

# 그룹웨어 시스템을 위한 XSL 스타일 편집기 개발

이은정<sup>0</sup>                      우  균  
경기대학교                  동아대학교  
[ejlee@kyonggi.ac.kr](mailto:ejlee@kyonggi.ac.kr)    [woogyun@mail.donga.ac.kr](mailto:woogyun@mail.donga.ac.kr)

## Development of an XSL Style Editor for Groupware Systems

Eunjung Lee                      Gyun Woo  
Kyonggi University              Dong-A University

### 요  약

XML 문서를 위한 XSL 스타일 생성 편집기 개발에 대해 소개한다. XSL은 XML 문서가 어떻게 보여질 것인가를 기술하는 언어인데, 이것으로 웹 상의 문서 스타일을 기술하기 위해서는 웹 에디터와 같은 편집기가 반드시 필요하다. 본 논문에서는 스타일 편집을 위하여 DTD를 사용하지 않고 XML 문서를 이용하는 편집기의 개발에 관하여 살펴보고, 문서의 형태를 고루 지원하지 못하는 문제점을 해결하기 위하여 DTD 정보를 보완할 수 있는 기능을 제안하였다. 본 시스템은 WYSIWYG 편집 기능을 제공하며, XSL의 템플릿 기능을 활용하여 스타일의 재귀적인 적용이라는 새로운 개념의 편집 방식을 제공한다

### 1. 서  론

최근 XML[1-3]이 데이터 교환의 표준으로 웹 상에서 광범위하게 사용되면서, HTML등 기존 문서 작성 언어를 대체할 기술로 주목 받고 있다. XML의 가장 큰 특징은 데이터 자체에 의미를 나타내는 태그가 포함될 수 있다는 점과, 스타일 부분이 분리되었다는 점이다. 즉 XML 문서에는 데이터와 태그 요소들만 포함되며, 스타일 정보는 CSS 나 XSL과 같은 별도의 표준에 의해서 제공된다.

XSL은 XML의 데이터 변환과 스타일을 위해 월드와이드웹 컨소시엄[4]에서 제정한 표준으로 HTML과 유사한 형식의 스타일 기능을 제공한다. XSL의 특징은 XML 문서에 대한 템플릿 규칙을 이용하여 문서의 여러 요소에 대해 하나의 스타일 규칙을 적용할 수 있다. 특히 XSL의 강력한 질의 기능을 바탕으로 스타일 편집이 체계적이고 일관된 형태로 XML 데이터에 적용될 수 있다.

XML 기술을 적용하여 시스템을 구축할 때 가장 필요한 기술은, XML 데이터 생성과 스타일 편집을 지원하는 도구이다. 개발자가 아닌 일반 사용자를 위한 스타일 편집기의 개발은 현재 많은 업체에서 시도하고 있으나[5,6] 아직 획기적인 해결책을 찾지는 못하고 있다. 본 연구에서는, XSL 기반 스타일 편집기에서 편리한 사용자 인터페이스를 제공하기 위한 방법으로 다음 두 가지 모델을 제안하고 구현하였다.

첫 번째 방법은 XML 문서 예를 이용한 편집 방식을 제공하는 것이다. XML 문서는 문서 구조를 나타내는 DTD를 가질 때 유효한 문서가 되는데 DTD는 문서의 종류마다 하나씩 정의된다. DTD는 BNF 형태의 문법 기술 언어로 일반 사용자가 사용하기 어렵다. 반면에 XML 데이터 인스턴스를 사용하면 실제 데이터를 가지고 편집하는 기존의 편집기와 유사한 인터페이스를 제공

<sup>0</sup> 본 연구는 대경컴퓨터사의 지원으로 수행되었음.

할 수 있다. 그러나 XML 문서 예는 DTD가 가지는 다양한 구조를 표현할 수 없으므로 본 시스템에서는 DTD 정보를 추가로 제공하는 방안을 채택하고 있다.

두 번째 방법은 사용자가 XSL의 데이터 선택 및 매칭 기능을 잘 이용할 수 있도록 편집 기능을 제공하는 것이다. XSL에서는 스타일을 적용할 XML 트리의 데이터 부분을 나타내기 위해 다양한 매칭 방식을 제공하고 있는데 이것을 잘 활용할 수 있다면 여러 개의 XML 문서에 대한 일관되고 효과적인 스타일을 기술할 수 있다. 그러나 XSL의 데이터 선택 기능이 워낙 광범위하므로 본 시스템은 그 중에서 일부 기능을 선택하여 효과적인 인터페이스를 구축하는 방법을 실험하였다.

본 논문의 구성은 다음과 같다. 2절에서는 그룹웨어 시스템에서 XML 및 XSL 방식을 이용한 결재 문서의 작성 및 처리를 살펴보고 3절에서는 본 시스템의 설계와 구성을 소개한다. 4절에서는 개발 시스템의 기능과 인터페이스를 살펴보고 5절에서는 결론을 맺는다.

### 2. XML 및 XSL에 의한 결재문서의 작성 및 처리

XML 기술은 결재 문서의 첨부라는 이중 처리 문제를 효과적으로 해결한다. 현재 대다수의 사내 그룹웨어 시스템에서 결재 문서의 처리는 결재를 위한 인터넷의 데이터 처리 부분에 주요 내용을 입력하고 실제 작성된 결재 문서를 첨부하는 형태인데(그림 1-(가)참고), 이 형태에서는 문서와 결재 부분이 분리되어 있으므로 데이터의 이중 입력 문제 뿐 아니라 결재를 위해 결재자가 첨부문서와 그룹웨어 상의 데이터를 모두 고려해야 한다는 번거로움이 있다.

XML 형태로 작성된 문서는 그 자체로서 문서의 역할을 할 뿐 아니라 자동 처리가 가능하다(그룹웨어 상에 입력한 데이터 역할도 할 수 있다). 그림 1-(나)는 XML 기반의 결재 처리 시스템에서 사용자가 결재 문서를 작성하는 과정과 그것을 결재하는 방식을 보여준다. 사용자가 결재 문서를 처리하는 과정은 폼 기반의 입력 형태가 되고 결재자는 그 내용을 XSL에 의해서 기존의 첨부

부 문서와 같은 형태로 볼 수 있다.

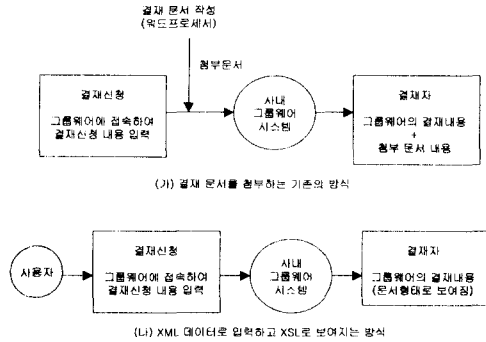


그림 1. 결재문서 처리 방법

XSL에서 정의한 스타일에 의해 결재 문서를 보여주는 상황을 도시하면 그림 2와 같다.

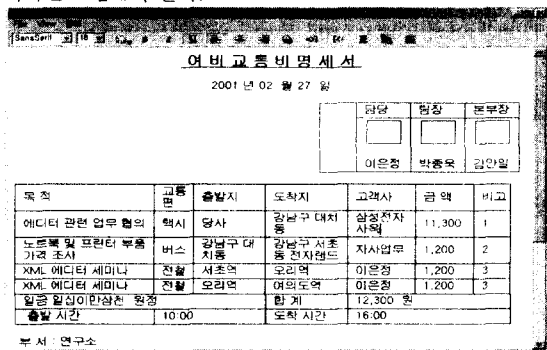


그림 2. 편집기의 사용형태

결재자가 그룹웨어 상에 입력된 데이터를 문서의 형태로 보기 위해서는 그 스타일을 나타내주는 XSL이 있어야 한다. 본 논문의 편집기 시스템은 해당 문서의 종류에 대한 XSL을 생성하는 도구이다. 이렇게 XML과 XSL을 도입하여 문서 처리 시스템을 구축했을 때 기대효과는 다음과 같다.

- (1) 문서의 내용을 자동으로 처리 가능하다.
- (2) 스타일을 일괄되게 유지 관리할 수 있다.
- (3) 필요에 따라 다른 스타일을 적용 가능하다.
- (4) 데이터의 접근을 XSL에 의해 제어할 수 있다.

XSL을 이용한 스타일 편집기는 사용자의 측면에서 다음과 같은 특징이 있다.

- (1) 데이터와 스타일을 별도로 유지해야 한다.
- (2) 텍스트 부분이 데이터와 스타일에 분리된다.
- (3) XML 문서의 임의의 부분에 스타일을 적용 가능하다.
- (4) 스타일 적용을 여러 요소에 반복 적용할 수 있다.

위의 특징들은 새로운 편집기 인터페이스의 목표를 제시하는데, 즉 사용자가 데이터와 스타일의 분리를 가능한 한 의식하지 않고 사용하도록 설계되어야 하며, 스타일 적용 부분의 선택이나 반복 적용 기능을 최대한 활용할 수 있게 편집 기능을 제공해야 한다. 사용의 편리성과 XSL의 선택 기능의 효과적인 사용은 상충되는 목표인데, 두 가지를 모두 달성하여 최선의 인터페이스를 설계하는 것이 스타일 편집기 개발의 가장 중요한 목표이다. 본 연구에서는 인터페이스 개발을 위한 여러 방법을 실험적으로 구현하여 그 결과를 확인하고자 한다. 특히 스타일의 반복 적용은 현재 HTML의 CSS 기능이나 MS 워드 편집기의 스타일 적용 기능보

다 더 강력한 개념의 편집 기능을 가능하게 할 것으로 기대된다.

### 3. 스타일 편집기의 기능설계

#### 3.1. 편집기의 구조 및 기능

본 시스템은 사용자가 XML 문서를 위한 스타일 파일을 작성할 수 있게 지원하는 편집 도구로서 그 구조는 그림 3과 같다. 편집 결과는 XSL로 저장되며, 이미 작성된 XSL 파일을 읽어 수정하는 것도 물론 가능하다. 편집기의 입력으로 XML 문서 또는 DTD가 주어질 수 있는데, XML 문서를 입력으로 받는 경우에는 그 문서의 내용을 기반으로 스타일을 편집하고 그것을 같은 종류의 다른 XML 문서에도 모두 적용할 수 있다. 한편 DTD를 입력으로 받으면 그 DTD에서 표현하고 있는 문서 구조에 대해 스타일을 작성할 수 있다.

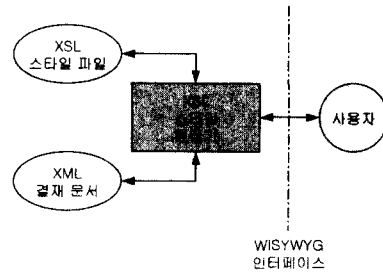


그림 3. XSL 스타일 편집기 구조

#### 3.2. 스타일 편집 기능

##### 3.2.1. XSLT에 의한 스타일 편집

XSL 표준은 XSLT와 XSL-FO로 구분되는데, XSL-FO는 스타일 편집을 목적으로 하는 언어인 반면 XSLT는 변환을 위한 기능을 제공한다. 변환은 XML 문서를 입력으로 받아 그것을 다양한 형태의 데이터로 바꾸어 줄 수 있는데, 현재 많이 사용되는 예가 HTML로의 변환이다. 그 결과 XSLT는 HTML 문서를 생성함으로써 그 자체로 스타일 기능까지 결합할 수 있다.

본 편집기에서는 문단 요소로 <p>, <ul>, <ol>, <li>, <div> 태그를 사용하고 인라인 요소로 <b>, <i>, <u>, <font> 등의 태그를 사용하며 이외에 <table>과 <img> 태그를 사용하고 있다. 이들 태그는 스타일 편집기의 결과인 XSL 파일에 포함되는 데, 스타일 편집기는 이들 태그에 의해 표현 가능한 스타일을 지원한다. 또한 CSS 기반의 스타일 속성을 사용하여 각 편집 객체의 속성을 지정한다.

##### 3.2.2. 데이터 선택 기능

XSL 스타일 편집기에서는 요소의 입력이 XML 문서에 있는 데이터에 대해서도 가능해야 하는데, 즉 그림 2의 결재 문서에서 제목이나 월, 일과 같은 부분은 스타일에 포함되고 2001, 1 등의 값은 XML 문서의 데이터이다. 아래 소스는 그림 2 XSL 문서의 일부를 보여준다.

```
<p align="center">
2001 년
<xsl:value-of select="/>교통비/일자/월"/> 월
<xsl:value-of select="/>교통비/일자/일"/> 일
</p>
```

[XSL 소스—날짜 부분]

여기서 <xsl:value-of/> 태그에 의해 XML 문서 상의 해당 데

이터 부분을 지정하는데, 이 때 주어진 select 속성의 경로가 선택할 데이터를 지정하게 된다. 이 경로는 절대 경로로 주어질 수도 있고(그림 4의 예) 특정 위치에 대한 상대 경로로 주어질 수도 있다. 상대 경로가 사용되는 경우는 XSL에서 <xsl:template>나 <xsl:for-each> 등의 매칭 기능을 이용하여 XML의 데이터 부분을 선택한 경우이다. 이 때 스타일 부분은 선택된 부분 트리 안에서 해당 노드에 대해 모두 적용될 수 있다. 아래 절에서 살펴보겠지만 이러한 기능은 XML 데이터의 여러 부분에 스타일을 반복 적용할 수 있게 해 준다.

XML 데이터 부분은 XSL 상에서는 구분되어 있지만 편집기에서는 그냥 텍스트 부분으로 보여진다. 그러나 XML 데이터는 텍스트의 수정이 되지 않고 삽입 또는 삭제만 가능하며 또 일부의 속성을 바꿀 수도 없어 한 단위로만 취급할 수 있다. 이렇게 편집 중에 현재 위치가 XML 데이터 상인지 XSL에 속하는 텍스트 부분인지를 사용자가 볼 수 있어야 한다. 본 시스템에서는 툴바의 버튼과 팝업 메뉴의 비활성화를 통해 이를 구분한다.

XML 데이터 부분을 선택하기 위하여 본 시스템에서는 아래 그림 4와 같은 대화상자를 제공하여, 해당 데이터를 선택하여 끌어놓기(drag-and-drop) 할 수 있도록 지원한다. 대화상자의 트리에서 특정 부분을 선택하여 편집창의 원하는 위치에 삽입한다. 동일한 방식으로 DTD 창에서 데이터를 선택하는 것도 가능하다.

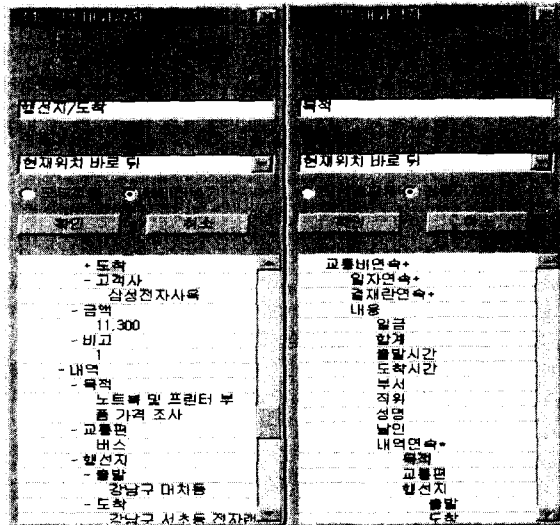


그림 4 데이터 선택 대화상자

### 3.2.3. 스타일의 반복 적용

XSL 기반 스타일 편집기의 가장 큰 특징은, 특정 스타일이 <xsl:for-each> 또는 <xsl:template> 태그에 의해 XML의 여러 부분에 적용될 수 있다는 점이다. 또 반복 구조가 중첩될 수 있어 XML 데이터의 구조에 탄력적으로 적용 가능하다. 아래 소스는 그림 2의 결재란이 이러한 XSL의 반복구조를 가짐을 보여준다.

```
<table width="35%" border="1" cellpadding="2" cellspacing="2" bgcolor="#ffffff">
  <tr height="12">
    <td rowspan="2"></td>
    <xsl:for-each select="/>교통비/결재란/경리부/결재자">
      <td><xsl:value-of select="직위"/></td>
    </xsl:for-each>
  </tr>
```

[XSL 소스—결재란 부분]

위와 같이 대상 데이터에 대해 스타일이 지정되면, XML 데이

터의 종류와 개수에 따라 결과 문서는 다른 형태가 된다. 위 소스에서 결재자가 몇 명이 있는가에 따라 결재란 표의 생성은 달라지게 된다. 이것은 기존의 편집방식에서 일정 크기의 결재란을 미리 만들어두는 문서 양식과 차이가 있다. 또한 반복 안에 다시 반복이 들어가는 것도 가능한데, 문서 편집 시에 현재 위치가 상대 위치로 XML 트리의 어느 부분에 있는가, 그리고 어느 부분에 대해 반복 중인가를 사용자에게 알려주는 것이 도움이 될 것이다.

### 3.3. XML 데이터를 이용한 스타일 편집 기능

본 시스템은 XML 데이터를 이용하여 스타일을 편집하는 기능을 제공한다. XML 데이터를 이용하여 편집하는 것은 DTD를 이용하는 것에 비하여 좀더 직관적인 편집 인터페이스를 제공한다.

그러나 XML 기반의 편집에서는 특정 XML 문서 인스턴스가 DTD의 문서 구조를 표현하지 못하므로 생략 또는 반복된 부분이 실제 데이터에서는 아예 나타나지 않을 수도 있다. 이러한 점을 보완하기 위해 본 시스템에서는 그림 4의 우측과 같이 DTD 구조를 추가로 보여준다. 이후에 문서 구조를 XML 트리 상에서 표현하도록 확장할 예정이다. 그림 5는 DTD 기반의 문서 편집 기능을 보여준다.

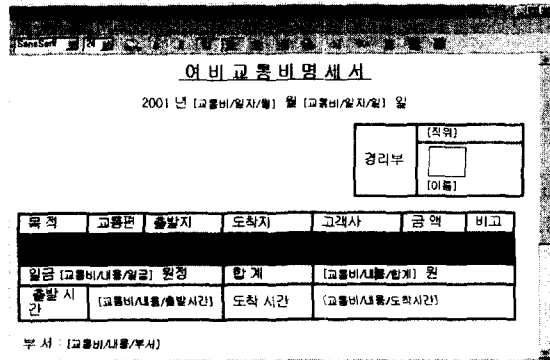


그림 5 DTD 기반의 문서 편집 화면

## 4. 구현 결과 및 결론

본 논문에서는 XSL 스타일 편집기의 개발을 소개하였다. 본 시스템은 자바(JDK 1.1.7) 언어[7]로 개발되었으며 그룹웨어 시스템에 활용될 수 있는 도구로서 시범적으로 개발되었다.

본 논문에서는 XSL 기반의 스타일 편집기에서 가질 수 있는 기능과 XML 문서 인스턴스를 이용한 스타일 편집의 방식을 실현적으로 구현하여 차세대 XML 기반 편집기에서 가져야 할 기능에 대해 살펴보았다. 본 논문에서 제기된 다양한 가능성은 이미 제품화되어 출시된 예도 있으나 아직은 많은 시도가 계속되고 있다[5,6,8].

### 참고문헌

- [1] 이강찬, 손홍, 박기식, "XML 표준화 동향," 정보과학회지, 19권 1호, pp.6-14, 2001년 1월.
- [2] Steven Holzner, XML Inside, 디지털 북스, 2001.
- [3] 이은정의 XML 홈페이지, <http://www.kyonggi.ac.kr/~ejlee/xml>.
- [4] W3C 홈페이지, <http://www.w3.org/>.
- [5] 휴먼컴, <http://www.human.co.kr/>.
- [6] 창해소프트, <http://www.css.co.kr/>.
- [7] The Source for Java™ Technology, <http://java.sun.com/>.
- [8] 이은정, "XML 기반 결재문서 처리기 개발," 인터넷 정보학회 2001년 봄 학술발표대회, 2001년 5월.