

한국 전자책 문서표준 및 관련 표준과의 변환 기법에 관한 연구

손원성[†] · 고승규^{**} · 이경호^{***} · 임순범^{****} · 최윤철^{*****}

요 약

미국과 일본 그리고 한국에서는 전자책 문서 포맷의 표준화에 대한 필요성을 인식하고 정부기관과 관련업계 및 학계가 주축이 되어 OEB PS 및 JepaX, 그리고 EBKS 표준안을 제정한 바 있다. 그러나 EBKS는 미국의 OEB 및 일본의 JepaX와 비교하여, 제정 목적은 물론이고 내부적으로 서로 다른 구조를 포함하고 있다. 그 결과 이러한 표준 문서간의 변환을 위해서는 서로간의 문서 구조에 대한 정확한 분석이 요구되며, 실제 문서 변환시 정보 손실을 최소화 하여야 한다. 따라서 본 논문에서는 이러한 표준 문서간에 정확한 교환 및 문서 구조 손실을 최소화하기 위한 방법을 제시하고 이에 대한 실례를 살펴본다. 그 결과 본 연구에서는 EBKS 및 OEB, JepaX간의 콘텐츠 교환 가능성을 제시하며, 이러한 결과는 전자책 표준 및 기타 포맷간의 교환시에도 적용 가능하다.

A Study of Conversion Techniques between Korea eBook Documents Standard and Related eBook Standards

Won-Sung Sohn[†], Seung-Kyu Ko^{**}, Kyong-Ho Lee^{***},
Soon-Bum Lim^{****} and Yoon-Chul Choy^{*****}

ABSTRACT

The U.S., Japan and Korea have realized the importance of establishing standards for eBook document formats and with government bodies, related industries and academic circles, OEB PS, JepaX and EBKS were established. However, compared to OEB of the U.S. and JepaX of Japan, EBKS contains different objective for establishment and different structures. As a result, for converting between documents of different standards, exact analysis of the document structures are called for and minimization of information loss when actually converting documents is required. Therefore, a method for explicit exchange between documents of different standards minimizing document structure is proposed and examples are looked at. As a result, in this paper, conversion possibilities between EBKS and OEB/JepaX are introduced and the result can be applied to cases when exchanging eBook standards and other formats.

Key words: eBook, eBook Standard, eBook Conversion, XML

본 연구는 정보통신부 정보통신연구진흥원에서 지원하고 있는 정보통신기초연구지원사업의 연구결과입니다.

접수일 : 2001년 12월 31일, 완료일 : 2003년 3월 3일

[†] 준회원, 연세대학교 컴퓨터과학과 박사과정

^{**} 준회원, 연세대학교 컴퓨터과학과 공학박사

^{***} 정회원, 연세대학교 컴퓨터산업공학부 조교수

^{****} 중신회원, 숙명여자대학교 멀티미디어학과 조교수

^{*****} 중신회원, 연세대학교 컴퓨터과학과 교수

1. 서론

전자책의 의미는 일반적으로 책의 콘텐츠를 디지털 형태의 정보로 가공 및 저장한 출판물이라 할 수 있으며, eBook, e-텍스트, 온라인북, 파일북 등과 같이 다양한 이름으로 불리고 있다[1-3]. 이러한 전자책은 기존의 인쇄된 책과 비교하여 유통 과정의 단순화 및 재고 부담 절감, 멀티미디어 기능 제공, 다양한 콘텐츠 출력 기능, 그리고 저장 관리의 효율성 등의 장점들을 제공할 수 있다[1-3,28-30].

그러나 현재 전자책은 XML(eXtensible Markup Language)[6], HTML(Hypertext Markup Language)[7], PDF(Portable Document Format)[13], DVI[37], FLASH[38], 그리고 업체에서 자체 개발한 포맷 등 다양한 형태로 서비스되고 있기 때문에 전자책 콘텐츠에 대한 교환 및 공유를 어렵게 하는 가장 큰 원인이 되고 있다. 그 결과 전자책 업계에서는 여러 종류의 뷰어와 변환 도구의 개발을 위한 중복투자가 이루어지고 있으며, 이는 전자책의 시장 활성화를 가로막는 가장 큰 장애가 되고 있다.

따라서 미국과 일본에서는 전자책 문서 포맷의 표준화에 대한 필요성을 인식하고 각각 OEBF(Open eBook Forum)[14] 및 Jepa(Japanese Electronic Publishing Assosiation)[16]를 구성하여 OEB 출판 구조 명세(Open eBook Publication Structure Specification)[15]와 전자책 문서표준 JepaX[16]를 제정한 바 있다.

OEBF는 OEB PS는 OEB 패키지(OEB Package)와 OEB 문서(OEB Document)로 구성되어 있으며 이는 전자책의 배포 및 출력을 목적으로 한다(그림. 1).

OEB 패키지 파일은 몇 개의 주요 부분으로 구성되어 있다. OEB 출판물 자체를 구별할 수 있는 고유

식별자인 패키지 식별(Package Identity), 제목, 저자, 발행자 등의 메타데이터(Metadata), 하나의 출판물을 구성하는 문서, 이미지, 스타일시트 등의 파일 목록(Manifest), 순차적으로 읽도록 순서를 배열해주는 문서 배치구조(Spine), 다양한 목적과 경우에 따라 읽는 순서를 변경해 줄 수 있는 문서 탐방순서(Tours), 그리고 출판물의 기본 구성 내용 이외에 목차, 참고문헌, 색인 등의 부가적인 정보를 표현하는 안내정보(Guide)로 구성된다.

실제 전자책 내용을 표현하는 OEB 문서(OEB Document)는 OEB 사양에 부합되는 기본(basic) 문서와 임의의 XML문서인 확장(extended) 문서로 나눌 수 있다. 기본 문서(Basic OEB Document)는 OEB 사양에서 정의된 구문만을 이용하여 작성된 문서로 기본적으로 XHTML 1.0[8]과 유사한 구조를 갖는다. XHTML 1.0은 HTML 4.0의 XML 형태이므로 기존의 HTML로 작성된 전자책들은 손쉽게 OEB 기본 문서로 변환 가능하다. 그러나 이와 같은 OEB 문서에서는 구체적인 문서 구조를 정의하지 않음으로 인해 전자책 교환이나 가공 시 모호한 경우가 발생할 수 있는 단점이 있다.

JepaX 사양은 출판 업계 내부의 콘텐츠 축적이나 교환 포맷으로 사용하는 것이 목적이며 OEB와는 달리 배포 포맷으로 이용되는 것은 의도하고 있지 않기 때문에 스타일 정보는 배제하고 있다. JepaX의 구조는 그림. 1과 같이 jepaxinfo, bookinfo, cover, front, body, back으로 구성되어 있다. 또한 JepaX는 책의 구조를 “div” 엘리먼트의 속성으로 표기하고, 사용자 정의가 가능한 확장 엘리먼트를 제공하고 있다. 그러나 JepaX도 OEB PS와 마찬가지로 임의의 사용자 정의 엘리먼트가 무분별하게 발생할 수 있기 때문에 명확한 전자책 콘텐츠 교환을 보장하기 어렵다는 단점을 포함하고 있다.

한국에서도 지난 2000년 9월 문화관광부 지원 아래 59개 관련 회원사들로 이루어진 한국전자책 컨소시엄(EBK)[4]이 구성되었고, EBK의 표준화 분과위원회[5]를 통하여 2001년 4월 한국전자책문서표준(EBKS)을 제정하였으며, 현재 대한민국 산업 표준 KS 6100[39]으로 채택된 바 있다. 또한 본 표준은 특히 전자책 콘텐츠에 대한 정확한 교환을 목적으로 한다. 이를 위하여 EBKS는 미국 OEB PS와 일본 JepaX와는 달리 고정된 문서 구조를 포함하고 있는

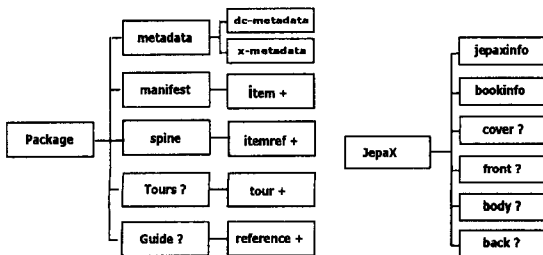


그림. 1 OEB Package와 JepaX의 전체 구조

며, 한국에서 발간되는 대부분의 책의 구조를 지원하고 있다. 또한 한국 문헌의 특징인 세로쓰기 및 다단 편집, 그리고 고어 등을 고려하였으며, 기타 문서 구조에 대한 확장 방법을 제공할 수 있다.

한편 EBKS와 OEB, JepaX는 앞서 살펴본 바와 같이 그 제정 목적과 세부적인 문서 구조간에는 상당한 차이가 존재한다. 따라서 이러한 전자책 표준간의 교환을 위해서는 각 표준들의 구조 및 특징들에 대한 자세한 정보를 유지하여야 하며, 또한 교환시 발생할 수 있는 콘텐츠 및 구조에 대한 손실을 최소화 할 수 있어야 한다. 이를 위해서는 각 표준간의 구조에 대한 특성 및 차이점에 대한 분류를 명확히 하고 그 결과를 변환과정에 적절히 적용하여야 한다. 본 논문에서는 이러한 기준에 근거하여 대한민국 전자책 표준과 미국 그리고 일본 표준간의 변환에 필요한 여러 사항들을 소개하도록 한다.

본 논문의 구성은 다음과 같다. 2장에서는 EBKS의 개요에 대하여 설명하고 3장에서는 EBKS의 자세한 구성 및 내용을 살펴본다. 4장에서는 EBKS와 OEB, JepaX간의 변환기법 및 실제 예제를 제시하며, 5장에서 본 논문의 결론을 맺는다.

2. EBKS의 개요

한국 전자책 시장은 1990년대 후반부터 전자책 전문 서비스 업체 및 전용 단말기 업체들을 중심으로 형성되었으며, 현재 오프라인 출판사들도 관련 업체와의 연계를 통하여 다양한 서비스 사업을 진행하고 있다. 그러나 한국에서의 전자책 시장은 아직 크게 활성화되지 못하고 있다. 그 이유는 EBK 표준화 분과위원회에서 EBK 59개 회원사들을 대상으로 한 설문조사[3]에서 알 수 있듯이, 업체간의 상이한 전자책 문서 포맷 사용에 따른 콘텐츠 교환 방법의 부재가 가장 큰 원인이라 할 수 있다.

따라서 표준화분과위원회에서는 위와 같은 업체들의 다양한 요구사항을 기초로, 한국에서의 전자책 문서 표준 포맷은 상호운용성, 수용성, 확장성, 응용성, 공개성, 간결성 등과 같은 이슈를 모두 지원하고자 하였다. 한편 문서 표준 포맷으로서의 HTML은 누구나 쉽게 콘텐츠를 제작할 수 있다는 장점에도 불구하고 구조 확장 및 콘텐츠와 스타일 분리가 어렵고, 재사용성이 떨어진다는 단점을 포함하고 있다. 또한 PDF는 멀티 플랫폼 지원 및 미려한 출력 결과

를 보장한다는 장점을 제공하나 부가정보 및 구조 정보 표현의 부재, 재사용성이 떨어진다는 문제가 있다. 따라서 본 분과위원회에서는 위의 이슈들을 모두 만족하는 동시에 HTML, PDF의 단점을 극복할 수 있는 XML을 한국전자책문서표준(EBKS)의 기본 포맷에 적용하기로 결정하였다[3].

이러한 EBKS 제정의 기본 목표는 가장 먼저 전자책 콘텐츠에 대한 정확한 교환과 한국 출판환경을 충분히 반영하고자 하였다. 이를 위하여 문서 구조를 위해서는 고정된 논리적 구조를 정의하였으며, 보다 유연한 문서 정의를 위한 확장 방법을 제공하고 있다. 또한 한글 인코딩[23-26]은 OEB, JepaX와의 호환을 위하여 Unicode[21]를 기반으로 하며, 고어나 EBKS에서 지원되지 않는 특수 문자 표기를 위한 메커니즘을 제공한다. 또한 한국의 출판환경의 특성중 세로쓰기와 다단 편집 등의 논리적 표기를 고려하였다. 마지막으로 EBKS는 기존 OEB PS와 JepaX, 그리고 문서 표준인 TEI[18], ISO 12083[20], DocBook DTD[19]와의 호환성을 보장하기 위하여 일반적으로 통용될 수 있는 엘리먼트나 속성들을 최대한 반영하고자 하였다.

이러한 EBKS에 기반한 콘텐츠의 교환 범위는 그림. 2와 같이 저자, 출판사와 서비스업체를 포함하고 있으며, 콘텐츠의 정확한 교환은 외국 표준과의 호환성, 관련 소프트웨어 공유, 재사용의 용이성 등을 제공할 수 있다.

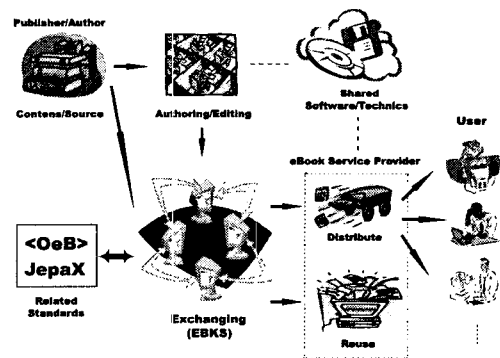


그림. 2 EBKS의 생성, 교환 배포 및 재사용의 범위 및 과정

3. EBKS DTD의 상세 구조 및 내용

EBKS는 전자책의 명확한 교환과 효율적인 재사용, 그리고 생성의 용이성, 확장 메커니즘 제공을 위

하여 고정된 문서의 논리적인 구조를 정의하였다. 문서에 대한 고정된 구조를 제공하기 위해서는 반드시 다양한 출판물에 대한 분석이 요구된다. 따라서 이를 위하여 본 분과위원회에서는 먼저 한국에서 출판된 다양한 문서들을 TEI 기준 및 전자책 업체에서 많이 판매되는 분야 중 특별히 스타일이나 문서의 구조가 복잡하다고 생각되는 분야를 분석하여 최종적으로 산문, 운문, 실용서, 스피치, 잡지, 실용서, 전집, 사진 등으로 분류하였다. 그리고 이러한 분류 작업을 바탕으로 한국 전자책 문서 표준의 상세 구조 DTD를 정의한 것이다. 다음절에서는 이와 같은 EBKS의 세부 내용을 설명한다.

3.1 EBKS 문서의 논리적 구조

EBKS의 루트 엘리먼트인 ebook 엘리먼트는 그림. 3에서처럼 메타데이터를 표현하는 metainfo와 책들의 집합인 books 엘리먼트로 구성되어 있다.

EBKS로 작성된 전자책 문서는 한 권 이상의 책(book)으로 구성될 수 있으며, 이는 모음집(books) 혹은 전집 형태의 서비스를 지원하기 위함이다. 그 결과 그림. 3과 같이 모음집(books)은 cover, front, 하나 이상의 book, back 형태의 기본적인 계층구조를 이루고, 하나의 책(book)은 cover, front, body, back 형태의 계층적 구조를 이룬다.

일반적으로 cover 요소는 종이책의 앞과 뒤 표지를 구성하는 부분이다. 그러나 전자책에서의 커버는 그 형태가 다양하게 변경되어 구성될 수 있으며, 특히 이미지로 구성된 figure 혹은 출판사에 대한 상세 정보들은 필수적으로 포함되어야 한다.

Front는 책의 본문이 시작되기 이전의 머리말, 감사의 글, 서문 등의 요소로 구성되는 부분이다. 한편 책의 콘텐츠 순서를 표현하는 목차(toc)도 front에 나타나며, 특히 EBKS에서의 toc 엘리먼트는 교환시에

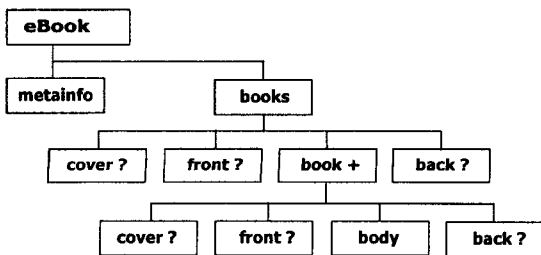


그림. 3 EBKS의 전체 구조

는 공백(EMPTY) 엘리먼트로 처리된다. 이는 목차가 실질적으로 출력 상황에 따라 매우 다양한 형태로 나타날 수 있기 때문에 이 부분은 스타일 시트에 의해 혹은 소프트웨어에 의해 처리되는 것이 바람직하기 때문이다.

Back 요소는 cover, front, body를 제외한 부가적인 콘텐츠를 표기하는 부분으로 부록편(appendix), 용어집(glossary), 색인(index)등이 이 부분에 올 수 있다.

Body는 전자책의 실질적인 콘텐츠를 기술하는 부분으로 서비스하려는 책의 종류에 따라 part, chapter, section 엘리먼트 중 하나를 선택하여 콘텐츠의 Top level 엘리먼트를 구성할 수 있다(그림. 4).

Part(그림. 4) 요소는 콘텐츠의 body부분에서 매우 중요하게 사용될 수 있으며, 그 이유는 대부분의 본문 구조는 Part 요소를 이용하여 상위 레벨의 구조를 정의할 수 있기 때문이다. 또한 Part 요소를 이용한 본문 구조는 대부분 Chapter를 동반하기 때문에 EBKS에서는 이것의 순서를 고정한 것이다.

Chapter(그림. 4)는 일반적으로 어떤 한 주제가 일 단락이 되지만 전후와의 연관성을 갖고 있는 책의 한 부분으로서, 타이틀 성격의 구조 혹은 내부적으로 section 요소를 포함한 경우 등으로 나타난다.

EBKS의 section(그림. 4) 요소는 하부로 총 6 레벨의 subsection으로 구성되며, 이는 대부분의 경우 6레벨의 subsection 구조로 처리가 가능하기 때문이다.

전자책의 구조를 정의할 때에 가장 다양한 구조가 나타나는 부분은 body이하 부분이라 할 수 있다(그림 4의 ? 경우). 따라서 EBKS DTD에서는 body 부분뿐만 아니라 모든 엘리먼트 부분에서 구조를 확장

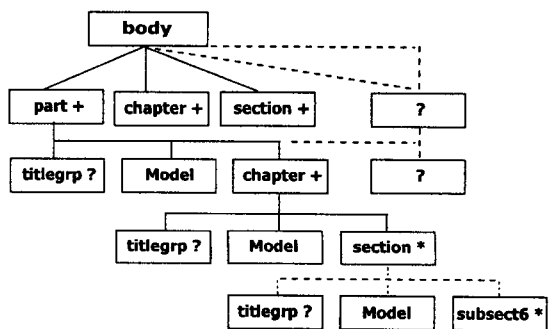


그림. 4 EBKS의 body 구조

할 수 있는 메커니즘을 제공하고 있으며, 그 예는 다음(그림. 5)와 같다.

그러나 무분별한 확장 엘리먼트의 정의는 책의 전체 구조를 위배할 수 있으므로, EBKS에서는 확장 메커니즘을 반드시 필요할 경우에만 사용할 것을 권고하고 있다.

```
<!ENTITY % local.book "" >
<!ELEMENT book (cover?, front?, (body %local.book;), back?) >
```

그림. 5 EBKS에서의 엘리먼트 확장 메커니즘

3.2 Content 엘리먼트의 내용

EBKS의 모든 엘리먼트는 공통속성(Common Attribute)을 갖는다. 그 기능은 가장 먼저 엘리먼트 참조를 위한 id, 엘리먼트 구분을 위한 메타 정보 부여 기능의 role, 사용언어를 뜻하는 lang, 가로, 세로 쓰기를 정의한 hdir, vdir 등이 있다.

Paragraph model은 본문이나 기타 엘리먼트에서의 실제 콘텐츠 데이터를 처리하며, 이를 위하여 #PCDATA 및 여러 inline 요소들 및 float 요소들을 포함하고 있으며, 대부분의 출판물에서 발생하는 논리적, 스타일 요소를 처리할 수 있다.

EBKS에서는 표현이 불가능한 문자들을 xcharacter 엘리먼트로 표현할 수 있다. 이는 한국 출판물에서는 기존 폰트체계로 표현이 불가능한 고어(한문)가 자주 사용되며, 이를 위해서는 자체 폰트가 필요하다. 따라서 이러한 문자표기를 위해서 문서 내에 EBKS 엔터티에 정의되지 않은 특수 문자를 사용하거나, 사용자가 특수 문자를 직접 정의하여 사용하는 경우를 위하여 xcharacter란 엘리먼트를 제공하고, 콘텐츠 교환시 이에 대한 명확한 결과를 보장할 수 있다.

EBKS 문서의 모든 부분에서 발생할 수 있는 요소들은 float 모델로 정의된다. 이는 제한된 구조로 정의하기 어려운 주석, 이미지, 수식, 참조(Reference) 모델 등이 포함된다.

또한 EBKS 문서 내부 및 외부로의 참조를 위한 참조모델은 내부 참조인 xref, 외부 자원 참조를 위한 uri, 등으로 구성된다. EBKS 참조모델은 기존 HTML 및 XML 환경, 그리고 기존 문서 표준과의 호환을 고려하여 W3C의 링크 표기 및 ISO 12083 등의 링크 표기법을 최대한 수용하였다.

리스트는 논리적으로, 또는 스타일 측면에서 매우 중요한 요소이다. 특히 예를 들어 학습서, 교재, 참고서와 같은 출판물들은 많은 수의 리스트 모델로 정의될 수 있기 때문이다. 따라서 EBKS의 리스트 모델은 다양한 리스트 형태를 지원하는 다양한 type 속성을 통하여 정의되며, 특히 기존 웹 문서와의 호환을 고려하여 CSS의 속성과 호환성을 유지하도록 고안되었다.

EBKS의 테이블 모델은 CALS 테이블[27]을 지원한다. CALS 테이블 모델은 Column Specification과 Span Specification을 이용하여 각 Entry에 특정 의미를 부여할 수 있기 때문에 셀에 대한 의미정보 부여, 재사용, 검색 등을 위한 장점을 제공할 수 있다. 따라서 이를 전자책 문서 표준에 활용할 경우 기존 HTML과는 달리 테이블 내용에 대한 명확한 교환을 보장할 수 있으며, 이러한 이유로 EBKS에서 CALS 테이블 모델을 고려한 것이다.

3.3 메타데이터

EBKS 메타데이터는 국내의 메타데이터 표준 및 전자책 메타데이터들을 비교하고 한국 전자책 여건 및 국내에서 사용되고 있는 메타데이터 특성을 고려하여 적합한 메타데이터 표준을 작성하였다. EBKS 메타데이터 표준은 간결성, 사용의 용이성, 일관성, 확장성, 기타 메타데이터와의 상호운용성을 원칙으로 한다. 따라서 위 원칙을 최대한 수용하는 국제 표준인 더블린 코어 메타데이터 표준[17,33-36]을 적용하여, 더블린 코어의 15개 엘리먼트를 반복적이면서 선택적으로 사용하게 된다. 또한 더블린 코어의 한정어는 가능한 사용하지 않도록 하며, 이는 전문가뿐만 아니라 일반인들도 쉽게 메타데이터를 적용할 수 있도록 간결하고 사용이 쉬운 메타데이터를 작성할 수 있도록 하기 위함이다. 그리고 OEB 메타데이터 방식과 같이 x-metadata 엘리먼트를 반복해 사용하여 더블린 코어에서 표현되지 않는 전자책 상거래 및 검색 등을 위한 정보를 추가할 수 있도록 하였다. 따라서 OEB 메타데이터와 완벽한 상호 호환이 가능하며 태그 매치가 불가능한 기타 메타데이터 표준은 확장 태그의 비교를 통하여 그 변환이 가능하다.

3.4 XSL을 이용한EBKS의 스타일 제어

전자책의 스타일과 관련해서는 기존 CSS1[9],

CSS2[11], DSSSL[22], XSL[10]을 살펴볼 수 있다. CSS1은 HTML문서를 웹 환경에서 출력하기 위한 스타일 정보를 정의한 것으로 폰트, 컬러, 행간 등의 정보를 표현할 수 있다.

CSS1은 HTML에서 제공하는 스타일 정보와 일부 중복되지만 효과적인 스타일 표현이 가능하다. CSS1은 서양의 웹 환경만을 고려하였기 때문에 세로쓰기와 다단편집, 페이지 단위 처리 등의 기능은 정의하고 있지 않다.

한편 CSS2는 페이지, 테이블, 청각 스타일 등, CSS1에 비해 좀 더 정밀한 스타일 정보 표현이 가능하며, HTML 뿐 아니라 XML 등 다른 문서 형식에서도 사용 가능한 장점이 있으나, 국내 출판 환경에서 필요한 세로쓰기와 다단 편집 등의 기능은 여전히 지원하지 않고 있지 않다. 이에 반해 XSL은 XML 환경에서 스타일을 작성할 수 있도록 한 것으로 XML 형식으로 표현된다.

XSL은 SGML에서 스타일 정보를 표현하는 DSSSL(Document Style Semantics and Specification Language)과 CSS를 기반으로 만들어졌다. DSSSL은 오프라인 출력을 주목적으로 만들어진 언어로, 출력 스타일에 대한 정밀한 제어가 가능하지만 기능이 다양하여 구현하기가 어렵다. 반면에 CSS는 웹 상의 출력을 목적으로 만들어진 언어로 출력 스타일에 대한 제어가 DSSSL만큼 정밀하지 못하다. 이에 비해 XSL은 DSSSL과 CSS의 중간 수준으로 페이지, 세로쓰기, 다단편집, 정교한 스타일 지정 등 전자책에서 필요한 기본적인 스타일 기능을 제공한다.

XSL은 현재 W3C에서 표준으로 발표되기 바로 전 단계에 있으며 조만간 웹 표준으로 발표될 것으로 예상된다. 따라서 전자책에서도 XSL을 이용한 스타일 처리가 가능하며, 향후 기존 CSS 보다 정교한 출력 스타일 결과를 보장할 수 있기 때문에 EBKS의 출력 스타일은 XSL-FO[10]를 권장하고 있다.

다음 예제 (그림. 6)는 EBKS를 이용하여 작성한 XML 문서에 XSL-FO를 이용하여 다단, 테이블, 리스트 및 다양한 스타일을 출력한 결과를 보여주고 있다.

4. EBKS와 OEB 및 JepaX간의 변환기법

EBKS의 제정 목적은 전자책 콘텐츠에 대한 명확

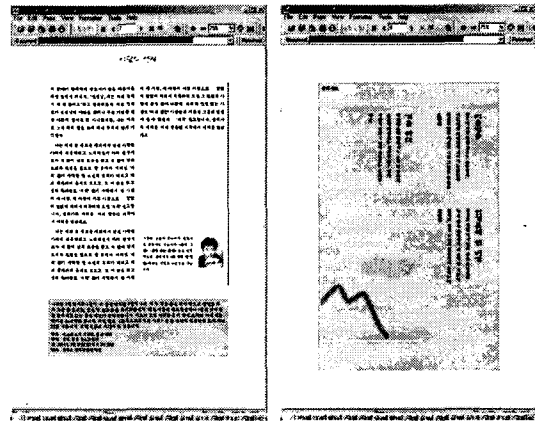


그림. 6 XSL-FO를 이용한 EBKS 문서의 출력 결과(31)

한 교환을 보장하는 것이며 이를 위한 EBKS의 구조는 앞절에서 살펴본 바와 같다. 가장 먼저 EBKS와 OEB, JepaX와 같은 표준문서를 명확히 교환하기 위해서는 각 표준들의 구조 형태 및 특징들을 정확히 파악하여야 한다. 그 이유는 각 표준들의 제정 목적에 따라 그 구조가 상이하기 때문이다. 따라서 이러한 환경에서 각 표준간에 명확한 교환을 위해서는 먼저 각 표준들의 구조를 상세히 분석하여 실제 변환시 이를 반영하여야 하며, 특히 변환시에 발생할 수 있는 구조 정보의 손실을 최소화 하여야 한다. 본 장에서는 이러한 EBKS와 OEB 및 JepaX간의 변환 및 그 역변환을 위한 고려 사항 및 기법을 제안하며, 차후 그 실제 결과에 대해서도 살펴보기로 한다.

4.1 EBKS와 OEB간의 변환

4.1.1 EBKS와 OEB와의 구조적 차이점

EBKS는 명확한 논리적인 구조를 포함한다. 이에 비하여 OEB는 문서의 구조가 기본 OEB 문서(Basic OEB Document)와 확장 OEB 문서(Extended OEB Document)로 구분된다. 기본 OEB 문서의 경우 HTML 4.0에 기반한 XHTML로 구성되어 있어 문서에 대한 명확한 논리 구조가 부족하다. 한편 확장 OEB 문서는 OEB스펙에 정의되어 있지 않은 엘리먼트 및 속성을 사용하거나 기본 OEB 문서 DTD이외의 다른 DTD를 사용하는 경우로서, 사용하는 DTD에 따라 그 구조가 달라진다. 이러한 결과 EBKS를 OEB로 변환하고자 할 경우, 먼저 확장 OEB 문서의 경우에는 EBKS DTD를 그대로 사용할 수 있기 때문

에 별도의 변환 작업이 필요하지 않다. 즉, EBKS 문서에 OEB의 Package 정보를 추가하면 간단한 확장 OEB 문서가 될 수 있다. 따라서 다음절에서 살펴볼 “EBKS에서 OEB로의 변환”이란 의미는 기본 OEB 문서로의 변환을 뜻한다.

4.1.2 EBKS에서 OEB로의 변환

먼저 메타데이터 변환은 EBKS와 OEB가 모두 더블린코어를 사용하므로 EBKS의 메타데이터와 OEB Package의 metadata 매핑 작업은 비교적 용이하다고 할 수 있다. 이를 위해서는 기본 OEB 문서의 <meta>태그 정보 및 OEB Package의 메타데이터 정보를 활용하는 것이 가장 바람직하다.

앞서 살펴본 EBKS와 OEB의 차이점에서 알 수 있듯이 EBKS에서 OEB로의 변환은 구조가 있는 문서에서 구조가 없는 문서로의 변환이다. 따라서 EBKS문서가 갖고 있던 문서의 논리적 구조 정보는 XHTML기반의 기본 OEB 문서로 변환됨과 동시에 사라지게 된다. 이때에 만약 구조 정보를 최대한 유지하기 위해서는 EBKS를 확장하여 <div> 및 태그로 구조 정보를 표현할 수도 있다.

그러나 위와 같은 방법을 사용하더라도 실제 변환 시에는 각 구성요소들을 한번에 완벽히 변환하는 것은 불가능하다. 예를 들어 EBKS의 <title>의 경우, OEB에서 <h1>~<h6>중 어느 것을 변환 타겟으로 설정해야 하는지는 불명확할 수 있으며, 또한 문서 구조 표현이 사용자마다 다를 수 있으므로 본 논문에서는 먼저 기본적인 변환 스크립트를 통한 1차 변환과, 2차적으로 상세 구조에 대한 커스터마이징과정을 수행하였다. 이를 위하여 본 사례에서는 W3C의 XSLT(XSL Transformations) 1.0 스펙[12]을 이용하여 변환 내용을 기술하였고, 기술된 XSLT 내용의 실행 및 결과 확인을 위해 MSXML3.0 및 MSXSL을 사용하였으며, 샘플 문서는 영어 교재 1권[32]을 사용하였다. 이러한 결과는 다음 그림. 7, 그림. 8과 같다.

4.1.3 OEB에서 EBKS로의 변환

앞서 살펴본 변환의 예는 명확한 구조로부터 구조가 없는 문서로의 변환(Down-Translation)의 경우로서, 구조정보를 상실하는 문제점은 있으나 변환 자체는 큰 무리 없이 가능하다는 점을 확인하였다. 그러나 구조가 없는 문서로부터 구조가 있는 문서로의

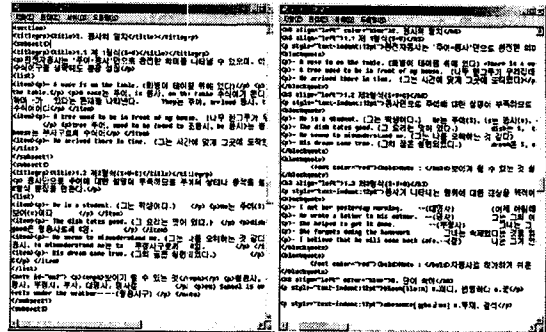


그림. 7 EBKS 원본 문서 및 변환된 OEB 문서

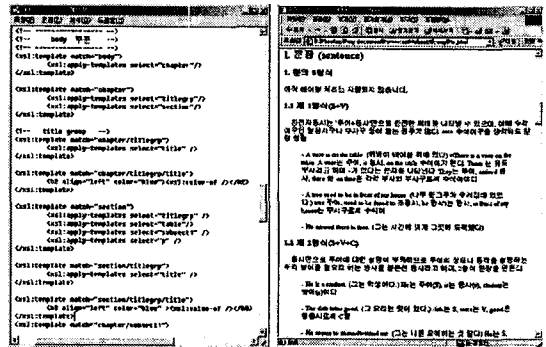


그림. 8 EBKS문서를 기본 OEB 문서로 변환하는 XSLT 내용 및 변환된 OEB 문서의 출력 결과

변환인, OEB에서 EBKS로의 변환(Up-Translation) 시에는 완전 자동변환이 불가능하다. 그 이유는 기본 OEB 문서에 구조적 정보 및 구조적 정보로의 변환을 위한 판단 요소 및 정보가 부족하기 때문이다. 예를 들어 기본 OEB 문서에 나열되어 있는 <p> 엘리먼트를 EBKS의 <chapter>로 변환해야 할지 혹은 <section>으로 변환해야 할지는 사람이 직접 읽고 판단하는 수 밖에 없기 때문이다. 따라서 이러한 경우에는 앞서 서술한 바와 같이 EBKS 구조를 이나 <div> 형태로 표현하는 방법을 적용할 수도 있다.

4.1.4 결과 분석

앞서 살펴보았듯이 EBKS와 OEB간에는 다음 표. 1과 같은 구조적인 차이점을 포함하고 있으며 이에 따른 각각의 변환 전략이 요구된다.

EBKS에서 OEB로의 변환시 메타데이터의 경우 다음 표. 2의 내용과 같이 두가지 표준 모두 더블린코어를 지원하기 때문에 확장 태그인 X-Metadata를

표. 1 EBKS와 OEB간의 구조적 특성

	EBKS	OEB
메타데이터	더블린 코어, 확장방법 제공	더블린 코어, 확장방법 제공
문서	명확한 논리적 구조	기본문서 : 명확한 구조 없음
		확장문서 : 임의의 구조

표. 2 EBKS에서 OEB로 변환시 발생하는 매핑 요소간의 관계

	EBKS 요소	OEB 요소	특징
메타데이터	기본 요소	기본 요소	더블린 코어 기본 엘리먼트는 1:1 매핑이 가능.
	X-Metadata	X-Metadata	EBKS 확장 태그를 더블린 코어 확장형식으로 변환.
문서	공통요소	HTML 요소	EBKS의 구조모델 중 HTML과 중복되는 논리적 구조, 콘텐츠 엘리먼트는 1:1 매핑이 가능.
	논리적구조	<div "id=ebks 엘리먼트">, 	EBKS의 논리적 구조를 HTML의 div 및 span 속성으로 변환하여야 함.
	컨텐츠 엘리먼트	해당 없음	공통요소에 포함되지 않은 콘텐츠 엘리먼트는 수작업을 통하여 HTML 태그와 매핑하여야 함.
	참조모델	HTML 링크모델	EBKS의 참조모델을 HTML의 링크모델로 하향 변환하여야 함.
	테이블	HTML 테이블모델	CALS 테이블 모델을 HTML 테이블 모델로 하향 변환하여야 함.

사용하지 않는다는 조건하에는 항상 1:1 매핑이 가능하다. 또한 EBKS에서 사용된 확장 메타데이터 모델의 경우 더블린 코어에서도 확장 모델을 제공하기 때문에 이에 알맞은 형식으로만 변환한다면 자동적인 변환 작업을 수행할 수 있다.

EBKS의 문서 모델은 기본적으로 하향식 변환의 경우이기 때문에 대부분의 경우 OEB 형식에 알맞은 새로운 태그를 생성하여야 한다. 이때에 다음 표. 2와 같이 EBKS와 OEB 모델간에 서로 공통적으로 발생하는 요소들은 1:1 매핑을 통하여 자동적으로 변환이 가능하며, 동시에 논리적 구조 또한 HTML의 <div>나 태그의 속성을 이용한다면 기술적으로 큰 어려움 없이 변환작업을 수행 할 수 있다. 그러나 EBKS 및 OEB 요소 어느 부분에도 일치하지 않는 모델과, <div>나 으로 변환된 EBKS의 논리적 요소 중, HTML의 스타일 요소와 연관된 경우는 기계적인 변환 작업이 불가능하다. 따라서 이러한 경우는 반드시 수작업을 통하여 적절한 태그를 매핑하여야 한다. 기타 참조 모델 및 테이블 모델 등은 각각 HTML의 링크 모델 및 테이블 모델로 기계적인 하향 변환이 가능하다.

OEB에서 EBKS 문서로의 변환은 상향식 변환의

관계이기 때문에 다음 표. 3과 같이 이미 공통적으로 정의된 모델 혹은 <div> 속성으로 정의되어 자동으로 변환가능한 경우를 제외하고는 모든 경우를 수작업으로 문서를 변환하여야 한다. 따라서 이러한 수작업을 최대한 줄이기 위해서는 HTML 모델의 의미적 정도를 추출하여 이를 EBKS의 구조와 매핑하는 기법 또는 전문가가 작성한 변환용 주석 처리 등을 통하여 1차적인 기계 변환 후에 최종적으로 수작업을 행하는 것이 가장 바람직하다.

4.2 EBKS와 JapaX간의 변환

4.2.1 논리 구조 정보를 갖는 문서간의 변환

JapaX는 일본 전자책 표준 DTD로서 div태그의 type속성에 태그의 의미 정보를 기술하는 방식을 사용하고는 있으나, 문서에 대한 논리적 구조를 정의하고 있다고 볼 수 있다. 따라서 EBKS와 JapaX간의 변환은 앞서 살펴본 EBKS와 OEB간의 변환과는 달리 양쪽 모두 구조 정보를 갖고 있는 문서간의 변환이라 할 수 있다.

한편 위와 같이 각각 구조 정보를 포함하고 있는 경우에는 두 문서의 구조가 서로를 얼마나 수용하는

표. 3 OEB에서 EBKS로 변환시 발생하는 매핑 요소간의 관계

	OEB 요소	EBKS 요소	특징
메타데이터	기본 요소	기본 요소	더블린 코어 기본 엘리먼트는 1:1 매핑이 가능.
	X-Metaddata	X-Metaddata	OEB 확장 태그를 EBKS 메타 확장형으로 변환.
문서	공통요소	EBKS 요소	EBKS의 구조모델 중 HTML과 중복되는 논리적 구조, 콘텐츠 엘리먼트는 1:1 매핑이 가능.
	<div "id=ebks 엘리먼트">, 	논리적구조	HTML의 div 및 span 속성을 EBKS의 논리적 구조로 변환하여야 함.
	공통 요소 및 <div> 속성으로 표기된 경우를 제외한 모든 HTML 태그	해당 없음	수작업이 필요함. 최소한의 작업로드를 보장하기 위해서는 별도의 의미적 변환 모델 혹은 주석처리 등을 이용한 변환기법 개발이 요구됨.

가에 따라 변환 후 구조 정보의 유지 정도가 달라진다. 즉 A라는 구조의 문서를 B라는 구조의 문서로 변환할 경우에, A구조에 존재하는 "vita"라는 엘리먼트가 B구조에 존재하지 않는다면 표현이 불가능하거나 비슷한 다른 엘리먼트로 대체할 수 밖에 없으므로 본래의 의미 정보를 상실하게 된다. 그러나 임의의 구조를 확장할 수 있는 확장 메커니즘을 서로간에 제공할 수 있다면 이를 어느 정도 극복할 수가 있다. 더욱이 EBKS와 JepaX 모두 문서 구조에 대한 확장 메커니즘을 제공하고 있기 때문에 이를 효과적으로 응용 가능하며, 이러한 내용은 다음절에서 설명한다.

4.2.2 EBKS에서 JepaX로의 변환

구조적 정보를 갖고 있는 문서간의 변환은 두 문서 구조에 대한 비교 분석 및 매핑 테이블이 매우 중요하다. 실제로 EBKS 및 JepaX 구조를 분석한 결과, EBKS나 JepaX 모두 책을 정의한 구조이므로 두 구조간에 일대일 매칭되는 부분이 많음을 알 수 있었다. 그 예로 cover, front, body, back 등은 동일하였고, 메타데이터의 경우에도 bookinfo내의 엘리먼트들과 상당수 매칭되었으며, 그 내용을 다음 표. 4에서 살펴볼 수 있다.

표. 4 EBKS와 JepaX의 메타데이터에 대한 매핑테이블

EBKS의 Metadata	JepaX의 Bookinfo
dc:Identifier	<isbn>
dc:Title	<book-title>
dc:Creator	<book-author>
dc:Date	<pub-date>
dc:Publisher	<publisher>

EBKS는 JepaX 구조에 비하여 보다 다양한 구조를 포함하고 있으며, 결과적으로 JepaX에는 없는 다수의 엘리먼트가 존재한다. 예를 들면, foreword, preface, dedication, abstract, othersect 등의 경우이며, 따라서 본 연구에서는 이를 JepaX 구조에 반영하기 위하여 1차적으로 EBKS의 구조와 JepaX의 매핑 테이블을 기준으로 각 구조의 유무를 판단하였으며, 2차적으로 그 결과를 JepaX의 확장 태그를 이용하여 문서를 자동 변환하였다. EBKS를 JepaX로의 변환하기 위하여 본 논문에서는 ASP(Active Server Pages)와 MSXML 3.0을 사용하였으며 다음 그림. 9에서 자동으로 JepaX에서 foreword 태그를 확장한 경우를 설명하고 있다.

이를 바탕으로 본 논문에서는 각각 산문과 운문의 책을 이용하여 EBKS를 JepaX로 변환하였으며 그 결과는 다음 그림. 10과 같다.

4.2.3 JepaX에서 EBKS로의 변환

JepaX에서 EBKS로의 변환 역시 각 DTD에 대한 정확한 분석을 바탕으로 맵핑테이블을 구성하여야 한다. JepaX의 cover, front, back, 기본적인 div 엘리먼트 등은 대부분 EBKS와 매핑이 가능하나, JepaX에서 확장한 구조가 존재하는 경우에는 EBKS의 어

```

<xtype-desc element="div" value="foreword"/>
Foreword Part of EBKS
<xtype-desc/>

<!------- Omitted ----->

<div xtype="foreword">
  <head> <title>감역자의 말 </title> </head>
</div>
    
```

그림. 9 EBKS를 JepaX의 "forewor"로 자동 확장한 결과

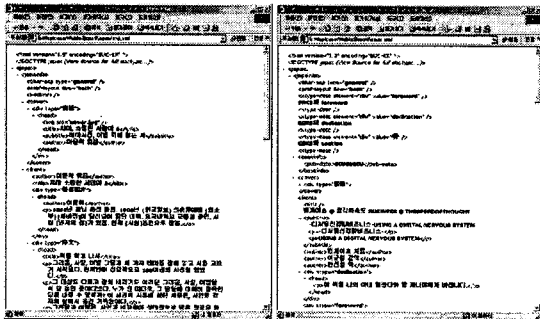


그림. 10 운문 및 산문인 EBKS 문서를 JepaX로 변환한 결과

는 엘리먼트와도 매칭되지 않는 경우가 발생할 수 있다. 이 경우는 EBKS의 DTD 확장 메커니즘을 통해 해결하여야 하며, 이를 위해서는 변환하려는 JepaX문서의 확장 태그의 의미를 미리 판단하여 EBKS변환에 반영하여야 한다. 또한 JepaX에서 EBKS로의 변환시 고려해야 하는 점은 JepaX에서 사용되는 물리적 스타일 요소들과 EBKS의 논리적 스타일의 매칭 방법이다. 따라서 이를 해결하기 위한 가장 간단한 방법은 추후 출력 부분에서 손실된 스타일 정보를 직접 보완하는 것이며, 앞으로 이에 대한 보다 심도 있는 연구가 요구된다.

4.2.4 결과 분석

EBKS에서 JepaX로의 변환은 특별한 경우를 제외하고는 다음 표. 5의 경우와 같이 대부분 기계적으로 변환이 가능하다. 이는 각 표준간에는 확장 메커

니즘을 제공하기 때문이며 각각의 확장 메커니즘에 기반한 통합 모듈을 통하여 적절한 변환결과를 얻을 수 있었다. 먼저 각 표준들의 메타데이터는 외형상 유사한 부분들이 다수 포함되나 세부적으로는 1:1의 형태로 매핑되지는 않으며, 특히 확장된 EBKS의 메타 모델의 경우는 JepaX의 <Bookinfo> 모델과 전혀 매핑되지 않는다. 그러나 앞서 언급한 바와 같이 JepaX의 <X-type>이라는 확장 모델을 통하여 <div id="EBKS 확장 메타모델">의 형태로 기계적인 변환이 가능하다. 따라서 논리적으로는 모든 EBKS의 모델을 JepaX로 변환이 가능하다고 볼 수 있으나, 실제 변환 과정시 상당히 복잡한 변환 절차가 요구되며, 생성된 JepaX 형태의 구조는 다소 산만한 문서 구조를 포함할 수도 있다. 따라서 이러한 문제를 최소화하기 위해서는 확장 태그를 생성하는 과정시 반드시 EBKS의 구조와 해당 JepaX의 모델을 주석 처리하여 저장할 수 있어야 한다.

한편 JepaX에서 EBKS로의 문서 변환시 공통 요소간의 변환은 기계적으로 완벽히 변환 가능하다. 그러나 다음 표. 6과 같이 메타데이터 및 문서 구조간에서 JepaX에서 확장한 모델의 경우 EBKS 모델과 전혀 매핑되지 않기 때문에 EBKS의 확장 메커니즘을 이용하여 이를 반영하여야 한다. 그러나 EBKS의 확장 모델은 JepaX와는 달리 인스턴스가 아닌 DTD 차원에서 확장되어야 하기 때문에 반드시 전문가의 수작업을 통한 변환 절차가 요구된다. 따라서 이때에 발생하는 로드를 최소화하기 위해서는 DTD 확장시 요구되는 오류 발생을 최소화하고 시각적으로 효과

표. 5 EBKS에서 JepaX로 변환시 발생하는 매핑 요소간의 관계

	EBKS 요소	JepaX 요소	특징
메타데이터	더블린 코어	자체 모델	대부분의 경우 더블린 코어 기본 엘리먼트와 1:1 매핑이 가능하나, 그 외의 경우는 JepaX의 확장태그를 통하여 새로운 엘리먼트를 생성하여야 함.
	X-Metadata	자체 확장모델	EBKS 메타 확장 태그를 JepaX 확장형으로 변환.
문서	공통요소	공통요소	EBKS의 구조모델 중 JepaX 모델과 중복되는 논리적 구조, 콘텐츠 엘리먼트는 1:1 매핑이 가능.
	논리적구조	<div "id=ebks 엘리먼트">	EBKS의 논리적 구조를 JepaX의 확장 구조로 변환하여 적절한 <div> 태그를 자동으로 생성가능.
	기타 요소	해당 확장 엘리먼트	매핑 테이블에 존재하지 않는 모델은 자동적으로 확장 태그로 변환하고, 기타 테이블, 참조 모델 등은 JepaX의 구문을 따르는 형태로 변환할 필요가 있음.

표. 6 JepaX에서 EBKS로 변환시 발생하는 매핑 요소간의 관계

	JepaX 요소	EBKS 요소	특징
메타데이터	<bookinfo>	더블린코어	1:1 매핑이 가능한 경우를 제외하고는 모두 EBKS의 구조를 확장하여야 한다.
	자체 확장모델	X-Metadata	JepaX 확장 모델을 EBKS 형으로 변환.
문서	공통요소	공통요소	EBKS의 구조모델 중 JepaX 모델과 중복되는 논리적 구조, 컨텐츠 엘리먼트는 1:1 매핑이 가능.
	확장구조	DTD 확장	EBKS DTD에 존재하지 않는 JepaX의 모델을 확장. 이때에 DTD 수정은 기계적으로 불가능함.
	기타 요소	해당 확장 엘리먼트	매핑 테이블에 존재하지 않는 모델은 수동적으로 DTD를 확장하고, 기타 테이블, 참조 모델 등은 EBKS의 구조에 따르는 형태로 직접 변환.

적인 별도의 인터페이스가 반드시 필요하다.

참고 문헌

5. 결 론

EBKS는 전자책 컨텐츠에 대한 정확한 교환을 목적으로 제정되었으며 현재 대한민국 산업 표준인 KS 6100으로 공식된 바 있다. EBKS는 미국 OEB PS와 일본 JepaX와는 달리 고정된 문서구조를 포함하고 있으며, 한국에서 발간되는 대부분의 책의 구조를 지원하고 있다. 또한 한국 문헌의 특징을 최대한 고려하였으며, 다양한 문서 구조를 반영하기 위한 구조 확장 방법 등을 제공한다.

한편 EBKS는 컨텐츠의 명확한 교환을 목적으로 제정되었으며, 실제로 EBKS문서간의 교환시에는 이에 대한 정확한 결과를 보장한다. 그러나 EBKS와 OEB 및 JepaX와 같은 서로 다른 전자책 표준들은 그 제정 목적에 따라 서로 상이한 구조를 포함한다. 따라서 이러한 표준간의 정확한 변환시에는 각각 상이한 구조간의 특성을 반영한 변환 전략이 필요하며, 이를 통하여 정확한 변환 결과 및 구조 정보에 대한 최소한의 손실을 보장할 수 있어야 한다.

이를 위하여 본 논문에서는 비구조문서에서 구조 문서로의 변환 기법 및 역변환시에서 발생할 수 있는 정보 손실을 최소화하기 위한 다양한 변환 사례를 제시하였고, 본 과정에서 발생하는 문제점 및 향후 개선되어야 할 부분에 대해서도 살펴보았다. 그 결과 본 연구의 결과를 통하여 앞으로 전자책 컨텐츠의 교환시 이를 적절히 적용할 수 있을 것으로 기대되며, 표준이외의 다른 포맷간에도 효과적으로 응용될 수 있을 것으로 예상된다.

[1] Ministry of Culture & Tourism, Republic of Korea, A Study of developmental plan for Korean eBook industry, Technical Report, 2000.

[2] Ministry of Culture & Tourism, Republic of Korea, The Goal of e-Book development in the digital age, Technical Report, 2000.

[3] EBK (e-Book of Korea) Consortium, A Study of Korean Standardization of eBook documents, Technical Report, 2001.

[4] EBK (e-Book Korea) Consortium, <http://www.ebk.or.kr>, 2001.

[5] Working Group of EBK Standardization Committee, <http://orange.yonsei.ac.kr/ebook>, 2001.

[6] W3C Consortium, Extensible Markup Language (XML) 1.0, <http://www.w3.org/TR/1998/RECxml-19980210>, 1998.

[7] W3C Consortium, Hypertext Markup Language (HTML) 4.0, <http://www.w3.org/TR/REChtml40-971218>, 1997.

[8] W3C Consortium, eXtensible Hypertext Language (XHTML), <http://www.w3.org/TR/2000/REC-xhtml1-20000126>, 2000.

[9] W3C Consortium, Cascading Style Sheets (CSS) level 1.0, <http://www.w3.org/TR/REC-CSS1-961217>, 1996.

[10] W3C Consortium, Extensible Stylesheet Language (XSL), <http://www.w3.org/TR/2000/>

- CR-xsl-20001121, 2000.
- [11] W3C Consortium, Cascading Style Sheets level 2 CSS2, <http://www.w3.org/TR/1998/REC-CSS2-19980512>, 1998.
- [12] W3C Consortium, XSL Transformations (XSLT), <http://www.w3.org/TR/1999/REC-xslt-19991116>, 1999.
- [13] Adobe Corporation, Portable Document Format (PDF), <http://www.adobe.com/products/acrobat/-main.html>, 2001.
- [14] Open eBook Forum (OEBF), <http://www.openebook.org>, 2001.
- [15] Open eBook Forum (OEBF), Open eBook Publication Structure 1.0, <http://www.openebook.org>, 1999.
- [16] Japanese Electronic Publishing Association (JEPA), <http://www.jepa.or.jp>, 2001.
- [17] Dublin Core Metadata Element Set, Version 1.1: Reference Description, Internet RFC 2413, <http://dublincore.org/documents/1999/07/02/dces>, 1999.
- [18] N. Ide and J. Klavans, The text encoding initiative guidelines and their application to building digital libraries, in : Proc. ICDL 96 Bethesda, MD USA (ACM March, 1996).
- [19] N. Walsh and L. Muellner, DocBook, (OREILLY, CA, 1999).
- [20] Information and Documentation - Electronic Manuscript Preparation and Markup, ISO 12083 DTD, International Organization for Standardization, 1993.
- [21] The Unicode Consortium, Unicode, The Unicode Standard Version 2.0 Reading, Mass, (Addison-Wesley, Developers Press, 1996).
- [22] Information Technology - Text and Office Systems - Document Style Semantics and Specification Language (DSSSL), ISO/IEC 10179, International Organization for Standardization, 1996.
- [23] Yergeau, F., UTF-8, a transformation format of ISO 10646, RFC 2279, 1998.
- [24] Hoffman, P. and F. Yergeau, UTF-16, an encoding of ISO 10646, RFC 2781, 2000.
- [25] Alvestrand, H., Tags for the Identification of Languages, BCP 47, RFC 1766, 1995.
- [26] ISO 639:1988 (E/F) - Code for the representation of names of languages - The International Organization for Standardization, 1st edition, 1988.
- [27] Harvey Bingham, CALS Table Model Document Type Definition, OASIS Technical Memorandum TM 9502:1995, <http://www.oasis-open.org/specs/a502.htm>, 1995.
- [28] Beverly L. Harrison, E-Books and the Future of Reading, in : Proc. Computer Graphics and Applications, vol. 20, no. 3, (IEEE, 2000) 32'39.
- [29] Bill N. Schilit, Morgan N. Price, Gene Golovchinsky, Kei Tanaka, and Catherine C. Marshall, "The Reading Appliance Revolution", Computer. Vol. 32, No. 1, (IEEE, 1999) 65'73.
- [30] Bill N. Schilit, Gene Golovchinsky, and Morgan N. Price, Beyond Paper: Supporting Active Reading with Free Form Digital Ink Annotations, In : Proc. Int'l Conf. CHI, (ACM, New York, 1998) 249'256.
- [31] Digital Samto, <http://www.samto.com>, 2001.
- [32] Park Nam Hei, Common English, (Jung Hun Sa, Seoul, ISBN : 8988853024, 1998).
- [33] Preben Hansen, User guidelines for Dublin core creation, http://www.sics.se/~preben/DC/DC_guide.html, 1997.
- [34] Online Computer Library Center (OCLC), <http://www.oclc.org/home>, 2001.
- [35] National Centre for Supercomputer Applications (NCSA), <http://www.ncsa.uiuc.edu>, 2001.
- [36] American National Standards Institute (ANSI) - Dublin Core Metadata Element Set : Z39.85-2001, <http://www.niso.org/news/releases/PRDubCr.html>, 2001.
- [37] Donald E. Knuth, Computers & Typesetting, Volume B: TeX: The Program, Massachusetts, (Addison-Wesley, 1986).

[38] Macromedia, Flash 5, <http://www.macromedia.com>, 2001.

[39] 대한민국 전자책 문서표준형 정의 : KS 6100, 산업표준원, 2002. 5.



손 원 성

1998년 동국대학교 컴퓨터공학과 졸업(학사)
 2000년 동국대학교 컴퓨터공학과 졸업(석사)
 2001년 현재 연세대학교 컴퓨터과학과 박사과정

관심분야 : 웹 Annotation 생성 및 응용



고 승 규

1992년 연세대학교 컴퓨터과학과 졸업(학사)
 1994년 연세대학교 컴퓨터과학과(공학석사)
 2002년 현재 연세대학교 컴퓨터과학과(공학박사)

관심분야 : 웹문서 통합, 변환, 검색

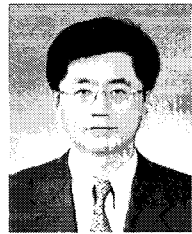


이 경 호

1995년 2월 연세대학교 전산과학과 졸업(이학사)
 1997년 2월 연세대학교 컴퓨터과학과(공학석사)
 2001년 2월 연세대학교 컴퓨터과학과(공학박사)
 2001년 4월~2002년 8월 National

Institute of Standards and Technology(NIST) 객원연구원.

2002년 9월~현재 연세대학교 컴퓨터산업공학부 조교수
 관심분야 : XML 기반 멀티미디어 문서 처리



임 순 범

1982년 서울대학교 계산통계학과 졸업(학사)
 1983년 한국과학기술원 전산학과 졸업(석사)
 1992년 한국과학기술원 전산학과 졸업(박사)
 1989년~1992년 (주)휴먼컴퓨터

이사/연구소장

1992년~1997년 (주)삼보컴퓨터 부장
 1997년~2001년 건국대학교 컴퓨터과학과 조교수
 2001년~현재 숙명여자대학교 멀티미디어학과 조교수
 관심분야 : 컴퓨터 그래픽스, 멀티미디어 응용, 전자출판(폰트, 전자책, 사이버교재)



최 윤 철

1973년 서울대학교(학사)
 1975년 Univ. of Pittsburgh(석사)
 1976년 Univ. of California, Berkeley(석사)
 1979년 Univ. of California, Berkeley(박사)
 1979년~1982년 Lockheed 사 및

Rockwell Internatinal 사 연구원

1990년~1991년 University of Massachusetts 교환교수
 1984년~현재 연세대학교 컴퓨터과학과 교수
 2002년~현재 일본 게이오대 방문교수
 관심분야 : 멀티미디어와 웹, 멀티미디어 문서처리, 3D 사용자 인터페이스, 아바타 인터페이스, 컴퓨터 그래픽스, eLearning 및 Cyber Class

교 신 저 자

손 원 성 120-749 서울시 서대문구 신촌동 134 연세대학교 공과대학 컴퓨터 산업공학부 대학원 멀티미디어 연구실