PubMed Central (PMC) 작업 과정

허선(대한의학학술지편집인협의회 정보관리위원장)
10:35 - 11:00

머리말

KoreaMed를 처음 시작하던 1996년, 미국 NCBI에서 PubMed 사업을 하면서 앞으로 XML 파일만 보내주면 MEDLINE 학술지가 아니더라도 등재된다는 내용의 글을 보고 열심히 XML 파일을 만들자며 KoreaMed를 시작하였다. 처음에 KoreaMed에 등재되면 PubMed에도 등재된다는 언급을 학회에 하고 시작하였는데, 그 후 MEDLINE 학술지만 PubMed로 간다는 것으로 바뀌어서 여러 학회의 꿈은 무산되었다. 그런 일이 있은지 10년 뒤인 2006년, 이제 2008년이면 정보위 활동을 접어야 하는데 10년 전의 꿈을 구현시키는 방법이 무엇일까 생각하다가 2001년부터 시작한 PMC 사업을 한번 다시 들여다보게 되었다. 이것을 어떻게 해서든지 구현하여 영문으로 발행된 학술지는 PMC에 등재되어 검색가능 하도록 하여야 하겠다는 생각으로 공부하였고 그 가능성을 타진하여 본격적으로 시작하였다. 우선 편집인으로 일하고 있는 Journal of Educational Evaluation for Health Professions를 가지고 PMC XML을 만들어 웹에서 보이게 하는 작업을 하였다. 그리고 워드 파일에서 쉽게 PMC XML로 변환하는 filtering program을 김수녕 교수께 의뢰하여 제작하였다. Xalan-Java, Xalna-C의 사용법 때문에 어려움을 겪다가 프로그래머 고항훈씨의 도움으로 HTML 파일로 변환하였다. 지금부터 어떻게 구현하였는지, 그리고 그것을 구현하여 화면에 띄웠을 때에 어떤 느낌이었는지 설명하려 한다. 기생충학자가 이런 분야까지 공부하는 것이 과연 옳은 일인가 돌이켜 보기도 하지만, 아무도 국내에서 먼저 시도하지 않으니까 할 수밖에 없다고 여긴다. 이런 일은 의편협 실무자가 먼저 이해하여 가르쳐 주었으면 하는 바람이었다.

예제 파일

http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=1874508 은 PMC 학술지 중 알파벳에서 가장 앞선 잡지의 최근 full text이다.

다음 Fig. 1 같이 보인다.

소스를 보면

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd"><html><head><meta http-equiv="Content-Type" content="text/html; charset=UTF-8"><meta name="robots" content="INDEX,NOFOLLOW,NOARCHIVE">

이렇게 시작하여

Fig. 1. Example screen of full text from PMC.

<script type="text/javascript" language="JavaScript"><!--
try{initUnObscureEmail ("e_id2726096", '' + reverseAndReplaceString('if.uluo/ta/irattnam.utas', '/at/','@') + '')}catch(e){}

이렇게 끝난다. 이 화면을 봐서는 도저히 알 수 없다. 이 화면은 내부에서 HTML로 자료를 변환하여 연결시켜서 변환된 HTML 코드로 보아도 내용을 알 수 없다.

그러므로, PMC XML 예제 파일을 가지고 공부하였다.

Journal Publishing XML DTD and schema (http://dtd.nlm.nih.gov/publishing/)에서 내용을 찾을 수 있다. 이 화면에서 왼쪽 차림표에 Tag Library를 누르면 우리가 코딩 할 Tag에 대한 내용이 나온다(Fig. 2, http://dtd.nlm.nih.gov/publishing/tag-library/2.3/index.html)

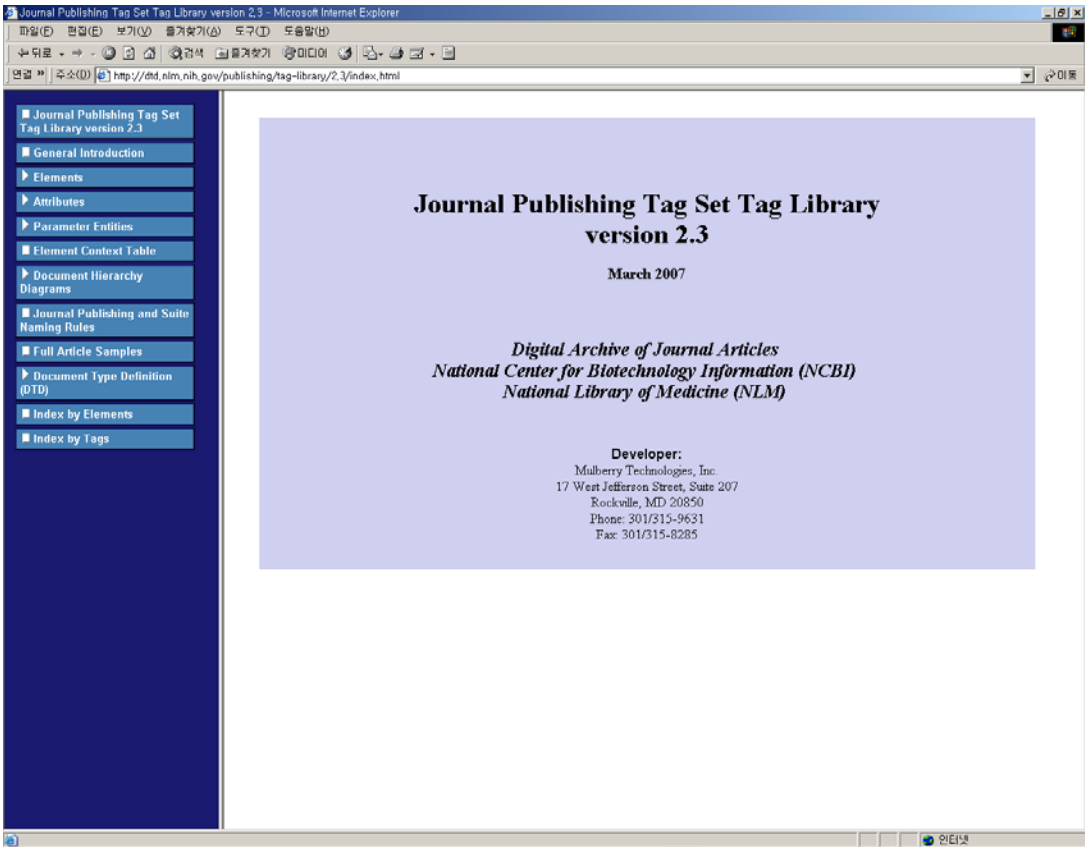

Fig. 2. XML Tag Library 화면.

이 화면의 차림표에서 Full Article Samples을 누르면 BMJ, PNAS 예제 파일이 나온다. 여기서 XML 파일의 구조를 알 수 있다. 그중에 BMJ XML 파일을 보면

```
<!DOCTYPE article PUBLIC "-//NLM//DTD Journal Publishing DTD v2.3 20070202//EN"
"nlm_lib/2.3/journalpublishing.dtd">
<article>
  <front>
    <journal-meta>
      <journal-id journal-id-type="pmc">bmj</journal-id>
      <journal-id journal-id-type="pubmed">BMJ</journal-id>
      <journal-id journal-id-type="publisher">BMJ</journal-id>
      <issn>0959-8138</issn>
      <publisher>
        <publisher-name>BMJ</publisher-name>
      </publisher>
    </journal-meta>
    <article-meta>
      <article-id pub-id-type="other">jBMJ.v324.i7342.pg880</article-id>
```

```
      <article-id pub-id-type="pmid">11950738</article-id>
      <article-categories>
        <subj-group>
          <subject>Primary care</subject>
```

이렇게 시작하여

```
 <fn>
        <p>Funding: Meetings of the working group in 1999-2000 were funded by the Scientific Foundation Board of the RCGP.</p>
      </fn>
      <fn>
        <p>Competing interests: None declared.</p>
      </fn>
    </fn-group>
  </back>
</article>
```

이렇게 끝나는 것을 알 수 있다. 이대로 따라서 JEEHP 코딩을 하고 XML 파일을 만들어 열어서 오류를 수정하여 웹 화면에서 볼 수 있다.

JEEHP 파일 만들기

예를 들면

```
<?XML version="1.0"  encoding="UTF-8"?>
<!DOCTYPE article PUBLIC "-//NLM//DTD Journal Publishing DTD v2.1 20060430//EN" "http://jeehp.org/drXML/journalpublishing.dtd">
<?XML-stylesheet type="text/xsl" href="http://jeehp.org/drXML/viewnlm-v2.xsl"?>
```

이렇게 시작하여 만들었다. 이 초기 heading이 매우 중요하다.

<?XML version="1.0" encoding="UTF-8"?>은 이 문서가 XML 문서이고 encoding은 TF-8 으로 정하여 준다는 것이고
<!DOCTYPE article PUBLIC "-//NLM//DTD Journal Publishing DTD v2.1 20060430//EN" "http://jeehp.org/drXML/journalpublishing.dtd"> 에서는 DTD를 v2.1로 하고 그 파일은 http://jeehp.org/drXML/journalpublishing.dtd 에 있다는 것이다. 지금의 DTD의 최신 버전이 v2.3이다.
<?XML-stylesheet type="text/xsl" href="http://jeehp.org/drXML/viewnlm-v2.xsl"?> 은 화면에 보여주는 것은 viewnlm-v2.xsl이라는 xsl에 따른다는 것이다.

여기서 DTD는 따로 내용을 이해하여야 하지만 우리는 그냥 최신판을 쓰는 것이므로 따로 공부할 필요는 없다. 단지 어떤 tag가 어떤 내용이라는 것을 정의한 것이다 수준에서 이해하면 된다. 또한 viewnlm-v2.xsl 역시 XML을 화면에 어떤 형식으로 뿌려 주는 지를 정하는 것이므로 이것을 바꾸면 화면에 나오는 것이 바뀌지만 우리는 역시 손을 대거나 공부할 필요 없이 최신판을 쓰면 된다. 만약 Response page를 만들다가 화면 배열이 맘에 들지 않는 것이 있다면 이것을 수정하여 쓸 수 있다. XML 전문가가 도와주어야 하므로 대부분의 경우는 프로그래머에게 부탁하여 수정한다. 이렇게 하여 처음 만들어 올린 화면은 Fig.3과 같다.

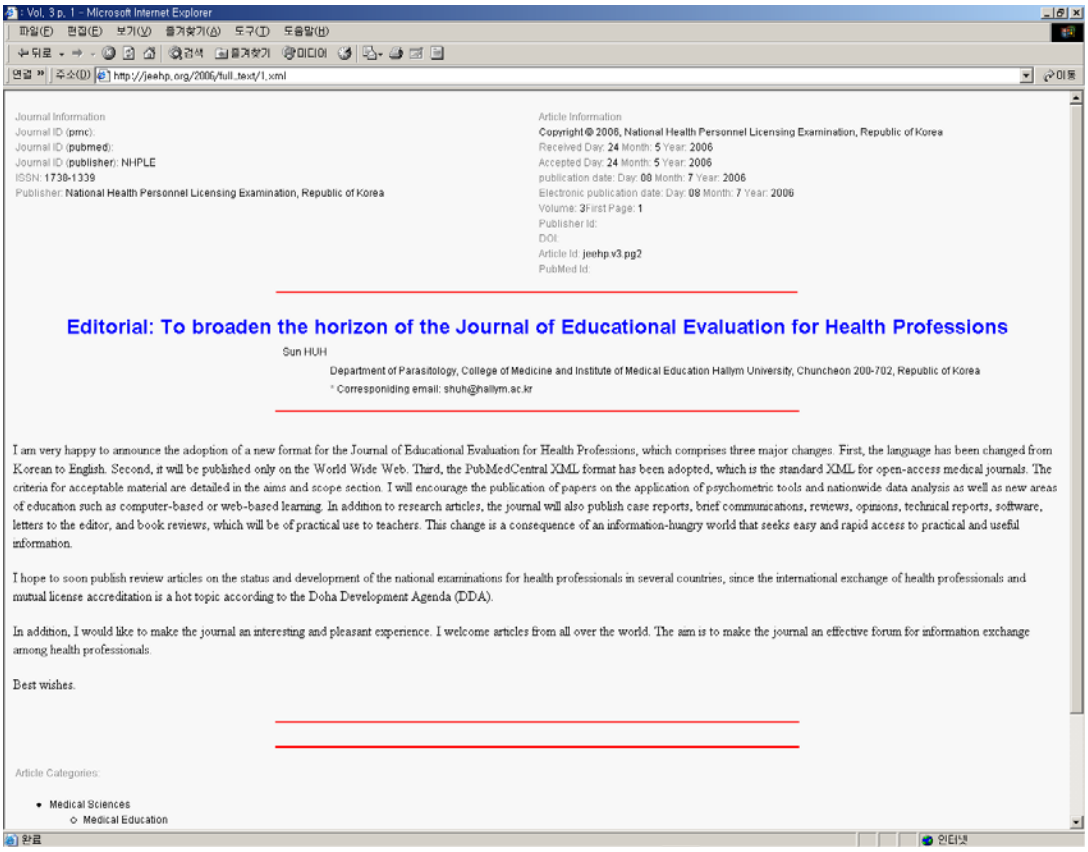

Fig. 3. 처음 xml 화면으로 띄운 Journal of Educational Evaluation for Health Professions.

이렇게 XML 파일로 띄우고 말면 간단하지만 문제는 브라우저 중에서는 XML을 지원하지 않는 opera 같은 것도 있고 또한 매번 journalpublishing.dtd과 viewnlml.xsl 파일을 읽어서 가져와야 하므로 웹브라우저에서 XML 파일을 읽어 오는데 시간이 꽤 걸린다. 그러므로 XML 파일을 그대로 올리는 것이 아니라 HTML 파일로 변환하여 올린다. 그러면 그 파일만 읽으므로 빠르게 사용자가 화면에서 볼 수 있다.

Xalan-C 및 Xalan-Java

이 두가지 프로그램 모두 XML을 xsl 파일에 맞추어 HTML 파일로 변환시켜 주는 것인데

자바용은 자바프로그램 설치 때문인지 잘 되지 않아서 C로 만든 Xalan-C 가 된다는 것을 고항훈씨에게서 배워서 시행하여 성공하였다. 그래서 최종으로는 HTML 파일을 올린다. 사용법은 오후에 배울 것이다.

프로그램은 무료이고

XALAN 설치는 http://blog.naver.com/huhsoleil2/70006714909 에 설명이 있다.

1. 두개의 프로그램을 다운로드 한다.

Xalan-C downloads (윈도우 사용자용
http://ftp.kaist.ac.kr/pub/Apache/XML/xalan-c/binaries/Xalan-C_1_10_0-win32-msvc_60.zip)
Xerces-C downloads (윈도우 사용자용
http://mirror.apache.or.kr/XML/xerces-c/binaries/xerces-c_2_7_0-windows_2000-msvc_60.zip)

2. 압축을 푼다.

3. 압축을 풀은 XERCES 폴더의 BIN 폴더에서 xerces-c_2_7.dll를 XALAN의 BIN 폴더에 복사한다.

4. Xalan-C_1-10_0-win32-msvc_60₩BIN 폴더를 환경변수에 저장한다.
방법> 내컴퓨터 → 마우스 오른쪽 눌러 속성 → 고급 → 환경변수 → 시스템변수의 PATH의 끝에 경로를 추가한다.

이때 ; 를 먼저 붙인다.

;C:₩Program Files₩Xalan-C_1_10_0-win32-msvc_60₩bin

5. xalan 밑의 bin 가지에 journalpublishing.dtd, 및 이 DTD와 관련된 모든 파일, nlmview-v2.xsl이 같이 있어야 한다.

6. CMD 창에서 다음과 같이 변환한다.

XALANTRANSFORM INPUT파일명.XML STYLE파일명.XSL OUTPUT파일명.HTML

예) xalantransform 1.XML viewnlm-v2.xsl 1.html

7) 만약 4의 과정처럼 path에 연결시키지 않은 경우에는

c:₩xalan-c₩ 아래에다가 xalan 의 파일을 심고, bin 가지 아래 xerces-c_2_7.dll 을 심고, bin 가지 아래 나머지 journalpublishing.dtd 파일과 관련 dtd 파일을 전부 심고, nlmview-v2.xsl 을 심고

c:₩xalan-c₩bin> xalantransform 1.XML viewnlm-v2.xsl 1.html

이렇게 치면 변환된다.

코딩을 모두 수작업으로 할것인가?

그럴 수는 없다. 대개 하나의 파일을 다루는데 숙련되면 15분 걸린다고는 하지만 필터링 프로그램이 5,000만원 정도 하는데 5,000만원이면 그냥 하나 만드는 것이 낫겠다고 여겨 김수녕 교수께 부탁을 드려 DrXML이라는 필터링 프로그램을 하나 제작하였다. 작업할 때 가장 손이 많이 가는 것을 나열하면 Table, Reference, 참고 문헌 hyperlink, 그리고 저자 사항 순이다. 나머지 text파일 부분은 매우 단순하고, 그림도 단순하다. 이런 부분만이라도 저렴한 비용으로 만들어 주신 것 프로그램으로 자동처리 할 수 있다면 입력시간을 매우 짧게 단축시킬 수 있다. 아직 개선할 점이 있지만, 경비의 문제도 있으므로 그것은 작업을 하면서 천천히 개선하여 나가야 할 것이다. 또한 학술지마다 다른 참고문헌 작성법에 따라 수정을 하여 주어야 한다. DrXML도 사용법을 배울 것이다(Fig. 4).

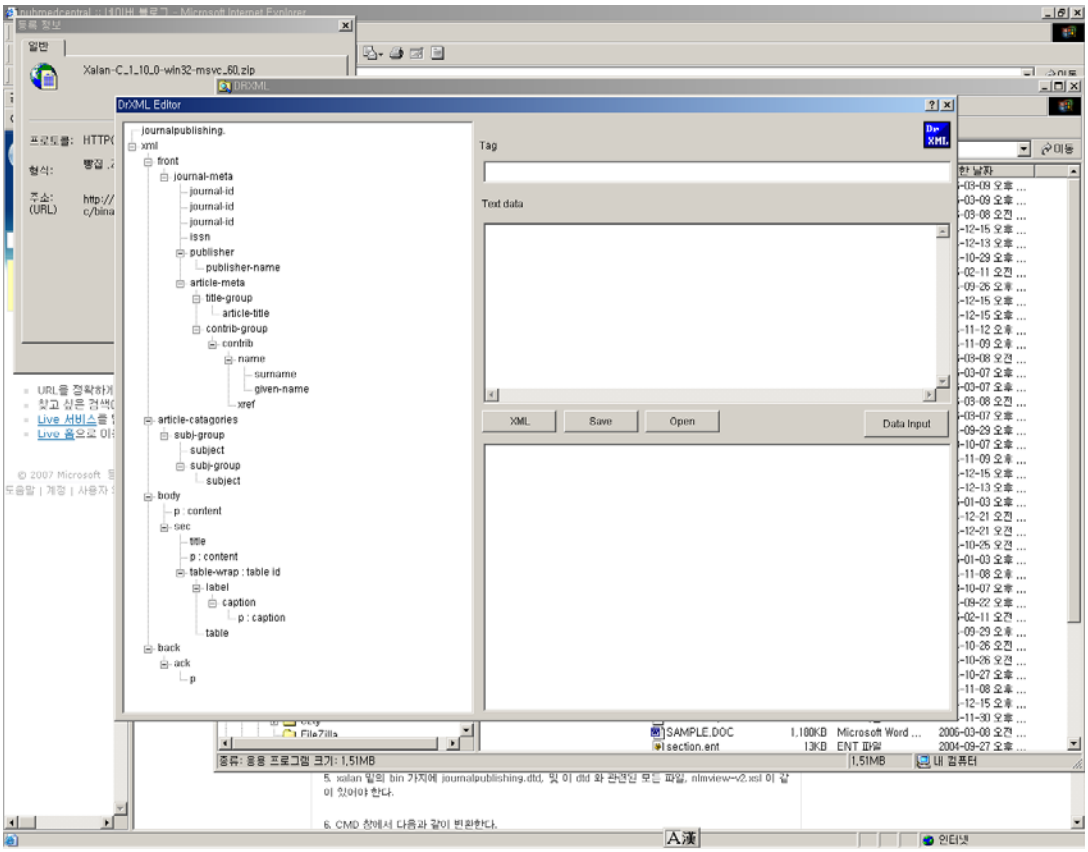

Fig. 4. DrXML 초기 화면.

PMC XML 파일의 점검

파일이 제대로 만들어 졌는지 점검할 필요가 있다. 다음과 같은 곳에서 한다.

PMC XML Validator
http://www.pubmedcentral.nih.gov/utils/validate/XMLcheck.cgi

PMC Style Checker
http://www.pubmedcentral.nih.gov/utils/style_checker/stylechecker.cgi

이 두가지를 점검하는 데 PMC Style Checker은 완벽하게 하기 어렵지만 PMC XML Validator에서는 오류가 나면 안된다.

문서 편집기

crimson editor를 사용하는 것이 좋다. notepad나 wordpad보다 많은 기능을 제공하고 전 세계 최고 수준이면서도 무료이다. http://crimsoneditor.com/에서 내려 받을 수 있다.

맺는 말

이렇게 작업하여 처음 XML 파일을 올리고 다시 HTML로 변환하여 웹잡지를 꾸몄다. 남들도 하는 것을 구현하는 것이 대단한 것은 아니나 적어도 국내에서는 최초로 웹전용 학술지가 탄생하였다는 점에서 매우 기뻤다. 영문이라서 국내에서 별로 투고도 없고, 국제적으로 알려지지도 않은 학술지이나 PMC에 등재되면 조금 사정이 달라질 것이다. 등재시키고 나면 다시 PubMed에 등재 준비를 할 작정이다. 아무쪼록 나머지 자세한 내용은 실무진을 통하여 공부할 수 있기 바란다. 전체적인 PMC XML 파일 작성은 이런 순으로 하였다.