E-commerce를 위한 XML 문서 변환 도구 개발

최정익, 정재우, 하상호
순천향대학교 정보기술공학부

Development of A Tool for Transforming XML Document for E-commerce

Jungik Choi, Jeawoo Jung, Sangho Ha
Dept. of Information Technology, Soonchunhyang University
E-mail: timing96@hotmail.com, jeawoo76@hotmail.net, hsh@sch.ac.kr

요 약

현재 전자상거래에서 정보전달에 쓰이고 있는 HTML이나 EDI는 정보의 전달과 효율방면에는 별로 유용하지가 않다. 그 대처 방안으로 정보전달에 매우 유용하고 효율적인 XML이 대두되고 있다. 그러나 XML문서는 정보제공자 마다 그 구성과 태그이름이 다르다. 이러한 타 XML문서를 사용하기 위해서는 정보제공자와 같은 구조의 XML을 사용하거나 XML문서의 변환이 불가피하다. XML은 XSL을 사용하여 변환이 가능하지만 구조와 요소이름이 파악되지 않은 XML문서의 변환은 그 문서를 분석뿐만 아니라 변환시킬 수 있는 XSL을 직접 제작 해줘야 한다. 이와 같은 작업은 전자 상거래에서 저해요인이 될 뿐 아니라 시간적으로도 크나큰 손실이다. 본 논문에서는 Java를 기반으로 사용자의 편의적인 GUI(Graphical Use Interface)에 입각한 XSL문서 제작 도구의 구현 및 향후 개발 방향을 제시하였다.

1. 서론

웹(web)관련 기술의 발전과 대중화로 인하여, 복잡해진 지식 체계에 따른 다양해진 정보를 손쉽게 구할 수 있게 되었다. 이런 웹의 비약적인 발달은 상거래에도 영향을 미쳐 기존의 상거래의 형태를 벗어나 웹을 이용한 상거래가 증가하고 있다. 또한 기업 간의 거래도 웹상에서 이루어지기에 이르렀다. 기존의 기업 간의 거래에서 메시지 교환에서는 EDI(전자 자료 교환, Electronic Data Interchange)[1]방법이 주를 이루고 있었다. 그러나 전문가가 아닌 이상 EDI문서를 이해하기란 쉬운 일이 아니며 이 EDI메시지들을 처리하는 프로그램을 개발하고 유지보수 하는데 많은 비용이 소요된다. 또한 EDI메시지의 교환에 사용되던 부가가치망의 정보전달 속도를 인터넷의 정보전달 속도가 훨씬 뛰어 넘고 있다. 이런 이유 때문에 기업 간의 거래 역시 인터넷을 이용하는 쪽으로 눈을 돌리고 있다.[1,2] 그러나 웹 기반 언어인 HTML(Hypertext Markup Language)[3]로 정보를 전달 및 교환하려면 어려움이 많다. HTML은 태그가 고정되어 있기 때문에 HTML로 이루어진 정보는 태그만으로는 정보를 분석할 수 없고 집적 내용을 읽어 분석하는 방법밖에는 존재하질 않는다. 이에 비하여 XML(eXtensible Markup Language)은 HTML과는 다르게 태그가 고정되어 있지 않고 자신만의 태그 집합을 정의하고 사용할 수 있다. 때문에 태그만으로도 정보를 판단할 수 있다. 그러나 이와 같은 사용을 위해서는 두 기업 간의 같은 형태의 태그를 가지는 XML문서를 사용해야 되는 조건이 있다. 만약 XML문서의 형태를 바꿀 수 없는 입장이라면 XML문서를 변환할 수 있는

XSL(eXtensible Stylesheet Language)[4]문서를 사용하여 자신의 XML문서의 구조로 변환할 수 있다. 그러나 XSL문서의 작성은 매우 어렵고 시간이 많이 걸리는 일로서 두 XML문서의 구조를 잘 알고 있어야 한다. 그러나 웹 상의 수많은 다른 형태의 XML문서를 그때그때 XSL문서를 이용하여 변환한다는 것은 매우 불편하고도 비효율적인 일이 된다. 또한 지금까지의 XSL문서의 제작은 제작자가 XML문서의 구조를 파악하고 텍스트 상에서 XSL 구문을 사용하여 직접 타이핑하는 방식이었다. 이와 관련해서 XSL문서를 제작하여 주는 툴들이 많이 개발되기는 하였지만 텍스트 상에서의 작업을 벗어나지 못하고 제작자가 직접 타이핑을 해주어야하는 문제가 남아 있었다.

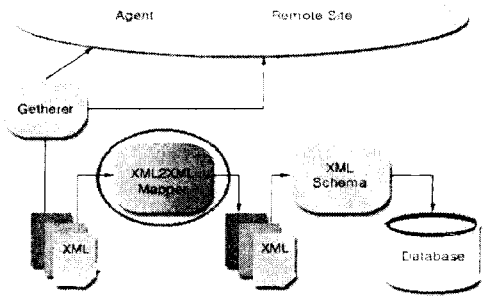
본 연구에서는 이런 문제를 해결하기 위하여 서로 다른 구조를 가지는 XML문서를 변환할 수 있는 XSL문서 제작을 손쉽게 해주는 도구를 개발한다.

본 연구는 Java를 기반으로 하여 사용자 편의적인 GUI(Graphical User Interface)에 입각한 XSL문서 제작 도구로서 XML문서의 변환을 좀더 빠르고 손쉽게 하기 위하여 변경하고자 하는 XML문서의 DTD[3]를 비주얼하게 트리 형태로 표현하고 비주얼하게 표현된 트리를 보면서 노드의 쌍을 사용자는 마우스로 클릭함으로써 그에 맞는 XSL문서를 제작해주는 변환 시스템을 개발한다.

2. XML문서 변환

본 연구는 E-commerce의 정보교환에서 그림 1과 같이 사용될 수 있다. 정보사용자는 XML문서로 이루어진 정보를 Getherer 모듈을 통하여 인터넷의 Agent나 Remote Site에서 XML문서를 수집한다. 그러나 이 XML문서의 정보를 사용하기 위해서는 자신의 XML문서의 구조에 맞게 변경해주어야 한다. 이때 본 도구를 사용하여 XML문서를 변환하기 위한 XSL문서를 제작한다. 본 도구를 사용하여 제작한 XSL문서를 사용하여 인터넷에서 수집한 XML문서를 자신의 XML문서 구조에 맞게 변형한다. 변형한 XML문서를 XML저장 모듈이 DataBase에 XML문서의 정보를 분리하여 저장한다.

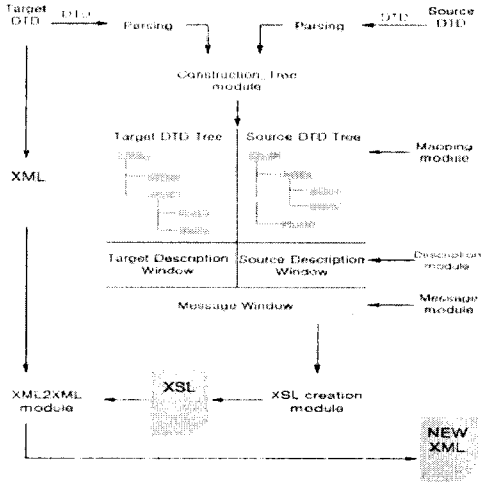
본 도구는 XSL문서를 제작할 때 문법을 모르더라도 비주얼하게 표현된 문서의 트리에서 변경하려는 Element쌍을 단순히 매치 시키기만 함으로써 손쉽게 XSL문서를 제작할 수 있다. 이 시스템은 Wutka Consulting. Inc.에서 개발한 DTD Parser Ver 1.3[10]을 사용하여 DTD를 파싱하고 DTD를 트리형태로 표



<그림 1. E commerce 정보교환 시스템>

현하기 위하여 Java의 동적 버튼 생성방법을 사용하였다.

본 도구는 그림 2의 방법으로 XSL문서와 XML문서를 생성하고 사용자에게 XML문서의 정보를 알려준다. 사용자는 변환 하고자 하는 DTD(Target)와 변환하려고 하는 DTD(Source)를 선택하면 Parsing 모듈이 파싱을 하여 문서의 유효성을 검사한다. 유효성 검사 후 유효한 문서라면 DTD문서의 정보를 트리를 구성하는 Construct 모듈에게 전송한다. Construct 모듈은 문서의 정보를 가지고 트리를 구성한다. 트리는 문서의 Element와 Attribute를 나타낸다. 사용자는 구성된 트리의 노드(버튼으로 이루어져있다.)를 선택하면 Description 모듈이 노드에 대한 정보(노드의 이름, 타입, 부모 Element의 이름, Cardinality, Declaration)를 Description 창에 보여준다. 사용자는 이 정보를



<그림 2. XML문서 변환 시스템>

보면서 DTD문서의 구조를 파악한다. 사용자가 Target과 Source의 두 Element를 선택하여 사상시키면 Mapping 모듈이 동작하여 두 Element의 이름을 해쉬테이블에 입력한다. 사상이 완료되면 Create XSL 모듈이 동작하여 사상정보를 가지고 XSL문서를 만들어 낸다. 사용자는 Target DTD의 구조를 따르는 XML 문서를 선택하면 Convert 모듈은 XML문서와 생성한 XSL문서를 가지고 사용자가 원하는 형태의 XML문서의 구조를 가지는 새로운 XML문서를 만들어 낸다.

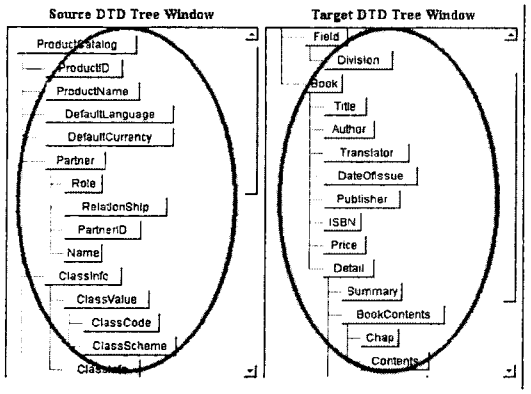
Parsing 모듈

DTD를 파싱하고 그 DTD가 유효하다면 각 Element마다 부모 Element와 Element의 Attribute, Element의 Cardinality(Node, ?, *, +), Attribute의 Declaration(REQUIRED, FIXED, IMPLIED)을 얻어내어 정보를 저장한다.

DTD 파싱은 Wutka Consulting, Inc.에서 개발한 DTD Parser Ver 1.3[10]을 사용하여 DTD를 파싱하였다. 이 파서는 문서의 정형성(Well-Formed)과 유효성(Validity)을 검사한다.

Construct Tree 모듈

DTD를 파싱한 후 정보를 가지고 트리를 구성한다. 트리는 자바의 동적 버튼 생성방법을 이용하여 DTD의 Element와 Attribute를 트리구조로 표현하고 버튼의 생성은 동적으로 하여 Element의 개수에 상관없이 버튼을 생성할 수 있도록 한다. 트리는 두 개의 창을 분리하여 Target DTD 문서와 Source DTD 문서를 보여준다. 이 모듈은 시스템의 GUI의 핵심으로서 DTD문서의 구조를 트리로 나타냄으로서 문서의 구조를 한눈에 파악할 수 있게 하여 사용자에게 좀더 편한 작업 환경을 만들어 준다.



<그림 3. 트리 윈도우 구성>

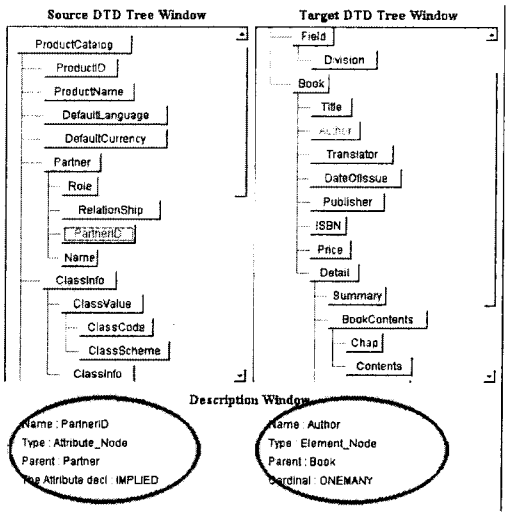
Description 모듈

생성된 DTD트리의 Element를 사용자가 선택하였을 때 그 Element의 정보를 Description창에 보여줌으로써 DTD문서의 구조와 정보를 학습할 수 있게 한다. 이 모듈로 인하여 사용자는 DTD의 구조를 파악할 때 텍스트 상태로 보면서 문서의 구조와 정보를 학습하는 기존의 방법에서 벗어나 GUI 환경에서 문서의 구조와 정보를 손쉽게 학습할 수 있다.

Description 창에 나타나는 정보는 표 1과 같다.

Element	Attribute
· Element의 이름 · 부모 Element의 이름 · Cardinality(None, ?, *, +)	· Attribute의 이름 · 부모 Element의 이름 · Declaration (REQUIRED, FIXED, IMPLIED)

<표 1. Description 창에 표현되는 정보>

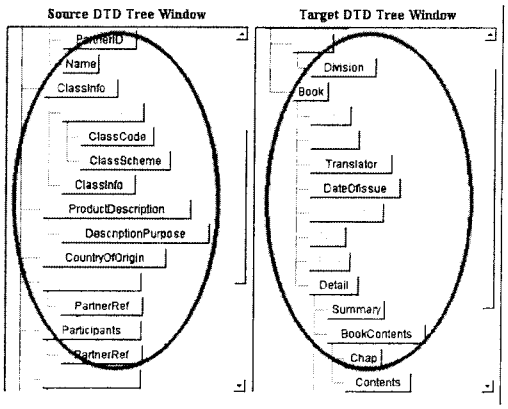


<그림 4. Description Window>

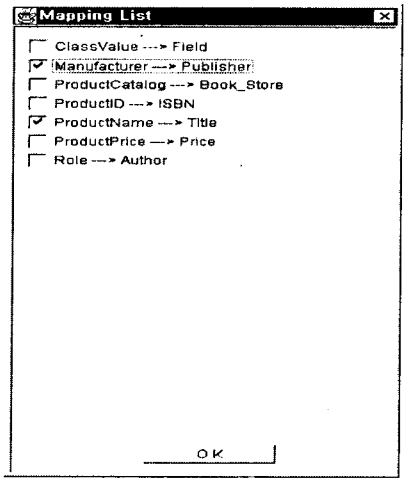
Mapping 모듈

사상이 될 대상 즉, 두 Element(Target DTD의 Element와 Source DTD의 Element)가 선택되었을 때 두 Element의 이름을 해쉬테이블에 넣게되고 사상이 된 버튼은 다시 사상이 되지 않게 사용할 수 없게 만든다. 사용자가 지금 까지의 사상정보를 보고자 하면 새로운 창에 모든 사상정보를 보여주고 그 중에서 취소하고 싶은 사상 쌍은 체크버튼을 클릭 함으로써 사상대상에서 삭제시킨다. 그리고 사용할 수 없게 만들어 놓은 버튼을 다시 사용할 수 있게 한다. 또한 일대일 사상이 아닌 다대일 사상의 경우 즉 두개 이상의 Target Element가 한 개의 Source Element에 사상이

될 때는 쉬프트 버튼을 누른 상태에서 Target DTD Element를 선택할 수 있게 지원한다. 그림 5는 사상이 진행되고 있는 과정을 나타내었고 그림 6은 사상 정보를 볼 수 있는 윈도우이다.



<그림 5. 두 트리 사상>



<그림 6. 사상 정보 윈도우>

Message Display 모듈

시스템이 진행되면서 일어나고 있는 상황이나 작업 도중에 사용자의 실수로 인하여 발생하는 에러를 사용자에게 알려주고 사용법을 사용자에게 알려주어 올바른 사용을 할 수 있게 인도한다.

XSL Creation 모듈

사상이 완료된 후 사상 정보를 가지고 XSL문서를 작성해 준다. Source DTD의 Element를 하나씩 읽어들여 사상정보에 현재 Element에 대한 사상 정보가 있는지 파악하고 사상정보가 존재한다면 사상종류에 따른 알맞은 XSL문을 작성하여 XSL문서에 추가한 후 사상이 완료되었으면 XSL문서를 사용자에게 보여주어 자신의 생성한 XSL문서를 확인할 수 있게 한다. 사상 종류에 따른 XSL문법은 표 2와 같다.

사상 종류	XSL 문법
Element - Element (Node1 - Node2)	<Node2> <xsl:value-of select="Node1"/> </Node2>
Element - Attribute (Node - Attr)	<Attr_Parent> <Attr_Parent Attr="(Element)"/> </Attr>
Attribute - Element (Attr - Node)	<Node> <xsl:value-of select="@Attr"/> </Node>
Attribute - Attribute (Attr1 - Attr2)	<Attr2_Parent> <Attr2_Parent Attr2="@Attr1"/> </Attr_Parent>

XML2XML 모듈

생성된 XSL문서와 Target DTD를 가지는 XML문서를 가지고 새로운 XML문서를 생성해준다. XSL 변환 엔진은 xml.apache.org project. 의 xalan for Java[11]를 사용하였다.

3. 결론 및 향후 연구과제

지금까지 본 논문에서는 DTD문서를 입력으로 하여 XML문서를 변환하기 위한 XSL문서 생성 시스템에 대하여 그에 따른 설계 및 구현을 기술한다. 전문가가 아니라면 매우 어렵고 많은 시간을 요하게 되는 XML문서를 변환하는 XSL문서를 GUI환경에서 손쉽게 생성할 수 있음으로 전자상거래에 발전에 많은 도움을 줄 수 있을 것으로 기대한다. 향후 연구로는 XSL문서 제작에 사용되는 XSL문법이 제한적으로 사용되던 것을 발전시켜 모든 XSL문법을 사용할 수 있게 하고 XML문서의 변환을 사용자가 사상 대상을 선택해 주지 않고 시스템에서 자동으로 의미를 파악하여 사상을 시키고 XSL문서를 생성시킬 수 있도록 연구가 계속되어야 할 것이다.

[참고 문헌]

[1] Hiroshi Maruyama, Kent Tamura, Naohiko Uramoto : XML and Java: Developing Web Applications, Addison-Wesley Pub Co, 1999
[2] Benoit Marshal : Applied XML Solutions (Sams Professional Publishing), Sams, 2000
[3] Simon St. Laurent, Ethan Cerami : Building XML Applications, McGraw-Hill Professional Publishing, 1999
[4] Khun Yee Fung : XSLT Working with XML and HTML, Addison-Wesley Pub Co, 2000
[5] Cay S. Horstmann, Gary Cornell : Core Java 2, Volume 1 Fundamentals, Sunsoft Press Inc, 1999
[6] Cay S. Horstmann, Gary Cornell : Core Java 2, Volume 2: Advanced Features, Sunsoft Press Inc, 1999
[7] Michael H. Kay : XSLT Programmer's Reference 2nd Edition, Wrox Press Inc, 2001
[8] Elliotte Rusty Harold, Elliote Rusty Harold : XML Bible 2nd Edition, Hungry Minds, Inc, June 2001
[9]Frank Boumphrey외 11명, XML Applications, Wrox Press Inc, 1999
[10]DTD Parser Ver 1.3 : Wutka Consulting. Inc. http://www.wutka.com
[11]xml.apache.org project. : xalan for Java http://xml.apache.org