Form을 지원하는 XML 문서 편집기 구현

고 탁 현[U] 황 인 준
아주대학교 정보통신 전문대학원
{sanha, ehwang}@madang.ajou.ac.kr

An implementation of Form-supporting XML Document Editor

Takhyun Ko[U] Eenjun Hwang
The Graduate School of Information and Communication, Ajou University

요약

XML이 인터넷 상에서 정보 교환을 위한 새로운 표준으로 채택됨에 따라 XML 문서의 효율적인 작성과 관리를 위한 다양한 형태의 편집기가 계속 개발되고 있다. 본 논문은 기존의 XML 문서 편집기가 가진 제약점을 살펴보고 이를 해결하기 위한 새로운 형태의 XML 문서 편집기를 제안한다. 기존의 XML 문서 편집은 대부분 트리 구조를 기반으로 하여 이루어지고 있으며, 이것은 사용자로 하여금 XML에 대한 어느 정도의 지식을 필요로 한다. 그러나 XML 문서의 작성과 활용이 보편화되기 위해서는 이러한 문서 편집 방식에서 벗어나 누구나 쉽게 작성할 수 있는 환경이 제공되어야 한다. 본 논문에서는 이를 위해 트리 방식의 문서 편집뿐 만 아니라 문서의 양식을 반영하는 form을 기반으로 하는 편집 환경도 지원하는 새로운 XML 문서 편집기를 개발한다. Form의 생성은 template XML 문서와 XSLT을 통해 이루어지며 생성된 form은 내장된 HTML 브라우저를 통해 사용자에게 보여진다. Form에 입력된 사용자 데이터는 해당 XML 문서로 변환되어 데이터베이스에 저장된다.

1. 서론

XML 문서 편집기의 주요한 기능은 크게 XML 문서의 작성과 검증(validation)에 있다고 할 수 있다. 즉, XML이 정의하는 syntax에 적합한 well- formed 문서나 주어진 DTD (Document Type Definition)에 부합되는 유효한(valid) 문서를 어떠한 오류 없이 사용자가 보다 쉽게 제작할 수 있도록 향상된 인터페이스를 제공해야 한다. 이를 실현하기 위해 현재까지 많은 XML 문서 편집기가 개발되어 왔지만 대부분 아직까지 많은 제약점을 가지고 있다. 구체적으로 기존의 XML 문서 편집기는 효율적인 브라우징 기능을 제공하지 못하고 있으며, XML 문서의 구조적 정보를 표현하는 단순한 트리 구조를 기반으로 하는 인터페이스 상에서 XML 문서를 제작하는 환경을 제공한다. 이것은 XML 문서를 제작하기 위해서는 사용자가 어느 정도 XML에 대한 사전 지식을 가지고 있어야 한다는 것을 의미한다. 그러나, XML 문서의 사용이 보편화되고 있는 현실에 비추어 보면, 사용자가 보다 쉽게 사용할 수 있고, XML에 대한 지식이 없이도 XML 문서를 쉽게 작성할 수 있는 환경을 제공하는 문서 편집 도구의 개발이 절실하다.

본 논문에서는 문서의 양식을 반영하는 form을 기반으로 하는 XML 문서 제작 환경을 제공함으로써 XML에 대한 사전 지식이 없이도 일반 사용자가 쉽게 XML 문서를 작성할 수 있게 해주며 form을 통해 작성된 문서는 자동적으로 XML 문서로 변환되는 새로운 XML 문서 편집기를 개발하고자 한다.

2. 관련연구

2.1 HTML Form 생성

XML 문서는 문서의 구조와 의미에 관한 정보만을 기술하며, 그 요소들의 표현에 대한 정보는 XSL 문서가 가지고 있다. 또한 XSL 문서의 하위 구성요소인 XSLT를 사용하면 XML 문서를 다른 형태의 XML 문서뿐 만 아니라 HTML과 같은 다른 형식의 문서로도 변환이 가능하다. 본 시스템에서는 form 기반의 XML 문서 생성을 지원하기 위하여 이러한 XSLT 기술을 이용하였다. 입력된 XML 문서와 XSL 문서를 기반으로 XSLT 처리 과정을 통해 사용자 입력을 위한 HTML form을 생성하고, form을 통해 입력된 데이터는 새로운 XML 문서의 제작에 사용된다.

3. 편집기의 전체적인 구성

본 편집기는 크게 트리 구조를 기반으로 하는 XML 문서 작성 모듈과 form을 기반으로 하는 문서 작성 모듈로 나누어진다. 전자는 기존의 문서 편집기가 지원하는 방식을 취하여 구현되었고, 후자는 문서의 양식에 따르는 form 상에서 필요한 XML 문서를 작성하는 환경을 제공한다.

3.1 트리 기반 인터페이스를 위한 구성요소

트리 기반의 XML 문서 작성 모듈은 DTD의 사용 여부에 따라 유효한 XML 문서와 well-formed XML 문서 작성을 지

원하며 트리의 생성 및 저장 모듈, 트리의 확대, 축소, 선택과 같은 트리 조작 모듈, 트리 엘리먼트의 편집모듈, DTD를 통한 트리 엘리먼트 filtering 모듈 등으로 구성되어 있다.

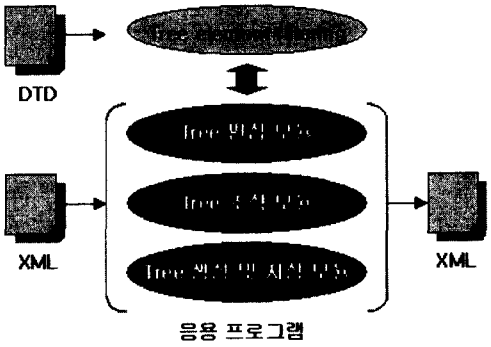

그림 1 트리 기반의 XML 문서 작성 구성요소

3.2 Form 기반 인터페이스를 위한 구성요소

Form을 기반으로 하는 인터페이스 상에서 XML 문서를 작성하기 위해서는 template XML 문서와 XSL 문서가 필요하다. 즉 사용자의 입력 form 양식을 생성하기 위해서 XML 문서의 표현을 담당하는 XSL 문서를 중간 매개체로 사용하였으며, template XML 문서와 함께 입력 form 생성에 사용되어진다.

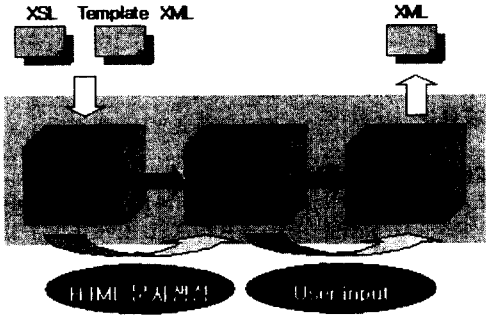

그림 2 Form 기반의 XML 문서 작성 구성요소

Form 기반의 XML 문서 작성 모듈은 그림 2와 같이 XSLT processor, HTML browser, XML generator 등으로 구성되어있다. XSLT 처리기는 template XML 문서와 XSL 문서를 입력으로 받아서 사용자의 입력 form에 해당하는 HTML 문서를 생성한다. 생성된 HTML 문서가 브라우저로 전달되면, 브라우저는 이 문서를 브라우징하여 해당 입력 form을 사용자에게 보여주게 된다. 끝으로 XML 생성기는 브라우저 상에서 입력 form을 통해 입력받은 데이터를 바탕으로 새로운 XML 문서를 생성하게 된다.

3.3 XML 문서 처리 전체 흐름도 및 구현 화면

본 논문에서 제안한 트리/form 기반 XML 문서 편집 시스템은 크게 DTD 파서, XML 파서, XSLT 처리기, HTML 브라우저, XML 문서 생성기, 트리 처리 API 모듈로 구성되어 있다. 그림 3은 시스템을 구성하는 각 요소와 실제 XML 문서 처리를 위한 요소들 간의 상호 작용 및 전체 흐름을 보여주고 있다.

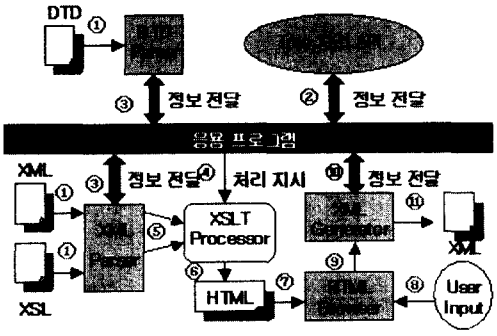

그림 3 XML 문서 처리 전체 흐름도

그림 3에서 화살표를 따라 표시된 숫자는 XML 문서의 작성 시 흐름 순서를 나타내고 있으며 트리 처리를 위한 API에는 트리의 생성, 저장, 추가, 삭제, 확대, 축소, 선택 등을 위한 기능이 제공된다.

그림 4는 제안된 XML 문서 편집기의 화면 구성 예를 보여주고 있다. 화면의 왼쪽 창은 DTD나 XML, XSL 문서를 파싱하여 그 결과를 JTree 형태로 보여주며, 오른쪽 창은 HTML 브라우저를 통하여 사용자 입력을 지원하는 form 문서를 브라우징하고 있다.

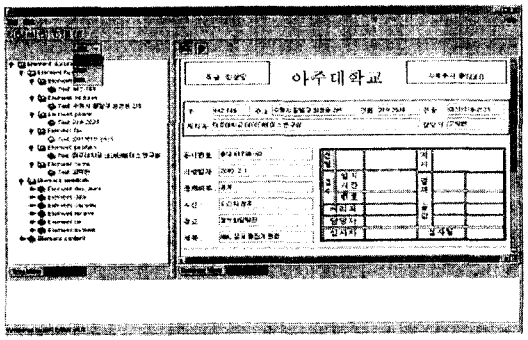

그림 4 XML 문서 편집기 화면 구성

4. Form 기반의 XML 문서 생성

4.1 HTML Forms 생성

HTML 형태의 form을 생성하기 위해서는 먼저 template XML 문서와 XSL 문서를 제작하여야 한다. XSL 문서를 통해 생성된 form은 해당 template XML 문서의 디스플레이 정보를 담고 있으며, 사용자 입력을 받기 위한 입력 양식을 포함하고 있어야 한다. 사용자 입력 양식으로는 버튼 입력 양식, 체크박스 입력 양식, 패스워드 입력 양식, 라디오 입력 양식, 텍스트 입력 양식, 선택 입력 양식 등이 있으며, 이외에도 입력된 정보의 전송과 재 입력을 지원하기 위한 submit 입력 양식과 reset 입력 양식이 포함된다. 다음은 다양한 입력 양식을 지원하기 위한 XSL 문서의 작성 방법의 예를 보여주고 있다.

▸ **텍스트 입력 양식**

```
<xsl:template match="XML_TAG_NAME">
  <input type="text" name="TXT1" size="30%"
                          value="{text()}"/>
</xsl:template>
```

▸ **라디오 입력 양식**

```
<xsl:template match="XML_TAG_NAME">
  <input type="radio" name="R1" value="M"/>man
  <input type="radio" name="R1" value="F"/>woman
</xsl:template>
```

▸ **선택 입력 양식**

```
<xsl:template match="XML_TAG_NAME">
  <select name="SEL">
    <option value="A"> DATABASE </option>
    <option value="B">SYSTEM</option>
  </select>
</xsl:template>
```

위에 나타나는 HTML 입력 태그를 포함한 모든 입력 태그들은 HTML <FORM> 태그 속에 포함되어야 하며, <FORM> 태그 속의 마지막 부분에는 사용자 입력 내용의 전송과 재 입력을 위한 버튼 입력 양식이 정의되어야 한다.

4.2. XML 문서 생성

Form을 기반으로 하는 XML 문서의 작성을 위해 다음과 같은 두 가지 매핑 방안을 고려해 볼 수 있다. 우선 기본적인 매핑 방안으로 template XML 문서의 엘리먼트 이름에 기초한 매핑이다. 이러한 매핑을 지원하기 위해서 브라우저 상에서 발생되는 form event는 아래 (1)에 나타난 형태로 XML 생성기에 전달된다. (1)과 같이 발생된 form event에 근거하여 각 엘리먼트 이름에 대한 value 매핑으로 새로운 XML 문서가 생성된다. 하지만 이러한 방법을 이용하는 XML 문서 생성기는 중복된 태그 명을 가지는 XML 문서의 생성 시 많은 문제점을 내포하게 된다. 이에 대한 해결 방안으로 본 논문에서는 template XML 문서의 각 엘리먼트에 이름 대신에 고유한 키 값(ID)을 부여하고 그것에 기초한 매핑 방식을 채택하였다. 고유한 키 값에 기초한 매핑 방식을 지원하기 위해 발생되는 form event는 아래 (2)와 같다. Template XML의 각 엘리먼트에 고유한 키 값을 부여함으로써 중복된 태그를 가진 XML 문서의 생성도 가능하게 하였다.

▸ **form event (엘리먼트 이름을 이용한 매핑 시)**
element_name1=value1&element_name2=value2 - (1)

▸ **form event (고유한 키 값을 이용한 매핑시)**
element_id1=value1&element_id2=value2 - (2)

그림 5는 XML 엘리먼트에 키 값을 부여한 예를 보여주는데, 키 값은 각 엘리먼트의 고유 경로를 근거로 하여 부여된다. 그림과 같이 template XML의 각 엘리먼트에 고유한 키 값을 부여함으로써 중복된 태그를 가진 XML 문서의 생성도 가능하게 하였다. 그림에서와 같이 루트(root)노드에 대한 키 값을 1로 정의하였고, 자식(child) 노드들의 키 값을 부여하기 위해 재귀 호출을 통한 깊이 우선 탐색(depth firsrt search) 방식으로 모든 노드들을 순회한다. 이를 통해 탐색된 모든 자식 노드들의 키 값은 루트 노드로부터의 고유한 경로에 기초하여 부여된다.

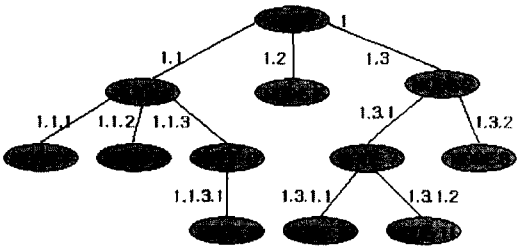

그림 5 고유한 경로에 근거한 XML 엘리먼트의 키 값

위와 같이 고유한 경로를 기초로 하여 template XML 문서의 각 엘리먼트들에 대한 키 값을 부여함으로써, form 양식을 통해 얻어진 엘리먼트와 엘리먼트가 가지는 데이터를 매핑시킬 때 엘리먼트의 탐색 비용을 최소화시킬 수 있다. 즉, 엘리먼트의 키 값 속에는 루트 노드에서부터 해당 엘리먼트로의 경로 정보가 포함되어 있고, 그런 키 값을 통한 엘리먼트의 탐색은 루트 노드에서부터의 최소 경로가 된다.

5. 결론 및 향후 연구과제

본 논문에서는 기존의 트리 구조 인터페이스를 통한 XML 문서 작성 방법에서 벗어나 문서의 양식을 반영하는 form 인터페이스를 통해서 XML 문서의 작성까지도 가능한 XML 문서 편집기를 설계하였다. Form을 지원하는 문서 제작 환경을 제공함으로써 XML에 능숙하지 못한 사용자도 쉽고 편리하게 XML 문서를 작성할 수 있게 되었다. 하지만 입력 form을 생성하기 위해서는 template XML 문서와 XSL 문서를 필요로 하기 때문에, 초기 template XML과 XSL 문서의 제작이라는 부담은 남는다. 하지만 기업이나 관공서 등과 같이 이미 정형화된 양식의 문서를 사용하는 곳에서는 위와 같이 설계된 XML 문서 편집기를 사용하여 미리 정의된 form을 통해 다량의 문서를 작성할 때 상당한 편리함을 제공할 수 있을 것으로 기대된다.

향후 연구 과제로는 본 편집기에서 사용되는 HTML form의 효율적인 생성을 위하여 그래픽 적인 형태에서의 form 제작 환경과 그러한 form에서 XSL 문서의 자동적인 생성을 지원하는 문서 편집기가 연구되어져야 할 것이다.

참고문헌

[1] Tim Bray, Jean Paoli, C.M.Sperberg-McQueen, Extensible Markup Language (XML) 1.0, W3C Recommendation, 10 Feb. 1998, http://www.w3.org /TR/REC-xml

[2] James Clark, XSL Transformations(XSLT) Version 1.0, W3C Recommendation, 16 Nov. 1999, http://www.w3.org /TR/xslt

[3] Vidur Apparaoet al., Document Object Model (DOM) Level 1 Spec, W3C Recommendation,1 Oct. 1998, http://www.w3.org/TR/REC-DOM-Level-1

[4] 고윤희, 김현철, 이원규, 고려대학교 컴퓨터교육과, "데이터베이스와의 연동 기반 XML 에디터의 설계 및 구현," 한국정보과학회, 2001 봄 학술발표 논문집(B) 제28권 제1호, 2001

[5] 김동욱, 최한석, 목포대학교 컴퓨터과학과, "XSL을 지원하는 XML문서 편집기 설계 및 구현," 목포대학교 정보산업연구지 제 7한, pp. 35-44, 1999

[6] 김길준, 정용득, 숭실대학교 컴퓨터학부, "XML 저작도구의 설계," 정보산업기술연구(제 3집), 1998