# 멀티미디어 기술문서를 위한 자동 XML 변환기 개발

°박 건 일, 김 유 성
인하대학교 전자계산공학과

# Development of Automatic XML Converter for Multimedia Technical Documents

°Keon-Il Park, Yoo-Sung Kim
Dept. of Computer Science & Engineering, INHA University

**요 약**

전자도서관과 같은 문서 정보 검색 시스템의 구축을 위한 중요한 요소 기술은 지금까지 출판물로 만들어 놓은 기존의 방대한 자료와 이미 다양한 형식으로 전자문서화 되어 있는 문서정보를 사용자에게 얼마나 단일 형식으로 통일화시켜 효과적으로 제공할 수 있는가이다. 본 논문은 이러한 문제에 있어서 표준화된 단일 전자문서 형식으로 XML 문서를 적용시키기 위해 국립 중앙 도서관 표준 SGML DTD를 XML DTD로 재 정의한 후, 일반적이고, 다양한 특성을 지닌 멀티미디어 기술문서를 표준화된 XML 문서로 자동 변환하는 자동 XML 변환기를 개발하는 것을 목적으로 하고 있다. 자동 XML 변환기는 다양한 문서형식의 전자문서를 표준화된 XML문서로 자동변환함으로써 문서 정보검색 시스템에서의 문서정보의 교환, 저장방법상의 표준화 및 문서형식의 단일화를 제공해 줄 수 있다.

## 1. 서 론

전자도서관과 같은 기술 문서 검색 시스템에서 중점적으로 해결해야 할 과제는 일관된 형식의 전자 기술문서에 대한 효과적인 검색 및 저장 방식과 기존의 출판문서 및 다양한 형식의 전자문서들을 얼마만큼 사용자에게 단일형식으로 구조적인 문서정보를 제공해 줄 수 있느냐이다. 지금까지 각 전자도서관 구축 기관에서 다양한 형식의 전자문서들을 단일한 형식으로 사용자에게 제공하기 위해 사용하고 있는 방법들은 다음과 같다. 첫째, 도서관 출판자료를 스캔된 이미지로 관리, 저장하여 사용자에게 이미지 형태의 화일로 제공하는 방법과 둘째, 출판자료와 같은 문서자료를 SGML 또는 XML과 같은 표준화된 문서형태로 사용자가 재 입력하는 방법이 있다.

두 방법 모두 각각 장·단점을 가지고 있다. 첫 번째 방법은 출판자료를 전자문서로 만드는 데에는 두 번째 방법보다 인력·비용 면에서는 효율적이지만, 이미지 기반의 화일이 가지고 있는 편집상의 어려움, 통신상에서의 속도 문제, 디스플레이 문제 등이 난관으로 대두된다. 이에 비해 두 번째 방법은 앞서 말한 첫 번째 방법보다 통신상에서의 속도, 편집/수정 문제, 디스플레이에 대한 문제가 적고 표준화 및 단일화된 문서 제공의 장점이 있지만 기존의 출판자료나 이미 만들어져 있는 문서를 표준화된 형태의 SGML이나 XML로 만들기 위해서는 사용자의 작업이 필요하다는 인력 및 시간 낭비의 단점이 있다.

본 논문에서는 위의 두 가지 기술문서 검색 시스템의 문서 제공 방법 중 두 번째 방법이 가지고 있는 사용자 수작업 문제를 해결하고, 표준화 및 구조화된 형식의 문서인 XML 문서를 자동으로 변환하여 사용자에게 제공해 줄 수 있는 자동 XML 변환기 개발을 그 목적으로 한다.

본 논문의 구성은 다음과 같다. 제 2장에서는 멀티미디어 기술문서를 위한 구조화 정보로서 국립 중앙 도서관 표준 SGML 형식인 TOC DTD를 재 정의한 XML TOC DTD 설계에 대해서 기술하고, 3장에서는 멀티미디어 기술문서를 위한 자동 XML 변환기의 전체 개요와 구조를 기술한다. 그리고 마지막으로 제 4장에서는 앞의 장들에서 기술한 내용에 대한 간단한 결론 및 향후 연구 방향을 기술한다.

## 2. 멀티미디어 기술문서를 위한 DTD 설계

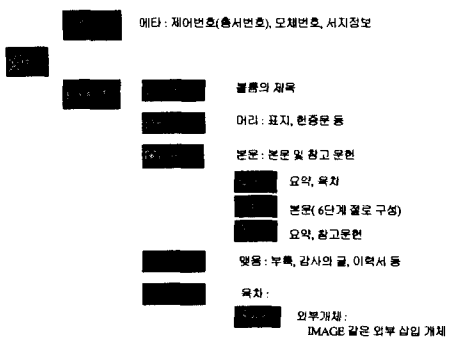
현재까지 국내에서는 전자기술문서의 표현 표준을 위해 국립 중앙 도서관에서 1999년 4월에 SGML TOC DTD를 정의하여 사용하고 있다[1][2]. 따라서 본 연구에서는 메타 언어의 새로운 표준으로 제정된 XML언어를 이용하여 멀티미디어 기술문서를 위한 XML TOC DTD를 정의하였다. 아울러 설계 내부적으로는 XML TOC DTD뿐만 아니라 기존의 SGML TOC DTD 또한 사용가능하게 하였다. 기술 원문서를 효과적으로 구조화하기 위한 기술문서 구조체계는 [그림 1]과 같다.

XML TOC DTD는 원문 이미지 또는 디지털 원문의 목차를 원문 페이지 단위로 기술하기 위해 만들어졌다. 머리, 본문, 맺음 및 본문의 장과 절 등 문서의 논리적 구조를 표현할 수 있도록 하였으나 원문이 페이지라는 물리적인 단위로 구성되어 있으므로 기본적인 단위로 페이지라는 물리적 단위를 기술할 수 있도록 하였다. 본문은 논리적인 구조를 나타낼 수 있도록 6단계의 깊이를 지원하도록 하였고 목차는 본문이 지원하는 6단계의 깊이보다는 일반적으로 적은 깊이로서 나타내므로 4단계의 깊이만을 지원하도록 하였다[1].

XML 문법과 연결해 보면, XML TOC DTD는 IE 5.0에서 사

용하고 있는 파싱방법을 기초로 하고 있으므로 구문에 대한 오류 체크도 그것에 기초한다. 루트태그인 TOC 태그만을 제외하고는 기술 원문서의 텍스트 정보 유무에 따라 모든 태그를 생략 가능하게 하였지만 정보가 있는 경우에는 해당 정보에 대한 태그는 반드시 시작태그와 종료태그가 같이 존재해야 하는 즉, 쌍을 이루는 구조로 되어 있다. 또한 기술 원 문서 텍스트 정보의 중복 가능여부로 인해 모든 태그를 0번 이상 반복 가능하게 하였다. 그리고 목차부분은 페이지 목차뿐만 아니라 그림 목차, 표 목차 등도 기술하게 하였으며, XSL로의 정확한 결과 출력을 위해 볼륨이나 페이지, 절 등의 제목부분을 나타내는 TI 태그에는 FontName과 FontSize 속성을 두어 기술 원문서의 뷰와 XML 문서 뷰의 결과가 시각적으로 차이가 없도록 하였다[4].

TOC DTD에 맞게 작성된 XML 문서는 Internet Explorer 5.0 웹 브라우저에서 정상적으로 출력될 수 있도록 하였다.

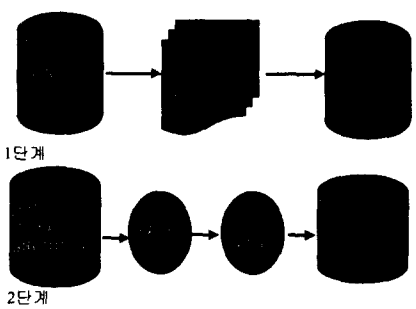



[그림 1] XML TOC DTD 구조

## 3. 자동 XML 변환기

### 3.1 자동 XML 변환기 구조

멀티미디어 기술문서를 위한 자동 XML 변환기의 전체적인 흐름은 [그림 2]와 같다.




[그림 2] 자동 XML 변환기 전체 흐름도

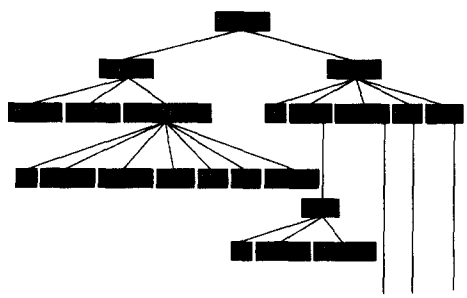
1단계에서는 일반적인 문서 편집 소프트웨어가 프린터 출력 파일로 생성한 prn형태의 중간 파일을 JetDocument([5])를 이용하여 파싱하여 XLX화일 구조로 변환한다. 그리고 2단계에서 변환된 XLX 화일 구조에서 Filter가 TOC DTD구조에 필요한 텍스트 정보를 추출하여 트리 구조의 자동 XML 변환기 엔진에 전달하면, 이곳에서 XML 파싱과 기존의 TOC XML 문서 로딩 및 XML 문서 생성을 담당하게 된다. 본 논문은 위의 2단계 전체 변환작업 중 트리 구조의 자동 XML 변환기 엔진에 대해 중점적으로 기술하였다.

### 3.2 트리 구조의 자동 XML 변환기 엔진

원 기술문서를 [그림 1]과 같은 트리 구조 정보로 정의하였고, 이 트리 구조 정보에서 기술문서의 각 부분은 최상위 레벨인 <TOC>부분으로부터 경로(Path)를 따라 구조화될 수 있다는 시각에 따라 엔진을 트리 모형으로 설계하였다. 즉, 문서의 자체적인 트리 구조에 맞게 구성하기 위해서 자동 XML변환기의 엔진 구조도 트리 모형을 갖도록 설계하였다.

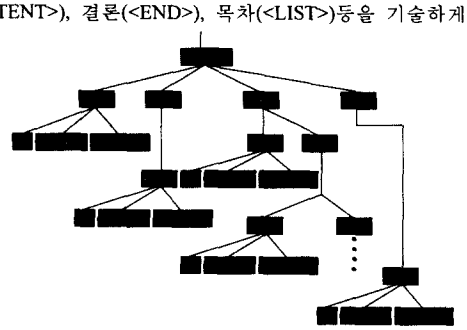
[그림 2]에서 Filter는 XLX 파일 구조에서 XML TOC DTD구조에 해당되는 텍스트정보를 추출하며, Filter의 내부처리는 일반적인 문서 편집 소프트웨어가 중간파일로 변환한 prn화일의 형식구조에서 필요한 텍스트 정보를 추출해내는 것과 같다.

XML 변환을 위한 트리 엔진은 입력, 파싱, 출력의 세 과정으로 나누어 볼 수 있다. 입력 과정에서는 XML TOC 문서 및 Filter에서 추출된 텍스트 정보를 입력하는 과정이다. 파싱 과정은 이렇게 입력된 텍스트 정보와 XML 문서를 XML TOC DTD 구문에 맞게 재배열 및 구문 체크를 하는 과정이다. 출력과정은 이러한 파싱 작업이 끝난 후, 해당 텍스트 정보에 따른 XML 문서를 생성하는 과정이다. 변환기 엔진의 전체 트리 구조는 다음 [그림 3-(a)], [그림 3-(b)], [그림 3-(c)]와 같다.



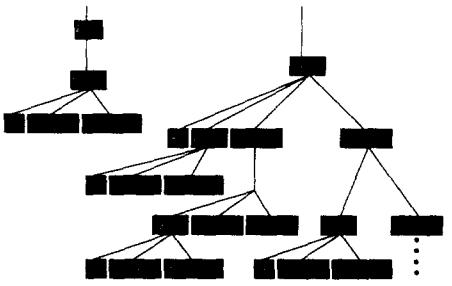
[그림 3-(a)] 자동 XML 변환기 트리 엔진 구성도

트리의 각 노드들은 [그림 1]에서 보는 기술문서 구조의 정의에 따른 것이며, 세분화하여 설계한 것이다. 기술문서는 전체를 <TOC>노드로 기술하며, <TOC>노드는 기술문서의 서지정보를 기술하는 <META>노드와 실제 본문인 <VOLUME> 노드로 구성된다. <META>노드는 기술문서의 총서번호, 서지정보(BIBLOIGRAPHY) 등을 기술하며, <VOLUME> 노드는 실제 본문을 페이지 단위로 표현한다. 그리고 <VOLUME>을 부모(parent)로 하여 각 페이지에 <VOLUME>노드의 자식(Child) 노드들 중 해당정보에 대한 노드로, 본문의 제목(<TI>), 표지(<COVER>), 본문(<CONTENT>), 결론(<END>), 목차(<LIST>)등을 기술하게 된다



[그림 3-(b)] 자동 XML 변환기 트리 엔진 구성도

[그림 3-(c)]의 <LIST>는 그림목차, 테이블 목차등 각 종 목차에 대한 정보를 기술하기 위한 노드이며, <PAGE>는 해당 페이지의 정보를 기술한다. 그리고 이 <PAGE>에는 해당 페이지의 제목(<TI>)과 예외처리(<EXCEPTION>)를 기술한다. 제목(<TI>)과 예외처리(<EXCEPTION>), 페이지(<PAGE>)는 각각 속성을 가지고 있어 좀 더 자세한 정보를 기술하게 하였다. 이 속성은 원 기술문서의 정보와 생성된 XML 문서가 시각적인 차이가 없게 하기 위한 상세한 정보를 기술하기 위한 것이다.



[그림 3-(c)] 자동 XML 변환기 트리 엔진 구성도

위에서 본 트리 엔진 구성도는 XML TOC DTD에 충실하여 만들어졌으며 현재 가장 범용화되어 있는 XML 뷰어 및 파서인 Internet Explorer 5.0의 정보를 기준으로 설계되었다[3][4]. 위에서 기술한 작업 과정(입력, 파싱, 출력)을 트리 엔진과 연관해 살펴보면 기술 원문서(XLX File Block을 통해 Filtering 된 텍스트정보)와 XML TOC 문서는 입력 후 간단한 입력 파싱작업을 거친 다음, 위 트리 엔진의 해당 노드에 입력 정보로 들어가게 되고 이 정보는 트리 엔진에서 파싱 과정을 내부적으로 처리하여 입력된 텍스트 정보와 연결해 XML TOC DTD 에 따른 해당노드의 구문 체크 및 출력을 위한 입력정보의 재배열과정을 처리하게 된다. 그리고, 파싱 작업이 완료되면 트리 엔진은 해당정보를 XSL과 연결할 것인지에 대한 선택과정을 포함해, 하나의 XML 문서 화일로 조합하여 생성하게 된다.

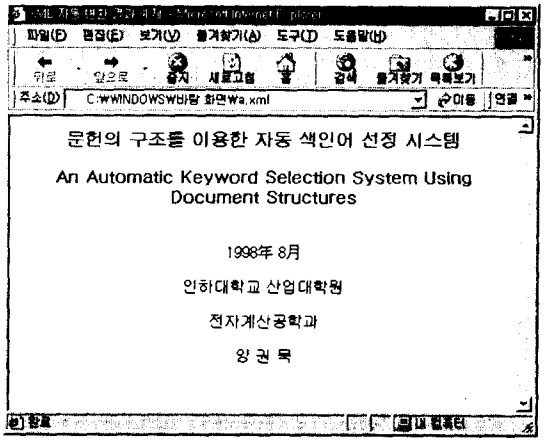
### 3.3 결과 및 XSL 설계.

위의 트리 엔진을 통해 생성되는 XML 문서의 예제는 [그림 4]와 같다. [그림 4]의 결과 예제는 입력 정보에서 문서의 서지정보만 출력된 형태의 예제이다

```
<?xml version="1.0" encoding="ksc5601"?>
<!DOCTYPE TOC SYSTEM "paper.dtd">
<?xml:stylesheet type="text/xsl" href="paper.xsl"?>
<TOC>
<META>
<AN>MO193023</AN>
<BIBLIOGRAPHY>
<TI FONTNAME="굴림체" FONTSIZE="15pt"> 문헌의 구조를 이
용한 자동 색인어 선정 시스템</TI>
<AUTHOR>양권묵</AUTHOR>
<COURSE>학위논문(석사)</COURSE>
<YEAR>1998年 8月</YEAR>  <ORG>인하대학교</ORG>
</BIBLIOGRAPHY>
</META>  <VOLUME> ..... </VOLUME>
</TOC>
```

[그림 4] 자동 XML 변환기 결과 예제

[그림 4]의 XML 문서에 대한 Internet Explorer 5.0결과 화면은 [그림 5]와 같다 [그림 5]는 [그림 4]에서 생성된 XML 문서에 기술 원 문서와의 시각적인 차이를 없애는데 중점을 두어 XSL에 대한 설계를 적용 시켰으며, 이 과정은 트리 엔진에서 선택적으로 적용되어진다.




[그림 5] XSL 적용후의 XML 문서

## 4. 결론 및 과제

현재 대부분의 문헌정보 검색시스템에서는 원 기술문서의 저장, 관리방법에 있어서 이미지 기반과, 표준화 문서(SGML, XML)방식, 또는 혼합방식을 사용하고 있으나 이 방법들은 인력 및 비용면에서 아직 많은 문제점을 갖고 있는 것으로 지적되고 있다.

본 논문은 멀티미디어 기술문서의 자동 XML 변환을 통해 전자도서관과 같은 기술문서 검색 시스템에서 사용자에게 구조화된 전자문서를 제공하고, 지금까지 생성되어온 출판문서 및 다양한 형식의 멀티미디어 전자문서를 표준화된 XML 문서로 자동변환하여 시간 및 인력낭비를 줄일 수 있는 방안으로 자동 XML 변환기를 개발하였다. 향후, 모든 멀티미디어 기술문서뿐만 아니라 일반형식의 원문서 또한 자동변환 할 수 있도록 구조화정보를 정의하여 표현하는 노력과 함께 자동변환하려는 노력을 경주해야 할 것이다.

[참고문헌]

[1] 국립중앙도서관, 원문 디지털화를 위한 목차입력 기술규칙 표준화(안) 국립중앙도서관 외 7, 1999.

[2] Frank Boumphrey, Olivia Direnzo, Jon Duckett, Joe Graf, Paul Houle, Dave Hollander, Trevor Jenkins, et al., *XML Applications*, Wrox, 1999.

[3] Alex Homer, *XML IE5 Programmer's Reference*, Wrox 1999.

[4] W3C, Extensible Markup Language 1.0, http://www.w3c.org, Recommendation, 1998.

[5] (주) 한양정보통신, JetDocument 파일 구조 변경 시나리오, (주) 한양정보통신, 1998.