 서비스를 할 수 있는 예를 <그림 8>과 <그림 9>의 이용자 서비스 흐름도가 보여주고 있다. 그 흐름도를 보면, 이용자는 ‘저작’에 해당하는 ‘베토벤 교향곡 5번’ 작품을 선택하면, 그 작품이 1 악장에서 4 악장으로 구성되는 ‘저작구조’를 지니고 있다는 것을 알게 되고, 그 중에서 ‘2 악장’을 선택하면, 그 악장에 대한 ‘표현물’로서 ‘악보’와 ‘연주’가 있음을 보여 준다. 이용자는 이 중에서 자신이 원하는 ‘표현물’을 선택할 수 있고



<그림 8> 음악 자료 이용자 서비스 흐름도 1



〈그림 9〉 음악 자료 이용자 서비스 흐름도 2

그리고 선택된 표현물이 담겨진 '컴퓨터 파일' 중 하나를 선택하면, 자신이 원하는 형태로 그 정보를 직접 사용할 수 있다는 서비스 흐름도를 보여 주고 있다.

3. XML 스키마 기반의 메타데이터 표현 체계

3. 1 XML 스키마 구현 원칙

본 연구에서는 시맨틱웹의 기저를 이루고 있는 URI(Uniform Resource Identifier)와 이름공간(Namespace) 개념을 체계적으로 적용하였다. 본 연구에서 새로 설계한 응용 프로파일(Application Profile)에서는 국제표준으로 활용되고 있는 메타데이터 스키마에서 지정한 각 요소의 불변의 식별자인 URI를 적극 활용하였고, 국제적으로 존재하지 않는 요소에 대해서는 이름공간과 연관시킨 불변의 식별체계를 체계적으로 부과함으로써 국제적으로 의

미충돌이 없이 통용될 수 있도록 설계하였다.

본 연구에서 설계한 응용 프로파일은, 더블린 코어(Dublin Core: DC) 기반의 메타데이터 집합을 구현하는 XML 스키마는 더블린 코어에서 제안한 다음과 같은 자료를 참고하였다.

- Guidelines for implementing Dublin Core in XML (Powell 2002)
- Example Dublin Core XML Schemas[11]

이렇게 더블린 코어를 기본 메타데이터 요소로 적용함에 따른 더블린 코어 기본요소에 대한 의미 세분화에 있어서 더블린 코어 한정어만으로 해결할 수 없는 기술요구사항을 충족시키기 위해서 고서, 고문서, 음악자료, 학술회의 및 세미나 자료 자체의 더블린코어 한정어를 제 1 하위요소로 선정하였고, 더블린 코어 한정어에 대해서 추가적인 세분화가 필요한 경우에는 제 2 하위요소를 선정하였다.

주요소 집합(Element Set)과 한정어(Qu-

alifiers)의 사용 원칙은 다음과 같다.

- 하위 요소가 없는 요소는 주요소가 값을 취할 수 있지만 하위 요소가 있는 주요소는 값을 취하지 않고 컨테이너 역할을 한다.
- DC 주요소 집합의 하위 요소로는 DC 한정어와 서울대학교(이하, SNU) 한정어가 올 수 있다.
- SNU 주요소 집합의 하위 요소로는 SNU 한정어가 온다.

XML 스키마 구현 시 사용한 이름공간(namespace)은 다음과 같다.

- 15 DC 요소 이름공간 (dc):
http://purl.org/dc/elements/1.1/
- DC 추가요소, 한정어 및 인코딩스킴 이름공간 (dcterms):
http://purl.org/dc/terms/
- SNU 주요소 이름공간 (snu):
http://www.snu.ac.kr/metadata/
- SNU 한정어 이름공간 (snuterms):
http://www.snu.ac.kr/metadata/terms/
- DC 타입 이름공간 (dcmitype):
http://purl.org/dc/dimtype/

XML 스키마를 사용한 구현 예는 다음과 같다.

(1) 자식 요소가 있는 경우

```
<xs:complexType name="titleType">
  <xs:sequence>
    <xs:element ref="snuterms:mainTitle"/>
    <xs:element ref="dcterms:alternative"
      minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
```

위의 예에서 <titleType>은 하위요소가 있기 때문에 XML 스키마에서 <complexType>으로 정의되었고, 이 요소는 자식요소가 있기 때문에는 container 역할만 하게 되고, 실제 요소의 값으로는 하위요소인 <mainTitle>(대표 제목)과 <alternative>(대체 제목)에만 제목의 값이 입력된다. 원칙적으로 자식요소가 있는 경우에 상위요소는 컨테이너 역할만 하도록 설계할 것을 권한다. 여기에서 <sequence>의 의미는 데이터 입력 시에 '대표 제목'이 먼저, 그리고 '대체 제목'의 순으로 입력되어야 한다는 의미를 나타내는 것이다. 위의 스키마 부분에 대한 데이터 인스턴스의 예를 보이면 다음과 같다. 이 인스턴스를 살펴보면, 'mainTitle'은 snuterms(서울대학교 이름공간)에서 정의한 용어, <alternative>는 dcterms(더블린코어 이름공간)에서 정의한 용어임을 분명하게 식별할 수 있다.

```
<titleType>
  <snuterms:mainTitle>삶의 의미는 무엇인가?
</snuterms:mainTitle>
  <dcterms:alternative>What is meaning of life?
</dcterms:alternative>
  <dcterms:alternative>인생철학에 대한 논단
</dcterms:alternative>
  <dcterms:alternative>Discussion on Philosophy
of Life</dcterms:alternative>
</titleType>
```

(2) 자식 요소가 없이 스키마만을 가지는 경우

자식요소가 없는 경우에도 'attribute' 즉 '속성'을 사용해서 그 요소의 값에 대한 의미를 더 구체적으로 표현하고자 할 경우에는 XML 스키마 <complexType>으로 정의해야 한다. 이런 경우에는 XML 스키마에서 이미 정의한

기본 데이터 유형을 'base'를 정하거나 아니면 새로 정의한 데이터 유형을 기반으로 해서 필요에 따라 확장하거나 제한하는 형태를 취할 수 있다. 여기에서는 본 연구에서 이미 정의된 <elementType>이라는 데이터 유형을 확장하는 방법을 보이고 있다. 이 기본 데이터 유형에 속성을 정의하는데 이미 정의된 'subjectScheme'이라는 객체를 참조하는 형태로 정의하였다. 재활용성을 최대한 하기 위한 설계의 원칙을 적용한 것이다.

```
<xs:complexType name="subjectType">
  <xs:simpleContent>
    <xs:extension base="dctype:elementType">
      <xs:attributeGroup ref="dcxml:subjectScheme"/>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>
```

(3) 자식 요소가 없이 스킴과 Dumb Down에 의한 상세 한정어를 갖는 경우

다음은 자식 요소가 없는 스킴과 답다운 원칙에 의한 상세 한정어를 정의하는 한 예로써 'workRelationRefinementType' 요소에 관한 것이다. 위의 예와 같이, 이미 정의된 이 요소도 하위요소 없이 3개의 속성을 정의하고 있다. 첫째는 'relationType'이라는 속성을 정

```
<xs:complexType name="workRelationRefinementType">
  <xs:simpleContent>
    <xs:extension base="dctype:elementType">
      <xs:attribute name="relationType" type="relationType"/>
      <xs:attributeGroup ref="dcxml:identifierScheme"/>
      <xs:attributeGroup ref="dcxml:relationRefines"/>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>
```

의하고, 둘째는 이미 정의된 'identifierType'을 재활용, 셋째도 역시 이미 정의된 'relationRefines'를 재활용 하고 있다.

(4) 요소의 값으로 나열 값을 갖는 경우

다음은 요소의 값을 나열할 필요가 있는 경우에 대한 모델로서 'relationTypeType' 태그를 정의하는 예를 보인 것이다. 이 요소는 XML 스키마에서 정의한 'string' 데이터 유형을 제한적으로 사용하도록 정의하는 것으로 허용하는 값을 열거하여 그 외의 값을 취할 수 없도록 제한하는 형태를 취하게 된다. 이 정의에 의하면 어떤 요소가 'relationTypeType' 데이터 유형으로 정의한다면, 그 요소의 값으로는 'derived'나 'isMemberOf' 둘 중의 하나만 가능하게 된다.

```
<xs:simpleType name="relationTypeType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="derived"/>
    <xs:enumeration value="isMemberOf"/>
  </xs:restriction>
</xs:simpleType>
```

(5) 두 요소 중 하나만 자식 요소로 오는 경우

다음은 어떤 요소의 하위요소로 둘 중에서 하나만 선택해야 하는 경우에 대한 예를 보인 것이다. 여기에서 'relationType'을 글로벌 <complexType>으로 정의하였고, 이 객체의 하위요소로는 'isArticleOfConference'나 'isArticleOfSession' 둘 중의 하나를 선택해야 한다고 정의하고자 할 때는, XML 스키마에서 정의한 <choice>를 사용해야 한다는 것을 보인 것이다.

```
<xs:complexType name="relationType">
  <xs:choice>
    <xs:element ref="snuterm:isArticleOfConference"/>
    <xs:element ref="snuterm:isArticleOfSession"/>
  </xs:choice>
</xs:complexType>
```

(6) 스킴 정의 (여러 가지 스킴이 있을 경우)

다음은 단순한 스킴의 값들을 열거하기 위한 XML 스키마 모델을 제시한 것이다. 예를 들면, 'subjectSchemeType'을 하위요소나 속성이 필요치 않기 때문에 <simpleType>으로 정의하면 되고, 'string'을 제한적으로 사용하도록 정의한 예이다. 스킴의 값으로 허용하고자 하는 리스트를 열거하면 되고, 여기에서 정의된 'subjectSchemeType'은 글로벌 영역에 속하기 때문에 어디에서나 재사용 될 수 있다.

```
<xs:simpleType name="subjectSchemeType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="LCSH"/>
    <xs:enumeration value="MESH"/>
    <xs:enumeration value="DDC"/>
    <xs:enumeration value="LCC"/>
    <xs:enumeration value="UDC"/>
  </xs:restriction>
</xs:simpleType>
<xs:attributeGroup name="subjectScheme">
  <xs:attribute name="scheme" type="subjectSchemeType"/>
</xs:attributeGroup>
```

(7) 스킴 정의 (한가지 스킴만 있을 경우)

스킴의 값이 여러개 있는 것이 아니라 한 값만 있을 경우에는 XML 스키마에서 정의한 'fixed' 속성을 활용해서 값을 고정시키면 된다.

```
<xs:attributeGroup name="publisherScheme">
  <xs:attribute name="scheme" type="xs:string" fixed="SNUCR"/>
</xs:attributeGroup>
```

(8) 덤 다운(Dumb Down)에 의한 상세 한정어 (Refinement Qualifier)

다음은 덤다운에 관한 정보를 어떻게 표현할 것인가에 대한 예를 보인 것이다. 예를 들면, 어떤 응용 프로파일에서 제목에 관련된 새로운 요소로 더블링크어 요소가 아닌 것으로 정의할 필요가 있고 상호운영을 증진시키기 위해서 그 요소를 더블링크어의 'title' 요소로 덤다운 시키고자 할 때, 먼저 'titleRefines'를 글로벌 속성그룹으로 정의하고 그 속성을 더블링크어의 'title' URI로 덤다운이 되어야 한다고 정의하면 된다.

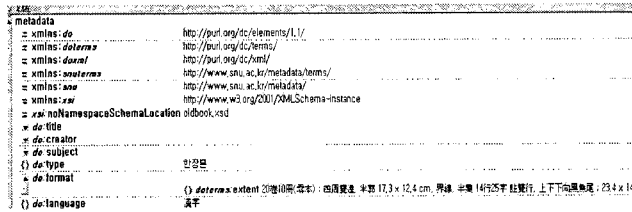
```
<xs:attributeGroup name="titleRefines">
  <xs:attribute name="refines" type="xs:string" fixed="http://purl.org/dc/elements/1.1/title" />
</xs:attributeGroup>
```

3. 2 XML 스키마 기반 메타데이터 입력 예제

이 절에서는 각 메타데이터별로 XML 스키마를 사용하여 표현된 XML 인스턴스(instance) 문서의 예제를 소개한다.

3. 2. 1 고서 메타데이터

<그림 10>은 고서 관련 메타데이터의 표현 예를 보여주고 있다. 먼저 이름공간을 정의하고 구체적 인스턴스를 입력하는 예제를 보인 것이다.



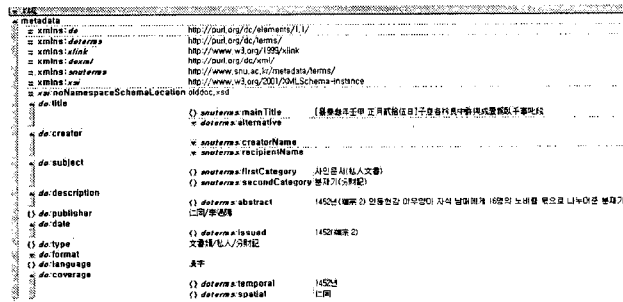
〈그림 10〉 고서 메타데이터의 표현 예

3. 2. 2 고문서 메타데이터

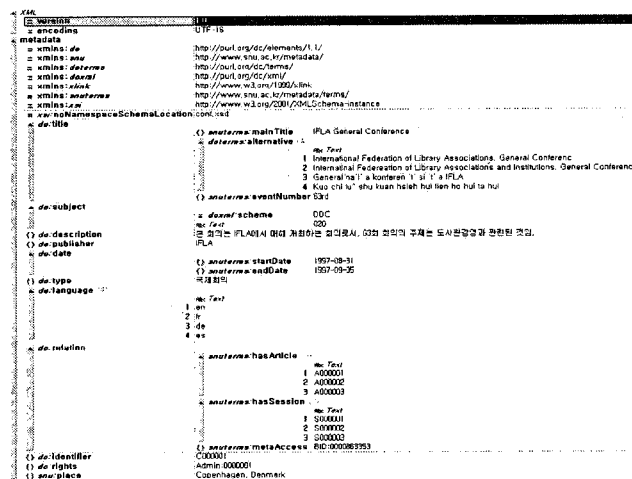
〈그림 11〉은 고문서 관련 메타데이터의 표현 예를 보여주고 있다. 먼저 이름공간을 정의하고 구체적 인스턴스를 입력하는 예제를 보인 것이다.

3. 2. 3 학술회의, 세미나, 강연회 및 전시회

관련 메타데이터
〈그림 12〉는 학술회의의 관련 메타데이터의 표현 예를 보여주고 있다. 먼저 이름공간을 정의하고 기본요소에 관한 인스턴스의 입력과 특



〈그림 11〉 고문서 메타데이터의 표현 예



〈그림 12〉 학술회의의 등 메타데이터 중 회의에 대한 표현 예

정 컨퍼런스의 이름을 다양한 언어로 표현된 대체제목 등을 입력하는 예제를 보여 주고 있다.

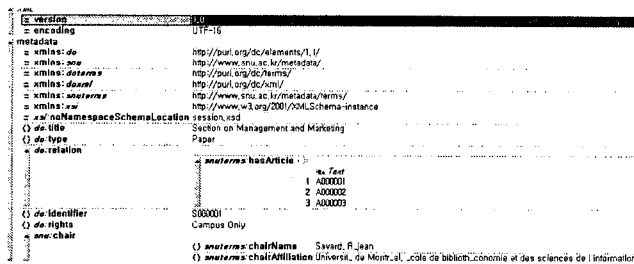
〈그림 13〉은 학술회의 관련 메타데이터를 입력하는 예를 보여주고 있다. 먼저 이름공간을 정의하고 기본정보와 학술회의에서 발표된 논문들이 '관계' 요소의 값으로 입력되어 학술회의 연결되고 있는 예제를 보인 것이다.

〈그림 14〉는 강연회 및 전시회 관련 메타데이터를 입력하는 예를 보여주고 있다. 먼저 강연회 및 전시회를 기술하는데 필요한 이름공간, 기본적인 요소 정보, 전시회나 강연회의 내용을 담고 있는 컴퓨터 파일의 바인딩 정보를 입력하는 예를 보이고 있다.

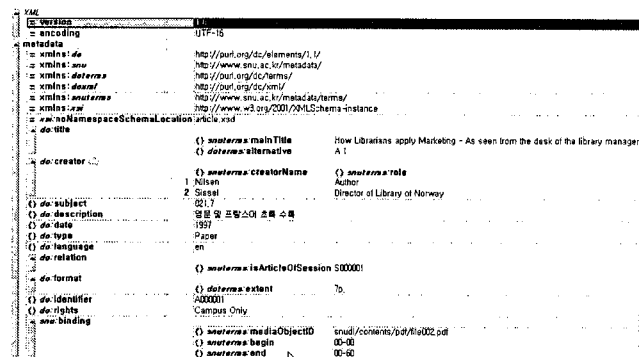
3. 2. 4 음악 메타데이터

〈그림 15〉에서는 음악 메타데이터 중 '저작권'에 대한 메타데이터를 입력하는 예제를 XMLSpy를 사용해서 보인 것이다.〈그림 16〉에서는 음악 메타데이터 중 '저작권 선언'에 대한 메타데이터를 입력하는 예제를 XMLSpy를 사용해서 보인 것이다. 특히, 한 '저작권' 정보와 연관된 '저작권' 정보인 1 약장에서 4 약장의 정보가 연계되어 입력되고 있다.

〈그림 17〉에서는 음악 메타데이터 중 '저작권 바인딩'에 대한 메타데이터를 입력하는 예제이고, '저작권 바인딩' 인스턴스가 '표현물', '저작권'과 연계되고, 이 '저작권 바인딩'에 포함된 모든 '섹션' 정보가 연결된 상태를 보여 주고 있다.



〈그림 13〉 학술회의 등 메타데이터 중 세션에 대한 표현 예



〈그림 14〉 학술회의 등 메타데이터 중 발표 자료에 대한 표현 예

xmlns:do	http://purl.org/dc/elements/1.1/
xmlns:snu	http://www.snu.ac.kr/metadata/
xmlns:doterms	http://purl.org/dc/terms/
xmlns:doxml	http://purl.org/dc/xml/
xmlns:snuterms	http://www.snu.ac.kr/metadata/terms/
xmlns:xsi	http://www.w3.org/2001/XMLSchema-instance
xsi:noNamespaceSchemaLocation	work_stru.xsd
do:title	(1) snuterm:mainTitle Beethoven, Symphony No. 5 Op.67 (3) doterms:alternative 제목곡 고함곡 5번 악장번호 07
do:creator	(1) snuterm:creatorName = doxml:scheme SNUCR (3) doterms:role =:Text Beethoven, Ludwig van, 1770-1827
do:subject	30182
do:description	F. J. von Lobkowitz and Graf A. von Rasumovsky에게 헌정된 곡들.
do:date	(1) snuterm:startDate 1803 (1) snuterm:endDate 1808 (3) snuterm:dateOfFirstPerformance 1808-12-22 (3) snuterm:dateOfFirstPublication 1809-04
do:type	Symphony
do:identifier	M001110
do:language	de
do:relation	snuterm:workRelation = snuterm:relationType isMemberOf =:Text M001102
snu:workType	Single
snu:instrumentation	(3) snuterm:instrumentName 2 Flutes, 1 Piccolo, 2 Oboes, 2 Clarinets, 2 Bassoons, 1 Cor Anglais, 2 Cors, C-bell
snu:act	(1) snuterm:placeOfFirstPerformance Theater-an-der-Wien
snu:hasStructure	Yes
snu:simpleBinding	(1) snuterm:instantiationID 1001234 (3) snuterm:mediaObjectID =:Text snud:/contents/A00/classic/Beethoven/synt

〈그림 15〉 음악 메타데이터 중 '저작'에 대한 표현 예

xmlns:do	http://purl.org/dc/elements/1.1/
xmlns:snu	http://www.snu.ac.kr/metadata/
xmlns:doterms	http://purl.org/dc/terms/
xmlns:doxml	http://purl.org/dc/xml/
xmlns:snuterms	http://www.snu.ac.kr/metadata/terms/
xmlns:xsi	http://www.w3.org/2001/XMLSchema-instance
xsi:noNamespaceSchemaLocation	work_stru.xsd
do:identifier	M001110
snu:workLabel	Beethoven's Symphony No. 5
snu:section	(1) snuterm:sectionTitle (1) snuterm:sectionIdentifier (1) snuterm:sectionLabel (1) snuterm:sectionType 1 Allegro con brio S000001 19악 movement 2 Andante con moto S000002 24악 movement 3 Allegro S000003 30악 movement 4 Allegro S000004 32악 movement

〈그림 16〉 음악 메타데이터 중 '저작구조 선언'에 대한 표현 예

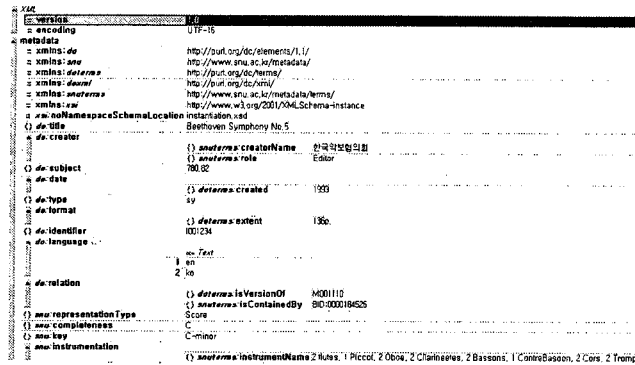
xmlns:do	http://purl.org/dc/elements/1.1/
xmlns:snu	http://www.snu.ac.kr/metadata/
xmlns:doterms	http://purl.org/dc/terms/
xmlns:doxml	http://purl.org/dc/xml/
xmlns:snuterms	http://www.snu.ac.kr/metadata/terms/
xmlns:xsi	http://www.w3.org/2001/XMLSchema-instance
xsi:noNamespaceSchemaLocation	work_stru_bind.xsd
snu:instantiationID	1001234
snu:workID	M001110
snu:section	(4) (1) snuterm:sectionID (1) snuterm:binding 1 S000001 = snuterm:binding 2 S000002 = snuterm:binding 3 S000003 = snuterm:binding 4 S000004 = snuterm:binding

〈그림 17〉 음악 메타데이터 중 '저작구조 바인딩'에 대한 표현 예

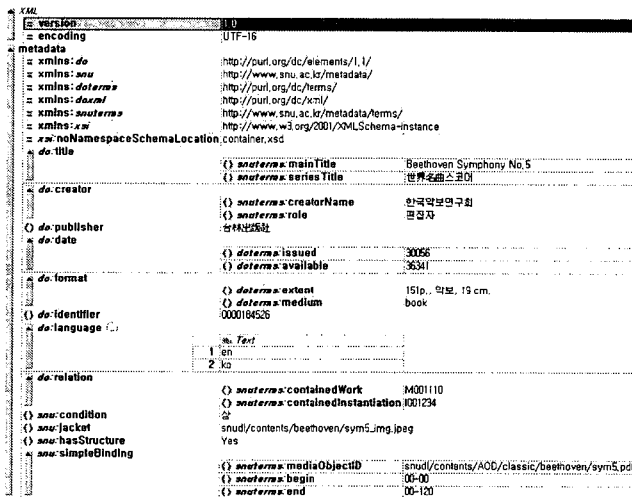
〈그림 18〉에서는 음악 메타데이터 중 '표현물'에 대한 메타데이터를 입력하는 예제이다. 표현물을 기술하는데 사용된 메타데이터 요소들이 이름공간과 연결되어 있고, 또한 이 표현물이 영어와 한국어로 되어 있다는 것과 그 표

현물과 연관관계가 있는 정보를 제시하고 있다.

〈그림 19〉에서는 음악 메타데이터 중 '컨테이너'에 대한 메타데이터를 입력하는 예제이다. 각 '컨테이너'에 대한 정보는 이름공간과 상위 메타데이터를 입력하고, 컨테이너에 대한 언



〈그림 18〉 음악 메타데이터 중 Instantiation에 대한 표현 예



〈그림 19〉 음악 메타데이터 중 '컨테이너'에 대한 표현 예

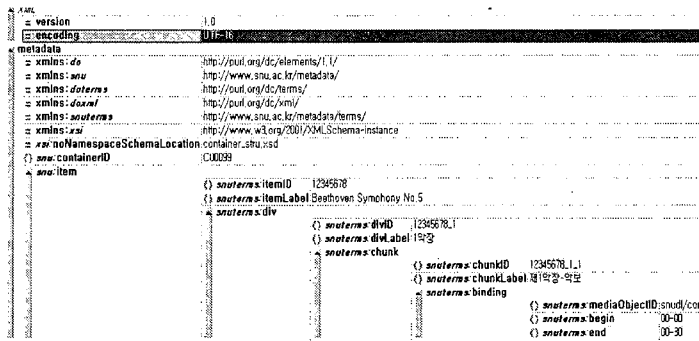
어 정보, 관련된 다른 URI 정보, 그리고 각 '컨테이너'의 바인딩에 관한 정보, 즉 '컴퓨터 파일'에 대한 정보를 입력한다.

〈그림 20〉에서는 음악 메타데이터 중 '컨테이너 바인딩'에 대한 메타데이터를 입력하는 예제이다. 컨테이너 바인딩에 관한 정보는 먼저 '아이템'에 대해서 입력하고, 각 '아이템'과 연결된 'chunk'를 입력하고, 각 'chunk'가 어떤 '컴퓨터 파일'에 담겨 있는지에 대한 정보를

함께 입력한다.

4. 디지털 특수자료 관리를 위한 발전 방향

본 연구의 큰 성과 중의 하나라고 판단되는 점은 체계적인 네임스페이스와 URI의 활용이었다고 본다. 국제적으로 널리 활용되고 있는



〈그림 20〉 음악 메타데이터 중 '컨테이너 바인딩'에 대한 표현 예

다양한 메타데이터 스키마를 네임스페이스와 URI 원칙에 근거하여 도입하였고, 디지털 특수자료의 성격을 고려하여 필요한 요소를 네임스페이스와 URI를 부착하여 확장하였다. 이러한 시도는 시맨틱웹의 하부구조를 튼튼하게 하는 것이며 앞으로 축적될 정보를 체계적으로 통합할 수 있는 기반을 마련했다고 본다(Bernes-Lee 2003, Bray 1999).

미래의 발전방향에 대해서 언급하자면, XML 스키마 기반 메타데이터는 네임스페이스와 URI를 지원한다는 점에서는 그 우수성이 있지만, 메타데이터의 의미를 기계가 이해할 수 없기 때문에 의미있는 지식정보의 통합이나 새로운 데이터의 유추를 가능케 하는 데이터 마이닝의 역할은 할 수 없다는 것이다. 이러한 방향으로 발전하기 위해서는 모든 메타데이터 스키마와 응용 프로파일이 RDF 스키마 기반으로 표현되어야 한다(Miller 1998). 정의된 모든 메타데이터 요소는 고유의 식별자를 지녀야 하고, 그 식별자를 클릭하였을 경우에 RDF 스키마로 정의된 내용이 웹상에 지속적으로 제공되어야 한다(Beckett 2002, Kokkelink 2002). 이러한 정의를 한 곳에 저장해서 제공할 수 있

는 메타데이터 레지스트리(Metadata Registry)를 구축한다면 더욱 향상된 서비스를 제공할 수 있을 것이다(오삼균 2002).

마지막으로 현 브라우징이나 검색체계는 이용자가 원하는 정보를 제대로 필터링(filtering) 하는데 제한을 지니고 있다. 가장 큰 이유는 현 정보시스템의 데이터 구조에 있다고 본다. 이에 대한 대응책으로 온톨로지 기반으로 브라우징과 검색체계를 바꾸려는 움직임이 국제적으로 활발히 진행되고 있다. 유연한 브라우징 체계와 거침없는 정보의 통합을 가능케 하는 기반으로는 국제표준(ISO 13250)인 토픽맵(Topic Maps)을 활용하고 있으며(Pepper 2004), 추론과 데이터 마이닝을 주안점으로 하고자 할 경우에는 RDF와 OWL(Web Ontology Language)을 적극 활용하고 있다. 향후 디지털 도서관의 방향을 시맨틱웹 기술을 적용하여 유연한 브라우징 체계를 이용자의 특성에 맞게 개인화 할 수 있는 체제와 새로운 암묵적 지식을 연계시킬 수 있는 서비스를 갖추는 방향으로 나가야 할 것이다(Smith and McGuinness 2002).

5. 결 론

본 연구에서는 IFLA FRBR 기본 모델을 기초로 하여 고서, 고문서, 음악 자료, 학술회의/세미나 자료의 관리에 있어서 이용자의 정보요구를 최대한 수용할 수 있는 최적의 메타데이터 모델과 이에 대한 XML 표현체계를 제시하였다.

MARC의 데이터 모델은 자원기술의 대상을 단일의 객체로 파악하는 단층 데이터 모델이기 때문에 여러 객체들 간의 연결 관계를 설정할 수 있는 다층 데이터 모델을 이용한 정보 자원 기술이 필요한 경우는 적절치 못하다. 이러한 MARC 데이터 모델의 기본적인 한계점을 극복하고자 이용자 서비스 지향적인 21세기형 서지관리 모델로서 IFLA에서는 FRBR을 제시하였다.

목록의 첫 번째 목적은 찾거나 하는 자원을 제대로 검색할 수 있도록 하는 것이고 두 번째 목적은 검색된 자원 중에서 서로 연관성이 있

는 정보를 그룹해서 보여 줄 수 있어야 한다는 것이다. 현존하는 목록 시스템 중에서 목록의 둘째 목적을 제대로 구현한 사례는 찾아보기 힘든데, 본 연구는 IFLA FRBR 모델을 충실히 구현하려는 노력의 일환으로 이루어진 것이다.

또한 메타데이터의 각 요소를 철저히 이름공간과 연결시켜 고유의 식별자인 URI를 할당함으로써, 시맨틱웹 환경에서 정보의 상호운용성을 크게 증진시킬 방법을 이 연구에서 실질적으로 구현하였다. 국내의 모든 메타데이터가 이와같이 고유한 URI로 관리되는 체계로 빨리 전환하기 위해서는 각 기관에서 이름공간과 URI 할당에 대한 정책을 세워야 하고, 이렇게 제정된 모든 메타데이터 요소를 등록하여 메타데이터 설계자들이 항상 참조할 수 있는 메타데이터 레지스트리의 운영을 서두려야 할 것이다. 본 연구가 국내의 메타데이터 관리의 측면에서 새로운 시발점을 제시하는 계기가 되기를 바란다.

참 고 문 헌

- 서울대학교 중앙도서관. 2002. 디지털 도서관 구축을 위한 XML Schema 메타데이터 구현연구.
- 오삼균. 2002. 시맨틱웹 기술과 활용방안. 『정보관리학회지』, 19(4): 297-320.
- 오삼균. 2002. 디지털도서관에서의 메타데이터의 역할. 『정보과학회지』, 20(8): 45-57.
- 조재인. 2003. 연관 저록의 구조적 접근을 위한 목록 체계 구축 및 평가에 관한 연구: 객체 지향적 접근. 『한국문헌정보학회』, 37(3): 15-34.
- 허정희, 강상욱, 윤세진, 오삼균, 서태설, 김희석. 2003. 국가 URN기반 UCI 식별체계의 식별메타데이터 연구. 한국전산원.
- Ahmed, K. et al. 2001. Professional XML Meta Data. Birmingham, UK: Wrox.
- Bearman, D. et al. 1999. "A Common Model to Support Interoperable Meta-

- data: Progress report on reconciling metadata requirements from the Dublin Core and INDECS/DOI". D-Lib Magazine January. <<http://www.dlib.org/dlib/january99/bearman/01bearman.html>>.
- Beckett, D., Miller, E. and Brickley, D. 2002. "Expressing Simple Dublin Core in RDF/XML." <<http://www.dublincore.org/documents/2002/07/31/dcmes-xml/>>.
- Berners-Lee, Tim. 2003. "What do HTTP URIs Identify?" <<http://www.w3.org/DesignIssues/HTTP-URI>>.
- Berners-Lee, T. et al. 2000. "Semantic Web Development: Technical Proposal." <<http://www.w3.org/2000/01/sw/DevelopmentProposal>>.
- Booth, David. 2003. "Four Uses of a URL: Name, Concept, Web Location and Document Instance." <http://www.w3.org/2002/11/dbooth-names/dbooth-names_clean.htm>.
- Bray, T., D. Hollander, and A. Layman. 1999. "Namespaces in XML." <<http://www.w3.org/TR/REC-xml-names/>>.
- Clark, Kendall Grant. Identity Crisis. XML.com, September 11, 2002. <<http://www.xml.com/pub/a/2002/09/11/deviant.html>>.
- Connolly, D., F. Harmelen, I. Horrocks, D. McGuinness, P. Patel-Schneider, L. Stein. 2001. "Annotated DAML+OIL Markup." <<http://www.w3.org/TR/daml+oil-walkthru/>>.
- Costello, R. 2002. "XML Schema Tutorial." <<http://www.xfront.com/#schema>>
- DCMI. 1999. "Dublin Core Qualifiers in RDF." <<http://purl.org/dc/terms/>>.
- DC-XML-Guideline. <<http://www.ukoln.ac.uk/metadata/dcmi/dc-xml-guidelines/>>. UKOLN
- Dean, M., D. Connolly, F. Harmelen, J. Hendler, I. Horrocks, D. McGuinness, P. Patel-Schneider, and L. Stein. 2002. "Web Ontology Language (OWL) Reference Version 1.0." <<http://www.w3.org/TR/owl-ref/>>.
- Denker, G., et al. 2001. "Accessing Information and Services on the DAML-Enabled Web." Semantic Web Workshop 2001. Hongkong, China.
- Fallside, D. 2001. "XML Schema Part 0: Primer." <<http://www.w3.org/TR/xmlschema-0/>>.
- Garshol, LM. 2004. "Metadata? Thesaurus? Taxonomies, Topic Maps?: Making sense of it all." XML 2004 Conference, Netherland.

- <<http://www.ontopia.net/topicmaps/materials/tm-vs-thesauri.html>>.
- Hawke, Sandro. 2003. "Disabiguating RDF Identifiers."
<<http://www.w3.org/2002/12/rdf-identifiers/>>.
- Hickery, T., O'Neil, E. and Toves, J. 2002. "Experiments with the IFLA Functional Requirements for Bibliographic Records (FRBR)." *D-Lib Magazine*, September.
<<http://www.dlib.org/dlib/september02/hickey/09hickey.html>>.
- International Federation of Library Associations and Institutions. 1998. *Functional Requirements for Bibliographic Records: Final Report, Recommended by the IFLA Study Group on the Functional Requirements for Bibliographic Records*. UBCIM Publications, New Series Vol. 19. Munich: K.G. Saur.
- Kokkelink, S. and Schwanzl, R. 2002. "Expressing Qualified Dublin Core in RDF / XML."
<<http://www.dublincore.org/documents/2002/05/15/dcq-rdf-xml/>>.
- KORMARC. 한국문헌자동화목록.
<<http://www.nl.go.kr/kormarc/>>.
국립중앙도서관.
- Manola, F. and E. Miller. 2002. "RDF Primer."
<<http://www.w3.org/TR/rdf-primer/>>.
- Miller, E. 1998. "An introduction to the Resource Description Framework."
<<http://www.dlib.org/dlib/may98/miller/05miller.html>>.
- OASIS: Organization for the Advancement of Structured Information Standards. 2004.
<http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=tm-pubsubj>.
- Pepper, S. and L. Garshol. 2004. "Toward seamless knowledge: integrating public sector portals in Norway." XML 2004 Conference and Exposition. Amsterdam, Netherland.
- Pepper, Steve & Schwab, Sylvia. 2003. "Curing the Web's Identity Crisis: Subject Indicators for RDF."
<<http://www.ontopia.net/topicmaps/materials/identitycrisis.html>>.
- Pepper, S. and L. Garshol. 2002. "The XML Papers: Lessons Learned on Applying Topic Maps." XML 2002 Conference and Exposition. Baltimore, Maryland. USA.
- Powell, A., H. Wagner, S. Weibel, T. Baker, T. Matola, and E. Miller. 2001. "Namespace Policy for the Dublin Core Metadata Initiative (DCMI)."
<<http://dublincore.org/documents/2001/10/26/dcmi-namespace/>>.

- Powell, A. and Johnston, P. 2003. "Guidelines for implementing Dublin Core in XML." <http://www.dublincore.org/documents/2003/04/02/dc-xml-guidelines/>.
- Smith, M., D. McGuinness, R. Volz, and C. Welty. 2002. "Web Ontology Language (OWL) Guide Version 1.0." <http://www.w3.org/TR/owl-guide/>.
- Stevens, Roberts, et al. 2003. "Building ontologies in DAML + OIL." *Comparative and Functional Genomics*, 4(1): 134-141.